

## STM8 Device Headers

Generated by Doxygen 1.8.13



# Contents

<b>1</b>	<b>Main Page</b>	<b>1</b>
1.1	Device Overview . . . . .	1
1.2	Toolchains . . . . .	1
1.3	Open-Source Headers . . . . .	1
1.3.1	Content . . . . .	2
1.3.2	Features . . . . .	2
<b>2</b>	<b>Module Index</b>	<b>3</b>
2.1	Modules . . . . .	3
<b>3</b>	<b>Data Structure Index</b>	<b>5</b>
3.1	Data Structures . . . . .	5
<b>4</b>	<b>File Index</b>	<b>7</b>
4.1	File List . . . . .	7
<b>5</b>	<b>Module Documentation</b>	<b>9</b>
5.1	STM8AF_STM8S . . . . .	9
5.1.1	Detailed Description . . . . .	125
5.1.2	Macro Definition Documentation . . . . .	125
5.1.2.1	__ADC1_VECTOR__ . . . . .	126
5.1.2.2	__AWU_VECTOR__ . . . . .	126
5.1.2.3	__CAN_RX_VECTOR__ . . . . .	126
5.1.2.4	__CAN_TX_VECTOR__ . . . . .	126
5.1.2.5	__CLK_VECTOR__ . . . . .	126

5.1.2.6	<a href="#">__FLASH_VECTOR__</a>	127
5.1.2.7	<a href="#">__I2C_VECTOR__</a>	127
5.1.2.8	<a href="#">__PORTA_VECTOR__</a>	127
5.1.2.9	<a href="#">__PORTB_VECTOR__</a>	127
5.1.2.10	<a href="#">__PORTC_VECTOR__</a>	127
5.1.2.11	<a href="#">__PORTD_VECTOR__</a>	127
5.1.2.12	<a href="#">__PORTE_VECTOR__</a>	128
5.1.2.13	<a href="#">__PORTF_VECTOR__</a>	128
5.1.2.14	<a href="#">__SPI_VECTOR__</a>	128
5.1.2.15	<a href="#">__TIM1_CAPCOM_VECTOR__</a>	128
5.1.2.16	<a href="#">__TIM1_UPD_OVF_VECTOR__</a>	128
5.1.2.17	<a href="#">__TIM2_CAPCOM_VECTOR__</a>	128
5.1.2.18	<a href="#">__TIM2_UPD_OVF_VECTOR__</a>	129
5.1.2.19	<a href="#">__TIM3_CAPCOM_VECTOR__</a>	129
5.1.2.20	<a href="#">__TIM3_UPD_OVF_VECTOR__</a>	129
5.1.2.21	<a href="#">__TIM4_UPD_OVF_VECTOR__</a>	129
5.1.2.22	<a href="#">__TIM5_UPD_OVF_VECTOR__</a>	129
5.1.2.23	<a href="#">__TLI_VECTOR__</a>	130
5.1.2.24	<a href="#">__UART1_RXF_VECTOR__</a>	130
5.1.2.25	<a href="#">__UART1_TXE_VECTOR__</a>	130
5.1.2.26	<a href="#">__UART2_RXF_VECTOR__</a>	130
5.1.2.27	<a href="#">__UART2_TXE_VECTOR__</a>	130
5.1.2.28	<a href="#">_ADC1</a>	131
5.1.2.29	<a href="#">_ADC1_ADON</a>	131
5.1.2.30	<a href="#">_ADC1_ALIGN</a>	131
5.1.2.31	<a href="#">_ADC1_AWCRH</a>	131
5.1.2.32	<a href="#">_ADC1_AWCRH_RESET_VALUE</a>	131
5.1.2.33	<a href="#">_ADC1_AWCRL</a>	132
5.1.2.34	<a href="#">_ADC1_AWCRL_RESET_VALUE</a>	132
5.1.2.35	<a href="#">_ADC1_AWD</a>	132

5.1.2.36	<a href="#">_ADC1_AWDIE</a>	132
5.1.2.37	<a href="#">_ADC1_AWSRH</a>	132
5.1.2.38	<a href="#">_ADC1_AWSRL</a>	133
5.1.2.39	<a href="#">_ADC1_CH</a>	133
5.1.2.40	<a href="#">_ADC1_CH0</a>	133
5.1.2.41	<a href="#">_ADC1_CH1</a>	133
5.1.2.42	<a href="#">_ADC1_CH2</a>	133
5.1.2.43	<a href="#">_ADC1_CH3</a>	134
5.1.2.44	<a href="#">_ADC1_CONT</a>	134
5.1.2.45	<a href="#">_ADC1_CR1</a>	134
5.1.2.46	<a href="#">_ADC1_CR1_RESET_VALUE</a>	134
5.1.2.47	<a href="#">_ADC1_CR2</a>	134
5.1.2.48	<a href="#">_ADC1_CR2_RESET_VALUE</a>	135
5.1.2.49	<a href="#">_ADC1_CR3</a>	135
5.1.2.50	<a href="#">_ADC1_CR3_RESET_VALUE</a>	135
5.1.2.51	<a href="#">_ADC1_CSR</a>	135
5.1.2.52	<a href="#">_ADC1_CSR_RESET_VALUE</a>	135
5.1.2.53	<a href="#">_ADC1_DB0RH</a>	136
5.1.2.54	<a href="#">_ADC1_DB0RL</a>	136
5.1.2.55	<a href="#">_ADC1_DB1RH</a>	136
5.1.2.56	<a href="#">_ADC1_DB1RL</a>	136
5.1.2.57	<a href="#">_ADC1_DB2RH</a>	136
5.1.2.58	<a href="#">_ADC1_DB2RL</a>	137
5.1.2.59	<a href="#">_ADC1_DB3RH</a>	137
5.1.2.60	<a href="#">_ADC1_DB3RL</a>	137
5.1.2.61	<a href="#">_ADC1_DB4RH</a>	137
5.1.2.62	<a href="#">_ADC1_DB4RL</a>	137
5.1.2.63	<a href="#">_ADC1_DB5RH</a>	138
5.1.2.64	<a href="#">_ADC1_DB5RL</a>	138
5.1.2.65	<a href="#">_ADC1_DB6RH</a>	138

5.1.2.66	_ADC1_DB6RL	138
5.1.2.67	_ADC1_DB7RH	138
5.1.2.68	_ADC1_DB7RL	139
5.1.2.69	_ADC1_DB8RH	139
5.1.2.70	_ADC1_DB8RL	139
5.1.2.71	_ADC1_DB9RH	139
5.1.2.72	_ADC1_DB9RL	139
5.1.2.73	_ADC1_DBUF	140
5.1.2.74	_ADC1_DRH	140
5.1.2.75	_ADC1_DRL	140
5.1.2.76	_ADC1_EOC	140
5.1.2.77	_ADC1_EOCIE	140
5.1.2.78	_ADC1_EXTSEL	141
5.1.2.79	_ADC1_EXTSEL0	141
5.1.2.80	_ADC1_EXTSEL1	141
5.1.2.81	_ADC1_EXTTRIG	141
5.1.2.82	_ADC1_HTRH	141
5.1.2.83	_ADC1_HTRH_RESET_VALUE	142
5.1.2.84	_ADC1_HTRL	142
5.1.2.85	_ADC1_HTRL_RESET_VALUE	142
5.1.2.86	_ADC1_LTRH	142
5.1.2.87	_ADC1_LTRH_RESET_VALUE	142
5.1.2.88	_ADC1_LTRL	143
5.1.2.89	_ADC1_LTRL_RESET_VALUE	143
5.1.2.90	_ADC1_OVR	143
5.1.2.91	_ADC1_SCAN	143
5.1.2.92	_ADC1_SPSEL	143
5.1.2.93	_ADC1_SPSEL0	144
5.1.2.94	_ADC1_SPSEL1	144
5.1.2.95	_ADC1_SPSEL2	144

5.1.2.96	_ADC1_TDRH	144
5.1.2.97	_ADC1_TDRH_RESET_VALUE	144
5.1.2.98	_ADC1_TDRL	145
5.1.2.99	_ADC1_TDRL_RESET_VALUE	145
5.1.2.100	_ADC2	145
5.1.2.101	_ADC2_ADON	145
5.1.2.102	_ADC2_ALIGN	145
5.1.2.103	_ADC2_CH	146
5.1.2.104	_ADC2_CH0	146
5.1.2.105	_ADC2_CH1	146
5.1.2.106	_ADC2_CH2	146
5.1.2.107	_ADC2_CH3	146
5.1.2.108	_ADC2_CONT	147
5.1.2.109	_ADC2_CR1	147
5.1.2.110	_ADC2_CR1_RESET_VALUE	147
5.1.2.111	_ADC2_CR2	147
5.1.2.112	_ADC2_CR2_RESET_VALUE	147
5.1.2.113	_ADC2_CSR	148
5.1.2.114	_ADC2_CSR_RESET_VALUE	148
5.1.2.115	_ADC2_DRH	148
5.1.2.116	_ADC2_DRL	148
5.1.2.117	_ADC2_EOC	148
5.1.2.118	_ADC2_EOCIE	149
5.1.2.119	_ADC2_EXTSEL	149
5.1.2.120	_ADC2_EXTSEL0	149
5.1.2.121	_ADC2_EXTSEL1	149
5.1.2.122	_ADC2_EXTTRIG	149
5.1.2.123	_ADC2_SPSEL	150
5.1.2.124	_ADC2_SPSEL0	150
5.1.2.125	_ADC2_SPSEL1	150

5.1.2.126 _ADC2_SPSEL2 . . . . .	150
5.1.2.127 _ADC2_TDRH . . . . .	150
5.1.2.128 _ADC2_TDRH_RESET_VALUE . . . . .	151
5.1.2.129 _ADC2_TDRL . . . . .	151
5.1.2.130 _ADC2_TDRL_RESET_VALUE . . . . .	151
5.1.2.131 _AWU . . . . .	151
5.1.2.132 _AWU_APR . . . . .	151
5.1.2.133 _AWU_APR_RESET_VALUE . . . . .	152
5.1.2.134 _AWU_APRE . . . . .	152
5.1.2.135 _AWU_APRE0 . . . . .	152
5.1.2.136 _AWU_APRE1 . . . . .	152
5.1.2.137 _AWU_APRE2 . . . . .	152
5.1.2.138 _AWU_APRE3 . . . . .	153
5.1.2.139 _AWU_APRE4 . . . . .	153
5.1.2.140 _AWU_APRE5 . . . . .	153
5.1.2.141 _AWU_AWUEN . . . . .	153
5.1.2.142 _AWU_AWUF . . . . .	153
5.1.2.143 _AWU_AWUTB . . . . .	154
5.1.2.144 _AWU_AWUTB0 . . . . .	154
5.1.2.145 _AWU_AWUTB1 . . . . .	154
5.1.2.146 _AWU_AWUTB2 . . . . .	154
5.1.2.147 _AWU_AWUTB3 . . . . .	154
5.1.2.148 _AWU_CSR . . . . .	155
5.1.2.149 _AWU_CSR_RESET_VALUE . . . . .	155
5.1.2.150 _AWU_MSR . . . . .	155
5.1.2.151 _AWU_TBR . . . . .	155
5.1.2.152 _AWU_TBR_RESET_VALUE . . . . .	155
5.1.2.153 _BEEP . . . . .	156
5.1.2.154 _BEEP_BEEP_DIV . . . . .	156
5.1.2.155 _BEEP_BEEP_DIV0 . . . . .	156



5.1.2.156 _BEEP_BEEPDIV1 . . . . .	156
5.1.2.157 _BEEP_BEEPDIV2 . . . . .	156
5.1.2.158 _BEEP_BEEPDIV3 . . . . .	157
5.1.2.159 _BEEP_BEEPDIV4 . . . . .	157
5.1.2.160 _BEEP_BEEPEN . . . . .	157
5.1.2.161 _BEEP_BEEPSEL . . . . .	157
5.1.2.162 _BEEP_BEEPSEL0 . . . . .	157
5.1.2.163 _BEEP_BEEPSEL1 . . . . .	158
5.1.2.164 _BEEP_CSR . . . . .	158
5.1.2.165 _BEEP_CSR_RESET_VALUE . . . . .	158
5.1.2.166 _BITS . . . . .	158
5.1.2.167 _CAN . . . . .	158
5.1.2.168 _CAN_ABOM . . . . .	159
5.1.2.169 _CAN_ABRQ . . . . .	159
5.1.2.170 _CAN_ALST . . . . .	159
5.1.2.171 _CAN_AWUM . . . . .	159
5.1.2.172 _CAN_BOFF . . . . .	159
5.1.2.173 _CAN_BOFIE . . . . .	160
5.1.2.174 _CAN_BRP . . . . .	160
5.1.2.175 _CAN_BRP0 . . . . .	160
5.1.2.176 _CAN_BRP1 . . . . .	160
5.1.2.177 _CAN_BRP2 . . . . .	160
5.1.2.178 _CAN_BRP3 . . . . .	161
5.1.2.179 _CAN_BRP4 . . . . .	161
5.1.2.180 _CAN_BRP5 . . . . .	161
5.1.2.181 _CAN_BS1 . . . . .	161
5.1.2.182 _CAN_BS10 . . . . .	161
5.1.2.183 _CAN_BS11 . . . . .	162
5.1.2.184 _CAN_BS12 . . . . .	162
5.1.2.185 _CAN_BS13 . . . . .	162

5.1.2.186 _CAN_BS2 . . . . .	162
5.1.2.187 _CAN_BS20 . . . . .	162
5.1.2.188 _CAN_BS21 . . . . .	163
5.1.2.189 _CAN_BS22 . . . . .	163
5.1.2.190 _CAN_BTR1 . . . . .	163
5.1.2.191 _CAN_BTR1_RESET_VALUE . . . . .	163
5.1.2.192 _CAN_BTR2 . . . . .	163
5.1.2.193 _CAN_BTR2_RESET_VALUE . . . . .	164
5.1.2.194 _CAN_CODE . . . . .	164
5.1.2.195 _CAN_CODE0 . . . . .	164
5.1.2.196 _CAN_CODE1 . . . . .	164
5.1.2.197 _CAN_DGR . . . . .	164
5.1.2.198 _CAN_DGR_RESET_VALUE . . . . .	165
5.1.2.199 _CAN_DLC . . . . .	165
5.1.2.200 _CAN_DLC0 . . . . .	165
5.1.2.201 _CAN_DLC1 . . . . .	165
5.1.2.202 _CAN_DLC2 . . . . .	165
5.1.2.203 _CAN_DLC3 . . . . .	166
5.1.2.204 _CAN_EIER . . . . .	166
5.1.2.205 _CAN_EIER_RESET_VALUE . . . . .	166
5.1.2.206 _CAN_EPVF . . . . .	166
5.1.2.207 _CAN_EPVIE . . . . .	166
5.1.2.208 _CAN_ERRI . . . . .	167
5.1.2.209 _CAN_ERRIE . . . . .	167
5.1.2.210 _CAN_ESR . . . . .	167
5.1.2.211 _CAN_ESR_RESET_VALUE . . . . .	167
5.1.2.212 _CAN_EWGF . . . . .	167
5.1.2.213 _CAN_EWGIE . . . . .	168
5.1.2.214 _CAN_F0R1 . . . . .	168
5.1.2.215 _CAN_F0R2 . . . . .	168

5.1.2.216 _CAN_F0R3 . . . . .	168
5.1.2.217 _CAN_F0R4 . . . . .	168
5.1.2.218 _CAN_F0R5 . . . . .	169
5.1.2.219 _CAN_F0R6 . . . . .	169
5.1.2.220 _CAN_F0R7 . . . . .	169
5.1.2.221 _CAN_F0R8 . . . . .	169
5.1.2.222 _CAN_F1R1 . . . . .	169
5.1.2.223 _CAN_F1R2 . . . . .	170
5.1.2.224 _CAN_F1R3 . . . . .	170
5.1.2.225 _CAN_F1R4 . . . . .	170
5.1.2.226 _CAN_F1R5 . . . . .	170
5.1.2.227 _CAN_F1R6 . . . . .	170
5.1.2.228 _CAN_F1R7 . . . . .	171
5.1.2.229 _CAN_F1R8 . . . . .	171
5.1.2.230 _CAN_F2R1 . . . . .	171
5.1.2.231 _CAN_F2R2 . . . . .	171
5.1.2.232 _CAN_F2R3 . . . . .	171
5.1.2.233 _CAN_F2R4 . . . . .	172
5.1.2.234 _CAN_F2R5 . . . . .	172
5.1.2.235 _CAN_F2R6 . . . . .	172
5.1.2.236 _CAN_F2R7 . . . . .	172
5.1.2.237 _CAN_F2R8 . . . . .	172
5.1.2.238 _CAN_F3R1 . . . . .	173
5.1.2.239 _CAN_F3R2 . . . . .	173
5.1.2.240 _CAN_F3R3 . . . . .	173
5.1.2.241 _CAN_F3R4 . . . . .	173
5.1.2.242 _CAN_F3R5 . . . . .	173
5.1.2.243 _CAN_F3R6 . . . . .	174
5.1.2.244 _CAN_F3R7 . . . . .	174
5.1.2.245 _CAN_F3R8 . . . . .	174

5.1.2.246 _CAN_F4R1 . . . . .	174
5.1.2.247 _CAN_F4R2 . . . . .	174
5.1.2.248 _CAN_F4R3 . . . . .	175
5.1.2.249 _CAN_F4R4 . . . . .	175
5.1.2.250 _CAN_F4R5 . . . . .	175
5.1.2.251 _CAN_F4R6 . . . . .	175
5.1.2.252 _CAN_F4R7 . . . . .	175
5.1.2.253 _CAN_F4R8 . . . . .	176
5.1.2.254 _CAN_F5R1 . . . . .	176
5.1.2.255 _CAN_F5R2 . . . . .	176
5.1.2.256 _CAN_F5R3 . . . . .	176
5.1.2.257 _CAN_F5R4 . . . . .	176
5.1.2.258 _CAN_F5R5 . . . . .	177
5.1.2.259 _CAN_F5R6 . . . . .	177
5.1.2.260 _CAN_F5R7 . . . . .	177
5.1.2.261 _CAN_F5R8 . . . . .	177
5.1.2.262 _CAN_FACT0 . . . . .	177
5.1.2.263 _CAN_FACT1 . . . . .	178
5.1.2.264 _CAN_FACT2 . . . . .	178
5.1.2.265 _CAN_FACT3 . . . . .	178
5.1.2.266 _CAN_FACT4 . . . . .	178
5.1.2.267 _CAN_FACT5 . . . . .	178
5.1.2.268 _CAN_FCR1 . . . . .	179
5.1.2.269 _CAN_FCR2 . . . . .	179
5.1.2.270 _CAN_FCR3 . . . . .	179
5.1.2.271 _CAN_FCR_RESET_VALUE . . . . .	179
5.1.2.272 _CAN_FFIE . . . . .	179
5.1.2.273 _CAN_FMH0 . . . . .	180
5.1.2.274 _CAN_FMH1 . . . . .	180
5.1.2.275 _CAN_FMH2 . . . . .	180

5.1.2.276 _CAN_FMH3 . . . . .	180
5.1.2.277 _CAN_FMH4 . . . . .	180
5.1.2.278 _CAN_FMH5 . . . . .	181
5.1.2.279 _CAN_FML0 . . . . .	181
5.1.2.280 _CAN_FML1 . . . . .	181
5.1.2.281 _CAN_FML2 . . . . .	181
5.1.2.282 _CAN_FML3 . . . . .	181
5.1.2.283 _CAN_FML4 . . . . .	182
5.1.2.284 _CAN_FML5 . . . . .	182
5.1.2.285 _CAN_FMP . . . . .	182
5.1.2.286 _CAN_FMP0 . . . . .	182
5.1.2.287 _CAN_FMP1 . . . . .	182
5.1.2.288 _CAN_FMPIE . . . . .	183
5.1.2.289 _CAN_FMR1 . . . . .	183
5.1.2.290 _CAN_FMR1_RESET_VALUE . . . . .	183
5.1.2.291 _CAN_FMR2 . . . . .	183
5.1.2.292 _CAN_FMR2_RESET_VALUE . . . . .	183
5.1.2.293 _CAN_FOVIE . . . . .	184
5.1.2.294 _CAN_FOVR . . . . .	184
5.1.2.295 _CAN_FSC0 . . . . .	184
5.1.2.296 _CAN_FSC00 . . . . .	184
5.1.2.297 _CAN_FSC01 . . . . .	184
5.1.2.298 _CAN_FSC1 . . . . .	185
5.1.2.299 _CAN_FSC10 . . . . .	185
5.1.2.300 _CAN_FSC11 . . . . .	185
5.1.2.301 _CAN_FSC2 . . . . .	185
5.1.2.302 _CAN_FSC20 . . . . .	185
5.1.2.303 _CAN_FSC21 . . . . .	186
5.1.2.304 _CAN_FSC3 . . . . .	186
5.1.2.305 _CAN_FSC30 . . . . .	186

5.1.2.306 _CAN_FSC31 . . . . .	186
5.1.2.307 _CAN_FSC4 . . . . .	186
5.1.2.308 _CAN_FSC40 . . . . .	187
5.1.2.309 _CAN_FSC41 . . . . .	187
5.1.2.310 _CAN_FSC5 . . . . .	187
5.1.2.311 _CAN_FSC50 . . . . .	187
5.1.2.312 _CAN_FSC51 . . . . .	187
5.1.2.313 _CAN_FULL . . . . .	188
5.1.2.314 _CAN_IDE . . . . .	188
5.1.2.315 _CAN_IER . . . . .	188
5.1.2.316 _CAN_IER_RESET_VALUE . . . . .	188
5.1.2.317 _CAN_INAK . . . . .	188
5.1.2.318 _CAN_INRQ . . . . .	189
5.1.2.319 _CAN_LBKM . . . . .	189
5.1.2.320 _CAN_LEC . . . . .	189
5.1.2.321 _CAN_LEC0 . . . . .	189
5.1.2.322 _CAN_LEC1 . . . . .	189
5.1.2.323 _CAN_LEC2 . . . . .	190
5.1.2.324 _CAN_LECIE . . . . .	190
5.1.2.325 _CAN_LOW0 . . . . .	190
5.1.2.326 _CAN_LOW1 . . . . .	190
5.1.2.327 _CAN_LOW2 . . . . .	190
5.1.2.328 _CAN_MCR . . . . .	191
5.1.2.329 _CAN_MCR_RESET_VALUE . . . . .	191
5.1.2.330 _CAN_MCSR . . . . .	191
5.1.2.331 _CAN_MCSR_RESET_VALUE . . . . .	191
5.1.2.332 _CAN_MDAR1 . . . . .	191
5.1.2.333 _CAN_MDAR2 . . . . .	192
5.1.2.334 _CAN_MDAR3 . . . . .	192
5.1.2.335 _CAN_MDAR4 . . . . .	192

5.1.2.336 _CAN_MDAR5 . . . . .	192
5.1.2.337 _CAN_MDAR6 . . . . .	192
5.1.2.338 _CAN_MDAR7 . . . . .	193
5.1.2.339 _CAN_MDAR8 . . . . .	193
5.1.2.340 _CAN_MDLCR . . . . .	193
5.1.2.341 _CAN_MDLCR_RESET_VALUE . . . . .	193
5.1.2.342 _CAN_MFMIR . . . . .	193
5.1.2.343 _CAN_MFMIR_RESET_VALUE . . . . .	194
5.1.2.344 _CAN_MIDR1 . . . . .	194
5.1.2.345 _CAN_MIDR2 . . . . .	194
5.1.2.346 _CAN_MIDR3 . . . . .	194
5.1.2.347 _CAN_MIDR4 . . . . .	194
5.1.2.348 _CAN_MSR . . . . .	195
5.1.2.349 _CAN_MSR_RESET_VALUE . . . . .	195
5.1.2.350 _CAN_MTSRH . . . . .	195
5.1.2.351 _CAN_MTSRL . . . . .	195
5.1.2.352 _CAN_NART . . . . .	195
5.1.2.353 _CAN_PS . . . . .	196
5.1.2.354 _CAN_PS0 . . . . .	196
5.1.2.355 _CAN_PS1 . . . . .	196
5.1.2.356 _CAN_PS2 . . . . .	196
5.1.2.357 _CAN_PSR . . . . .	196
5.1.2.358 _CAN_PSR_RESET_VALUE . . . . .	197
5.1.2.359 _CAN_RECR . . . . .	197
5.1.2.360 _CAN_RECR_RESET_VALUE . . . . .	197
5.1.2.361 _CAN_RFLM . . . . .	197
5.1.2.362 _CAN_RFOM . . . . .	197
5.1.2.363 _CAN_RFR . . . . .	198
5.1.2.364 _CAN_RFR_RESET_VALUE . . . . .	198
5.1.2.365 _CAN_RQCP . . . . .	198

5.1.2.366 _CAN_RQCP0 . . . . .	198
5.1.2.367 _CAN_RQCP1 . . . . .	198
5.1.2.368 _CAN_RQCP2 . . . . .	199
5.1.2.369 _CAN_RTR . . . . .	199
5.1.2.370 _CAN_RX . . . . .	199
5.1.2.371 _CAN_RXS . . . . .	199
5.1.2.372 _CAN_SAMP . . . . .	199
5.1.2.373 _CAN_SILM . . . . .	200
5.1.2.374 _CAN_SJW . . . . .	200
5.1.2.375 _CAN_SJW0 . . . . .	200
5.1.2.376 _CAN_SJW1 . . . . .	200
5.1.2.377 _CAN_SLAK . . . . .	200
5.1.2.378 _CAN_SLEEP . . . . .	201
5.1.2.379 _CAN_TECR . . . . .	201
5.1.2.380 _CAN_TECR_RESET_VALUE . . . . .	201
5.1.2.381 _CAN_TERR . . . . .	201
5.1.2.382 _CAN_TGT . . . . .	201
5.1.2.383 _CAN_TME0 . . . . .	202
5.1.2.384 _CAN_TME1 . . . . .	202
5.1.2.385 _CAN_TME2 . . . . .	202
5.1.2.386 _CAN_TMEIE . . . . .	202
5.1.2.387 _CAN_TPR . . . . .	202
5.1.2.388 _CAN_TPR_RESET_VALUE . . . . .	203
5.1.2.389 _CAN_TSR . . . . .	203
5.1.2.390 _CAN_TSR_RESET_VALUE . . . . .	203
5.1.2.391 _CAN_TTCM . . . . .	203
5.1.2.392 _CAN_TX . . . . .	203
5.1.2.393 _CAN_TXFP . . . . .	204
5.1.2.394 _CAN_TXM2E . . . . .	204
5.1.2.395 _CAN_TXOK . . . . .	204



5.1.2.396 _CAN_TXOK0 . . . . .	204
5.1.2.397 _CAN_TXOK1 . . . . .	204
5.1.2.398 _CAN_TXOK2 . . . . .	205
5.1.2.399 _CAN_TXRQ . . . . .	205
5.1.2.400 _CAN_WKUI . . . . .	205
5.1.2.401 _CAN_WKUIE . . . . .	205
5.1.2.402 _CFG . . . . .	205
5.1.2.403 _CFG_AL . . . . .	206
5.1.2.404 _CFG_GCR . . . . .	206
5.1.2.405 _CFG_GCR_RESET_VALUE . . . . .	206
5.1.2.406 _CFG_SWD . . . . .	206
5.1.2.407 _CLK . . . . .	206
5.1.2.408 _CLK_ADC . . . . .	207
5.1.2.409 _CLK_AUX . . . . .	207
5.1.2.410 _CLK_AWU . . . . .	207
5.1.2.411 _CLK_CAN . . . . .	207
5.1.2.412 _CLK_CCOBSY . . . . .	207
5.1.2.413 _CLK_CCOEN . . . . .	208
5.1.2.414 _CLK_CCOR . . . . .	208
5.1.2.415 _CLK_CCOR_RESET_VALUE . . . . .	208
5.1.2.416 _CLK_CCORDY . . . . .	208
5.1.2.417 _CLK_CCOSEL . . . . .	208
5.1.2.418 _CLK_CCOSEL0 . . . . .	209
5.1.2.419 _CLK_CCOSEL1 . . . . .	209
5.1.2.420 _CLK_CCOSEL2 . . . . .	209
5.1.2.421 _CLK_CCOSEL3 . . . . .	209
5.1.2.422 _CLK_CKDIVR . . . . .	209
5.1.2.423 _CLK_CKDIVR_RESET_VALUE . . . . .	210
5.1.2.424 _CLK_CMSR . . . . .	210
5.1.2.425 _CLK_CMSR_RESET_VALUE . . . . .	210

5.1.2.426 _CLK_CPUDIV . . . . .	210
5.1.2.427 _CLK_CPUDIV0 . . . . .	210
5.1.2.428 _CLK_CPUDIV1 . . . . .	211
5.1.2.429 _CLK_CPUDIV2 . . . . .	211
5.1.2.430 _CLK_CSSD . . . . .	211
5.1.2.431 _CLK_CSSDIE . . . . .	211
5.1.2.432 _CLK_CSSEN . . . . .	211
5.1.2.433 _CLK_CSSR . . . . .	212
5.1.2.434 _CLK_CSSR_RESET_VALUE . . . . .	212
5.1.2.435 _CLK_ECKR . . . . .	212
5.1.2.436 _CLK_ECKR_HSERDY . . . . .	212
5.1.2.437 _CLK_ECKR_RESET_VALUE . . . . .	212
5.1.2.438 _CLK_FHWU . . . . .	213
5.1.2.439 _CLK_HSEEN . . . . .	213
5.1.2.440 _CLK_HSIDIV . . . . .	213
5.1.2.441 _CLK_HSIDIV0 . . . . .	213
5.1.2.442 _CLK_HSIDIV1 . . . . .	213
5.1.2.443 _CLK_HSIEN . . . . .	214
5.1.2.444 _CLK_HSIRDY . . . . .	214
5.1.2.445 _CLK_HSITRIM . . . . .	214
5.1.2.446 _CLK_HSITRIM0 . . . . .	214
5.1.2.447 _CLK_HSITRIM1 . . . . .	214
5.1.2.448 _CLK_HSITRIM2 . . . . .	215
5.1.2.449 _CLK_HSITRIM3 . . . . .	215
5.1.2.450 _CLK_HSITRIMR . . . . .	215
5.1.2.451 _CLK_HSITRIMR_RESET_VALUE . . . . .	215
5.1.2.452 _CLK_I2C . . . . .	215
5.1.2.453 _CLK_ICKR . . . . .	216
5.1.2.454 _CLK_ICKR_RESET_VALUE . . . . .	216
5.1.2.455 _CLK_LSIEN . . . . .	216

5.1.2.456 _CLK_LSIRDY . . . . .	216
5.1.2.457 _CLK_PCKENR1 . . . . .	216
5.1.2.458 _CLK_PCKENR1_RESET_VALUE . . . . .	217
5.1.2.459 _CLK_PCKENR2 . . . . .	217
5.1.2.460 _CLK_PCKENR2_RESET_VALUE . . . . .	217
5.1.2.461 _CLK_REGAH . . . . .	217
5.1.2.462 _CLK_SPI . . . . .	217
5.1.2.463 _CLK_SWBSY . . . . .	218
5.1.2.464 _CLK_SWCR . . . . .	218
5.1.2.465 _CLK_SWCR_RESET_VALUE . . . . .	218
5.1.2.466 _CLK_SWEN . . . . .	218
5.1.2.467 _CLK_SWI_HSE . . . . .	218
5.1.2.468 _CLK_SWI_HSI . . . . .	219
5.1.2.469 _CLK_SWI_LSI . . . . .	219
5.1.2.470 _CLK_SWIEN . . . . .	219
5.1.2.471 _CLK_SWIF . . . . .	219
5.1.2.472 _CLK_SWIMCCR . . . . .	219
5.1.2.473 _CLK_SWIMCCR_RESET_VALUE . . . . .	220
5.1.2.474 _CLK_SWIMCLK . . . . .	220
5.1.2.475 _CLK_SWR . . . . .	220
5.1.2.476 _CLK_SWR_RESET_VALUE . . . . .	220
5.1.2.477 _CLK_TIM1 . . . . .	220
5.1.2.478 _CLK_TIM2_TIM5 . . . . .	221
5.1.2.479 _CLK_TIM3 . . . . .	221
5.1.2.480 _CLK_TIM4_TIM6 . . . . .	221
5.1.2.481 _CLK_UART1 . . . . .	221
5.1.2.482 _CLK_UART2 . . . . .	221
5.1.2.483 _EXTI . . . . .	222
5.1.2.484 _EXTI_CR1 . . . . .	222
5.1.2.485 _EXTI_CR1_RESET_VALUE . . . . .	222

5.1.2.486 _EXTI_CR2 . . . . .	222
5.1.2.487 _EXTI_CR2_RESET_VALUE . . . . .	222
5.1.2.488 _EXTI_PAIS . . . . .	223
5.1.2.489 _EXTI_PAIS0 . . . . .	223
5.1.2.490 _EXTI_PAIS1 . . . . .	223
5.1.2.491 _EXTI_PBS . . . . .	223
5.1.2.492 _EXTI_PBS0 . . . . .	223
5.1.2.493 _EXTI_PBS1 . . . . .	224
5.1.2.494 _EXTI_PCIS . . . . .	224
5.1.2.495 _EXTI_PCIS0 . . . . .	224
5.1.2.496 _EXTI_PCIS1 . . . . .	224
5.1.2.497 _EXTI_PDIS . . . . .	224
5.1.2.498 _EXTI_PDIS0 . . . . .	225
5.1.2.499 _EXTI_PDIS1 . . . . .	225
5.1.2.500 _EXTI_PEIS . . . . .	225
5.1.2.501 _EXTI_PEIS0 . . . . .	225
5.1.2.502 _EXTI_PEIS1 . . . . .	225
5.1.2.503 _EXTI_TLIS . . . . .	226
5.1.2.504 _FLASH . . . . .	226
5.1.2.505 _FLASH_AHALT . . . . .	226
5.1.2.506 _FLASH_CR1 . . . . .	226
5.1.2.507 _FLASH_CR1_RESET_VALUE . . . . .	226
5.1.2.508 _FLASH_CR2 . . . . .	227
5.1.2.509 _FLASH_CR2_RESET_VALUE . . . . .	227
5.1.2.510 _FLASH_DUKR . . . . .	227
5.1.2.511 _FLASH_DUKR_RESET_VALUE . . . . .	227
5.1.2.512 _FLASH_DUL . . . . .	227
5.1.2.513 _FLASH_EOP . . . . .	228
5.1.2.514 _FLASH_ERASE . . . . .	228
5.1.2.515 _FLASH_FIX . . . . .	228

5.1.2.516 _FLASH_FPR . . . . .	228
5.1.2.517 _FLASH_FPRG . . . . .	228
5.1.2.518 _FLASH_HALT . . . . .	229
5.1.2.519 _FLASH_HVOFF . . . . .	229
5.1.2.520 _FLASH_IAPSR . . . . .	229
5.1.2.521 _FLASH_IAPSR_RESET_VALUE . . . . .	229
5.1.2.522 _FLASH_IE . . . . .	229
5.1.2.523 _FLASH_NCR2 . . . . .	230
5.1.2.524 _FLASH_NCR2_RESET_VALUE . . . . .	230
5.1.2.525 _FLASH_NFPR . . . . .	230
5.1.2.526 _FLASH_OPT . . . . .	230
5.1.2.527 _FLASH_PRG . . . . .	230
5.1.2.528 _FLASH_PUKR . . . . .	231
5.1.2.529 _FLASH_PUKR_RESET_VALUE . . . . .	231
5.1.2.530 _FLASH_PUL . . . . .	231
5.1.2.531 _FLASH_WPB . . . . .	231
5.1.2.532 _FLASH_WPB0 . . . . .	231
5.1.2.533 _FLASH_WPB1 . . . . .	232
5.1.2.534 _FLASH_WPB2 . . . . .	232
5.1.2.535 _FLASH_WPB3 . . . . .	232
5.1.2.536 _FLASH_WPB4 . . . . .	232
5.1.2.537 _FLASH_WPB5 . . . . .	232
5.1.2.538 _FLASH_WPRG . . . . .	233
5.1.2.539 _FLASH_WR_PG_DIS . . . . .	233
5.1.2.540 _GPIO_CR1_RESET_VALUE . . . . .	233
5.1.2.541 _GPIO_CR2_RESET_VALUE . . . . .	233
5.1.2.542 _GPIO_DDR_RESET_VALUE . . . . .	233
5.1.2.543 _GPIO_ODR_RESET_VALUE . . . . .	234
5.1.2.544 _GPIO_PIN0 . . . . .	234
5.1.2.545 _GPIO_PIN1 . . . . .	234

5.1.2.546 _GPIO_PIN2 . . . . .	234
5.1.2.547 _GPIO_PIN3 . . . . .	234
5.1.2.548 _GPIO_PIN4 . . . . .	235
5.1.2.549 _GPIO_PIN5 . . . . .	235
5.1.2.550 _GPIO_PIN6 . . . . .	235
5.1.2.551 _GPIO_PIN7 . . . . .	235
5.1.2.552 _GPIOA . . . . .	235
5.1.2.553 _GPIOA_CR1 . . . . .	236
5.1.2.554 _GPIOA_CR2 . . . . .	236
5.1.2.555 _GPIOA_DDR . . . . .	236
5.1.2.556 _GPIOA_IDR . . . . .	236
5.1.2.557 _GPIOA_ODR . . . . .	236
5.1.2.558 _GPIOB . . . . .	237
5.1.2.559 _GPIOB_CR1 . . . . .	237
5.1.2.560 _GPIOB_CR2 . . . . .	237
5.1.2.561 _GPIOB_DDR . . . . .	237
5.1.2.562 _GPIOB_IDR . . . . .	237
5.1.2.563 _GPIOB_ODR . . . . .	238
5.1.2.564 _GPIOC . . . . .	238
5.1.2.565 _GPIOC_CR1 . . . . .	238
5.1.2.566 _GPIOC_CR2 . . . . .	238
5.1.2.567 _GPIOC_DDR . . . . .	238
5.1.2.568 _GPIOC_IDR . . . . .	239
5.1.2.569 _GPIOC_ODR . . . . .	239
5.1.2.570 _GPIOD . . . . .	239
5.1.2.571 _GPIOD_CR1 . . . . .	239
5.1.2.572 _GPIOD_CR2 . . . . .	239
5.1.2.573 _GPIOD_DDR . . . . .	240
5.1.2.574 _GPIOD_IDR . . . . .	240
5.1.2.575 _GPIOD_ODR . . . . .	240

5.1.2.576 _GPIOE . . . . .	240
5.1.2.577 _GPIOE_CR1 . . . . .	240
5.1.2.578 _GPIOE_CR2 . . . . .	241
5.1.2.579 _GPIOE_DDR . . . . .	241
5.1.2.580 _GPIOE_IDR . . . . .	241
5.1.2.581 _GPIOE_ODR . . . . .	241
5.1.2.582 _GPIOF . . . . .	241
5.1.2.583 _GPIOF_CR1 . . . . .	242
5.1.2.584 _GPIOF_CR2 . . . . .	242
5.1.2.585 _GPIOF_DDR . . . . .	242
5.1.2.586 _GPIOF_IDR . . . . .	242
5.1.2.587 _GPIOF_ODR . . . . .	242
5.1.2.588 _GPIOG . . . . .	243
5.1.2.589 _GPIOG_CR1 . . . . .	243
5.1.2.590 _GPIOG_CR2 . . . . .	243
5.1.2.591 _GPIOG_DDR . . . . .	243
5.1.2.592 _GPIOG_IDR . . . . .	243
5.1.2.593 _GPIOG_ODR . . . . .	244
5.1.2.594 _GPIOH . . . . .	244
5.1.2.595 _GPIOH_CR1 . . . . .	244
5.1.2.596 _GPIOH_CR2 . . . . .	244
5.1.2.597 _GPIOH_DDR . . . . .	244
5.1.2.598 _GPIOH_IDR . . . . .	245
5.1.2.599 _GPIOH_ODR . . . . .	245
5.1.2.600 _GPIOI . . . . .	245
5.1.2.601 _GPIOI_CR1 . . . . .	245
5.1.2.602 _GPIOI_CR2 . . . . .	245
5.1.2.603 _GPIOI_DDR . . . . .	246
5.1.2.604 _GPIOI_IDR . . . . .	246
5.1.2.605 _GPIOI_ODR . . . . .	246

5.1.2.606 _I2C	246
5.1.2.607 _I2C_ACK	246
5.1.2.608 _I2C_ADD0	247
5.1.2.609 _I2C_ADD1	247
5.1.2.610 _I2C_ADD10	247
5.1.2.611 _I2C_ADD2	247
5.1.2.612 _I2C_ADD3	247
5.1.2.613 _I2C_ADD4	248
5.1.2.614 _I2C_ADD5	248
5.1.2.615 _I2C_ADD6	248
5.1.2.616 _I2C_ADD7	248
5.1.2.617 _I2C_ADD8	248
5.1.2.618 _I2C_ADD9	249
5.1.2.619 _I2C_ADD_8_9	249
5.1.2.620 _I2C_ADDCONF	249
5.1.2.621 _I2C_ADDMODE	249
5.1.2.622 _I2C_ADDR	249
5.1.2.623 _I2C_AF	250
5.1.2.624 _I2C_ARLO	250
5.1.2.625 _I2C_BERR	250
5.1.2.626 _I2C_BTf	250
5.1.2.627 _I2C_BUSY	250
5.1.2.628 _I2C_CCR	251
5.1.2.629 _I2C_CCR0	251
5.1.2.630 _I2C_CCR1	251
5.1.2.631 _I2C_CCR2	251
5.1.2.632 _I2C_CCR3	251
5.1.2.633 _I2C_CCRH	252
5.1.2.634 _I2C_CCRH_RESET_VALUE	252
5.1.2.635 _I2C_CCRL	252



5.1.2.636 _I2C_CCRL_RESET_VALUE . . . . .	252
5.1.2.637 _I2C_CR1 . . . . .	252
5.1.2.638 _I2C_CR1_RESET_VALUE . . . . .	253
5.1.2.639 _I2C_CR2 . . . . .	253
5.1.2.640 _I2C_CR2_RESET_VALUE . . . . .	253
5.1.2.641 _I2C_DR . . . . .	253
5.1.2.642 _I2C_DR_RESET_VALUE . . . . .	253
5.1.2.643 _I2C_DUTY . . . . .	254
5.1.2.644 _I2C_ENGC . . . . .	254
5.1.2.645 _I2C_FREQ . . . . .	254
5.1.2.646 _I2C_FREQ0 . . . . .	254
5.1.2.647 _I2C_FREQ1 . . . . .	254
5.1.2.648 _I2C_FREQ2 . . . . .	255
5.1.2.649 _I2C_FREQ3 . . . . .	255
5.1.2.650 _I2C_FREQ4 . . . . .	255
5.1.2.651 _I2C_FREQ5 . . . . .	255
5.1.2.652 _I2C_FREQR . . . . .	255
5.1.2.653 _I2C_FREQR_RESET_VALUE . . . . .	256
5.1.2.654 _I2C_FS . . . . .	256
5.1.2.655 _I2C_GENCALL . . . . .	256
5.1.2.656 _I2C_ITBUFEN . . . . .	256
5.1.2.657 _I2C_ITERREN . . . . .	256
5.1.2.658 _I2C_ITEVTEN . . . . .	257
5.1.2.659 _I2C_ITR . . . . .	257
5.1.2.660 _I2C_ITR_RESET_VALUE . . . . .	257
5.1.2.661 _I2C_MSL . . . . .	257
5.1.2.662 _I2C_NOSTRETCH . . . . .	257
5.1.2.663 _I2C_OARH . . . . .	258
5.1.2.664 _I2C_OARH_RESET_VALUE . . . . .	258
5.1.2.665 _I2C_OARL . . . . .	258

5.1.2.666 _I2C_OARL_RESET_VALUE . . . . .	258
5.1.2.667 _I2C_OVR . . . . .	258
5.1.2.668 _I2C_PE . . . . .	259
5.1.2.669 _I2C_POS . . . . .	259
5.1.2.670 _I2C_RXNE . . . . .	259
5.1.2.671 _I2C_SB . . . . .	259
5.1.2.672 _I2C_SR1 . . . . .	259
5.1.2.673 _I2C_SR1_RESET_VALUE . . . . .	260
5.1.2.674 _I2C_SR2 . . . . .	260
5.1.2.675 _I2C_SR2_RESET_VALUE . . . . .	260
5.1.2.676 _I2C_SR3 . . . . .	260
5.1.2.677 _I2C_SR3_RESET_VALUE . . . . .	260
5.1.2.678 _I2C_START . . . . .	261
5.1.2.679 _I2C_STOP . . . . .	261
5.1.2.680 _I2C_STOPF . . . . .	261
5.1.2.681 _I2C_SWRST . . . . .	261
5.1.2.682 _I2C_TRA . . . . .	261
5.1.2.683 _I2C_TRISE . . . . .	262
5.1.2.684 _I2C_TRISE0 . . . . .	262
5.1.2.685 _I2C_TRISE1 . . . . .	262
5.1.2.686 _I2C_TRISE2 . . . . .	262
5.1.2.687 _I2C_TRISE3 . . . . .	262
5.1.2.688 _I2C_TRISE4 . . . . .	263
5.1.2.689 _I2C_TRISE5 . . . . .	263
5.1.2.690 _I2C_TRISER . . . . .	263
5.1.2.691 _I2C_TRISER_RESET_VALUE . . . . .	263
5.1.2.692 _I2C_TXE . . . . .	263
5.1.2.693 _I2C_WUFH . . . . .	264
5.1.2.694 _ITC . . . . .	264
5.1.2.695 _ITC_SPR1 . . . . .	264

5.1.2.696 _ITC_SPR1_RESET_VALUE . . . . .	264
5.1.2.697 _ITC_SPR2 . . . . .	264
5.1.2.698 _ITC_SPR2_RESET_VALUE . . . . .	265
5.1.2.699 _ITC_SPR3 . . . . .	265
5.1.2.700 _ITC_SPR3_RESET_VALUE . . . . .	265
5.1.2.701 _ITC_SPR4 . . . . .	265
5.1.2.702 _ITC_SPR4_RESET_VALUE . . . . .	265
5.1.2.703 _ITC_SPR5 . . . . .	266
5.1.2.704 _ITC_SPR5_RESET_VALUE . . . . .	266
5.1.2.705 _ITC_SPR6 . . . . .	266
5.1.2.706 _ITC_SPR6_RESET_VALUE . . . . .	266
5.1.2.707 _ITC_SPR7 . . . . .	266
5.1.2.708 _ITC_SPR7_RESET_VALUE . . . . .	267
5.1.2.709 _ITC_SPR8 . . . . .	267
5.1.2.710 _ITC_SPR8_RESET_VALUE . . . . .	267
5.1.2.711 _ITC_VECT10SPR . . . . .	267
5.1.2.712 _ITC_VECT10SPR0 . . . . .	267
5.1.2.713 _ITC_VECT10SPR1 . . . . .	268
5.1.2.714 _ITC_VECT11SPR . . . . .	268
5.1.2.715 _ITC_VECT11SPR0 . . . . .	268
5.1.2.716 _ITC_VECT11SPR1 . . . . .	268
5.1.2.717 _ITC_VECT12SPR . . . . .	268
5.1.2.718 _ITC_VECT12SPR0 . . . . .	269
5.1.2.719 _ITC_VECT12SPR1 . . . . .	269
5.1.2.720 _ITC_VECT13SPR . . . . .	269
5.1.2.721 _ITC_VECT13SPR0 . . . . .	269
5.1.2.722 _ITC_VECT13SPR1 . . . . .	269
5.1.2.723 _ITC_VECT14SPR . . . . .	270
5.1.2.724 _ITC_VECT14SPR0 . . . . .	270
5.1.2.725 _ITC_VECT14SPR1 . . . . .	270

5.1.2.726 _ITC_VECT15SPR . . . . .	270
5.1.2.727 _ITC_VECT15SPR0 . . . . .	270
5.1.2.728 _ITC_VECT15SPR1 . . . . .	271
5.1.2.729 _ITC_VECT16SPR . . . . .	271
5.1.2.730 _ITC_VECT16SPR0 . . . . .	271
5.1.2.731 _ITC_VECT16SPR1 . . . . .	271
5.1.2.732 _ITC_VECT17SPR . . . . .	271
5.1.2.733 _ITC_VECT17SPR0 . . . . .	272
5.1.2.734 _ITC_VECT17SPR1 . . . . .	272
5.1.2.735 _ITC_VECT18SPR . . . . .	272
5.1.2.736 _ITC_VECT18SPR0 . . . . .	272
5.1.2.737 _ITC_VECT18SPR1 . . . . .	272
5.1.2.738 _ITC_VECT19SPR . . . . .	273
5.1.2.739 _ITC_VECT19SPR0 . . . . .	273
5.1.2.740 _ITC_VECT19SPR1 . . . . .	273
5.1.2.741 _ITC_VECT1SPR . . . . .	273
5.1.2.742 _ITC_VECT1SPR0 . . . . .	273
5.1.2.743 _ITC_VECT1SPR1 . . . . .	274
5.1.2.744 _ITC_VECT20SPR . . . . .	274
5.1.2.745 _ITC_VECT20SPR0 . . . . .	274
5.1.2.746 _ITC_VECT20SPR1 . . . . .	274
5.1.2.747 _ITC_VECT21SPR . . . . .	274
5.1.2.748 _ITC_VECT21SPR0 . . . . .	275
5.1.2.749 _ITC_VECT21SPR1 . . . . .	275
5.1.2.750 _ITC_VECT22SPR . . . . .	275
5.1.2.751 _ITC_VECT22SPR0 . . . . .	275
5.1.2.752 _ITC_VECT22SPR1 . . . . .	275
5.1.2.753 _ITC_VECT23SPR . . . . .	276
5.1.2.754 _ITC_VECT23SPR0 . . . . .	276
5.1.2.755 _ITC_VECT23SPR1 . . . . .	276

5.1.2.756 _ITC_VECT24SPR . . . . .	276
5.1.2.757 _ITC_VECT24SPR0 . . . . .	276
5.1.2.758 _ITC_VECT24SPR1 . . . . .	277
5.1.2.759 _ITC_VECT25SPR . . . . .	277
5.1.2.760 _ITC_VECT25SPR0 . . . . .	277
5.1.2.761 _ITC_VECT25SPR1 . . . . .	277
5.1.2.762 _ITC_VECT26SPR . . . . .	277
5.1.2.763 _ITC_VECT26SPR0 . . . . .	278
5.1.2.764 _ITC_VECT26SPR1 . . . . .	278
5.1.2.765 _ITC_VECT27SPR . . . . .	278
5.1.2.766 _ITC_VECT27SPR0 . . . . .	278
5.1.2.767 _ITC_VECT27SPR1 . . . . .	278
5.1.2.768 _ITC_VECT28SPR . . . . .	279
5.1.2.769 _ITC_VECT28SPR0 . . . . .	279
5.1.2.770 _ITC_VECT28SPR1 . . . . .	279
5.1.2.771 _ITC_VECT29SPR . . . . .	279
5.1.2.772 _ITC_VECT29SPR0 . . . . .	279
5.1.2.773 _ITC_VECT29SPR1 . . . . .	280
5.1.2.774 _ITC_VECT2SPR . . . . .	280
5.1.2.775 _ITC_VECT2SPR0 . . . . .	280
5.1.2.776 _ITC_VECT2SPR1 . . . . .	280
5.1.2.777 _ITC_VECT3SPR . . . . .	280
5.1.2.778 _ITC_VECT3SPR0 . . . . .	281
5.1.2.779 _ITC_VECT3SPR1 . . . . .	281
5.1.2.780 _ITC_VECT4SPR . . . . .	281
5.1.2.781 _ITC_VECT4SPR0 . . . . .	281
5.1.2.782 _ITC_VECT4SPR1 . . . . .	281
5.1.2.783 _ITC_VECT5SPR . . . . .	282
5.1.2.784 _ITC_VECT5SPR0 . . . . .	282
5.1.2.785 _ITC_VECT5SPR1 . . . . .	282

5.1.2.786 _ITC_VECT6SPR . . . . .	282
5.1.2.787 _ITC_VECT6SPR0 . . . . .	282
5.1.2.788 _ITC_VECT6SPR1 . . . . .	283
5.1.2.789 _ITC_VECT7SPR . . . . .	283
5.1.2.790 _ITC_VECT7SPR0 . . . . .	283
5.1.2.791 _ITC_VECT7SPR1 . . . . .	283
5.1.2.792 _ITC_VECT8SPR . . . . .	283
5.1.2.793 _ITC_VECT8SPR0 . . . . .	284
5.1.2.794 _ITC_VECT8SPR1 . . . . .	284
5.1.2.795 _ITC_VECT9SPR . . . . .	284
5.1.2.796 _ITC_VECT9SPR0 . . . . .	284
5.1.2.797 _ITC_VECT9SPR1 . . . . .	284
5.1.2.798 _IWDG . . . . .	285
5.1.2.799 _IWDG_KEY_ACCESS . . . . .	285
5.1.2.800 _IWDG_KEY_ENABLE . . . . .	285
5.1.2.801 _IWDG_KEY_REFRESH . . . . .	285
5.1.2.802 _IWDG_KR . . . . .	285
5.1.2.803 _IWDG_PR . . . . .	286
5.1.2.804 _IWDG_PR_RESET_VALUE . . . . .	286
5.1.2.805 _IWDG_PRE . . . . .	286
5.1.2.806 _IWDG_PRE0 . . . . .	286
5.1.2.807 _IWDG_PRE1 . . . . .	286
5.1.2.808 _IWDG_PRE2 . . . . .	287
5.1.2.809 _IWDG_RLR . . . . .	287
5.1.2.810 _IWDG_RLR_RESET_VALUE . . . . .	287
5.1.2.811 _RST . . . . .	287
5.1.2.812 _RST_EMCF . . . . .	287
5.1.2.813 _RST_ILLOPF . . . . .	288
5.1.2.814 _RST_IWDGF . . . . .	288
5.1.2.815 _RST_SR . . . . .	288

5.1.2.816 _RST_SWIMF . . . . .	288
5.1.2.817 _RST_WWDGF . . . . .	288
5.1.2.818 _SFR . . . . .	289
5.1.2.819 _SPI . . . . .	289
5.1.2.820 _SPI_BDM . . . . .	289
5.1.2.821 _SPI_BDOE . . . . .	289
5.1.2.822 _SPI_BR . . . . .	289
5.1.2.823 _SPI_BR0 . . . . .	290
5.1.2.824 _SPI_BR1 . . . . .	290
5.1.2.825 _SPI_BR2 . . . . .	290
5.1.2.826 _SPI_BSY . . . . .	290
5.1.2.827 _SPI_CPHA . . . . .	290
5.1.2.828 _SPI_CPOL . . . . .	291
5.1.2.829 _SPI_CR1 . . . . .	291
5.1.2.830 _SPI_CR1_RESET_VALUE . . . . .	291
5.1.2.831 _SPI_CR2 . . . . .	291
5.1.2.832 _SPI_CR2_RESET_VALUE . . . . .	291
5.1.2.833 _SPI_CRCEN . . . . .	292
5.1.2.834 _SPI_CRCERR . . . . .	292
5.1.2.835 _SPI_CRCNEXT . . . . .	292
5.1.2.836 _SPI_CRCPR . . . . .	292
5.1.2.837 _SPI_CRCPR_RESET_VALUE . . . . .	292
5.1.2.838 _SPI_DR . . . . .	293
5.1.2.839 _SPI_DR_RESET_VALUE . . . . .	293
5.1.2.840 _SPI_ERRIE . . . . .	293
5.1.2.841 _SPI_ICR . . . . .	293
5.1.2.842 _SPI_ICR_RESET_VALUE . . . . .	293
5.1.2.843 _SPI_LSBFIRST . . . . .	294
5.1.2.844 _SPI_MODF . . . . .	294
5.1.2.845 _SPI_MSTR . . . . .	294

5.1.2.846 _SPI_OVR . . . . .	294
5.1.2.847 _SPI_RXCRCR . . . . .	294
5.1.2.848 _SPI_RXCRCR_RESET_VALUE . . . . .	295
5.1.2.849 _SPI_RXIE . . . . .	295
5.1.2.850 _SPI_RXNE . . . . .	295
5.1.2.851 _SPI_RXONLY . . . . .	295
5.1.2.852 _SPI_SPE . . . . .	295
5.1.2.853 _SPI_SR . . . . .	296
5.1.2.854 _SPI_SR_RESET_VALUE . . . . .	296
5.1.2.855 _SPI_SSI . . . . .	296
5.1.2.856 _SPI_SSM . . . . .	296
5.1.2.857 _SPI_TXCRCR . . . . .	296
5.1.2.858 _SPI_TXCRCR_RESET_VALUE . . . . .	297
5.1.2.859 _SPI_TXE . . . . .	297
5.1.2.860 _SPI_TXIE . . . . .	297
5.1.2.861 _SPI_WKIE . . . . .	297
5.1.2.862 _SPI_WKUP . . . . .	297
5.1.2.863 _TIM1 . . . . .	298
5.1.2.864 _TIM1_AOE . . . . .	298
5.1.2.865 _TIM1_ARPE . . . . .	298
5.1.2.866 _TIM1_ARRH . . . . .	298
5.1.2.867 _TIM1_ARRH_RESET_VALUE . . . . .	298
5.1.2.868 _TIM1_ARRL . . . . .	299
5.1.2.869 _TIM1_ARRL_RESET_VALUE . . . . .	299
5.1.2.870 _TIM1_BG . . . . .	299
5.1.2.871 _TIM1_BIE . . . . .	299
5.1.2.872 _TIM1_BIF . . . . .	299
5.1.2.873 _TIM1_BKE . . . . .	300
5.1.2.874 _TIM1_BKP . . . . .	300
5.1.2.875 _TIM1_BKR . . . . .	300



5.1.2.876 _TIM1_BKR_RESET_VALUE . . . . .	300
5.1.2.877 _TIM1_CC1E . . . . .	300
5.1.2.878 _TIM1_CC1G . . . . .	301
5.1.2.879 _TIM1_CC1IE . . . . .	301
5.1.2.880 _TIM1_CC1IF . . . . .	301
5.1.2.881 _TIM1_CC1NE . . . . .	301
5.1.2.882 _TIM1_CC1NP . . . . .	301
5.1.2.883 _TIM1_CC1OF . . . . .	302
5.1.2.884 _TIM1_CC1P . . . . .	302
5.1.2.885 _TIM1_CC1S . . . . .	302
5.1.2.886 _TIM1_CC1S0 . . . . .	302
5.1.2.887 _TIM1_CC1S1 . . . . .	302
5.1.2.888 _TIM1_CC2E . . . . .	303
5.1.2.889 _TIM1_CC2G . . . . .	303
5.1.2.890 _TIM1_CC2IE . . . . .	303
5.1.2.891 _TIM1_CC2IF . . . . .	303
5.1.2.892 _TIM1_CC2NE . . . . .	303
5.1.2.893 _TIM1_CC2NP . . . . .	304
5.1.2.894 _TIM1_CC2OF . . . . .	304
5.1.2.895 _TIM1_CC2P . . . . .	304
5.1.2.896 _TIM1_CC2S . . . . .	304
5.1.2.897 _TIM1_CC2S0 . . . . .	304
5.1.2.898 _TIM1_CC2S1 . . . . .	305
5.1.2.899 _TIM1_CC3E . . . . .	305
5.1.2.900 _TIM1_CC3G . . . . .	305
5.1.2.901 _TIM1_CC3IE . . . . .	305
5.1.2.902 _TIM1_CC3IF . . . . .	305
5.1.2.903 _TIM1_CC3NE . . . . .	306
5.1.2.904 _TIM1_CC3NP . . . . .	306
5.1.2.905 _TIM1_CC3OF . . . . .	306

5.1.2.906 _TIM1_CC3P . . . . .	306
5.1.2.907 _TIM1_CC3S . . . . .	306
5.1.2.908 _TIM1_CC3S0 . . . . .	307
5.1.2.909 _TIM1_CC3S1 . . . . .	307
5.1.2.910 _TIM1_CC4E . . . . .	307
5.1.2.911 _TIM1_CC4G . . . . .	307
5.1.2.912 _TIM1_CC4IE . . . . .	307
5.1.2.913 _TIM1_CC4IF . . . . .	308
5.1.2.914 _TIM1_CC4OF . . . . .	308
5.1.2.915 _TIM1_CC4P . . . . .	308
5.1.2.916 _TIM1_CC4S . . . . .	308
5.1.2.917 _TIM1_CC4S0 . . . . .	308
5.1.2.918 _TIM1_CC4S1 . . . . .	309
5.1.2.919 _TIM1_CCER1 . . . . .	309
5.1.2.920 _TIM1_CCER1_RESET_VALUE . . . . .	309
5.1.2.921 _TIM1_CCER2 . . . . .	309
5.1.2.922 _TIM1_CCER2_RESET_VALUE . . . . .	309
5.1.2.923 _TIM1_CCMR1 . . . . .	310
5.1.2.924 _TIM1_CCMR1_RESET_VALUE . . . . .	310
5.1.2.925 _TIM1_CCMR2 . . . . .	310
5.1.2.926 _TIM1_CCMR2_RESET_VALUE . . . . .	310
5.1.2.927 _TIM1_CCMR3 . . . . .	310
5.1.2.928 _TIM1_CCMR3_RESET_VALUE . . . . .	311
5.1.2.929 _TIM1_CCMR4 . . . . .	311
5.1.2.930 _TIM1_CCMR4_RESET_VALUE . . . . .	311
5.1.2.931 _TIM1_CCPC . . . . .	311
5.1.2.932 _TIM1_CCR1H . . . . .	311
5.1.2.933 _TIM1_CCR1H_RESET_VALUE . . . . .	312
5.1.2.934 _TIM1_CCR1L . . . . .	312
5.1.2.935 _TIM1_CCR1L_RESET_VALUE . . . . .	312

5.1.2.936 _TIM1_CCR2H . . . . .	312
5.1.2.937 _TIM1_CCR2H_RESET_VALUE . . . . .	312
5.1.2.938 _TIM1_CCR2L . . . . .	313
5.1.2.939 _TIM1_CCR2L_RESET_VALUE . . . . .	313
5.1.2.940 _TIM1_CCR3H . . . . .	313
5.1.2.941 _TIM1_CCR3H_RESET_VALUE . . . . .	313
5.1.2.942 _TIM1_CCR3L . . . . .	313
5.1.2.943 _TIM1_CCR3L_RESET_VALUE . . . . .	314
5.1.2.944 _TIM1_CCR4H . . . . .	314
5.1.2.945 _TIM1_CCR4H_RESET_VALUE . . . . .	314
5.1.2.946 _TIM1_CCR4L . . . . .	314
5.1.2.947 _TIM1_CCR4L_RESET_VALUE . . . . .	314
5.1.2.948 _TIM1_CEN . . . . .	315
5.1.2.949 _TIM1_CMS . . . . .	315
5.1.2.950 _TIM1_CMS0 . . . . .	315
5.1.2.951 _TIM1_CMS1 . . . . .	315
5.1.2.952 _TIM1_CNTRH . . . . .	315
5.1.2.953 _TIM1_CNTRH_RESET_VALUE . . . . .	316
5.1.2.954 _TIM1_CNTRL . . . . .	316
5.1.2.955 _TIM1_CNTRL_RESET_VALUE . . . . .	316
5.1.2.956 _TIM1_COMG . . . . .	316
5.1.2.957 _TIM1_COMIE . . . . .	316
5.1.2.958 _TIM1_COMIF . . . . .	317
5.1.2.959 _TIM1_COMS . . . . .	317
5.1.2.960 _TIM1_CR1 . . . . .	317
5.1.2.961 _TIM1_CR1_RESET_VALUE . . . . .	317
5.1.2.962 _TIM1_CR2 . . . . .	317
5.1.2.963 _TIM1_CR2_RESET_VALUE . . . . .	318
5.1.2.964 _TIM1_DIR . . . . .	318
5.1.2.965 _TIM1_DTR . . . . .	318

5.1.2.966 _TIM1_DTR_RESET_VALUE . . . . .	318
5.1.2.967 _TIM1_ECE . . . . .	318
5.1.2.968 _TIM1_EGR . . . . .	319
5.1.2.969 _TIM1_EGR_RESET_VALUE . . . . .	319
5.1.2.970 _TIM1 ETF . . . . .	319
5.1.2.971 _TIM1 ETF0 . . . . .	319
5.1.2.972 _TIM1 ETF1 . . . . .	319
5.1.2.973 _TIM1 ETF2 . . . . .	320
5.1.2.974 _TIM1 ETF3 . . . . .	320
5.1.2.975 _TIM1 ETP . . . . .	320
5.1.2.976 _TIM1 ETPS . . . . .	320
5.1.2.977 _TIM1 ETPS0 . . . . .	320
5.1.2.978 _TIM1 ETPS1 . . . . .	321
5.1.2.979 _TIM1 ETR . . . . .	321
5.1.2.980 _TIM1 ETR_RESET_VALUE . . . . .	321
5.1.2.981 _TIM1_IC1F . . . . .	321
5.1.2.982 _TIM1_IC1F0 . . . . .	321
5.1.2.983 _TIM1_IC1F1 . . . . .	322
5.1.2.984 _TIM1_IC1F2 . . . . .	322
5.1.2.985 _TIM1_IC1F3 . . . . .	322
5.1.2.986 _TIM1_IC1PSC . . . . .	322
5.1.2.987 _TIM1_IC1PSC0 . . . . .	322
5.1.2.988 _TIM1_IC1PSC1 . . . . .	323
5.1.2.989 _TIM1_IC2F . . . . .	323
5.1.2.990 _TIM1_IC2F0 . . . . .	323
5.1.2.991 _TIM1_IC2F1 . . . . .	323
5.1.2.992 _TIM1_IC2F2 . . . . .	323
5.1.2.993 _TIM1_IC2F3 . . . . .	324
5.1.2.994 _TIM1_IC2PSC . . . . .	324
5.1.2.995 _TIM1_IC2PSC0 . . . . .	324

5.1.2.996_TIM1_IC2PSC1 . . . . .	324
5.1.2.997_TIM1_IC3F . . . . .	324
5.1.2.998_TIM1_IC3F0 . . . . .	325
5.1.2.999_TIM1_IC3F1 . . . . .	325
5.1.2.1000_TIM1_IC3F2 . . . . .	325
5.1.2.1001_TIM1_IC3F3 . . . . .	325
5.1.2.1002_TIM1_IC3PSC . . . . .	325
5.1.2.1003_TIM1_IC3PSC0 . . . . .	326
5.1.2.1004_TIM1_IC3PSC1 . . . . .	326
5.1.2.1005_TIM1_IC4F . . . . .	326
5.1.2.1006_TIM1_IC4F0 . . . . .	326
5.1.2.1007_TIM1_IC4F1 . . . . .	326
5.1.2.1008_TIM1_IC4F2 . . . . .	327
5.1.2.1009_TIM1_IC4F3 . . . . .	327
5.1.2.1010_TIM1_IC4PSC . . . . .	327
5.1.2.1011_TIM1_IC4PSC0 . . . . .	327
5.1.2.1012_TIM1_IC4PSC1 . . . . .	327
5.1.2.1013_TIM1_IER . . . . .	328
5.1.2.1014_TIM1_IER_RESET_VALUE . . . . .	328
5.1.2.1015_TIM1_LOCK . . . . .	328
5.1.2.1016_TIM1_LOCK0 . . . . .	328
5.1.2.1017_TIM1_LOCK1 . . . . .	328
5.1.2.1018_TIM1_MMS . . . . .	329
5.1.2.1019_TIM1_MMS0 . . . . .	329
5.1.2.1020_TIM1_MMS1 . . . . .	329
5.1.2.1021_TIM1_MMS2 . . . . .	329
5.1.2.1022_TIM1_MOE . . . . .	329
5.1.2.1023_TIM1_MSM . . . . .	330
5.1.2.1024_TIM1_OC1CE . . . . .	330
5.1.2.1025_TIM1_OC1FE . . . . .	330

5.1.2.1026_TIM1_OC1M . . . . .	330
5.1.2.1027_TIM1_OC1M0 . . . . .	330
5.1.2.1028_TIM1_OC1M1 . . . . .	331
5.1.2.1029_TIM1_OC1M2 . . . . .	331
5.1.2.1030_TIM1_OC1PE . . . . .	331
5.1.2.1031_TIM1_OC2CE . . . . .	331
5.1.2.1032_TIM1_OC2FE . . . . .	331
5.1.2.1033_TIM1_OC2M . . . . .	332
5.1.2.1034_TIM1_OC2M0 . . . . .	332
5.1.2.1035_TIM1_OC2M1 . . . . .	332
5.1.2.1036_TIM1_OC2M2 . . . . .	332
5.1.2.1037_TIM1_OC2PE . . . . .	332
5.1.2.1038_TIM1_OC3CE . . . . .	333
5.1.2.1039_TIM1_OC3FE . . . . .	333
5.1.2.1040_TIM1_OC3M . . . . .	333
5.1.2.1041_TIM1_OC3M0 . . . . .	333
5.1.2.1042_TIM1_OC3M1 . . . . .	333
5.1.2.1043_TIM1_OC3M2 . . . . .	334
5.1.2.1044_TIM1_OC3PE . . . . .	334
5.1.2.1045_TIM1_OC4CE . . . . .	334
5.1.2.1046_TIM1_OC4FE . . . . .	334
5.1.2.1047_TIM1_OC4M . . . . .	334
5.1.2.1048_TIM1_OC4M0 . . . . .	335
5.1.2.1049_TIM1_OC4M1 . . . . .	335
5.1.2.1050_TIM1_OC4M2 . . . . .	335
5.1.2.1051_TIM1_OC4PE . . . . .	335
5.1.2.1052_TIM1_OIS1 . . . . .	335
5.1.2.1053_TIM1_OIS1N . . . . .	336
5.1.2.1054_TIM1_OIS2 . . . . .	336
5.1.2.1055_TIM1_OIS2N . . . . .	336

5.1.2.1056_TIM1_OIS3 . . . . .	336
5.1.2.1057_TIM1_OIS3N . . . . .	336
5.1.2.1058_TIM1_OIS4 . . . . .	337
5.1.2.1059_TIM1_OISR . . . . .	337
5.1.2.1060_TIM1_OISR_RESET_VALUE . . . . .	337
5.1.2.1061_TIM1_OPM . . . . .	337
5.1.2.1062_TIM1_OSSI . . . . .	337
5.1.2.1063_TIM1_OSSR . . . . .	338
5.1.2.1064_TIM1_PSCRH . . . . .	338
5.1.2.1065_TIM1_PSCRH_RESET_VALUE . . . . .	338
5.1.2.1066_TIM1_PSCRL . . . . .	338
5.1.2.1067_TIM1_PSCRL_RESET_VALUE . . . . .	338
5.1.2.1068_TIM1_RCR . . . . .	339
5.1.2.1069_TIM1_RCR_RESET_VALUE . . . . .	339
5.1.2.1070_TIM1_SMCR . . . . .	339
5.1.2.1071_TIM1_SMCR_RESET_VALUE . . . . .	339
5.1.2.1072_TIM1_SMS . . . . .	339
5.1.2.1073_TIM1_SMS0 . . . . .	340
5.1.2.1074_TIM1_SMS1 . . . . .	340
5.1.2.1075_TIM1_SMS2 . . . . .	340
5.1.2.1076_TIM1_SR1 . . . . .	340
5.1.2.1077_TIM1_SR1_RESET_VALUE . . . . .	340
5.1.2.1078_TIM1_SR2 . . . . .	341
5.1.2.1079_TIM1_SR2_RESET_VALUE . . . . .	341
5.1.2.1080_TIM1_TG . . . . .	341
5.1.2.1081_TIM1_TIE . . . . .	341
5.1.2.1082_TIM1_TIF . . . . .	341
5.1.2.1083_TIM1_TS . . . . .	342
5.1.2.1084_TIM1_TS0 . . . . .	342
5.1.2.1085_TIM1_TS1 . . . . .	342

5.1.2.1086_TIM1_TS2 . . . . .	342
5.1.2.1087_TIM1_UDIS . . . . .	342
5.1.2.1088_TIM1_UG . . . . .	343
5.1.2.1089_TIM1_UIE . . . . .	343
5.1.2.1090_TIM1_UIF . . . . .	343
5.1.2.1091_TIM1_URS . . . . .	343
5.1.2.1092_TIM2 . . . . .	343
5.1.2.1093_TIM2_ARPE . . . . .	344
5.1.2.1094_TIM2_ARRH . . . . .	344
5.1.2.1095_TIM2_ARRH_RESET_VALUE . . . . .	344
5.1.2.1096_TIM2_ARRL . . . . .	344
5.1.2.1097_TIM2_ARRL_RESET_VALUE . . . . .	344
5.1.2.1098_TIM2_CC1E . . . . .	345
5.1.2.1099_TIM2_CC1G . . . . .	345
5.1.2.1100_TIM2_CC1IE . . . . .	345
5.1.2.1101_TIM2_CC1IF . . . . .	345
5.1.2.1102_TIM2_CC1OF . . . . .	345
5.1.2.1103_TIM2_CC1P . . . . .	346
5.1.2.1104_TIM2_CC1S . . . . .	346
5.1.2.1105_TIM2_CC1S0 . . . . .	346
5.1.2.1106_TIM2_CC1S1 . . . . .	346
5.1.2.1107_TIM2_CC2E . . . . .	346
5.1.2.1108_TIM2_CC2G . . . . .	347
5.1.2.1109_TIM2_CC2IE . . . . .	347
5.1.2.1110_TIM2_CC2IF . . . . .	347
5.1.2.1111_TIM2_CC2OF . . . . .	347
5.1.2.1112_TIM2_CC2P . . . . .	347
5.1.2.1113_TIM2_CC2S . . . . .	348
5.1.2.1114_TIM2_CC2S0 . . . . .	348
5.1.2.1115_TIM2_CC2S1 . . . . .	348



5.1.2.1116_TIM2_CC3E . . . . .	348
5.1.2.1117_TIM2_CC3G . . . . .	348
5.1.2.1118_TIM2_CC3IE . . . . .	349
5.1.2.1119_TIM2_CC3IF . . . . .	349
5.1.2.1120_TIM2_CC3OF . . . . .	349
5.1.2.1121_TIM2_CC3P . . . . .	349
5.1.2.1122_TIM2_CC3S . . . . .	349
5.1.2.1123_TIM2_CC3S0 . . . . .	350
5.1.2.1124_TIM2_CC3S1 . . . . .	350
5.1.2.1125_TIM2_CCER1 . . . . .	350
5.1.2.1126_TIM2_CCER1_RESET_VALUE . . . . .	350
5.1.2.1127_TIM2_CCER2 . . . . .	350
5.1.2.1128_TIM2_CCER2_RESET_VALUE . . . . .	351
5.1.2.1129_TIM2_CCMR1 . . . . .	351
5.1.2.1130_TIM2_CCMR1_RESET_VALUE . . . . .	351
5.1.2.1131_TIM2_CCMR2 . . . . .	351
5.1.2.1132_TIM2_CCMR2_RESET_VALUE . . . . .	351
5.1.2.1133_TIM2_CCMR3 . . . . .	352
5.1.2.1134_TIM2_CCMR3_RESET_VALUE . . . . .	352
5.1.2.1135_TIM2_CCR1H . . . . .	352
5.1.2.1136_TIM2_CCR1H_RESET_VALUE . . . . .	352
5.1.2.1137_TIM2_CCR1L . . . . .	352
5.1.2.1138_TIM2_CCR1L_RESET_VALUE . . . . .	353
5.1.2.1139_TIM2_CCR2H . . . . .	353
5.1.2.1140_TIM2_CCR2H_RESET_VALUE . . . . .	353
5.1.2.1141_TIM2_CCR2L . . . . .	353
5.1.2.1142_TIM2_CCR2L_RESET_VALUE . . . . .	353
5.1.2.1143_TIM2_CCR3H . . . . .	354
5.1.2.1144_TIM2_CCR3H_RESET_VALUE . . . . .	354
5.1.2.1145_TIM2_CCR3L . . . . .	354

5.1.2.1146_TIM2_CCR3L_RESET_VALUE . . . . .	354
5.1.2.1147_TIM2_CEN . . . . .	354
5.1.2.1148_TIM2_CNTRH . . . . .	355
5.1.2.1149_TIM2_CNTRH_RESET_VALUE . . . . .	355
5.1.2.1150_TIM2_CNTRL . . . . .	355
5.1.2.1151_TIM2_CNTRL_RESET_VALUE . . . . .	355
5.1.2.1152_TIM2_CR1 . . . . .	355
5.1.2.1153_TIM2_CR1_RESET_VALUE . . . . .	356
5.1.2.1154_TIM2_EGR . . . . .	356
5.1.2.1155_TIM2_EGR_RESET_VALUE . . . . .	356
5.1.2.1156_TIM2_IC1F . . . . .	356
5.1.2.1157_TIM2_IC1F0 . . . . .	356
5.1.2.1158_TIM2_IC1F1 . . . . .	357
5.1.2.1159_TIM2_IC1F2 . . . . .	357
5.1.2.1160_TIM2_IC1F3 . . . . .	357
5.1.2.1161_TIM2_IC1PSC . . . . .	357
5.1.2.1162_TIM2_IC1PSC0 . . . . .	357
5.1.2.1163_TIM2_IC1PSC1 . . . . .	358
5.1.2.1164_TIM2_IC2F . . . . .	358
5.1.2.1165_TIM2_IC2F0 . . . . .	358
5.1.2.1166_TIM2_IC2F1 . . . . .	358
5.1.2.1167_TIM2_IC2F2 . . . . .	358
5.1.2.1168_TIM2_IC2F3 . . . . .	359
5.1.2.1169_TIM2_IC2PSC . . . . .	359
5.1.2.1170_TIM2_IC2PSC0 . . . . .	359
5.1.2.1171_TIM2_IC2PSC1 . . . . .	359
5.1.2.1172_TIM2_IC3F . . . . .	359
5.1.2.1173_TIM2_IC3F0 . . . . .	360
5.1.2.1174_TIM2_IC3F1 . . . . .	360
5.1.2.1175_TIM2_IC3F2 . . . . .	360

5.1.2.1176_TIM2_IC3F3 . . . . .	360
5.1.2.1177_TIM2_IC3PSC . . . . .	360
5.1.2.1178_TIM2_IC3PSC0 . . . . .	361
5.1.2.1179_TIM2_IC3PSC1 . . . . .	361
5.1.2.1180_TIM2_IER . . . . .	361
5.1.2.1181_TIM2_IER_RESET_VALUE . . . . .	361
5.1.2.1182_TIM2_OC1M . . . . .	361
5.1.2.1183_TIM2_OC1M0 . . . . .	362
5.1.2.1184_TIM2_OC1M1 . . . . .	362
5.1.2.1185_TIM2_OC1M2 . . . . .	362
5.1.2.1186_TIM2_OC1PE . . . . .	362
5.1.2.1187_TIM2_OC2M . . . . .	362
5.1.2.1188_TIM2_OC2M0 . . . . .	363
5.1.2.1189_TIM2_OC2M1 . . . . .	363
5.1.2.1190_TIM2_OC2M2 . . . . .	363
5.1.2.1191_TIM2_OC2PE . . . . .	363
5.1.2.1192_TIM2_OC3M . . . . .	363
5.1.2.1193_TIM2_OC3M0 . . . . .	364
5.1.2.1194_TIM2_OC3M1 . . . . .	364
5.1.2.1195_TIM2_OC3M2 . . . . .	364
5.1.2.1196_TIM2_OC3PE . . . . .	364
5.1.2.1197_TIM2_OPM . . . . .	364
5.1.2.1198_TIM2_PSC . . . . .	365
5.1.2.1199_TIM2_PSC0 . . . . .	365
5.1.2.1200_TIM2_PSC1 . . . . .	365
5.1.2.1201_TIM2_PSC2 . . . . .	365
5.1.2.1202_TIM2_PSC3 . . . . .	365
5.1.2.1203_TIM2_PSCR . . . . .	366
5.1.2.1204_TIM2_PSCR_RESET_VALUE . . . . .	366
5.1.2.1205_TIM2_SR1 . . . . .	366

5.1.2.1206_TIM2_SR1_RESET_VALUE . . . . .	366
5.1.2.1207_TIM2_SR2 . . . . .	366
5.1.2.1208_TIM2_SR2_RESET_VALUE . . . . .	367
5.1.2.1209_TIM2_UDIS . . . . .	367
5.1.2.1210_TIM2_UG . . . . .	367
5.1.2.1211_TIM2_UIE . . . . .	367
5.1.2.1212_TIM2_UIF . . . . .	367
5.1.2.1213_TIM2_URS . . . . .	368
5.1.2.1214_TIM3 . . . . .	368
5.1.2.1215_TIM3_ARPE . . . . .	368
5.1.2.1216_TIM3_ARRH . . . . .	368
5.1.2.1217_TIM3_ARRH_RESET_VALUE . . . . .	368
5.1.2.1218_TIM3_ARRL . . . . .	369
5.1.2.1219_TIM3_ARRL_RESET_VALUE . . . . .	369
5.1.2.1220_TIM3_CC1E . . . . .	369
5.1.2.1221_TIM3_CC1G . . . . .	369
5.1.2.1222_TIM3_CC1IE . . . . .	369
5.1.2.1223_TIM3_CC1IF . . . . .	370
5.1.2.1224_TIM3_CC1OF . . . . .	370
5.1.2.1225_TIM3_CC1P . . . . .	370
5.1.2.1226_TIM3_CC1S . . . . .	370
5.1.2.1227_TIM3_CC1S0 . . . . .	370
5.1.2.1228_TIM3_CC1S1 . . . . .	371
5.1.2.1229_TIM3_CC2E . . . . .	371
5.1.2.1230_TIM3_CC2G . . . . .	371
5.1.2.1231_TIM3_CC2IE . . . . .	371
5.1.2.1232_TIM3_CC2IF . . . . .	371
5.1.2.1233_TIM3_CC2OF . . . . .	372
5.1.2.1234_TIM3_CC2P . . . . .	372
5.1.2.1235_TIM3_CC2S . . . . .	372

5.1.2.1236_TIM3_CC2S0 . . . . .	372
5.1.2.1237_TIM3_CC2S1 . . . . .	372
5.1.2.1238_TIM3_CCER1 . . . . .	373
5.1.2.1239_TIM3_CCER1_RESET_VALUE . . . . .	373
5.1.2.1240_TIM3_CCMR1 . . . . .	373
5.1.2.1241_TIM3_CCMR1_RESET_VALUE . . . . .	373
5.1.2.1242_TIM3_CCMR2 . . . . .	373
5.1.2.1243_TIM3_CCMR2_RESET_VALUE . . . . .	374
5.1.2.1244_TIM3_CCR1H . . . . .	374
5.1.2.1245_TIM3_CCR1H_RESET_VALUE . . . . .	374
5.1.2.1246_TIM3_CCR1L . . . . .	374
5.1.2.1247_TIM3_CCR1L_RESET_VALUE . . . . .	374
5.1.2.1248_TIM3_CCR2H . . . . .	375
5.1.2.1249_TIM3_CCR2H_RESET_VALUE . . . . .	375
5.1.2.1250_TIM3_CCR2L . . . . .	375
5.1.2.1251_TIM3_CCR2L_RESET_VALUE . . . . .	375
5.1.2.1252_TIM3_CEN . . . . .	375
5.1.2.1253_TIM3_CNTRH . . . . .	376
5.1.2.1254_TIM3_CNTRH_RESET_VALUE . . . . .	376
5.1.2.1255_TIM3_CNTRL . . . . .	376
5.1.2.1256_TIM3_CNTRL_RESET_VALUE . . . . .	376
5.1.2.1257_TIM3_CR1 . . . . .	376
5.1.2.1258_TIM3_CR1_RESET_VALUE . . . . .	377
5.1.2.1259_TIM3_EGR . . . . .	377
5.1.2.1260_TIM3_EGR_RESET_VALUE . . . . .	377
5.1.2.1261_TIM3_IC1F . . . . .	377
5.1.2.1262_TIM3_IC1F0 . . . . .	377
5.1.2.1263_TIM3_IC1F1 . . . . .	378
5.1.2.1264_TIM3_IC1F2 . . . . .	378
5.1.2.1265_TIM3_IC1F3 . . . . .	378

5.1.2.1266_TIM3_IC1PSC . . . . .	378
5.1.2.1267_TIM3_IC1PSC0 . . . . .	378
5.1.2.1268_TIM3_IC1PSC1 . . . . .	379
5.1.2.1269_TIM3_IC2F . . . . .	379
5.1.2.1270_TIM3_IC2F0 . . . . .	379
5.1.2.1271_TIM3_IC2F1 . . . . .	379
5.1.2.1272_TIM3_IC2F2 . . . . .	379
5.1.2.1273_TIM3_IC2F3 . . . . .	380
5.1.2.1274_TIM3_IC2PSC . . . . .	380
5.1.2.1275_TIM3_IC2PSC0 . . . . .	380
5.1.2.1276_TIM3_IC2PSC1 . . . . .	380
5.1.2.1277_TIM3_IER . . . . .	380
5.1.2.1278_TIM3_IER_RESET_VALUE . . . . .	381
5.1.2.1279_TIM3_OC1M . . . . .	381
5.1.2.1280_TIM3_OC1M0 . . . . .	381
5.1.2.1281_TIM3_OC1M1 . . . . .	381
5.1.2.1282_TIM3_OC1M2 . . . . .	381
5.1.2.1283_TIM3_OC1PE . . . . .	382
5.1.2.1284_TIM3_OC2M . . . . .	382
5.1.2.1285_TIM3_OC2M0 . . . . .	382
5.1.2.1286_TIM3_OC2M1 . . . . .	382
5.1.2.1287_TIM3_OC2M2 . . . . .	382
5.1.2.1288_TIM3_OC2PE . . . . .	383
5.1.2.1289_TIM3_OPM . . . . .	383
5.1.2.1290_TIM3_PSC . . . . .	383
5.1.2.1291_TIM3_PSC0 . . . . .	383
5.1.2.1292_TIM3_PSC1 . . . . .	383
5.1.2.1293_TIM3_PSC2 . . . . .	384
5.1.2.1294_TIM3_PSC3 . . . . .	384
5.1.2.1295_TIM3_PSCR . . . . .	384

5.1.2.1296_TIM3_PSCR_RESET_VALUE . . . . .	384
5.1.2.1297_TIM3_SR1 . . . . .	384
5.1.2.1298_TIM3_SR1_RESET_VALUE . . . . .	385
5.1.2.1299_TIM3_SR2 . . . . .	385
5.1.2.1300_TIM3_SR2_RESET_VALUE . . . . .	385
5.1.2.1301_TIM3_UDIS . . . . .	385
5.1.2.1302_TIM3_UG . . . . .	385
5.1.2.1303_TIM3_UIE . . . . .	386
5.1.2.1304_TIM3_UIF . . . . .	386
5.1.2.1305_TIM3_URS . . . . .	386
5.1.2.1306_TIM4 . . . . .	386
5.1.2.1307_TIM4_ARPE . . . . .	386
5.1.2.1308_TIM4_ARR . . . . .	387
5.1.2.1309_TIM4_ARR_RESET_VALUE . . . . .	387
5.1.2.1310_TIM4_CEN . . . . .	387
5.1.2.1311_TIM4_CNTR . . . . .	387
5.1.2.1312_TIM4_CNTR_RESET_VALUE . . . . .	387
5.1.2.1313_TIM4_CR . . . . .	388
5.1.2.1314_TIM4_CR_RESET_VALUE . . . . .	388
5.1.2.1315_TIM4_EGR . . . . .	388
5.1.2.1316_TIM4_EGR_RESET_VALUE . . . . .	388
5.1.2.1317_TIM4_IER . . . . .	388
5.1.2.1318_TIM4_IER_RESET_VALUE . . . . .	389
5.1.2.1319_TIM4_OPM . . . . .	389
5.1.2.1320_TIM4_PSC . . . . .	389
5.1.2.1321_TIM4_PSC0 . . . . .	389
5.1.2.1322_TIM4_PSC1 . . . . .	389
5.1.2.1323_TIM4_PSC2 . . . . .	390
5.1.2.1324_TIM4_PSCR . . . . .	390
5.1.2.1325_TIM4_PSCR_RESET_VALUE . . . . .	390

5.1.2.1326_TIM4_SR . . . . .	390
5.1.2.1327_TIM4_SR_RESET_VALUE . . . . .	390
5.1.2.1328_TIM4_UDIS . . . . .	391
5.1.2.1329_TIM4_UG . . . . .	391
5.1.2.1330_TIM4_UIE . . . . .	391
5.1.2.1331_TIM4_UIF . . . . .	391
5.1.2.1332_TIM4_URS . . . . .	391
5.1.2.1333_TIM5 . . . . .	392
5.1.2.1334_TIM5_ARPE . . . . .	392
5.1.2.1335_TIM5_ARRH . . . . .	392
5.1.2.1336_TIM5_ARRH_RESET_VALUE . . . . .	392
5.1.2.1337_TIM5_ARRL . . . . .	392
5.1.2.1338_TIM5_ARRL_RESET_VALUE . . . . .	393
5.1.2.1339_TIM5_CC1E . . . . .	393
5.1.2.1340_TIM5_CC1G . . . . .	393
5.1.2.1341_TIM5_CC1IE . . . . .	393
5.1.2.1342_TIM5_CC1IF . . . . .	393
5.1.2.1343_TIM5_CC1OF . . . . .	394
5.1.2.1344_TIM5_CC1P . . . . .	394
5.1.2.1345_TIM5_CC1S . . . . .	394
5.1.2.1346_TIM5_CC1S0 . . . . .	394
5.1.2.1347_TIM5_CC1S1 . . . . .	394
5.1.2.1348_TIM5_CC2E . . . . .	395
5.1.2.1349_TIM5_CC2G . . . . .	395
5.1.2.1350_TIM5_CC2IE . . . . .	395
5.1.2.1351_TIM5_CC2IF . . . . .	395
5.1.2.1352_TIM5_CC2OF . . . . .	395
5.1.2.1353_TIM5_CC2P . . . . .	396
5.1.2.1354_TIM5_CC2S . . . . .	396
5.1.2.1355_TIM5_CC2S0 . . . . .	396



5.1.2.1356_TIM5_CC2S1 . . . . .	396
5.1.2.1357_TIM5_CC3E . . . . .	396
5.1.2.1358_TIM5_CC3G . . . . .	397
5.1.2.1359_TIM5_CC3IE . . . . .	397
5.1.2.1360_TIM5_CC3IF . . . . .	397
5.1.2.1361_TIM5_CC3OF . . . . .	397
5.1.2.1362_TIM5_CC3P . . . . .	397
5.1.2.1363_TIM5_CC3S . . . . .	398
5.1.2.1364_TIM5_CC3S0 . . . . .	398
5.1.2.1365_TIM5_CC3S1 . . . . .	398
5.1.2.1366_TIM5_CCER1 . . . . .	398
5.1.2.1367_TIM5_CCER1_RESET_VALUE . . . . .	398
5.1.2.1368_TIM5_CCER2 . . . . .	399
5.1.2.1369_TIM5_CCER2_RESET_VALUE . . . . .	399
5.1.2.1370_TIM5_CCMR1 . . . . .	399
5.1.2.1371_TIM5_CCMR1_RESET_VALUE . . . . .	399
5.1.2.1372_TIM5_CCMR2 . . . . .	399
5.1.2.1373_TIM5_CCMR2_RESET_VALUE . . . . .	400
5.1.2.1374_TIM5_CCMR3 . . . . .	400
5.1.2.1375_TIM5_CCMR3_RESET_VALUE . . . . .	400
5.1.2.1376_TIM5_CCPC . . . . .	400
5.1.2.1377_TIM5_CCR1H . . . . .	400
5.1.2.1378_TIM5_CCR1H_RESET_VALUE . . . . .	401
5.1.2.1379_TIM5_CCR1L . . . . .	401
5.1.2.1380_TIM5_CCR1L_RESET_VALUE . . . . .	401
5.1.2.1381_TIM5_CCR2H . . . . .	401
5.1.2.1382_TIM5_CCR2H_RESET_VALUE . . . . .	401
5.1.2.1383_TIM5_CCR2L . . . . .	402
5.1.2.1384_TIM5_CCR2L_RESET_VALUE . . . . .	402
5.1.2.1385_TIM5_CCR3H . . . . .	402

5.1.2.1386_TIM5_CCR3H_RESET_VALUE . . . . .	402
5.1.2.1387_TIM5_CCR3L . . . . .	402
5.1.2.1388_TIM5_CCR3L_RESET_VALUE . . . . .	403
5.1.2.1389_TIM5_CEN . . . . .	403
5.1.2.1390_TIM5_CNTRH . . . . .	403
5.1.2.1391_TIM5_CNTRH_RESET_VALUE . . . . .	403
5.1.2.1392_TIM5_CNTRL . . . . .	403
5.1.2.1393_TIM5_CNTRL_RESET_VALUE . . . . .	404
5.1.2.1394_TIM5_COMS . . . . .	404
5.1.2.1395_TIM5_CR1 . . . . .	404
5.1.2.1396_TIM5_CR1_RESET_VALUE . . . . .	404
5.1.2.1397_TIM5_CR2 . . . . .	404
5.1.2.1398_TIM5_CR2_RESET_VALUE . . . . .	405
5.1.2.1399_TIM5_EGR . . . . .	405
5.1.2.1400_TIM5_EGR_RESET_VALUE . . . . .	405
5.1.2.1401_TIM5_IC1F . . . . .	405
5.1.2.1402_TIM5_IC1F0 . . . . .	405
5.1.2.1403_TIM5_IC1F1 . . . . .	406
5.1.2.1404_TIM5_IC1F2 . . . . .	406
5.1.2.1405_TIM5_IC1F3 . . . . .	406
5.1.2.1406_TIM5_IC1PSC . . . . .	406
5.1.2.1407_TIM5_IC1PSC0 . . . . .	406
5.1.2.1408_TIM5_IC1PSC1 . . . . .	407
5.1.2.1409_TIM5_IC2F . . . . .	407
5.1.2.1410_TIM5_IC2F0 . . . . .	407
5.1.2.1411_TIM5_IC2F1 . . . . .	407
5.1.2.1412_TIM5_IC2F2 . . . . .	407
5.1.2.1413_TIM5_IC2F3 . . . . .	408
5.1.2.1414_TIM5_IC2PSC . . . . .	408
5.1.2.1415_TIM5_IC2PSC0 . . . . .	408

5.1.2.1416_TIM5_IC2PSC1 . . . . .	408
5.1.2.1417_TIM5_IC3F . . . . .	408
5.1.2.1418_TIM5_IC3F0 . . . . .	409
5.1.2.1419_TIM5_IC3F1 . . . . .	409
5.1.2.1420_TIM5_IC3F2 . . . . .	409
5.1.2.1421_TIM5_IC3F3 . . . . .	409
5.1.2.1422_TIM5_IC3PSC . . . . .	409
5.1.2.1423_TIM5_IC3PSC0 . . . . .	410
5.1.2.1424_TIM5_IC3PSC1 . . . . .	410
5.1.2.1425_TIM5_IER . . . . .	410
5.1.2.1426_TIM5_IER_RESET_VALUE . . . . .	410
5.1.2.1427_TIM5_MMS . . . . .	410
5.1.2.1428_TIM5_MMS0 . . . . .	411
5.1.2.1429_TIM5_MMS1 . . . . .	411
5.1.2.1430_TIM5_MMS2 . . . . .	411
5.1.2.1431_TIM5_MSM . . . . .	411
5.1.2.1432_TIM5_OC1M . . . . .	411
5.1.2.1433_TIM5_OC1M0 . . . . .	412
5.1.2.1434_TIM5_OC1M1 . . . . .	412
5.1.2.1435_TIM5_OC1M2 . . . . .	412
5.1.2.1436_TIM5_OC1PE . . . . .	412
5.1.2.1437_TIM5_OC2M . . . . .	412
5.1.2.1438_TIM5_OC2M0 . . . . .	413
5.1.2.1439_TIM5_OC2M1 . . . . .	413
5.1.2.1440_TIM5_OC2M2 . . . . .	413
5.1.2.1441_TIM5_OC2PE . . . . .	413
5.1.2.1442_TIM5_OC3M . . . . .	413
5.1.2.1443_TIM5_OC3M0 . . . . .	414
5.1.2.1444_TIM5_OC3M1 . . . . .	414
5.1.2.1445_TIM5_OC3M2 . . . . .	414

5.1.2.1446_TIM5_OC3PE . . . . .	414
5.1.2.1447_TIM5_OPM . . . . .	414
5.1.2.1448_TIM5_PSC . . . . .	415
5.1.2.1449_TIM5_PSC0 . . . . .	415
5.1.2.1450_TIM5_PSC1 . . . . .	415
5.1.2.1451_TIM5_PSC2 . . . . .	415
5.1.2.1452_TIM5_PSC3 . . . . .	415
5.1.2.1453_TIM5_PSCR . . . . .	416
5.1.2.1454_TIM5_PSCR_RESET_VALUE . . . . .	416
5.1.2.1455_TIM5_SMCR . . . . .	416
5.1.2.1456_TIM5_SMCR_RESET_VALUE . . . . .	416
5.1.2.1457_TIM5_SMS . . . . .	416
5.1.2.1458_TIM5_SMS0 . . . . .	417
5.1.2.1459_TIM5_SMS1 . . . . .	417
5.1.2.1460_TIM5_SMS2 . . . . .	417
5.1.2.1461_TIM5_SR1 . . . . .	417
5.1.2.1462_TIM5_SR1_RESET_VALUE . . . . .	417
5.1.2.1463_TIM5_SR2 . . . . .	418
5.1.2.1464_TIM5_SR2_RESET_VALUE . . . . .	418
5.1.2.1465_TIM5_TG . . . . .	418
5.1.2.1466_TIM5_TIE . . . . .	418
5.1.2.1467_TIM5_TIF . . . . .	418
5.1.2.1468_TIM5_TS . . . . .	419
5.1.2.1469_TIM5_TS0 . . . . .	419
5.1.2.1470_TIM5_TS1 . . . . .	419
5.1.2.1471_TIM5_TS2 . . . . .	419
5.1.2.1472_TIM5_UDIS . . . . .	419
5.1.2.1473_TIM5_UG . . . . .	420
5.1.2.1474_TIM5_UIE . . . . .	420
5.1.2.1475_TIM5_UIF . . . . .	420

5.1.2.1476_TIM5_URS . . . . .	420
5.1.2.1477_TIM6 . . . . .	420
5.1.2.1478_TIM6_ARPE . . . . .	421
5.1.2.1479_TIM6_ARR . . . . .	421
5.1.2.1480_TIM6_ARR_RESET_VALUE . . . . .	421
5.1.2.1481_TIM6_CEN . . . . .	421
5.1.2.1482_TIM6_CNTR . . . . .	421
5.1.2.1483_TIM6_CNTR_RESET_VALUE . . . . .	422
5.1.2.1484_TIM6_CR . . . . .	422
5.1.2.1485_TIM6_CR_RESET_VALUE . . . . .	422
5.1.2.1486_TIM6_EGR . . . . .	422
5.1.2.1487_TIM6_EGR_RESET_VALUE . . . . .	422
5.1.2.1488_TIM6_IER . . . . .	423
5.1.2.1489_TIM6_IER_RESET_VALUE . . . . .	423
5.1.2.1490_TIM6_MMS . . . . .	423
5.1.2.1491_TIM6_MMS0 . . . . .	423
5.1.2.1492_TIM6_MMS1 . . . . .	423
5.1.2.1493_TIM6_MMS2 . . . . .	424
5.1.2.1494_TIM6_OPM . . . . .	424
5.1.2.1495_TIM6_PSC . . . . .	424
5.1.2.1496_TIM6_PSC0 . . . . .	424
5.1.2.1497_TIM6_PSC1 . . . . .	424
5.1.2.1498_TIM6_PSC2 . . . . .	425
5.1.2.1499_TIM6_PSCR . . . . .	425
5.1.2.1500_TIM6_PSCR_RESET_VALUE . . . . .	425
5.1.2.1501_TIM6_SMS . . . . .	425
5.1.2.1502_TIM6_SMS0 . . . . .	425
5.1.2.1503_TIM6_SMS1 . . . . .	426
5.1.2.1504_TIM6_SMS2 . . . . .	426
5.1.2.1505_TIM6_SR . . . . .	426

5.1.2.1506_TIM6_SR_RESET_VALUE . . . . .	426
5.1.2.1507_TIM6_TS . . . . .	426
5.1.2.1508_TIM6_TS0 . . . . .	427
5.1.2.1509_TIM6_TS1 . . . . .	427
5.1.2.1510_TIM6_TS2 . . . . .	427
5.1.2.1511_TIM6_UDIS . . . . .	427
5.1.2.1512_TIM6_UG . . . . .	427
5.1.2.1513_TIM6_UIE . . . . .	428
5.1.2.1514_TIM6_UIF . . . . .	428
5.1.2.1515_TIM6_URS . . . . .	428
5.1.2.1516_UART1 . . . . .	428
5.1.2.1517_UART1_ADD . . . . .	428
5.1.2.1518_UART1_ADD0 . . . . .	429
5.1.2.1519_UART1_ADD1 . . . . .	429
5.1.2.1520_UART1_ADD2 . . . . .	429
5.1.2.1521_UART1_ADD3 . . . . .	429
5.1.2.1522_UART1_BRR1 . . . . .	429
5.1.2.1523_UART1_BRR1_RESET_VALUE . . . . .	430
5.1.2.1524_UART1_BRR2 . . . . .	430
5.1.2.1525_UART1_BRR2_RESET_VALUE . . . . .	430
5.1.2.1526_UART1_CKEN . . . . .	430
5.1.2.1527_UART1_CPHA . . . . .	430
5.1.2.1528_UART1_CPOL . . . . .	431
5.1.2.1529_UART1_CR1 . . . . .	431
5.1.2.1530_UART1_CR1_RESET_VALUE . . . . .	431
5.1.2.1531_UART1_CR2 . . . . .	431
5.1.2.1532_UART1_CR2_RESET_VALUE . . . . .	431
5.1.2.1533_UART1_CR3 . . . . .	432
5.1.2.1534_UART1_CR3_RESET_VALUE . . . . .	432
5.1.2.1535_UART1_CR4 . . . . .	432

5.1.2.1536_UART1_CR4_RESET_VALUE . . . . .	432
5.1.2.1537_UART1_CR5 . . . . .	432
5.1.2.1538_UART1_CR5_RESET_VALUE . . . . .	433
5.1.2.1539_UART1_DR . . . . .	433
5.1.2.1540_UART1_FE . . . . .	433
5.1.2.1541_UART1_GTR . . . . .	433
5.1.2.1542_UART1_GTR_RESET_VALUE . . . . .	433
5.1.2.1543_UART1_HDSEL . . . . .	434
5.1.2.1544_UART1_IDLE . . . . .	434
5.1.2.1545_UART1_ILIEN . . . . .	434
5.1.2.1546_UART1_IREN . . . . .	434
5.1.2.1547_UART1_IRLP . . . . .	434
5.1.2.1548_UART1_LBCL . . . . .	435
5.1.2.1549_UART1_LBDF . . . . .	435
5.1.2.1550_UART1_LBDIEN . . . . .	435
5.1.2.1551_UART1_LBDL . . . . .	435
5.1.2.1552_UART1_LINEN . . . . .	435
5.1.2.1553_UART1_M . . . . .	436
5.1.2.1554_UART1_NACK . . . . .	436
5.1.2.1555_UART1_NF . . . . .	436
5.1.2.1556_UART1_OR_LHE . . . . .	436
5.1.2.1557_UART1_PCEN . . . . .	436
5.1.2.1558_UART1_PE . . . . .	437
5.1.2.1559_UART1_PIEN . . . . .	437
5.1.2.1560_UART1_PS . . . . .	437
5.1.2.1561_UART1_PSCR . . . . .	437
5.1.2.1562_UART1_PSCR_RESET_VALUE . . . . .	437
5.1.2.1563_UART1_R8 . . . . .	438
5.1.2.1564_UART1_REN . . . . .	438
5.1.2.1565_UART1_RIEN . . . . .	438

5.1.2.1566_UART1_RWU . . . . .	438
5.1.2.1567_UART1_RXNE . . . . .	438
5.1.2.1568_UART1_SBK . . . . .	439
5.1.2.1569_UART1_SCEN . . . . .	439
5.1.2.1570_UART1_SR . . . . .	439
5.1.2.1571_UART1_SR_RESET_VALUE . . . . .	439
5.1.2.1572_UART1_STOP . . . . .	439
5.1.2.1573_UART1_STOP0 . . . . .	440
5.1.2.1574_UART1_STOP1 . . . . .	440
5.1.2.1575_UART1_T8 . . . . .	440
5.1.2.1576_UART1_TC . . . . .	440
5.1.2.1577_UART1_TCIEN . . . . .	440
5.1.2.1578_UART1_TEN . . . . .	441
5.1.2.1579_UART1_TIEN . . . . .	441
5.1.2.1580_UART1_TXE . . . . .	441
5.1.2.1581_UART1_UARTD . . . . .	441
5.1.2.1582_UART1_WAKE . . . . .	441
5.1.2.1583_UART2 . . . . .	442
5.1.2.1584_UART2_ADD . . . . .	442
5.1.2.1585_UART2_ADD0 . . . . .	442
5.1.2.1586_UART2_ADD1 . . . . .	442
5.1.2.1587_UART2_ADD2 . . . . .	442
5.1.2.1588_UART2_ADD3 . . . . .	443
5.1.2.1589_UART2_BRR1 . . . . .	443
5.1.2.1590_UART2_BRR1_RESET_VALUE . . . . .	443
5.1.2.1591_UART2_BRR2 . . . . .	443
5.1.2.1592_UART2_BRR2_RESET_VALUE . . . . .	443
5.1.2.1593_UART2_CKEN . . . . .	444
5.1.2.1594_UART2_CPHA . . . . .	444
5.1.2.1595_UART2_CPOL . . . . .	444



5.1.2.1596_UART2_CR1 . . . . .	444
5.1.2.1597_UART2_CR1_RESET_VALUE . . . . .	444
5.1.2.1598_UART2_CR2 . . . . .	445
5.1.2.1599_UART2_CR2_RESET_VALUE . . . . .	445
5.1.2.1600_UART2_CR3 . . . . .	445
5.1.2.1601_UART2_CR3_RESET_VALUE . . . . .	445
5.1.2.1602_UART2_CR4 . . . . .	445
5.1.2.1603_UART2_CR4_RESET_VALUE . . . . .	446
5.1.2.1604_UART2_CR5 . . . . .	446
5.1.2.1605_UART2_CR5_RESET_VALUE . . . . .	446
5.1.2.1606_UART2_CR6 . . . . .	446
5.1.2.1607_UART2_CR6_RESET_VALUE . . . . .	446
5.1.2.1608_UART2_DR . . . . .	447
5.1.2.1609_UART2_FE . . . . .	447
5.1.2.1610_UART2_GTR . . . . .	447
5.1.2.1611_UART2_GTR_RESET_VALUE . . . . .	447
5.1.2.1612_UART2_IDLE . . . . .	447
5.1.2.1613_UART2_ILIEN . . . . .	448
5.1.2.1614_UART2_IREN . . . . .	448
5.1.2.1615_UART2_IRLP . . . . .	448
5.1.2.1616_UART2_LASE . . . . .	448
5.1.2.1617_UART2_LBCL . . . . .	448
5.1.2.1618_UART2_LBDF . . . . .	449
5.1.2.1619_UART2_LBDIEN . . . . .	449
5.1.2.1620_UART2_LBDL . . . . .	449
5.1.2.1621_UART2_LDUM . . . . .	449
5.1.2.1622_UART2_LHDF . . . . .	449
5.1.2.1623_UART2_LHDIEN . . . . .	450
5.1.2.1624_UART2_LINEN . . . . .	450
5.1.2.1625_UART2_LSF . . . . .	450

5.1.2.1626_UART2_LSLV . . . . .	450
5.1.2.1627_UART2_M . . . . .	450
5.1.2.1628_UART2_NACK . . . . .	451
5.1.2.1629_UART2_NF . . . . .	451
5.1.2.1630_UART2_OR_LHE . . . . .	451
5.1.2.1631_UART2_PCEN . . . . .	451
5.1.2.1632_UART2_PE . . . . .	451
5.1.2.1633_UART2_PIEN . . . . .	452
5.1.2.1634_UART2_PS . . . . .	452
5.1.2.1635_UART2_PSCR . . . . .	452
5.1.2.1636_UART2_PSCR_RESET_VALUE . . . . .	452
5.1.2.1637_UART2_R8 . . . . .	452
5.1.2.1638_UART2_REN . . . . .	453
5.1.2.1639_UART2_RIEN . . . . .	453
5.1.2.1640_UART2_RWU . . . . .	453
5.1.2.1641_UART2_RXNE . . . . .	453
5.1.2.1642_UART2_SBK . . . . .	453
5.1.2.1643_UART2_SCEN . . . . .	454
5.1.2.1644_UART2_SR . . . . .	454
5.1.2.1645_UART2_SR_RESET_VALUE . . . . .	454
5.1.2.1646_UART2_STOP . . . . .	454
5.1.2.1647_UART2_STOP0 . . . . .	454
5.1.2.1648_UART2_STOP1 . . . . .	455
5.1.2.1649_UART2_T8 . . . . .	455
5.1.2.1650_UART2_TC . . . . .	455
5.1.2.1651_UART2_TCIEN . . . . .	455
5.1.2.1652_UART2_TEN . . . . .	455
5.1.2.1653_UART2_TIEN . . . . .	456
5.1.2.1654_UART2_TXE . . . . .	456
5.1.2.1655_UART2_UARTD . . . . .	456

5.1.2.1656_UART2_WAKE . . . . .	456
5.1.2.1657_UART3 . . . . .	456
5.1.2.1658_UART3_ADD . . . . .	457
5.1.2.1659_UART3_ADD0 . . . . .	457
5.1.2.1660_UART3_ADD1 . . . . .	457
5.1.2.1661_UART3_ADD2 . . . . .	457
5.1.2.1662_UART3_ADD3 . . . . .	457
5.1.2.1663_UART3_BRR1 . . . . .	458
5.1.2.1664_UART3_BRR1_RESET_VALUE . . . . .	458
5.1.2.1665_UART3_BRR2 . . . . .	458
5.1.2.1666_UART3_BRR2_RESET_VALUE . . . . .	458
5.1.2.1667_UART3_CR1 . . . . .	458
5.1.2.1668_UART3_CR1_RESET_VALUE . . . . .	459
5.1.2.1669_UART3_CR2 . . . . .	459
5.1.2.1670_UART3_CR2_RESET_VALUE . . . . .	459
5.1.2.1671_UART3_CR3 . . . . .	459
5.1.2.1672_UART3_CR3_RESET_VALUE . . . . .	459
5.1.2.1673_UART3_CR4 . . . . .	460
5.1.2.1674_UART3_CR4_RESET_VALUE . . . . .	460
5.1.2.1675_UART3_CR6 . . . . .	460
5.1.2.1676_UART3_CR6_RESET_VALUE . . . . .	460
5.1.2.1677_UART3_DR . . . . .	460
5.1.2.1678_UART3_FE . . . . .	461
5.1.2.1679_UART3_IDLE . . . . .	461
5.1.2.1680_UART3_ILIEN . . . . .	461
5.1.2.1681_UART3_LASE . . . . .	461
5.1.2.1682_UART3_LBDF . . . . .	461
5.1.2.1683_UART3_LBDIEN . . . . .	462
5.1.2.1684_UART3_LBDL . . . . .	462
5.1.2.1685_UART3_LDUM . . . . .	462

5.1.2.1686_UART3_LHDF . . . . .	462
5.1.2.1687_UART3_LHDIEN . . . . .	462
5.1.2.1688_UART3_LINEN . . . . .	463
5.1.2.1689_UART3_LSF . . . . .	463
5.1.2.1690_UART3_LSLV . . . . .	463
5.1.2.1691_UART3_M . . . . .	463
5.1.2.1692_UART3_NF . . . . .	463
5.1.2.1693_UART3_OR_LHE . . . . .	464
5.1.2.1694_UART3_PCEN . . . . .	464
5.1.2.1695_UART3_PE . . . . .	464
5.1.2.1696_UART3_PIEN . . . . .	464
5.1.2.1697_UART3_PS . . . . .	464
5.1.2.1698_UART3_R8 . . . . .	465
5.1.2.1699_UART3_REN . . . . .	465
5.1.2.1700_UART3_RIEN . . . . .	465
5.1.2.1701_UART3_RWU . . . . .	465
5.1.2.1702_UART3_RXNE . . . . .	465
5.1.2.1703_UART3_SBK . . . . .	466
5.1.2.1704_UART3_SR . . . . .	466
5.1.2.1705_UART3_SR_RESET_VALUE . . . . .	466
5.1.2.1706_UART3_STOP . . . . .	466
5.1.2.1707_UART3_STOP0 . . . . .	466
5.1.2.1708_UART3_STOP1 . . . . .	467
5.1.2.1709_UART3_T8 . . . . .	467
5.1.2.1710_UART3_TC . . . . .	467
5.1.2.1711_UART3_TCIEN . . . . .	467
5.1.2.1712_UART3_TEN . . . . .	467
5.1.2.1713_UART3_TIEN . . . . .	468
5.1.2.1714_UART3_TXE . . . . .	468
5.1.2.1715_UART3_UARTD . . . . .	468

5.1.2.1716_UART3_WAKE . . . . .	468
5.1.2.1717_UART4 . . . . .	468
5.1.2.1718_UART4_ADD . . . . .	469
5.1.2.1719_UART4_ADD0 . . . . .	469
5.1.2.1720_UART4_ADD1 . . . . .	469
5.1.2.1721_UART4_ADD2 . . . . .	469
5.1.2.1722_UART4_ADD3 . . . . .	469
5.1.2.1723_UART4_BRR1 . . . . .	470
5.1.2.1724_UART4_BRR1_RESET_VALUE . . . . .	470
5.1.2.1725_UART4_BRR2 . . . . .	470
5.1.2.1726_UART4_BRR2_RESET_VALUE . . . . .	470
5.1.2.1727_UART4_CKEN . . . . .	470
5.1.2.1728_UART4_CPHA . . . . .	471
5.1.2.1729_UART4_CPOL . . . . .	471
5.1.2.1730_UART4_CR1 . . . . .	471
5.1.2.1731_UART4_CR1_RESET_VALUE . . . . .	471
5.1.2.1732_UART4_CR2 . . . . .	471
5.1.2.1733_UART4_CR2_RESET_VALUE . . . . .	472
5.1.2.1734_UART4_CR3 . . . . .	472
5.1.2.1735_UART4_CR3_RESET_VALUE . . . . .	472
5.1.2.1736_UART4_CR4 . . . . .	472
5.1.2.1737_UART4_CR4_RESET_VALUE . . . . .	472
5.1.2.1738_UART4_CR5 . . . . .	473
5.1.2.1739_UART4_CR5_RESET_VALUE . . . . .	473
5.1.2.1740_UART4_CR6 . . . . .	473
5.1.2.1741_UART4_CR6_RESET_VALUE . . . . .	473
5.1.2.1742_UART4_DR . . . . .	473
5.1.2.1743_UART4_FE . . . . .	474
5.1.2.1744_UART4_GTR . . . . .	474
5.1.2.1745_UART4_GTR_RESET_VALUE . . . . .	474

5.1.2.1746_UART4_HDSEL . . . . .	474
5.1.2.1747_UART4_IDLE . . . . .	474
5.1.2.1748_UART4_ILIEN . . . . .	475
5.1.2.1749_UART4_IREN . . . . .	475
5.1.2.1750_UART4_IRLP . . . . .	475
5.1.2.1751_UART4_LASE . . . . .	475
5.1.2.1752_UART4_LBCL . . . . .	475
5.1.2.1753_UART4_LBDF . . . . .	476
5.1.2.1754_UART4_LBDIEN . . . . .	476
5.1.2.1755_UART4_LBDL . . . . .	476
5.1.2.1756_UART4_LDUM . . . . .	476
5.1.2.1757_UART4_LHDF . . . . .	476
5.1.2.1758_UART4_LHDIEN . . . . .	477
5.1.2.1759_UART4_LINEN . . . . .	477
5.1.2.1760_UART4_LSF . . . . .	477
5.1.2.1761_UART4_LSLV . . . . .	477
5.1.2.1762_UART4_M . . . . .	477
5.1.2.1763_UART4_NACK . . . . .	478
5.1.2.1764_UART4_NF . . . . .	478
5.1.2.1765_UART4_OR_LHE . . . . .	478
5.1.2.1766_UART4_PCEN . . . . .	478
5.1.2.1767_UART4_PE . . . . .	478
5.1.2.1768_UART4_PIEN . . . . .	479
5.1.2.1769_UART4_PS . . . . .	479
5.1.2.1770_UART4_PSCR . . . . .	479
5.1.2.1771_UART4_PSCR_RESET_VALUE . . . . .	479
5.1.2.1772_UART4_R8 . . . . .	479
5.1.2.1773_UART4_REN . . . . .	480
5.1.2.1774_UART4_RIEN . . . . .	480
5.1.2.1775_UART4_RWU . . . . .	480

5.1.2.1776_UART4_RXNE . . . . .	480
5.1.2.1777_UART4_SBK . . . . .	480
5.1.2.1778_UART4_SCEN . . . . .	481
5.1.2.1779_UART4_SR . . . . .	481
5.1.2.1780_UART4_SR_RESET_VALUE . . . . .	481
5.1.2.1781_UART4_STOP . . . . .	481
5.1.2.1782_UART4_STOP0 . . . . .	481
5.1.2.1783_UART4_STOP1 . . . . .	482
5.1.2.1784_UART4_T8 . . . . .	482
5.1.2.1785_UART4_TC . . . . .	482
5.1.2.1786_UART4_TCIEN . . . . .	482
5.1.2.1787_UART4_TEN . . . . .	482
5.1.2.1788_UART4_TIEN . . . . .	483
5.1.2.1789_UART4_TXE . . . . .	483
5.1.2.1790_UART4_UARTD . . . . .	483
5.1.2.1791_UART4_WAKE . . . . .	483
5.1.2.1792_WWDG . . . . .	483
5.1.2.1793_WWDG_CR . . . . .	484
5.1.2.1794_WWDG_CR_RESET_VALUE . . . . .	484
5.1.2.1795_WWDG_T . . . . .	484
5.1.2.1796_WWDG_T0 . . . . .	484
5.1.2.1797_WWDG_T1 . . . . .	484
5.1.2.1798_WWDG_T2 . . . . .	485
5.1.2.1799_WWDG_T3 . . . . .	485
5.1.2.1800_WWDG_T4 . . . . .	485
5.1.2.1801_WWDG_T5 . . . . .	485
5.1.2.1802_WWDG_T6 . . . . .	485
5.1.2.1803_WWDG_W . . . . .	486
5.1.2.1804_WWDG_W0 . . . . .	486
5.1.2.1805_WWDG_W1 . . . . .	486

5.1.2.1806_WWDG_W2 . . . . .	486
5.1.2.1807_WWDG_W3 . . . . .	486
5.1.2.1808_WWDG_W4 . . . . .	487
5.1.2.1809_WWDG_W5 . . . . .	487
5.1.2.1810_WWDG_W6 . . . . .	487
5.1.2.1811_WWDG_WDGA . . . . .	487
5.1.2.1812_WWDG_WR . . . . .	487
5.1.2.1813_WWDG_WR_RESET_VALUE . . . . .	488
5.1.2.1814ADC1_AddressBase [1/27] . . . . .	488
5.1.2.1815ADC1_AddressBase [2/27] . . . . .	488
5.1.2.1816ADC1_AddressBase [3/27] . . . . .	488
5.1.2.1817ADC1_AddressBase [4/27] . . . . .	488
5.1.2.1818ADC1_AddressBase [5/27] . . . . .	488
5.1.2.1819ADC1_AddressBase [6/27] . . . . .	489
5.1.2.1820ADC1_AddressBase [7/27] . . . . .	489
5.1.2.1821ADC1_AddressBase [8/27] . . . . .	489
5.1.2.1822ADC1_AddressBase [9/27] . . . . .	489
5.1.2.1823ADC1_AddressBase [10/27] . . . . .	489
5.1.2.1824ADC1_AddressBase [11/27] . . . . .	489
5.1.2.1825ADC1_AddressBase [12/27] . . . . .	490
5.1.2.1826ADC1_AddressBase [13/27] . . . . .	490
5.1.2.1827ADC1_AddressBase [14/27] . . . . .	490
5.1.2.1828ADC1_AddressBase [15/27] . . . . .	490
5.1.2.1829ADC1_AddressBase [16/27] . . . . .	490
5.1.2.1830ADC1_AddressBase [17/27] . . . . .	490
5.1.2.1831ADC1_AddressBase [18/27] . . . . .	491
5.1.2.1832ADC1_AddressBase [19/27] . . . . .	491
5.1.2.1833ADC1_AddressBase [20/27] . . . . .	491
5.1.2.1834ADC1_AddressBase [21/27] . . . . .	491
5.1.2.1835ADC1_AddressBase [22/27] . . . . .	491



5.1.2.1836ADC1_AddressBase [23/27]	491
5.1.2.1837ADC1_AddressBase [24/27]	492
5.1.2.1838ADC1_AddressBase [25/27]	492
5.1.2.1839ADC1_AddressBase [26/27]	492
5.1.2.1840ADC1_AddressBase [27/27]	492
5.1.2.1841ADC2_AddressBase [1/43]	492
5.1.2.1842ADC2_AddressBase [2/43]	492
5.1.2.1843ADC2_AddressBase [3/43]	493
5.1.2.1844ADC2_AddressBase [4/43]	493
5.1.2.1845ADC2_AddressBase [5/43]	493
5.1.2.1846ADC2_AddressBase [6/43]	493
5.1.2.1847ADC2_AddressBase [7/43]	493
5.1.2.1848ADC2_AddressBase [8/43]	493
5.1.2.1849ADC2_AddressBase [9/43]	494
5.1.2.1850ADC2_AddressBase [10/43]	494
5.1.2.1851ADC2_AddressBase [11/43]	494
5.1.2.1852ADC2_AddressBase [12/43]	494
5.1.2.1853ADC2_AddressBase [13/43]	494
5.1.2.1854ADC2_AddressBase [14/43]	494
5.1.2.1855ADC2_AddressBase [15/43]	495
5.1.2.1856ADC2_AddressBase [16/43]	495
5.1.2.1857ADC2_AddressBase [17/43]	495
5.1.2.1858ADC2_AddressBase [18/43]	495
5.1.2.1859ADC2_AddressBase [19/43]	495
5.1.2.1860ADC2_AddressBase [20/43]	495
5.1.2.1861ADC2_AddressBase [21/43]	496
5.1.2.1862ADC2_AddressBase [22/43]	496
5.1.2.1863ADC2_AddressBase [23/43]	496
5.1.2.1864ADC2_AddressBase [24/43]	496
5.1.2.1865ADC2_AddressBase [25/43]	496

5.1.2.1866ADC2_AddressBase [26/43]	496
5.1.2.1867ADC2_AddressBase [27/43]	497
5.1.2.1868ADC2_AddressBase [28/43]	497
5.1.2.1869ADC2_AddressBase [29/43]	497
5.1.2.1870ADC2_AddressBase [30/43]	497
5.1.2.1871ADC2_AddressBase [31/43]	497
5.1.2.1872ADC2_AddressBase [32/43]	497
5.1.2.1873ADC2_AddressBase [33/43]	498
5.1.2.1874ADC2_AddressBase [34/43]	498
5.1.2.1875ADC2_AddressBase [35/43]	498
5.1.2.1876ADC2_AddressBase [36/43]	498
5.1.2.1877ADC2_AddressBase [37/43]	498
5.1.2.1878ADC2_AddressBase [38/43]	498
5.1.2.1879ADC2_AddressBase [39/43]	499
5.1.2.1880ADC2_AddressBase [40/43]	499
5.1.2.1881ADC2_AddressBase [41/43]	499
5.1.2.1882ADC2_AddressBase [42/43]	499
5.1.2.1883ADC2_AddressBase [43/43]	499
5.1.2.1884AWU_AddressBase [1/70]	499
5.1.2.1885AWU_AddressBase [2/70]	500
5.1.2.1886AWU_AddressBase [3/70]	500
5.1.2.1887AWU_AddressBase [4/70]	500
5.1.2.1888AWU_AddressBase [5/70]	500
5.1.2.1889AWU_AddressBase [6/70]	500
5.1.2.1890AWU_AddressBase [7/70]	500
5.1.2.1891AWU_AddressBase [8/70]	501
5.1.2.1892AWU_AddressBase [9/70]	501
5.1.2.1893AWU_AddressBase [10/70]	501
5.1.2.1894AWU_AddressBase [11/70]	501
5.1.2.1895AWU_AddressBase [12/70]	501

5.1.2.1896AWU_AddressBase [13/70]	501
5.1.2.1897AWU_AddressBase [14/70]	502
5.1.2.1898AWU_AddressBase [15/70]	502
5.1.2.1899AWU_AddressBase [16/70]	502
5.1.2.1900AWU_AddressBase [17/70]	502
5.1.2.1901AWU_AddressBase [18/70]	502
5.1.2.1902AWU_AddressBase [19/70]	502
5.1.2.1903AWU_AddressBase [20/70]	503
5.1.2.1904AWU_AddressBase [21/70]	503
5.1.2.1905AWU_AddressBase [22/70]	503
5.1.2.1906AWU_AddressBase [23/70]	503
5.1.2.1907AWU_AddressBase [24/70]	503
5.1.2.1908AWU_AddressBase [25/70]	503
5.1.2.1909AWU_AddressBase [26/70]	504
5.1.2.1910AWU_AddressBase [27/70]	504
5.1.2.1911AWU_AddressBase [28/70]	504
5.1.2.1912AWU_AddressBase [29/70]	504
5.1.2.1913AWU_AddressBase [30/70]	504
5.1.2.1914AWU_AddressBase [31/70]	504
5.1.2.1915AWU_AddressBase [32/70]	505
5.1.2.1916AWU_AddressBase [33/70]	505
5.1.2.1917AWU_AddressBase [34/70]	505
5.1.2.1918AWU_AddressBase [35/70]	505
5.1.2.1919AWU_AddressBase [36/70]	505
5.1.2.1920AWU_AddressBase [37/70]	505
5.1.2.1921AWU_AddressBase [38/70]	506
5.1.2.1922AWU_AddressBase [39/70]	506
5.1.2.1923AWU_AddressBase [40/70]	506
5.1.2.1924AWU_AddressBase [41/70]	506
5.1.2.1925AWU_AddressBase [42/70]	506

5.1.2.1926AWU_AddressBase [43/70]	506
5.1.2.1927AWU_AddressBase [44/70]	507
5.1.2.1928AWU_AddressBase [45/70]	507
5.1.2.1929AWU_AddressBase [46/70]	507
5.1.2.1930AWU_AddressBase [47/70]	507
5.1.2.1931AWU_AddressBase [48/70]	507
5.1.2.1932AWU_AddressBase [49/70]	507
5.1.2.1933AWU_AddressBase [50/70]	508
5.1.2.1934AWU_AddressBase [51/70]	508
5.1.2.1935AWU_AddressBase [52/70]	508
5.1.2.1936AWU_AddressBase [53/70]	508
5.1.2.1937AWU_AddressBase [54/70]	508
5.1.2.1938AWU_AddressBase [55/70]	508
5.1.2.1939AWU_AddressBase [56/70]	509
5.1.2.1940AWU_AddressBase [57/70]	509
5.1.2.1941AWU_AddressBase [58/70]	509
5.1.2.1942AWU_AddressBase [59/70]	509
5.1.2.1943AWU_AddressBase [60/70]	509
5.1.2.1944AWU_AddressBase [61/70]	509
5.1.2.1945AWU_AddressBase [62/70]	510
5.1.2.1946AWU_AddressBase [63/70]	510
5.1.2.1947AWU_AddressBase [64/70]	510
5.1.2.1948AWU_AddressBase [65/70]	510
5.1.2.1949AWU_AddressBase [66/70]	510
5.1.2.1950AWU_AddressBase [67/70]	510
5.1.2.1951AWU_AddressBase [68/70]	511
5.1.2.1952AWU_AddressBase [69/70]	511
5.1.2.1953AWU_AddressBase [70/70]	511
5.1.2.1954BEEP_AddressBase [1/70]	511
5.1.2.1955BEEP_AddressBase [2/70]	511

5.1.2.1956BEEP_AddressBase [3/70]	511
5.1.2.1957BEEP_AddressBase [4/70]	512
5.1.2.1958BEEP_AddressBase [5/70]	512
5.1.2.1959BEEP_AddressBase [6/70]	512
5.1.2.1960BEEP_AddressBase [7/70]	512
5.1.2.1961BEEP_AddressBase [8/70]	512
5.1.2.1962BEEP_AddressBase [9/70]	512
5.1.2.1963BEEP_AddressBase [10/70]	513
5.1.2.1964BEEP_AddressBase [11/70]	513
5.1.2.1965BEEP_AddressBase [12/70]	513
5.1.2.1966BEEP_AddressBase [13/70]	513
5.1.2.1967BEEP_AddressBase [14/70]	513
5.1.2.1968BEEP_AddressBase [15/70]	513
5.1.2.1969BEEP_AddressBase [16/70]	514
5.1.2.1970BEEP_AddressBase [17/70]	514
5.1.2.1971BEEP_AddressBase [18/70]	514
5.1.2.1972BEEP_AddressBase [19/70]	514
5.1.2.1973BEEP_AddressBase [20/70]	514
5.1.2.1974BEEP_AddressBase [21/70]	514
5.1.2.1975BEEP_AddressBase [22/70]	515
5.1.2.1976BEEP_AddressBase [23/70]	515
5.1.2.1977BEEP_AddressBase [24/70]	515
5.1.2.1978BEEP_AddressBase [25/70]	515
5.1.2.1979BEEP_AddressBase [26/70]	515
5.1.2.1980BEEP_AddressBase [27/70]	515
5.1.2.1981BEEP_AddressBase [28/70]	516
5.1.2.1982BEEP_AddressBase [29/70]	516
5.1.2.1983BEEP_AddressBase [30/70]	516
5.1.2.1984BEEP_AddressBase [31/70]	516
5.1.2.1985BEEP_AddressBase [32/70]	516

5.1.2.1986BEEP_AddressBase [33/70]	516
5.1.2.1987BEEP_AddressBase [34/70]	517
5.1.2.1988BEEP_AddressBase [35/70]	517
5.1.2.1989BEEP_AddressBase [36/70]	517
5.1.2.1990BEEP_AddressBase [37/70]	517
5.1.2.1991BEEP_AddressBase [38/70]	517
5.1.2.1992BEEP_AddressBase [39/70]	517
5.1.2.1993BEEP_AddressBase [40/70]	518
5.1.2.1994BEEP_AddressBase [41/70]	518
5.1.2.1995BEEP_AddressBase [42/70]	518
5.1.2.1996BEEP_AddressBase [43/70]	518
5.1.2.1997BEEP_AddressBase [44/70]	518
5.1.2.1998BEEP_AddressBase [45/70]	518
5.1.2.1999BEEP_AddressBase [46/70]	519
5.1.2.2000BEEP_AddressBase [47/70]	519
5.1.2.2001BEEP_AddressBase [48/70]	519
5.1.2.2002BEEP_AddressBase [49/70]	519
5.1.2.2003BEEP_AddressBase [50/70]	519
5.1.2.2004BEEP_AddressBase [51/70]	519
5.1.2.2005BEEP_AddressBase [52/70]	520
5.1.2.2006BEEP_AddressBase [53/70]	520
5.1.2.2007BEEP_AddressBase [54/70]	520
5.1.2.2008BEEP_AddressBase [55/70]	520
5.1.2.2009BEEP_AddressBase [56/70]	520
5.1.2.2010BEEP_AddressBase [57/70]	520
5.1.2.2011BEEP_AddressBase [58/70]	521
5.1.2.2012BEEP_AddressBase [59/70]	521
5.1.2.2013BEEP_AddressBase [60/70]	521
5.1.2.2014BEEP_AddressBase [61/70]	521
5.1.2.2015BEEP_AddressBase [62/70]	521

5.1.2.2016BEEP_AddressBase [63/70]	521
5.1.2.2017BEEP_AddressBase [64/70]	522
5.1.2.2018BEEP_AddressBase [65/70]	522
5.1.2.2019BEEP_AddressBase [66/70]	522
5.1.2.2020BEEP_AddressBase [67/70]	522
5.1.2.2021BEEP_AddressBase [68/70]	522
5.1.2.2022BEEP_AddressBase [69/70]	522
5.1.2.2023BEEP_AddressBase [70/70]	523
5.1.2.2024CAN_AddressBase [1/20]	523
5.1.2.2025CAN_AddressBase [2/20]	523
5.1.2.2026CAN_AddressBase [3/20]	523
5.1.2.2027CAN_AddressBase [4/20]	523
5.1.2.2028CAN_AddressBase [5/20]	523
5.1.2.2029CAN_AddressBase [6/20]	524
5.1.2.2030CAN_AddressBase [7/20]	524
5.1.2.2031CAN_AddressBase [8/20]	524
5.1.2.2032CAN_AddressBase [9/20]	524
5.1.2.2033CAN_AddressBase [10/20]	524
5.1.2.2034CAN_AddressBase [11/20]	524
5.1.2.2035CAN_AddressBase [12/20]	525
5.1.2.2036CAN_AddressBase [13/20]	525
5.1.2.2037CAN_AddressBase [14/20]	525
5.1.2.2038CAN_AddressBase [15/20]	525
5.1.2.2039CAN_AddressBase [16/20]	525
5.1.2.2040CAN_AddressBase [17/20]	525
5.1.2.2041CAN_AddressBase [18/20]	526
5.1.2.2042CAN_AddressBase [19/20]	526
5.1.2.2043CAN_AddressBase [20/20]	526
5.1.2.2044CFG_AddressBase [1/70]	526
5.1.2.2045CFG_AddressBase [2/70]	526

5.1.2.2046CFG_AddressBase [3/70]	526
5.1.2.2047CFG_AddressBase [4/70]	527
5.1.2.2048CFG_AddressBase [5/70]	527
5.1.2.2049CFG_AddressBase [6/70]	527
5.1.2.2050CFG_AddressBase [7/70]	527
5.1.2.2051CFG_AddressBase [8/70]	527
5.1.2.2052CFG_AddressBase [9/70]	527
5.1.2.2053CFG_AddressBase [10/70]	528
5.1.2.2054CFG_AddressBase [11/70]	528
5.1.2.2055CFG_AddressBase [12/70]	528
5.1.2.2056CFG_AddressBase [13/70]	528
5.1.2.2057CFG_AddressBase [14/70]	528
5.1.2.2058CFG_AddressBase [15/70]	528
5.1.2.2059CFG_AddressBase [16/70]	529
5.1.2.2060CFG_AddressBase [17/70]	529
5.1.2.2061CFG_AddressBase [18/70]	529
5.1.2.2062CFG_AddressBase [19/70]	529
5.1.2.2063CFG_AddressBase [20/70]	529
5.1.2.2064CFG_AddressBase [21/70]	529
5.1.2.2065CFG_AddressBase [22/70]	530
5.1.2.2066CFG_AddressBase [23/70]	530
5.1.2.2067CFG_AddressBase [24/70]	530
5.1.2.2068CFG_AddressBase [25/70]	530
5.1.2.2069CFG_AddressBase [26/70]	530
5.1.2.2070CFG_AddressBase [27/70]	530
5.1.2.2071CFG_AddressBase [28/70]	531
5.1.2.2072CFG_AddressBase [29/70]	531
5.1.2.2073CFG_AddressBase [30/70]	531
5.1.2.2074CFG_AddressBase [31/70]	531
5.1.2.2075CFG_AddressBase [32/70]	531



5.1.2.2076CFG_AddressBase [33/70] . . . . .	531
5.1.2.2077CFG_AddressBase [34/70] . . . . .	532
5.1.2.2078CFG_AddressBase [35/70] . . . . .	532
5.1.2.2079CFG_AddressBase [36/70] . . . . .	532
5.1.2.2080CFG_AddressBase [37/70] . . . . .	532
5.1.2.2081CFG_AddressBase [38/70] . . . . .	532
5.1.2.2082CFG_AddressBase [39/70] . . . . .	532
5.1.2.2083CFG_AddressBase [40/70] . . . . .	533
5.1.2.2084CFG_AddressBase [41/70] . . . . .	533
5.1.2.2085CFG_AddressBase [42/70] . . . . .	533
5.1.2.2086CFG_AddressBase [43/70] . . . . .	533
5.1.2.2087CFG_AddressBase [44/70] . . . . .	533
5.1.2.2088CFG_AddressBase [45/70] . . . . .	533
5.1.2.2089CFG_AddressBase [46/70] . . . . .	534
5.1.2.2090CFG_AddressBase [47/70] . . . . .	534
5.1.2.2091CFG_AddressBase [48/70] . . . . .	534
5.1.2.2092CFG_AddressBase [49/70] . . . . .	534
5.1.2.2093CFG_AddressBase [50/70] . . . . .	534
5.1.2.2094CFG_AddressBase [51/70] . . . . .	534
5.1.2.2095CFG_AddressBase [52/70] . . . . .	535
5.1.2.2096CFG_AddressBase [53/70] . . . . .	535
5.1.2.2097CFG_AddressBase [54/70] . . . . .	535
5.1.2.2098CFG_AddressBase [55/70] . . . . .	535
5.1.2.2099CFG_AddressBase [56/70] . . . . .	535
5.1.2.2100CFG_AddressBase [57/70] . . . . .	535
5.1.2.2101CFG_AddressBase [58/70] . . . . .	536
5.1.2.2102CFG_AddressBase [59/70] . . . . .	536
5.1.2.2103CFG_AddressBase [60/70] . . . . .	536
5.1.2.2104CFG_AddressBase [61/70] . . . . .	536
5.1.2.2105CFG_AddressBase [62/70] . . . . .	536

5.1.2.2106CFG_AddressBase [63/70] . . . . .	536
5.1.2.2107CFG_AddressBase [64/70] . . . . .	537
5.1.2.2108CFG_AddressBase [65/70] . . . . .	537
5.1.2.2109CFG_AddressBase [66/70] . . . . .	537
5.1.2.2110CFG_AddressBase [67/70] . . . . .	537
5.1.2.2111CFG_AddressBase [68/70] . . . . .	537
5.1.2.2112CFG_AddressBase [69/70] . . . . .	537
5.1.2.2113CFG_AddressBase [70/70] . . . . .	538
5.1.2.2114CLK_AddressBase [1/70] . . . . .	538
5.1.2.2115CLK_AddressBase [2/70] . . . . .	538
5.1.2.2116CLK_AddressBase [3/70] . . . . .	538
5.1.2.2117CLK_AddressBase [4/70] . . . . .	538
5.1.2.2118CLK_AddressBase [5/70] . . . . .	538
5.1.2.2119CLK_AddressBase [6/70] . . . . .	539
5.1.2.2120CLK_AddressBase [7/70] . . . . .	539
5.1.2.2121CLK_AddressBase [8/70] . . . . .	539
5.1.2.2122CLK_AddressBase [9/70] . . . . .	539
5.1.2.2123CLK_AddressBase [10/70] . . . . .	539
5.1.2.2124CLK_AddressBase [11/70] . . . . .	539
5.1.2.2125CLK_AddressBase [12/70] . . . . .	540
5.1.2.2126CLK_AddressBase [13/70] . . . . .	540
5.1.2.2127CLK_AddressBase [14/70] . . . . .	540
5.1.2.2128CLK_AddressBase [15/70] . . . . .	540
5.1.2.2129CLK_AddressBase [16/70] . . . . .	540
5.1.2.2130CLK_AddressBase [17/70] . . . . .	540
5.1.2.2131CLK_AddressBase [18/70] . . . . .	541
5.1.2.2132CLK_AddressBase [19/70] . . . . .	541
5.1.2.2133CLK_AddressBase [20/70] . . . . .	541
5.1.2.2134CLK_AddressBase [21/70] . . . . .	541
5.1.2.2135CLK_AddressBase [22/70] . . . . .	541

5.1.2.2136CLK_AddressBase [23/70] . . . . .	541
5.1.2.2137CLK_AddressBase [24/70] . . . . .	542
5.1.2.2138CLK_AddressBase [25/70] . . . . .	542
5.1.2.2139CLK_AddressBase [26/70] . . . . .	542
5.1.2.2140CLK_AddressBase [27/70] . . . . .	542
5.1.2.2141CLK_AddressBase [28/70] . . . . .	542
5.1.2.2142CLK_AddressBase [29/70] . . . . .	542
5.1.2.2143CLK_AddressBase [30/70] . . . . .	543
5.1.2.2144CLK_AddressBase [31/70] . . . . .	543
5.1.2.2145CLK_AddressBase [32/70] . . . . .	543
5.1.2.2146CLK_AddressBase [33/70] . . . . .	543
5.1.2.2147CLK_AddressBase [34/70] . . . . .	543
5.1.2.2148CLK_AddressBase [35/70] . . . . .	543
5.1.2.2149CLK_AddressBase [36/70] . . . . .	544
5.1.2.2150CLK_AddressBase [37/70] . . . . .	544
5.1.2.2151CLK_AddressBase [38/70] . . . . .	544
5.1.2.2152CLK_AddressBase [39/70] . . . . .	544
5.1.2.2153CLK_AddressBase [40/70] . . . . .	544
5.1.2.2154CLK_AddressBase [41/70] . . . . .	544
5.1.2.2155CLK_AddressBase [42/70] . . . . .	545
5.1.2.2156CLK_AddressBase [43/70] . . . . .	545
5.1.2.2157CLK_AddressBase [44/70] . . . . .	545
5.1.2.2158CLK_AddressBase [45/70] . . . . .	545
5.1.2.2159CLK_AddressBase [46/70] . . . . .	545
5.1.2.2160CLK_AddressBase [47/70] . . . . .	545
5.1.2.2161CLK_AddressBase [48/70] . . . . .	546
5.1.2.2162CLK_AddressBase [49/70] . . . . .	546
5.1.2.2163CLK_AddressBase [50/70] . . . . .	546
5.1.2.2164CLK_AddressBase [51/70] . . . . .	546
5.1.2.2165CLK_AddressBase [52/70] . . . . .	546

5.1.2.2166CLK_AddressBase [53/70] . . . . .	546
5.1.2.2167CLK_AddressBase [54/70] . . . . .	547
5.1.2.2168CLK_AddressBase [55/70] . . . . .	547
5.1.2.2169CLK_AddressBase [56/70] . . . . .	547
5.1.2.2170CLK_AddressBase [57/70] . . . . .	547
5.1.2.2171CLK_AddressBase [58/70] . . . . .	547
5.1.2.2172CLK_AddressBase [59/70] . . . . .	547
5.1.2.2173CLK_AddressBase [60/70] . . . . .	548
5.1.2.2174CLK_AddressBase [61/70] . . . . .	548
5.1.2.2175CLK_AddressBase [62/70] . . . . .	548
5.1.2.2176CLK_AddressBase [63/70] . . . . .	548
5.1.2.2177CLK_AddressBase [64/70] . . . . .	548
5.1.2.2178CLK_AddressBase [65/70] . . . . .	548
5.1.2.2179CLK_AddressBase [66/70] . . . . .	549
5.1.2.2180CLK_AddressBase [67/70] . . . . .	549
5.1.2.2181CLK_AddressBase [68/70] . . . . .	549
5.1.2.2182CLK_AddressBase [69/70] . . . . .	549
5.1.2.2183CLK_AddressBase [70/70] . . . . .	549
5.1.2.2184DISABLE_INTERRUPTS . . . . .	549
5.1.2.2185DM_AddressBase [1/70] . . . . .	550
5.1.2.2186DM_AddressBase [2/70] . . . . .	550
5.1.2.2187DM_AddressBase [3/70] . . . . .	550
5.1.2.2188DM_AddressBase [4/70] . . . . .	550
5.1.2.2189DM_AddressBase [5/70] . . . . .	550
5.1.2.2190DM_AddressBase [6/70] . . . . .	550
5.1.2.2191DM_AddressBase [7/70] . . . . .	551
5.1.2.2192DM_AddressBase [8/70] . . . . .	551
5.1.2.2193DM_AddressBase [9/70] . . . . .	551
5.1.2.2194DM_AddressBase [10/70] . . . . .	551
5.1.2.2195DM_AddressBase [11/70] . . . . .	551

5.1.2.2196DM_AddressBase [12/70]	551
5.1.2.2197DM_AddressBase [13/70]	552
5.1.2.2198DM_AddressBase [14/70]	552
5.1.2.2199DM_AddressBase [15/70]	552
5.1.2.2200DM_AddressBase [16/70]	552
5.1.2.2201DM_AddressBase [17/70]	552
5.1.2.2202DM_AddressBase [18/70]	552
5.1.2.2203DM_AddressBase [19/70]	553
5.1.2.2204DM_AddressBase [20/70]	553
5.1.2.2205DM_AddressBase [21/70]	553
5.1.2.2206DM_AddressBase [22/70]	553
5.1.2.2207DM_AddressBase [23/70]	553
5.1.2.2208DM_AddressBase [24/70]	553
5.1.2.2209DM_AddressBase [25/70]	554
5.1.2.2210DM_AddressBase [26/70]	554
5.1.2.2211DM_AddressBase [27/70]	554
5.1.2.2212DM_AddressBase [28/70]	554
5.1.2.2213DM_AddressBase [29/70]	554
5.1.2.2214DM_AddressBase [30/70]	554
5.1.2.2215DM_AddressBase [31/70]	555
5.1.2.2216DM_AddressBase [32/70]	555
5.1.2.2217DM_AddressBase [33/70]	555
5.1.2.2218DM_AddressBase [34/70]	555
5.1.2.2219DM_AddressBase [35/70]	555
5.1.2.2220DM_AddressBase [36/70]	555
5.1.2.2221DM_AddressBase [37/70]	556
5.1.2.2222DM_AddressBase [38/70]	556
5.1.2.2223DM_AddressBase [39/70]	556
5.1.2.2224DM_AddressBase [40/70]	556
5.1.2.2225DM_AddressBase [41/70]	556

5.1.2.2226DM_AddressBase [42/70]	556
5.1.2.2227DM_AddressBase [43/70]	557
5.1.2.2228DM_AddressBase [44/70]	557
5.1.2.2229DM_AddressBase [45/70]	557
5.1.2.2230DM_AddressBase [46/70]	557
5.1.2.2231DM_AddressBase [47/70]	557
5.1.2.2232DM_AddressBase [48/70]	557
5.1.2.2233DM_AddressBase [49/70]	558
5.1.2.2234DM_AddressBase [50/70]	558
5.1.2.2235DM_AddressBase [51/70]	558
5.1.2.2236DM_AddressBase [52/70]	558
5.1.2.2237DM_AddressBase [53/70]	558
5.1.2.2238DM_AddressBase [54/70]	558
5.1.2.2239DM_AddressBase [55/70]	559
5.1.2.2240DM_AddressBase [56/70]	559
5.1.2.2241DM_AddressBase [57/70]	559
5.1.2.2242DM_AddressBase [58/70]	559
5.1.2.2243DM_AddressBase [59/70]	559
5.1.2.2244DM_AddressBase [60/70]	559
5.1.2.2245DM_AddressBase [61/70]	560
5.1.2.2246DM_AddressBase [62/70]	560
5.1.2.2247DM_AddressBase [63/70]	560
5.1.2.2248DM_AddressBase [64/70]	560
5.1.2.2249DM_AddressBase [65/70]	560
5.1.2.2250DM_AddressBase [66/70]	560
5.1.2.2251DM_AddressBase [67/70]	561
5.1.2.2252DM_AddressBase [68/70]	561
5.1.2.2253DM_AddressBase [69/70]	561
5.1.2.2254DM_AddressBase [70/70]	561
5.1.2.2255ENABLE_INTERRUPTS	561

5.1.2.2256	ENTER_HALT	561
5.1.2.2257	EXTI_AddressBase [1/70]	562
5.1.2.2258	EXTI_AddressBase [2/70]	562
5.1.2.2259	EXTI_AddressBase [3/70]	562
5.1.2.2260	EXTI_AddressBase [4/70]	562
5.1.2.2261	EXTI_AddressBase [5/70]	562
5.1.2.2262	EXTI_AddressBase [6/70]	562
5.1.2.2263	EXTI_AddressBase [7/70]	563
5.1.2.2264	EXTI_AddressBase [8/70]	563
5.1.2.2265	EXTI_AddressBase [9/70]	563
5.1.2.2266	EXTI_AddressBase [10/70]	563
5.1.2.2267	EXTI_AddressBase [11/70]	563
5.1.2.2268	EXTI_AddressBase [12/70]	563
5.1.2.2269	EXTI_AddressBase [13/70]	564
5.1.2.2270	EXTI_AddressBase [14/70]	564
5.1.2.2271	EXTI_AddressBase [15/70]	564
5.1.2.2272	EXTI_AddressBase [16/70]	564
5.1.2.2273	EXTI_AddressBase [17/70]	564
5.1.2.2274	EXTI_AddressBase [18/70]	564
5.1.2.2275	EXTI_AddressBase [19/70]	565
5.1.2.2276	EXTI_AddressBase [20/70]	565
5.1.2.2277	EXTI_AddressBase [21/70]	565
5.1.2.2278	EXTI_AddressBase [22/70]	565
5.1.2.2279	EXTI_AddressBase [23/70]	565
5.1.2.2280	EXTI_AddressBase [24/70]	565
5.1.2.2281	EXTI_AddressBase [25/70]	566
5.1.2.2282	EXTI_AddressBase [26/70]	566
5.1.2.2283	EXTI_AddressBase [27/70]	566
5.1.2.2284	EXTI_AddressBase [28/70]	566
5.1.2.2285	EXTI_AddressBase [29/70]	566

5.1.2.2286EXTI_AddressBase [30/70]	566
5.1.2.2287EXTI_AddressBase [31/70]	567
5.1.2.2288EXTI_AddressBase [32/70]	567
5.1.2.2289EXTI_AddressBase [33/70]	567
5.1.2.2290EXTI_AddressBase [34/70]	567
5.1.2.2291EXTI_AddressBase [35/70]	567
5.1.2.2292EXTI_AddressBase [36/70]	567
5.1.2.2293EXTI_AddressBase [37/70]	568
5.1.2.2294EXTI_AddressBase [38/70]	568
5.1.2.2295EXTI_AddressBase [39/70]	568
5.1.2.2296EXTI_AddressBase [40/70]	568
5.1.2.2297EXTI_AddressBase [41/70]	568
5.1.2.2298EXTI_AddressBase [42/70]	568
5.1.2.2299EXTI_AddressBase [43/70]	569
5.1.2.2300EXTI_AddressBase [44/70]	569
5.1.2.2301EXTI_AddressBase [45/70]	569
5.1.2.2302EXTI_AddressBase [46/70]	569
5.1.2.2303EXTI_AddressBase [47/70]	569
5.1.2.2304EXTI_AddressBase [48/70]	569
5.1.2.2305EXTI_AddressBase [49/70]	570
5.1.2.2306EXTI_AddressBase [50/70]	570
5.1.2.2307EXTI_AddressBase [51/70]	570
5.1.2.2308EXTI_AddressBase [52/70]	570
5.1.2.2309EXTI_AddressBase [53/70]	570
5.1.2.2310EXTI_AddressBase [54/70]	570
5.1.2.2311EXTI_AddressBase [55/70]	571
5.1.2.2312EXTI_AddressBase [56/70]	571
5.1.2.2313EXTI_AddressBase [57/70]	571
5.1.2.2314EXTI_AddressBase [58/70]	571
5.1.2.2315EXTI_AddressBase [59/70]	571



5.1.2.2316EXTI_AddressBase [60/70]	571
5.1.2.2317EXTI_AddressBase [61/70]	572
5.1.2.2318EXTI_AddressBase [62/70]	572
5.1.2.2319EXTI_AddressBase [63/70]	572
5.1.2.2320EXTI_AddressBase [64/70]	572
5.1.2.2321EXTI_AddressBase [65/70]	572
5.1.2.2322EXTI_AddressBase [66/70]	572
5.1.2.2323EXTI_AddressBase [67/70]	573
5.1.2.2324EXTI_AddressBase [68/70]	573
5.1.2.2325EXTI_AddressBase [69/70]	573
5.1.2.2326EXTI_AddressBase [70/70]	573
5.1.2.2327FLASH_AddressBase [1/70]	573
5.1.2.2328FLASH_AddressBase [2/70]	573
5.1.2.2329FLASH_AddressBase [3/70]	574
5.1.2.2330FLASH_AddressBase [4/70]	574
5.1.2.2331FLASH_AddressBase [5/70]	574
5.1.2.2332FLASH_AddressBase [6/70]	574
5.1.2.2333FLASH_AddressBase [7/70]	574
5.1.2.2334FLASH_AddressBase [8/70]	574
5.1.2.2335FLASH_AddressBase [9/70]	575
5.1.2.2336FLASH_AddressBase [10/70]	575
5.1.2.2337FLASH_AddressBase [11/70]	575
5.1.2.2338FLASH_AddressBase [12/70]	575
5.1.2.2339FLASH_AddressBase [13/70]	575
5.1.2.2340FLASH_AddressBase [14/70]	575
5.1.2.2341FLASH_AddressBase [15/70]	576
5.1.2.2342FLASH_AddressBase [16/70]	576
5.1.2.2343FLASH_AddressBase [17/70]	576
5.1.2.2344FLASH_AddressBase [18/70]	576
5.1.2.2345FLASH_AddressBase [19/70]	576

5.1.2.2346	FLASH_AddressBase [20/70]	576
5.1.2.2347	FLASH_AddressBase [21/70]	577
5.1.2.2348	FLASH_AddressBase [22/70]	577
5.1.2.2349	FLASH_AddressBase [23/70]	577
5.1.2.2350	FLASH_AddressBase [24/70]	577
5.1.2.2351	FLASH_AddressBase [25/70]	577
5.1.2.2352	FLASH_AddressBase [26/70]	577
5.1.2.2353	FLASH_AddressBase [27/70]	578
5.1.2.2354	FLASH_AddressBase [28/70]	578
5.1.2.2355	FLASH_AddressBase [29/70]	578
5.1.2.2356	FLASH_AddressBase [30/70]	578
5.1.2.2357	FLASH_AddressBase [31/70]	578
5.1.2.2358	FLASH_AddressBase [32/70]	578
5.1.2.2359	FLASH_AddressBase [33/70]	579
5.1.2.2360	FLASH_AddressBase [34/70]	579
5.1.2.2361	FLASH_AddressBase [35/70]	579
5.1.2.2362	FLASH_AddressBase [36/70]	579
5.1.2.2363	FLASH_AddressBase [37/70]	579
5.1.2.2364	FLASH_AddressBase [38/70]	579
5.1.2.2365	FLASH_AddressBase [39/70]	580
5.1.2.2366	FLASH_AddressBase [40/70]	580
5.1.2.2367	FLASH_AddressBase [41/70]	580
5.1.2.2368	FLASH_AddressBase [42/70]	580
5.1.2.2369	FLASH_AddressBase [43/70]	580
5.1.2.2370	FLASH_AddressBase [44/70]	580
5.1.2.2371	FLASH_AddressBase [45/70]	581
5.1.2.2372	FLASH_AddressBase [46/70]	581
5.1.2.2373	FLASH_AddressBase [47/70]	581
5.1.2.2374	FLASH_AddressBase [48/70]	581
5.1.2.2375	FLASH_AddressBase [49/70]	581

5.1.2.2376FLASH_AddressBase [50/70]	581
5.1.2.2377FLASH_AddressBase [51/70]	582
5.1.2.2378FLASH_AddressBase [52/70]	582
5.1.2.2379FLASH_AddressBase [53/70]	582
5.1.2.2380FLASH_AddressBase [54/70]	582
5.1.2.2381FLASH_AddressBase [55/70]	582
5.1.2.2382FLASH_AddressBase [56/70]	582
5.1.2.2383FLASH_AddressBase [57/70]	583
5.1.2.2384FLASH_AddressBase [58/70]	583
5.1.2.2385FLASH_AddressBase [59/70]	583
5.1.2.2386FLASH_AddressBase [60/70]	583
5.1.2.2387FLASH_AddressBase [61/70]	583
5.1.2.2388FLASH_AddressBase [62/70]	583
5.1.2.2389FLASH_AddressBase [63/70]	584
5.1.2.2390FLASH_AddressBase [64/70]	584
5.1.2.2391FLASH_AddressBase [65/70]	584
5.1.2.2392FLASH_AddressBase [66/70]	584
5.1.2.2393FLASH_AddressBase [67/70]	584
5.1.2.2394FLASH_AddressBase [68/70]	584
5.1.2.2395FLASH_AddressBase [69/70]	585
5.1.2.2396FLASH_AddressBase [70/70]	585
5.1.2.23972C_AddressBase [1/70]	585
5.1.2.23982C_AddressBase [2/70]	585
5.1.2.23992C_AddressBase [3/70]	585
5.1.2.24002C_AddressBase [4/70]	585
5.1.2.24012C_AddressBase [5/70]	586
5.1.2.24022C_AddressBase [6/70]	586
5.1.2.24032C_AddressBase [7/70]	586
5.1.2.24042C_AddressBase [8/70]	586
5.1.2.24052C_AddressBase [9/70]	586

5.1.2.24062C_AddressBase [10/70]	586
5.1.2.24072C_AddressBase [11/70]	587
5.1.2.24082C_AddressBase [12/70]	587
5.1.2.24092C_AddressBase [13/70]	587
5.1.2.24102C_AddressBase [14/70]	587
5.1.2.24112C_AddressBase [15/70]	587
5.1.2.24122C_AddressBase [16/70]	587
5.1.2.24132C_AddressBase [17/70]	588
5.1.2.24142C_AddressBase [18/70]	588
5.1.2.24152C_AddressBase [19/70]	588
5.1.2.24162C_AddressBase [20/70]	588
5.1.2.24172C_AddressBase [21/70]	588
5.1.2.24182C_AddressBase [22/70]	588
5.1.2.24192C_AddressBase [23/70]	589
5.1.2.24202C_AddressBase [24/70]	589
5.1.2.24212C_AddressBase [25/70]	589
5.1.2.24222C_AddressBase [26/70]	589
5.1.2.24232C_AddressBase [27/70]	589
5.1.2.24242C_AddressBase [28/70]	589
5.1.2.24252C_AddressBase [29/70]	590
5.1.2.24262C_AddressBase [30/70]	590
5.1.2.24272C_AddressBase [31/70]	590
5.1.2.24282C_AddressBase [32/70]	590
5.1.2.24292C_AddressBase [33/70]	590
5.1.2.24302C_AddressBase [34/70]	590
5.1.2.24312C_AddressBase [35/70]	591
5.1.2.24322C_AddressBase [36/70]	591
5.1.2.24332C_AddressBase [37/70]	591
5.1.2.24342C_AddressBase [38/70]	591
5.1.2.24352C_AddressBase [39/70]	591

5.1.2.24362C_AddressBase [40/70]	591
5.1.2.24372C_AddressBase [41/70]	592
5.1.2.24382C_AddressBase [42/70]	592
5.1.2.24392C_AddressBase [43/70]	592
5.1.2.24402C_AddressBase [44/70]	592
5.1.2.24412C_AddressBase [45/70]	592
5.1.2.24422C_AddressBase [46/70]	592
5.1.2.24432C_AddressBase [47/70]	593
5.1.2.24442C_AddressBase [48/70]	593
5.1.2.24452C_AddressBase [49/70]	593
5.1.2.24462C_AddressBase [50/70]	593
5.1.2.24472C_AddressBase [51/70]	593
5.1.2.24482C_AddressBase [52/70]	593
5.1.2.24492C_AddressBase [53/70]	594
5.1.2.24502C_AddressBase [54/70]	594
5.1.2.24512C_AddressBase [55/70]	594
5.1.2.24522C_AddressBase [56/70]	594
5.1.2.24532C_AddressBase [57/70]	594
5.1.2.24542C_AddressBase [58/70]	594
5.1.2.24552C_AddressBase [59/70]	595
5.1.2.24562C_AddressBase [60/70]	595
5.1.2.24572C_AddressBase [61/70]	595
5.1.2.24582C_AddressBase [62/70]	595
5.1.2.24592C_AddressBase [63/70]	595
5.1.2.24602C_AddressBase [64/70]	595
5.1.2.24612C_AddressBase [65/70]	596
5.1.2.24622C_AddressBase [66/70]	596
5.1.2.24632C_AddressBase [67/70]	596
5.1.2.24642C_AddressBase [68/70]	596
5.1.2.24652C_AddressBase [69/70]	596

5.1.2.246d2C_AddressBase [70/70]	596
5.1.2.2467SR_HANDLER	597
5.1.2.2468SR_HANDLER_TRAP	597
5.1.2.2469TC_AddressBase [1/70]	597
5.1.2.2470TC_AddressBase [2/70]	597
5.1.2.2471ITC_AddressBase [3/70]	597
5.1.2.2472TC_AddressBase [4/70]	598
5.1.2.2473TC_AddressBase [5/70]	598
5.1.2.2474TC_AddressBase [6/70]	598
5.1.2.2475TC_AddressBase [7/70]	598
5.1.2.2476TC_AddressBase [8/70]	598
5.1.2.2477TC_AddressBase [9/70]	598
5.1.2.2478TC_AddressBase [10/70]	599
5.1.2.2479TC_AddressBase [11/70]	599
5.1.2.2480TC_AddressBase [12/70]	599
5.1.2.2481ITC_AddressBase [13/70]	599
5.1.2.2482TC_AddressBase [14/70]	599
5.1.2.2483TC_AddressBase [15/70]	599
5.1.2.2484TC_AddressBase [16/70]	600
5.1.2.2485TC_AddressBase [17/70]	600
5.1.2.2486TC_AddressBase [18/70]	600
5.1.2.2487TC_AddressBase [19/70]	600
5.1.2.2488TC_AddressBase [20/70]	600
5.1.2.2489TC_AddressBase [21/70]	600
5.1.2.2490TC_AddressBase [22/70]	601
5.1.2.2491ITC_AddressBase [23/70]	601
5.1.2.2492TC_AddressBase [24/70]	601
5.1.2.2493TC_AddressBase [25/70]	601
5.1.2.2494TC_AddressBase [26/70]	601
5.1.2.2495TC_AddressBase [27/70]	601

5.1.2.2496TC_AddressBase [28/70]	602
5.1.2.2497TC_AddressBase [29/70]	602
5.1.2.2498TC_AddressBase [30/70]	602
5.1.2.2499TC_AddressBase [31/70]	602
5.1.2.2500TC_AddressBase [32/70]	602
5.1.2.2501TC_AddressBase [33/70]	602
5.1.2.2502TC_AddressBase [34/70]	603
5.1.2.2503TC_AddressBase [35/70]	603
5.1.2.2504TC_AddressBase [36/70]	603
5.1.2.2505TC_AddressBase [37/70]	603
5.1.2.2506TC_AddressBase [38/70]	603
5.1.2.2507TC_AddressBase [39/70]	603
5.1.2.2508TC_AddressBase [40/70]	604
5.1.2.2509TC_AddressBase [41/70]	604
5.1.2.2510TC_AddressBase [42/70]	604
5.1.2.2511TC_AddressBase [43/70]	604
5.1.2.2512TC_AddressBase [44/70]	604
5.1.2.2513TC_AddressBase [45/70]	604
5.1.2.2514TC_AddressBase [46/70]	605
5.1.2.2515TC_AddressBase [47/70]	605
5.1.2.2516TC_AddressBase [48/70]	605
5.1.2.2517TC_AddressBase [49/70]	605
5.1.2.2518TC_AddressBase [50/70]	605
5.1.2.2519TC_AddressBase [51/70]	605
5.1.2.2520TC_AddressBase [52/70]	606
5.1.2.2521TC_AddressBase [53/70]	606
5.1.2.2522TC_AddressBase [54/70]	606
5.1.2.2523TC_AddressBase [55/70]	606
5.1.2.2524TC_AddressBase [56/70]	606
5.1.2.2525TC_AddressBase [57/70]	606

5.1.2.2526TC_AddressBase [58/70]	607
5.1.2.2527TC_AddressBase [59/70]	607
5.1.2.2528TC_AddressBase [60/70]	607
5.1.2.2529TC_AddressBase [61/70]	607
5.1.2.2530TC_AddressBase [62/70]	607
5.1.2.2531TC_AddressBase [63/70]	607
5.1.2.2532TC_AddressBase [64/70]	608
5.1.2.2533TC_AddressBase [65/70]	608
5.1.2.2534TC_AddressBase [66/70]	608
5.1.2.2535TC_AddressBase [67/70]	608
5.1.2.2536TC_AddressBase [68/70]	608
5.1.2.2537TC_AddressBase [69/70]	608
5.1.2.2538TC_AddressBase [70/70]	609
5.1.2.2539WDG_AddressBase [1/70]	609
5.1.2.2540WDG_AddressBase [2/70]	609
5.1.2.2541WDG_AddressBase [3/70]	609
5.1.2.2542WDG_AddressBase [4/70]	609
5.1.2.2543WDG_AddressBase [5/70]	609
5.1.2.2544WDG_AddressBase [6/70]	610
5.1.2.2545WDG_AddressBase [7/70]	610
5.1.2.2546WDG_AddressBase [8/70]	610
5.1.2.2547WDG_AddressBase [9/70]	610
5.1.2.2548WDG_AddressBase [10/70]	610
5.1.2.2549WDG_AddressBase [11/70]	610
5.1.2.2550WDG_AddressBase [12/70]	611
5.1.2.2551WDG_AddressBase [13/70]	611
5.1.2.2552WDG_AddressBase [14/70]	611
5.1.2.2553WDG_AddressBase [15/70]	611
5.1.2.2554WDG_AddressBase [16/70]	611
5.1.2.2555WDG_AddressBase [17/70]	611



5.1.2.2556WDG_AddressBase [18/70]	612
5.1.2.2557WDG_AddressBase [19/70]	612
5.1.2.2558WDG_AddressBase [20/70]	612
5.1.2.2559WDG_AddressBase [21/70]	612
5.1.2.2560WDG_AddressBase [22/70]	612
5.1.2.2561IWDG_AddressBase [23/70]	612
5.1.2.2562WDG_AddressBase [24/70]	613
5.1.2.2563WDG_AddressBase [25/70]	613
5.1.2.2564IWDG_AddressBase [26/70]	613
5.1.2.2565WDG_AddressBase [27/70]	613
5.1.2.2566WDG_AddressBase [28/70]	613
5.1.2.2567WDG_AddressBase [29/70]	613
5.1.2.2568WDG_AddressBase [30/70]	614
5.1.2.2569WDG_AddressBase [31/70]	614
5.1.2.2570IWDG_AddressBase [32/70]	614
5.1.2.2571IWDG_AddressBase [33/70]	614
5.1.2.2572WDG_AddressBase [34/70]	614
5.1.2.2573WDG_AddressBase [35/70]	614
5.1.2.2574IWDG_AddressBase [36/70]	615
5.1.2.2575WDG_AddressBase [37/70]	615
5.1.2.2576WDG_AddressBase [38/70]	615
5.1.2.2577WDG_AddressBase [39/70]	615
5.1.2.2578WDG_AddressBase [40/70]	615
5.1.2.2579WDG_AddressBase [41/70]	615
5.1.2.2580WDG_AddressBase [42/70]	616
5.1.2.2581IWDG_AddressBase [43/70]	616
5.1.2.2582WDG_AddressBase [44/70]	616
5.1.2.2583WDG_AddressBase [45/70]	616
5.1.2.2584IWDG_AddressBase [46/70]	616
5.1.2.2585WDG_AddressBase [47/70]	616

5.1.2.2586WDG_AddressBase [48/70]	617
5.1.2.2587WDG_AddressBase [49/70]	617
5.1.2.2588WDG_AddressBase [50/70]	617
5.1.2.2589WDG_AddressBase [51/70]	617
5.1.2.2590WDG_AddressBase [52/70]	617
5.1.2.2591WDG_AddressBase [53/70]	617
5.1.2.2592WDG_AddressBase [54/70]	618
5.1.2.2593WDG_AddressBase [55/70]	618
5.1.2.2594WDG_AddressBase [56/70]	618
5.1.2.2595WDG_AddressBase [57/70]	618
5.1.2.2596WDG_AddressBase [58/70]	618
5.1.2.2597WDG_AddressBase [59/70]	618
5.1.2.2598WDG_AddressBase [60/70]	619
5.1.2.2599WDG_AddressBase [61/70]	619
5.1.2.2600WDG_AddressBase [62/70]	619
5.1.2.2601WDG_AddressBase [63/70]	619
5.1.2.2602WDG_AddressBase [64/70]	619
5.1.2.2603WDG_AddressBase [65/70]	619
5.1.2.2604WDG_AddressBase [66/70]	620
5.1.2.2605WDG_AddressBase [67/70]	620
5.1.2.2606WDG_AddressBase [68/70]	620
5.1.2.2607WDG_AddressBase [69/70]	620
5.1.2.2608WDG_AddressBase [70/70]	620
5.1.2.2609NOP	620
5.1.2.2610OPT_AddressBase [1/70]	621
5.1.2.2611OPT_AddressBase [2/70]	621
5.1.2.2612OPT_AddressBase [3/70]	621
5.1.2.2613OPT_AddressBase [4/70]	621
5.1.2.2614OPT_AddressBase [5/70]	621
5.1.2.2615OPT_AddressBase [6/70]	621

5.1.2.2616OPT_AddressBase [7/70]	622
5.1.2.2617OPT_AddressBase [8/70]	622
5.1.2.2618OPT_AddressBase [9/70]	622
5.1.2.2619OPT_AddressBase [10/70]	622
5.1.2.2620OPT_AddressBase [11/70]	622
5.1.2.2621OPT_AddressBase [12/70]	622
5.1.2.2622OPT_AddressBase [13/70]	623
5.1.2.2623OPT_AddressBase [14/70]	623
5.1.2.2624OPT_AddressBase [15/70]	623
5.1.2.2625OPT_AddressBase [16/70]	623
5.1.2.2626OPT_AddressBase [17/70]	623
5.1.2.2627OPT_AddressBase [18/70]	623
5.1.2.2628OPT_AddressBase [19/70]	624
5.1.2.2629OPT_AddressBase [20/70]	624
5.1.2.2630OPT_AddressBase [21/70]	624
5.1.2.2631OPT_AddressBase [22/70]	624
5.1.2.2632OPT_AddressBase [23/70]	624
5.1.2.2633OPT_AddressBase [24/70]	624
5.1.2.2634OPT_AddressBase [25/70]	625
5.1.2.2635OPT_AddressBase [26/70]	625
5.1.2.2636OPT_AddressBase [27/70]	625
5.1.2.2637OPT_AddressBase [28/70]	625
5.1.2.2638OPT_AddressBase [29/70]	625
5.1.2.2639OPT_AddressBase [30/70]	625
5.1.2.2640OPT_AddressBase [31/70]	626
5.1.2.2641OPT_AddressBase [32/70]	626
5.1.2.2642OPT_AddressBase [33/70]	626
5.1.2.2643OPT_AddressBase [34/70]	626
5.1.2.2644OPT_AddressBase [35/70]	626
5.1.2.2645OPT_AddressBase [36/70]	626

5.1.2.2646OPT_AddressBase [37/70] . . . . .	627
5.1.2.2647OPT_AddressBase [38/70] . . . . .	627
5.1.2.2648OPT_AddressBase [39/70] . . . . .	627
5.1.2.2649OPT_AddressBase [40/70] . . . . .	627
5.1.2.2650OPT_AddressBase [41/70] . . . . .	627
5.1.2.2651OPT_AddressBase [42/70] . . . . .	627
5.1.2.2652OPT_AddressBase [43/70] . . . . .	628
5.1.2.2653OPT_AddressBase [44/70] . . . . .	628
5.1.2.2654OPT_AddressBase [45/70] . . . . .	628
5.1.2.2655OPT_AddressBase [46/70] . . . . .	628
5.1.2.2656OPT_AddressBase [47/70] . . . . .	628
5.1.2.2657OPT_AddressBase [48/70] . . . . .	628
5.1.2.2658OPT_AddressBase [49/70] . . . . .	629
5.1.2.2659OPT_AddressBase [50/70] . . . . .	629
5.1.2.2660OPT_AddressBase [51/70] . . . . .	629
5.1.2.2661OPT_AddressBase [52/70] . . . . .	629
5.1.2.2662OPT_AddressBase [53/70] . . . . .	629
5.1.2.2663OPT_AddressBase [54/70] . . . . .	629
5.1.2.2664OPT_AddressBase [55/70] . . . . .	630
5.1.2.2665OPT_AddressBase [56/70] . . . . .	630
5.1.2.2666OPT_AddressBase [57/70] . . . . .	630
5.1.2.2667OPT_AddressBase [58/70] . . . . .	630
5.1.2.2668OPT_AddressBase [59/70] . . . . .	630
5.1.2.2669OPT_AddressBase [60/70] . . . . .	630
5.1.2.2670OPT_AddressBase [61/70] . . . . .	631
5.1.2.2671OPT_AddressBase [62/70] . . . . .	631
5.1.2.2672OPT_AddressBase [63/70] . . . . .	631
5.1.2.2673OPT_AddressBase [64/70] . . . . .	631
5.1.2.2674OPT_AddressBase [65/70] . . . . .	631
5.1.2.2675OPT_AddressBase [66/70] . . . . .	631

5.1.2.2676OPT_AddressBase [67/70] . . . . .	632
5.1.2.2677OPT_AddressBase [68/70] . . . . .	632
5.1.2.2678OPT_AddressBase [69/70] . . . . .	632
5.1.2.2679OPT_AddressBase [70/70] . . . . .	632
5.1.2.2680PORTA_AddressBase [1/70] . . . . .	632
5.1.2.2681PORTA_AddressBase [2/70] . . . . .	632
5.1.2.2682PORTA_AddressBase [3/70] . . . . .	633
5.1.2.2683PORTA_AddressBase [4/70] . . . . .	633
5.1.2.2684PORTA_AddressBase [5/70] . . . . .	633
5.1.2.2685PORTA_AddressBase [6/70] . . . . .	633
5.1.2.2686PORTA_AddressBase [7/70] . . . . .	633
5.1.2.2687PORTA_AddressBase [8/70] . . . . .	633
5.1.2.2688PORTA_AddressBase [9/70] . . . . .	634
5.1.2.2689PORTA_AddressBase [10/70] . . . . .	634
5.1.2.2690PORTA_AddressBase [11/70] . . . . .	634
5.1.2.2691PORTA_AddressBase [12/70] . . . . .	634
5.1.2.2692PORTA_AddressBase [13/70] . . . . .	634
5.1.2.2693PORTA_AddressBase [14/70] . . . . .	634
5.1.2.2694PORTA_AddressBase [15/70] . . . . .	635
5.1.2.2695PORTA_AddressBase [16/70] . . . . .	635
5.1.2.2696PORTA_AddressBase [17/70] . . . . .	635
5.1.2.2697PORTA_AddressBase [18/70] . . . . .	635
5.1.2.2698PORTA_AddressBase [19/70] . . . . .	635
5.1.2.2699PORTA_AddressBase [20/70] . . . . .	635
5.1.2.2700PORTA_AddressBase [21/70] . . . . .	636
5.1.2.2701PORTA_AddressBase [22/70] . . . . .	636
5.1.2.2702PORTA_AddressBase [23/70] . . . . .	636
5.1.2.2703PORTA_AddressBase [24/70] . . . . .	636
5.1.2.2704PORTA_AddressBase [25/70] . . . . .	636
5.1.2.2705PORTA_AddressBase [26/70] . . . . .	636

5.1.2.2706	PORTA_AddressBase [27/70]	637
5.1.2.2707	PORTA_AddressBase [28/70]	637
5.1.2.2708	PORTA_AddressBase [29/70]	637
5.1.2.2709	PORTA_AddressBase [30/70]	637
5.1.2.2710	PORTA_AddressBase [31/70]	637
5.1.2.2711	PORTA_AddressBase [32/70]	637
5.1.2.2712	PORTA_AddressBase [33/70]	638
5.1.2.2713	PORTA_AddressBase [34/70]	638
5.1.2.2714	PORTA_AddressBase [35/70]	638
5.1.2.2715	PORTA_AddressBase [36/70]	638
5.1.2.2716	PORTA_AddressBase [37/70]	638
5.1.2.2717	PORTA_AddressBase [38/70]	638
5.1.2.2718	PORTA_AddressBase [39/70]	639
5.1.2.2719	PORTA_AddressBase [40/70]	639
5.1.2.2720	PORTA_AddressBase [41/70]	639
5.1.2.2721	PORTA_AddressBase [42/70]	639
5.1.2.2722	PORTA_AddressBase [43/70]	639
5.1.2.2723	PORTA_AddressBase [44/70]	639
5.1.2.2724	PORTA_AddressBase [45/70]	640
5.1.2.2725	PORTA_AddressBase [46/70]	640
5.1.2.2726	PORTA_AddressBase [47/70]	640
5.1.2.2727	PORTA_AddressBase [48/70]	640
5.1.2.2728	PORTA_AddressBase [49/70]	640
5.1.2.2729	PORTA_AddressBase [50/70]	640
5.1.2.2730	PORTA_AddressBase [51/70]	641
5.1.2.2731	PORTA_AddressBase [52/70]	641
5.1.2.2732	PORTA_AddressBase [53/70]	641
5.1.2.2733	PORTA_AddressBase [54/70]	641
5.1.2.2734	PORTA_AddressBase [55/70]	641
5.1.2.2735	PORTA_AddressBase [56/70]	641

5.1.2.2736	PORTA_AddressBase [57/70]	642
5.1.2.2737	PORTA_AddressBase [58/70]	642
5.1.2.2738	PORTA_AddressBase [59/70]	642
5.1.2.2739	PORTA_AddressBase [60/70]	642
5.1.2.2740	PORTA_AddressBase [61/70]	642
5.1.2.2741	PORTA_AddressBase [62/70]	642
5.1.2.2742	PORTA_AddressBase [63/70]	643
5.1.2.2743	PORTA_AddressBase [64/70]	643
5.1.2.2744	PORTA_AddressBase [65/70]	643
5.1.2.2745	PORTA_AddressBase [66/70]	643
5.1.2.2746	PORTA_AddressBase [67/70]	643
5.1.2.2747	PORTA_AddressBase [68/70]	643
5.1.2.2748	PORTA_AddressBase [69/70]	644
5.1.2.2749	PORTA_AddressBase [70/70]	644
5.1.2.2750	PORTB_AddressBase [1/70]	644
5.1.2.2751	PORTB_AddressBase [2/70]	644
5.1.2.2752	PORTB_AddressBase [3/70]	644
5.1.2.2753	PORTB_AddressBase [4/70]	644
5.1.2.2754	PORTB_AddressBase [5/70]	645
5.1.2.2755	PORTB_AddressBase [6/70]	645
5.1.2.2756	PORTB_AddressBase [7/70]	645
5.1.2.2757	PORTB_AddressBase [8/70]	645
5.1.2.2758	PORTB_AddressBase [9/70]	645
5.1.2.2759	PORTB_AddressBase [10/70]	645
5.1.2.2760	PORTB_AddressBase [11/70]	646
5.1.2.2761	PORTB_AddressBase [12/70]	646
5.1.2.2762	PORTB_AddressBase [13/70]	646
5.1.2.2763	PORTB_AddressBase [14/70]	646
5.1.2.2764	PORTB_AddressBase [15/70]	646
5.1.2.2765	PORTB_AddressBase [16/70]	646

5.1.2.2766	PORTB_AddressBase [17/70]	647
5.1.2.2767	PORTB_AddressBase [18/70]	647
5.1.2.2768	PORTB_AddressBase [19/70]	647
5.1.2.2769	PORTB_AddressBase [20/70]	647
5.1.2.2770	PORTB_AddressBase [21/70]	647
5.1.2.2771	PORTB_AddressBase [22/70]	647
5.1.2.2772	PORTB_AddressBase [23/70]	648
5.1.2.2773	PORTB_AddressBase [24/70]	648
5.1.2.2774	PORTB_AddressBase [25/70]	648
5.1.2.2775	PORTB_AddressBase [26/70]	648
5.1.2.2776	PORTB_AddressBase [27/70]	648
5.1.2.2777	PORTB_AddressBase [28/70]	648
5.1.2.2778	PORTB_AddressBase [29/70]	649
5.1.2.2779	PORTB_AddressBase [30/70]	649
5.1.2.2780	PORTB_AddressBase [31/70]	649
5.1.2.2781	PORTB_AddressBase [32/70]	649
5.1.2.2782	PORTB_AddressBase [33/70]	649
5.1.2.2783	PORTB_AddressBase [34/70]	649
5.1.2.2784	PORTB_AddressBase [35/70]	650
5.1.2.2785	PORTB_AddressBase [36/70]	650
5.1.2.2786	PORTB_AddressBase [37/70]	650
5.1.2.2787	PORTB_AddressBase [38/70]	650
5.1.2.2788	PORTB_AddressBase [39/70]	650
5.1.2.2789	PORTB_AddressBase [40/70]	650
5.1.2.2790	PORTB_AddressBase [41/70]	651
5.1.2.2791	PORTB_AddressBase [42/70]	651
5.1.2.2792	PORTB_AddressBase [43/70]	651
5.1.2.2793	PORTB_AddressBase [44/70]	651
5.1.2.2794	PORTB_AddressBase [45/70]	651
5.1.2.2795	PORTB_AddressBase [46/70]	651



5.1.2.2796	PORTB_AddressBase [47/70]	652
5.1.2.2797	PORTB_AddressBase [48/70]	652
5.1.2.2798	PORTB_AddressBase [49/70]	652
5.1.2.2799	PORTB_AddressBase [50/70]	652
5.1.2.2800	PORTB_AddressBase [51/70]	652
5.1.2.2801	PORTB_AddressBase [52/70]	652
5.1.2.2802	PORTB_AddressBase [53/70]	653
5.1.2.2803	PORTB_AddressBase [54/70]	653
5.1.2.2804	PORTB_AddressBase [55/70]	653
5.1.2.2805	PORTB_AddressBase [56/70]	653
5.1.2.2806	PORTB_AddressBase [57/70]	653
5.1.2.2807	PORTB_AddressBase [58/70]	653
5.1.2.2808	PORTB_AddressBase [59/70]	654
5.1.2.2809	PORTB_AddressBase [60/70]	654
5.1.2.2810	PORTB_AddressBase [61/70]	654
5.1.2.2811	PORTB_AddressBase [62/70]	654
5.1.2.2812	PORTB_AddressBase [63/70]	654
5.1.2.2813	PORTB_AddressBase [64/70]	654
5.1.2.2814	PORTB_AddressBase [65/70]	655
5.1.2.2815	PORTB_AddressBase [66/70]	655
5.1.2.2816	PORTB_AddressBase [67/70]	655
5.1.2.2817	PORTB_AddressBase [68/70]	655
5.1.2.2818	PORTB_AddressBase [69/70]	655
5.1.2.2819	PORTB_AddressBase [70/70]	655
5.1.2.2820	PORTC_AddressBase [1/70]	656
5.1.2.2821	PORTC_AddressBase [2/70]	656
5.1.2.2822	PORTC_AddressBase [3/70]	656
5.1.2.2823	PORTC_AddressBase [4/70]	656
5.1.2.2824	PORTC_AddressBase [5/70]	656
5.1.2.2825	PORTC_AddressBase [6/70]	656

5.1.2.2826	PORTC_AddressBase [7/70]	657
5.1.2.2827	PORTC_AddressBase [8/70]	657
5.1.2.2828	PORTC_AddressBase [9/70]	657
5.1.2.2829	PORTC_AddressBase [10/70]	657
5.1.2.2830	PORTC_AddressBase [11/70]	657
5.1.2.2831	PORTC_AddressBase [12/70]	657
5.1.2.2832	PORTC_AddressBase [13/70]	658
5.1.2.2833	PORTC_AddressBase [14/70]	658
5.1.2.2834	PORTC_AddressBase [15/70]	658
5.1.2.2835	PORTC_AddressBase [16/70]	658
5.1.2.2836	PORTC_AddressBase [17/70]	658
5.1.2.2837	PORTC_AddressBase [18/70]	658
5.1.2.2838	PORTC_AddressBase [19/70]	659
5.1.2.2839	PORTC_AddressBase [20/70]	659
5.1.2.2840	PORTC_AddressBase [21/70]	659
5.1.2.2841	PORTC_AddressBase [22/70]	659
5.1.2.2842	PORTC_AddressBase [23/70]	659
5.1.2.2843	PORTC_AddressBase [24/70]	659
5.1.2.2844	PORTC_AddressBase [25/70]	660
5.1.2.2845	PORTC_AddressBase [26/70]	660
5.1.2.2846	PORTC_AddressBase [27/70]	660
5.1.2.2847	PORTC_AddressBase [28/70]	660
5.1.2.2848	PORTC_AddressBase [29/70]	660
5.1.2.2849	PORTC_AddressBase [30/70]	660
5.1.2.2850	PORTC_AddressBase [31/70]	661
5.1.2.2851	PORTC_AddressBase [32/70]	661
5.1.2.2852	PORTC_AddressBase [33/70]	661
5.1.2.2853	PORTC_AddressBase [34/70]	661
5.1.2.2854	PORTC_AddressBase [35/70]	661
5.1.2.2855	PORTC_AddressBase [36/70]	661

5.1.2.2856	PORTC_AddressBase [37/70]	662
5.1.2.2857	PORTC_AddressBase [38/70]	662
5.1.2.2858	PORTC_AddressBase [39/70]	662
5.1.2.2859	PORTC_AddressBase [40/70]	662
5.1.2.2860	PORTC_AddressBase [41/70]	662
5.1.2.2861	PORTC_AddressBase [42/70]	662
5.1.2.2862	PORTC_AddressBase [43/70]	663
5.1.2.2863	PORTC_AddressBase [44/70]	663
5.1.2.2864	PORTC_AddressBase [45/70]	663
5.1.2.2865	PORTC_AddressBase [46/70]	663
5.1.2.2866	PORTC_AddressBase [47/70]	663
5.1.2.2867	PORTC_AddressBase [48/70]	663
5.1.2.2868	PORTC_AddressBase [49/70]	664
5.1.2.2869	PORTC_AddressBase [50/70]	664
5.1.2.2870	PORTC_AddressBase [51/70]	664
5.1.2.2871	PORTC_AddressBase [52/70]	664
5.1.2.2872	PORTC_AddressBase [53/70]	664
5.1.2.2873	PORTC_AddressBase [54/70]	664
5.1.2.2874	PORTC_AddressBase [55/70]	665
5.1.2.2875	PORTC_AddressBase [56/70]	665
5.1.2.2876	PORTC_AddressBase [57/70]	665
5.1.2.2877	PORTC_AddressBase [58/70]	665
5.1.2.2878	PORTC_AddressBase [59/70]	665
5.1.2.2879	PORTC_AddressBase [60/70]	665
5.1.2.2880	PORTC_AddressBase [61/70]	666
5.1.2.2881	PORTC_AddressBase [62/70]	666
5.1.2.2882	PORTC_AddressBase [63/70]	666
5.1.2.2883	PORTC_AddressBase [64/70]	666
5.1.2.2884	PORTC_AddressBase [65/70]	666
5.1.2.2885	PORTC_AddressBase [66/70]	666

5.1.2.2886	PORTC_AddressBase [67/70]	667
5.1.2.2887	PORTC_AddressBase [68/70]	667
5.1.2.2888	PORTC_AddressBase [69/70]	667
5.1.2.2889	PORTC_AddressBase [70/70]	667
5.1.2.2890	PORTD_AddressBase [1/70]	667
5.1.2.2891	PORTD_AddressBase [2/70]	667
5.1.2.2892	PORTD_AddressBase [3/70]	668
5.1.2.2893	PORTD_AddressBase [4/70]	668
5.1.2.2894	PORTD_AddressBase [5/70]	668
5.1.2.2895	PORTD_AddressBase [6/70]	668
5.1.2.2896	PORTD_AddressBase [7/70]	668
5.1.2.2897	PORTD_AddressBase [8/70]	668
5.1.2.2898	PORTD_AddressBase [9/70]	669
5.1.2.2899	PORTD_AddressBase [10/70]	669
5.1.2.2900	PORTD_AddressBase [11/70]	669
5.1.2.2901	PORTD_AddressBase [12/70]	669
5.1.2.2902	PORTD_AddressBase [13/70]	669
5.1.2.2903	PORTD_AddressBase [14/70]	669
5.1.2.2904	PORTD_AddressBase [15/70]	670
5.1.2.2905	PORTD_AddressBase [16/70]	670
5.1.2.2906	PORTD_AddressBase [17/70]	670
5.1.2.2907	PORTD_AddressBase [18/70]	670
5.1.2.2908	PORTD_AddressBase [19/70]	670
5.1.2.2909	PORTD_AddressBase [20/70]	670
5.1.2.2910	PORTD_AddressBase [21/70]	671
5.1.2.2911	PORTD_AddressBase [22/70]	671
5.1.2.2912	PORTD_AddressBase [23/70]	671
5.1.2.2913	PORTD_AddressBase [24/70]	671
5.1.2.2914	PORTD_AddressBase [25/70]	671
5.1.2.2915	PORTD_AddressBase [26/70]	671

5.1.2.2916	PORTD_AddressBase [27/70]	672
5.1.2.2917	PORTD_AddressBase [28/70]	672
5.1.2.2918	PORTD_AddressBase [29/70]	672
5.1.2.2919	PORTD_AddressBase [30/70]	672
5.1.2.2920	PORTD_AddressBase [31/70]	672
5.1.2.2921	PORTD_AddressBase [32/70]	672
5.1.2.2922	PORTD_AddressBase [33/70]	673
5.1.2.2923	PORTD_AddressBase [34/70]	673
5.1.2.2924	PORTD_AddressBase [35/70]	673
5.1.2.2925	PORTD_AddressBase [36/70]	673
5.1.2.2926	PORTD_AddressBase [37/70]	673
5.1.2.2927	PORTD_AddressBase [38/70]	673
5.1.2.2928	PORTD_AddressBase [39/70]	674
5.1.2.2929	PORTD_AddressBase [40/70]	674
5.1.2.2930	PORTD_AddressBase [41/70]	674
5.1.2.2931	PORTD_AddressBase [42/70]	674
5.1.2.2932	PORTD_AddressBase [43/70]	674
5.1.2.2933	PORTD_AddressBase [44/70]	674
5.1.2.2934	PORTD_AddressBase [45/70]	675
5.1.2.2935	PORTD_AddressBase [46/70]	675
5.1.2.2936	PORTD_AddressBase [47/70]	675
5.1.2.2937	PORTD_AddressBase [48/70]	675
5.1.2.2938	PORTD_AddressBase [49/70]	675
5.1.2.2939	PORTD_AddressBase [50/70]	675
5.1.2.2940	PORTD_AddressBase [51/70]	676
5.1.2.2941	PORTD_AddressBase [52/70]	676
5.1.2.2942	PORTD_AddressBase [53/70]	676
5.1.2.2943	PORTD_AddressBase [54/70]	676
5.1.2.2944	PORTD_AddressBase [55/70]	676
5.1.2.2945	PORTD_AddressBase [56/70]	676

5.1.2.2946	PORTD_AddressBase [57/70]	677
5.1.2.2947	PORTD_AddressBase [58/70]	677
5.1.2.2948	PORTD_AddressBase [59/70]	677
5.1.2.2949	PORTD_AddressBase [60/70]	677
5.1.2.2950	PORTD_AddressBase [61/70]	677
5.1.2.2951	PORTD_AddressBase [62/70]	677
5.1.2.2952	PORTD_AddressBase [63/70]	678
5.1.2.2953	PORTD_AddressBase [64/70]	678
5.1.2.2954	PORTD_AddressBase [65/70]	678
5.1.2.2955	PORTD_AddressBase [66/70]	678
5.1.2.2956	PORTD_AddressBase [67/70]	678
5.1.2.2957	PORTD_AddressBase [68/70]	678
5.1.2.2958	PORTD_AddressBase [69/70]	679
5.1.2.2959	PORTD_AddressBase [70/70]	679
5.1.2.2960	PORTE_AddressBase [1/70]	679
5.1.2.2961	PORTE_AddressBase [2/70]	679
5.1.2.2962	PORTE_AddressBase [3/70]	679
5.1.2.2963	PORTE_AddressBase [4/70]	679
5.1.2.2964	PORTE_AddressBase [5/70]	680
5.1.2.2965	PORTE_AddressBase [6/70]	680
5.1.2.2966	PORTE_AddressBase [7/70]	680
5.1.2.2967	PORTE_AddressBase [8/70]	680
5.1.2.2968	PORTE_AddressBase [9/70]	680
5.1.2.2969	PORTE_AddressBase [10/70]	680
5.1.2.2970	PORTE_AddressBase [11/70]	681
5.1.2.2971	PORTE_AddressBase [12/70]	681
5.1.2.2972	PORTE_AddressBase [13/70]	681
5.1.2.2973	PORTE_AddressBase [14/70]	681
5.1.2.2974	PORTE_AddressBase [15/70]	681
5.1.2.2975	PORTE_AddressBase [16/70]	681

5.1.2.2976	PORTE_AddressBase [17/70]	682
5.1.2.2977	PORTE_AddressBase [18/70]	682
5.1.2.2978	PORTE_AddressBase [19/70]	682
5.1.2.2979	PORTE_AddressBase [20/70]	682
5.1.2.2980	PORTE_AddressBase [21/70]	682
5.1.2.2981	PORTE_AddressBase [22/70]	682
5.1.2.2982	PORTE_AddressBase [23/70]	683
5.1.2.2983	PORTE_AddressBase [24/70]	683
5.1.2.2984	PORTE_AddressBase [25/70]	683
5.1.2.2985	PORTE_AddressBase [26/70]	683
5.1.2.2986	PORTE_AddressBase [27/70]	683
5.1.2.2987	PORTE_AddressBase [28/70]	683
5.1.2.2988	PORTE_AddressBase [29/70]	684
5.1.2.2989	PORTE_AddressBase [30/70]	684
5.1.2.2990	PORTE_AddressBase [31/70]	684
5.1.2.2991	PORTE_AddressBase [32/70]	684
5.1.2.2992	PORTE_AddressBase [33/70]	684
5.1.2.2993	PORTE_AddressBase [34/70]	684
5.1.2.2994	PORTE_AddressBase [35/70]	685
5.1.2.2995	PORTE_AddressBase [36/70]	685
5.1.2.2996	PORTE_AddressBase [37/70]	685
5.1.2.2997	PORTE_AddressBase [38/70]	685
5.1.2.2998	PORTE_AddressBase [39/70]	685
5.1.2.2999	PORTE_AddressBase [40/70]	685
5.1.2.3000	PORTE_AddressBase [41/70]	686
5.1.2.3001	PORTE_AddressBase [42/70]	686
5.1.2.3002	PORTE_AddressBase [43/70]	686
5.1.2.3003	PORTE_AddressBase [44/70]	686
5.1.2.3004	PORTE_AddressBase [45/70]	686
5.1.2.3005	PORTE_AddressBase [46/70]	686

5.1.2.3006	PORTE_AddressBase [47/70]	687
5.1.2.3007	PORTE_AddressBase [48/70]	687
5.1.2.3008	PORTE_AddressBase [49/70]	687
5.1.2.3009	PORTE_AddressBase [50/70]	687
5.1.2.3010	PORTE_AddressBase [51/70]	687
5.1.2.3011	PORTE_AddressBase [52/70]	687
5.1.2.3012	PORTE_AddressBase [53/70]	688
5.1.2.3013	PORTE_AddressBase [54/70]	688
5.1.2.3014	PORTE_AddressBase [55/70]	688
5.1.2.3015	PORTE_AddressBase [56/70]	688
5.1.2.3016	PORTE_AddressBase [57/70]	688
5.1.2.3017	PORTE_AddressBase [58/70]	688
5.1.2.3018	PORTE_AddressBase [59/70]	689
5.1.2.3019	PORTE_AddressBase [60/70]	689
5.1.2.3020	PORTE_AddressBase [61/70]	689
5.1.2.3021	PORTE_AddressBase [62/70]	689
5.1.2.3022	PORTE_AddressBase [63/70]	689
5.1.2.3023	PORTE_AddressBase [64/70]	689
5.1.2.3024	PORTE_AddressBase [65/70]	690
5.1.2.3025	PORTE_AddressBase [66/70]	690
5.1.2.3026	PORTE_AddressBase [67/70]	690
5.1.2.3027	PORTE_AddressBase [68/70]	690
5.1.2.3028	PORTE_AddressBase [69/70]	690
5.1.2.3029	PORTE_AddressBase [70/70]	690
5.1.2.3030	PORTF_AddressBase [1/70]	691
5.1.2.3031	PORTF_AddressBase [2/70]	691
5.1.2.3032	PORTF_AddressBase [3/70]	691
5.1.2.3033	PORTF_AddressBase [4/70]	691
5.1.2.3034	PORTF_AddressBase [5/70]	691
5.1.2.3035	PORTF_AddressBase [6/70]	691



5.1.2.3036	PORTF_AddressBase [7/70]	692
5.1.2.3037	PORTF_AddressBase [8/70]	692
5.1.2.3038	PORTF_AddressBase [9/70]	692
5.1.2.3039	PORTF_AddressBase [10/70]	692
5.1.2.3040	PORTF_AddressBase [11/70]	692
5.1.2.3041	PORTF_AddressBase [12/70]	692
5.1.2.3042	PORTF_AddressBase [13/70]	693
5.1.2.3043	PORTF_AddressBase [14/70]	693
5.1.2.3044	PORTF_AddressBase [15/70]	693
5.1.2.3045	PORTF_AddressBase [16/70]	693
5.1.2.3046	PORTF_AddressBase [17/70]	693
5.1.2.3047	PORTF_AddressBase [18/70]	693
5.1.2.3048	PORTF_AddressBase [19/70]	694
5.1.2.3049	PORTF_AddressBase [20/70]	694
5.1.2.3050	PORTF_AddressBase [21/70]	694
5.1.2.3051	PORTF_AddressBase [22/70]	694
5.1.2.3052	PORTF_AddressBase [23/70]	694
5.1.2.3053	PORTF_AddressBase [24/70]	694
5.1.2.3054	PORTF_AddressBase [25/70]	695
5.1.2.3055	PORTF_AddressBase [26/70]	695
5.1.2.3056	PORTF_AddressBase [27/70]	695
5.1.2.3057	PORTF_AddressBase [28/70]	695
5.1.2.3058	PORTF_AddressBase [29/70]	695
5.1.2.3059	PORTF_AddressBase [30/70]	695
5.1.2.3060	PORTF_AddressBase [31/70]	696
5.1.2.3061	PORTF_AddressBase [32/70]	696
5.1.2.3062	PORTF_AddressBase [33/70]	696
5.1.2.3063	PORTF_AddressBase [34/70]	696
5.1.2.3064	PORTF_AddressBase [35/70]	696
5.1.2.3065	PORTF_AddressBase [36/70]	696

5.1.2.3066	PORTF_AddressBase [37/70]	697
5.1.2.3067	PORTF_AddressBase [38/70]	697
5.1.2.3068	PORTF_AddressBase [39/70]	697
5.1.2.3069	PORTF_AddressBase [40/70]	697
5.1.2.3070	PORTF_AddressBase [41/70]	697
5.1.2.3071	PORTF_AddressBase [42/70]	697
5.1.2.3072	PORTF_AddressBase [43/70]	698
5.1.2.3073	PORTF_AddressBase [44/70]	698
5.1.2.3074	PORTF_AddressBase [45/70]	698
5.1.2.3075	PORTF_AddressBase [46/70]	698
5.1.2.3076	PORTF_AddressBase [47/70]	698
5.1.2.3077	PORTF_AddressBase [48/70]	698
5.1.2.3078	PORTF_AddressBase [49/70]	699
5.1.2.3079	PORTF_AddressBase [50/70]	699
5.1.2.3080	PORTF_AddressBase [51/70]	699
5.1.2.3081	PORTF_AddressBase [52/70]	699
5.1.2.3082	PORTF_AddressBase [53/70]	699
5.1.2.3083	PORTF_AddressBase [54/70]	699
5.1.2.3084	PORTF_AddressBase [55/70]	700
5.1.2.3085	PORTF_AddressBase [56/70]	700
5.1.2.3086	PORTF_AddressBase [57/70]	700
5.1.2.3087	PORTF_AddressBase [58/70]	700
5.1.2.3088	PORTF_AddressBase [59/70]	700
5.1.2.3089	PORTF_AddressBase [60/70]	700
5.1.2.3090	PORTF_AddressBase [61/70]	701
5.1.2.3091	PORTF_AddressBase [62/70]	701
5.1.2.3092	PORTF_AddressBase [63/70]	701
5.1.2.3093	PORTF_AddressBase [64/70]	701
5.1.2.3094	PORTF_AddressBase [65/70]	701
5.1.2.3095	PORTF_AddressBase [66/70]	701

5.1.2.3096	PORTF_AddressBase [67/70]	702
5.1.2.3097	PORTF_AddressBase [68/70]	702
5.1.2.3098	PORTF_AddressBase [69/70]	702
5.1.2.3099	PORTF_AddressBase [70/70]	702
5.1.2.3100	PORTG_AddressBase [1/56]	702
5.1.2.3101	PORTG_AddressBase [2/56]	702
5.1.2.3102	PORTG_AddressBase [3/56]	703
5.1.2.3103	PORTG_AddressBase [4/56]	703
5.1.2.3104	PORTG_AddressBase [5/56]	703
5.1.2.3105	PORTG_AddressBase [6/56]	703
5.1.2.3106	PORTG_AddressBase [7/56]	703
5.1.2.3107	PORTG_AddressBase [8/56]	703
5.1.2.3108	PORTG_AddressBase [9/56]	704
5.1.2.3109	PORTG_AddressBase [10/56]	704
5.1.2.3110	PORTG_AddressBase [11/56]	704
5.1.2.3111	PORTG_AddressBase [12/56]	704
5.1.2.3112	PORTG_AddressBase [13/56]	704
5.1.2.3113	PORTG_AddressBase [14/56]	704
5.1.2.3114	PORTG_AddressBase [15/56]	705
5.1.2.3115	PORTG_AddressBase [16/56]	705
5.1.2.3116	PORTG_AddressBase [17/56]	705
5.1.2.3117	PORTG_AddressBase [18/56]	705
5.1.2.3118	PORTG_AddressBase [19/56]	705
5.1.2.3119	PORTG_AddressBase [20/56]	705
5.1.2.3120	PORTG_AddressBase [21/56]	706
5.1.2.3121	PORTG_AddressBase [22/56]	706
5.1.2.3122	PORTG_AddressBase [23/56]	706
5.1.2.3123	PORTG_AddressBase [24/56]	706
5.1.2.3124	PORTG_AddressBase [25/56]	706
5.1.2.3125	PORTG_AddressBase [26/56]	706

5.1.2.3126	PORTG_AddressBase [27/56]	707
5.1.2.3127	PORTG_AddressBase [28/56]	707
5.1.2.3128	PORTG_AddressBase [29/56]	707
5.1.2.3129	PORTG_AddressBase [30/56]	707
5.1.2.3130	PORTG_AddressBase [31/56]	707
5.1.2.3131	PORTG_AddressBase [32/56]	707
5.1.2.3132	PORTG_AddressBase [33/56]	708
5.1.2.3133	PORTG_AddressBase [34/56]	708
5.1.2.3134	PORTG_AddressBase [35/56]	708
5.1.2.3135	PORTG_AddressBase [36/56]	708
5.1.2.3136	PORTG_AddressBase [37/56]	708
5.1.2.3137	PORTG_AddressBase [38/56]	708
5.1.2.3138	PORTG_AddressBase [39/56]	709
5.1.2.3139	PORTG_AddressBase [40/56]	709
5.1.2.3140	PORTG_AddressBase [41/56]	709
5.1.2.3141	PORTG_AddressBase [42/56]	709
5.1.2.3142	PORTG_AddressBase [43/56]	709
5.1.2.3143	PORTG_AddressBase [44/56]	709
5.1.2.3144	PORTG_AddressBase [45/56]	710
5.1.2.3145	PORTG_AddressBase [46/56]	710
5.1.2.3146	PORTG_AddressBase [47/56]	710
5.1.2.3147	PORTG_AddressBase [48/56]	710
5.1.2.3148	PORTG_AddressBase [49/56]	710
5.1.2.3149	PORTG_AddressBase [50/56]	710
5.1.2.3150	PORTG_AddressBase [51/56]	711
5.1.2.3151	PORTG_AddressBase [52/56]	711
5.1.2.3152	PORTG_AddressBase [53/56]	711
5.1.2.3153	PORTG_AddressBase [54/56]	711
5.1.2.3154	PORTG_AddressBase [55/56]	711
5.1.2.3155	PORTG_AddressBase [56/56]	711

5.1.2.3156	PORTH_AddressBase [1/43]	712
5.1.2.3157	PORTH_AddressBase [2/43]	712
5.1.2.3158	PORTH_AddressBase [3/43]	712
5.1.2.3159	PORTH_AddressBase [4/43]	712
5.1.2.3160	PORTH_AddressBase [5/43]	712
5.1.2.3161	PORTH_AddressBase [6/43]	712
5.1.2.3162	PORTH_AddressBase [7/43]	713
5.1.2.3163	PORTH_AddressBase [8/43]	713
5.1.2.3164	PORTH_AddressBase [9/43]	713
5.1.2.3165	PORTH_AddressBase [10/43]	713
5.1.2.3166	PORTH_AddressBase [11/43]	713
5.1.2.3167	PORTH_AddressBase [12/43]	713
5.1.2.3168	PORTH_AddressBase [13/43]	714
5.1.2.3169	PORTH_AddressBase [14/43]	714
5.1.2.3170	PORTH_AddressBase [15/43]	714
5.1.2.3171	PORTH_AddressBase [16/43]	714
5.1.2.3172	PORTH_AddressBase [17/43]	714
5.1.2.3173	PORTH_AddressBase [18/43]	714
5.1.2.3174	PORTH_AddressBase [19/43]	715
5.1.2.3175	PORTH_AddressBase [20/43]	715
5.1.2.3176	PORTH_AddressBase [21/43]	715
5.1.2.3177	PORTH_AddressBase [22/43]	715
5.1.2.3178	PORTH_AddressBase [23/43]	715
5.1.2.3179	PORTH_AddressBase [24/43]	715
5.1.2.3180	PORTH_AddressBase [25/43]	716
5.1.2.3181	PORTH_AddressBase [26/43]	716
5.1.2.3182	PORTH_AddressBase [27/43]	716
5.1.2.3183	PORTH_AddressBase [28/43]	716
5.1.2.3184	PORTH_AddressBase [29/43]	716
5.1.2.3185	PORTH_AddressBase [30/43]	716

5.1.2.3186	PORTH_AddressBase [31/43]	717
5.1.2.3187	PORTH_AddressBase [32/43]	717
5.1.2.3188	PORTH_AddressBase [33/43]	717
5.1.2.3189	PORTH_AddressBase [34/43]	717
5.1.2.3190	PORTH_AddressBase [35/43]	717
5.1.2.3191	PORTH_AddressBase [36/43]	717
5.1.2.3192	PORTH_AddressBase [37/43]	718
5.1.2.3193	PORTH_AddressBase [38/43]	718
5.1.2.3194	PORTH_AddressBase [39/43]	718
5.1.2.3195	PORTH_AddressBase [40/43]	718
5.1.2.3196	PORTH_AddressBase [41/43]	718
5.1.2.3197	PORTH_AddressBase [42/43]	718
5.1.2.3198	PORTH_AddressBase [43/43]	719
5.1.2.3199	PORTI_AddressBase [1/43]	719
5.1.2.3200	PORTI_AddressBase [2/43]	719
5.1.2.3201	PORTI_AddressBase [3/43]	719
5.1.2.3202	PORTI_AddressBase [4/43]	719
5.1.2.3203	PORTI_AddressBase [5/43]	719
5.1.2.3204	PORTI_AddressBase [6/43]	720
5.1.2.3205	PORTI_AddressBase [7/43]	720
5.1.2.3206	PORTI_AddressBase [8/43]	720
5.1.2.3207	PORTI_AddressBase [9/43]	720
5.1.2.3208	PORTI_AddressBase [10/43]	720
5.1.2.3209	PORTI_AddressBase [11/43]	720
5.1.2.3210	PORTI_AddressBase [12/43]	721
5.1.2.3211	PORTI_AddressBase [13/43]	721
5.1.2.3212	PORTI_AddressBase [14/43]	721
5.1.2.3213	PORTI_AddressBase [15/43]	721
5.1.2.3214	PORTI_AddressBase [16/43]	721
5.1.2.3215	PORTI_AddressBase [17/43]	721

5.1.2.3216PORTI_AddressBase [18/43]	722
5.1.2.3217PORTI_AddressBase [19/43]	722
5.1.2.3218PORTI_AddressBase [20/43]	722
5.1.2.3219PORTI_AddressBase [21/43]	722
5.1.2.3220PORTI_AddressBase [22/43]	722
5.1.2.3221PORTI_AddressBase [23/43]	722
5.1.2.3222PORTI_AddressBase [24/43]	723
5.1.2.3223PORTI_AddressBase [25/43]	723
5.1.2.3224PORTI_AddressBase [26/43]	723
5.1.2.3225PORTI_AddressBase [27/43]	723
5.1.2.3226PORTI_AddressBase [28/43]	723
5.1.2.3227PORTI_AddressBase [29/43]	723
5.1.2.3228PORTI_AddressBase [30/43]	724
5.1.2.3229PORTI_AddressBase [31/43]	724
5.1.2.3230PORTI_AddressBase [32/43]	724
5.1.2.3231PORTI_AddressBase [33/43]	724
5.1.2.3232PORTI_AddressBase [34/43]	724
5.1.2.3233PORTI_AddressBase [35/43]	724
5.1.2.3234PORTI_AddressBase [36/43]	725
5.1.2.3235PORTI_AddressBase [37/43]	725
5.1.2.3236PORTI_AddressBase [38/43]	725
5.1.2.3237PORTI_AddressBase [39/43]	725
5.1.2.3238PORTI_AddressBase [40/43]	725
5.1.2.3239PORTI_AddressBase [41/43]	725
5.1.2.3240PORTI_AddressBase [42/43]	726
5.1.2.3241PORTI_AddressBase [43/43]	726
5.1.2.3242RST_AddressBase [1/70]	726
5.1.2.3243RST_AddressBase [2/70]	726
5.1.2.3244RST_AddressBase [3/70]	726
5.1.2.3245RST_AddressBase [4/70]	726

5.1.2.3246RST_AddressBase [5/70]	727
5.1.2.3247RST_AddressBase [6/70]	727
5.1.2.3248RST_AddressBase [7/70]	727
5.1.2.3249RST_AddressBase [8/70]	727
5.1.2.3250RST_AddressBase [9/70]	727
5.1.2.3251RST_AddressBase [10/70]	727
5.1.2.3252RST_AddressBase [11/70]	728
5.1.2.3253RST_AddressBase [12/70]	728
5.1.2.3254RST_AddressBase [13/70]	728
5.1.2.3255RST_AddressBase [14/70]	728
5.1.2.3256RST_AddressBase [15/70]	728
5.1.2.3257RST_AddressBase [16/70]	728
5.1.2.3258RST_AddressBase [17/70]	729
5.1.2.3259RST_AddressBase [18/70]	729
5.1.2.3260RST_AddressBase [19/70]	729
5.1.2.3261RST_AddressBase [20/70]	729
5.1.2.3262RST_AddressBase [21/70]	729
5.1.2.3263RST_AddressBase [22/70]	729
5.1.2.3264RST_AddressBase [23/70]	730
5.1.2.3265RST_AddressBase [24/70]	730
5.1.2.3266RST_AddressBase [25/70]	730
5.1.2.3267RST_AddressBase [26/70]	730
5.1.2.3268RST_AddressBase [27/70]	730
5.1.2.3269RST_AddressBase [28/70]	730
5.1.2.3270RST_AddressBase [29/70]	731
5.1.2.3271RST_AddressBase [30/70]	731
5.1.2.3272RST_AddressBase [31/70]	731
5.1.2.3273RST_AddressBase [32/70]	731
5.1.2.3274RST_AddressBase [33/70]	731
5.1.2.3275RST_AddressBase [34/70]	731



5.1.2.3276RST_AddressBase [35/70]	732
5.1.2.3277RST_AddressBase [36/70]	732
5.1.2.3278RST_AddressBase [37/70]	732
5.1.2.3279RST_AddressBase [38/70]	732
5.1.2.3280RST_AddressBase [39/70]	732
5.1.2.3281RST_AddressBase [40/70]	732
5.1.2.3282RST_AddressBase [41/70]	733
5.1.2.3283RST_AddressBase [42/70]	733
5.1.2.3284RST_AddressBase [43/70]	733
5.1.2.3285RST_AddressBase [44/70]	733
5.1.2.3286RST_AddressBase [45/70]	733
5.1.2.3287RST_AddressBase [46/70]	733
5.1.2.3288RST_AddressBase [47/70]	734
5.1.2.3289RST_AddressBase [48/70]	734
5.1.2.3290RST_AddressBase [49/70]	734
5.1.2.3291RST_AddressBase [50/70]	734
5.1.2.3292RST_AddressBase [51/70]	734
5.1.2.3293RST_AddressBase [52/70]	734
5.1.2.3294RST_AddressBase [53/70]	735
5.1.2.3295RST_AddressBase [54/70]	735
5.1.2.3296RST_AddressBase [55/70]	735
5.1.2.3297RST_AddressBase [56/70]	735
5.1.2.3298RST_AddressBase [57/70]	735
5.1.2.3299RST_AddressBase [58/70]	735
5.1.2.3300RST_AddressBase [59/70]	736
5.1.2.3301RST_AddressBase [60/70]	736
5.1.2.3302RST_AddressBase [61/70]	736
5.1.2.3303RST_AddressBase [62/70]	736
5.1.2.3304RST_AddressBase [63/70]	736
5.1.2.3305RST_AddressBase [64/70]	736

5.1.2.3306RST_AddressBase [65/70] . . . . .	737
5.1.2.3307RST_AddressBase [66/70] . . . . .	737
5.1.2.3308RST_AddressBase [67/70] . . . . .	737
5.1.2.3309RST_AddressBase [68/70] . . . . .	737
5.1.2.3310RST_AddressBase [69/70] . . . . .	737
5.1.2.3311RST_AddressBase [70/70] . . . . .	737
5.1.2.3312SPI_AddressBase [1/70] . . . . .	738
5.1.2.3313SPI_AddressBase [2/70] . . . . .	738
5.1.2.3314SPI_AddressBase [3/70] . . . . .	738
5.1.2.3315SPI_AddressBase [4/70] . . . . .	738
5.1.2.3316SPI_AddressBase [5/70] . . . . .	738
5.1.2.3317SPI_AddressBase [6/70] . . . . .	738
5.1.2.3318SPI_AddressBase [7/70] . . . . .	739
5.1.2.3319SPI_AddressBase [8/70] . . . . .	739
5.1.2.3320SPI_AddressBase [9/70] . . . . .	739
5.1.2.3321SPI_AddressBase [10/70] . . . . .	739
5.1.2.3322SPI_AddressBase [11/70] . . . . .	739
5.1.2.3323SPI_AddressBase [12/70] . . . . .	739
5.1.2.3324SPI_AddressBase [13/70] . . . . .	740
5.1.2.3325SPI_AddressBase [14/70] . . . . .	740
5.1.2.3326SPI_AddressBase [15/70] . . . . .	740
5.1.2.3327SPI_AddressBase [16/70] . . . . .	740
5.1.2.3328SPI_AddressBase [17/70] . . . . .	740
5.1.2.3329SPI_AddressBase [18/70] . . . . .	740
5.1.2.3330SPI_AddressBase [19/70] . . . . .	741
5.1.2.3331SPI_AddressBase [20/70] . . . . .	741
5.1.2.3332SPI_AddressBase [21/70] . . . . .	741
5.1.2.3333SPI_AddressBase [22/70] . . . . .	741
5.1.2.3334SPI_AddressBase [23/70] . . . . .	741
5.1.2.3335SPI_AddressBase [24/70] . . . . .	741

5.1.2.3336SPI_AddressBase [25/70]	742
5.1.2.3337SPI_AddressBase [26/70]	742
5.1.2.3338SPI_AddressBase [27/70]	742
5.1.2.3339SPI_AddressBase [28/70]	742
5.1.2.3340SPI_AddressBase [29/70]	742
5.1.2.3341SPI_AddressBase [30/70]	742
5.1.2.3342SPI_AddressBase [31/70]	743
5.1.2.3343SPI_AddressBase [32/70]	743
5.1.2.3344SPI_AddressBase [33/70]	743
5.1.2.3345SPI_AddressBase [34/70]	743
5.1.2.3346SPI_AddressBase [35/70]	743
5.1.2.3347SPI_AddressBase [36/70]	743
5.1.2.3348SPI_AddressBase [37/70]	744
5.1.2.3349SPI_AddressBase [38/70]	744
5.1.2.3350SPI_AddressBase [39/70]	744
5.1.2.3351SPI_AddressBase [40/70]	744
5.1.2.3352SPI_AddressBase [41/70]	744
5.1.2.3353SPI_AddressBase [42/70]	744
5.1.2.3354SPI_AddressBase [43/70]	745
5.1.2.3355SPI_AddressBase [44/70]	745
5.1.2.3356SPI_AddressBase [45/70]	745
5.1.2.3357SPI_AddressBase [46/70]	745
5.1.2.3358SPI_AddressBase [47/70]	745
5.1.2.3359SPI_AddressBase [48/70]	745
5.1.2.3360SPI_AddressBase [49/70]	746
5.1.2.3361SPI_AddressBase [50/70]	746
5.1.2.3362SPI_AddressBase [51/70]	746
5.1.2.3363SPI_AddressBase [52/70]	746
5.1.2.3364SPI_AddressBase [53/70]	746
5.1.2.3365SPI_AddressBase [54/70]	746

5.1.2.3366SPI_AddressBase [55/70]	747
5.1.2.3367SPI_AddressBase [56/70]	747
5.1.2.3368SPI_AddressBase [57/70]	747
5.1.2.3369SPI_AddressBase [58/70]	747
5.1.2.3370SPI_AddressBase [59/70]	747
5.1.2.3371SPI_AddressBase [60/70]	747
5.1.2.3372SPI_AddressBase [61/70]	748
5.1.2.3373SPI_AddressBase [62/70]	748
5.1.2.3374SPI_AddressBase [63/70]	748
5.1.2.3375SPI_AddressBase [64/70]	748
5.1.2.3376SPI_AddressBase [65/70]	748
5.1.2.3377SPI_AddressBase [66/70]	748
5.1.2.3378SPI_AddressBase [67/70]	749
5.1.2.3379SPI_AddressBase [68/70]	749
5.1.2.3380SPI_AddressBase [69/70]	749
5.1.2.3381SPI_AddressBase [70/70]	749
5.1.2.3382STM8_ADDR_WIDTH	749
5.1.2.3383STM8_EEPROM_END	749
5.1.2.3384STM8_EEPROM_SIZE [1/71]	750
5.1.2.3385STM8_EEPROM_SIZE [2/71]	750
5.1.2.3386STM8_EEPROM_SIZE [3/71]	750
5.1.2.3387STM8_EEPROM_SIZE [4/71]	750
5.1.2.3388STM8_EEPROM_SIZE [5/71]	750
5.1.2.3389STM8_EEPROM_SIZE [6/71]	750
5.1.2.3390STM8_EEPROM_SIZE [7/71]	751
5.1.2.3391STM8_EEPROM_SIZE [8/71]	751
5.1.2.3392STM8_EEPROM_SIZE [9/71]	751
5.1.2.3393STM8_EEPROM_SIZE [10/71]	751
5.1.2.3394STM8_EEPROM_SIZE [11/71]	751
5.1.2.3395STM8_EEPROM_SIZE [12/71]	751

5.1.2.3396STM8_EEPROM_SIZE [13/71]	752
5.1.2.3397STM8_EEPROM_SIZE [14/71]	752
5.1.2.3398STM8_EEPROM_SIZE [15/71]	752
5.1.2.3399STM8_EEPROM_SIZE [16/71]	752
5.1.2.3400STM8_EEPROM_SIZE [17/71]	752
5.1.2.3401STM8_EEPROM_SIZE [18/71]	752
5.1.2.3402STM8_EEPROM_SIZE [19/71]	753
5.1.2.3403STM8_EEPROM_SIZE [20/71]	753
5.1.2.3404STM8_EEPROM_SIZE [21/71]	753
5.1.2.3405STM8_EEPROM_SIZE [22/71]	753
5.1.2.3406STM8_EEPROM_SIZE [23/71]	753
5.1.2.3407STM8_EEPROM_SIZE [24/71]	753
5.1.2.3408STM8_EEPROM_SIZE [25/71]	754
5.1.2.3409STM8_EEPROM_SIZE [26/71]	754
5.1.2.3410STM8_EEPROM_SIZE [27/71]	754
5.1.2.3411STM8_EEPROM_SIZE [28/71]	754
5.1.2.3412STM8_EEPROM_SIZE [29/71]	754
5.1.2.3413STM8_EEPROM_SIZE [30/71]	754
5.1.2.3414STM8_EEPROM_SIZE [31/71]	755
5.1.2.3415STM8_EEPROM_SIZE [32/71]	755
5.1.2.3416STM8_EEPROM_SIZE [33/71]	755
5.1.2.3417STM8_EEPROM_SIZE [34/71]	755
5.1.2.3418STM8_EEPROM_SIZE [35/71]	755
5.1.2.3419STM8_EEPROM_SIZE [36/71]	755
5.1.2.3420STM8_EEPROM_SIZE [37/71]	756
5.1.2.3421STM8_EEPROM_SIZE [38/71]	756
5.1.2.3422STM8_EEPROM_SIZE [39/71]	756
5.1.2.3423STM8_EEPROM_SIZE [40/71]	756
5.1.2.3424STM8_EEPROM_SIZE [41/71]	756
5.1.2.3425STM8_EEPROM_SIZE [42/71]	756

5.1.2.3426STM8_EEPROM_SIZE [43/71]	757
5.1.2.3427STM8_EEPROM_SIZE [44/71]	757
5.1.2.3428STM8_EEPROM_SIZE [45/71]	757
5.1.2.3429STM8_EEPROM_SIZE [46/71]	757
5.1.2.3430STM8_EEPROM_SIZE [47/71]	757
5.1.2.3431STM8_EEPROM_SIZE [48/71]	757
5.1.2.3432STM8_EEPROM_SIZE [49/71]	758
5.1.2.3433STM8_EEPROM_SIZE [50/71]	758
5.1.2.3434STM8_EEPROM_SIZE [51/71]	758
5.1.2.3435STM8_EEPROM_SIZE [52/71]	758
5.1.2.3436STM8_EEPROM_SIZE [53/71]	758
5.1.2.3437STM8_EEPROM_SIZE [54/71]	758
5.1.2.3438STM8_EEPROM_SIZE [55/71]	759
5.1.2.3439STM8_EEPROM_SIZE [56/71]	759
5.1.2.3440STM8_EEPROM_SIZE [57/71]	759
5.1.2.3441STM8_EEPROM_SIZE [58/71]	759
5.1.2.3442STM8_EEPROM_SIZE [59/71]	759
5.1.2.3443STM8_EEPROM_SIZE [60/71]	759
5.1.2.3444STM8_EEPROM_SIZE [61/71]	760
5.1.2.3445STM8_EEPROM_SIZE [62/71]	760
5.1.2.3446STM8_EEPROM_SIZE [63/71]	760
5.1.2.3447STM8_EEPROM_SIZE [64/71]	760
5.1.2.3448STM8_EEPROM_SIZE [65/71]	760
5.1.2.3449STM8_EEPROM_SIZE [66/71]	760
5.1.2.3450STM8_EEPROM_SIZE [67/71]	761
5.1.2.3451STM8_EEPROM_SIZE [68/71]	761
5.1.2.3452STM8_EEPROM_SIZE [69/71]	761
5.1.2.3453STM8_EEPROM_SIZE [70/71]	761
5.1.2.3454STM8_EEPROM_SIZE [71/71]	761
5.1.2.3455STM8_EEPROM_START	761

5.1.2.3456STM8_MEM_POINTER_T . . . . .	762
5.1.2.3457STM8_PFLASH_END . . . . .	762
5.1.2.3458STM8_PFLASH_SIZE [1/71] . . . . .	762
5.1.2.3459STM8_PFLASH_SIZE [2/71] . . . . .	762
5.1.2.3460STM8_PFLASH_SIZE [3/71] . . . . .	762
5.1.2.3461STM8_PFLASH_SIZE [4/71] . . . . .	762
5.1.2.3462STM8_PFLASH_SIZE [5/71] . . . . .	763
5.1.2.3463STM8_PFLASH_SIZE [6/71] . . . . .	763
5.1.2.3464STM8_PFLASH_SIZE [7/71] . . . . .	763
5.1.2.3465STM8_PFLASH_SIZE [8/71] . . . . .	763
5.1.2.3466STM8_PFLASH_SIZE [9/71] . . . . .	763
5.1.2.3467STM8_PFLASH_SIZE [10/71] . . . . .	763
5.1.2.3468STM8_PFLASH_SIZE [11/71] . . . . .	764
5.1.2.3469STM8_PFLASH_SIZE [12/71] . . . . .	764
5.1.2.3470STM8_PFLASH_SIZE [13/71] . . . . .	764
5.1.2.3471STM8_PFLASH_SIZE [14/71] . . . . .	764
5.1.2.3472STM8_PFLASH_SIZE [15/71] . . . . .	764
5.1.2.3473STM8_PFLASH_SIZE [16/71] . . . . .	764
5.1.2.3474STM8_PFLASH_SIZE [17/71] . . . . .	765
5.1.2.3475STM8_PFLASH_SIZE [18/71] . . . . .	765
5.1.2.3476STM8_PFLASH_SIZE [19/71] . . . . .	765
5.1.2.3477STM8_PFLASH_SIZE [20/71] . . . . .	765
5.1.2.3478STM8_PFLASH_SIZE [21/71] . . . . .	765
5.1.2.3479STM8_PFLASH_SIZE [22/71] . . . . .	765
5.1.2.3480STM8_PFLASH_SIZE [23/71] . . . . .	766
5.1.2.3481STM8_PFLASH_SIZE [24/71] . . . . .	766
5.1.2.3482STM8_PFLASH_SIZE [25/71] . . . . .	766
5.1.2.3483STM8_PFLASH_SIZE [26/71] . . . . .	766
5.1.2.3484STM8_PFLASH_SIZE [27/71] . . . . .	766
5.1.2.3485STM8_PFLASH_SIZE [28/71] . . . . .	766

5.1.2.3486STM8_PFLASH_SIZE [29/71]	767
5.1.2.3487STM8_PFLASH_SIZE [30/71]	767
5.1.2.3488STM8_PFLASH_SIZE [31/71]	767
5.1.2.3489STM8_PFLASH_SIZE [32/71]	767
5.1.2.3490STM8_PFLASH_SIZE [33/71]	767
5.1.2.3491STM8_PFLASH_SIZE [34/71]	767
5.1.2.3492STM8_PFLASH_SIZE [35/71]	768
5.1.2.3493STM8_PFLASH_SIZE [36/71]	768
5.1.2.3494STM8_PFLASH_SIZE [37/71]	768
5.1.2.3495STM8_PFLASH_SIZE [38/71]	768
5.1.2.3496STM8_PFLASH_SIZE [39/71]	768
5.1.2.3497STM8_PFLASH_SIZE [40/71]	768
5.1.2.3498STM8_PFLASH_SIZE [41/71]	769
5.1.2.3499STM8_PFLASH_SIZE [42/71]	769
5.1.2.3500STM8_PFLASH_SIZE [43/71]	769
5.1.2.3501STM8_PFLASH_SIZE [44/71]	769
5.1.2.3502STM8_PFLASH_SIZE [45/71]	769
5.1.2.3503STM8_PFLASH_SIZE [46/71]	769
5.1.2.3504STM8_PFLASH_SIZE [47/71]	770
5.1.2.3505STM8_PFLASH_SIZE [48/71]	770
5.1.2.3506STM8_PFLASH_SIZE [49/71]	770
5.1.2.3507STM8_PFLASH_SIZE [50/71]	770
5.1.2.3508STM8_PFLASH_SIZE [51/71]	770
5.1.2.3509STM8_PFLASH_SIZE [52/71]	770
5.1.2.3510STM8_PFLASH_SIZE [53/71]	771
5.1.2.3511STM8_PFLASH_SIZE [54/71]	771
5.1.2.3512STM8_PFLASH_SIZE [55/71]	771
5.1.2.3513STM8_PFLASH_SIZE [56/71]	771
5.1.2.3514STM8_PFLASH_SIZE [57/71]	771
5.1.2.3515STM8_PFLASH_SIZE [58/71]	771



5.1.2.3516STM8_PFLASH_SIZE [59/71]	772
5.1.2.3517STM8_PFLASH_SIZE [60/71]	772
5.1.2.3518STM8_PFLASH_SIZE [61/71]	772
5.1.2.3519STM8_PFLASH_SIZE [62/71]	772
5.1.2.3520STM8_PFLASH_SIZE [63/71]	772
5.1.2.3521STM8_PFLASH_SIZE [64/71]	772
5.1.2.3522STM8_PFLASH_SIZE [65/71]	773
5.1.2.3523STM8_PFLASH_SIZE [66/71]	773
5.1.2.3524STM8_PFLASH_SIZE [67/71]	773
5.1.2.3525STM8_PFLASH_SIZE [68/71]	773
5.1.2.3526STM8_PFLASH_SIZE [69/71]	773
5.1.2.3527STM8_PFLASH_SIZE [70/71]	773
5.1.2.3528STM8_PFLASH_SIZE [71/71]	774
5.1.2.3529STM8_PFLASH_START	774
5.1.2.3530STM8_RAM_END	774
5.1.2.3531STM8_RAM_SIZE [1/71]	774
5.1.2.3532STM8_RAM_SIZE [2/71]	774
5.1.2.3533STM8_RAM_SIZE [3/71]	775
5.1.2.3534STM8_RAM_SIZE [4/71]	775
5.1.2.3535STM8_RAM_SIZE [5/71]	775
5.1.2.3536STM8_RAM_SIZE [6/71]	775
5.1.2.3537STM8_RAM_SIZE [7/71]	775
5.1.2.3538STM8_RAM_SIZE [8/71]	775
5.1.2.3539STM8_RAM_SIZE [9/71]	776
5.1.2.3540STM8_RAM_SIZE [10/71]	776
5.1.2.3541STM8_RAM_SIZE [11/71]	776
5.1.2.3542STM8_RAM_SIZE [12/71]	776
5.1.2.3543STM8_RAM_SIZE [13/71]	776
5.1.2.3544STM8_RAM_SIZE [14/71]	776
5.1.2.3545STM8_RAM_SIZE [15/71]	777

5.1.2.3546STM8_RAM_SIZE [16/71]	777
5.1.2.3547STM8_RAM_SIZE [17/71]	777
5.1.2.3548STM8_RAM_SIZE [18/71]	777
5.1.2.3549STM8_RAM_SIZE [19/71]	777
5.1.2.3550STM8_RAM_SIZE [20/71]	777
5.1.2.3551STM8_RAM_SIZE [21/71]	778
5.1.2.3552STM8_RAM_SIZE [22/71]	778
5.1.2.3553STM8_RAM_SIZE [23/71]	778
5.1.2.3554STM8_RAM_SIZE [24/71]	778
5.1.2.3555STM8_RAM_SIZE [25/71]	778
5.1.2.3556STM8_RAM_SIZE [26/71]	778
5.1.2.3557STM8_RAM_SIZE [27/71]	779
5.1.2.3558STM8_RAM_SIZE [28/71]	779
5.1.2.3559STM8_RAM_SIZE [29/71]	779
5.1.2.3560STM8_RAM_SIZE [30/71]	779
5.1.2.3561STM8_RAM_SIZE [31/71]	779
5.1.2.3562STM8_RAM_SIZE [32/71]	779
5.1.2.3563STM8_RAM_SIZE [33/71]	780
5.1.2.3564STM8_RAM_SIZE [34/71]	780
5.1.2.3565STM8_RAM_SIZE [35/71]	780
5.1.2.3566STM8_RAM_SIZE [36/71]	780
5.1.2.3567STM8_RAM_SIZE [37/71]	780
5.1.2.3568STM8_RAM_SIZE [38/71]	780
5.1.2.3569STM8_RAM_SIZE [39/71]	781
5.1.2.3570STM8_RAM_SIZE [40/71]	781
5.1.2.3571STM8_RAM_SIZE [41/71]	781
5.1.2.3572STM8_RAM_SIZE [42/71]	781
5.1.2.3573STM8_RAM_SIZE [43/71]	781
5.1.2.3574STM8_RAM_SIZE [44/71]	781
5.1.2.3575STM8_RAM_SIZE [45/71]	782

5.1.2.3576STM8_RAM_SIZE [46/71]	782
5.1.2.3577STM8_RAM_SIZE [47/71]	782
5.1.2.3578STM8_RAM_SIZE [48/71]	782
5.1.2.3579STM8_RAM_SIZE [49/71]	782
5.1.2.3580STM8_RAM_SIZE [50/71]	782
5.1.2.3581STM8_RAM_SIZE [51/71]	783
5.1.2.3582STM8_RAM_SIZE [52/71]	783
5.1.2.3583STM8_RAM_SIZE [53/71]	783
5.1.2.3584STM8_RAM_SIZE [54/71]	783
5.1.2.3585STM8_RAM_SIZE [55/71]	783
5.1.2.3586STM8_RAM_SIZE [56/71]	783
5.1.2.3587STM8_RAM_SIZE [57/71]	784
5.1.2.3588STM8_RAM_SIZE [58/71]	784
5.1.2.3589STM8_RAM_SIZE [59/71]	784
5.1.2.3590STM8_RAM_SIZE [60/71]	784
5.1.2.3591STM8_RAM_SIZE [61/71]	784
5.1.2.3592STM8_RAM_SIZE [62/71]	784
5.1.2.3593STM8_RAM_SIZE [63/71]	785
5.1.2.3594STM8_RAM_SIZE [64/71]	785
5.1.2.3595STM8_RAM_SIZE [65/71]	785
5.1.2.3596STM8_RAM_SIZE [66/71]	785
5.1.2.3597STM8_RAM_SIZE [67/71]	785
5.1.2.3598STM8_RAM_SIZE [68/71]	785
5.1.2.3599STM8_RAM_SIZE [69/71]	786
5.1.2.3600STM8_RAM_SIZE [70/71]	786
5.1.2.3601STM8_RAM_SIZE [71/71]	786
5.1.2.3602STM8_RAM_START	786
5.1.2.3603STM8AF5268	786
5.1.2.3604STM8AF5269	786
5.1.2.3605STM8AF526x [1/2]	787

5.1.2.3606STM8AF526x [2/2] . . . . .	787
5.1.2.3607STM8AF5286 . . . . .	787
5.1.2.3608STM8AF5288 . . . . .	787
5.1.2.3609STM8AF5289 . . . . .	787
5.1.2.3610STM8AF528A . . . . .	787
5.1.2.3611STM8AF528x [1/4] . . . . .	788
5.1.2.3612STM8AF528x [2/4] . . . . .	788
5.1.2.3613STM8AF528x [3/4] . . . . .	788
5.1.2.3614STM8AF528x [4/4] . . . . .	788
5.1.2.3615STM8AF52A6 . . . . .	788
5.1.2.3616STM8AF52A8 . . . . .	788
5.1.2.3617STM8AF52A9 . . . . .	789
5.1.2.3618STM8AF52AA . . . . .	789
5.1.2.3619STM8AF52Ax [1/4] . . . . .	789
5.1.2.3620STM8AF52Ax [2/4] . . . . .	789
5.1.2.3621STM8AF52Ax [3/4] . . . . .	789
5.1.2.3622STM8AF52Ax [4/4] . . . . .	789
5.1.2.3623STM8AF6213 . . . . .	790
5.1.2.3624STM8AF6213A . . . . .	790
5.1.2.3625STM8AF621x [1/2] . . . . .	790
5.1.2.3626STM8AF621x [2/2] . . . . .	790
5.1.2.3627STM8AF6223 . . . . .	790
5.1.2.3628STM8AF6223A . . . . .	790
5.1.2.3629STM8AF6226 . . . . .	791
5.1.2.3630STM8AF622x [1/3] . . . . .	791
5.1.2.3631STM8AF622x [2/3] . . . . .	791
5.1.2.3632STM8AF622x [3/3] . . . . .	791
5.1.2.3633STM8AF6246 . . . . .	791
5.1.2.3634STM8AF6248 . . . . .	791
5.1.2.3635STM8AF624x [1/2] . . . . .	792

5.1.2.3636STM8AF624x [2/2] . . . . .	792
5.1.2.3637STM8AF6266 . . . . .	792
5.1.2.3638STM8AF6268 . . . . .	792
5.1.2.3639STM8AF6269 . . . . .	792
5.1.2.3640STM8AF626x [1/3] . . . . .	792
5.1.2.3641STM8AF626x [2/3] . . . . .	793
5.1.2.3642STM8AF626x [3/3] . . . . .	793
5.1.2.3643STM8AF6286 . . . . .	793
5.1.2.3644STM8AF6288 . . . . .	793
5.1.2.3645STM8AF6289 . . . . .	793
5.1.2.3646STM8AF628A . . . . .	793
5.1.2.3647STM8AF628x [1/4] . . . . .	794
5.1.2.3648STM8AF628x [2/4] . . . . .	794
5.1.2.3649STM8AF628x [3/4] . . . . .	794
5.1.2.3650STM8AF628x [4/4] . . . . .	794
5.1.2.3651STM8AF62A6 . . . . .	794
5.1.2.3652STM8AF62A8 . . . . .	794
5.1.2.3653STM8AF62A9 . . . . .	795
5.1.2.3654STM8AF62AA . . . . .	795
5.1.2.3655STM8AF62Ax [1/4] . . . . .	795
5.1.2.3656STM8AF62Ax [2/4] . . . . .	795
5.1.2.3657STM8AF62Ax [3/4] . . . . .	795
5.1.2.3658STM8AF62Ax [4/4] . . . . .	795
5.1.2.3659STM8AF6366 . . . . .	796
5.1.2.3660STM8AF636x . . . . .	796
5.1.2.3661STM8AF6388 . . . . .	796
5.1.2.3662STM8AF638x . . . . .	796
5.1.2.3663STM8S001 . . . . .	796
5.1.2.3664STM8S001J3 . . . . .	796
5.1.2.3665STM8S003 [1/2] . . . . .	797

5.1.2.3666STM8S003 [2/2]	797
5.1.2.3667STM8S003F3	797
5.1.2.3668STM8S003K3	797
5.1.2.3669STM8S005 [1/2]	797
5.1.2.3670STM8S005 [2/2]	797
5.1.2.3671STM8S005C6	798
5.1.2.3672STM8S005K6	798
5.1.2.3673STM8S007	798
5.1.2.3674STM8S007C8	798
5.1.2.3675STM8S103 [1/3]	798
5.1.2.3676STM8S103 [2/3]	798
5.1.2.3677STM8S103 [3/3]	799
5.1.2.3678STM8S103F2	799
5.1.2.3679STM8S103F3	799
5.1.2.3680STM8S103K3	799
5.1.2.3681STM8S105 [1/6]	799
5.1.2.3682STM8S105 [2/6]	799
5.1.2.3683STM8S105 [3/6]	800
5.1.2.3684STM8S105 [4/6]	800
5.1.2.3685STM8S105 [5/6]	800
5.1.2.3686STM8S105 [6/6]	800
5.1.2.3687STM8S105C4	800
5.1.2.3688STM8S105C6	800
5.1.2.3689STM8S105K4	801
5.1.2.3690STM8S105K6	801
5.1.2.3691STM8S105S4	801
5.1.2.3692STM8S105S6	801
5.1.2.3693STM8S207 [1/13]	801
5.1.2.3694STM8S207 [2/13]	801
5.1.2.3695STM8S207 [3/13]	802

5.1.2.3696STM8S207 [4/13]	802
5.1.2.3697STM8S207 [5/13]	802
5.1.2.3698STM8S207 [6/13]	802
5.1.2.3699STM8S207 [7/13]	802
5.1.2.3700STM8S207 [8/13]	802
5.1.2.3701STM8S207 [9/13]	803
5.1.2.3702STM8S207 [10/13]	803
5.1.2.3703STM8S207 [11/13]	803
5.1.2.3704STM8S207 [12/13]	803
5.1.2.3705STM8S207 [13/13]	803
5.1.2.3706STM8S207C6	803
5.1.2.3707STM8S207C8	804
5.1.2.3708STM8S207CB	804
5.1.2.3709STM8S207K6	804
5.1.2.3710STM8S207K8	804
5.1.2.3711STM8S207M8	804
5.1.2.3712STM8S207MB	804
5.1.2.3713STM8S207R6	805
5.1.2.3714STM8S207R8	805
5.1.2.3715STM8S207RB	805
5.1.2.3716STM8S207S6	805
5.1.2.3717STM8S207S8	805
5.1.2.3718STM8S207SB	805
5.1.2.3719STM8S208 [1/10]	806
5.1.2.3720STM8S208 [2/10]	806
5.1.2.3721STM8S208 [3/10]	806
5.1.2.3722STM8S208 [4/10]	806
5.1.2.3723STM8S208 [5/10]	806
5.1.2.3724STM8S208 [6/10]	806
5.1.2.3725STM8S208 [7/10]	807

5.1.2.3726STM8S208 [8/10]	807
5.1.2.3727STM8S208 [9/10]	807
5.1.2.3728STM8S208 [10/10]	807
5.1.2.3729STM8S208C6	807
5.1.2.3730STM8S208C8	807
5.1.2.3731STM8S208CB	808
5.1.2.3732STM8S208MB	808
5.1.2.3733STM8S208R6	808
5.1.2.3734STM8S208R8	808
5.1.2.3735STM8S208RB	808
5.1.2.3736STM8S208S6	808
5.1.2.3737STM8S208S8	809
5.1.2.3738STM8S208SB	809
5.1.2.3739STM8S903 [1/2]	809
5.1.2.3740STM8S903 [2/2]	809
5.1.2.3741STM8S903F3	809
5.1.2.3742STM8S903K3	809
5.1.2.3743SW_RESET	810
5.1.2.3744TIM1_AddressBase [1/70]	810
5.1.2.3745TIM1_AddressBase [2/70]	810
5.1.2.3746TIM1_AddressBase [3/70]	810
5.1.2.3747TIM1_AddressBase [4/70]	810
5.1.2.3748TIM1_AddressBase [5/70]	810
5.1.2.3749TIM1_AddressBase [6/70]	811
5.1.2.3750TIM1_AddressBase [7/70]	811
5.1.2.3751TIM1_AddressBase [8/70]	811
5.1.2.3752TIM1_AddressBase [9/70]	811
5.1.2.3753TIM1_AddressBase [10/70]	811
5.1.2.3754TIM1_AddressBase [11/70]	811
5.1.2.3755TIM1_AddressBase [12/70]	812



5.1.2.3756TIM1_AddressBase [13/70]	812
5.1.2.3757TIM1_AddressBase [14/70]	812
5.1.2.3758TIM1_AddressBase [15/70]	812
5.1.2.3759TIM1_AddressBase [16/70]	812
5.1.2.3760TIM1_AddressBase [17/70]	812
5.1.2.3761TIM1_AddressBase [18/70]	813
5.1.2.3762TIM1_AddressBase [19/70]	813
5.1.2.3763TIM1_AddressBase [20/70]	813
5.1.2.3764TIM1_AddressBase [21/70]	813
5.1.2.3765TIM1_AddressBase [22/70]	813
5.1.2.3766TIM1_AddressBase [23/70]	813
5.1.2.3767TIM1_AddressBase [24/70]	814
5.1.2.3768TIM1_AddressBase [25/70]	814
5.1.2.3769TIM1_AddressBase [26/70]	814
5.1.2.3770TIM1_AddressBase [27/70]	814
5.1.2.3771TIM1_AddressBase [28/70]	814
5.1.2.3772TIM1_AddressBase [29/70]	814
5.1.2.3773TIM1_AddressBase [30/70]	815
5.1.2.3774TIM1_AddressBase [31/70]	815
5.1.2.3775TIM1_AddressBase [32/70]	815
5.1.2.3776TIM1_AddressBase [33/70]	815
5.1.2.3777TIM1_AddressBase [34/70]	815
5.1.2.3778TIM1_AddressBase [35/70]	815
5.1.2.3779TIM1_AddressBase [36/70]	816
5.1.2.3780TIM1_AddressBase [37/70]	816
5.1.2.3781TIM1_AddressBase [38/70]	816
5.1.2.3782TIM1_AddressBase [39/70]	816
5.1.2.3783TIM1_AddressBase [40/70]	816
5.1.2.3784TIM1_AddressBase [41/70]	816
5.1.2.3785TIM1_AddressBase [42/70]	817

5.1.2.3786TIM1_AddressBase [43/70]	817
5.1.2.3787TIM1_AddressBase [44/70]	817
5.1.2.3788TIM1_AddressBase [45/70]	817
5.1.2.3789TIM1_AddressBase [46/70]	817
5.1.2.3790TIM1_AddressBase [47/70]	817
5.1.2.3791TIM1_AddressBase [48/70]	818
5.1.2.3792TIM1_AddressBase [49/70]	818
5.1.2.3793TIM1_AddressBase [50/70]	818
5.1.2.3794TIM1_AddressBase [51/70]	818
5.1.2.3795TIM1_AddressBase [52/70]	818
5.1.2.3796TIM1_AddressBase [53/70]	818
5.1.2.3797TIM1_AddressBase [54/70]	819
5.1.2.3798TIM1_AddressBase [55/70]	819
5.1.2.3799TIM1_AddressBase [56/70]	819
5.1.2.3800TIM1_AddressBase [57/70]	819
5.1.2.3801TIM1_AddressBase [58/70]	819
5.1.2.3802TIM1_AddressBase [59/70]	819
5.1.2.3803TIM1_AddressBase [60/70]	820
5.1.2.3804TIM1_AddressBase [61/70]	820
5.1.2.3805TIM1_AddressBase [62/70]	820
5.1.2.3806TIM1_AddressBase [63/70]	820
5.1.2.3807TIM1_AddressBase [64/70]	820
5.1.2.3808TIM1_AddressBase [65/70]	820
5.1.2.3809TIM1_AddressBase [66/70]	821
5.1.2.3810TIM1_AddressBase [67/70]	821
5.1.2.3811TIM1_AddressBase [68/70]	821
5.1.2.3812TIM1_AddressBase [69/70]	821
5.1.2.3813TIM1_AddressBase [70/70]	821
5.1.2.3814TIM2_AddressBase [1/63]	821
5.1.2.3815TIM2_AddressBase [2/63]	822

5.1.2.3816TIM2_AddressBase [3/63]	822
5.1.2.3817TIM2_AddressBase [4/63]	822
5.1.2.3818TIM2_AddressBase [5/63]	822
5.1.2.3819TIM2_AddressBase [6/63]	822
5.1.2.3820TIM2_AddressBase [7/63]	822
5.1.2.3821TIM2_AddressBase [8/63]	823
5.1.2.3822TIM2_AddressBase [9/63]	823
5.1.2.3823TIM2_AddressBase [10/63]	823
5.1.2.3824TIM2_AddressBase [11/63]	823
5.1.2.3825TIM2_AddressBase [12/63]	823
5.1.2.3826TIM2_AddressBase [13/63]	823
5.1.2.3827TIM2_AddressBase [14/63]	824
5.1.2.3828TIM2_AddressBase [15/63]	824
5.1.2.3829TIM2_AddressBase [16/63]	824
5.1.2.3830TIM2_AddressBase [17/63]	824
5.1.2.3831TIM2_AddressBase [18/63]	824
5.1.2.3832TIM2_AddressBase [19/63]	824
5.1.2.3833TIM2_AddressBase [20/63]	825
5.1.2.3834TIM2_AddressBase [21/63]	825
5.1.2.3835TIM2_AddressBase [22/63]	825
5.1.2.3836TIM2_AddressBase [23/63]	825
5.1.2.3837TIM2_AddressBase [24/63]	825
5.1.2.3838TIM2_AddressBase [25/63]	825
5.1.2.3839TIM2_AddressBase [26/63]	826
5.1.2.3840TIM2_AddressBase [27/63]	826
5.1.2.3841TIM2_AddressBase [28/63]	826
5.1.2.3842TIM2_AddressBase [29/63]	826
5.1.2.3843TIM2_AddressBase [30/63]	826
5.1.2.3844TIM2_AddressBase [31/63]	826
5.1.2.3845TIM2_AddressBase [32/63]	827

5.1.2.3846TIM2_AddressBase [33/63]	827
5.1.2.3847TIM2_AddressBase [34/63]	827
5.1.2.3848TIM2_AddressBase [35/63]	827
5.1.2.3849TIM2_AddressBase [36/63]	827
5.1.2.3850TIM2_AddressBase [37/63]	827
5.1.2.3851TIM2_AddressBase [38/63]	828
5.1.2.3852TIM2_AddressBase [39/63]	828
5.1.2.3853TIM2_AddressBase [40/63]	828
5.1.2.3854TIM2_AddressBase [41/63]	828
5.1.2.3855TIM2_AddressBase [42/63]	828
5.1.2.3856TIM2_AddressBase [43/63]	828
5.1.2.3857TIM2_AddressBase [44/63]	829
5.1.2.3858TIM2_AddressBase [45/63]	829
5.1.2.3859TIM2_AddressBase [46/63]	829
5.1.2.3860TIM2_AddressBase [47/63]	829
5.1.2.3861TIM2_AddressBase [48/63]	829
5.1.2.3862TIM2_AddressBase [49/63]	829
5.1.2.3863TIM2_AddressBase [50/63]	830
5.1.2.3864TIM2_AddressBase [51/63]	830
5.1.2.3865TIM2_AddressBase [52/63]	830
5.1.2.3866TIM2_AddressBase [53/63]	830
5.1.2.3867TIM2_AddressBase [54/63]	830
5.1.2.3868TIM2_AddressBase [55/63]	830
5.1.2.3869TIM2_AddressBase [56/63]	831
5.1.2.3870TIM2_AddressBase [57/63]	831
5.1.2.3871TIM2_AddressBase [58/63]	831
5.1.2.3872TIM2_AddressBase [59/63]	831
5.1.2.3873TIM2_AddressBase [60/63]	831
5.1.2.3874TIM2_AddressBase [61/63]	831
5.1.2.3875TIM2_AddressBase [62/63]	832

5.1.2.3876TIM2_AddressBase [63/63]	832
5.1.2.3877TIM3_AddressBase [1/57]	832
5.1.2.3878TIM3_AddressBase [2/57]	832
5.1.2.3879TIM3_AddressBase [3/57]	832
5.1.2.3880TIM3_AddressBase [4/57]	832
5.1.2.3881TIM3_AddressBase [5/57]	833
5.1.2.3882TIM3_AddressBase [6/57]	833
5.1.2.3883TIM3_AddressBase [7/57]	833
5.1.2.3884TIM3_AddressBase [8/57]	833
5.1.2.3885TIM3_AddressBase [9/57]	833
5.1.2.3886TIM3_AddressBase [10/57]	833
5.1.2.3887TIM3_AddressBase [11/57]	834
5.1.2.3888TIM3_AddressBase [12/57]	834
5.1.2.3889TIM3_AddressBase [13/57]	834
5.1.2.3890TIM3_AddressBase [14/57]	834
5.1.2.3891TIM3_AddressBase [15/57]	834
5.1.2.3892TIM3_AddressBase [16/57]	834
5.1.2.3893TIM3_AddressBase [17/57]	835
5.1.2.3894TIM3_AddressBase [18/57]	835
5.1.2.3895TIM3_AddressBase [19/57]	835
5.1.2.3896TIM3_AddressBase [20/57]	835
5.1.2.3897TIM3_AddressBase [21/57]	835
5.1.2.3898TIM3_AddressBase [22/57]	835
5.1.2.3899TIM3_AddressBase [23/57]	836
5.1.2.3900TIM3_AddressBase [24/57]	836
5.1.2.3901TIM3_AddressBase [25/57]	836
5.1.2.3902TIM3_AddressBase [26/57]	836
5.1.2.3903TIM3_AddressBase [27/57]	836
5.1.2.3904TIM3_AddressBase [28/57]	836
5.1.2.3905TIM3_AddressBase [29/57]	837

5.1.2.3906TIM3_AddressBase [30/57]	837
5.1.2.3907TIM3_AddressBase [31/57]	837
5.1.2.3908TIM3_AddressBase [32/57]	837
5.1.2.3909TIM3_AddressBase [33/57]	837
5.1.2.3910TIM3_AddressBase [34/57]	837
5.1.2.3911TIM3_AddressBase [35/57]	838
5.1.2.3912TIM3_AddressBase [36/57]	838
5.1.2.3913TIM3_AddressBase [37/57]	838
5.1.2.3914TIM3_AddressBase [38/57]	838
5.1.2.3915TIM3_AddressBase [39/57]	838
5.1.2.3916TIM3_AddressBase [40/57]	838
5.1.2.3917TIM3_AddressBase [41/57]	839
5.1.2.3918TIM3_AddressBase [42/57]	839
5.1.2.3919TIM3_AddressBase [43/57]	839
5.1.2.3920TIM3_AddressBase [44/57]	839
5.1.2.3921TIM3_AddressBase [45/57]	839
5.1.2.3922TIM3_AddressBase [46/57]	839
5.1.2.3923TIM3_AddressBase [47/57]	840
5.1.2.3924TIM3_AddressBase [48/57]	840
5.1.2.3925TIM3_AddressBase [49/57]	840
5.1.2.3926TIM3_AddressBase [50/57]	840
5.1.2.3927TIM3_AddressBase [51/57]	840
5.1.2.3928TIM3_AddressBase [52/57]	840
5.1.2.3929TIM3_AddressBase [53/57]	841
5.1.2.3930TIM3_AddressBase [54/57]	841
5.1.2.3931TIM3_AddressBase [55/57]	841
5.1.2.3932TIM3_AddressBase [56/57]	841
5.1.2.3933TIM3_AddressBase [57/57]	841
5.1.2.3934TIM4_AddressBase [1/63]	841
5.1.2.3935TIM4_AddressBase [2/63]	842

5.1.2.3936TIM4_AddressBase [3/63]	842
5.1.2.3937TIM4_AddressBase [4/63]	842
5.1.2.3938TIM4_AddressBase [5/63]	842
5.1.2.3939TIM4_AddressBase [6/63]	842
5.1.2.3940TIM4_AddressBase [7/63]	842
5.1.2.3941TIM4_AddressBase [8/63]	843
5.1.2.3942TIM4_AddressBase [9/63]	843
5.1.2.3943TIM4_AddressBase [10/63]	843
5.1.2.3944TIM4_AddressBase [11/63]	843
5.1.2.3945TIM4_AddressBase [12/63]	843
5.1.2.3946TIM4_AddressBase [13/63]	843
5.1.2.3947TIM4_AddressBase [14/63]	844
5.1.2.3948TIM4_AddressBase [15/63]	844
5.1.2.3949TIM4_AddressBase [16/63]	844
5.1.2.3950TIM4_AddressBase [17/63]	844
5.1.2.3951TIM4_AddressBase [18/63]	844
5.1.2.3952TIM4_AddressBase [19/63]	844
5.1.2.3953TIM4_AddressBase [20/63]	845
5.1.2.3954TIM4_AddressBase [21/63]	845
5.1.2.3955TIM4_AddressBase [22/63]	845
5.1.2.3956TIM4_AddressBase [23/63]	845
5.1.2.3957TIM4_AddressBase [24/63]	845
5.1.2.3958TIM4_AddressBase [25/63]	845
5.1.2.3959TIM4_AddressBase [26/63]	846
5.1.2.3960TIM4_AddressBase [27/63]	846
5.1.2.3961TIM4_AddressBase [28/63]	846
5.1.2.3962TIM4_AddressBase [29/63]	846
5.1.2.3963TIM4_AddressBase [30/63]	846
5.1.2.3964TIM4_AddressBase [31/63]	846
5.1.2.3965TIM4_AddressBase [32/63]	847

5.1.2.3966TIM4_AddressBase [33/63]	847
5.1.2.3967TIM4_AddressBase [34/63]	847
5.1.2.3968TIM4_AddressBase [35/63]	847
5.1.2.3969TIM4_AddressBase [36/63]	847
5.1.2.3970TIM4_AddressBase [37/63]	847
5.1.2.3971TIM4_AddressBase [38/63]	848
5.1.2.3972TIM4_AddressBase [39/63]	848
5.1.2.3973TIM4_AddressBase [40/63]	848
5.1.2.3974TIM4_AddressBase [41/63]	848
5.1.2.3975TIM4_AddressBase [42/63]	848
5.1.2.3976TIM4_AddressBase [43/63]	848
5.1.2.3977TIM4_AddressBase [44/63]	849
5.1.2.3978TIM4_AddressBase [45/63]	849
5.1.2.3979TIM4_AddressBase [46/63]	849
5.1.2.3980TIM4_AddressBase [47/63]	849
5.1.2.3981TIM4_AddressBase [48/63]	849
5.1.2.3982TIM4_AddressBase [49/63]	849
5.1.2.3983TIM4_AddressBase [50/63]	850
5.1.2.3984TIM4_AddressBase [51/63]	850
5.1.2.3985TIM4_AddressBase [52/63]	850
5.1.2.3986TIM4_AddressBase [53/63]	850
5.1.2.3987TIM4_AddressBase [54/63]	850
5.1.2.3988TIM4_AddressBase [55/63]	850
5.1.2.3989TIM4_AddressBase [56/63]	851
5.1.2.3990TIM4_AddressBase [57/63]	851
5.1.2.3991TIM4_AddressBase [58/63]	851
5.1.2.3992TIM4_AddressBase [59/63]	851
5.1.2.3993TIM4_AddressBase [60/63]	851
5.1.2.3994TIM4_AddressBase [61/63]	851
5.1.2.3995TIM4_AddressBase [62/63]	852



5.1.2.3996TIM4_AddressBase [63/63]	852
5.1.2.3997TIM5_AddressBase [1/7]	852
5.1.2.3998TIM5_AddressBase [2/7]	852
5.1.2.3999TIM5_AddressBase [3/7]	852
5.1.2.4000TIM5_AddressBase [4/7]	852
5.1.2.4001TIM5_AddressBase [5/7]	853
5.1.2.4002TIM5_AddressBase [6/7]	853
5.1.2.4003TIM5_AddressBase [7/7]	853
5.1.2.4004TIM6_AddressBase [1/7]	853
5.1.2.4005TIM6_AddressBase [2/7]	853
5.1.2.4006TIM6_AddressBase [3/7]	853
5.1.2.4007TIM6_AddressBase [4/7]	854
5.1.2.4008TIM6_AddressBase [5/7]	854
5.1.2.4009TIM6_AddressBase [6/7]	854
5.1.2.4010TIM6_AddressBase [7/7]	854
5.1.2.4011TRIGGER_TRAP	854
5.1.2.4012JART1_AddressBase [1/51]	854
5.1.2.4013JART1_AddressBase [2/51]	855
5.1.2.4014JART1_AddressBase [3/51]	855
5.1.2.4015JART1_AddressBase [4/51]	855
5.1.2.4016JART1_AddressBase [5/51]	855
5.1.2.4017JART1_AddressBase [6/51]	855
5.1.2.4018JART1_AddressBase [7/51]	855
5.1.2.4019JART1_AddressBase [8/51]	856
5.1.2.4020JART1_AddressBase [9/51]	856
5.1.2.4021JART1_AddressBase [10/51]	856
5.1.2.4022JART1_AddressBase [11/51]	856
5.1.2.4023JART1_AddressBase [12/51]	856
5.1.2.4024JART1_AddressBase [13/51]	856
5.1.2.4025JART1_AddressBase [14/51]	857

5.1.2.4026UART1_AddressBase [15/51]	857
5.1.2.4027UART1_AddressBase [16/51]	857
5.1.2.4028UART1_AddressBase [17/51]	857
5.1.2.4029UART1_AddressBase [18/51]	857
5.1.2.4030UART1_AddressBase [19/51]	857
5.1.2.4031UART1_AddressBase [20/51]	858
5.1.2.4032UART1_AddressBase [21/51]	858
5.1.2.4033UART1_AddressBase [22/51]	858
5.1.2.4034UART1_AddressBase [23/51]	858
5.1.2.4035UART1_AddressBase [24/51]	858
5.1.2.4036UART1_AddressBase [25/51]	858
5.1.2.4037UART1_AddressBase [26/51]	859
5.1.2.4038UART1_AddressBase [27/51]	859
5.1.2.4039UART1_AddressBase [28/51]	859
5.1.2.4040UART1_AddressBase [29/51]	859
5.1.2.4041UART1_AddressBase [30/51]	859
5.1.2.4042UART1_AddressBase [31/51]	859
5.1.2.4043UART1_AddressBase [32/51]	860
5.1.2.4044UART1_AddressBase [33/51]	860
5.1.2.4045UART1_AddressBase [34/51]	860
5.1.2.4046UART1_AddressBase [35/51]	860
5.1.2.4047UART1_AddressBase [36/51]	860
5.1.2.4048UART1_AddressBase [37/51]	860
5.1.2.4049UART1_AddressBase [38/51]	861
5.1.2.4050UART1_AddressBase [39/51]	861
5.1.2.4051UART1_AddressBase [40/51]	861
5.1.2.4052UART1_AddressBase [41/51]	861
5.1.2.4053UART1_AddressBase [42/51]	861
5.1.2.4054UART1_AddressBase [43/51]	861
5.1.2.4055UART1_AddressBase [44/51]	862

5.1.2.4056UART1_AddressBase [45/51] . . . . .	862
5.1.2.4057UART1_AddressBase [46/51] . . . . .	862
5.1.2.4058UART1_AddressBase [47/51] . . . . .	862
5.1.2.4059UART1_AddressBase [48/51] . . . . .	862
5.1.2.4060UART1_AddressBase [49/51] . . . . .	862
5.1.2.4061UART1_AddressBase [50/51] . . . . .	863
5.1.2.4062UART1_AddressBase [51/51] . . . . .	863
5.1.2.4063UART2_AddressBase [1/14] . . . . .	863
5.1.2.4064UART2_AddressBase [2/14] . . . . .	863
5.1.2.4065UART2_AddressBase [3/14] . . . . .	863
5.1.2.4066UART2_AddressBase [4/14] . . . . .	863
5.1.2.4067UART2_AddressBase [5/14] . . . . .	864
5.1.2.4068UART2_AddressBase [6/14] . . . . .	864
5.1.2.4069UART2_AddressBase [7/14] . . . . .	864
5.1.2.4070UART2_AddressBase [8/14] . . . . .	864
5.1.2.4071UART2_AddressBase [9/14] . . . . .	864
5.1.2.4072UART2_AddressBase [10/14] . . . . .	864
5.1.2.4073UART2_AddressBase [11/14] . . . . .	865
5.1.2.4074UART2_AddressBase [12/14] . . . . .	865
5.1.2.4075UART2_AddressBase [13/14] . . . . .	865
5.1.2.4076UART2_AddressBase [14/14] . . . . .	865
5.1.2.4077UART3_AddressBase [1/43] . . . . .	865
5.1.2.4078UART3_AddressBase [2/43] . . . . .	865
5.1.2.4079UART3_AddressBase [3/43] . . . . .	866
5.1.2.4080UART3_AddressBase [4/43] . . . . .	866
5.1.2.4081UART3_AddressBase [5/43] . . . . .	866
5.1.2.4082UART3_AddressBase [6/43] . . . . .	866
5.1.2.4083UART3_AddressBase [7/43] . . . . .	866
5.1.2.4084UART3_AddressBase [8/43] . . . . .	866
5.1.2.4085UART3_AddressBase [9/43] . . . . .	867

5.1.2.4086UART3_AddressBase [10/43]	867
5.1.2.4087UART3_AddressBase [11/43]	867
5.1.2.4088UART3_AddressBase [12/43]	867
5.1.2.4089UART3_AddressBase [13/43]	867
5.1.2.4090UART3_AddressBase [14/43]	867
5.1.2.4091UART3_AddressBase [15/43]	868
5.1.2.4092UART3_AddressBase [16/43]	868
5.1.2.4093UART3_AddressBase [17/43]	868
5.1.2.4094UART3_AddressBase [18/43]	868
5.1.2.4095UART3_AddressBase [19/43]	868
5.1.2.4096UART3_AddressBase [20/43]	868
5.1.2.4097UART3_AddressBase [21/43]	869
5.1.2.4098UART3_AddressBase [22/43]	869
5.1.2.4099UART3_AddressBase [23/43]	869
5.1.2.4100UART3_AddressBase [24/43]	869
5.1.2.4101UART3_AddressBase [25/43]	869
5.1.2.4102UART3_AddressBase [26/43]	869
5.1.2.4103UART3_AddressBase [27/43]	870
5.1.2.4104UART3_AddressBase [28/43]	870
5.1.2.4105UART3_AddressBase [29/43]	870
5.1.2.4106UART3_AddressBase [30/43]	870
5.1.2.4107UART3_AddressBase [31/43]	870
5.1.2.4108UART3_AddressBase [32/43]	870
5.1.2.4109UART3_AddressBase [33/43]	871
5.1.2.4110UART3_AddressBase [34/43]	871
5.1.2.4111UART3_AddressBase [35/43]	871
5.1.2.4112UART3_AddressBase [36/43]	871
5.1.2.4113UART3_AddressBase [37/43]	871
5.1.2.4114UART3_AddressBase [38/43]	871
5.1.2.4115UART3_AddressBase [39/43]	872

5.1.2.4116UART3_AddressBase [40/43]	872
5.1.2.4117UART3_AddressBase [41/43]	872
5.1.2.4118UART3_AddressBase [42/43]	872
5.1.2.4119UART3_AddressBase [43/43]	872
5.1.2.4120UART4_AddressBase [1/5]	872
5.1.2.4121UART4_AddressBase [2/5]	873
5.1.2.4122UART4_AddressBase [3/5]	873
5.1.2.4123UART4_AddressBase [4/5]	873
5.1.2.4124UART4_AddressBase [5/5]	873
5.1.2.4125UID_AddressBase [1/34]	873
5.1.2.4126UID_AddressBase [2/34]	873
5.1.2.4127UID_AddressBase [3/34]	874
5.1.2.4128UID_AddressBase [4/34]	874
5.1.2.4129UID_AddressBase [5/34]	874
5.1.2.4130UID_AddressBase [6/34]	874
5.1.2.4131UID_AddressBase [7/34]	874
5.1.2.4132UID_AddressBase [8/34]	874
5.1.2.4133UID_AddressBase [9/34]	875
5.1.2.4134UID_AddressBase [10/34]	875
5.1.2.4135UID_AddressBase [11/34]	875
5.1.2.4136UID_AddressBase [12/34]	875
5.1.2.4137UID_AddressBase [13/34]	875
5.1.2.4138UID_AddressBase [14/34]	875
5.1.2.4139UID_AddressBase [15/34]	876
5.1.2.4140UID_AddressBase [16/34]	876
5.1.2.4141UID_AddressBase [17/34]	876
5.1.2.4142UID_AddressBase [18/34]	876
5.1.2.4143UID_AddressBase [19/34]	876
5.1.2.4144UID_AddressBase [20/34]	876
5.1.2.4145UID_AddressBase [21/34]	877

5.1.2.4146UID_AddressBase [22/34]	877
5.1.2.4147UID_AddressBase [23/34]	877
5.1.2.4148UID_AddressBase [24/34]	877
5.1.2.4149UID_AddressBase [25/34]	877
5.1.2.4150UID_AddressBase [26/34]	877
5.1.2.4151UID_AddressBase [27/34]	878
5.1.2.4152UID_AddressBase [28/34]	878
5.1.2.4153UID_AddressBase [29/34]	878
5.1.2.4154UID_AddressBase [30/34]	878
5.1.2.4155UID_AddressBase [31/34]	878
5.1.2.4156UID_AddressBase [32/34]	878
5.1.2.4157UID_AddressBase [33/34]	879
5.1.2.4158UID_AddressBase [34/34]	879
5.1.2.4159WAIT_FOR_INTERRUPT	879
5.1.2.4160WWDG_AddressBase [1/70]	879
5.1.2.4161WWDG_AddressBase [2/70]	879
5.1.2.4162WWDG_AddressBase [3/70]	879
5.1.2.4163WWDG_AddressBase [4/70]	880
5.1.2.4164WWDG_AddressBase [5/70]	880
5.1.2.4165WWDG_AddressBase [6/70]	880
5.1.2.4166WWDG_AddressBase [7/70]	880
5.1.2.4167WWDG_AddressBase [8/70]	880
5.1.2.4168WWDG_AddressBase [9/70]	880
5.1.2.4169WWDG_AddressBase [10/70]	881
5.1.2.4170WWDG_AddressBase [11/70]	881
5.1.2.4171WWDG_AddressBase [12/70]	881
5.1.2.4172WWDG_AddressBase [13/70]	881
5.1.2.4173WWDG_AddressBase [14/70]	881
5.1.2.4174WWDG_AddressBase [15/70]	881
5.1.2.4175WWDG_AddressBase [16/70]	882

5.1.2.4176WWDG_AddressBase [17/70]	882
5.1.2.4177WWDG_AddressBase [18/70]	882
5.1.2.4178WWDG_AddressBase [19/70]	882
5.1.2.4179WWDG_AddressBase [20/70]	882
5.1.2.4180WWDG_AddressBase [21/70]	882
5.1.2.4181WWDG_AddressBase [22/70]	883
5.1.2.4182WWDG_AddressBase [23/70]	883
5.1.2.4183WWDG_AddressBase [24/70]	883
5.1.2.4184WWDG_AddressBase [25/70]	883
5.1.2.4185WWDG_AddressBase [26/70]	883
5.1.2.4186WWDG_AddressBase [27/70]	883
5.1.2.4187WWDG_AddressBase [28/70]	884
5.1.2.4188WWDG_AddressBase [29/70]	884
5.1.2.4189WWDG_AddressBase [30/70]	884
5.1.2.4190WWDG_AddressBase [31/70]	884
5.1.2.4191WWDG_AddressBase [32/70]	884
5.1.2.4192WWDG_AddressBase [33/70]	884
5.1.2.4193WWDG_AddressBase [34/70]	885
5.1.2.4194WWDG_AddressBase [35/70]	885
5.1.2.4195WWDG_AddressBase [36/70]	885
5.1.2.4196WWDG_AddressBase [37/70]	885
5.1.2.4197WWDG_AddressBase [38/70]	885
5.1.2.4198WWDG_AddressBase [39/70]	885
5.1.2.4199WWDG_AddressBase [40/70]	886
5.1.2.4200WWDG_AddressBase [41/70]	886
5.1.2.4201WWDG_AddressBase [42/70]	886
5.1.2.4202WWDG_AddressBase [43/70]	886
5.1.2.4203WWDG_AddressBase [44/70]	886
5.1.2.4204WWDG_AddressBase [45/70]	886
5.1.2.4205WWDG_AddressBase [46/70]	887

5.1.2.4206WWDG_AddressBase [47/70]	887
5.1.2.4207WWDG_AddressBase [48/70]	887
5.1.2.4208WWDG_AddressBase [49/70]	887
5.1.2.4209WWDG_AddressBase [50/70]	887
5.1.2.4210WWDG_AddressBase [51/70]	887
5.1.2.4211WWDG_AddressBase [52/70]	888
5.1.2.4212WWDG_AddressBase [53/70]	888
5.1.2.4213WWDG_AddressBase [54/70]	888
5.1.2.4214WWDG_AddressBase [55/70]	888
5.1.2.4215WWDG_AddressBase [56/70]	888
5.1.2.4216WWDG_AddressBase [57/70]	888
5.1.2.4217WWDG_AddressBase [58/70]	889
5.1.2.4218WWDG_AddressBase [59/70]	889
5.1.2.4219WWDG_AddressBase [60/70]	889
5.1.2.4220WWDG_AddressBase [61/70]	889
5.1.2.4221WWDG_AddressBase [62/70]	889
5.1.2.4222WWDG_AddressBase [63/70]	889
5.1.2.4223WWDG_AddressBase [64/70]	890
5.1.2.4224WWDG_AddressBase [65/70]	890
5.1.2.4225WWDG_AddressBase [66/70]	890
5.1.2.4226WWDG_AddressBase [67/70]	890
5.1.2.4227WWDG_AddressBase [68/70]	890
5.1.2.4228WWDG_AddressBase [69/70]	890
5.1.2.4229WWDG_AddressBase [70/70]	890



<b>6 Data Structure Documentation</b>	<b>891</b>
6.1 ADC1_t Struct Reference	891
6.1.1 Detailed Description	896
6.1.2 Field Documentation	896
6.1.2.1 ADON	896
6.1.2.2 ALIGN	897
6.1.2.3 AWCRH	897
6.1.2.4 AWCRL	897
6.1.2.5 AWD	897
6.1.2.6 AWDIE	897
6.1.2.7 AWEN	898
6.1.2.8 AWS	898
6.1.2.9 AWSRH	898
6.1.2.10 AWSRL	898
6.1.2.11 CH	898
6.1.2.12 CONT	899
6.1.2.13 CR1	899
6.1.2.14 CR2	899
6.1.2.15 CR3	899
6.1.2.16 CSR	899
6.1.2.17 DATA	900
6.1.2.18 DB0RH	900
6.1.2.19 DB0RL	901
6.1.2.20 DB1RH	901
6.1.2.21 DB1RL	901
6.1.2.22 DB2RH	901
6.1.2.23 DB2RL	901
6.1.2.24 DB3RH	901
6.1.2.25 DB3RL	902
6.1.2.26 DB4RH	902

6.1.2.27	DB4RL	902
6.1.2.28	DB5RH	902
6.1.2.29	DB5RL	902
6.1.2.30	DB6RH	902
6.1.2.31	DB6RL	903
6.1.2.32	DB7RH	903
6.1.2.33	DB7RL	903
6.1.2.34	DB8RH	903
6.1.2.35	DB8RL	903
6.1.2.36	DB9RH	903
6.1.2.37	DB9RL	904
6.1.2.38	DBUF	904
6.1.2.39	DRH	904
6.1.2.40	DRL	904
6.1.2.41	EOC	904
6.1.2.42	EOCIE	904
6.1.2.43	EXTSEL	905
6.1.2.44	EXTTRIG	905
6.1.2.45	HT	905
6.1.2.46	HTRH	905
6.1.2.47	HTRL	905
6.1.2.48	LT	906
6.1.2.49	LTRH	906
6.1.2.50	LTRL	906
6.1.2.51	OVR	906
6.1.2.52	res <sup>[1/2]</sup>	906
6.1.2.53	res <sup>[2/2]</sup>	907
6.1.2.54	res2	907
6.1.2.55	res3	907
6.1.2.56	SCAN	907

6.1.2.57	SPSEL	907
6.1.2.58	TDH	908
6.1.2.59	TDL	908
6.1.2.60	TDRH	908
6.1.2.61	TDRL	908
6.2	ADC2_t Struct Reference	908
6.2.1	Detailed Description	910
6.2.2	Field Documentation	910
6.2.2.1	ADON	910
6.2.2.2	ALIGN	910
6.2.2.3	CH	911
6.2.2.4	CONT	911
6.2.2.5	CR1	911
6.2.2.6	CR2	911
6.2.2.7	CSR	911
6.2.2.8	DATA	912
6.2.2.9	DRH	912
6.2.2.10	DRL	912
6.2.2.11	EOC	912
6.2.2.12	EOCIE	912
6.2.2.13	EXTSEL	913
6.2.2.14	EXTTRIG	913
6.2.2.15	res [1/2]	913
6.2.2.16	res [2/2]	913
6.2.2.17	res2	913
6.2.2.18	res3	914
6.2.2.19	SPSEL	914
6.2.2.20	TDH	914
6.2.2.21	TDL	914
6.2.2.22	TDRH	914

6.2.2.23	TDRL	915
6.3	AWU_t Struct Reference	915
6.3.1	Detailed Description	916
6.3.2	Field Documentation	916
6.3.2.1	APR	916
6.3.2.2	APRE	916
6.3.2.3	AWUEN	916
6.3.2.4	AWUF	916
6.3.2.5	AWUTB	917
6.3.2.6	CSR	917
6.3.2.7	MSR	917
6.3.2.8	res	917
6.3.2.9	res2	917
6.3.2.10	TBR	918
6.4	BEEP_t Struct Reference	918
6.4.1	Detailed Description	918
6.4.2	Field Documentation	918
6.4.2.1	BEEPDIV	918
6.4.2.2	BEEPEN	919
6.4.2.3	BEEPSEL	919
6.4.2.4	CSR	919
6.5	CAN_t Struct Reference	919
6.5.1	Detailed Description	936
6.5.2	Field Documentation	936
6.5.2.1	ABOM	936
6.5.2.2	ABRQ	936
6.5.2.3	ALST	937
6.5.2.4	AWUM	937
6.5.2.5	BOFF	937
6.5.2.6	BOFIE	937

6.5.2.7	BRP	937
6.5.2.8	BS1	938
6.5.2.9	BS2	938
6.5.2.10	BTR1	938
6.5.2.11	BTR2	938
6.5.2.12	CODE	938
6.5.2.13	DATA	939
6.5.2.14	DGR	941
6.5.2.15	DLC	941
6.5.2.16	EIER	941
6.5.2.17	EPVF	941
6.5.2.18	EPVIE	941
6.5.2.19	ERRI	942
6.5.2.20	ERRIE	942
6.5.2.21	ESR	942
6.5.2.22	EWGF	942
6.5.2.23	EWGIE	942
6.5.2.24	EXID	943
6.5.2.25	F0R1	943
6.5.2.26	F0R2	943
6.5.2.27	F0R3	943
6.5.2.28	F0R4	943
6.5.2.29	F0R5	944
6.5.2.30	F0R6	944
6.5.2.31	F0R7	944
6.5.2.32	F0R8	944
6.5.2.33	F1R1	944
6.5.2.34	F1R2	944
6.5.2.35	F1R3	945
6.5.2.36	F1R4	945

6.5.2.37	F1R5	945
6.5.2.38	F1R6	945
6.5.2.39	F1R7	945
6.5.2.40	F1R8	945
6.5.2.41	F2R1	946
6.5.2.42	F2R2	946
6.5.2.43	F2R3	946
6.5.2.44	F2R4	946
6.5.2.45	F2R5	946
6.5.2.46	F2R6	946
6.5.2.47	F2R7	947
6.5.2.48	F2R8	947
6.5.2.49	F3R1	947
6.5.2.50	F3R2	947
6.5.2.51	F3R3	947
6.5.2.52	F3R4	947
6.5.2.53	F3R5	948
6.5.2.54	F3R6	948
6.5.2.55	F3R7	948
6.5.2.56	F3R8	948
6.5.2.57	F4R1	948
6.5.2.58	F4R2	948
6.5.2.59	F4R3	949
6.5.2.60	F4R4	949
6.5.2.61	F4R5	949
6.5.2.62	F4R6	949
6.5.2.63	F4R7	949
6.5.2.64	F4R8	949
6.5.2.65	F5R1	950
6.5.2.66	F5R2	950

6.5.2.67	F5R3	950
6.5.2.68	F5R4	950
6.5.2.69	F5R5	950
6.5.2.70	F5R6	950
6.5.2.71	F5R7	951
6.5.2.72	F5R8	951
6.5.2.73	FACT0	951
6.5.2.74	FACT1	951
6.5.2.75	FACT2	951
6.5.2.76	FACT3	952
6.5.2.77	FACT4	952
6.5.2.78	FACT5	952
6.5.2.79	FCR1	952
6.5.2.80	FCR2	952
6.5.2.81	FCR3	953
6.5.2.82	FFIE	953
6.5.2.83	FMH0	953
6.5.2.84	FMH1	953
6.5.2.85	FMH2	953
6.5.2.86	FMH3	954
6.5.2.87	FMH4	954
6.5.2.88	FMH5	954
6.5.2.89	FMI	954
6.5.2.90	FML0	954
6.5.2.91	FML1	955
6.5.2.92	FML2	955
6.5.2.93	FML3	955
6.5.2.94	FML4	955
6.5.2.95	FML5	955
6.5.2.96	FMP	956

6.5.2.97 FMPIE . . . . .	956
6.5.2.98 FMR1 . . . . .	956
6.5.2.99 FMR2 . . . . .	956
6.5.2.100 FOVIE . . . . .	956
6.5.2.101 FOVR . . . . .	957
6.5.2.102 FSC0 . . . . .	957
6.5.2.103 FSC1 . . . . .	957
6.5.2.104 FSC2 . . . . .	957
6.5.2.105 FSC3 . . . . .	957
6.5.2.106 FSC4 . . . . .	958
6.5.2.107 FSC5 . . . . .	958
6.5.2.108 FULL . . . . .	958
6.5.2.109 ID . . . . .	958
6.5.2.110 IDE . . . . .	958
6.5.2.111 IER . . . . .	959
6.5.2.112 INAK . . . . .	959
6.5.2.113 INRQ . . . . .	959
6.5.2.114 LBKM . . . . .	959
6.5.2.115 LEC . . . . .	959
6.5.2.116 LECIE . . . . .	960
6.5.2.117 LOW0 . . . . .	960
6.5.2.118 LOW1 . . . . .	960
6.5.2.119 LOW2 . . . . .	960
6.5.2.120 MCR . . . . .	960
6.5.2.121 MCSR [1/3] . . . . .	961
6.5.2.122 MCSR [2/3] . . . . .	961
6.5.2.123 MCSR [3/3] . . . . .	961
6.5.2.124 MDAR1 [1/4] . . . . .	961
6.5.2.125 MDAR1 [2/4] . . . . .	961
6.5.2.126 MDAR1 [3/4] . . . . .	961



6.5.2.127 MDAR1 [4/4]	962
6.5.2.128 MDAR2 [1/4]	962
6.5.2.129 MDAR2 [2/4]	962
6.5.2.130 MDAR2 [3/4]	962
6.5.2.131 MDAR2 [4/4]	962
6.5.2.132 MDAR3 [1/4]	962
6.5.2.133 MDAR3 [2/4]	963
6.5.2.134 MDAR3 [3/4]	963
6.5.2.135 MDAR3 [4/4]	963
6.5.2.136 MDAR4 [1/4]	963
6.5.2.137 MDAR4 [2/4]	963
6.5.2.138 MDAR4 [3/4]	963
6.5.2.139 MDAR4 [4/4]	964
6.5.2.140 MDAR5 [1/4]	964
6.5.2.141 MDAR5 [2/4]	964
6.5.2.142 MDAR5 [3/4]	964
6.5.2.143 MDAR5 [4/4]	964
6.5.2.144 MDAR6 [1/4]	964
6.5.2.145 MDAR6 [2/4]	965
6.5.2.146 MDAR6 [3/4]	965
6.5.2.147 MDAR6 [4/4]	965
6.5.2.148 MDAR7 [1/4]	965
6.5.2.149 MDAR7 [2/4]	965
6.5.2.150 MDAR7 [3/4]	965
6.5.2.151 MDAR7 [4/4]	966
6.5.2.152 MDAR8 [1/4]	966
6.5.2.153 MDAR8 [2/4]	966
6.5.2.154 MDAR8 [3/4]	966
6.5.2.155 MDAR8 [4/4]	966
6.5.2.156 MDLCR [1/4]	966

6.5.2.157 MDLCR [2/4]	967
6.5.2.158 MDLCR [3/4]	967
6.5.2.159 MDLCR [4/4]	967
6.5.2.160 MFMIR	967
6.5.2.161 MIDR1 [1/4]	967
6.5.2.162 MIDR1 [2/4]	967
6.5.2.163 MIDR1 [3/4]	968
6.5.2.164 MIDR1 [4/4]	968
6.5.2.165 MIDR2 [1/4]	968
6.5.2.166 MIDR2 [2/4]	968
6.5.2.167 MIDR2 [3/4]	968
6.5.2.168 MIDR2 [4/4]	968
6.5.2.169 MIDR3 [1/4]	969
6.5.2.170 MIDR3 [2/4]	969
6.5.2.171 MIDR3 [3/4]	969
6.5.2.172 MIDR3 [4/4]	969
6.5.2.173 MIDR4 [1/4]	969
6.5.2.174 MIDR4 [2/4]	969
6.5.2.175 MIDR4 [3/4]	970
6.5.2.176 MIDR4 [4/4]	970
6.5.2.177 MSR	970
6.5.2.178 MTSRH [1/4]	970
6.5.2.179 MTSRH [2/4]	970
6.5.2.180 MTSRH [3/4]	970
6.5.2.181 MTSRH [4/4]	971
6.5.2.182 MTSRL [1/4]	971
6.5.2.183 MTSRL [2/4]	971
6.5.2.184 MTSRL [3/4]	971
6.5.2.185 MTSRL [4/4]	971
6.5.2.186 NART	971

6.5.2.187 Page	972
6.5.2.188 PAGE_0	972
6.5.2.189 PAGE_1	972
6.5.2.190 PAGE_2	972
6.5.2.191 PAGE_3	972
6.5.2.192 PAGE_4	972
6.5.2.193 PAGE_5	973
6.5.2.194 PAGE_6	973
6.5.2.195 PAGE_7	973
6.5.2.196 PS	973
6.5.2.197 PSR	973
6.5.2.198 REC	973
6.5.2.199 RECR	974
6.5.2.200 res [1/2]	974
6.5.2.201 res [2/2]	974
6.5.2.202 res2	974
6.5.2.203 RFLM	974
6.5.2.204 RFOM	975
6.5.2.205 RFR	975
6.5.2.206 RQCP	975
6.5.2.207 RQCP0	975
6.5.2.208 RQCP1	975
6.5.2.209 RQCP2	976
6.5.2.210 RTR	976
6.5.2.211 RX	976
6.5.2.212 SAMP	976
6.5.2.213 SILM	976
6.5.2.214 SJW	977
6.5.2.215 SLAK	977
6.5.2.216 SLEEP	977

6.5.2.217 TEC	977
6.5.2.218 TECR	977
6.5.2.219 TERR	978
6.5.2.220 TGT	978
6.5.2.221 TIME	978
6.5.2.222 TME0	978
6.5.2.223 TME1	978
6.5.2.224 TME2	979
6.5.2.225 TMEIE	979
6.5.2.226 TPR	979
6.5.2.227 TSR	979
6.5.2.228 TTCM	979
6.5.2.229 TX	980
6.5.2.230 TXFP	980
6.5.2.231 TXM2E	980
6.5.2.232 TXOK	980
6.5.2.233 TXOK0	980
6.5.2.234 TXOK1	981
6.5.2.235 TXOK2	981
6.5.2.236 TXRQ	981
6.5.2.237 WKUI	981
6.5.2.238 WKUIE	981
6.6 CFG_t Struct Reference	982
6.6.1 Detailed Description	982
6.6.2 Field Documentation	982
6.6.2.1 AL	982
6.6.2.2 GCR	982
6.6.2.3 res	983
6.6.2.4 SWD	983
6.7 CLK_t Struct Reference	983

6.7.1	Detailed Description	986
6.7.2	Field Documentation	986
6.7.2.1	AUX	986
6.7.2.2	CCOBSY	986
6.7.2.3	CCOEN	987
6.7.2.4	CCOR	987
6.7.2.5	CCORDY	987
6.7.2.6	CCOSEL	987
6.7.2.7	CKDIVR	987
6.7.2.8	CKM	988
6.7.2.9	CMSR	988
6.7.2.10	CPUDIV	988
6.7.2.11	CSSD	988
6.7.2.12	CSSDIE	988
6.7.2.13	CSSEN	989
6.7.2.14	CSSR	989
6.7.2.15	ECKR	989
6.7.2.16	FHWU	989
6.7.2.17	HSEEN	989
6.7.2.18	HSERDY	990
6.7.2.19	HSIDIV	990
6.7.2.20	HSIEN	990
6.7.2.21	HSIRDY	990
6.7.2.22	HSITRIM	990
6.7.2.23	HSITRIMR	991
6.7.2.24	ICKR	991
6.7.2.25	LSIEN	991
6.7.2.26	LSIRDY	991
6.7.2.27	PCKEN_ADC	991
6.7.2.28	PCKEN_AWU	992

6.7.2.29	PCKEN_CAN	992
6.7.2.30	PCKEN_I2C	992
6.7.2.31	PCKEN_SPI	992
6.7.2.32	PCKEN_TIM1	992
6.7.2.33	PCKEN_TIM2_TIM5	993
6.7.2.34	PCKEN_TIM3	993
6.7.2.35	PCKEN_TIM4_TIM6	993
6.7.2.36	PCKEN_UART1	993
6.7.2.37	PCKEN_UART2	993
6.7.2.38	PCKENR1	994
6.7.2.39	PCKENR2	994
6.7.2.40	REGAH	994
6.7.2.41	res [1/2]	994
6.7.2.42	res [2/2]	994
6.7.2.43	res2 [1/2]	995
6.7.2.44	res2 [2/2]	995
6.7.2.45	SWBSY	995
6.7.2.46	SWCR	995
6.7.2.47	SWEN	995
6.7.2.48	SWI	996
6.7.2.49	SWIEN	996
6.7.2.50	SWIF	996
6.7.2.51	SWIMCCR	996
6.7.2.52	SWIMCLK	996
6.7.2.53	SWR	997
6.8	EXTI_t Struct Reference	997
6.8.1	Detailed Description	997
6.8.2	Field Documentation	997
6.8.2.1	CR1	998
6.8.2.2	CR2	998

6.8.2.3	PAIS	998
6.8.2.4	PBIS	998
6.8.2.5	PCIS	998
6.8.2.6	PDIS	999
6.8.2.7	PEIS	999
6.8.2.8	res	999
6.8.2.9	TLIS	999
6.9	FLASH_t Struct Reference	999
6.9.1	Detailed Description	1001
6.9.2	Field Documentation	1001
6.9.2.1	AHALT	1002
6.9.2.2	CR1	1002
6.9.2.3	CR2	1002
6.9.2.4	DUK	1002
6.9.2.5	DUKR	1002
6.9.2.6	DUL	1002
6.9.2.7	EOP	1003
6.9.2.8	ERASE	1003
6.9.2.9	FIX	1003
6.9.2.10	FPR	1003
6.9.2.11	FPRG	1003
6.9.2.12	HALT	1004
6.9.2.13	HVOFF	1004
6.9.2.14	IAPSR	1004
6.9.2.15	IE	1004
6.9.2.16	NCR2	1004
6.9.2.17	NERASE	1005
6.9.2.18	NFPR	1005
6.9.2.19	NFPRG	1005
6.9.2.20	NOPT	1005

6.9.2.21	NPRG	1005
6.9.2.22	NWPB	1006
6.9.2.23	NWPRG	1006
6.9.2.24	OPT	1006
6.9.2.25	PRG	1006
6.9.2.26	PUK	1006
6.9.2.27	PUKR	1007
6.9.2.28	PUL	1007
6.9.2.29	res [1/2]	1007
6.9.2.30	res [2/2]	1007
6.9.2.31	res2 [1/2]	1007
6.9.2.32	res2 [2/2]	1008
6.9.2.33	WPB	1008
6.9.2.34	WPRG	1008
6.9.2.35	WR_PG_DIS	1008
6.10	I2C_t Struct Reference	1008
6.10.1	Detailed Description	1011
6.10.2	Field Documentation	1012
6.10.2.1	ACK	1012
6.10.2.2	ADD	1012
6.10.2.3	ADD0	1012
6.10.2.4	ADD10	1012
6.10.2.5	ADDCONF	1013
6.10.2.6	ADDMODE	1013
6.10.2.7	ADDR	1013
6.10.2.8	AF	1013
6.10.2.9	ARLO	1013
6.10.2.10	BERR	1014
6.10.2.11	BTF	1014
6.10.2.12	BUSY	1014



6.10.2.13 CCR	1014
6.10.2.14 CCRH	1014
6.10.2.15 CCRL	1015
6.10.2.16 CR1	1015
6.10.2.17 CR2	1015
6.10.2.18 DATA	1015
6.10.2.19 DR	1015
6.10.2.20 DUTY	1015
6.10.2.21 ENGC	1016
6.10.2.22 FREQ	1016
6.10.2.23 FREQR	1016
6.10.2.24 FS	1016
6.10.2.25 GENCALL	1016
6.10.2.26 ITBUFEN	1017
6.10.2.27 ITERREN	1017
6.10.2.28 ITEVTEN	1017
6.10.2.29 ITR	1017
6.10.2.30 MSL	1017
6.10.2.31 NOSTRETCH	1018
6.10.2.32 OARH	1018
6.10.2.33 OARL	1018
6.10.2.34 OVR	1018
6.10.2.35 PE	1018
6.10.2.36 POS	1019
6.10.2.37 res <sup>[1/2]</sup>	1019
6.10.2.38 res <sup>[2/2]</sup>	1019
6.10.2.39 res2 <sup>[1/2]</sup>	1019
6.10.2.40 res2 <sup>[2/2]</sup>	1019
6.10.2.41 RXNE	1020
6.10.2.42 SB	1020

6.10.2.43 SR1	1020
6.10.2.44 SR2	1020
6.10.2.45 SR3	1020
6.10.2.46 START	1020
6.10.2.47 STOP	1021
6.10.2.48 STOPF	1021
6.10.2.49 SWRST	1021
6.10.2.50 TRA	1021
6.10.2.51 TRISE	1021
6.10.2.52 TRISER	1022
6.10.2.53 TXE	1022
6.10.2.54 WUFH	1022
6.11 ITC_t Struct Reference	1022
6.11.1 Detailed Description	1024
6.11.2 Field Documentation	1024
6.11.2.1 res	1025
6.11.2.2 SPR1	1025
6.11.2.3 SPR2	1025
6.11.2.4 SPR3	1025
6.11.2.5 SPR4	1025
6.11.2.6 SPR5	1025
6.11.2.7 SPR6	1026
6.11.2.8 SPR7	1026
6.11.2.9 SPR8	1026
6.11.2.10 VECT10SPR	1026
6.11.2.11 VECT11SPR	1026
6.11.2.12 VECT12SPR	1026
6.11.2.13 VECT13SPR	1027
6.11.2.14 VECT14SPR	1027
6.11.2.15 VECT15SPR	1027

6.11.2.16 VECT16SPR . . . . .	1027
6.11.2.17 VECT17SPR . . . . .	1027
6.11.2.18 VECT18SPR . . . . .	1028
6.11.2.19 VECT19SPR . . . . .	1028
6.11.2.20 VECT1SPR . . . . .	1028
6.11.2.21 VECT20SPR . . . . .	1028
6.11.2.22 VECT21SPR . . . . .	1028
6.11.2.23 VECT22SPR . . . . .	1029
6.11.2.24 VECT23SPR . . . . .	1029
6.11.2.25 VECT24SPR . . . . .	1029
6.11.2.26 VECT25SPR . . . . .	1029
6.11.2.27 VECT26SPR . . . . .	1029
6.11.2.28 VECT27SPR . . . . .	1030
6.11.2.29 VECT28SPR . . . . .	1030
6.11.2.30 VECT29SPR . . . . .	1030
6.11.2.31 VECT2SPR . . . . .	1030
6.11.2.32 VECT3SPR . . . . .	1030
6.11.2.33 VECT4SPR . . . . .	1031
6.11.2.34 VECT5SPR . . . . .	1031
6.11.2.35 VECT6SPR . . . . .	1031
6.11.2.36 VECT7SPR . . . . .	1031
6.11.2.37 VECT8SPR . . . . .	1031
6.11.2.38 VECT9SPR . . . . .	1032
6.12 IWDG_t Struct Reference . . . . .	1032
6.12.1 Detailed Description . . . . .	1032
6.12.2 Field Documentation . . . . .	1032
6.12.2.1 KEY . . . . .	1033
6.12.2.2 KR . . . . .	1033
6.12.2.3 PR . . . . .	1033
6.12.2.4 PRE . . . . .	1033

6.12.2.5	res	1033
6.12.2.6	RL	1034
6.12.2.7	RLR	1034
6.13	PORT_t Struct Reference	1034
6.13.1	Detailed Description	1036
6.13.2	Field Documentation	1036
6.13.2.1	CR1	1036
6.13.2.2	CR2	1036
6.13.2.3	DDR	1037
6.13.2.4	IDR	1037
6.13.2.5	ODR	1037
6.13.2.6	PIN0	1037
6.13.2.7	PIN1	1037
6.13.2.8	PIN2	1038
6.13.2.9	PIN3	1038
6.13.2.10	PIN4	1038
6.13.2.11	PIN5	1039
6.13.2.12	PIN6	1039
6.13.2.13	PIN7	1039
6.14	RST_t Struct Reference	1040
6.14.1	Detailed Description	1040
6.14.2	Field Documentation	1040
6.14.2.1	EMCF	1040
6.14.2.2	ILLOPF	1041
6.14.2.3	IWDGF	1041
6.14.2.4	res	1041
6.14.2.5	SR	1041
6.14.2.6	SWIMF	1041
6.14.2.7	WWDGF	1042
6.15	SPI_t Struct Reference	1042

6.15.1 Detailed Description . . . . .	1044
6.15.2 Field Documentation . . . . .	1044
6.15.2.1 BDM . . . . .	1044
6.15.2.2 BDOE . . . . .	1044
6.15.2.3 BR . . . . .	1044
6.15.2.4 BSY . . . . .	1045
6.15.2.5 CPHA . . . . .	1045
6.15.2.6 CPOL . . . . .	1045
6.15.2.7 CR1 . . . . .	1045
6.15.2.8 CR2 . . . . .	1045
6.15.2.9 CRCEN . . . . .	1046
6.15.2.10 CRCERR . . . . .	1046
6.15.2.11 CRCNEXT . . . . .	1046
6.15.2.12 CRCPOLY . . . . .	1046
6.15.2.13 CRCPR . . . . .	1046
6.15.2.14 DATA . . . . .	1047
6.15.2.15 DR . . . . .	1047
6.15.2.16 ERRIE . . . . .	1047
6.15.2.17 ICR . . . . .	1047
6.15.2.18 LSBFIRST . . . . .	1047
6.15.2.19 MODF . . . . .	1048
6.15.2.20 MSTR . . . . .	1048
6.15.2.21 OVR . . . . .	1048
6.15.2.22 res . . . . .	1048
6.15.2.23 RxCRC . . . . .	1048
6.15.2.24 RXCRCR . . . . .	1049
6.15.2.25 RXIE . . . . .	1049
6.15.2.26 RXNE . . . . .	1049
6.15.2.27 RXONLY . . . . .	1049
6.15.2.28 SPE . . . . .	1049

6.15.2.29 SR . . . . .	1050
6.15.2.30 SSI . . . . .	1050
6.15.2.31 SSM . . . . .	1050
6.15.2.32 TxCRC . . . . .	1050
6.15.2.33 TXCRCR . . . . .	1050
6.15.2.34 TXE . . . . .	1051
6.15.2.35 TXIE . . . . .	1051
6.15.2.36 WKIE . . . . .	1051
6.15.2.37 WKUP . . . . .	1051
6.16 TIM1_t Struct Reference . . . . .	1051
6.16.1 Detailed Description . . . . .	1059
6.16.2 Field Documentation . . . . .	1059
6.16.2.1 AOE . . . . .	1059
6.16.2.2 ARPE . . . . .	1059
6.16.2.3 ARR . . . . .	1059
6.16.2.4 ARRH . . . . .	1060
6.16.2.5 ARRL . . . . .	1060
6.16.2.6 BG . . . . .	1060
6.16.2.7 BIE . . . . .	1060
6.16.2.8 BIF . . . . .	1060
6.16.2.9 BKE . . . . .	1061
6.16.2.10 BKP . . . . .	1061
6.16.2.11 BKR . . . . .	1061
6.16.2.12 CC1E . . . . .	1061
6.16.2.13 CC1G . . . . .	1061
6.16.2.14 CC1IE . . . . .	1062
6.16.2.15 CC1IF . . . . .	1062
6.16.2.16 CC1NE . . . . .	1062
6.16.2.17 CC1NP . . . . .	1062
6.16.2.18 CC1OF . . . . .	1062

6.16.2.19 CC1P . . . . .	1063
6.16.2.20 CC1S . . . . .	1063
6.16.2.21 CC2E . . . . .	1063
6.16.2.22 CC2G . . . . .	1063
6.16.2.23 CC2IE . . . . .	1063
6.16.2.24 CC2IF . . . . .	1064
6.16.2.25 CC2NE . . . . .	1064
6.16.2.26 CC2NP . . . . .	1064
6.16.2.27 CC2OF . . . . .	1064
6.16.2.28 CC2P . . . . .	1064
6.16.2.29 CC2S . . . . .	1065
6.16.2.30 CC3E . . . . .	1065
6.16.2.31 CC3G . . . . .	1065
6.16.2.32 CC3IE . . . . .	1065
6.16.2.33 CC3IF . . . . .	1065
6.16.2.34 CC3NE . . . . .	1066
6.16.2.35 CC3NP . . . . .	1066
6.16.2.36 CC3OF . . . . .	1066
6.16.2.37 CC3P . . . . .	1066
6.16.2.38 CC3S . . . . .	1066
6.16.2.39 CC4E . . . . .	1067
6.16.2.40 CC4G . . . . .	1067
6.16.2.41 CC4IE . . . . .	1067
6.16.2.42 CC4IF . . . . .	1067
6.16.2.43 CC4OF . . . . .	1067
6.16.2.44 CC4P . . . . .	1068
6.16.2.45 CC4S . . . . .	1068
6.16.2.46 CCER1 . . . . .	1068
6.16.2.47 CCER2 . . . . .	1068
6.16.2.48 CCMR1 . . . . .	1068

6.16.2.49 CCMR2	1068
6.16.2.50 CCMR3	1069
6.16.2.51 CCMR4	1069
6.16.2.52 CCPC	1069
6.16.2.53 CCR1	1069
6.16.2.54 CCR1H	1069
6.16.2.55 CCR1L	1070
6.16.2.56 CCR2	1070
6.16.2.57 CCR2H	1070
6.16.2.58 CCR2L	1070
6.16.2.59 CCR3	1070
6.16.2.60 CCR3H	1071
6.16.2.61 CCR3L	1071
6.16.2.62 CCR4	1071
6.16.2.63 CCR4H	1071
6.16.2.64 CCR4L	1071
6.16.2.65 CEN	1071
6.16.2.66 CMS	1072
6.16.2.67 CNT	1072
6.16.2.68 CNTRH	1072
6.16.2.69 CNTRL	1072
6.16.2.70 COMG	1072
6.16.2.71 COMIE	1073
6.16.2.72 COMIF	1073
6.16.2.73 COMS	1073
6.16.2.74 CR1	1073
6.16.2.75 CR2	1073
6.16.2.76 DIR	1074
6.16.2.77 DTG	1074
6.16.2.78 DTR	1074



6.16.2.79 ECE	1074
6.16.2.80 EGR	1074
6.16.2.81 ETF	1075
6.16.2.82 ETP	1075
6.16.2.83 ETPS	1075
6.16.2.84 ETR	1075
6.16.2.85 IC1F	1075
6.16.2.86 IC1PSC	1076
6.16.2.87 IC2F	1076
6.16.2.88 IC2PSC	1076
6.16.2.89 IC3F	1076
6.16.2.90 IC3PSC	1076
6.16.2.91 IC4F	1077
6.16.2.92 IC4PSC	1077
6.16.2.93 IER	1077
6.16.2.94 IN <sup>[1/4]</sup>	1077
6.16.2.95 IN <sup>[2/4]</sup>	1077
6.16.2.96 IN <sup>[3/4]</sup>	1077
6.16.2.97 IN <sup>[4/4]</sup>	1078
6.16.2.98 LOCK	1078
6.16.2.99 MMS	1078
6.16.2.100 MOE	1078
6.16.2.101 MSM	1078
6.16.2.102 OC1CE	1079
6.16.2.103 OC1FE	1079
6.16.2.104 OC1M	1079
6.16.2.105 OC1PE	1079
6.16.2.106 OC2CE	1079
6.16.2.107 OC2FE	1080
6.16.2.108 OC2M	1080

6.16.2.109OC2PE . . . . .	1080
6.16.2.110OC3CE . . . . .	1080
6.16.2.111OC3FE . . . . .	1080
6.16.2.112OC3M . . . . .	1081
6.16.2.113OC3PE . . . . .	1081
6.16.2.114OC4CE . . . . .	1081
6.16.2.115OC4FE . . . . .	1081
6.16.2.116OC4M . . . . .	1081
6.16.2.117OC4PE . . . . .	1082
6.16.2.118OIS1 . . . . .	1082
6.16.2.119OIS1N . . . . .	1082
6.16.2.120OIS2 . . . . .	1082
6.16.2.121OIS2N . . . . .	1082
6.16.2.122OIS3 . . . . .	1083
6.16.2.123OIS3N . . . . .	1083
6.16.2.124OIS4 . . . . .	1083
6.16.2.125OISR . . . . .	1083
6.16.2.126OPM . . . . .	1083
6.16.2.127OSSI . . . . .	1084
6.16.2.128OSSR . . . . .	1084
6.16.2.129OUT [1/4] . . . . .	1084
6.16.2.130OUT [2/4] . . . . .	1084
6.16.2.131OUT [3/4] . . . . .	1084
6.16.2.132OUT [4/4] . . . . .	1084
6.16.2.133PSC . . . . .	1085
6.16.2.134PSCRH . . . . .	1085
6.16.2.135PSCRL . . . . .	1085
6.16.2.136RCR . . . . .	1085
6.16.2.137REP . . . . .	1085
6.16.2.138res . . . . .	1086

6.16.2.139res2	1086
6.16.2.140res3	1086
6.16.2.141SMCR	1086
6.16.2.142SMS	1086
6.16.2.143SR1	1087
6.16.2.144SR2	1087
6.16.2.145TG	1087
6.16.2.146TIE	1087
6.16.2.147TIF	1087
6.16.2.148TS	1088
6.16.2.149UDIS	1088
6.16.2.150UG	1088
6.16.2.151UIE	1088
6.16.2.152JIF	1088
6.16.2.153JRS	1089
6.17 TIM2_t Struct Reference	1089
6.17.1 Detailed Description	1093
6.17.2 Field Documentation	1093
6.17.2.1 ARPE	1094
6.17.2.2 ARR	1094
6.17.2.3 ARRH	1094
6.17.2.4 ARRL	1094
6.17.2.5 CC1E	1094
6.17.2.6 CC1G	1095
6.17.2.7 CC1IE	1095
6.17.2.8 CC1IF	1095
6.17.2.9 CC1OF	1095
6.17.2.10 CC1P	1095
6.17.2.11 CC1S	1096
6.17.2.12 CC2E	1096

6.17.2.13 CC2G . . . . .	1096
6.17.2.14 CC2IE . . . . .	1096
6.17.2.15 CC2IF . . . . .	1096
6.17.2.16 CC2OF . . . . .	1097
6.17.2.17 CC2P . . . . .	1097
6.17.2.18 CC2S . . . . .	1097
6.17.2.19 CC3E . . . . .	1097
6.17.2.20 CC3G . . . . .	1097
6.17.2.21 CC3IE . . . . .	1098
6.17.2.22 CC3IF . . . . .	1098
6.17.2.23 CC3OF . . . . .	1098
6.17.2.24 CC3P . . . . .	1098
6.17.2.25 CC3S . . . . .	1098
6.17.2.26 CCER1 . . . . .	1099
6.17.2.27 CCER2 . . . . .	1099
6.17.2.28 CCMR1 . . . . .	1099
6.17.2.29 CCMR2 . . . . .	1099
6.17.2.30 CCMR3 . . . . .	1099
6.17.2.31 CCR1 . . . . .	1099
6.17.2.32 CCR1H . . . . .	1100
6.17.2.33 CCR1L . . . . .	1100
6.17.2.34 CCR2 . . . . .	1100
6.17.2.35 CCR2H . . . . .	1100
6.17.2.36 CCR2L . . . . .	1100
6.17.2.37 CCR3 . . . . .	1101
6.17.2.38 CCR3H . . . . .	1101
6.17.2.39 CCR3L . . . . .	1101
6.17.2.40 CEN . . . . .	1101
6.17.2.41 CNT . . . . .	1101
6.17.2.42 CNTRH . . . . .	1102

6.17.2.43 CNTRL . . . . .	1102
6.17.2.44 CR1 . . . . .	1102
6.17.2.45 EGR . . . . .	1102
6.17.2.46 IC1F . . . . .	1102
6.17.2.47 IC1PSC . . . . .	1102
6.17.2.48 IC2F . . . . .	1103
6.17.2.49 IC2PSC . . . . .	1103
6.17.2.50 IC3F . . . . .	1103
6.17.2.51 IC3PSC . . . . .	1103
6.17.2.52 IER . . . . .	1103
6.17.2.53 IN <sup>[1/3]</sup> . . . . .	1104
6.17.2.54 IN <sup>[2/3]</sup> . . . . .	1104
6.17.2.55 IN <sup>[3/3]</sup> . . . . .	1104
6.17.2.56 OC1M . . . . .	1104
6.17.2.57 OC1PE . . . . .	1104
6.17.2.58 OC2M . . . . .	1104
6.17.2.59 OC2PE . . . . .	1105
6.17.2.60 OC3CE . . . . .	1105
6.17.2.61 OC3M . . . . .	1105
6.17.2.62 OC3PE . . . . .	1105
6.17.2.63 OPM . . . . .	1105
6.17.2.64 OUT <sup>[1/3]</sup> . . . . .	1106
6.17.2.65 OUT <sup>[2/3]</sup> . . . . .	1106
6.17.2.66 OUT <sup>[3/3]</sup> . . . . .	1106
6.17.2.67 PSC . . . . .	1106
6.17.2.68 PSCR . . . . .	1106
6.17.2.69 res . . . . .	1106
6.17.2.70 res2 . . . . .	1107
6.17.2.71 SR1 . . . . .	1107
6.17.2.72 SR2 . . . . .	1107

6.17.2.73 UDIS . . . . .	1107
6.17.2.74 UG . . . . .	1107
6.17.2.75 UIE . . . . .	1108
6.17.2.76 UIF . . . . .	1108
6.17.2.77 URS . . . . .	1108
6.18 TIM3_t Struct Reference . . . . .	1108
6.18.1 Detailed Description . . . . .	1112
6.18.2 Field Documentation . . . . .	1112
6.18.2.1 ARPE . . . . .	1112
6.18.2.2 ARR . . . . .	1112
6.18.2.3 ARRH . . . . .	1113
6.18.2.4 ARRL . . . . .	1113
6.18.2.5 CC1E . . . . .	1113
6.18.2.6 CC1G . . . . .	1113
6.18.2.7 CC1IE . . . . .	1113
6.18.2.8 CC1IF . . . . .	1114
6.18.2.9 CC1OF . . . . .	1114
6.18.2.10 CC1P . . . . .	1114
6.18.2.11 CC1S . . . . .	1114
6.18.2.12 CC2E . . . . .	1114
6.18.2.13 CC2G . . . . .	1115
6.18.2.14 CC2IE . . . . .	1115
6.18.2.15 CC2IF . . . . .	1115
6.18.2.16 CC2OF . . . . .	1115
6.18.2.17 CC2P . . . . .	1115
6.18.2.18 CC2S . . . . .	1116
6.18.2.19 CCER1 . . . . .	1116
6.18.2.20 CCMR1 . . . . .	1116
6.18.2.21 CCMR2 . . . . .	1116
6.18.2.22 CCR1 . . . . .	1116

6.18.2.23 CCR1H . . . . .	1117
6.18.2.24 CCR1L . . . . .	1117
6.18.2.25 CCR2 . . . . .	1117
6.18.2.26 CCR2H . . . . .	1117
6.18.2.27 CCR2L . . . . .	1117
6.18.2.28 CEN . . . . .	1117
6.18.2.29 CNT . . . . .	1118
6.18.2.30 CNTRH . . . . .	1118
6.18.2.31 CNTRL . . . . .	1118
6.18.2.32 CR1 . . . . .	1118
6.18.2.33 EGR . . . . .	1118
6.18.2.34 IC1F . . . . .	1118
6.18.2.35 IC1PSC . . . . .	1119
6.18.2.36 IC2F . . . . .	1119
6.18.2.37 IC2PSC . . . . .	1119
6.18.2.38 IER . . . . .	1119
6.18.2.39 IN <sup>[1/2]</sup> . . . . .	1119
6.18.2.40 IN <sup>[2/2]</sup> . . . . .	1120
6.18.2.41 OC1M . . . . .	1120
6.18.2.42 OC1PE . . . . .	1120
6.18.2.43 OC2M . . . . .	1120
6.18.2.44 OC2PE . . . . .	1120
6.18.2.45 OPM . . . . .	1121
6.18.2.46 OUT <sup>[1/2]</sup> . . . . .	1121
6.18.2.47 OUT <sup>[2/2]</sup> . . . . .	1121
6.18.2.48 PSC . . . . .	1121
6.18.2.49 PSCR . . . . .	1121
6.18.2.50 res . . . . .	1122
6.18.2.51 res2 . . . . .	1122
6.18.2.52 SR1 . . . . .	1122

6.18.2.53 SR2	1122
6.18.2.54 UDIS	1122
6.18.2.55 UG	1123
6.18.2.56 UIE	1123
6.18.2.57 UIF	1123
6.18.2.58 URS	1123
6.19 TIM4_t Struct Reference	1123
6.19.1 Detailed Description	1125
6.19.2 Field Documentation	1125
6.19.2.1 ARPE	1125
6.19.2.2 ARR [1/2]	1125
6.19.2.3 ARR [2/2]	1125
6.19.2.4 CEN	1126
6.19.2.5 CNT	1126
6.19.2.6 CNTR	1126
6.19.2.7 CR	1126
6.19.2.8 EGR	1126
6.19.2.9 IER	1126
6.19.2.10 OPM	1127
6.19.2.11 PSC	1127
6.19.2.12 PSCR	1127
6.19.2.13 res	1127
6.19.2.14 SR	1127
6.19.2.15 UDIS	1128
6.19.2.16 UG	1128
6.19.2.17 UIE	1128
6.19.2.18 UIF	1128
6.19.2.19 URS	1128
6.20 TIM5_t Struct Reference	1129
6.20.1 Detailed Description	1134



6.20.2 Field Documentation . . . . .	1134
6.20.2.1 ARPE . . . . .	1134
6.20.2.2 ARR . . . . .	1134
6.20.2.3 ARRH . . . . .	1134
6.20.2.4 ARRL . . . . .	1135
6.20.2.5 CC1E . . . . .	1135
6.20.2.6 CC1G . . . . .	1135
6.20.2.7 CC1IE . . . . .	1135
6.20.2.8 CC1IF . . . . .	1135
6.20.2.9 CC1OF . . . . .	1136
6.20.2.10 CC1P . . . . .	1136
6.20.2.11 CC1S . . . . .	1136
6.20.2.12 CC2E . . . . .	1136
6.20.2.13 CC2G . . . . .	1136
6.20.2.14 CC2IE . . . . .	1137
6.20.2.15 CC2IF . . . . .	1137
6.20.2.16 CC2OF . . . . .	1137
6.20.2.17 CC2P . . . . .	1137
6.20.2.18 CC2S . . . . .	1137
6.20.2.19 CC3E . . . . .	1138
6.20.2.20 CC3G . . . . .	1138
6.20.2.21 CC3IE . . . . .	1138
6.20.2.22 CC3IF . . . . .	1138
6.20.2.23 CC3OF . . . . .	1138
6.20.2.24 CC3P . . . . .	1139
6.20.2.25 CC3S . . . . .	1139
6.20.2.26 CCER1 . . . . .	1139
6.20.2.27 CCER2 . . . . .	1139
6.20.2.28 CCMR1 . . . . .	1139
6.20.2.29 CCMR2 . . . . .	1139

6.20.2.30 CCMR3	1140
6.20.2.31 CCPC	1140
6.20.2.32 CCR1	1140
6.20.2.33 CCR1H	1140
6.20.2.34 CCR1L	1140
6.20.2.35 CCR2	1141
6.20.2.36 CCR2H	1141
6.20.2.37 CCR2L	1141
6.20.2.38 CCR3	1141
6.20.2.39 CCR3H	1141
6.20.2.40 CCR3L	1142
6.20.2.41 CEN	1142
6.20.2.42 CNT	1142
6.20.2.43 CNTRH	1142
6.20.2.44 CNTRL	1142
6.20.2.45 COMS	1143
6.20.2.46 CR1	1143
6.20.2.47 CR2	1143
6.20.2.48 EGR	1143
6.20.2.49 IC1F	1143
6.20.2.50 IC1PSC	1143
6.20.2.51 IC2F	1144
6.20.2.52 IC2PSC	1144
6.20.2.53 IC3F	1144
6.20.2.54 IC3PSC	1144
6.20.2.55 IER	1144
6.20.2.56 IN <sup>[1/3]</sup>	1145
6.20.2.57 IN <sup>[2/3]</sup>	1145
6.20.2.58 IN <sup>[3/3]</sup>	1145
6.20.2.59 MMS	1145

6.20.2.60 MSM . . . . .	1145
6.20.2.61 OC1M . . . . .	1145
6.20.2.62 OC1PE . . . . .	1146
6.20.2.63 OC2M . . . . .	1146
6.20.2.64 OC2PE . . . . .	1146
6.20.2.65 OC3M . . . . .	1146
6.20.2.66 OC3PE . . . . .	1146
6.20.2.67 OPM . . . . .	1147
6.20.2.68 OUT [1/3] . . . . .	1147
6.20.2.69 OUT [2/3] . . . . .	1147
6.20.2.70 OUT [3/3] . . . . .	1147
6.20.2.71 PSC . . . . .	1147
6.20.2.72 PSCR . . . . .	1147
6.20.2.73 res . . . . .	1148
6.20.2.74 res2 . . . . .	1148
6.20.2.75 res3 . . . . .	1148
6.20.2.76 SMCR . . . . .	1148
6.20.2.77 SMS . . . . .	1148
6.20.2.78 SR1 . . . . .	1149
6.20.2.79 SR2 . . . . .	1149
6.20.2.80 TG . . . . .	1149
6.20.2.81 TIE . . . . .	1149
6.20.2.82 TIF . . . . .	1149
6.20.2.83 TS . . . . .	1150
6.20.2.84 UDIS . . . . .	1150
6.20.2.85 UG . . . . .	1150
6.20.2.86 UIE . . . . .	1150
6.20.2.87 UIF . . . . .	1150
6.20.2.88 URS . . . . .	1151
6.21 TIM6_t Struct Reference . . . . .	1151

6.21.1 Detailed Description	1152
6.21.2 Field Documentation	1152
6.21.2.1 ARPE	1153
6.21.2.2 ARR [1/2]	1153
6.21.2.3 ARR [2/2]	1153
6.21.2.4 CEN	1153
6.21.2.5 CNT	1153
6.21.2.6 CNTR	1154
6.21.2.7 CR1	1154
6.21.2.8 CR2	1154
6.21.2.9 EGR	1154
6.21.2.10 IER	1154
6.21.2.11 MMS	1154
6.21.2.12 MSM	1155
6.21.2.13 OPM	1155
6.21.2.14 PSC	1155
6.21.2.15 PSCR	1155
6.21.2.16 res	1155
6.21.2.17 res2	1156
6.21.2.18 SMCR	1156
6.21.2.19 SMS	1156
6.21.2.20 SR	1156
6.21.2.21 TS	1156
6.21.2.22 UDIS	1157
6.21.2.23 UG	1157
6.21.2.24 UIE	1157
6.21.2.25 UIF	1157
6.21.2.26 URS	1157
6.22 UART1_t Struct Reference	1158
6.22.1 Detailed Description	1160

6.22.2 Field Documentation	1160
6.22.2.1 ADD	1160
6.22.2.2 BRR1	1161
6.22.2.3 BRR2	1161
6.22.2.4 CKEN	1161
6.22.2.5 CPHA	1161
6.22.2.6 CPOL	1161
6.22.2.7 CR1	1162
6.22.2.8 CR2	1162
6.22.2.9 CR3	1162
6.22.2.10 CR4	1162
6.22.2.11 CR5	1162
6.22.2.12 DATA	1162
6.22.2.13 DIV_0_3	1163
6.22.2.14 DIV_12_15	1163
6.22.2.15 DIV_4_11	1163
6.22.2.16 DR	1163
6.22.2.17 FE	1163
6.22.2.18 GT	1164
6.22.2.19 GTR	1164
6.22.2.20 HDSEL	1164
6.22.2.21 IDLE	1164
6.22.2.22 ILIEN	1164
6.22.2.23 IREN	1165
6.22.2.24 IRLP	1165
6.22.2.25 LBCL	1165
6.22.2.26 LBDF	1165
6.22.2.27 LBDIEN	1165
6.22.2.28 LBDL	1166
6.22.2.29 LINEN	1166

6.22.2.30 M	1166
6.22.2.31 NACK	1166
6.22.2.32 NF	1166
6.22.2.33 OR	1167
6.22.2.34 PCEN	1167
6.22.2.35 PE	1167
6.22.2.36 PIEN	1167
6.22.2.37 PS	1167
6.22.2.38 PSC	1168
6.22.2.39 PSCR	1168
6.22.2.40 R8	1168
6.22.2.41 REN	1168
6.22.2.42 res	1168
6.22.2.43 res2	1169
6.22.2.44 RIEN	1169
6.22.2.45 RWU	1169
6.22.2.46 RXNE	1169
6.22.2.47 SBK	1169
6.22.2.48 SCEN	1170
6.22.2.49 SR	1170
6.22.2.50 STOP	1170
6.22.2.51 T8	1170
6.22.2.52 TC	1170
6.22.2.53 TCEN	1171
6.22.2.54 TEN	1171
6.22.2.55 TIEN	1171
6.22.2.56 TXE	1171
6.22.2.57 UARTD	1171
6.22.2.58 WAKE	1172
6.23 UART2_t Struct Reference	1172

6.23.1 Detailed Description . . . . .	1175
6.23.2 Field Documentation . . . . .	1175
6.23.2.1 ADD . . . . .	1175
6.23.2.2 BRR1 . . . . .	1175
6.23.2.3 BRR2 . . . . .	1175
6.23.2.4 CKEN . . . . .	1176
6.23.2.5 CPHA . . . . .	1176
6.23.2.6 CPOL . . . . .	1176
6.23.2.7 CR1 . . . . .	1176
6.23.2.8 CR2 . . . . .	1176
6.23.2.9 CR3 . . . . .	1177
6.23.2.10 CR4 . . . . .	1177
6.23.2.11 CR5 . . . . .	1177
6.23.2.12 CR6 . . . . .	1177
6.23.2.13 DATA . . . . .	1177
6.23.2.14 DIV_0_3 . . . . .	1177
6.23.2.15 DIV_12_15 . . . . .	1178
6.23.2.16 DIV_4_11 . . . . .	1178
6.23.2.17 DR . . . . .	1178
6.23.2.18 FE . . . . .	1178
6.23.2.19 GT . . . . .	1178
6.23.2.20 GTR . . . . .	1179
6.23.2.21 IDLE . . . . .	1179
6.23.2.22 ILIEN . . . . .	1179
6.23.2.23 IREN . . . . .	1179
6.23.2.24 IRLP . . . . .	1179
6.23.2.25 LASE . . . . .	1180
6.23.2.26 LBCL . . . . .	1180
6.23.2.27 LBDF . . . . .	1180
6.23.2.28 LBDIEN . . . . .	1180

6.23.2.29 LBDL . . . . .	1180
6.23.2.30 LDUM . . . . .	1181
6.23.2.31 LHDF . . . . .	1181
6.23.2.32 LHDEN . . . . .	1181
6.23.2.33 LINEN . . . . .	1181
6.23.2.34 LSF . . . . .	1181
6.23.2.35 LSLV . . . . .	1182
6.23.2.36 M . . . . .	1182
6.23.2.37 NACK . . . . .	1182
6.23.2.38 NF . . . . .	1182
6.23.2.39 OR . . . . .	1182
6.23.2.40 PCEN . . . . .	1183
6.23.2.41 PE . . . . .	1183
6.23.2.42 PIEN . . . . .	1183
6.23.2.43 PS . . . . .	1183
6.23.2.44 PSC . . . . .	1183
6.23.2.45 PSCR . . . . .	1184
6.23.2.46 R8 . . . . .	1184
6.23.2.47 REN . . . . .	1184
6.23.2.48 res . . . . .	1184
6.23.2.49 res2 . . . . .	1184
6.23.2.50 res3 . . . . .	1185
6.23.2.51 RIEN . . . . .	1185
6.23.2.52 RWU . . . . .	1185
6.23.2.53 RXNE . . . . .	1185
6.23.2.54 SBK . . . . .	1185
6.23.2.55 SCEN . . . . .	1186
6.23.2.56 SR . . . . .	1186
6.23.2.57 STOP . . . . .	1186
6.23.2.58 T8 . . . . .	1186



6.23.2.59 TC	1186
6.23.2.60 TCEN	1187
6.23.2.61 TEN	1187
6.23.2.62 TIEN	1187
6.23.2.63 TXE	1187
6.23.2.64 UARTD	1187
6.23.2.65 WAKE	1188
6.24 UART3_t Struct Reference	1188
6.24.1 Detailed Description	1190
6.24.2 Field Documentation	1190
6.24.2.1 ADD	1191
6.24.2.2 BRR1	1191
6.24.2.3 BRR2	1191
6.24.2.4 CR1	1191
6.24.2.5 CR2	1191
6.24.2.6 CR3	1191
6.24.2.7 CR4	1192
6.24.2.8 CR6	1192
6.24.2.9 DATA	1192
6.24.2.10 DIV_0_3	1192
6.24.2.11 DIV_12_15	1192
6.24.2.12 DIV_4_11	1193
6.24.2.13 DR	1193
6.24.2.14 FE	1193
6.24.2.15 IDLE	1193
6.24.2.16 ILIEN	1193
6.24.2.17 LASE	1194
6.24.2.18 LBDF	1194
6.24.2.19 LBDIEN	1194
6.24.2.20 LBDL	1194

6.24.2.21 LDUM . . . . .	1194
6.24.2.22 LHDF . . . . .	1195
6.24.2.23 LHDEN . . . . .	1195
6.24.2.24 LINEN . . . . .	1195
6.24.2.25 LSF . . . . .	1195
6.24.2.26 LSLV . . . . .	1195
6.24.2.27 M . . . . .	1196
6.24.2.28 NF . . . . .	1196
6.24.2.29 OR . . . . .	1196
6.24.2.30 PCEN . . . . .	1196
6.24.2.31 PE . . . . .	1196
6.24.2.32 PIEN . . . . .	1197
6.24.2.33 PS . . . . .	1197
6.24.2.34 R8 . . . . .	1197
6.24.2.35 REN . . . . .	1197
6.24.2.36 res <sup>[1/2]</sup> . . . . .	1197
6.24.2.37 res <sup>[2/2]</sup> . . . . .	1198
6.24.2.38 res2 . . . . .	1198
6.24.2.39 RIEN . . . . .	1198
6.24.2.40 RWU . . . . .	1198
6.24.2.41 RXNE . . . . .	1198
6.24.2.42 SBK . . . . .	1199
6.24.2.43 SR . . . . .	1199
6.24.2.44 STOP . . . . .	1199
6.24.2.45 T8 . . . . .	1199
6.24.2.46 TC . . . . .	1199
6.24.2.47 TCEN . . . . .	1200
6.24.2.48 TEN . . . . .	1200
6.24.2.49 TIEN . . . . .	1200
6.24.2.50 TXE . . . . .	1200

6.24.2.51 UARTD . . . . .	1200
6.24.2.52 WAKE . . . . .	1201
6.25 UART4_t Struct Reference . . . . .	1201
6.25.1 Detailed Description . . . . .	1204
6.25.2 Field Documentation . . . . .	1204
6.25.2.1 ADD . . . . .	1204
6.25.2.2 BRR1 . . . . .	1204
6.25.2.3 BRR2 . . . . .	1204
6.25.2.4 CKEN . . . . .	1205
6.25.2.5 CPHA . . . . .	1205
6.25.2.6 CPOL . . . . .	1205
6.25.2.7 CR1 . . . . .	1205
6.25.2.8 CR2 . . . . .	1205
6.25.2.9 CR3 . . . . .	1206
6.25.2.10 CR4 . . . . .	1206
6.25.2.11 CR5 . . . . .	1206
6.25.2.12 CR6 . . . . .	1206
6.25.2.13 DATA . . . . .	1206
6.25.2.14 DIV_0_3 . . . . .	1206
6.25.2.15 DIV_12_15 . . . . .	1207
6.25.2.16 DIV_4_11 . . . . .	1207
6.25.2.17 DR . . . . .	1207
6.25.2.18 FE . . . . .	1207
6.25.2.19 GT . . . . .	1207
6.25.2.20 GTR . . . . .	1208
6.25.2.21 HDSEL . . . . .	1208
6.25.2.22 IDLE . . . . .	1208
6.25.2.23 ILIEN . . . . .	1208
6.25.2.24 IREN . . . . .	1208
6.25.2.25 IRLP . . . . .	1209

6.25.2.26 LASE . . . . .	1209
6.25.2.27 LBCL . . . . .	1209
6.25.2.28 LBDF . . . . .	1209
6.25.2.29 LBDIEN . . . . .	1209
6.25.2.30 LBDL . . . . .	1210
6.25.2.31 LDUM . . . . .	1210
6.25.2.32 LHDF . . . . .	1210
6.25.2.33 LHDIEN . . . . .	1210
6.25.2.34 LINEN . . . . .	1210
6.25.2.35 LSF . . . . .	1211
6.25.2.36 LSLV . . . . .	1211
6.25.2.37 M . . . . .	1211
6.25.2.38 NACK . . . . .	1211
6.25.2.39 NF . . . . .	1211
6.25.2.40 OR . . . . .	1212
6.25.2.41 PCEN . . . . .	1212
6.25.2.42 PE . . . . .	1212
6.25.2.43 PIEN . . . . .	1212
6.25.2.44 PS . . . . .	1212
6.25.2.45 PSC . . . . .	1213
6.25.2.46 PSCR . . . . .	1213
6.25.2.47 R8 . . . . .	1213
6.25.2.48 REN . . . . .	1213
6.25.2.49 res . . . . .	1213
6.25.2.50 res2 . . . . .	1214
6.25.2.51 RIEN . . . . .	1214
6.25.2.52 RWU . . . . .	1214
6.25.2.53 RXNE . . . . .	1214
6.25.2.54 SBK . . . . .	1214
6.25.2.55 SCEN . . . . .	1215

6.25.2.56 SR . . . . .	1215
6.25.2.57 STOP . . . . .	1215
6.25.2.58 T8 . . . . .	1215
6.25.2.59 TC . . . . .	1215
6.25.2.60 TCEN . . . . .	1216
6.25.2.61 TEN . . . . .	1216
6.25.2.62 TIEN . . . . .	1216
6.25.2.63 TXE . . . . .	1216
6.25.2.64 UARTD . . . . .	1216
6.25.2.65 WAKE . . . . .	1217
6.26 WWDG_t Struct Reference . . . . .	1217
6.26.1 Detailed Description . . . . .	1217
6.26.2 Field Documentation . . . . .	1217
6.26.2.1 CR . . . . .	1218
6.26.2.2 res . . . . .	1218
6.26.2.3 T . . . . .	1218
6.26.2.4 W . . . . .	1218
6.26.2.5 WDGA . . . . .	1218
6.26.2.6 WR . . . . .	1218
<b>7 File Documentation</b>	<b>1219</b>
7.1 pages/mainpage.txt File Reference . . . . .	1219
7.2 /home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5268.h File Reference . . . . .	1219
7.3 /home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5269.h File Reference . . . . .	1220
7.4 /home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5286.h File Reference . . . . .	1222
7.5 /home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5288.h File Reference . . . . .	1223
7.6 /home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5289.h File Reference . . . . .	1224
7.7 /home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF528A.h File Reference . . . . .	1226

7.8	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52A6.h</a> File Reference . . . . .	1227
7.9	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52A8.h</a> File Reference . . . . .	1228
7.10	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52A9.h</a> File Reference . . . . .	1230
7.11	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52AA.h</a> File Reference . . . . .	1231
7.12	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6213.h</a> File Reference . . . . .	1232
7.13	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6213A.h</a> File Reference . . . . .	1234
7.14	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6223.h</a> File Reference . . . . .	1235
7.15	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6223A.h</a> File Reference . . . . .	1236
7.16	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6226.h</a> File Reference . . . . .	1237
7.17	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6246.h</a> File Reference . . . . .	1238
7.18	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6248.h</a> File Reference . . . . .	1240
7.19	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6266.h</a> File Reference . . . . .	1241
7.20	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6268.h</a> File Reference . . . . .	1242
7.21	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6269.h</a> File Reference . . . . .	1243
7.22	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6286.h</a> File Reference . . . . .	1245
7.23	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6288.h</a> File Reference . . . . .	1246
7.24	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6289.h</a> File Reference . . . . .	1247
7.25	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF628A.h</a> File Reference . . . . .	1248
7.26	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62A6.h</a> File Reference . . . . .	1250
7.27	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62A8.h</a> File Reference . . . . .	1251

7.28	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62A9.h</a> File Reference . . . . .	1252
7.29	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62↵ AA.h</a> File Reference . . . . .	1254
7.30	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6366.h</a> File Reference . . . . .	1255
7.31	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6388.h</a> File Reference . . . . .	1256
7.32	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_ST↵ M8S.h</a> File Reference . . . . .	1258
7.33	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S001J3.h</a> File Reference . . . . .	1331
7.34	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S003F3.h</a> File Reference . . . . .	1332
7.35	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S003K3.h</a> File Reference . . . . .	1334
7.36	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S005C6.h</a> File Reference . . . . .	1335
7.37	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S005K6.h</a> File Reference . . . . .	1336
7.38	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S007C8.h</a> File Reference . . . . .	1337
7.39	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S103F2.h</a> File Reference . . . . .	1339
7.40	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S103F3.h</a> File Reference . . . . .	1340
7.41	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S103K3.h</a> File Reference . . . . .	1341
7.42	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105C4.h</a> File Reference . . . . .	1342
7.43	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105C6.h</a> File Reference . . . . .	1344
7.44	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105K4.h</a> File Reference . . . . .	1345
7.45	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105K6.h</a> File Reference . . . . .	1346
7.46	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105S4.h</a> File Reference . . . . .	1348
7.47	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105S6.h</a> File Reference . . . . .	1349

7.48	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207C6.h</a> File Reference . . . . .	1350
7.49	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207C8.h</a> File Reference . . . . .	1352
7.50	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207CB.h</a> File Reference . . . . .	1353
7.51	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207K6.h</a> File Reference . . . . .	1354
7.52	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207K8.h</a> File Reference . . . . .	1356
7.53	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207↵ M8.h</a> File Reference . . . . .	1357
7.54	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207↵ MB.h</a> File Reference . . . . .	1358
7.55	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207R6.h</a> File Reference . . . . .	1360
7.56	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207R8.h</a> File Reference . . . . .	1361
7.57	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207RB.h</a> File Reference . . . . .	1362
7.58	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207S6.h</a> File Reference . . . . .	1364
7.59	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207S8.h</a> File Reference . . . . .	1365
7.60	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207SB.h</a> File Reference . . . . .	1366
7.61	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208C6.h</a> File Reference . . . . .	1368
7.62	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208C8.h</a> File Reference . . . . .	1369
7.63	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208CB.h</a> File Reference . . . . .	1370
7.64	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208↵ MB.h</a> File Reference . . . . .	1372
7.65	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208R6.h</a> File Reference . . . . .	1373
7.66	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208R8.h</a> File Reference . . . . .	1374
7.67	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208RB.h</a> File Reference . . . . .	1376
7.68	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208S6.h</a> File Reference . . . . .	1377
7.69	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208S8.h</a> File Reference . . . . .	1378
7.70	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208SB.h</a> File Reference . . . . .	1380
7.71	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S903F3.h</a> File Reference . . . . .	1381
7.72	<a href="#">/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S903K3.h</a> File Reference . . . . .	1382







# Chapter 1

## Main Page

### 1.1 Device Overview

This is a set of open-source device headers for the 8-bit [STM8 microcontroller](#) family by [ST↵Microelectronics](#). For a brief introduction to the STM8 see e.g. [here](#) or [here](#).

Due to a superb performance to cost ratio, low current consumption and ease-of-use, STM8s are widely used in low-end electronic [gadgets](#), and a lot of [development boards](#) are readily available.

### 1.2 Toolchains

Several commercial cross-compilers support the STM8, namely [Cosmic](#), [IAR](#) and [Raisonance](#). In addition, the open-source compiler [SDCC](#) also supports the STM8 since v3.4.0.

All commercial compilers come with device headers pre-installed, but these are mutually incompatible and (of course) not open-source. SDCC does not come with STM8 device headers due to license issues.

To facilitate development, STM provides the so-called [Standard Peripheral Library \(SPL\)](#) which contains not only device headers, but also standardized peripherals functions. The SPL is compatible with all above commercial compilers out of the box, and with SDCC via a [patch](#). Unfortunately the SPL is also not 100% open-source, as discussed [here](#) or [here](#).

### 1.3 Open-Source Headers

A lot of (more or less complete) STM8 device headers is available on the internet, e.g. [here](#) or [here](#). Alternatively one can also use the SPL headers, or those from the above commercial compilers (if license allows). However, these are all mutually incompatible, and thus hinder the exchange of code snippets between projects. Compare this to the [Arduino](#) and [avr-libc](#) multiverse... :-)

Following a longer [discussion](#), here is a proposal for open-source STM8 device headers, published under [G↵PL3+](#). They are intended for inclusion into the [SDCC](#) distribution, but are compatible with all above compilers. My final aim is to provide standardized device headers for all available STM8 devices, which can be used in open-source projects without restrictions, and facilitate code exchange withon the STM8 OS-community.

### 1.3.1 Content

- device headers for all STM8AS and STM8S devices (70 of 138 STM8 devices and growing)
- this reference (created by [Doxygen](#))
- example projects for SDCC, Cosmic, IAR and Raisonance toolchains
  - pin toggle (aka blink)
  - timer interrupts
  - serial communication (incl. printf() and gets())
  - analog measurements
  - PWM generation
  - mixing with SPL headers and routines

### 1.3.2 Features

- direct/bitwise or byte-wise access to all peripheral registers
- assembler macros for common operations like [NOP\(\)](#) or [DISABLE\\_INTERRUPTS\(\)](#)
- can be mixed with SPL headers and functions (see example [blink\\_TIM4\\_SPL](#))

Example:

```
// device header (STM8 Discovery board)
#include "STM8S105C6.h"

// configure LED pin to output push-pull (bitmasks)
_GPIOD_DDR |= _GPIO_PIN0;           // input(=0) or output(=1)
_GPIOD_CR1 |= _GPIO_PIN0;           // input: 0=float, 1=pull-up; output: 0=open-drain,
                                     // 1=push-pull
_GPIOD_CR2 |= _GPIO_PIN0;           // input: 0=no exint, 1=exint; output: 0=2MHz slope,
                                     // 1=10MHz slope

// main loop
while (1) {

    // toggle LED
    //_GPIOD_ODR ^= _GPIO_PIN0;       // byte access (smaller)
    _GPIOD_ODR.PIN0 ^= 1;             // bit access (more convenient)

    // wait a bit
    for (uint16_t i=20000L; i; i--)
        NOP();

} // main loop
```

## Chapter 2

# Module Index

### 2.1 Modules

Here is a list of all modules:

STM8AF_STM8S . . . . .	9
------------------------	---



## Chapter 3

# Data Structure Index

### 3.1 Data Structures

Here are the data structures with brief descriptions:

<a href="#">ADC1_t</a>	Struct containing Analog Digital Converter 1 (ADC1)	891
<a href="#">ADC2_t</a>	Struct containing Analog Digital Converter 2 (ADC2)	908
<a href="#">AWU_t</a>	Struct for configuring the Auto Wake-Up Module (AWU)	915
<a href="#">BEEP_t</a>	Struct for beeper control (BEEP)	918
<a href="#">CAN_t</a>	Struct for controlling Controller Area Network Module (CAN)	919
<a href="#">CFG_t</a>	Struct for Global Configuration registers (CFG)	982
<a href="#">CLK_t</a>	Struct for configuring/monitoring clock module (CLK)	983
<a href="#">EXTI_t</a>	Struct for configuring external port interrupts (EXTI)	997
<a href="#">FLASH_t</a>	Struct to control write/erase of flash memory (FLASH)	999
<a href="#">I2C_t</a>	Struct for controlling I2C module (I2C)	1008
<a href="#">ITC_t</a>	Struct for setting interrupt Priority (ITC)	1022
<a href="#">IWDG_t</a>	Struct for access to Independent Timeout Watchdog registers (IWDG)	1032
<a href="#">PORT_t</a>	Structure for controlling pins in PORT mode (PORTx, x=A..I)	1034
<a href="#">RST_t</a>	Struct for determining reset source (RST)	1040
<a href="#">SPI_t</a>	Struct for controlling SPI module (SPI)	1042
<a href="#">TIM1_t</a>	Struct for controlling 16-Bit Timer 1 (TIM1)	1051
<a href="#">TIM2_t</a>	Struct for controlling 16-Bit Timer 2 (TIM2)	1089
<a href="#">TIM3_t</a>	Struct for controlling 16-Bit Timer 3 (TIM3)	1108

<a href="#">TIM4_t</a>	Struct for controlling 8-Bit Timer 4 (TIM4) . . . . .	<a href="#">1123</a>
<a href="#">TIM5_t</a>	Struct for controlling 16-Bit Timer 5 (TIM5) . . . . .	<a href="#">1129</a>
<a href="#">TIM6_t</a>	Struct for controlling 8-Bit Timer 6 (TIM6) . . . . .	<a href="#">1151</a>
<a href="#">UART1_t</a>	Struct for controlling Universal Asynchronous Receiver Transmitter 1 (UART1) . . . . .	<a href="#">1158</a>
<a href="#">UART2_t</a>	Struct for controlling Universal Asynchronous Receiver Transmitter 2 (UART2) . . . . .	<a href="#">1172</a>
<a href="#">UART3_t</a>	Struct for controlling Universal Asynchronous Receiver Transmitter 3 (UART3) . . . . .	<a href="#">1188</a>
<a href="#">UART4_t</a>	Struct for controlling Universal Asynchronous Receiver Transmitter 4 (UART4) . . . . .	<a href="#">1201</a>
<a href="#">WWDG_t</a>	Struct for access to Window Watchdog registers (WWDG) . . . . .	<a href="#">1217</a>



## Chapter 4

# File Index

### 4.1 File List

Here is a list of all files with brief descriptions:

/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5268.h	1219
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5269.h	1220
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5286.h	1222
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5288.h	1223
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5289.h	1224
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF528A.h	1226
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52A6.h	1227
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52A8.h	1228
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52A9.h	1230
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52AA.h	1231
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6213.h	1232
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6213A.h	1234
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6223.h	1235
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6223A.h	1236
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6226.h	1237
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6246.h	1238
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6248.h	1240
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6266.h	1241
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6268.h	1242
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6269.h	1243
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6286.h	1245
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6288.h	1246
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6289.h	1247
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF628A.h	1248
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62A6.h	1250
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62A8.h	1251
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62A9.h	1252
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF62AA.h	1254
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6366.h	1255
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6388.h	1256
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h	1258
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S001J3.h	1331
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S003F3.h	1332
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S003K3.h	1334
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S005C6.h	1335

/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S005K6.h	1336
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S007C8.h	1337
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S103F2.h	1339
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S103F3.h	1340
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S103K3.h	1341
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105C4.h	1342
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105C6.h	1344
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105K4.h	1345
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105K6.h	1346
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105S4.h	1348
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105S6.h	1349
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207C6.h	1350
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207C8.h	1352
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207CB.h	1353
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207K6.h	1354
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207K8.h	1356
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207M8.h	1357
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207MB.h	1358
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207R6.h	1360
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207R8.h	1361
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207RB.h	1362
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207S6.h	1364
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207S8.h	1365
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207SB.h	1366
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208C6.h	1368
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208C8.h	1369
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208CB.h	1370
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208MB.h	1372
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208R6.h	1373
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208R8.h	1374
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208RB.h	1376
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208S6.h	1377
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208S8.h	1378
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208SB.h	1380
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S903F3.h	1381
/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S903K3.h	1382

## Chapter 5

# Module Documentation

### 5.1 STM8AF\_STM8S

#### Data Structures

- struct [PORT\\_t](#)  
*structure for controlling pins in PORT mode (PORTx, x=A..I)*
- struct [FLASH\\_t](#)  
*struct to control write/erase of flash memory (FLASH)*
- struct [EXTI\\_t](#)  
*struct for configuring external port interrupts (EXTI)*
- struct [RST\\_t](#)  
*struct for determining reset source (RST)*
- struct [CLK\\_t](#)  
*struct for configuring/monitoring clock module (CLK)*
- struct [WWDG\\_t](#)  
*struct for access to Window Watchdog registers (WWDG)*
- struct [IWDG\\_t](#)  
*struct for access to Independent Timeout Watchdog registers (IWDG)*
- struct [AWU\\_t](#)  
*struct for configuring the Auto Wake-Up Module (AWU)*
- struct [BEEP\\_t](#)  
*struct for beeper control (BEEP)*
- struct [SPI\\_t](#)  
*struct for controlling SPI module (SPI)*
- struct [I2C\\_t](#)  
*struct for controlling I2C module (I2C)*
- struct [UART1\\_t](#)  
*struct for controlling Universal Asynchronous Receiver Transmitter 1 (UART1)*
- struct [UART2\\_t](#)  
*struct for controlling Universal Asynchronous Receiver Transmitter 2 (UART2)*
- struct [UART3\\_t](#)  
*struct for controlling Universal Asynchronous Receiver Transmitter 3 (UART3)*
- struct [UART4\\_t](#)  
*struct for controlling Universal Asynchronous Receiver Transmitter 4 (UART4)*
- struct [TIM1\\_t](#)

- struct for controlling 16-Bit Timer 1 (TIM1)*
- struct [TIM2\\_t](#)
  - struct for controlling 16-Bit Timer 2 (TIM2)*
- struct [TIM3\\_t](#)
  - struct for controlling 16-Bit Timer 3 (TIM3)*
- struct [TIM4\\_t](#)
  - struct for controlling 8-Bit Timer 4 (TIM4)*
- struct [TIM5\\_t](#)
  - struct for controlling 16-Bit Timer 5 (TIM5)*
- struct [TIM6\\_t](#)
  - struct for controlling 8-Bit Timer 6 (TIM6)*
- struct [ADC1\\_t](#)
  - struct containing Analog Digital Converter 1 (ADC1)*
- struct [ADC2\\_t](#)
  - struct containing Analog Digital Converter 2 (ADC2)*
- struct [CAN\\_t](#)
  - struct for controlling Controller Area Network Module (CAN)*
- struct [CFG\\_t](#)
  - struct for Global Configuration registers (CFG)*
- struct [ITC\\_t](#)
  - struct for setting interrupt Priority (ITC)*

## Macros

- #define [STM8AF5268](#)
- #define [STM8AF526x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 32\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 1024
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250

- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF5269
- #define STM8AF526x
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF5286
- #define STM8AF528x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014

- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define STM8AF5288`
- `#define STM8AF528x`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`

- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF5289
- #define STM8AF528x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF528A
- #define STM8AF528x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E



- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define STM8AF52A6`
- `#define STM8AF52Ax`
- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`



- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF52A8
- #define STM8AF52Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF52A9
- #define STM8AF52Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028

- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [STM8AF52AA](#)
- #define [STM8AF52Ax](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 128\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60

- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6213
- #define STM8AF621x
- #define STM8\_PFLASH\_SIZE 4\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART4\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM5\_AddressBase 0x5300
- #define TIM6\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6213A
- #define STM8AF621x
- #define STM8\_PFLASH\_SIZE 4\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART4\_AddressBase 0x5230

- `#define TIM1_AddressBase 0x5250`
- `#define TIM5_AddressBase 0x5300`
- `#define TIM6_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define STM8AF6223`
- `#define STM8AF622x`
- `#define STM8_PFLASH_SIZE 8*1024`
- `#define STM8_RAM_SIZE 1*1024`
- `#define STM8_EEPROM_SIZE 640`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART4_AddressBase 0x5230`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM5_AddressBase 0x5300`
- `#define TIM6_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define STM8AF6223A`
- `#define STM8AF622x`
- `#define STM8_PFLASH_SIZE 8*1024`
- `#define STM8_RAM_SIZE 1*1024`
- `#define STM8_EEPROM_SIZE 640`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`

- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART4\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM5\_AddressBase 0x5300
- #define TIM6\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6226
- #define STM8AF622x
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART4\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM5\_AddressBase 0x5300
- #define TIM6\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6246
- #define STM8AF624x
- #define STM8\_PFLASH\_SIZE 16\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 512
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E

- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6248
- #define STM8AF624x
- #define STM8\_PFLASH\_SIZE 16\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 512
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6266
- #define STM8AF626x
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 2\*1024

- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6268
- #define STM8AF626x
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320

- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6269
- #define STM8AF626x
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6286
- #define STM8AF628x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3



- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6288
- #define STM8AF628x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6289
- #define STM8AF628x
- #define STM8\_PFLASH\_SIZE 64\*1024

- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [STM8AF628A](#)
- #define [STM8AF628x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 64\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3

- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF62A6
- #define STM8AF62Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF62A8
- #define STM8AF62Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005

- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define STM8AF62A9`
- `#define STM8AF62Ax`
- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`

- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF62AA
- #define STM8AF62Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8AF6366
- #define STM8AF636x
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A

- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [STM8AF6388](#)
- #define [STM8AF638x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 64\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [STM8\\_PFLASH\\_SIZE](#) 2\*1024

- size of program flash [B]*
- #define `STM8_RAM_SIZE` 1\*1024
- size of RAM [B]*
- #define `STM8_EEPROM_SIZE` 128
- size of data EEPROM [B]*
- #define `STM8_PFLASH_START` 0x8000
- first address in program flash*
- #define `STM8_PFLASH_END` (`STM8_PFLASH_START` + `STM8_PFLASH_SIZE` - 1)
- last address in program flash*
- #define `STM8_RAM_START` 0x0000
- first address in RAM*
- #define `STM8_RAM_END` (`STM8_RAM_START` + `STM8_RAM_SIZE` - 1)
- last address in RAM*
- #define `STM8_EEPROM_START` 0x4000
- first address in EEPROM*
- #define `STM8_EEPROM_END` (`STM8_EEPROM_START` + `STM8_EEPROM_SIZE` - 1)
- last address in EEPROM*
- #define `STM8_ADDR_WIDTH` 16
- width of address space*
- #define `STM8_MEM_POINTER_T` uint16\_t
- address variable type*
- #define `ISR_HANDLER`(func, irq) void func(void) \_\_interrupt(irq)
- handler for interrupt service routine*
- #define `ISR_HANDLER_TRAP`(func) void func() \_\_trap
- handler for trap service routine*
- #define `NOP`() \_\_asm\_\_("nop")
- perform a nop() operation (=minimum delay)*
- #define `DISABLE_INTERRUPTS`() \_\_asm\_\_("sim")
- disable interrupt handling*
- #define `ENABLE_INTERRUPTS`() \_\_asm\_\_("rim")
- enable interrupt handling*
- #define `TRIGGER_TRAP` \_\_asm\_\_("trap")
- trigger a trap (=soft interrupt) e.g. for EMC robustness (see AN1015)*
- #define `WAIT_FOR_INTERRUPT`() \_\_asm\_\_("wfi")
- stop code execution and wait for interrupt*
- #define `ENTER_HALT`() \_\_asm\_\_("halt")
- put controller to HALT mode*
- #define `SW_RESET`() (`_WWDG_CR`=0xBF)
- reset controller via WWDG module*
- #define `_BITS` unsigned int
- data type in bit structs (follow C90 standard)*
- #define `_SFR`(type, addr) (\*(volatile type\*) (addr))
- peripheral register*
- #define `__TLI_VECTOR__` 0
- irq0 - External Top Level interrupt (TLI) for pin PD7*
- #define `__AWU_VECTOR__` 1
- irq1 - Auto Wake Up from Halt interrupt (AWU)*
- #define `__CLK_VECTOR__` 2
- #define `__PORTA_VECTOR__` 3
- #define `__PORTB_VECTOR__` 4
- #define `__PORTC_VECTOR__` 5

- #define `__PORTD_VECTOR__` 6
- #define `__PORTE_VECTOR__` 7
- #define `__CAN_RX_VECTOR__` 8
  - irq8 - CAN receive interrupt (shared with `__PORTF_VECTOR__`)*
- #define `__PORTF_VECTOR__` 8
  - irq8 - External interrupt 5 (GPIOF, shared with `__CAN_RX_VECTOR__`)*
- #define `__CAN_TX_VECTOR__` 9
  - irq9 - CAN transmit interrupt*
- #define `__SPI_VECTOR__` 10
- #define `__TIM1_UPD_OVF_VECTOR__` 11
- #define `__TIM1_CAPCOM_VECTOR__` 12
- #define `__TIM2_UPD_OVF_VECTOR__` 13
  - irq13 - TIM2 Update/overflow interrupt (shared with `__TIM5_UPD_OVF_VECTOR__`)*
- #define `__TIM5_UPD_OVF_VECTOR__` 13
  - irq13 - TIM5 Update/overflow interrupt (shared with `__TIM2_UPD_OVF_VECTOR__`)*
- #define `__TIM2_CAPCOM_VECTOR__` 14
  - irq14 - TIM2 Capture/Compare interrupt (shared with `__TIM5_CAPCOM_VECTOR__`)*
- #define `__TIM3_UPD_OVF_VECTOR__` 15
  - irq15 - TIM3 Update/overflow interrupt*
- #define `__TIM3_CAPCOM_VECTOR__` 16
  - irq16 - TIM3 Capture/Compare interrupt*
- #define `__UART1_TXE_VECTOR__` 17
  - irq17 - USART/UART1 send (TX empty) interrupt*
- #define `__UART1_RXF_VECTOR__` 18
  - irq18 - USART/UART1 receive (RX full) interrupt*
- #define `__I2C_VECTOR__` 19
  - irq19 - I2C interrupt*
- #define `__UART2_TXE_VECTOR__` 20
  - irq20 - UART2 send (TX empty) interrupt (shared with `__UART3_TXE_VECTOR__` and `__UART4_TXE_VECTOR__`)*
- #define `__UART2_RXF_VECTOR__` 21
  - irq21 - UART2 receive (RX full) interrupt (shared with `__UART3_RXF_VECTOR__` and `__UART4_RXF_VECTOR__`)*
- #define `__ADC1_VECTOR__` 22
  - irq22 - ADC1 end of conversion (shared with `__ADC2_VECTOR__`)*
- #define `__TIM4_UPD_OVF_VECTOR__` 23
  - irq23 - TIM4 Update/Overflow interrupt (shared with `__TIM6_UPD_OVF_VECTOR__`)*
- #define `__FLASH_VECTOR__` 24
- #define `_GPIOA_SFR(PORT_t, PORTA_AddressBase)`
  - port A struct/bit access*
- #define `_GPIOA_ODR_SFR(uint8_t, PORTA_AddressBase+0x00)`
  - port A output register*
- #define `_GPIOA_IDR_SFR(uint8_t, PORTA_AddressBase+0x01)`
  - port A input register*
- #define `_GPIOA_DDR_SFR(uint8_t, PORTA_AddressBase+0x02)`
  - port A direction register*
- #define `_GPIOA_CR1_SFR(uint8_t, PORTA_AddressBase+0x03)`
  - port A control register 1*
- #define `_GPIOA_CR2_SFR(uint8_t, PORTA_AddressBase+0x04)`
  - port A control register 2*
- #define `_GPIOB_SFR(PORT_t, PORTB_AddressBase)`



```

    port B struct/bit access
    • #define _GPIOB_ODR_SFR(uint8_t, PORTB_AddressBase+0x00)

    port B output register
    • #define _GPIOB_IDR_SFR(uint8_t, PORTB_AddressBase+0x01)

    port B input register
    • #define _GPIOB_DDR_SFR(uint8_t, PORTB_AddressBase+0x02)

    port B direction register
    • #define _GPIOB_CR1_SFR(uint8_t, PORTB_AddressBase+0x03)

    port B control register 1
    • #define _GPIOB_CR2_SFR(uint8_t, PORTB_AddressBase+0x04)

    port B control register 2
    • #define _GPIOC_SFR(PORT_t, PORTC_AddressBase)

    port C struct/bit access
    • #define _GPIOC_ODR_SFR(uint8_t, PORTC_AddressBase+0x00)

    port C output register
    • #define _GPIOC_IDR_SFR(uint8_t, PORTC_AddressBase+0x01)

    port C input register
    • #define _GPIOC_DDR_SFR(uint8_t, PORTC_AddressBase+0x02)

    port C direction register
    • #define _GPIOC_CR1_SFR(uint8_t, PORTC_AddressBase+0x03)

    port C control register 1
    • #define _GPIOC_CR2_SFR(uint8_t, PORTC_AddressBase+0x04)

    port C control register 2
    • #define _GPIOD_SFR(PORT_t, PORTD_AddressBase)

    port D struct/bit access
    • #define _GPIOD_ODR_SFR(uint8_t, PORTD_AddressBase+0x00)

    port D output register
    • #define _GPIOD_IDR_SFR(uint8_t, PORTD_AddressBase+0x01)

    port D input register
    • #define _GPIOD_DDR_SFR(uint8_t, PORTD_AddressBase+0x02)

    port D direction register
    • #define _GPIOD_CR1_SFR(uint8_t, PORTD_AddressBase+0x03)

    port D control register 1
    • #define _GPIOD_CR2_SFR(uint8_t, PORTD_AddressBase+0x04)

    port D control register 2
    • #define _GPIOE_SFR(PORT_t, PORTE_AddressBase)

    port E struct/bit access
    • #define _GPIOE_ODR_SFR(uint8_t, PORTE_AddressBase+0x00)

    port E output register
    • #define _GPIOE_IDR_SFR(uint8_t, PORTE_AddressBase+0x01)

    port E input register
    • #define _GPIOE_DDR_SFR(uint8_t, PORTE_AddressBase+0x02)

    port E direction register
    • #define _GPIOE_CR1_SFR(uint8_t, PORTE_AddressBase+0x03)

    port E control register 1
    • #define _GPIOE_CR2_SFR(uint8_t, PORTE_AddressBase+0x04)

    port E control register 2
    • #define _GPIOF_SFR(PORT_t, PORTF_AddressBase)

    port F struct/bit access
    • #define _GPIOF_ODR_SFR(uint8_t, PORTF_AddressBase+0x00)

    port F output register

```

- `#define _GPIOF_IDR_SFR(uint8_t, PORTF_AddressBase+0x01)`  
*port F input register*
- `#define _GPIOF_DDR_SFR(uint8_t, PORTF_AddressBase+0x02)`  
*port F direction register*
- `#define _GPIOF_CR1_SFR(uint8_t, PORTF_AddressBase+0x03)`  
*port F control register 1*
- `#define _GPIOF_CR2_SFR(uint8_t, PORTF_AddressBase+0x04)`  
*port F control register 2*
- `#define _GPIOG_SFR(PORT_t, PORTG_AddressBase)`  
*port G struct/bit access*
- `#define _GPIOG_ODR_SFR(uint8_t, PORTG_AddressBase+0x00)`  
*port G output register*
- `#define _GPIOG_IDR_SFR(uint8_t, PORTG_AddressBase+0x01)`  
*port G input register*
- `#define _GPIOG_DDR_SFR(uint8_t, PORTG_AddressBase+0x02)`  
*port G direction register*
- `#define _GPIOG_CR1_SFR(uint8_t, PORTG_AddressBase+0x03)`  
*port G control register 1*
- `#define _GPIOG_CR2_SFR(uint8_t, PORTG_AddressBase+0x04)`  
*port G control register 2*
- `#define _GPIOH_SFR(PORT_t, PORTH_AddressBase)`  
*port H struct/bit access*
- `#define _GPIOH_ODR_SFR(uint8_t, PORTH_AddressBase+0x00)`  
*port H output register*
- `#define _GPIOH_IDR_SFR(uint8_t, PORTH_AddressBase+0x01)`  
*port H input register*
- `#define _GPIOH_DDR_SFR(uint8_t, PORTH_AddressBase+0x02)`  
*port H direction register*
- `#define _GPIOH_CR1_SFR(uint8_t, PORTH_AddressBase+0x03)`  
*port H control register 1*
- `#define _GPIOH_CR2_SFR(uint8_t, PORTH_AddressBase+0x04)`  
*port H control register 2*
- `#define _GPIOI_SFR(PORT_t, PORTI_AddressBase)`  
*port I struct/bit access*
- `#define _GPIOI_ODR_SFR(uint8_t, PORTI_AddressBase+0x00)`  
*port I output register*
- `#define _GPIOI_IDR_SFR(uint8_t, PORTI_AddressBase+0x01)`  
*port I input register*
- `#define _GPIOI_DDR_SFR(uint8_t, PORTI_AddressBase+0x02)`  
*port I direction register*
- `#define _GPIOI_CR1_SFR(uint8_t, PORTI_AddressBase+0x03)`  
*port I control register 1*
- `#define _GPIOI_CR2_SFR(uint8_t, PORTI_AddressBase+0x04)`  
*port I control register 2*
- `#define _GPIO_ODR_RESET_VALUE ((uint8_t) 0x00)`  
*port output register reset value*
- `#define _GPIO_DDR_RESET_VALUE ((uint8_t) 0x00)`  
*port direction register reset value*
- `#define _GPIO_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*port control register 1 reset value*
- `#define _GPIO_CR2_RESET_VALUE ((uint8_t) 0x00)`

```

    port control register 2 reset value
• #define _GPIO_PIN0 ((uint8_t) (0x01 << 0))
    port bit mask for pin 0 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _GPIO_PIN1 ((uint8_t) (0x01 << 1))
    port bit mask for pin 1 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _GPIO_PIN2 ((uint8_t) (0x01 << 2))
    port bit mask for pin 2 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _GPIO_PIN3 ((uint8_t) (0x01 << 3))
    port bit mask for pin 3 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _GPIO_PIN4 ((uint8_t) (0x01 << 4))
    port bit mask for pin 4 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _GPIO_PIN5 ((uint8_t) (0x01 << 5))
    port bit mask for pin 5 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _GPIO_PIN6 ((uint8_t) (0x01 << 6))
    port bit mask for pin 6 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _GPIO_PIN7 ((uint8_t) (0x01 << 7))
    port bit mask for pin 7 (in _GPIOI_ODR, _GPIOI_IDR, _GPIOI_DDR, _GPIOI_CR1, _GPIOI_CR2)
• #define _FLASH_SFR(FLASH_t, FLASH_AddressBase)

    Flash struct/bit access.
• #define _FLASH_CR1_SFR(uint8_t, FLASH_AddressBase+0x00)
    Flash control register 1 (FLASH_CR1)
• #define _FLASH_CR2_SFR(uint8_t, FLASH_AddressBase+0x01)
    Flash control register 2 (FLASH_CR2)
• #define _FLASH_NCR2_SFR(uint8_t, FLASH_AddressBase+0x02)
    complementary Flash control register 2 (FLASH_NCR2)
• #define _FLASH_FPR_SFR(uint8_t, FLASH_AddressBase+0x03)
    Flash protection register (FLASH_FPR)
• #define _FLASH_NFPR_SFR(uint8_t, FLASH_AddressBase+0x04)
    complementary Flash protection register (FLASH_NFPR)
• #define _FLASH_IAPSR_SFR(uint8_t, FLASH_AddressBase+0x05)
    Flash status register (FLASH_IAPSR)
• #define _FLASH_PUKR_SFR(uint8_t, FLASH_AddressBase+0x08)
    Flash program memory unprotecting key register (FLASH_PUKR)
• #define _FLASH_DUKR_SFR(uint8_t, FLASH_AddressBase+0x0A)
    Data EEPROM unprotection key register (FLASH_DUKR)
• #define _FLASH_CR1_RESET_VALUE ((uint8_t) 0x00)
    Flash control register 1 reset value.
• #define _FLASH_CR2_RESET_VALUE ((uint8_t) 0x00)
    Flash control register 2 reset value.
• #define _FLASH_NCR2_RESET_VALUE ((uint8_t) 0xFF)
    complementary Flash control register 2 reset value
• #define _FLASH_IAPSR_RESET_VALUE ((uint8_t) 0x40)
    Flash status register reset value.
• #define _FLASH_PUKR_RESET_VALUE ((uint8_t) 0x00)
    Flash program memory unprotecting key reset value.
• #define _FLASH_DUKR_RESET_VALUE ((uint8_t) 0x00)
    Data EEPROM unprotection key reset value.
• #define _FLASH_FIX ((uint8_t) (0x01 << 0))
    Fixed Byte programming time [0] (in _FLASH_CR1)
• #define _FLASH_IE ((uint8_t) (0x01 << 1))
    Flash Interrupt enable [0] (in _FLASH_CR1)

```

- `#define _FLASH_AHALT ((uint8_t) (0x01 << 2))`  
Power-down in Active-halt mode [0] (in \_FLASH\_CR1)
- `#define _FLASH_HALT ((uint8_t) (0x01 << 3))`  
Power-down in Halt mode [0] (in \_FLASH\_CR1)
- `#define _FLASH_PRG ((uint8_t) (0x01 << 0))`  
Standard block programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)
- `#define _FLASH_FPRG ((uint8_t) (0x01 << 4))`  
Fast block programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)
- `#define _FLASH_ERASE ((uint8_t) (0x01 << 5))`  
Block erasing [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)
- `#define _FLASH_WPRG ((uint8_t) (0x01 << 6))`  
Word programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)
- `#define _FLASH_OPT ((uint8_t) (0x01 << 7))`  
Write option bytes [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)
- `#define _FLASH_WPB ((uint8_t) (0x3F << 0))`  
User boot code area protection bits [5:0] (in \_FLASH\_FPR and \_FLASH\_NFPR)
- `#define _FLASH_WPB0 ((uint8_t) (0x01 << 0))`  
User boot code area protection bit [0] (in \_FLASH\_FPR and \_FLASH\_NFPR)
- `#define _FLASH_WPB1 ((uint8_t) (0x01 << 1))`  
User boot code area protection bit [1] (in \_FLASH\_FPR and \_FLASH\_NFPR)
- `#define _FLASH_WPB2 ((uint8_t) (0x01 << 2))`  
User boot code area protection bit [2] (in \_FLASH\_FPR and \_FLASH\_NFPR)
- `#define _FLASH_WPB3 ((uint8_t) (0x01 << 3))`  
User boot code area protection bit [3] (in \_FLASH\_FPR and \_FLASH\_NFPR)
- `#define _FLASH_WPB4 ((uint8_t) (0x01 << 4))`  
User boot code area protection bit [4] (in \_FLASH\_FPR and \_FLASH\_NFPR)
- `#define _FLASH_WPB5 ((uint8_t) (0x01 << 5))`  
User boot code area protection bit [5] (in \_FLASH\_FPR and \_FLASH\_NFPR)
- `#define _FLASH_WR_PG_DIS ((uint8_t) (0x01 << 0))`  
Write attempted to protected page flag [0] (in \_FLASH\_IAPSR)
- `#define _FLASH_PUL ((uint8_t) (0x01 << 1))`  
Flash Program memory unlocked flag [0] (in \_FLASH\_IAPSR)
- `#define _FLASH_EOP ((uint8_t) (0x01 << 2))`  
End of programming (write or erase operation) flag [0] (in \_FLASH\_IAPSR)
- `#define _FLASH_DUL ((uint8_t) (0x01 << 3))`  
Data EEPROM area unlocked flag [0] (in \_FLASH\_IAPSR)
- `#define _FLASH_HVOFF ((uint8_t) (0x01 << 5))`  
End of high voltage flag [0] (in \_FLASH\_IAPSR)
- `#define _EXTI_SFR(EXTI_t, EXTI_AddressBase)`  
External interrupt struct/bit access.
- `#define _EXTI_CR1_SFR(uint8_t, EXTI_AddressBase+0x00)`  
External interrupt control register 1 (EXTI\_CR1)
- `#define _EXTI_CR2_SFR(uint8_t, EXTI_AddressBase+0x01)`  
External interrupt control register 2 (EXTI\_CR2)
- `#define _EXTI_CR1_RESET_VALUE ((uint8_t) 0x00)`  
External interrupt control register 1 reset value.
- `#define _EXTI_CR2_RESET_VALUE ((uint8_t) 0x00)`  
External interrupt control register 2 reset value.
- `#define _EXTI_PAIS ((uint8_t) (0x03 << 0))`  
External interrupt sensitivity for Port A [1:0] (in \_EXTI\_CR1)
- `#define _EXTI_PAIS0 ((uint8_t) (0x01 << 0))`

- External interrupt sensitivity for Port A [0] (in \_EXTI\_CR1)*

  - #define `_EXTI_PAIS1` ((uint8\_t) (0x01 << 1))
- External interrupt sensitivity for Port A [1] (in \_EXTI\_CR1)*

  - #define `_EXTI_PBIS` ((uint8\_t) (0x03 << 2))
- External interrupt sensitivity for Port B [1:0] (in \_EXTI\_CR1)*

  - #define `_EXTI_PBIS0` ((uint8\_t) (0x01 << 2))
- External interrupt sensitivity for Port B [0] (in \_EXTI\_CR1)*

  - #define `_EXTI_PBIS1` ((uint8\_t) (0x01 << 3))
- External interrupt sensitivity for Port B [1] (in \_EXTI\_CR1)*

  - #define `_EXTI_PCIS` ((uint8\_t) (0x03 << 4))
- External interrupt sensitivity for Port C [1:0] (in \_EXTI\_CR1)*

  - #define `_EXTI_PCIS0` ((uint8\_t) (0x01 << 4))
- External interrupt sensitivity for Port C [0] (in \_EXTI\_CR1)*

  - #define `_EXTI_PCIS1` ((uint8\_t) (0x01 << 5))
- External interrupt sensitivity for Port C [1] (in \_EXTI\_CR1)*

  - #define `_EXTI_PDIS` ((uint8\_t) (0x03 << 6))
- External interrupt sensitivity for Port D [1:0] (in \_EXTI\_CR1)*

  - #define `_EXTI_PDIS0` ((uint8\_t) (0x01 << 6))
- External interrupt sensitivity for Port D [0] (in \_EXTI\_CR1)*

  - #define `_EXTI_PDIS1` ((uint8\_t) (0x01 << 7))
- External interrupt sensitivity for Port D [1] (in \_EXTI\_CR1)*

  - #define `_EXTI_PEIS` ((uint8\_t) (0x03 << 0))
- Port E external interrupt sensitivity bits [1:0] (in \_EXTI\_CR2)*

  - #define `_EXTI_PEIS0` ((uint8\_t) (0x01 << 0))
- Port E external interrupt sensitivity bits [0] (in \_EXTI\_CR2)*

  - #define `_EXTI_PEIS1` ((uint8\_t) (0x01 << 1))
- Port E external interrupt sensitivity bits [1] (in \_EXTI\_CR2)*

  - #define `_EXTI_TLIS` ((uint8\_t) (0x01 << 2))
- Top level interrupt sensitivity [0] (in \_EXTI\_CR2)*

  - #define `_RST_SFR(RST_t, RST_AddressBase)`
- Reset module struct/bit access.*

  - #define `_RST_SR_SFR`(uint8\_t, `RST_AddressBase`+0x00)
- Reset module status register (RST\_SR)*

  - #define `_RST_WWDGF` ((uint8\_t) (0x01 << 0))
- Window Watchdog reset flag [0] (in \_RST\_SR)*

  - #define `_RST_IWDGF` ((uint8\_t) (0x01 << 1))
- Independent Watchdog reset flag [0] (in \_RST\_SR)*

  - #define `_RST_ILLOPF` ((uint8\_t) (0x01 << 2))
- Illegal opcode reset flag [0] (in \_RST\_SR)*

  - #define `_RST_SWIMF` ((uint8\_t) (0x01 << 3))
- SWIM reset flag [0] (in \_RST\_SR)*

  - #define `_RST_EMCF` ((uint8\_t) (0x01 << 4))
- EMC reset flag [0] (in \_RST\_SR)*

  - #define `_CLK_SFR(CLK_t, CLK_AddressBase)`
- Clock module struct/bit access.*

  - #define `_CLK_ICKR_SFR`(uint8\_t, `CLK_AddressBase`+0x00)
- Internal clock register.*

  - #define `_CLK_ECKR_SFR`(uint8\_t, `CLK_AddressBase`+0x01)
- External clock register.*

  - #define `_CLK_CMSR_SFR`(uint8\_t, `CLK_AddressBase`+0x03)
- Clock master status register.*

- `#define _CLK_SWR_SFR(uint8_t, CLK_AddressBase+0x04)`  
*Clock master switch register.*
- `#define _CLK_SWCR_SFR(uint8_t, CLK_AddressBase+0x05)`  
*Clock switch control register.*
- `#define _CLK_CKDIVR_SFR(uint8_t, CLK_AddressBase+0x06)`  
*Clock divider register.*
- `#define _CLK_PCKENR1_SFR(uint8_t, CLK_AddressBase+0x07)`  
*Peripheral clock gating register 1.*
- `#define _CLK_CSSR_SFR(uint8_t, CLK_AddressBase+0x08)`  
*Clock security system register.*
- `#define _CLK_CCOR_SFR(uint8_t, CLK_AddressBase+0x09)`  
*Configurable clock output register.*
- `#define _CLK_PCKENR2_SFR(uint8_t, CLK_AddressBase+0x0A)`  
*Peripheral clock gating register 2.*
- `#define _CLK_HSTRIMR_SFR(uint8_t, CLK_AddressBase+0x0C)`  
*HSI clock calibration trimming register.*
- `#define _CLK_SWIMCCR_SFR(uint8_t, CLK_AddressBase+0x0D)`  
*SWIM clock control register.*
- `#define _CLK_ICKR_RESET_VALUE ((uint8_t) 0x01)`  
*Internal clock register reset value.*
- `#define _CLK_ECKR_RESET_VALUE ((uint8_t) 0x00)`  
*External clock register reset value.*
- `#define _CLK_CMSR_RESET_VALUE ((uint8_t) 0xE1)`  
*Clock master status reset value.*
- `#define _CLK_SWR_RESET_VALUE ((uint8_t) 0xE1)`  
*Clock master switch reset value.*
- `#define _CLK_SWCR_RESET_VALUE ((uint8_t) 0x00)`  
*Clock switch control reset value.*
- `#define _CLK_CKDIVR_RESET_VALUE ((uint8_t) 0x18)`  
*Clock divider register reset value.*
- `#define _CLK_PCKENR1_RESET_VALUE ((uint8_t) 0xFF)`  
*Peripheral clock gating register 1 reset value.*
- `#define _CLK_PCKENR2_RESET_VALUE ((uint8_t) 0xFF)`  
*Peripheral clock gating register 2 reset value.*
- `#define _CLK_CSSR_RESET_VALUE ((uint8_t) 0x00)`  
*Clock security system register reset value.*
- `#define _CLK_CCOR_RESET_VALUE ((uint8_t) 0x00)`  
*Configurable clock output register reset value.*
- `#define _CLK_HSTRIMR_RESET_VALUE ((uint8_t) 0x00)`  
*HSI clock calibration trimming register reset value.*
- `#define _CLK_SWIMCCR_RESET_VALUE ((uint8_t) 0x00)`  
*SWIM clock control register reset value.*
- `#define _CLK_HSIEN ((uint8_t) (0x01 << 0))`  
*High speed internal RC oscillator enable [0] (in \_CLK\_ICKR)*
- `#define _CLK_HSIRDY ((uint8_t) (0x01 << 1))`  
*High speed internal oscillator ready [0] (in \_CLK\_ICKR)*
- `#define _CLK_FHWU ((uint8_t) (0x01 << 2))`  
*Fast wakeup from Halt/Active-halt modes [0] (in \_CLK\_ICKR)*
- `#define _CLK_LSIEN ((uint8_t) (0x01 << 3))`  
*Low speed internal RC oscillator enable [0] (in \_CLK\_ICKR)*
- `#define _CLK_LSIRDY ((uint8_t) (0x01 << 4))`

- Low speed internal oscillator ready [0] (in \_CLK\_ICKR)*
- #define `_CLK_REGAH` ((uint8\_t) (0x01 << 5))
- Regulator power off in Active-halt mode [0] (in \_CLK\_ICKR)*
- #define `_CLK_HSEEN` ((uint8\_t) (0x01 << 0))
- High speed external crystal oscillator enable [0] (in \_CLK\_ECKR)*
- #define `_CLK_ECKR_HSERDY` ((uint8\_t) (0x01 << 1))
- High speed external crystal oscillator ready [0] (in \_CLK\_ECKR)*
- #define `_CLK_SWI_HSI` ((uint8\_t) 0xE1)
- write to CLK\_SWR for HSI clock (in \_CLK\_SWR)*
- #define `_CLK_SWI_LSI` ((uint8\_t) 0xD2)
- write to CLK\_SWR for LSI clock (in \_CLK\_SWR)*
- #define `_CLK_SWI_HSE` ((uint8\_t) 0xB4)
- write to CLK\_SWR for HSE clock (in \_CLK\_SWR)*
- #define `_CLK_SWBSY` ((uint8\_t) (0x01 << 0))
- Switch busy flag [0] (in \_CLK\_SWCR)*
- #define `_CLK_SWEN` ((uint8\_t) (0x01 << 1))
- Switch start/stop enable [0] (in \_CLK\_SWCR)*
- #define `_CLK_SWIEN` ((uint8\_t) (0x01 << 2))
- Clock switch interrupt enable [0] (in \_CLK\_SWCR)*
- #define `_CLK_SWIF` ((uint8\_t) (0x01 << 3))
- Clock switch interrupt flag [0] (in \_CLK\_SWCR)*
- #define `_CLK_CPUDIV` ((uint8\_t) (0x07 << 0))
- CPU clock prescaler [2:0] (in \_CLK\_CKDIVR)*
- #define `_CLK_CPUDIV0` ((uint8\_t) (0x01 << 0))
- CPU clock prescaler [0] (in \_CLK\_CKDIVR)*
- #define `_CLK_CPUDIV1` ((uint8\_t) (0x01 << 1))
- CPU clock prescaler [1] (in \_CLK\_CKDIVR)*
- #define `_CLK_CPUDIV2` ((uint8\_t) (0x01 << 2))
- CPU clock prescaler [2] (in \_CLK\_CKDIVR)*
- #define `_CLK_HSIDIV` ((uint8\_t) (0x03 << 3))
- High speed internal clock prescaler [1:0] (in \_CLK\_CKDIVR)*
- #define `_CLK_HSIDIV0` ((uint8\_t) (0x01 << 3))
- High speed internal clock prescaler [0] (in \_CLK\_CKDIVR)*
- #define `_CLK_HSIDIV1` ((uint8\_t) (0x01 << 4))
- High speed internal clock prescaler [1] (in \_CLK\_CKDIVR)*
- #define `_CLK_I2C` ((uint8\_t) (0x01 << 0))
- clock enable I2C [0] (in \_CLK\_PCKENR1)*
- #define `_CLK_SPI` ((uint8\_t) (0x01 << 1))
- clock enable SPI [0] (in \_CLK\_PCKENR1)*
- #define `_CLK_UART1` ((uint8\_t) (0x01 << 2))
- clock enable UART1 [0] (in \_CLK\_PCKENR1)*
- #define `_CLK_UART2` ((uint8\_t) (0x01 << 3))
- clock enable UART2 [0] (in \_CLK\_PCKENR1)*
- #define `_CLK_TIM4_TIM6` ((uint8\_t) (0x01 << 4))
- clock enable TIM4/TIM6 [0] (in \_CLK\_PCKENR1)*
- #define `_CLK_TIM2_TIM5` ((uint8\_t) (0x01 << 5))
- clock enable TIM2/TIM5 [0] (in \_CLK\_PCKENR1)*
- #define `_CLK_TIM3` ((uint8\_t) (0x01 << 6))
- clock enable TIM3 [0] (in \_CLK\_PCKENR1)*
- #define `_CLK_TIM1` ((uint8\_t) (0x01 << 7))
- clock enable TIM1 [0] (in \_CLK\_PCKENR1)*



- `#define _CLK_CSSEN ((uint8_t) (0x01 << 0))`  
*Clock security system enable [0] (in \_CLK\_CSSR)*
- `#define _CLK_AUX ((uint8_t) (0x01 << 1))`  
*Auxiliary oscillator connected to master clock [0] (in \_CLK\_CSSR)*
- `#define _CLK_CSSDIE ((uint8_t) (0x01 << 2))`  
*Clock security system detection interrupt enable [0] (in \_CLK\_CSSR)*
- `#define _CLK_CSSD ((uint8_t) (0x01 << 3))`  
*Clock security system detection [0] (in \_CLK\_CSSR)*
- `#define _CLK_CCOEN ((uint8_t) (0x01 << 0))`  
*Configurable clock output enable [0] (in \_CLK\_CCOR)*
- `#define _CLK_CCOSEL ((uint8_t) (0x0F << 1))`  
*Configurable clock output selection [3:0] (in \_CLK\_CCOR)*
- `#define _CLK_CCOSEL0 ((uint8_t) (0x01 << 1))`  
*Configurable clock output selection [0] (in \_CLK\_CCOR)*
- `#define _CLK_CCOSEL1 ((uint8_t) (0x01 << 2))`  
*Configurable clock output selection [1] (in \_CLK\_CCOR)*
- `#define _CLK_CCOSEL2 ((uint8_t) (0x01 << 3))`  
*Configurable clock output selection [2] (in \_CLK\_CCOR)*
- `#define _CLK_CCOSEL3 ((uint8_t) (0x01 << 4))`  
*Configurable clock output selection [3] (in \_CLK\_CCOR)*
- `#define _CLK_CCORDY ((uint8_t) (0x01 << 5))`  
*Configurable clock output ready [0] (in \_CLK\_CCOR)*
- `#define _CLK_CCOBSY ((uint8_t) (0x01 << 6))`  
*Configurable clock output busy [0] (in \_CLK\_CCOR)*
- `#define _CLK_AWU ((uint8_t) (0x01 << 2))`  
*clock enable AWU [0] (in \_CLK\_PCKENR2)*
- `#define _CLK_ADC ((uint8_t) (0x01 << 3))`  
*clock enable ADC [0] (in \_CLK\_PCKENR2)*
- `#define _CLK_CAN ((uint8_t) (0x01 << 7))`  
*clock enable CAN [0] (in \_CLK\_PCKENR2)*
- `#define _CLK_HSITRIM ((uint8_t) (0x0F << 0))`  
*HSI trimming value (some devices only support 3 bits, see DS!) [3:0] (in \_CLK\_HSITRIMR)*
- `#define _CLK_HSITRIM0 ((uint8_t) (0x01 << 0))`  
*HSI trimming value [0] (in \_CLK\_HSITRIMR)*
- `#define _CLK_HSITRIM1 ((uint8_t) (0x01 << 1))`  
*HSI trimming value [1] (in \_CLK\_HSITRIMR)*
- `#define _CLK_HSITRIM2 ((uint8_t) (0x01 << 2))`  
*HSI trimming value [2] (in \_CLK\_HSITRIMR)*
- `#define _CLK_HSITRIM3 ((uint8_t) (0x01 << 3))`  
*HSI trimming value [3] (in \_CLK\_HSITRIMR)*
- `#define _CLK_SWIMCLK ((uint8_t) (0x01 << 0))`  
*SWIM clock divider [0] (in \_CLK\_SWIMCCR)*
- `#define _WWDG_SFR(WWDG_t, WWDG_AddressBase)`  
*Window Watchdog struct/bit access.*
- `#define _WWDG_CR_SFR(uint8_t, WWDG_AddressBase+0x00)`  
*Window Watchdog Control register (WWDG\_CR)*
- `#define _WWDG_WR_SFR(uint8_t, WWDG_AddressBase+0x01)`  
*Window Watchdog Window register (WWDG\_WR)*
- `#define _WWDG_CR_RESET_VALUE ((uint8_t) 0x7F)`  
*Window Watchdog Control register reset value.*
- `#define _WWDG_WR_RESET_VALUE ((uint8_t) 0x7F)`



- Window Watchdog Window register reset value.*
  - #define `_WWDG_T` ((uint8\_t) (0x7F << 0))
    - Window Watchdog 7-bit counter [6:0] (in \_WWDG\_CR)*
  - #define `_WWDG_T0` ((uint8\_t) (0x01 << 0))
    - Window Watchdog 7-bit counter [0] (in \_WWDG\_CR)*
  - #define `_WWDG_T1` ((uint8\_t) (0x01 << 1))
    - Window Watchdog 7-bit counter [1] (in \_WWDG\_CR)*
  - #define `_WWDG_T2` ((uint8\_t) (0x01 << 2))
    - Window Watchdog 7-bit counter [2] (in \_WWDG\_CR)*
  - #define `_WWDG_T3` ((uint8\_t) (0x01 << 3))
    - Window Watchdog 7-bit counter [3] (in \_WWDG\_CR)*
  - #define `_WWDG_T4` ((uint8\_t) (0x01 << 4))
    - Window Watchdog 7-bit counter [4] (in \_WWDG\_CR)*
  - #define `_WWDG_T5` ((uint8\_t) (0x01 << 5))
    - Window Watchdog 7-bit counter [5] (in \_WWDG\_CR)*
  - #define `_WWDG_T6` ((uint8\_t) (0x01 << 6))
    - Window Watchdog 7-bit counter [6] (in \_WWDG\_CR)*
  - #define `_WWDG_WDGA` ((uint8\_t) (0x01 << 7))
    - Window Watchdog activation bit (n/a if WWDG enabled by option byte) [0] (in \_WWDG\_CR)*
  - #define `_WWDG_W` ((uint8\_t) (0x7F << 0))
    - Window Watchdog 7-bit window value [6:0] (in \_WWDG\_WR)*
  - #define `_WWDG_W0` ((uint8\_t) (0x01 << 0))
    - Window Watchdog 7-bit window value [0] (in \_WWDG\_WR)*
  - #define `_WWDG_W1` ((uint8\_t) (0x01 << 1))
    - Window Watchdog 7-bit window value [1] (in \_WWDG\_WR)*
  - #define `_WWDG_W2` ((uint8\_t) (0x01 << 2))
    - Window Watchdog 7-bit window value [2] (in \_WWDG\_WR)*
  - #define `_WWDG_W3` ((uint8\_t) (0x01 << 3))
    - Window Watchdog 7-bit window value [3] (in \_WWDG\_WR)*
  - #define `_WWDG_W4` ((uint8\_t) (0x01 << 4))
    - Window Watchdog 7-bit window value [4] (in \_WWDG\_WR)*
  - #define `_WWDG_W5` ((uint8\_t) (0x01 << 5))
    - Window Watchdog 7-bit window value [5] (in \_WWDG\_WR)*
  - #define `_WWDG_W6` ((uint8\_t) (0x01 << 6))
    - Window Watchdog 7-bit window value [6] (in \_WWDG\_WR)*
  - #define `_IWDG_SFR(IWDG_t, IWDG_AddressBase)`
    - Independent Timeout Watchdog struct/bit access.*
  - #define `_IWDG_KR_SFR`(uint8\_t, `IWDG_AddressBase`+0x00)
    - Independent Timeout Watchdog Key register (IWDG\_KR)*
  - #define `_IWDG_PR_SFR`(uint8\_t, `IWDG_AddressBase`+0x01)
    - Independent Timeout Watchdog Prescaler register (IWDG\_PR)*
  - #define `_IWDG_RLR_SFR`(uint8\_t, `IWDG_AddressBase`+0x02)
    - Independent Timeout Watchdog Reload register (IWDG\_RLR)*
  - #define `_IWDG_PR_RESET_VALUE` ((uint8\_t) 0x00)
    - Independent Timeout Watchdog Prescaler register reset value.*
  - #define `_IWDG_RLR_RESET_VALUE` ((uint8\_t) 0xFF)
    - Independent Timeout Watchdog Reload register reset value.*
  - #define `_IWDG_KEY_ENABLE` ((uint8\_t) 0xCC)
    - Independent Timeout Watchdog enable (in \_IWDG\_KR)*
  - #define `_IWDG_KEY_REFRESH` ((uint8\_t) 0xAA)
    - Independent Timeout Watchdog refresh (in \_IWDG\_KR)*

- `#define _IWDG_KEY_ACCESS ((uint8_t) 0x55)`  
*Independent Timeout Watchdog unlock write to IWDG\_PR and IWDG\_RLR (in \_IWDG\_KR)*
- `#define _IWDG_PRE ((uint8_t) (0x07 << 0))`  
*Independent Timeout Watchdog Prescaler divider [2:0] (in \_IWDG\_PR)*
- `#define _IWDG_PRE0 ((uint8_t) (0x01 << 0))`  
*Independent Timeout Watchdog Prescaler divider [0] (in \_IWDG\_PR)*
- `#define _IWDG_PRE1 ((uint8_t) (0x01 << 1))`  
*Independent Timeout Watchdog Prescaler divider [1] (in \_IWDG\_PR)*
- `#define _IWDG_PRE2 ((uint8_t) (0x01 << 2))`  
*Independent Timeout Watchdog Prescaler divider [2] (in \_IWDG\_PR)*
- `#define _AWU_SFR(AWU_t, AWU_AddressBase)`  
*Auto Wake-Up struct/bit access.*
- `#define _AWU_CSR_SFR(uint8_t, AWU_AddressBase+0x00)`  
*Auto Wake-Up Control/status register (AWU\_CSR)*
- `#define _AWU_APR_SFR(uint8_t, AWU_AddressBase+0x01)`  
*Auto Wake-Up Asynchronous prescaler register (AWU\_APR)*
- `#define _AWU_TBR_SFR(uint8_t, AWU_AddressBase+0x02)`  
*Auto Wake-Up Timebase selection register (AWU\_TBR)*
- `#define _AWU_CSR_RESET_VALUE ((uint8_t) 0x00)`  
*Auto Wake-Up Control/status register reset value.*
- `#define _AWU_APR_RESET_VALUE ((uint8_t) 0x3F)`  
*Auto Wake-Up Asynchronous prescaler register reset value.*
- `#define _AWU_TBR_RESET_VALUE ((uint8_t) 0x00)`  
*Auto Wake-Up Timebase selection register reset value.*
- `#define _AWU_MSR ((uint8_t) (0x01 << 0))`  
*Auto Wake-Up LSI measurement enable [0] (in \_AWU\_CSR)*
- `#define _AWU_AWUEN ((uint8_t) (0x01 << 4))`  
*Auto-wakeup enable [0] (in \_AWU\_CSR)*
- `#define _AWU_AWUF ((uint8_t) (0x01 << 5))`  
*Auto-wakeup status flag [0] (in \_AWU\_CSR)*
- `#define _AWU_APRE ((uint8_t) (0x3F << 0))`  
*Auto-wakeup asynchronous prescaler divider [5:0] (in \_AWU\_APR)*
- `#define _AWU_APRE0 ((uint8_t) (0x01 << 0))`  
*Auto-wakeup asynchronous prescaler divider [0] (in \_AWU\_APR)*
- `#define _AWU_APRE1 ((uint8_t) (0x01 << 1))`  
*Auto-wakeup asynchronous prescaler divider [1] (in \_AWU\_APR)*
- `#define _AWU_APRE2 ((uint8_t) (0x01 << 2))`  
*Auto-wakeup asynchronous prescaler divider [2] (in \_AWU\_APR)*
- `#define _AWU_APRE3 ((uint8_t) (0x01 << 3))`  
*Auto-wakeup asynchronous prescaler divider [3] (in \_AWU\_APR)*
- `#define _AWU_APRE4 ((uint8_t) (0x01 << 4))`  
*Auto-wakeup asynchronous prescaler divider [4] (in \_AWU\_APR)*
- `#define _AWU_APRE5 ((uint8_t) (0x01 << 5))`  
*Auto-wakeup asynchronous prescaler divider [5] (in \_AWU\_APR)*
- `#define _AWU_AWUTB ((uint8_t) (0x0F << 0))`  
*Auto-wakeup timebase selection [3:0] (in \_AWU\_APR)*
- `#define _AWU_AWUTB0 ((uint8_t) (0x01 << 0))`  
*Auto-wakeup timebase selection [0] (in \_AWU\_APR)*
- `#define _AWU_AWUTB1 ((uint8_t) (0x01 << 1))`  
*Auto-wakeup timebase selection [1] (in \_AWU\_APR)*
- `#define _AWU_AWUTB2 ((uint8_t) (0x01 << 2))`

- Auto-wakeup timebase selection [2] (in \_AWU\_APR)*
  - #define `_AWU_AWUTB3` ((uint8\_t) (0x01 << 3))
  - Auto-wakeup timebase selection [3] (in \_AWU\_APR)*
  - #define `_BEEP_SFR`(BEEP\_t, BEEP\_AddressBase)
  - Beeper struct/bit access.*
  - #define `_BEEP_CSR_SFR`(uint8\_t, BEEP\_AddressBase+0x00)
  - Beeper control/status register (BEEP\_CSR)*
  - #define `_BEEP_CSR_RESET_VALUE` ((uint8\_t) 0x1F)
  - Beeper control/status register reset value.*
  - #define `_BEEP_BEEP_DIV` ((uint8\_t) (0x1F << 0))
  - Beeper clock prescaler divider [4:0] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_DIV0` ((uint8\_t) (0x01 << 0))
  - Beeper clock prescaler divider [0] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_DIV1` ((uint8\_t) (0x01 << 1))
  - Beeper clock prescaler divider [1] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_DIV2` ((uint8\_t) (0x01 << 2))
  - Beeper clock prescaler divider [2] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_DIV3` ((uint8\_t) (0x01 << 3))
  - Beeper clock prescaler divider [3] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_DIV4` ((uint8\_t) (0x01 << 4))
  - Beeper clock prescaler divider [4] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_EN` ((uint8\_t) (0x01 << 5))
  - Beeper enable [0] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_SEL` ((uint8\_t) (0x03 << 6))
  - Beeper frequency selection [1:0] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_SEL0` ((uint8\_t) (0x01 << 6))
  - Beeper frequency selection [0] (in \_BEEP\_CSR)*
  - #define `_BEEP_BEEP_SEL1` ((uint8\_t) (0x01 << 7))
  - Beeper frequency selection [1] (in \_BEEP\_CSR)*
  - #define `_SPI_SFR`(SPI\_t, SPI\_AddressBase)
  - register for SPI control*
  - #define `_SPI_CR1_SFR`(uint8\_t, SPI\_AddressBase+0x00)
  - SPI control register 1.*
  - #define `_SPI_CR2_SFR`(uint8\_t, SPI\_AddressBase+0x01)
  - SPI control register 2.*
  - #define `_SPI_ICR_SFR`(uint8\_t, SPI\_AddressBase+0x02)
  - SPI interrupt control register.*
  - #define `_SPI_SR_SFR`(uint8\_t, SPI\_AddressBase+0x03)
  - SPI status register.*
  - #define `_SPI_DR_SFR`(uint8\_t, SPI\_AddressBase+0x04)
  - SPI data register.*
  - #define `_SPI_CRCPR_SFR`(uint8\_t, SPI\_AddressBase+0x05)
  - SPI CRC polynomial register.*
  - #define `_SPI_RXCRCR_SFR`(uint8\_t, SPI\_AddressBase+0x06)
  - SPI Rx CRC register.*
  - #define `_SPI_TXCRCR_SFR`(uint8\_t, SPI\_AddressBase+0x07)
  - SPI Tx CRC register.*
  - #define `_SPI_CR1_RESET_VALUE` ((uint8\_t) 0x00)
  - SPI Control Register 1 reset value.*
  - #define `_SPI_CR2_RESET_VALUE` ((uint8\_t) 0x00)
  - SPI Control Register 2 reset value.*

- `#define _SPI_ICR_RESET_VALUE ((uint8_t) 0x00)`  
*SPI Interrupt Control Register reset value.*
- `#define _SPI_SR_RESET_VALUE ((uint8_t) 0x02)`  
*SPI Status Register reset value.*
- `#define _SPI_DR_RESET_VALUE ((uint8_t) 0x00)`  
*SPI Data Register reset value.*
- `#define _SPI_CRCPR_RESET_VALUE ((uint8_t) 0x07)`  
*SPI Polynomial Register reset value.*
- `#define _SPI_RXCRCR_RESET_VALUE ((uint8_t) 0x00)`  
*SPI RX CRC Register reset value.*
- `#define _SPI_TXCRCR_RESET_VALUE ((uint8_t) 0x00)`  
*SPI TX CRC Register reset value.*
- `#define _SPI_CPHA ((uint8_t) (0x01 << 0))`  
*SPI Clock phase [0] (in \_SPI\_CR1)*
- `#define _SPI_CPOL ((uint8_t) (0x01 << 1))`  
*SPI Clock polarity [0] (in \_SPI\_CR1)*
- `#define _SPI_MSTR ((uint8_t) (0x01 << 2))`  
*SPI Master/slave selection [0] (in \_SPI\_CR1)*
- `#define _SPI_BR ((uint8_t) (0x07 << 3))`  
*SPI Baudrate control [2:0] (in \_SPI\_CR1)*
- `#define _SPI_BR0 ((uint8_t) (0x01 << 3))`  
*SPI Baudrate control [0] (in \_SPI\_CR1)*
- `#define _SPI_BR1 ((uint8_t) (0x01 << 4))`  
*SPI Baudrate control [1] (in \_SPI\_CR1)*
- `#define _SPI_BR2 ((uint8_t) (0x01 << 5))`  
*SPI Baudrate control [2] (in \_SPI\_CR1)*
- `#define _SPI_SPE ((uint8_t) (0x01 << 6))`  
*SPI enable [0] (in \_SPI\_CR1)*
- `#define _SPI_LSBFIRST ((uint8_t) (0x01 << 7))`  
*SPI Frame format [0] (in \_SPI\_CR1)*
- `#define _SPI_SSI ((uint8_t) (0x01 << 0))`  
*SPI Internal slave select [0] (in \_SPI\_CR2)*
- `#define _SPI_SSM ((uint8_t) (0x01 << 1))`  
*SPI Software slave management [0] (in \_SPI\_CR2)*
- `#define _SPI_RXONLY ((uint8_t) (0x01 << 2))`  
*SPI Receive only [0] (in \_SPI\_CR2)*
- `#define _SPI_CRCNEXT ((uint8_t) (0x01 << 4))`  
*SPI Transmit CRC next [0] (in \_SPI\_CR2)*
- `#define _SPI_CRCEN ((uint8_t) (0x01 << 5))`  
*SPI Hardware CRC calculation enable [0] (in \_SPI\_CR2)*
- `#define _SPI_BDOE ((uint8_t) (0x01 << 6))`  
*SPI Input/Output enable in bidirectional mode [0] (in \_SPI\_CR2)*
- `#define _SPI_BDM ((uint8_t) (0x01 << 7))`  
*SPI Bidirectional data mode enable [0] (in \_SPI\_CR2)*
- `#define _SPI_WKIE ((uint8_t) (0x01 << 4))`  
*SPI Wakeup interrupt enable [0] (in \_SPI\_ICR)*
- `#define _SPI_ERRIE ((uint8_t) (0x01 << 5))`  
*SPI Error interrupt enable [0] (in \_SPI\_ICR)*
- `#define _SPI_RXIE ((uint8_t) (0x01 << 6))`  
*SPI Rx buffer not empty interrupt enable [0] (in \_SPI\_ICR)*
- `#define _SPI_TXIE ((uint8_t) (0x01 << 7))`

- SPI Tx buffer empty interrupt enable [0] (in \_SPI\_ICR)*
  - #define `_SPI_RXNE` ((uint8\_t) (0x01 << 0))
- SPI Receive buffer not empty [0] (in \_SPI\_SR)*
  - #define `_SPI_TXE` ((uint8\_t) (0x01 << 1))
- SPI Transmit buffer empty [0] (in \_SPI\_SR)*
  - #define `_SPI_WKUP` ((uint8\_t) (0x01 << 3))
- SPI Wakeup flag [0] (in \_SPI\_SR)*
  - #define `_SPI_CRCERR` ((uint8\_t) (0x01 << 4))
- SPI CRC error flag [0] (in \_SPI\_SR)*
  - #define `_SPI_MODF` ((uint8\_t) (0x01 << 5))
- SPI Mode fault [0] (in \_SPI\_SR)*
  - #define `_SPI_OVR` ((uint8\_t) (0x01 << 6))
- SPI Overrun flag [0] (in \_SPI\_SR)*
  - #define `_SPI_BSY` ((uint8\_t) (0x01 << 7))
- SPI Busy flag [0] (in \_SPI\_SR)*
  - #define `_I2C_SFR(I2C_t, I2C_AddressBase)`
- register for SPI control*
  - #define `_I2C_CR1_SFR`(uint8\_t, I2C\_AddressBase+0x00)
- I2C Control register 1.*
  - #define `_I2C_CR2_SFR`(uint8\_t, I2C\_AddressBase+0x01)
- I2C Control register 2.*
  - #define `_I2C_FREQR_SFR`(uint8\_t, I2C\_AddressBase+0x02)
- I2C Frequency register.*
  - #define `_I2C_OARL_SFR`(uint8\_t, I2C\_AddressBase+0x03)
- I2C own address register low byte.*
  - #define `_I2C_OARH_SFR`(uint8\_t, I2C\_AddressBase+0x04)
- I2C own address register high byte.*
  - #define `_I2C_DR_SFR`(uint8\_t, I2C\_AddressBase+0x06)
- I2C data register.*
  - #define `_I2C_SR1_SFR`(uint8\_t, I2C\_AddressBase+0x07)
- I2C Status register 1.*
  - #define `_I2C_SR2_SFR`(uint8\_t, I2C\_AddressBase+0x08)
- I2C Status register 2.*
  - #define `_I2C_SR3_SFR`(uint8\_t, I2C\_AddressBase+0x09)
- I2C Status register 3.*
  - #define `_I2C_ITR_SFR`(uint8\_t, I2C\_AddressBase+0x0A)
- I2C Interrupt register.*
  - #define `_I2C_CCRL_SFR`(uint8\_t, I2C\_AddressBase+0x0B)
- I2C Clock control register low byte.*
  - #define `_I2C_CCRH_SFR`(uint8\_t, I2C\_AddressBase+0x0C)
- I2C Clock control register high byte.*
  - #define `_I2C_TRISER_SFR`(uint8\_t, I2C\_AddressBase+0x0D)
- I2C rise time register.*
  - #define `_I2C_CR1_RESET_VALUE` ((uint8\_t) 0x00)
- I2C Control register 1 reset value.*
  - #define `_I2C_CR2_RESET_VALUE` ((uint8\_t) 0x00)
- I2C Control register 2 reset value.*
  - #define `_I2C_FREQR_RESET_VALUE` ((uint8\_t) 0x00)
- I2C Frequency register reset value.*
  - #define `_I2C_OARL_RESET_VALUE` ((uint8\_t) 0x00)
- I2C own address register low byte reset value.*

- `#define _I2C_OARH_RESET_VALUE ((uint8_t) 0x00)`  
*I2C own address register high byte reset value.*
- `#define _I2C_DR_RESET_VALUE ((uint8_t) 0x00)`  
*I2C data register reset value.*
- `#define _I2C_SR1_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Status register 1 reset value.*
- `#define _I2C_SR2_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Status register 2 reset value.*
- `#define _I2C_SR3_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Status register 3 reset value.*
- `#define _I2C_ITR_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Interrupt register reset value.*
- `#define _I2C_CCRL_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Clock control register low byte reset value.*
- `#define _I2C_CCRH_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Clock control register high byte reset value.*
- `#define _I2C_TRISER_RESET_VALUE ((uint8_t) 0x02)`  
*I2C rise time register reset value.*
- `#define _I2C_PE ((uint8_t) (0x01 << 0))`  
*I2C Peripheral enable [0] (in \_I2C\_CR1)*
- `#define _I2C_ENGC ((uint8_t) (0x01 << 6))`  
*I2C General call enable [0] (in \_I2C\_CR1)*
- `#define _I2C_NOSTRETCH ((uint8_t) (0x01 << 7))`  
*I2C Clock stretching disable (Slave mode) [0] (in \_I2C\_CR1)*
- `#define _I2C_START ((uint8_t) (0x01 << 0))`  
*I2C Start generation [0] (in \_I2C\_CR2)*
- `#define _I2C_STOP ((uint8_t) (0x01 << 1))`  
*I2C Stop generation [0] (in \_I2C\_CR2)*
- `#define _I2C_ACK ((uint8_t) (0x01 << 2))`  
*I2C Acknowledge enable [0] (in \_I2C\_CR2)*
- `#define _I2C_POS ((uint8_t) (0x01 << 3))`  
*I2C Acknowledge position (for data reception) [0] (in \_I2C\_CR2)*
- `#define _I2C_SWRST ((uint8_t) (0x01 << 7))`  
*I2C Software reset [0] (in \_I2C\_CR2)*
- `#define _I2C_FREQ ((uint8_t) (0x3F << 0))`  
*I2C Peripheral clock frequency [5:0] (in \_I2C\_FREQR)*
- `#define _I2C_FREQ0 ((uint8_t) (0x01 << 0))`  
*I2C Peripheral clock frequency [0] (in \_I2C\_FREQR)*
- `#define _I2C_FREQ1 ((uint8_t) (0x01 << 1))`  
*I2C Peripheral clock frequency [1] (in \_I2C\_FREQR)*
- `#define _I2C_FREQ2 ((uint8_t) (0x01 << 2))`  
*I2C Peripheral clock frequency [2] (in \_I2C\_FREQR)*
- `#define _I2C_FREQ3 ((uint8_t) (0x01 << 3))`  
*I2C Peripheral clock frequency [3] (in \_I2C\_FREQR)*
- `#define _I2C_FREQ4 ((uint8_t) (0x01 << 4))`  
*I2C Peripheral clock frequency [4] (in \_I2C\_FREQR)*
- `#define _I2C_FREQ5 ((uint8_t) (0x01 << 5))`  
*I2C Peripheral clock frequency [5] (in \_I2C\_FREQR)*
- `#define _I2C_ADD0 ((uint8_t) (0x01 << 0))`  
*I2C Interface address [0] (in 10-bit address mode) (in \_I2C\_OARL)*
- `#define _I2C_ADD1 ((uint8_t) (0x01 << 1))`

- I2C Interface address [1] (in \_I2C\_OARL)*
- #define `_I2C_ADD2` ((uint8\_t) (0x01 << 2))
- I2C Interface address [2] (in \_I2C\_OARL)*
- #define `_I2C_ADD3` ((uint8\_t) (0x01 << 3))
- I2C Interface address [3] (in \_I2C\_OARL)*
- #define `_I2C_ADD4` ((uint8\_t) (0x01 << 4))
- I2C Interface address [4] (in \_I2C\_OARL)*
- #define `_I2C_ADD5` ((uint8\_t) (0x01 << 5))
- I2C Interface address [5] (in \_I2C\_OARL)*
- #define `_I2C_ADD6` ((uint8\_t) (0x01 << 6))
- I2C Interface address [6] (in \_I2C\_OARL)*
- #define `_I2C_ADD7` ((uint8\_t) (0x01 << 7))
- I2C Interface address [7] (in \_I2C\_OARL)*
- #define `_I2C_ADD_8_9` ((uint8\_t) (0x03 << 1))
- I2C Interface address [9:8] (in 10-bit address mode) (in \_I2C\_OARH)*
- #define `_I2C_ADD8` ((uint8\_t) (0x01 << 1))
- I2C Interface address [8] (in \_I2C\_OARH)*
- #define `_I2C_ADD9` ((uint8\_t) (0x01 << 2))
- I2C Interface address [9] (in \_I2C\_OARH)*
- #define `_I2C_ADDCONF` ((uint8\_t) (0x01 << 6))
- I2C Address mode configuration [0] (in \_I2C\_OARH)*
- #define `_I2C_ADDMODE` ((uint8\_t) (0x01 << 7))
- I2C 7-/10-bit addressing mode (Slave mode) [0] (in \_I2C\_OARH)*
- #define `_I2C_SB` ((uint8\_t) (0x01 << 0))
- I2C Start bit (Mastermode) [0] (in \_I2C\_SR1)*
- #define `_I2C_ADDR` ((uint8\_t) (0x01 << 1))
- I2C Address sent (master mode) / matched (slave mode) [0] (in \_I2C\_SR1)*
- #define `_I2C_BTf` ((uint8\_t) (0x01 << 2))
- I2C Byte transfer finished [0] (in \_I2C\_SR1)*
- #define `_I2C_ADD10` ((uint8\_t) (0x01 << 3))
- I2C 10-bit header sent (Master mode) [0] (in \_I2C\_SR1)*
- #define `_I2C_STOPF` ((uint8\_t) (0x01 << 4))
- I2C Stop detection (Slave mode) [0] (in \_I2C\_SR1)*
- #define `_I2C_RXNE` ((uint8\_t) (0x01 << 6))
- I2C Data register not empty (receivers) [0] (in \_I2C\_SR1)*
- #define `_I2C_TXE` ((uint8\_t) (0x01 << 7))
- I2C Data register empty (transmitters) [0] (in \_I2C\_SR1)*
- #define `_I2C_BERR` ((uint8\_t) (0x01 << 0))
- I2C Bus error [0] (in \_I2C\_SR2)*
- #define `_I2C_ARLO` ((uint8\_t) (0x01 << 1))
- I2C Arbitration lost (master mode) [0] (in \_I2C\_SR2)*
- #define `_I2C_AF` ((uint8\_t) (0x01 << 2))
- I2C Acknowledge failure [0] (in \_I2C\_SR2)*
- #define `_I2C_OVR` ((uint8\_t) (0x01 << 3))
- I2C Overrun/underrun [0] (in \_I2C\_SR2)*
- #define `_I2C_WUFH` ((uint8\_t) (0x01 << 5))
- I2C Wakeup from Halt [0] (in \_I2C\_SR2)*
- #define `_I2C_MSL` ((uint8\_t) (0x01 << 0))
- I2C Master/Slave [0] (in \_I2C\_SR3)*
- #define `_I2C_BUSY` ((uint8\_t) (0x01 << 1))
- I2C Bus busy [0] (in \_I2C\_SR3)*



- `#define _I2C_TRA ((uint8_t) (0x01 << 2))`  
*I2C Transmitter/Receiver [0] (in \_I2C\_SR3)*
- `#define _I2C_GENCALL ((uint8_t) (0x01 << 4))`  
*I2C General call header (Slavemode) [0] (in \_I2C\_SR3)*
- `#define _I2C_ITERREN ((uint8_t) (0x01 << 0))`  
*I2C Error interrupt enable [0] (in \_I2C\_ITR)*
- `#define _I2C_ITEVTEN ((uint8_t) (0x01 << 1))`  
*I2C Event interrupt enable [0] (in \_I2C\_ITR)*
- `#define _I2C_ITBUFEN ((uint8_t) (0x01 << 2))`  
*I2C Buffer interrupt enable [0] (in \_I2C\_ITR)*
- `#define _I2C_CCR ((uint8_t) (0x0F << 0))`  
*I2C Clock control register (Master mode) [3:0] (in \_I2C\_CCRH)*
- `#define _I2C_CCR0 ((uint8_t) (0x01 << 0))`  
*I2C Clock control register (Master mode) [0] (in \_I2C\_CCRH)*
- `#define _I2C_CCR1 ((uint8_t) (0x01 << 1))`  
*I2C Clock control register (Master mode) [1] (in \_I2C\_CCRH)*
- `#define _I2C_CCR2 ((uint8_t) (0x01 << 2))`  
*I2C Clock control register (Master mode) [2] (in \_I2C\_CCRH)*
- `#define _I2C_CCR3 ((uint8_t) (0x01 << 3))`  
*I2C Clock control register (Master mode) [3] (in \_I2C\_CCRH)*
- `#define _I2C_DUTY ((uint8_t) (0x01 << 6))`  
*I2C Fast mode duty cycle [0] (in \_I2C\_CCRH)*
- `#define _I2C_FS ((uint8_t) (0x01 << 7))`  
*I2C master mode selection [0] (in \_I2C\_CCRH)*
- `#define _I2C_TRISE ((uint8_t) (0x3F << 0))`  
*I2C Maximum rise time (Master mode) [5:0] (in \_I2C\_TRISER)*
- `#define _I2C_TRISE0 ((uint8_t) (0x01 << 0))`  
*I2C Maximum rise time (Master mode) [0] (in \_I2C\_TRISER)*
- `#define _I2C_TRISE1 ((uint8_t) (0x01 << 1))`  
*I2C Maximum rise time (Master mode) [1] (in \_I2C\_TRISER)*
- `#define _I2C_TRISE2 ((uint8_t) (0x01 << 2))`  
*I2C Maximum rise time (Master mode) [2] (in \_I2C\_TRISER)*
- `#define _I2C_TRISE3 ((uint8_t) (0x01 << 3))`  
*I2C Maximum rise time (Master mode) [3] (in \_I2C\_TRISER)*
- `#define _I2C_TRISE4 ((uint8_t) (0x01 << 4))`  
*I2C Maximum rise time (Master mode) [4] (in \_I2C\_TRISER)*
- `#define _I2C_TRISE5 ((uint8_t) (0x01 << 5))`  
*I2C Maximum rise time (Master mode) [5] (in \_I2C\_TRISER)*
- `#define _UART1_SFR(UART1_t, UART1_AddressBase)`  
*UART1 struct/bit access.*
- `#define _UART1_SR_SFR(uint8_t, UART1_AddressBase+0x00)`  
*UART1 Status register.*
- `#define _UART1_DR_SFR(uint8_t, UART1_AddressBase+0x01)`  
*UART1 data register.*
- `#define _UART1_BRR1_SFR(uint8_t, UART1_AddressBase+0x02)`  
*UART1 Baud rate register 1.*
- `#define _UART1_BRR2_SFR(uint8_t, UART1_AddressBase+0x03)`  
*UART1 Baud rate register 2.*
- `#define _UART1_CR1_SFR(uint8_t, UART1_AddressBase+0x04)`  
*UART1 Control register 1.*
- `#define _UART1_CR2_SFR(uint8_t, UART1_AddressBase+0x05)`



- UART1 Control register 2.*
  - #define `_UART1_CR3_SFR`(uint8\_t, `UART1_AddressBase`+0x06)
- UART1 Control register 3.*
  - #define `_UART1_CR4_SFR`(uint8\_t, `UART1_AddressBase`+0x07)
- UART1 Control register 4.*
  - #define `_UART1_CR5_SFR`(uint8\_t, `UART1_AddressBase`+0x08)
- UART1 Control register 5.*
  - #define `_UART1_GTR_SFR`(uint8\_t, `UART1_AddressBase`+0x09)
- UART1 guard time register.*
  - #define `_UART1_PSCR_SFR`(uint8\_t, `UART1_AddressBase`+0x0A)
- UART1 prescaler register.*
  - #define `_UART1_SR_RESET_VALUE` ((uint8\_t) 0xC0)
- UART1 Status register reset value.*
  - #define `_UART1_BRR1_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 Baud rate register 1 reset value.*
  - #define `_UART1_BRR2_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 Baud rate register 2 reset value.*
  - #define `_UART1_CR1_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 Control register 1 reset value.*
  - #define `_UART1_CR2_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 Control register 2 reset value.*
  - #define `_UART1_CR3_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 Control register 3 reset value.*
  - #define `_UART1_CR4_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 Control register 4 reset value.*
  - #define `_UART1_CR5_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 Control register 5 reset value.*
  - #define `_UART1_GTR_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 guard time register reset value.*
  - #define `_UART1_PSCR_RESET_VALUE` ((uint8\_t) 0x00)
- UART1 prescaler register reset value.*
  - #define `_UART1_PE` ((uint8\_t) (0x01 << 0))
- UART1 Parity error [0] (in \_UART1\_SR)*
  - #define `_UART1_FE` ((uint8\_t) (0x01 << 1))
- UART1 Framing error [0] (in \_UART1\_SR)*
  - #define `_UART1_NF` ((uint8\_t) (0x01 << 2))
- UART1 Noise flag [0] (in \_UART1\_SR)*
  - #define `_UART1_OR_LHE` ((uint8\_t) (0x01 << 3))
- UART1 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART1\_SR)*
  - #define `_UART1_IDLE` ((uint8\_t) (0x01 << 4))
- UART1 IDLE line detected [0] (in \_UART1\_SR)*
  - #define `_UART1_RXNE` ((uint8\_t) (0x01 << 5))
- UART1 Read data register not empty [0] (in \_UART1\_SR)*
  - #define `_UART1_TC` ((uint8\_t) (0x01 << 6))
- UART1 Transmission complete [0] (in \_UART1\_SR)*
  - #define `_UART1_TXE` ((uint8\_t) (0x01 << 7))
- UART1 Transmit data register empty [0] (in \_UART1\_SR)*
  - #define `_UART1_PIEN` ((uint8\_t) (0x01 << 0))
- UART1 Parity interrupt enable [0] (in \_UART1\_CR1)*
  - #define `_UART1_PS` ((uint8\_t) (0x01 << 1))
- UART1 Parity selection [0] (in \_UART1\_CR1)*

- `#define _UART1_PCEN` ((uint8\_t) (0x01 << 2))  
*UART1 Parity control enable [0] (in \_UART1\_CR1)*
- `#define _UART1_WAKE` ((uint8\_t) (0x01 << 3))  
*UART1 Wakeup method [0] (in \_UART1\_CR1)*
- `#define _UART1_M` ((uint8\_t) (0x01 << 4))  
*UART1 word length [0] (in \_UART1\_CR1)*
- `#define _UART1_UARTD` ((uint8\_t) (0x01 << 5))  
*UART1 Disable (for low power consumption) [0] (in \_UART1\_CR1)*
- `#define _UART1_T8` ((uint8\_t) (0x01 << 6))  
*UART1 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART1\_CR1)*
- `#define _UART1_R8` ((uint8\_t) (0x01 << 7))  
*UART1 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART1\_CR1)*
- `#define _UART1_SBK` ((uint8\_t) (0x01 << 0))  
*UART1 Send break [0] (in \_UART1\_CR2)*
- `#define _UART1_RWU` ((uint8\_t) (0x01 << 1))  
*UART1 Receiver wakeup [0] (in \_UART1\_CR2)*
- `#define _UART1_REN` ((uint8\_t) (0x01 << 2))  
*UART1 Receiver enable [0] (in \_UART1\_CR2)*
- `#define _UART1_TEN` ((uint8\_t) (0x01 << 3))  
*UART1 Transmitter enable [0] (in \_UART1\_CR2)*
- `#define _UART1_ILIEN` ((uint8\_t) (0x01 << 4))  
*UART1 IDLE Line interrupt enable [0] (in \_UART1\_CR2)*
- `#define _UART1_RIEN` ((uint8\_t) (0x01 << 5))  
*UART1 Receiver interrupt enable [0] (in \_UART1\_CR2)*
- `#define _UART1_TCIEN` ((uint8\_t) (0x01 << 6))  
*UART1 Transmission complete interrupt enable [0] (in \_UART1\_CR2)*
- `#define _UART1_TIEN` ((uint8\_t) (0x01 << 7))  
*UART1 Transmitter interrupt enable [0] (in \_UART1\_CR2)*
- `#define _UART1_LBCL` ((uint8\_t) (0x01 << 0))  
*UART1 Last bit clock pulse [0] (in \_UART1\_CR3)*
- `#define _UART1_CPHA` ((uint8\_t) (0x01 << 1))  
*UART1 Clock phase [0] (in \_UART1\_CR3)*
- `#define _UART1_CPOL` ((uint8\_t) (0x01 << 2))  
*UART1 Clock polarity [0] (in \_UART1\_CR3)*
- `#define _UART1_CKEN` ((uint8\_t) (0x01 << 3))  
*UART1 Clock enable [0] (in \_UART1\_CR3)*
- `#define _UART1_STOP` ((uint8\_t) (0x03 << 4))  
*UART1 STOP bits [1:0] (in \_UART1\_CR3)*
- `#define _UART1_STOP0` ((uint8\_t) (0x01 << 4))  
*UART1 STOP bits [0] (in \_UART1\_CR3)*
- `#define _UART1_STOP1` ((uint8\_t) (0x01 << 5))  
*UART1 STOP bits [1] (in \_UART1\_CR3)*
- `#define _UART1_LINEN` ((uint8\_t) (0x01 << 6))  
*UART1 LIN mode enable [0] (in \_UART1\_CR3)*
- `#define _UART1_ADD` ((uint8\_t) (0x0F << 0))  
*UART1 Address of the UART node [3:0] (in \_UART1\_CR4)*
- `#define _UART1_ADD0` ((uint8\_t) (0x01 << 0))  
*UART1 Address of the UART node [0] (in \_UART1\_CR4)*
- `#define _UART1_ADD1` ((uint8\_t) (0x01 << 1))  
*UART1 Address of the UART node [1] (in \_UART1\_CR4)*
- `#define _UART1_ADD2` ((uint8\_t) (0x01 << 2))

- ```

UART1 Address of the UART node [2] (in _UART1_CR4)
• #define _UART1_ADD3 ((uint8_t) (0x01 << 3))
    UART1 Address of the UART node [3] (in _UART1_CR4)
• #define _UART1_LBDF ((uint8_t) (0x01 << 4))
    UART1 LIN Break Detection Flag [0] (in _UART1_CR4)
• #define _UART1_LBDL ((uint8_t) (0x01 << 5))
    UART1 LIN Break Detection Length [0] (in _UART1_CR4)
• #define _UART1_LBDIEN ((uint8_t) (0x01 << 6))
    UART1 LIN Break Detection Interrupt Enable [0] (in _UART1_CR4)
• #define _UART1_IREN ((uint8_t) (0x01 << 1))
    UART1 IrDA mode Enable [0] (in _UART1_CR5)
• #define _UART1_IRLP ((uint8_t) (0x01 << 2))
    UART1 IrDA Low Power [0] (in _UART1_CR5)
• #define _UART1_HDSEL ((uint8_t) (0x01 << 3))
    UART1 Half-Duplex Selection [0] (in _UART1_CR5)
• #define _UART1_NACK ((uint8_t) (0x01 << 4))
    UART1 Smartcard NACK enable [0] (in _UART1_CR5)
• #define _UART1_SCEN ((uint8_t) (0x01 << 5))
    UART1 Smartcard mode enable [0] (in _UART1_CR5)
• #define _UART2_SFR(UART2_t, UART2_AddressBase)
    UART2 struct/bit access.
• #define _UART2_SR_SFR(uint8_t, UART2_AddressBase+0x00)
    UART2 Status register.
• #define _UART2_DR_SFR(uint8_t, UART2_AddressBase+0x01)
    UART2 data register.
• #define _UART2_BRR1_SFR(uint8_t, UART2_AddressBase+0x02)
    UART2 Baud rate register 1.
• #define _UART2_BRR2_SFR(uint8_t, UART2_AddressBase+0x03)
    UART2 Baud rate register 2.
• #define _UART2_CR1_SFR(uint8_t, UART2_AddressBase+0x04)
    UART2 Control register 1.
• #define _UART2_CR2_SFR(uint8_t, UART2_AddressBase+0x05)
    UART2 Control register 2.
• #define _UART2_CR3_SFR(uint8_t, UART2_AddressBase+0x06)
    UART2 Control register 3.
• #define _UART2_CR4_SFR(uint8_t, UART2_AddressBase+0x07)
    UART2 Control register 4.
• #define _UART2_CR5_SFR(uint8_t, UART2_AddressBase+0x08)
    UART2 Control register 5.
• #define _UART2_CR6_SFR(uint8_t, UART2_AddressBase+0x09)
    UART2 Control register 6.
• #define _UART2_GTR_SFR(uint8_t, UART2_AddressBase+0x0A)
    UART2 guard time register.
• #define _UART2_PSCR_SFR(uint8_t, UART2_AddressBase+0x0B)
    UART2 prescaler register.
• #define _UART2_SR_RESET_VALUE ((uint8_t) 0xC0)
    UART2 Status register reset value.
• #define _UART2_BRR1_RESET_VALUE ((uint8_t) 0x00)
    UART2 Baud rate register 1 reset value.
• #define _UART2_BRR2_RESET_VALUE ((uint8_t) 0x00)
    UART2 Baud rate register 2 reset value.

```

- `#define _UART2_CR1_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 1 reset value.*
- `#define _UART2_CR2_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 2 reset value.*
- `#define _UART2_CR3_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 3 reset value.*
- `#define _UART2_CR4_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 4 reset value.*
- `#define _UART2_CR5_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 5 reset value.*
- `#define _UART2_CR6_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 6 reset value.*
- `#define _UART2_GTR_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 guard time register reset value.*
- `#define _UART2_PSCR_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 prescaler register reset value.*
- `#define _UART2_PE` ((uint8\_t) (0x01 << 0))  
*UART2 Parity error [0] (in \_UART2\_SR)*
- `#define _UART2_FE` ((uint8\_t) (0x01 << 1))  
*UART2 Framing error [0] (in \_UART2\_SR)*
- `#define _UART2_NF` ((uint8\_t) (0x01 << 2))  
*UART2 Noise flag [0] (in \_UART2\_SR)*
- `#define _UART2_OR_LHE` ((uint8\_t) (0x01 << 3))  
*UART2 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART2\_SR)*
- `#define _UART2_IDLE` ((uint8\_t) (0x01 << 4))  
*UART2 IDLE line detected [0] (in \_UART2\_SR)*
- `#define _UART2_RXNE` ((uint8\_t) (0x01 << 5))  
*UART2 Read data register not empty [0] (in \_UART2\_SR)*
- `#define _UART2_TC` ((uint8\_t) (0x01 << 6))  
*UART2 Transmission complete [0] (in \_UART2\_SR)*
- `#define _UART2_TXE` ((uint8\_t) (0x01 << 7))  
*UART2 Transmit data register empty [0] (in \_UART2\_SR)*
- `#define _UART2_PIEN` ((uint8\_t) (0x01 << 0))  
*UART2 Parity interrupt enable [0] (in \_UART2\_CR1)*
- `#define _UART2_PS` ((uint8\_t) (0x01 << 1))  
*UART2 Parity selection [0] (in \_UART2\_CR1)*
- `#define _UART2_PCEN` ((uint8\_t) (0x01 << 2))  
*UART2 Parity control enable [0] (in \_UART2\_CR1)*
- `#define _UART2_WAKE` ((uint8\_t) (0x01 << 3))  
*UART2 Wakeup method [0] (in \_UART2\_CR1)*
- `#define _UART2_M` ((uint8\_t) (0x01 << 4))  
*UART2 word length [0] (in \_UART2\_CR1)*
- `#define _UART2_UARTD` ((uint8\_t) (0x01 << 5))  
*UART2 Disable (for low power consumption) [0] (in \_UART2\_CR1)*
- `#define _UART2_T8` ((uint8\_t) (0x01 << 6))  
*UART2 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART2\_CR1)*
- `#define _UART2_R8` ((uint8\_t) (0x01 << 7))  
*UART2 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART2\_CR1)*
- `#define _UART2_SBK` ((uint8\_t) (0x01 << 0))  
*UART2 Send break [0] (in \_UART2\_CR2)*
- `#define _UART2_RWU` ((uint8\_t) (0x01 << 1))

- UART2 Receiver wakeup [0] (in \_UART2\_CR2)*
- #define `_UART2_REN` ((uint8\_t) (0x01 << 2))
- UART2 Receiver enable [0] (in \_UART2\_CR2)*
- #define `_UART2_TEN` ((uint8\_t) (0x01 << 3))
- UART2 Transmitter enable [0] (in \_UART2\_CR2)*
- #define `_UART2_ILIEN` ((uint8\_t) (0x01 << 4))
- UART2 IDLE Line interrupt enable [0] (in \_UART2\_CR2)*
- #define `_UART2_RIEN` ((uint8\_t) (0x01 << 5))
- UART2 Receiver interrupt enable [0] (in \_UART2\_CR2)*
- #define `_UART2_TCIEN` ((uint8\_t) (0x01 << 6))
- UART2 Transmission complete interrupt enable [0] (in \_UART2\_CR2)*
- #define `_UART2_TIEN` ((uint8\_t) (0x01 << 7))
- UART2 Transmitter interrupt enable [0] (in \_UART2\_CR2)*
- #define `_UART2_LBCL` ((uint8\_t) (0x01 << 0))
- UART2 Last bit clock pulse [0] (in \_UART2\_CR3)*
- #define `_UART2_CPHA` ((uint8\_t) (0x01 << 1))
- UART2 Clock phase [0] (in \_UART2\_CR3)*
- #define `_UART2_CPOL` ((uint8\_t) (0x01 << 2))
- UART2 Clock polarity [0] (in \_UART2\_CR3)*
- #define `_UART2_CKEN` ((uint8\_t) (0x01 << 3))
- UART2 Clock enable [0] (in \_UART2\_CR3)*
- #define `_UART2_STOP` ((uint8\_t) (0x03 << 4))
- UART2 STOP bits [1:0] (in \_UART2\_CR3)*
- #define `_UART2_STOP0` ((uint8\_t) (0x01 << 4))
- UART2 STOP bits [0] (in \_UART2\_CR3)*
- #define `_UART2_STOP1` ((uint8\_t) (0x01 << 5))
- UART2 STOP bits [1] (in \_UART2\_CR3)*
- #define `_UART2_LINEN` ((uint8\_t) (0x01 << 6))
- UART2 LIN mode enable [0] (in \_UART2\_CR3)*
- #define `_UART2_ADD` ((uint8\_t) (0x0F << 0))
- UART2 Address of the UART node [3:0] (in \_UART2\_CR4)*
- #define `_UART2_ADD0` ((uint8\_t) (0x01 << 0))
- UART2 Address of the UART node [0] (in \_UART2\_CR4)*
- #define `_UART2_ADD1` ((uint8\_t) (0x01 << 1))
- UART2 Address of the UART node [1] (in \_UART2\_CR4)*
- #define `_UART2_ADD2` ((uint8\_t) (0x01 << 2))
- UART2 Address of the UART node [2] (in \_UART2\_CR4)*
- #define `_UART2_ADD3` ((uint8\_t) (0x01 << 3))
- UART2 Address of the UART node [3] (in \_UART2\_CR4)*
- #define `_UART2_LBDF` ((uint8\_t) (0x01 << 4))
- UART2 LIN Break Detection Flag [0] (in \_UART2\_CR4)*
- #define `_UART2_LBDL` ((uint8\_t) (0x01 << 5))
- UART2 LIN Break Detection Length [0] (in \_UART2\_CR4)*
- #define `_UART2_LBDIEN` ((uint8\_t) (0x01 << 6))
- UART2 LIN Break Detection Interrupt Enable [0] (in \_UART2\_CR4)*
- #define `_UART2_IREN` ((uint8\_t) (0x01 << 1))
- UART2 IrDA mode Enable [0] (in \_UART2\_CR5)*
- #define `_UART2_IRLP` ((uint8\_t) (0x01 << 2))
- UART2 IrDA Low Power [0] (in \_UART2\_CR5)*
- #define `_UART2_NACK` ((uint8\_t) (0x01 << 4))
- UART2 Smartcard NACK enable [0] (in \_UART2\_CR5)*

- `#define _UART2_SCEN ((uint8_t) (0x01 << 5))`  
*UART2 Smartcard mode enable [0] (in \_UART2\_CR5)*
- `#define _UART2_LSF ((uint8_t) (0x01 << 0))`  
*UART2 LIN Sync Field [0] (in \_UART2\_CR6)*
- `#define _UART2_LHDF ((uint8_t) (0x01 << 1))`  
*UART2 LIN Header Detection Flag [0] (in \_UART2\_CR6)*
- `#define _UART2_LHDIEN ((uint8_t) (0x01 << 2))`  
*UART2 LIN Header Detection Interrupt Enable [0] (in \_UART2\_CR6)*
- `#define _UART2_LASE ((uint8_t) (0x01 << 4))`  
*UART2 LIN automatic resynchronisation enable [0] (in \_UART2\_CR6)*
- `#define _UART2_LSLV ((uint8_t) (0x01 << 5))`  
*UART2 LIN Slave Enable [0] (in \_UART2\_CR6)*
- `#define _UART2_LDUM ((uint8_t) (0x01 << 7))`  
*UART2 LIN Divider Update Method [0] (in \_UART2\_CR6)*
- `#define _UART3_SFR(UART3_t, UART3_AddressBase)`  
*UART3 struct/bit access.*
- `#define _UART3_SR_SFR(uint8_t, UART3_AddressBase+0x00)`  
*UART3 Status register.*
- `#define _UART3_DR_SFR(uint8_t, UART3_AddressBase+0x01)`  
*UART3 data register.*
- `#define _UART3_BRR1_SFR(uint8_t, UART3_AddressBase+0x02)`  
*UART3 Baud rate register 1.*
- `#define _UART3_BRR2_SFR(uint8_t, UART3_AddressBase+0x03)`  
*UART3 Baud rate register 2.*
- `#define _UART3_CR1_SFR(uint8_t, UART3_AddressBase+0x04)`  
*UART3 Control register 1.*
- `#define _UART3_CR2_SFR(uint8_t, UART3_AddressBase+0x05)`  
*UART3 Control register 2.*
- `#define _UART3_CR3_SFR(uint8_t, UART3_AddressBase+0x06)`  
*UART3 Control register 3.*
- `#define _UART3_CR4_SFR(uint8_t, UART3_AddressBase+0x07)`  
*UART3 Control register 4.*
- `#define _UART3_CR6_SFR(uint8_t, UART3_AddressBase+0x09)`  
*UART3 Control register 6.*
- `#define _UART3_SR_RESET_VALUE ((uint8_t) 0xC0)`  
*UART3 Status register reset value.*
- `#define _UART3_BRR1_RESET_VALUE ((uint8_t) 0x00)`  
*UART3 Baud rate register 1 reset value.*
- `#define _UART3_BRR2_RESET_VALUE ((uint8_t) 0x00)`  
*UART3 Baud rate register 2 reset value.*
- `#define _UART3_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*UART3 Control register 1 reset value.*
- `#define _UART3_CR2_RESET_VALUE ((uint8_t) 0x00)`  
*UART3 Control register 2 reset value.*
- `#define _UART3_CR3_RESET_VALUE ((uint8_t) 0x00)`  
*UART3 Control register 3 reset value.*
- `#define _UART3_CR4_RESET_VALUE ((uint8_t) 0x00)`  
*UART3 Control register 4 reset value.*
- `#define _UART3_CR6_RESET_VALUE ((uint8_t) 0x00)`  
*UART3 Control register 6 reset value.*
- `#define _UART3_PE ((uint8_t) (0x01 << 0))`

- `UART3 Parity error [0] (in _UART3_SR)`
- `#define _UART3_FE ((uint8_t) (0x01 << 1))`  
`UART3 Framing error [0] (in _UART3_SR)`
- `#define _UART3_NF ((uint8_t) (0x01 << 2))`  
`UART3 Noise flag [0] (in _UART3_SR)`
- `#define _UART3_OR_LHE ((uint8_t) (0x01 << 3))`  
`UART3 LIN Header Error (LIN slave mode) / Overrun error [0] (in _UART3_SR)`
- `#define _UART3_IDLE ((uint8_t) (0x01 << 4))`  
`UART3 IDLE line detected [0] (in _UART3_SR)`
- `#define _UART3_RXNE ((uint8_t) (0x01 << 5))`  
`UART3 Read data register not empty [0] (in _UART3_SR)`
- `#define _UART3_TC ((uint8_t) (0x01 << 6))`  
`UART3 Transmission complete [0] (in _UART3_SR)`
- `#define _UART3_TXE ((uint8_t) (0x01 << 7))`  
`UART3 Transmit data register empty [0] (in _UART3_SR)`
- `#define _UART3_PIEN ((uint8_t) (0x01 << 0))`  
`UART3 Parity interrupt enable [0] (in _UART3_CR1)`
- `#define _UART3_PS ((uint8_t) (0x01 << 1))`  
`UART3 Parity selection [0] (in _UART3_CR1)`
- `#define _UART3_PCEN ((uint8_t) (0x01 << 2))`  
`UART3 Parity control enable [0] (in _UART3_CR1)`
- `#define _UART3_WAKE ((uint8_t) (0x01 << 3))`  
`UART3 Wakeup method [0] (in _UART3_CR1)`
- `#define _UART3_M ((uint8_t) (0x01 << 4))`  
`UART3 word length [0] (in _UART3_CR1)`
- `#define _UART3_UARTD ((uint8_t) (0x01 << 5))`  
`UART3 Disable (for low power consumption) [0] (in _UART3_CR1)`
- `#define _UART3_T8 ((uint8_t) (0x01 << 6))`  
`UART3 Transmit Data bit 8 (in 9-bit mode) [0] (in _UART3_CR1)`
- `#define _UART3_R8 ((uint8_t) (0x01 << 7))`  
`UART3 Receive Data bit 8 (in 9-bit mode) [0] (in _UART3_CR1)`
- `#define _UART3_SBK ((uint8_t) (0x01 << 0))`  
`UART3 Send break [0] (in _UART3_CR2)`
- `#define _UART3_RWU ((uint8_t) (0x01 << 1))`  
`UART3 Receiver wakeup [0] (in _UART3_CR2)`
- `#define _UART3_REN ((uint8_t) (0x01 << 2))`  
`UART3 Receiver enable [0] (in _UART3_CR2)`
- `#define _UART3_TEN ((uint8_t) (0x01 << 3))`  
`UART3 Transmitter enable [0] (in _UART3_CR2)`
- `#define _UART3_ILIEN ((uint8_t) (0x01 << 4))`  
`UART3 IDLE Line interrupt enable [0] (in _UART3_CR2)`
- `#define _UART3_RIEN ((uint8_t) (0x01 << 5))`  
`UART3 Receiver interrupt enable [0] (in _UART3_CR2)`
- `#define _UART3_TCIEN ((uint8_t) (0x01 << 6))`  
`UART3 Transmission complete interrupt enable [0] (in _UART3_CR2)`
- `#define _UART3_TIEN ((uint8_t) (0x01 << 7))`  
`UART3 Transmitter interrupt enable [0] (in _UART3_CR2)`
- `#define _UART3_STOP ((uint8_t) (0x03 << 4))`  
`UART3 STOP bits [1:0] (in _UART3_CR3)`
- `#define _UART3_STOP0 ((uint8_t) (0x01 << 4))`  
`UART3 STOP bits [0] (in _UART3_CR3)`



- `#define _UART3_STOP1 ((uint8_t) (0x01 << 5))`  
*UART3 STOP bits [1] (in \_UART3\_CR3)*
- `#define _UART3_LINEN ((uint8_t) (0x01 << 6))`  
*UART3 LIN mode enable [0] (in \_UART3\_CR3)*
- `#define _UART3_ADD ((uint8_t) (0x0F << 0))`  
*UART3 Address of the UART node [3:0] (in \_UART3\_CR4)*
- `#define _UART3_ADD0 ((uint8_t) (0x01 << 0))`  
*UART3 Address of the UART node [0] (in \_UART3\_CR4)*
- `#define _UART3_ADD1 ((uint8_t) (0x01 << 1))`  
*UART3 Address of the UART node [1] (in \_UART3\_CR4)*
- `#define _UART3_ADD2 ((uint8_t) (0x01 << 2))`  
*UART3 Address of the UART node [2] (in \_UART3\_CR4)*
- `#define _UART3_ADD3 ((uint8_t) (0x01 << 3))`  
*UART3 Address of the UART node [3] (in \_UART3\_CR4)*
- `#define _UART3_LBDF ((uint8_t) (0x01 << 4))`  
*UART3 LIN Break Detection Flag [0] (in \_UART3\_CR4)*
- `#define _UART3_LBDL ((uint8_t) (0x01 << 5))`  
*UART3 LIN Break Detection Length [0] (in \_UART3\_CR4)*
- `#define _UART3_LBDIEN ((uint8_t) (0x01 << 6))`  
*UART3 LIN Break Detection Interrupt Enable [0] (in \_UART3\_CR4)*
- `#define _UART3_LSF ((uint8_t) (0x01 << 0))`  
*UART3 LIN Sync Field [0] (in \_UART3\_CR6)*
- `#define _UART3_LHDF ((uint8_t) (0x01 << 1))`  
*UART3 LIN Header Detection Flag [0] (in \_UART3\_CR6)*
- `#define _UART3_LHDIEN ((uint8_t) (0x01 << 2))`  
*UART3 LIN Header Detection Interrupt Enable [0] (in \_UART3\_CR6)*
- `#define _UART3_LASE ((uint8_t) (0x01 << 4))`  
*UART3 LIN automatic resynchronisation enable [0] (in \_UART3\_CR6)*
- `#define _UART3_LSLV ((uint8_t) (0x01 << 5))`  
*UART3 LIN Slave Enable [0] (in \_UART3\_CR6)*
- `#define _UART3_LDUM ((uint8_t) (0x01 << 7))`  
*UART3 LIN Divider Update Method [0] (in \_UART3\_CR6)*
- `#define _UART4_SFR(UART4_t, UART4_AddressBase)`  
*UART4 struct/bit access.*
- `#define _UART4_SR_SFR(uint8_t, UART4_AddressBase+0x00)`  
*UART4 Status register.*
- `#define _UART4_DR_SFR(uint8_t, UART4_AddressBase+0x01)`  
*UART4 data register.*
- `#define _UART4_BRR1_SFR(uint8_t, UART4_AddressBase+0x02)`  
*UART4 Baud rate register 1.*
- `#define _UART4_BRR2_SFR(uint8_t, UART4_AddressBase+0x03)`  
*UART4 Baud rate register 2.*
- `#define _UART4_CR1_SFR(uint8_t, UART4_AddressBase+0x04)`  
*UART4 Control register 1.*
- `#define _UART4_CR2_SFR(uint8_t, UART4_AddressBase+0x05)`  
*UART4 Control register 2.*
- `#define _UART4_CR3_SFR(uint8_t, UART4_AddressBase+0x06)`  
*UART4 Control register 3.*
- `#define _UART4_CR4_SFR(uint8_t, UART4_AddressBase+0x07)`  
*UART4 Control register 4.*
- `#define _UART4_CR5_SFR(uint8_t, UART4_AddressBase+0x08)`



- UART4 Control register 5.*
  - #define `_UART4_CR6_SFR`(uint8\_t, `UART4_AddressBase+0x09`)
- UART4 Control register 6.*
  - #define `_UART4_GTR_SFR`(uint8\_t, `UART4_AddressBase+0x0A`)
- UART4 guard time register.*
  - #define `_UART4_PSCR_SFR`(uint8\_t, `UART4_AddressBase+0x0B`)
- UART4 prescaler register.*
  - #define `_UART4_SR_RESET_VALUE` ((uint8\_t) 0xC0)
- UART4 Status register reset value.*
  - #define `_UART4_BRR1_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Baud rate register 1 reset value.*
  - #define `_UART4_BRR2_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Baud rate register 2 reset value.*
  - #define `_UART4_CR1_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Control register 1 reset value.*
  - #define `_UART4_CR2_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Control register 2 reset value.*
  - #define `_UART4_CR3_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Control register 3 reset value.*
  - #define `_UART4_CR4_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Control register 4 reset value.*
  - #define `_UART4_CR5_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Control register 5 reset value.*
  - #define `_UART4_CR6_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Control register 6 reset value.*
  - #define `_UART4_GTR_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 guard time register reset value.*
  - #define `_UART4_PSCR_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 prescaler register reset value.*
  - #define `_UART4_PE` ((uint8\_t) (0x01 << 0))
- UART4 Parity error [0] (in \_UART4\_SR)*
  - #define `_UART4_FE` ((uint8\_t) (0x01 << 1))
- UART4 Framing error [0] (in \_UART4\_SR)*
  - #define `_UART4_NF` ((uint8\_t) (0x01 << 2))
- UART4 Noise flag [0] (in \_UART4\_SR)*
  - #define `_UART4_OR_LHE` ((uint8\_t) (0x01 << 3))
- UART4 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART4\_SR)*
  - #define `_UART4_IDLE` ((uint8\_t) (0x01 << 4))
- UART4 IDLE line detected [0] (in \_UART4\_SR)*
  - #define `_UART4_RXNE` ((uint8\_t) (0x01 << 5))
- UART4 Read data register not empty [0] (in \_UART4\_SR)*
  - #define `_UART4_TC` ((uint8\_t) (0x01 << 6))
- UART4 Transmission complete [0] (in \_UART4\_SR)*
  - #define `_UART4_TXE` ((uint8\_t) (0x01 << 7))
- UART4 Transmit data register empty [0] (in \_UART4\_SR)*
  - #define `_UART4_PIEN` ((uint8\_t) (0x01 << 0))
- UART4 Parity interrupt enable [0] (in \_UART4\_CR1)*
  - #define `_UART4_PS` ((uint8\_t) (0x01 << 1))
- UART4 Parity selection [0] (in \_UART4\_CR1)*
  - #define `_UART4_PCEN` ((uint8\_t) (0x01 << 2))
- UART4 Parity control enable [0] (in \_UART4\_CR1)*

- `#define _UART4_WAKE ((uint8_t) (0x01 << 3))`  
*UART4 Wakeup method [0] (in \_UART4\_CR1)*
- `#define _UART4_M ((uint8_t) (0x01 << 4))`  
*UART4 word length [0] (in \_UART4\_CR1)*
- `#define _UART4_UARTD ((uint8_t) (0x01 << 5))`  
*UART4 Disable (for low power consumption) [0] (in \_UART4\_CR1)*
- `#define _UART4_T8 ((uint8_t) (0x01 << 6))`  
*UART4 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART4\_CR1)*
- `#define _UART4_R8 ((uint8_t) (0x01 << 7))`  
*UART4 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART4\_CR1)*
- `#define _UART4_SBK ((uint8_t) (0x01 << 0))`  
*UART4 Send break [0] (in \_UART4\_CR2)*
- `#define _UART4_RWU ((uint8_t) (0x01 << 1))`  
*UART4 Receiver wakeup [0] (in \_UART4\_CR2)*
- `#define _UART4_REN ((uint8_t) (0x01 << 2))`  
*UART4 Receiver enable [0] (in \_UART4\_CR2)*
- `#define _UART4_TEN ((uint8_t) (0x01 << 3))`  
*UART4 Transmitter enable [0] (in \_UART4\_CR2)*
- `#define _UART4_ILIEN ((uint8_t) (0x01 << 4))`  
*UART4 IDLE Line interrupt enable [0] (in \_UART4\_CR2)*
- `#define _UART4_RIEN ((uint8_t) (0x01 << 5))`  
*UART4 Receiver interrupt enable [0] (in \_UART4\_CR2)*
- `#define _UART4_TCIEN ((uint8_t) (0x01 << 6))`  
*UART4 Transmission complete interrupt enable [0] (in \_UART4\_CR2)*
- `#define _UART4_TIEN ((uint8_t) (0x01 << 7))`  
*UART4 Transmitter interrupt enable [0] (in \_UART4\_CR2)*
- `#define _UART4_LBCL ((uint8_t) (0x01 << 0))`  
*UART4 Last bit clock pulse [0] (in \_UART4\_CR3)*
- `#define _UART4_CPHA ((uint8_t) (0x01 << 1))`  
*UART4 Clock phase [0] (in \_UART4\_CR3)*
- `#define _UART4_CPOL ((uint8_t) (0x01 << 2))`  
*UART4 Clock polarity [0] (in \_UART4\_CR3)*
- `#define _UART4_CKEN ((uint8_t) (0x01 << 3))`  
*UART4 Clock enable [0] (in \_UART4\_CR3)*
- `#define _UART4_STOP ((uint8_t) (0x03 << 4))`  
*UART4 STOP bits [1:0] (in \_UART4\_CR3)*
- `#define _UART4_STOP0 ((uint8_t) (0x01 << 4))`  
*UART4 STOP bits [0] (in \_UART4\_CR3)*
- `#define _UART4_STOP1 ((uint8_t) (0x01 << 5))`  
*UART4 STOP bits [1] (in \_UART4\_CR3)*
- `#define _UART4_LINEN ((uint8_t) (0x01 << 6))`  
*UART4 LIN mode enable [0] (in \_UART4\_CR3)*
- `#define _UART4_ADD ((uint8_t) (0x0F << 0))`  
*UART4 Address of the UART node [3:0] (in \_UART4\_CR4)*
- `#define _UART4_ADD0 ((uint8_t) (0x01 << 0))`  
*UART4 Address of the UART node [0] (in \_UART4\_CR4)*
- `#define _UART4_ADD1 ((uint8_t) (0x01 << 1))`  
*UART4 Address of the UART node [1] (in \_UART4\_CR4)*
- `#define _UART4_ADD2 ((uint8_t) (0x01 << 2))`  
*UART4 Address of the UART node [2] (in \_UART4\_CR4)*
- `#define _UART4_ADD3 ((uint8_t) (0x01 << 3))`

- UART4 Address of the UART node [3] (in \_UART4\_CR4)*
  - #define `_UART4_LBDF` ((uint8\_t) (0x01 << 4))
  - UART4 LIN Break Detection Flag [0] (in \_UART4\_CR4)*
    - #define `_UART4_LBDL` ((uint8\_t) (0x01 << 5))
    - UART4 LIN Break Detection Length [0] (in \_UART4\_CR4)*
      - #define `_UART4_LBDIEN` ((uint8\_t) (0x01 << 6))
      - UART4 LIN Break Detection Interrupt Enable [0] (in \_UART4\_CR4)*
        - #define `_UART4_IREN` ((uint8\_t) (0x01 << 1))
        - UART4 IrDA mode Enable [0] (in \_UART4\_CR5)*
          - #define `_UART4_IRLP` ((uint8\_t) (0x01 << 2))
          - UART4 IrDA Low Power [0] (in \_UART4\_CR5)*
            - #define `_UART4_HDSEL` ((uint8\_t) (0x01 << 3))
            - UART4 Half-Duplex Selection [0] (in \_UART4\_CR5)*
              - #define `_UART4_NACK` ((uint8\_t) (0x01 << 4))
              - UART4 Smartcard NACK enable [0] (in \_UART4\_CR5)*
                - #define `_UART4_SCEN` ((uint8\_t) (0x01 << 5))
                - UART4 Smartcard mode enable [0] (in \_UART4\_CR5)*
                  - #define `_UART4_LSF` ((uint8\_t) (0x01 << 0))
                  - UART4 LIN Sync Field [0] (in \_UART4\_CR6)*
                    - #define `_UART4_LHDF` ((uint8\_t) (0x01 << 1))
                    - UART4 LIN Header Detection Flag [0] (in \_UART4\_CR6)*
                      - #define `_UART4_LHDIEN` ((uint8\_t) (0x01 << 2))
                      - UART4 LIN Header Detection Interrupt Enable [0] (in \_UART4\_CR6)*
                        - #define `_UART4_LASE` ((uint8\_t) (0x01 << 4))
                        - UART4 LIN automatic resynchronisation enable [0] (in \_UART4\_CR6)*
                          - #define `_UART4_LSLV` ((uint8\_t) (0x01 << 5))
                          - UART4 LIN Slave Enable [0] (in \_UART4\_CR6)*
                            - #define `_UART4_LDUM` ((uint8\_t) (0x01 << 7))
                            - UART4 LIN Divider Update Method [0] (in \_UART4\_CR6)*
                              - #define `_TIM1_SFR`(TIM1\_t, TIM1\_AddressBase)
                - TIM1 struct/bit access.*
                - #define `_TIM1_CR1_SFR`(uint8\_t, TIM1\_AddressBase+0x00)
                - TIM1 control register 1.*
                - #define `_TIM1_CR2_SFR`(uint8\_t, TIM1\_AddressBase+0x01)
                - TIM1 control register 2.*
                - #define `_TIM1_SMCR_SFR`(uint8\_t, TIM1\_AddressBase+0x02)
                - TIM1 Slave mode control register.*
                - #define `_TIM1_ETR_SFR`(uint8\_t, TIM1\_AddressBase+0x03)
                - TIM1 External trigger register.*
                - #define `_TIM1_IER_SFR`(uint8\_t, TIM1\_AddressBase+0x04)
                - TIM1 interrupt enable register.*
                - #define `_TIM1_SR1_SFR`(uint8\_t, TIM1\_AddressBase+0x05)
                - TIM1 status register 1.*
                - #define `_TIM1_SR2_SFR`(uint8\_t, TIM1\_AddressBase+0x06)
                - TIM1 status register 2.*
                - #define `_TIM1_EGR_SFR`(uint8\_t, TIM1\_AddressBase+0x07)
                - TIM1 Event generation register.*
                - #define `_TIM1_CCMR1_SFR`(uint8\_t, TIM1\_AddressBase+0x08)
                - TIM1 Capture/compare mode register 1.*
                - #define `_TIM1_CCMR2_SFR`(uint8\_t, TIM1\_AddressBase+0x09)
                - TIM1 Capture/compare mode register 2.*

- `#define _TIM1_CCMR3_SFR(uint8_t, TIM1_AddressBase+0x0A)`  
*TIM1 Capture/compare mode register 3.*
- `#define _TIM1_CCMR4_SFR(uint8_t, TIM1_AddressBase+0x0B)`  
*TIM1 Capture/compare mode register 4.*
- `#define _TIM1_CCER1_SFR(uint8_t, TIM1_AddressBase+0x0C)`  
*TIM1 Capture/compare enable register 1.*
- `#define _TIM1_CCER2_SFR(uint8_t, TIM1_AddressBase+0x0D)`  
*TIM1 Capture/compare enable register 2.*
- `#define _TIM1_CNTRH_SFR(uint8_t, TIM1_AddressBase+0x0E)`  
*TIM1 counter register high byte.*
- `#define _TIM1_CNTRL_SFR(uint8_t, TIM1_AddressBase+0x0F)`  
*TIM1 counter register low byte.*
- `#define _TIM1_PSCRH_SFR(uint8_t, TIM1_AddressBase+0x10)`  
*TIM1 clock prescaler register high byte.*
- `#define _TIM1_PSCRL_SFR(uint8_t, TIM1_AddressBase+0x11)`  
*TIM1 clock prescaler register low byte.*
- `#define _TIM1_ARRH_SFR(uint8_t, TIM1_AddressBase+0x12)`  
*TIM1 auto-reload register high byte.*
- `#define _TIM1_ARRL_SFR(uint8_t, TIM1_AddressBase+0x13)`  
*TIM1 auto-reload register low byte.*
- `#define _TIM1_RCR_SFR(uint8_t, TIM1_AddressBase+0x14)`  
*TIM1 Repetition counter.*
- `#define _TIM1_CCR1H_SFR(uint8_t, TIM1_AddressBase+0x15)`  
*TIM1 16-bit capture/compare value 1 high byte.*
- `#define _TIM1_CCR1L_SFR(uint8_t, TIM1_AddressBase+0x16)`  
*TIM1 16-bit capture/compare value 1 low byte.*
- `#define _TIM1_CCR2H_SFR(uint8_t, TIM1_AddressBase+0x17)`  
*TIM1 16-bit capture/compare value 2 high byte.*
- `#define _TIM1_CCR2L_SFR(uint8_t, TIM1_AddressBase+0x18)`  
*TIM1 16-bit capture/compare value 2 low byte.*
- `#define _TIM1_CCR3H_SFR(uint8_t, TIM1_AddressBase+0x19)`  
*TIM1 16-bit capture/compare value 3 high byte.*
- `#define _TIM1_CCR3L_SFR(uint8_t, TIM1_AddressBase+0x1A)`  
*TIM1 16-bit capture/compare value 3 low byte.*
- `#define _TIM1_CCR4H_SFR(uint8_t, TIM1_AddressBase+0x1B)`  
*TIM1 16-bit capture/compare value 4 high byte.*
- `#define _TIM1_CCR4L_SFR(uint8_t, TIM1_AddressBase+0x1C)`  
*TIM1 16-bit capture/compare value 4 low byte.*
- `#define _TIM1_BKR_SFR(uint8_t, TIM1_AddressBase+0x1D)`  
*TIM1 Break register.*
- `#define _TIM1_DTR_SFR(uint8_t, TIM1_AddressBase+0x1E)`  
*TIM1 Dead-time register.*
- `#define _TIM1_OISR_SFR(uint8_t, TIM1_AddressBase+0x1F)`  
*TIM1 Output idle state register.*
- `#define _TIM1_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM1 control register 1 reset value.*
- `#define _TIM1_CR2_RESET_VALUE ((uint8_t) 0x00)`  
*TIM1 control register 2 reset value.*
- `#define _TIM1_SMCR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM1 Slave mode control register reset value.*
- `#define _TIM1_ETR_RESET_VALUE ((uint8_t) 0x00)`

- TIM1 External trigger register reset value.*
- #define `_TIM1_IER_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 interrupt enable register reset value.*
- #define `_TIM1_SR1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 status register 1 reset value.*
- #define `_TIM1_SR2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 status register 2 reset value.*
- #define `_TIM1_EGR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Event generation register reset value.*
- #define `_TIM1_CCMR1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Capture/compare mode register 1 reset value.*
- #define `_TIM1_CCMR2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Capture/compare mode register 2 reset value.*
- #define `_TIM1_CCMR3_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Capture/compare mode register 3 reset value.*
- #define `_TIM1_CCMR4_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Capture/compare mode register 4 reset value.*
- #define `_TIM1_CCER1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Capture/compare enable register 1 reset value.*
- #define `_TIM1_CCER2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Capture/compare enable register 2 reset value.*
- #define `_TIM1_CNTRH_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 counter register high byte reset value.*
- #define `_TIM1_CNTRL_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 counter register low byte reset value.*
- #define `_TIM1_PSCRH_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 clock prescaler register high byte reset value.*
- #define `_TIM1_PSCRL_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 clock prescaler register low byte reset value.*
- #define `_TIM1_ARRH_RESET_VALUE` ((uint8\_t) 0xFF)
- TIM1 auto-reload register high byte reset value.*
- #define `_TIM1_ARRL_RESET_VALUE` ((uint8\_t) 0xFF)
- TIM1 auto-reload register low byte reset value.*
- #define `_TIM1_RCR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 Repetition counter reset value.*
- #define `_TIM1_CCR1H_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 1 high byte reset value.*
- #define `_TIM1_CCR1L_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 1 low byte reset value.*
- #define `_TIM1_CCR2H_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 2 high byte reset value.*
- #define `_TIM1_CCR2L_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 2 low byte reset value.*
- #define `_TIM1_CCR3H_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 3 high byte reset value.*
- #define `_TIM1_CCR3L_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 3 low byte reset value.*
- #define `_TIM1_CCR4H_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 4 high byte reset value.*
- #define `_TIM1_CCR4L_RESET_VALUE` ((uint8\_t) 0x00)
- TIM1 16-bit capture/compare value 4 low byte reset value.*

- `#define _TIM1_BKR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM1 Break register reset value.*
- `#define _TIM1_DTR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM1 Dead-time register reset value.*
- `#define _TIM1_OISR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM1 Output idle state register reset value.*
- `#define _TIM1_CEN ((uint8_t) (0x01 << 0))`  
*TIM1 Counter enable [0] (in \_TIM1\_CR1)*
- `#define _TIM1_UDIS ((uint8_t) (0x01 << 1))`  
*TIM1 Update disable [0] (in \_TIM1\_CR1)*
- `#define _TIM1_URS ((uint8_t) (0x01 << 2))`  
*TIM1 Update request source [0] (in \_TIM1\_CR1)*
- `#define _TIM1_OPM ((uint8_t) (0x01 << 3))`  
*TIM1 One-pulse mode [0] (in \_TIM1\_CR1)*
- `#define _TIM1_DIR ((uint8_t) (0x01 << 4))`  
*TIM1 Direction [0] (in \_TIM1\_CR1)*
- `#define _TIM1_CMS ((uint8_t) (0x03 << 5))`  
*TIM1 Center-aligned mode selection [1:0] (in \_TIM1\_CR1)*
- `#define _TIM1_CMS0 ((uint8_t) (0x01 << 5))`  
*TIM1 Center-aligned mode selection [0] (in \_TIM1\_CR1)*
- `#define _TIM1_CMS1 ((uint8_t) (0x01 << 6))`  
*TIM1 Center-aligned mode selection [1] (in \_TIM1\_CR1)*
- `#define _TIM1_ARPE ((uint8_t) (0x01 << 7))`  
*TIM1 Auto-reload preload enable [0] (in \_TIM1\_CR1)*
- `#define _TIM1_CCPC ((uint8_t) (0x01 << 0))`  
*TIM1 Capture/compare preloaded control [0] (in \_TIM1\_CR2)*
- `#define _TIM1_COMS ((uint8_t) (0x01 << 2))`  
*TIM1 Capture/compare control update selection [0] (in \_TIM1\_CR2)*
- `#define _TIM1_MMS ((uint8_t) (0x07 << 4))`  
*TIM1 Master mode selection [2:0] (in \_TIM1\_CR2)*
- `#define _TIM1_MMS0 ((uint8_t) (0x01 << 4))`  
*TIM1 Master mode selection [0] (in \_TIM1\_CR2)*
- `#define _TIM1_MMS1 ((uint8_t) (0x01 << 5))`  
*TIM1 Master mode selection [1] (in \_TIM1\_CR2)*
- `#define _TIM1_MMS2 ((uint8_t) (0x01 << 6))`  
*TIM1 Master mode selection [2] (in \_TIM1\_CR2)*
- `#define _TIM1_SMS ((uint8_t) (0x07 << 0))`  
*TIM1 Clock/trigger/slave mode selection [2:0] (in \_TIM1\_SMCR)*
- `#define _TIM1_SMS0 ((uint8_t) (0x01 << 0))`  
*TIM1 Clock/trigger/slave mode selection [0] (in \_TIM1\_SMCR)*
- `#define _TIM1_SMS1 ((uint8_t) (0x01 << 1))`  
*TIM1 Clock/trigger/slave mode selection [1] (in \_TIM1\_SMCR)*
- `#define _TIM1_SMS2 ((uint8_t) (0x01 << 2))`  
*TIM1 Clock/trigger/slave mode selection [2] (in \_TIM1\_SMCR)*
- `#define _TIM1_TS ((uint8_t) (0x07 << 4))`  
*TIM1 Trigger selection [2:0] (in \_TIM1\_SMCR)*
- `#define _TIM1_TS0 ((uint8_t) (0x01 << 4))`  
*TIM1 Trigger selection [0] (in \_TIM1\_SMCR)*
- `#define _TIM1_TS1 ((uint8_t) (0x01 << 5))`  
*TIM1 Trigger selection [1] (in \_TIM1\_SMCR)*
- `#define _TIM1_TS2 ((uint8_t) (0x01 << 6))`

- TIM1 Trigger selection [2] (in \_TIM1\_SMCR)*
- #define `_TIM1_MSM` ((uint8\_t) (0x01 << 7))
- TIM1 Master/slave mode [0] (in \_TIM1\_SMCR)*
- #define `_TIM1_ETF` ((uint8\_t) (0x0F << 0))
- TIM1 External trigger filter [3:0] (in \_TIM1\_ETR)*
- #define `_TIM1_ETF0` ((uint8\_t) (0x01 << 0))
- TIM1 External trigger filter [0] (in \_TIM1\_ETR)*
- #define `_TIM1_ETF1` ((uint8\_t) (0x01 << 1))
- TIM1 External trigger filter [1] (in \_TIM1\_ETR)*
- #define `_TIM1_ETF2` ((uint8\_t) (0x01 << 2))
- TIM1 External trigger filter [2] (in \_TIM1\_ETR)*
- #define `_TIM1_ETF3` ((uint8\_t) (0x01 << 3))
- TIM1 External trigger filter [3] (in \_TIM1\_ETR)*
- #define `_TIM1_ETPS` ((uint8\_t) (0x03 << 4))
- TIM1 External trigger prescaler [1:0] (in \_TIM1\_ETR)*
- #define `_TIM1_ETPS0` ((uint8\_t) (0x01 << 4))
- TIM1 External trigger prescaler [0] (in \_TIM1\_ETR)*
- #define `_TIM1_ETPS1` ((uint8\_t) (0x01 << 5))
- TIM1 External trigger prescaler [1] (in \_TIM1\_ETR)*
- #define `_TIM1_ECE` ((uint8\_t) (0x01 << 6))
- TIM1 External clock enable [0] (in \_TIM1\_ETR)*
- #define `_TIM1_ETP` ((uint8\_t) (0x01 << 7))
- TIM1 External trigger polarity [0] (in \_TIM1\_ETR)*
- #define `_TIM1_UIE` ((uint8\_t) (0x01 << 0))
- TIM1 Update interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_CC1IE` ((uint8\_t) (0x01 << 1))
- TIM1 Capture/compare 1 interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_CC2IE` ((uint8\_t) (0x01 << 2))
- TIM1 Capture/compare 2 interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_CC3IE` ((uint8\_t) (0x01 << 3))
- TIM1 Capture/compare 3 interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_CC4IE` ((uint8\_t) (0x01 << 4))
- TIM1 Capture/compare 4 interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_COMIE` ((uint8\_t) (0x01 << 5))
- TIM1 Commutation interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_TIE` ((uint8\_t) (0x01 << 6))
- TIM1 Trigger interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_BIE` ((uint8\_t) (0x01 << 7))
- TIM1 Break interrupt enable [0] (in \_TIM1\_IER)*
- #define `_TIM1_UIF` ((uint8\_t) (0x01 << 0))
- TIM1 Update interrupt flag [0] (in \_TIM1\_SR1)*
- #define `_TIM1_CC1IF` ((uint8\_t) (0x01 << 1))
- TIM1 Capture/compare 1 interrupt flag [0] (in \_TIM1\_SR1)*
- #define `_TIM1_CC2IF` ((uint8\_t) (0x01 << 2))
- TIM1 Capture/compare 2 interrupt flag [0] (in \_TIM1\_SR1)*
- #define `_TIM1_CC3IF` ((uint8\_t) (0x01 << 3))
- TIM1 Capture/compare 3 interrupt flag [0] (in \_TIM1\_SR1)*
- #define `_TIM1_CC4IF` ((uint8\_t) (0x01 << 4))
- TIM1 Capture/compare 4 interrupt flag [0] (in \_TIM1\_SR1)*
- #define `_TIM1_COMIF` ((uint8\_t) (0x01 << 5))
- TIM1 Commutation interrupt flag [0] (in \_TIM1\_SR1)*



- `#define _TIM1_TIF` ((uint8\_t) (0x01 << 6))  
*TIM1 Trigger interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_BIF` ((uint8\_t) (0x01 << 7))  
*TIM1 Break interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_CC1OF` ((uint8\_t) (0x01 << 1))  
*TIM1 Capture/compare 1 overcapture flag [0] (in \_TIM1\_SR2)*
- `#define _TIM1_CC2OF` ((uint8\_t) (0x01 << 2))  
*TIM1 Capture/compare 2 overcapture flag [0] (in \_TIM1\_SR2)*
- `#define _TIM1_CC3OF` ((uint8\_t) (0x01 << 3))  
*TIM1 Capture/compare 3 overcapture flag [0] (in \_TIM1\_SR2)*
- `#define _TIM1_CC4OF` ((uint8\_t) (0x01 << 4))  
*TIM1 Capture/compare 4 overcapture flag [0] (in \_TIM1\_SR2)*
- `#define _TIM1_UG` ((uint8\_t) (0x01 << 0))  
*TIM1 Update generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_CC1G` ((uint8\_t) (0x01 << 1))  
*TIM1 Capture/compare 1 generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_CC2G` ((uint8\_t) (0x01 << 2))  
*TIM1 Capture/compare 2 generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_CC3G` ((uint8\_t) (0x01 << 3))  
*TIM1 Capture/compare 3 generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_CC4G` ((uint8\_t) (0x01 << 4))  
*TIM1 Capture/compare 4 generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_COMG` ((uint8\_t) (0x01 << 5))  
*TIM1 Capture/compare control update generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_TG` ((uint8\_t) (0x01 << 6))  
*TIM1 Trigger generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_BG` ((uint8\_t) (0x01 << 7))  
*TIM1 Break generation [0] (in \_TIM1\_EGR)*
- `#define _TIM1_CC1S` ((uint8\_t) (0x03 << 0))  
*TIM1 Compare 1 selection [1:0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_CC1S0` ((uint8\_t) (0x01 << 0))  
*TIM1 Compare 1 selection [0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_CC1S1` ((uint8\_t) (0x01 << 1))  
*TIM1 Compare 1 selection [1] (in \_TIM1\_CCMR1)*
- `#define _TIM1_OC1FE` ((uint8\_t) (0x01 << 2))  
*TIM1 Output compare 1 fast enable [0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_OC1PE` ((uint8\_t) (0x01 << 3))  
*TIM1 Output compare 1 preload enable [0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_OC1M` ((uint8\_t) (0x07 << 4))  
*TIM1 Output compare 1 mode [2:0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_OC1M0` ((uint8\_t) (0x01 << 4))  
*TIM1 Output compare 1 mode [0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_OC1M1` ((uint8\_t) (0x01 << 5))  
*TIM1 Output compare 1 mode [1] (in \_TIM1\_CCMR1)*
- `#define _TIM1_OC1M2` ((uint8\_t) (0x01 << 6))  
*TIM1 Output compare 1 mode [2] (in \_TIM1\_CCMR1)*
- `#define _TIM1_OC1CE` ((uint8\_t) (0x01 << 7))  
*TIM1 Output compare 1 clear enable [0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_IC1PSC` ((uint8\_t) (0x03 << 2))  
*TIM1 Input capture 1 prescaler [1:0] (in \_TIM1\_CCMR1)*
- `#define _TIM1_IC1PSC0` ((uint8\_t) (0x01 << 2))



- TIM1 Input capture 1 prescaler [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1PSC1` ((uint8\_t) (0x01 << 3))
- TIM1 Input capture 1 prescaler [1] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F` ((uint8\_t) (0x0F << 4))
- TIM1 Output compare 1 mode [3:0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F0` ((uint8\_t) (0x01 << 4))
- TIM1 Input capture 1 filter [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F1` ((uint8\_t) (0x01 << 5))
- TIM1 Input capture 1 filter [1] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F2` ((uint8\_t) (0x01 << 6))
- TIM1 Input capture 1 filter [2] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F3` ((uint8\_t) (0x01 << 7))
- TIM1 Input capture 1 filter [3] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_CC2S` ((uint8\_t) (0x03 << 0))
- TIM1 Compare 2 selection [1:0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_CC2S0` ((uint8\_t) (0x01 << 0))
- TIM1 Compare 2 selection [0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_CC2S1` ((uint8\_t) (0x01 << 1))
- TIM1 Compare 2 selection [1] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_OC2FE` ((uint8\_t) (0x01 << 2))
- TIM1 Output compare 2 fast enable [0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_OC2PE` ((uint8\_t) (0x01 << 3))
- TIM1 Output compare 2 preload enable [0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_OC2M` ((uint8\_t) (0x07 << 4))
- TIM1 Output compare 2 mode [2:0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_OC2M0` ((uint8\_t) (0x01 << 4))
- TIM1 Output compare 2 mode [0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_OC2M1` ((uint8\_t) (0x01 << 5))
- TIM1 Output compare 2 mode [1] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_OC2M2` ((uint8\_t) (0x01 << 6))
- TIM1 Output compare 2 mode [2] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_OC2CE` ((uint8\_t) (0x01 << 7))
- TIM1 Output compare 2 clear enable [0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2PSC` ((uint8\_t) (0x03 << 2))
- TIM1 Input capture 2 prescaler [1:0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2PSC0` ((uint8\_t) (0x01 << 2))
- TIM1 Input capture 2 prescaler [0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2PSC1` ((uint8\_t) (0x01 << 3))
- TIM1 Input capture 2 prescaler [1] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2F` ((uint8\_t) (0x0F << 4))
- TIM1 Output compare 2 mode [3:0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2F0` ((uint8\_t) (0x01 << 4))
- TIM1 Input capture 2 filter [0] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2F1` ((uint8\_t) (0x01 << 5))
- TIM1 Input capture 2 filter [1] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2F2` ((uint8\_t) (0x01 << 6))
- TIM1 Input capture 2 filter [2] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_IC2F3` ((uint8\_t) (0x01 << 7))
- TIM1 Input capture 2 filter [3] (in \_TIM1\_CCMR2)*
  - #define `_TIM1_CC3S` ((uint8\_t) (0x03 << 0))
- TIM1 Compare 3 selection [1:0] (in \_TIM1\_CCMR3)*

- `#define _TIM1_CC3S0 ((uint8_t) (0x01 << 0))`  
*TIM1 Compare 3 selection [0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_CC3S1 ((uint8_t) (0x01 << 1))`  
*TIM1 Compare 3 selection [1] (in \_TIM1\_CCMR3)*
- `#define _TIM1_OC3FE ((uint8_t) (0x01 << 2))`  
*TIM1 Output compare 3 fast enable [0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_OC3PE ((uint8_t) (0x01 << 3))`  
*TIM1 Output compare 3 preload enable [0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_OC3M ((uint8_t) (0x07 << 4))`  
*TIM1 Output compare 3 mode [2:0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_OC3M0 ((uint8_t) (0x01 << 4))`  
*TIM1 Output compare 3 mode [0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_OC3M1 ((uint8_t) (0x01 << 5))`  
*TIM1 Output compare 3 mode [1] (in \_TIM1\_CCMR3)*
- `#define _TIM1_OC3M2 ((uint8_t) (0x01 << 6))`  
*TIM1 Output compare 3 mode [2] (in \_TIM1\_CCMR3)*
- `#define _TIM1_OC3CE ((uint8_t) (0x01 << 7))`  
*TIM1 Output compare 3 clear enable [0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3PSC ((uint8_t) (0x03 << 2))`  
*TIM1 Input capture 3 prescaler [1:0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3PSC0 ((uint8_t) (0x01 << 2))`  
*TIM1 Input capture 3 prescaler [0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3PSC1 ((uint8_t) (0x01 << 3))`  
*TIM1 Input capture 3 prescaler [1] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3F ((uint8_t) (0x0F << 4))`  
*TIM1 Output compare 3 mode [3:0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3F0 ((uint8_t) (0x01 << 4))`  
*TIM1 Input capture 3 filter [0] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3F1 ((uint8_t) (0x01 << 5))`  
*TIM1 Input capture 3 filter [1] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3F2 ((uint8_t) (0x01 << 6))`  
*TIM1 Input capture 3 filter [2] (in \_TIM1\_CCMR3)*
- `#define _TIM1_IC3F3 ((uint8_t) (0x01 << 7))`  
*TIM1 Input capture 3 filter [3] (in \_TIM1\_CCMR3)*
- `#define _TIM1_CC4S ((uint8_t) (0x03 << 0))`  
*TIM1 Compare 4 selection [1:0] (in \_TIM1\_CCMR4)*
- `#define _TIM1_CC4S0 ((uint8_t) (0x01 << 0))`  
*TIM1 Compare 4 selection [0] (in \_TIM1\_CCMR4)*
- `#define _TIM1_CC4S1 ((uint8_t) (0x01 << 1))`  
*TIM1 Compare 4 selection [1] (in \_TIM1\_CCMR4)*
- `#define _TIM1_OC4FE ((uint8_t) (0x01 << 2))`  
*TIM1 Output compare 4 fast enable [0] (in \_TIM1\_CCMR4)*
- `#define _TIM1_OC4PE ((uint8_t) (0x01 << 3))`  
*TIM1 Output compare 4 preload enable [0] (in \_TIM1\_CCMR4)*
- `#define _TIM1_OC4M ((uint8_t) (0x07 << 4))`  
*TIM1 Output compare 4 mode [2:0] (in \_TIM1\_CCMR4)*
- `#define _TIM1_OC4M0 ((uint8_t) (0x01 << 4))`  
*TIM1 Output compare 4 mode [0] (in \_TIM1\_CCMR4)*
- `#define _TIM1_OC4M1 ((uint8_t) (0x01 << 5))`  
*TIM1 Output compare 4 mode [1] (in \_TIM1\_CCMR4)*
- `#define _TIM1_OC4M2 ((uint8_t) (0x01 << 6))`

- TIM1 Output compare 4 mode [2] (in \_TIM1\_CCMR4)*
- #define `_TIM1_OC4CE` ((uint8\_t) (0x01 << 7))
- TIM1 Output compare 4 clear enable [0] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4PSC` ((uint8\_t) (0x03 << 2))
- TIM1 Input capture 4 prescaler [1:0] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4PSC0` ((uint8\_t) (0x01 << 2))
- TIM1 Input capture 4 prescaler [0] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4PSC1` ((uint8\_t) (0x01 << 3))
- TIM1 Input capture 4 prescaler [1] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4F` ((uint8\_t) (0x0F << 4))
- TIM1 Output compare 4 mode [3:0] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4F0` ((uint8\_t) (0x01 << 4))
- TIM1 Input capture 4 filter [0] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4F1` ((uint8\_t) (0x01 << 5))
- TIM1 Input capture 4 filter [1] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4F2` ((uint8\_t) (0x01 << 6))
- TIM1 Input capture 4 filter [2] (in \_TIM1\_CCMR4)*
- #define `_TIM1_IC4F3` ((uint8\_t) (0x01 << 7))
- TIM1 Input capture 4 filter [3] (in \_TIM1\_CCMR4)*
- #define `_TIM1_CC1E` ((uint8\_t) (0x01 << 0))
- TIM1 Capture/compare 1 output enable [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC1P` ((uint8\_t) (0x01 << 1))
- TIM1 Capture/compare 1 output polarity [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC1NE` ((uint8\_t) (0x01 << 2))
- TIM1 Capture/compare 1 complementary output enable [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC1NP` ((uint8\_t) (0x01 << 3))
- TIM1 Capture/compare 1 complementary output polarity [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC2E` ((uint8\_t) (0x01 << 4))
- TIM1 Capture/compare 2 output enable [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC2P` ((uint8\_t) (0x01 << 5))
- TIM1 Capture/compare 2 output polarity [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC2NE` ((uint8\_t) (0x01 << 6))
- TIM1 Capture/compare 2 complementary output enable [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC2NP` ((uint8\_t) (0x01 << 7))
- TIM1 Capture/compare 2 complementary output polarity [0] (in \_TIM1\_CCER1)*
- #define `_TIM1_CC3E` ((uint8\_t) (0x01 << 0))
- TIM1 Capture/compare 3 output enable [0] (in \_TIM1\_CCER2)*
- #define `_TIM1_CC3P` ((uint8\_t) (0x01 << 1))
- TIM1 Capture/compare 3 output polarity [0] (in \_TIM1\_CCER2)*
- #define `_TIM1_CC3NE` ((uint8\_t) (0x01 << 2))
- TIM1 Capture/compare 3 complementary output enable [0] (in \_TIM1\_CCER2)*
- #define `_TIM1_CC3NP` ((uint8\_t) (0x01 << 3))
- TIM1 Capture/compare 3 complementary output polarity [0] (in \_TIM1\_CCER2)*
- #define `_TIM1_CC4E` ((uint8\_t) (0x01 << 4))
- TIM1 Capture/compare 4 output enable [0] (in \_TIM1\_CCER2)*
- #define `_TIM1_CC4P` ((uint8\_t) (0x01 << 5))
- TIM1 Capture/compare 4 output polarity [0] (in \_TIM1\_CCER2)*
- #define `_TIM1_LOCK` ((uint8\_t) (0x03 << 0))
- TIM1 Lock configuration [1:0] (in \_TIM1\_BKR)*
- #define `_TIM1_LOCK0` ((uint8\_t) (0x01 << 0))
- TIM1 Lock configuration [0] (in \_TIM1\_BKR)*

- `#define _TIM1_LOCK1 ((uint8_t) (0x01 << 1))`  
*TIM1 Lock configuration [1] (in \_TIM1\_BKR)*
- `#define _TIM1_OSSI ((uint8_t) (0x01 << 2))`  
*TIM1 Off state selection for idle mode [0] (in \_TIM1\_BKR)*
- `#define _TIM1_OSSR ((uint8_t) (0x01 << 3))`  
*TIM1 Off state selection for Run mode [0] (in \_TIM1\_BKR)*
- `#define _TIM1_BKE ((uint8_t) (0x01 << 4))`  
*TIM1 Break enable [0] (in \_TIM1\_BKR)*
- `#define _TIM1_BKP ((uint8_t) (0x01 << 5))`  
*TIM1 Break polarity [0] (in \_TIM1\_BKR)*
- `#define _TIM1_AOE ((uint8_t) (0x01 << 6))`  
*TIM1 Automatic output enable [0] (in \_TIM1\_BKR)*
- `#define _TIM1_MOE ((uint8_t) (0x01 << 7))`  
*TIM1 Main output enable [0] (in \_TIM1\_BKR)*
- `#define _TIM1_OIS1 ((uint8_t) (0x01 << 0))`  
*TIM1 Output idle state 1 (OC1 output) [0] (in \_TIM1\_OISR)*
- `#define _TIM1_OIS1N ((uint8_t) (0x01 << 1))`  
*TIM1 Output idle state 1 (OC1N output) [0] (in \_TIM1\_OISR)*
- `#define _TIM1_OIS2 ((uint8_t) (0x01 << 2))`  
*TIM1 Output idle state 2 (OC2 output) [0] (in \_TIM1\_OISR)*
- `#define _TIM1_OIS2N ((uint8_t) (0x01 << 3))`  
*TIM1 Output idle state 2 (OC2N output) [0] (in \_TIM1\_OISR)*
- `#define _TIM1_OIS3 ((uint8_t) (0x01 << 4))`  
*TIM1 Output idle state 3 (OC3 output) [0] (in \_TIM1\_OISR)*
- `#define _TIM1_OIS3N ((uint8_t) (0x01 << 5))`  
*TIM1 Output idle state 3 (OC3N output) [0] (in \_TIM1\_OISR)*
- `#define _TIM1_OIS4 ((uint8_t) (0x01 << 6))`  
*TIM1 Output idle state 4 (OC4 output) [0] (in \_TIM1\_OISR)*
- `#define _TIM2_SFR(TIM2_t, TIM2_AddressBase)`  
*TIM2 struct/bit access.*
- `#define _TIM2_CR1_SFR(uint8_t, TIM2_AddressBase+0x00)`  
*TIM2 control register 1.*
- `#define _TIM2_IER_SFR(uint8_t, TIM2_AddressBase+0x01)`  
*TIM2 interrupt enable register.*
- `#define _TIM2_SR1_SFR(uint8_t, TIM2_AddressBase+0x02)`  
*TIM2 status register 1.*
- `#define _TIM2_SR2_SFR(uint8_t, TIM2_AddressBase+0x03)`  
*TIM2 status register 2.*
- `#define _TIM2_EGR_SFR(uint8_t, TIM2_AddressBase+0x04)`  
*TIM2 Event generation register.*
- `#define _TIM2_CCMR1_SFR(uint8_t, TIM2_AddressBase+0x05)`  
*TIM2 Capture/compare mode register 1.*
- `#define _TIM2_CCMR2_SFR(uint8_t, TIM2_AddressBase+0x06)`  
*TIM2 Capture/compare mode register 2.*
- `#define _TIM2_CCMR3_SFR(uint8_t, TIM2_AddressBase+0x07)`  
*TIM2 Capture/compare mode register 3.*
- `#define _TIM2_CCER1_SFR(uint8_t, TIM2_AddressBase+0x08)`  
*TIM2 Capture/compare enable register 1.*
- `#define _TIM2_CCER2_SFR(uint8_t, TIM2_AddressBase+0x09)`  
*TIM2 Capture/compare enable register 2.*
- `#define _TIM2_CNTRH_SFR(uint8_t, TIM2_AddressBase+0x0A)`

- TIM2 counter register high byte.*
- #define `_TIM2_CNTRL_SFR`(uint8\_t, `TIM2_AddressBase+0x0B`)
- TIM2 counter register low byte.*
- #define `_TIM2_PSCR_SFR`(uint8\_t, `TIM2_AddressBase+0x0C`)
- TIM2 clock prescaler register.*
- #define `_TIM2_ARRH_SFR`(uint8\_t, `TIM2_AddressBase+0x0D`)
- TIM2 auto-reload register high byte.*
- #define `_TIM2_ARRL_SFR`(uint8\_t, `TIM2_AddressBase+0x0E`)
- TIM2 auto-reload register low byte.*
- #define `_TIM2_CCR1H_SFR`(uint8\_t, `TIM2_AddressBase+0x0F`)
- TIM2 16-bit capture/compare value 1 high byte.*
- #define `_TIM2_CCR1L_SFR`(uint8\_t, `TIM2_AddressBase+0x10`)
- TIM2 16-bit capture/compare value 1 low byte.*
- #define `_TIM2_CCR2H_SFR`(uint8\_t, `TIM2_AddressBase+0x11`)
- TIM2 16-bit capture/compare value 2 high byte.*
- #define `_TIM2_CCR2L_SFR`(uint8\_t, `TIM2_AddressBase+0x12`)
- TIM2 16-bit capture/compare value 2 low byte.*
- #define `_TIM2_CCR3H_SFR`(uint8\_t, `TIM2_AddressBase+0x13`)
- TIM2 16-bit capture/compare value 3 high byte.*
- #define `_TIM2_CCR3L_SFR`(uint8\_t, `TIM2_AddressBase+0x14`)
- TIM2 16-bit capture/compare value 3 low byte.*
- #define `_TIM2_CR1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 control register 1 reset value.*
- #define `_TIM2_IER_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 interrupt enable register reset value.*
- #define `_TIM2_SR1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 status register 1 reset value.*
- #define `_TIM2_SR2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 status register 2 reset value.*
- #define `_TIM2_EGR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 Event generation register reset value.*
- #define `_TIM2_CCMR1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 Capture/compare mode register 1 reset value.*
- #define `_TIM2_CCMR2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 Capture/compare mode register 2 reset value.*
- #define `_TIM2_CCMR3_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 Capture/compare mode register 3 reset value.*
- #define `_TIM2_CCER1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 Capture/compare enable register 1 reset value.*
- #define `_TIM2_CCER2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 Capture/compare enable register 2 reset value.*
- #define `_TIM2_CNTRH_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 counter register high byte reset value.*
- #define `_TIM2_CNTRL_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 counter register low byte reset value.*
- #define `_TIM2_PSCR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM2 clock prescaler register reset value.*
- #define `_TIM2_ARRH_RESET_VALUE` ((uint8\_t) 0xFF)
- TIM2 auto-reload register high byte reset value.*
- #define `_TIM2_ARRL_RESET_VALUE` ((uint8\_t) 0xFF)
- TIM2 auto-reload register low byte reset value.*

- `#define _TIM2_CCR1H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 1 high byte reset value.*
- `#define _TIM2_CCR1L_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 1 low byte reset value.*
- `#define _TIM2_CCR2H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 2 high byte reset value.*
- `#define _TIM2_CCR2L_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 2 low byte reset value.*
- `#define _TIM2_CCR3H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 3 high byte reset value.*
- `#define _TIM2_CCR3L_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 3 low byte reset value.*
- `#define _TIM2_CEN ((uint8_t) (0x01 << 0))`  
*TIM2 Counter enable [0] (in \_TIM2\_CR1)*
- `#define _TIM2_UDIS ((uint8_t) (0x01 << 1))`  
*TIM2 Update disable [0] (in \_TIM2\_CR1)*
- `#define _TIM2_URS ((uint8_t) (0x01 << 2))`  
*TIM2 Update request source [0] (in \_TIM2\_CR1)*
- `#define _TIM2_OPM ((uint8_t) (0x01 << 3))`  
*TIM2 One-pulse mode [0] (in \_TIM2\_CR1)*
- `#define _TIM2_ARPE ((uint8_t) (0x01 << 7))`  
*TIM2 Auto-reload preload enable [0] (in \_TIM2\_CR1)*
- `#define _TIM2_UIE ((uint8_t) (0x01 << 0))`  
*TIM2 Update interrupt enable [0] (in \_TIM2\_IER)*
- `#define _TIM2_CC1IE ((uint8_t) (0x01 << 1))`  
*TIM2 Capture/compare 1 interrupt enable [0] (in \_TIM2\_IER)*
- `#define _TIM2_CC2IE ((uint8_t) (0x01 << 2))`  
*TIM2 Capture/compare 2 interrupt enable [0] (in \_TIM2\_IER)*
- `#define _TIM2_CC3IE ((uint8_t) (0x01 << 3))`  
*TIM2 Capture/compare 3 interrupt enable [0] (in \_TIM2\_IER)*
- `#define _TIM2_UIF ((uint8_t) (0x01 << 0))`  
*TIM2 Update interrupt flag [0] (in \_TIM2\_SR1)*
- `#define _TIM2_CC1IF ((uint8_t) (0x01 << 1))`  
*TIM2 Capture/compare 1 interrupt flag [0] (in \_TIM2\_SR1)*
- `#define _TIM2_CC2IF ((uint8_t) (0x01 << 2))`  
*TIM2 Capture/compare 2 interrupt flag [0] (in \_TIM2\_SR1)*
- `#define _TIM2_CC3IF ((uint8_t) (0x01 << 3))`  
*TIM2 Capture/compare 3 interrupt flag [0] (in \_TIM2\_SR1)*
- `#define _TIM2_CC1OF ((uint8_t) (0x01 << 1))`  
*TIM2 Capture/compare 1 overcapture flag [0] (in \_TIM2\_SR2)*
- `#define _TIM2_CC2OF ((uint8_t) (0x01 << 2))`  
*TIM2 Capture/compare 2 overcapture flag [0] (in \_TIM2\_SR2)*
- `#define _TIM2_CC3OF ((uint8_t) (0x01 << 3))`  
*TIM2 Capture/compare 3 overcapture flag [0] (in \_TIM2\_SR2)*
- `#define _TIM2_UG ((uint8_t) (0x01 << 0))`  
*TIM2 Update generation [0] (in \_TIM2\_EGR)*
- `#define _TIM2_CC1G ((uint8_t) (0x01 << 1))`  
*TIM2 Capture/compare 1 generation [0] (in \_TIM2\_EGR)*
- `#define _TIM2_CC2G ((uint8_t) (0x01 << 2))`  
*TIM2 Capture/compare 2 generation [0] (in \_TIM2\_EGR)*
- `#define _TIM2_CC3G ((uint8_t) (0x01 << 3))`

- TIM2 Capture/compare 3 generation [0] (in \_TIM2\_EGR)*
- #define `_TIM2_CC1S` ((uint8\_t) (0x03 << 0))
- TIM2 Compare 1 selection [1:0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_CC1S0` ((uint8\_t) (0x01 << 0))
- TIM2 Compare 1 selection [0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_CC1S1` ((uint8\_t) (0x01 << 1))
- TIM2 Compare 1 selection [1] (in \_TIM2\_CCMR1)*
- #define `_TIM2_OC1PE` ((uint8\_t) (0x01 << 3))
- TIM2 Output compare 1 preload enable [0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_OC1M` ((uint8\_t) (0x07 << 4))
- TIM2 Output compare 1 mode [2:0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_OC1M0` ((uint8\_t) (0x01 << 4))
- TIM2 Output compare 1 mode [0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_OC1M1` ((uint8\_t) (0x01 << 5))
- TIM2 Output compare 1 mode [1] (in \_TIM2\_CCMR1)*
- #define `_TIM2_OC1M2` ((uint8\_t) (0x01 << 6))
- TIM2 Output compare 1 mode [2] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1PSC` ((uint8\_t) (0x03 << 2))
- TIM2 Input capture 1 prescaler [1:0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1PSC0` ((uint8\_t) (0x01 << 2))
- TIM2 Input capture 1 prescaler [0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1PSC1` ((uint8\_t) (0x01 << 3))
- TIM2 Input capture 1 prescaler [1] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1F` ((uint8\_t) (0x0F << 4))
- TIM2 Output compare 1 mode [3:0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1F0` ((uint8\_t) (0x01 << 4))
- TIM2 Output compare 1 mode [0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1F1` ((uint8\_t) (0x01 << 5))
- TIM2 Output compare 1 mode [1] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1F2` ((uint8\_t) (0x01 << 6))
- TIM2 Output compare 1 mode [2] (in \_TIM2\_CCMR1)*
- #define `_TIM2_IC1F3` ((uint8\_t) (0x01 << 7))
- TIM2 Output compare 1 mode [3] (in \_TIM2\_CCMR1)*
- #define `_TIM2_CC2S` ((uint8\_t) (0x03 << 0))
- TIM2 Compare 2 selection [1:0] (in \_TIM2\_CCMR2)*
- #define `_TIM2_CC2S0` ((uint8\_t) (0x01 << 0))
- TIM2 Compare 2 selection [0] (in \_TIM2\_CCMR2)*
- #define `_TIM2_CC2S1` ((uint8\_t) (0x01 << 1))
- TIM2 Compare 2 selection [1] (in \_TIM2\_CCMR2)*
- #define `_TIM2_OC2PE` ((uint8\_t) (0x01 << 3))
- TIM2 Output compare 2 preload enable [0] (in \_TIM2\_CCMR2)*
- #define `_TIM2_OC2M` ((uint8\_t) (0x07 << 4))
- TIM2 Output compare 2 mode [2:0] (in \_TIM2\_CCMR2)*
- #define `_TIM2_OC2M0` ((uint8\_t) (0x01 << 4))
- TIM2 Output compare 2 mode [0] (in \_TIM2\_CCMR2)*
- #define `_TIM2_OC2M1` ((uint8\_t) (0x01 << 5))
- TIM2 Output compare 2 mode [1] (in \_TIM2\_CCMR2)*
- #define `_TIM2_OC2M2` ((uint8\_t) (0x01 << 6))
- TIM2 Output compare 2 mode [2] (in \_TIM2\_CCMR2)*
- #define `_TIM2_IC2PSC` ((uint8\_t) (0x03 << 2))
- TIM2 Input capture 2 prescaler [1:0] (in \_TIM2\_CCMR2)*



- `#define _TIM2_IC2PSC0` ((uint8\_t) (0x01 << 2))  
TIM2 Input capture 2 prescaler [0] (in \_TIM2\_CCMR2)
- `#define _TIM2_IC2PSC1` ((uint8\_t) (0x01 << 3))  
TIM2 Input capture 2 prescaler [1] (in \_TIM2\_CCMR2)
- `#define _TIM2_IC2F` ((uint8\_t) (0x0F << 4))  
TIM2 Output compare 2 mode [3:0] (in \_TIM2\_CCMR2)
- `#define _TIM2_IC2F0` ((uint8\_t) (0x01 << 4))  
TIM2 Output compare 2 mode [0] (in \_TIM2\_CCMR2)
- `#define _TIM2_IC2F1` ((uint8\_t) (0x01 << 5))  
TIM2 Output compare 2 mode [1] (in \_TIM2\_CCMR2)
- `#define _TIM2_IC2F2` ((uint8\_t) (0x01 << 6))  
TIM2 Output compare 2 mode [2] (in \_TIM2\_CCMR2)
- `#define _TIM2_IC2F3` ((uint8\_t) (0x01 << 7))  
TIM2 Output compare 2 mode [3] (in \_TIM2\_CCMR2)
- `#define _TIM2_CC3S` ((uint8\_t) (0x03 << 0))  
TIM2 Compare 3 selection [1:0] (in \_TIM2\_CCMR3)
- `#define _TIM2_CC3S0` ((uint8\_t) (0x01 << 0))  
TIM2 Compare 3 selection [0] (in \_TIM2\_CCMR3)
- `#define _TIM2_CC3S1` ((uint8\_t) (0x01 << 1))  
TIM2 Compare 3 selection [1] (in \_TIM2\_CCMR3)
- `#define _TIM2_OC3PE` ((uint8\_t) (0x01 << 3))  
TIM2 Output compare 3 preload enable [0] (in \_TIM2\_CCMR3)
- `#define _TIM2_OC3M` ((uint8\_t) (0x07 << 4))  
TIM2 Output compare 3 mode [2:0] (in \_TIM2\_CCMR3)
- `#define _TIM2_OC3M0` ((uint8\_t) (0x01 << 4))  
TIM2 Output compare 3 mode [0] (in \_TIM2\_CCMR3)
- `#define _TIM2_OC3M1` ((uint8\_t) (0x01 << 5))  
TIM2 Output compare 3 mode [1] (in \_TIM2\_CCMR3)
- `#define _TIM2_OC3M2` ((uint8\_t) (0x01 << 6))  
TIM2 Output compare 3 mode [2] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3PSC` ((uint8\_t) (0x03 << 2))  
TIM2 Input capture 3 prescaler [1:0] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3PSC0` ((uint8\_t) (0x01 << 2))  
TIM2 Input capture 3 prescaler [0] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3PSC1` ((uint8\_t) (0x01 << 3))  
TIM2 Input capture 3 prescaler [1] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3F` ((uint8\_t) (0x0F << 4))  
TIM2 Output compare 3 mode [3:0] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3F0` ((uint8\_t) (0x01 << 4))  
TIM2 Output compare 3 mode [0] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3F1` ((uint8\_t) (0x01 << 5))  
TIM2 Output compare 3 mode [1] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3F2` ((uint8\_t) (0x01 << 6))  
TIM2 Output compare 3 mode [2] (in \_TIM2\_CCMR3)
- `#define _TIM2_IC3F3` ((uint8\_t) (0x01 << 7))  
TIM2 Output compare 3 mode [3] (in \_TIM2\_CCMR3)
- `#define _TIM2_CC1E` ((uint8\_t) (0x01 << 0))  
TIM2 Capture/compare 1 output enable [0] (in \_TIM2\_CCER1)
- `#define _TIM2_CC1P` ((uint8\_t) (0x01 << 1))  
TIM2 Capture/compare 1 output polarity [0] (in \_TIM2\_CCER1)
- `#define _TIM2_CC2E` ((uint8\_t) (0x01 << 4))



- TIM2 Capture/compare 2 output enable [0] (in \_TIM2\_CCER1)*
- #define `_TIM2_CC2P` ((uint8\_t) (0x01 << 5))
- TIM2 Capture/compare 2 output polarity [0] (in \_TIM2\_CCER1)*
- #define `_TIM2_CC3E` ((uint8\_t) (0x01 << 0))
- TIM2 Capture/compare 3 output enable [0] (in \_TIM2\_CCER2)*
- #define `_TIM2_CC3P` ((uint8\_t) (0x01 << 1))
- TIM2 Capture/compare 3 output polarity [0] (in \_TIM2\_CCER2)*
- #define `_TIM2_PSC` ((uint8\_t) (0x0F << 0))
- TIM2 prescaler [3:0] (in \_TIM2\_PSCR)*
- #define `_TIM2_PSC0` ((uint8\_t) (0x01 << 0))
- TIM2 prescaler [0] (in \_TIM2\_PSCR)*
- #define `_TIM2_PSC1` ((uint8\_t) (0x01 << 1))
- TIM2 prescaler [1] (in \_TIM2\_PSCR)*
- #define `_TIM2_PSC2` ((uint8\_t) (0x01 << 2))
- TIM2 prescaler [2] (in \_TIM2\_PSCR)*
- #define `_TIM2_PSC3` ((uint8\_t) (0x01 << 3))
- TIM2 prescaler [3] (in \_TIM2\_PSCR)*
- #define `_TIM3_SFR`(TIM3\_t, TIM3\_AddressBase)
- TIM3 struct/bit access.*
- #define `_TIM3_CR1_SFR`(uint8\_t, TIM3\_AddressBase+0x00)
- TIM3 control register 1.*
- #define `_TIM3_IER_SFR`(uint8\_t, TIM3\_AddressBase+0x01)
- TIM3 interrupt enable register.*
- #define `_TIM3_SR1_SFR`(uint8\_t, TIM3\_AddressBase+0x02)
- TIM3 status register 1.*
- #define `_TIM3_SR2_SFR`(uint8\_t, TIM3\_AddressBase+0x03)
- TIM3 status register 2.*
- #define `_TIM3_EGR_SFR`(uint8\_t, TIM3\_AddressBase+0x04)
- TIM3 Event generation register.*
- #define `_TIM3_CCMR1_SFR`(uint8\_t, TIM3\_AddressBase+0x05)
- TIM3 Capture/compare mode register 1.*
- #define `_TIM3_CCMR2_SFR`(uint8\_t, TIM3\_AddressBase+0x06)
- TIM3 Capture/compare mode register 2.*
- #define `_TIM3_CCER1_SFR`(uint8\_t, TIM3\_AddressBase+0x08)
- TIM3 Capture/compare enable register 1.*
- #define `_TIM3_CNTRH_SFR`(uint8\_t, TIM3\_AddressBase+0x0A)
- TIM3 counter register high byte.*
- #define `_TIM3_CNTRL_SFR`(uint8\_t, TIM3\_AddressBase+0x0B)
- TIM3 counter register low byte.*
- #define `_TIM3_PSCR_SFR`(uint8\_t, TIM3\_AddressBase+0x0C)
- TIM3 clock prescaler register.*
- #define `_TIM3_ARRH_SFR`(uint8\_t, TIM3\_AddressBase+0x0D)
- TIM3 auto-reload register high byte.*
- #define `_TIM3_ARRL_SFR`(uint8\_t, TIM3\_AddressBase+0x0E)
- TIM3 auto-reload register low byte.*
- #define `_TIM3_CCR1H_SFR`(uint8\_t, TIM3\_AddressBase+0x0F)
- TIM3 16-bit capture/compare value 1 high byte.*
- #define `_TIM3_CCR1L_SFR`(uint8\_t, TIM3\_AddressBase+0x10)
- TIM3 16-bit capture/compare value 1 low byte.*
- #define `_TIM3_CCR2H_SFR`(uint8\_t, TIM3\_AddressBase+0x11)
- TIM3 16-bit capture/compare value 2 high byte.*

- `#define _TIM3_CCR2L_SFR(uint8_t, TIM3_AddressBase+0x12)`  
*TIM3 16-bit capture/compare value 2 low byte.*
- `#define _TIM3_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 control register 1 reset value.*
- `#define _TIM3_IER_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 interrupt enable register reset value.*
- `#define _TIM3_SR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 status register 1 reset value.*
- `#define _TIM3_SR2_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 status register 2 reset value.*
- `#define _TIM3_EGR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 Event generation register reset value.*
- `#define _TIM3_CCMR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 Capture/compare mode register 1 reset value.*
- `#define _TIM3_CCMR2_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 Capture/compare mode register 2 reset value.*
- `#define _TIM3_CCER1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 Capture/compare enable register 1 reset value.*
- `#define _TIM3_CNTRH_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 counter register high byte reset value.*
- `#define _TIM3_CNTRL_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 counter register low byte reset value.*
- `#define _TIM3_PSCR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 clock prescaler register reset value.*
- `#define _TIM3_ARRH_RESET_VALUE ((uint8_t) 0xFF)`  
*TIM3 auto-reload register high byte reset value.*
- `#define _TIM3_ARRL_RESET_VALUE ((uint8_t) 0xFF)`  
*TIM3 auto-reload register low byte reset value.*
- `#define _TIM3_CCR1H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 16-bit capture/compare value 1 high byte reset value.*
- `#define _TIM3_CCR1L_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 16-bit capture/compare value 1 low byte reset value.*
- `#define _TIM3_CCR2H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 16-bit capture/compare value 2 high byte reset value.*
- `#define _TIM3_CCR2L_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 16-bit capture/compare value 2 low byte reset value.*
- `#define _TIM3_CEN ((uint8_t) (0x01 << 0))`  
*TIM3 Counter enable [0] (in \_TIM3\_CR1)*
- `#define _TIM3_UDIS ((uint8_t) (0x01 << 1))`  
*TIM3 Update disable [0] (in \_TIM3\_CR1)*
- `#define _TIM3_URS ((uint8_t) (0x01 << 2))`  
*TIM3 Update request source [0] (in \_TIM3\_CR1)*
- `#define _TIM3_OPM ((uint8_t) (0x01 << 3))`  
*TIM3 One-pulse mode [0] (in \_TIM3\_CR1)*
- `#define _TIM3_ARPE ((uint8_t) (0x01 << 7))`  
*TIM3 Auto-reload preload enable [0] (in \_TIM3\_CR1)*
- `#define _TIM3_UIE ((uint8_t) (0x01 << 0))`  
*TIM3 Update interrupt enable [0] (in \_TIM3\_IER)*
- `#define _TIM3_CC1IE ((uint8_t) (0x01 << 1))`  
*TIM3 Capture/compare 1 interrupt enable [0] (in \_TIM3\_IER)*
- `#define _TIM3_CC2IE ((uint8_t) (0x01 << 2))`

- TIM3 Capture/compare 2 interrupt enable [0] (in \_TIM3\_IER)*
- #define `_TIM3_UIF` ((uint8\_t) (0x01 << 0))
- TIM3 Update interrupt flag [0] (in \_TIM3\_SR1)*
- #define `_TIM3_CC1IF` ((uint8\_t) (0x01 << 1))
- TIM3 Capture/compare 1 interrupt flag [0] (in \_TIM3\_SR1)*
- #define `_TIM3_CC2IF` ((uint8\_t) (0x01 << 2))
- TIM3 Capture/compare 2 interrupt flag [0] (in \_TIM3\_SR1)*
- #define `_TIM3_CC1OF` ((uint8\_t) (0x01 << 1))
- TIM3 Capture/compare 1 overcapture flag [0] (in \_TIM3\_SR2)*
- #define `_TIM3_CC2OF` ((uint8\_t) (0x01 << 2))
- TIM3 Capture/compare 2 overcapture flag [0] (in \_TIM3\_SR2)*
- #define `_TIM3_UG` ((uint8\_t) (0x01 << 0))
- TIM3 Update generation [0] (in \_TIM3\_EGR)*
- #define `_TIM3_CC1G` ((uint8\_t) (0x01 << 1))
- TIM3 Capture/compare 1 generation [0] (in \_TIM3\_EGR)*
- #define `_TIM3_CC2G` ((uint8\_t) (0x01 << 2))
- TIM3 Capture/compare 2 generation [0] (in \_TIM3\_EGR)*
- #define `_TIM3_CC1S` ((uint8\_t) (0x03 << 0))
- TIM3 Compare 1 selection [1:0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_CC1S0` ((uint8\_t) (0x01 << 0))
- TIM3 Compare 1 selection [0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_CC1S1` ((uint8\_t) (0x01 << 1))
- TIM3 Compare 1 selection [1] (in \_TIM3\_CCMR1)*
- #define `_TIM3_OC1PE` ((uint8\_t) (0x01 << 3))
- TIM3 Output compare 1 preload enable [0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_OC1M` ((uint8\_t) (0x07 << 4))
- TIM3 Output compare 1 mode [2:0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_OC1M0` ((uint8\_t) (0x01 << 4))
- TIM3 Output compare 1 mode [0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_OC1M1` ((uint8\_t) (0x01 << 5))
- TIM3 Output compare 1 mode [1] (in \_TIM3\_CCMR1)*
- #define `_TIM3_OC1M2` ((uint8\_t) (0x01 << 6))
- TIM3 Output compare 1 mode [2] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1PSC` ((uint8\_t) (0x03 << 2))
- TIM3 Input capture 1 prescaler [1:0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1PSC0` ((uint8\_t) (0x01 << 2))
- TIM3 Input capture 1 prescaler [0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1PSC1` ((uint8\_t) (0x01 << 3))
- TIM3 Input capture 1 prescaler [1] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1F` ((uint8\_t) (0x0F << 4))
- TIM3 Output compare 1 mode [3:0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1F0` ((uint8\_t) (0x01 << 4))
- TIM3 Output compare 1 mode [0] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1F1` ((uint8\_t) (0x01 << 5))
- TIM3 Output compare 1 mode [1] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1F2` ((uint8\_t) (0x01 << 6))
- TIM3 Output compare 1 mode [2] (in \_TIM3\_CCMR1)*
- #define `_TIM3_IC1F3` ((uint8\_t) (0x01 << 7))
- TIM3 Output compare 1 mode [3] (in \_TIM3\_CCMR1)*
- #define `_TIM3_CC2S` ((uint8\_t) (0x03 << 0))
- TIM3 Compare 2 selection [1:0] (in \_TIM3\_CCMR2)*

- `#define _TIM3_CC2S0` ((uint8\_t) (0x01 << 0))  
*TIM3 Compare 2 selection [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_CC2S1` ((uint8\_t) (0x01 << 1))  
*TIM3 Compare 2 selection [1] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2PE` ((uint8\_t) (0x01 << 3))  
*TIM3 Output compare 2 preload enable [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2M` ((uint8\_t) (0x07 << 4))  
*TIM3 Output compare 2 mode [2:0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2M0` ((uint8\_t) (0x01 << 4))  
*TIM3 Output compare 2 mode [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2M1` ((uint8\_t) (0x01 << 5))  
*TIM3 Output compare 2 mode [1] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2M2` ((uint8\_t) (0x01 << 6))  
*TIM3 Output compare 2 mode [2] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2PSC` ((uint8\_t) (0x03 << 2))  
*TIM3 Input capture 2 prescaler [1:0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2PSC0` ((uint8\_t) (0x01 << 2))  
*TIM3 Input capture 2 prescaler [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2PSC1` ((uint8\_t) (0x01 << 3))  
*TIM3 Input capture 2 prescaler [1] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2F` ((uint8\_t) (0x0F << 4))  
*TIM3 Output compare 2 mode [3:0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2F0` ((uint8\_t) (0x01 << 4))  
*TIM3 Output compare 2 mode [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2F1` ((uint8\_t) (0x01 << 5))  
*TIM3 Output compare 2 mode [1] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2F2` ((uint8\_t) (0x01 << 6))  
*TIM3 Output compare 2 mode [2] (in \_TIM3\_CCMR2)*
- `#define _TIM3_IC2F3` ((uint8\_t) (0x01 << 7))  
*TIM3 Output compare 2 mode [3] (in \_TIM3\_CCMR2)*
- `#define _TIM3_CC1E` ((uint8\_t) (0x01 << 0))  
*TIM3 Capture/compare 1 output enable [0] (in \_TIM3\_CCER1)*
- `#define _TIM3_CC1P` ((uint8\_t) (0x01 << 1))  
*TIM3 Capture/compare 1 output polarity [0] (in \_TIM3\_CCER1)*
- `#define _TIM3_CC2E` ((uint8\_t) (0x01 << 4))  
*TIM3 Capture/compare 2 output enable [0] (in \_TIM3\_CCER1)*
- `#define _TIM3_CC2P` ((uint8\_t) (0x01 << 5))  
*TIM3 Capture/compare 2 output polarity [0] (in \_TIM3\_CCER1)*
- `#define _TIM3_PSC` ((uint8\_t) (0x0F << 0))  
*TIM3 clock prescaler [3:0] (in \_TIM3\_PSCR)*
- `#define _TIM3_PSC0` ((uint8\_t) (0x01 << 0))  
*TIM3 clock prescaler [0] (in \_TIM3\_PSCR)*
- `#define _TIM3_PSC1` ((uint8\_t) (0x01 << 1))  
*TIM3 clock prescaler [1] (in \_TIM3\_PSCR)*
- `#define _TIM3_PSC2` ((uint8\_t) (0x01 << 2))  
*TIM3 clock prescaler [2] (in \_TIM3\_PSCR)*
- `#define _TIM3_PSC3` ((uint8\_t) (0x01 << 3))  
*TIM3 clock prescaler [3] (in \_TIM3\_PSCR)*
- `#define _TIM4_SFR`(TIM4\_t, TIM4\_AddressBase)  
*TIM4 struct/bit access.*
- `#define _TIM4_CR_SFR`(uint8\_t, TIM4\_AddressBase+0x00)

- TIM4 control register.*
- #define `_TIM4_IER_SFR`(uint8\_t, `TIM4_AddressBase`+0x01)
- TIM4 interrupt enable register.*
- #define `_TIM4_SR_SFR`(uint8\_t, `TIM4_AddressBase`+0x02)
- TIM4 status register.*
- #define `_TIM4_EGR_SFR`(uint8\_t, `TIM4_AddressBase`+0x03)
- TIM4 event generation register.*
- #define `_TIM4_CNTR_SFR`(uint8\_t, `TIM4_AddressBase`+0x04)
- TIM4 counter register.*
- #define `_TIM4_PSCR_SFR`(uint8\_t, `TIM4_AddressBase`+0x05)
- TIM4 clock prescaler register.*
- #define `_TIM4_ARR_SFR`(uint8\_t, `TIM4_AddressBase`+0x06)
- TIM4 auto-reload register.*
- #define `_TIM4_CR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM4 control register reset value.*
- #define `_TIM4_IER_RESET_VALUE` ((uint8\_t) 0x00)
- TIM4 interrupt enable register reset value.*
- #define `_TIM4_SR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM4 status register reset value.*
- #define `_TIM4_EGR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM4 event generation register reset value.*
- #define `_TIM4_CNTR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM4 counter register reset value.*
- #define `_TIM4_PSCR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM4 clock prescaler register reset value.*
- #define `_TIM4_ARR_RESET_VALUE` ((uint8\_t) 0xFF)
- TIM4 auto-reload register reset value.*
- #define `_TIM4_CEN` ((uint8\_t) (0x01 << 0))
- TIM4 Counter enable [0] (in \_TIM4\_CR)*
- #define `_TIM4_UDIS` ((uint8\_t) (0x01 << 1))
- TIM4 Update disable [0] (in \_TIM4\_CR)*
- #define `_TIM4_URS` ((uint8\_t) (0x01 << 2))
- TIM4 Update request source [0] (in \_TIM4\_CR)*
- #define `_TIM4_OPM` ((uint8\_t) (0x01 << 3))
- TIM4 One-pulse mode [0] (in \_TIM4\_CR)*
- #define `_TIM4_ARPE` ((uint8\_t) (0x01 << 7))
- TIM4 Auto-reload preload enable [0] (in \_TIM4\_CR)*
- #define `_TIM4_UIE` ((uint8\_t) (0x01 << 0))
- TIM4 Update interrupt enable [0] (in \_TIM4\_IER)*
- #define `_TIM4_UIF` ((uint8\_t) (0x01 << 0))
- TIM4 Update interrupt flag [0] (in \_TIM4\_SR)*
- #define `_TIM4_UG` ((uint8\_t) (0x01 << 0))
- TIM4 Update generation [0] (in \_TIM4\_EGR)*
- #define `_TIM4_PSC` ((uint8\_t) (0x07 << 0))
- TIM4 clock prescaler [2:0] (in \_TIM4\_PSCR)*
- #define `_TIM4_PSC0` ((uint8\_t) (0x01 << 0))
- TIM4 clock prescaler [0] (in \_TIM4\_PSCR)*
- #define `_TIM4_PSC1` ((uint8\_t) (0x01 << 1))
- TIM4 clock prescaler [1] (in \_TIM4\_PSCR)*
- #define `_TIM4_PSC2` ((uint8\_t) (0x01 << 2))
- TIM4 clock prescaler [2] (in \_TIM4\_PSCR)*

- `#define _TIM5_SFR(TIM5_t, TIM5_AddressBase)`  
*TIM5 struct/bit access.*
- `#define _TIM5_CR1_SFR(uint8_t, TIM5_AddressBase+0x00)`  
*TIM5 control register 1.*
- `#define _TIM5_CR2_SFR(uint8_t, TIM5_AddressBase+0x01)`  
*TIM5 control register 2.*
- `#define _TIM5_SMCR_SFR(uint8_t, TIM5_AddressBase+0x02)`  
*TIM5 Slave mode control register.*
- `#define _TIM5_IER_SFR(uint8_t, TIM5_AddressBase+0x03)`  
*TIM5 interrupt enable register.*
- `#define _TIM5_SR1_SFR(uint8_t, TIM5_AddressBase+0x04)`  
*TIM5 status register 1.*
- `#define _TIM5_SR2_SFR(uint8_t, TIM5_AddressBase+0x05)`  
*TIM5 status register 2.*
- `#define _TIM5_EGR_SFR(uint8_t, TIM5_AddressBase+0x06)`  
*TIM5 Event generation register.*
- `#define _TIM5_CCMR1_SFR(uint8_t, TIM5_AddressBase+0x07)`  
*TIM5 Capture/compare mode register 1.*
- `#define _TIM5_CCMR2_SFR(uint8_t, TIM5_AddressBase+0x08)`  
*TIM5 Capture/compare mode register 2.*
- `#define _TIM5_CCMR3_SFR(uint8_t, TIM5_AddressBase+0x09)`  
*TIM5 Capture/compare mode register 3.*
- `#define _TIM5_CCER1_SFR(uint8_t, TIM5_AddressBase+0x0A)`  
*TIM5 Capture/compare enable register 1.*
- `#define _TIM5_CCER2_SFR(uint8_t, TIM5_AddressBase+0x0B)`  
*TIM5 Capture/compare enable register 2.*
- `#define _TIM5_CNTRH_SFR(uint8_t, TIM5_AddressBase+0x0C)`  
*TIM5 counter register high byte.*
- `#define _TIM5_CNTRL_SFR(uint8_t, TIM5_AddressBase+0x0D)`  
*TIM5 counter register low byte.*
- `#define _TIM5_PSCR_SFR(uint8_t, TIM5_AddressBase+0x0E)`  
*TIM5 clock prescaler register.*
- `#define _TIM5_ARRH_SFR(uint8_t, TIM5_AddressBase+0x0F)`  
*TIM5 auto-reload register high byte.*
- `#define _TIM5_ARRL_SFR(uint8_t, TIM5_AddressBase+0x10)`  
*TIM5 auto-reload register low byte.*
- `#define _TIM5_CCR1H_SFR(uint8_t, TIM5_AddressBase+0x11)`  
*TIM5 16-bit capture/compare value 1 high byte.*
- `#define _TIM5_CCR1L_SFR(uint8_t, TIM5_AddressBase+0x12)`  
*TIM5 16-bit capture/compare value 1 low byte.*
- `#define _TIM5_CCR2H_SFR(uint8_t, TIM5_AddressBase+0x13)`  
*TIM5 16-bit capture/compare value 2 high byte.*
- `#define _TIM5_CCR2L_SFR(uint8_t, TIM5_AddressBase+0x14)`  
*TIM5 16-bit capture/compare value 2 low byte.*
- `#define _TIM5_CCR3H_SFR(uint8_t, TIM5_AddressBase+0x15)`  
*TIM5 16-bit capture/compare value 3 high byte.*
- `#define _TIM5_CCR3L_SFR(uint8_t, TIM5_AddressBase+0x16)`  
*TIM5 16-bit capture/compare value 3 low byte.*
- `#define _TIM5_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM5 control register 1 reset value.*
- `#define _TIM5_CR2_RESET_VALUE ((uint8_t) 0x00)`

- TIM5 control register 2 reset value.*

  - #define `_TIM5_SMCR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 Slave mode control register reset value.*

  - #define `_TIM5_IER_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 interrupt enable register reset value.*

  - #define `_TIM5_SR1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 status register 1 reset value.*

  - #define `_TIM5_SR2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 status register 2 reset value.*

  - #define `_TIM5_EGR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 Event generation register reset value.*

  - #define `_TIM5_CCMR1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 Capture/compare mode register 1 reset value.*

  - #define `_TIM5_CCMR2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 Capture/compare mode register 2 reset value.*

  - #define `_TIM5_CCMR3_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 Capture/compare mode register 3 reset value.*

  - #define `_TIM5_CCER1_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 Capture/compare enable register 1 reset value.*

  - #define `_TIM5_CCER2_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 Capture/compare enable register 2 reset value.*

  - #define `_TIM5_CNTRH_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 counter register high byte reset value.*

  - #define `_TIM5_CNTRL_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 counter register low byte reset value.*

  - #define `_TIM5_PSCR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 clock prescaler register reset value.*

  - #define `_TIM5_ARRH_RESET_VALUE` ((uint8\_t) 0xFF)
- TIM5 auto-reload register high byte reset value.*

  - #define `_TIM5_ARRL_RESET_VALUE` ((uint8\_t) 0xFF)
- TIM5 auto-reload register low byte reset value.*

  - #define `_TIM5_CCR1H_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 16-bit capture/compare value 1 high byte reset value.*

  - #define `_TIM5_CCR1L_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 16-bit capture/compare value 1 low byte reset value.*

  - #define `_TIM5_CCR2H_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 16-bit capture/compare value 2 high byte reset value.*

  - #define `_TIM5_CCR2L_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 16-bit capture/compare value 2 low byte reset value.*

  - #define `_TIM5_CCR3H_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 16-bit capture/compare value 3 high byte reset value.*

  - #define `_TIM5_CCR3L_RESET_VALUE` ((uint8\_t) 0x00)
- TIM5 16-bit capture/compare value 3 low byte reset value.*

  - #define `_TIM5_CEN` ((uint8\_t) (0x01 << 0))
- TIM5 Counter enable [0] (in \_TIM5\_CR1)*

  - #define `_TIM5_UDIS` ((uint8\_t) (0x01 << 1))
- TIM5 Update disable [0] (in \_TIM5\_CR1)*

  - #define `_TIM5_URS` ((uint8\_t) (0x01 << 2))
- TIM5 Update request source [0] (in \_TIM5\_CR1)*

  - #define `_TIM5_OPM` ((uint8\_t) (0x01 << 3))
- TIM5 One-pulse mode [0] (in \_TIM5\_CR1)*



- `#define _TIM5_ARPE ((uint8_t) (0x01 << 7))`  
*TIM5 Auto-reload preload enable [0] (in \_TIM5\_CR1)*
- `#define _TIM5_CCPC ((uint8_t) (0x01 << 0))`  
*TIM5 Capture/compare preloaded control [0] (in \_TIM5\_CR2)*
- `#define _TIM5_COMS ((uint8_t) (0x01 << 2))`  
*TIM5 Capture/compare control update selection [0] (in \_TIM5\_CR2)*
- `#define _TIM5_MMS ((uint8_t) (0x07 << 4))`  
*TIM5 Master mode selection [2:0] (in \_TIM5\_CR2)*
- `#define _TIM5_MMS0 ((uint8_t) (0x01 << 4))`  
*TIM5 Master mode selection [0] (in \_TIM5\_CR2)*
- `#define _TIM5_MMS1 ((uint8_t) (0x01 << 5))`  
*TIM5 Master mode selection [1] (in \_TIM5\_CR2)*
- `#define _TIM5_MMS2 ((uint8_t) (0x01 << 6))`  
*TIM5 Master mode selection [2] (in \_TIM5\_CR2)*
- `#define _TIM5_SMS ((uint8_t) (0x07 << 0))`  
*TIM5 Clock/trigger/slave mode selection [2:0] (in \_TIM5\_SMCR)*
- `#define _TIM5_SMS0 ((uint8_t) (0x01 << 0))`  
*TIM5 Clock/trigger/slave mode selection [0] (in \_TIM5\_SMCR)*
- `#define _TIM5_SMS1 ((uint8_t) (0x01 << 1))`  
*TIM5 Clock/trigger/slave mode selection [1] (in \_TIM5\_SMCR)*
- `#define _TIM5_SMS2 ((uint8_t) (0x01 << 2))`  
*TIM5 Clock/trigger/slave mode selection [2] (in \_TIM5\_SMCR)*
- `#define _TIM5_TS ((uint8_t) (0x07 << 4))`  
*TIM5 Trigger selection [2:0] (in \_TIM5\_SMCR)*
- `#define _TIM5_TS0 ((uint8_t) (0x01 << 4))`  
*TIM5 Trigger selection [0] (in \_TIM5\_SMCR)*
- `#define _TIM5_TS1 ((uint8_t) (0x01 << 5))`  
*TIM5 Trigger selection [1] (in \_TIM5\_SMCR)*
- `#define _TIM5_TS2 ((uint8_t) (0x01 << 6))`  
*TIM5 Trigger selection [2] (in \_TIM5\_SMCR)*
- `#define _TIM5_MSM ((uint8_t) (0x01 << 7))`  
*TIM5 Master/slave mode [0] (in \_TIM5\_SMCR)*
- `#define _TIM5_UIE ((uint8_t) (0x01 << 0))`  
*TIM5 Update interrupt enable [0] (in \_TIM5\_IER)*
- `#define _TIM5_CC1IE ((uint8_t) (0x01 << 1))`  
*TIM5 Capture/compare 1 interrupt enable [0] (in \_TIM5\_IER)*
- `#define _TIM5_CC2IE ((uint8_t) (0x01 << 2))`  
*TIM5 Capture/compare 2 interrupt enable [0] (in \_TIM5\_IER)*
- `#define _TIM5_CC3IE ((uint8_t) (0x01 << 3))`  
*TIM5 Capture/compare 3 interrupt enable [0] (in \_TIM5\_IER)*
- `#define _TIM5_TIE ((uint8_t) (0x01 << 6))`  
*TIM5 Trigger interrupt enable [0] (in \_TIM5\_IER)*
- `#define _TIM5_UIF ((uint8_t) (0x01 << 0))`  
*TIM5 Update interrupt flag [0] (in \_TIM5\_SR1)*
- `#define _TIM5_CC1IF ((uint8_t) (0x01 << 1))`  
*TIM5 Capture/compare 1 interrupt flag [0] (in \_TIM5\_SR1)*
- `#define _TIM5_CC2IF ((uint8_t) (0x01 << 2))`  
*TIM5 Capture/compare 2 interrupt flag [0] (in \_TIM5\_SR1)*
- `#define _TIM5_CC3IF ((uint8_t) (0x01 << 3))`  
*TIM5 Capture/compare 3 interrupt flag [0] (in \_TIM5\_SR1)*
- `#define _TIM5_TIF ((uint8_t) (0x01 << 6))`



- TIM5 Trigger interrupt flag [0] (in \_TIM5\_SR1)*
- #define `_TIM5_CC1OF` ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 1 overcapture flag [0] (in \_TIM5\_SR2)*
- #define `_TIM5_CC2OF` ((uint8\_t) (0x01 << 2))
- TIM5 Capture/compare 2 overcapture flag [0] (in \_TIM5\_SR2)*
- #define `_TIM5_CC3OF` ((uint8\_t) (0x01 << 3))
- TIM5 Capture/compare 3 overcapture flag [0] (in \_TIM5\_SR2)*
- #define `_TIM5_UG` ((uint8\_t) (0x01 << 0))
- TIM5 Update generation [0] (in \_TIM5\_EGR)*
- #define `_TIM5_CC1G` ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 1 generation [0] (in \_TIM5\_EGR)*
- #define `_TIM5_CC2G` ((uint8\_t) (0x01 << 2))
- TIM5 Capture/compare 2 generation [0] (in \_TIM5\_EGR)*
- #define `_TIM5_CC3G` ((uint8\_t) (0x01 << 3))
- TIM5 Capture/compare 3 generation [0] (in \_TIM5\_EGR)*
- #define `_TIM5_TG` ((uint8\_t) (0x01 << 6))
- TIM5 Trigger generation [0] (in \_TIM5\_EGR)*
- #define `_TIM5_CC1S` ((uint8\_t) (0x03 << 0))
- TIM5 Compare 1 selection [1:0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_CC1S0` ((uint8\_t) (0x01 << 0))
- TIM5 Compare 1 selection [0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_CC1S1` ((uint8\_t) (0x01 << 1))
- TIM5 Compare 1 selection [1] (in \_TIM5\_CCMR1)*
- #define `_TIM5_OC1PE` ((uint8\_t) (0x01 << 3))
- TIM5 Output compare 1 preload enable [0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_OC1M` ((uint8\_t) (0x07 << 4))
- TIM5 Output compare 1 mode [2:0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_OC1M0` ((uint8\_t) (0x01 << 4))
- TIM5 Output compare 1 mode [0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_OC1M1` ((uint8\_t) (0x01 << 5))
- TIM5 Output compare 1 mode [1] (in \_TIM5\_CCMR1)*
- #define `_TIM5_OC1M2` ((uint8\_t) (0x01 << 6))
- TIM5 Output compare 1 mode [2] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1PSC` ((uint8\_t) (0x03 << 2))
- TIM5 Input capture 1 prescaler [1:0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1PSC0` ((uint8\_t) (0x01 << 2))
- TIM5 Input capture 1 prescaler [0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1PSC1` ((uint8\_t) (0x01 << 3))
- TIM5 Input capture 1 prescaler [1] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1F` ((uint8\_t) (0x0F << 4))
- TIM5 Output compare 1 mode [3:0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1F0` ((uint8\_t) (0x01 << 4))
- TIM5 Input capture 1 filter [0] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1F1` ((uint8\_t) (0x01 << 5))
- TIM5 Input capture 1 filter [1] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1F2` ((uint8\_t) (0x01 << 6))
- TIM5 Input capture 1 filter [2] (in \_TIM5\_CCMR1)*
- #define `_TIM5_IC1F3` ((uint8\_t) (0x01 << 7))
- TIM5 Input capture 1 filter [3] (in \_TIM5\_CCMR1)*
- #define `_TIM5_CC2S` ((uint8\_t) (0x03 << 0))
- TIM5 Compare 2 selection [1:0] (in \_TIM5\_CCMR2)*

- `#define _TIM5_CC2S0` ((uint8\_t) (0x01 << 0))  
*TIM5 Compare 2 selection [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_CC2S1` ((uint8\_t) (0x01 << 1))  
*TIM5 Compare 2 selection [1] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2PE` ((uint8\_t) (0x01 << 3))  
*TIM5 Output compare 2 preload enable [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2M` ((uint8\_t) (0x07 << 4))  
*TIM5 Output compare 2 mode [2:0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2M0` ((uint8\_t) (0x01 << 4))  
*TIM5 Output compare 2 mode [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2M1` ((uint8\_t) (0x01 << 5))  
*TIM5 Output compare 2 mode [1] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2M2` ((uint8\_t) (0x01 << 6))  
*TIM5 Output compare 2 mode [2] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2PSC` ((uint8\_t) (0x03 << 2))  
*TIM5 Input capture 2 prescaler [1:0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2PSC0` ((uint8\_t) (0x01 << 2))  
*TIM5 Input capture 2 prescaler [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2PSC1` ((uint8\_t) (0x01 << 3))  
*TIM5 Input capture 2 prescaler [1] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2F` ((uint8\_t) (0x0F << 4))  
*TIM5 Output compare 2 mode [3:0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2F0` ((uint8\_t) (0x01 << 4))  
*TIM5 Input capture 2 filter [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2F1` ((uint8\_t) (0x01 << 5))  
*TIM5 Input capture 2 filter [1] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2F2` ((uint8\_t) (0x01 << 6))  
*TIM5 Input capture 2 filter [2] (in \_TIM5\_CCMR2)*
- `#define _TIM5_IC2F3` ((uint8\_t) (0x01 << 7))  
*TIM5 Input capture 2 filter [3] (in \_TIM5\_CCMR2)*
- `#define _TIM5_CC3S` ((uint8\_t) (0x03 << 0))  
*TIM5 Compare 3 selection [1:0] (in \_TIM5\_CCMR3)*
- `#define _TIM5_CC3S0` ((uint8\_t) (0x01 << 0))  
*TIM5 Compare 3 selection [0] (in \_TIM5\_CCMR3)*
- `#define _TIM5_CC3S1` ((uint8\_t) (0x01 << 1))  
*TIM5 Compare 3 selection [1] (in \_TIM5\_CCMR3)*
- `#define _TIM5_OC3PE` ((uint8\_t) (0x01 << 3))  
*TIM5 Output compare 3 preload enable [0] (in \_TIM5\_CCMR3)*
- `#define _TIM5_OC3M` ((uint8\_t) (0x07 << 4))  
*TIM5 Output compare 3 mode [2:0] (in \_TIM5\_CCMR3)*
- `#define _TIM5_OC3M0` ((uint8\_t) (0x01 << 4))  
*TIM5 Output compare 3 mode [0] (in \_TIM5\_CCMR3)*
- `#define _TIM5_OC3M1` ((uint8\_t) (0x01 << 5))  
*TIM5 Output compare 3 mode [1] (in \_TIM5\_CCMR3)*
- `#define _TIM5_OC3M2` ((uint8\_t) (0x01 << 6))  
*TIM5 Output compare 3 mode [2] (in \_TIM5\_CCMR3)*
- `#define _TIM5_IC3PSC` ((uint8\_t) (0x03 << 2))  
*TIM5 Input capture 3 prescaler [1:0] (in \_TIM5\_CCMR3)*
- `#define _TIM5_IC3PSC0` ((uint8\_t) (0x01 << 2))  
*TIM5 Input capture 3 prescaler [0] (in \_TIM5\_CCMR3)*
- `#define _TIM5_IC3PSC1` ((uint8\_t) (0x01 << 3))

- TIM5 Input capture 3 prescaler [1] (in \_TIM5\_CCMR3)*
- #define `_TIM5_IC3F` ((uint8\_t) (0x0F << 4))
- TIM5 Output compare 3 mode [3:0] (in \_TIM5\_CCMR3)*
- #define `_TIM5_IC3F0` ((uint8\_t) (0x01 << 4))
- TIM5 Input capture 3 filter [0] (in \_TIM5\_CCMR3)*
- #define `_TIM5_IC3F1` ((uint8\_t) (0x01 << 5))
- TIM5 Input capture 3 filter [1] (in \_TIM5\_CCMR3)*
- #define `_TIM5_IC3F2` ((uint8\_t) (0x01 << 6))
- TIM5 Input capture 3 filter [2] (in \_TIM5\_CCMR3)*
- #define `_TIM5_IC3F3` ((uint8\_t) (0x01 << 7))
- TIM5 Input capture 3 filter [3] (in \_TIM5\_CCMR3)*
- #define `_TIM5_CC1E` ((uint8\_t) (0x01 << 0))
- TIM5 Capture/compare 1 output enable [0] (in \_TIM5\_CCER1)*
- #define `_TIM5_CC1P` ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 1 output polarity [0] (in \_TIM5\_CCER1)*
- #define `_TIM5_CC2E` ((uint8\_t) (0x01 << 4))
- TIM5 Capture/compare 2 output enable [0] (in \_TIM5\_CCER1)*
- #define `_TIM5_CC2P` ((uint8\_t) (0x01 << 5))
- TIM5 Capture/compare 2 output polarity [0] (in \_TIM5\_CCER1)*
- #define `_TIM5_CC3E` ((uint8\_t) (0x01 << 0))
- TIM5 Capture/compare 3 output enable [0] (in \_TIM5\_CCER2)*
- #define `_TIM5_CC3P` ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 3 output polarity [0] (in \_TIM5\_CCER2)*
- #define `_TIM5_PSC` ((uint8\_t) (0x0F << 0))
- TIM5 clock prescaler [3:0] (in \_TIM5\_PSCR)*
- #define `_TIM5_PSC0` ((uint8\_t) (0x01 << 0))
- TIM5 clock prescaler [0] (in \_TIM5\_PSCR)*
- #define `_TIM5_PSC1` ((uint8\_t) (0x01 << 1))
- TIM5 clock prescaler [1] (in \_TIM5\_PSCR)*
- #define `_TIM5_PSC2` ((uint8\_t) (0x01 << 2))
- TIM5 clock prescaler [2] (in \_TIM5\_PSCR)*
- #define `_TIM5_PSC3` ((uint8\_t) (0x01 << 3))
- TIM5 clock prescaler [3] (in \_TIM5\_PSCR)*
- #define `_TIM6_SFR`(TIM6\_t, TIM6\_AddressBase)
- TIM6 struct/bit access.*
- #define `_TIM6_CR_SFR`(uint8\_t, TIM6\_AddressBase+0x00)
- TIM6 control register.*
- #define `_TIM6_IER_SFR`(uint8\_t, TIM6\_AddressBase+0x01)
- TIM6 interrupt enable register.*
- #define `_TIM6_SR_SFR`(uint8\_t, TIM6\_AddressBase+0x02)
- TIM6 status register.*
- #define `_TIM6_EGR_SFR`(uint8\_t, TIM6\_AddressBase+0x03)
- TIM6 event generation register.*
- #define `_TIM6_CNTR_SFR`(uint8\_t, TIM6\_AddressBase+0x04)
- TIM6 counter register.*
- #define `_TIM6_PSCR_SFR`(uint8\_t, TIM6\_AddressBase+0x05)
- TIM6 clock prescaler register.*
- #define `_TIM6_ARR_SFR`(uint8\_t, TIM6\_AddressBase+0x06)
- TIM6 auto-reload register.*
- #define `_TIM6_CR_RESET_VALUE` ((uint8\_t) 0x00)
- TIM6 control register reset value.*

- `#define _TIM6_IER_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 interrupt enable register reset value.*
- `#define _TIM6_SR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 status register reset value.*
- `#define _TIM6_EGR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 event generation register reset value.*
- `#define _TIM6_CNTR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 counter register reset value.*
- `#define _TIM6_PSCR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 clock prescaler register reset value.*
- `#define _TIM6_ARR_RESET_VALUE ((uint8_t) 0xFF)`  
*TIM6 auto-reload register reset value.*
- `#define _TIM6_CEN ((uint8_t) (0x01 << 0))`  
*TIM6 Counter enable [0] (in \_TIM6\_CR1)*
- `#define _TIM6_UDIS ((uint8_t) (0x01 << 1))`  
*TIM6 Update disable [0] (in \_TIM6\_CR1)*
- `#define _TIM6_URS ((uint8_t) (0x01 << 2))`  
*TIM6 Update request source [0] (in \_TIM6\_CR1)*
- `#define _TIM6_OPM ((uint8_t) (0x01 << 3))`  
*TIM6 One-pulse mode [0] (in \_TIM6\_CR1)*
- `#define _TIM6_ARPE ((uint8_t) (0x01 << 7))`  
*TIM6 Auto-reload preload enable [0] (in \_TIM6\_CR1)*
- `#define _TIM6_MMS ((uint8_t) (0x07 << 4))`  
*TIM6 Master mode selection [2:0] (in \_TIM6\_CR2)*
- `#define _TIM6_MMS0 ((uint8_t) (0x01 << 4))`  
*TIM6 Master mode selection [0] (in \_TIM6\_CR2)*
- `#define _TIM6_MMS1 ((uint8_t) (0x01 << 5))`  
*TIM6 Master mode selection [1] (in \_TIM6\_CR2)*
- `#define _TIM6_MMS2 ((uint8_t) (0x01 << 6))`  
*TIM6 Master mode selection [2] (in \_TIM6\_CR2)*
- `#define _TIM6_SMS ((uint8_t) (0x07 << 0))`  
*TIM6 Clock/trigger/slave mode selection [2:0] (in \_TIM6\_SMCR)*
- `#define _TIM6_SMS0 ((uint8_t) (0x01 << 0))`  
*TIM6 Clock/trigger/slave mode selection [0] (in \_TIM6\_SMCR)*
- `#define _TIM6_SMS1 ((uint8_t) (0x01 << 1))`  
*TIM6 Clock/trigger/slave mode selection [1] (in \_TIM6\_SMCR)*
- `#define _TIM6_SMS2 ((uint8_t) (0x01 << 2))`  
*TIM6 Clock/trigger/slave mode selection [2] (in \_TIM6\_SMCR)*
- `#define _TIM6_TS ((uint8_t) (0x07 << 4))`  
*TIM6 Trigger selection [2:0] (in \_TIM6\_SMCR)*
- `#define _TIM6_TS0 ((uint8_t) (0x01 << 4))`  
*TIM6 Trigger selection [0] (in \_TIM6\_SMCR)*
- `#define _TIM6_TS1 ((uint8_t) (0x01 << 5))`  
*TIM6 Trigger selection [1] (in \_TIM6\_SMCR)*
- `#define _TIM6_TS2 ((uint8_t) (0x01 << 6))`  
*TIM6 Trigger selection [2] (in \_TIM6\_SMCR)*
- `#define _TIM6_UIE ((uint8_t) (0x01 << 0))`  
*TIM6 Update interrupt enable [0] (in \_TIM6\_IER)*
- `#define _TIM6_UIF ((uint8_t) (0x01 << 0))`  
*TIM6 Update interrupt flag [0] (in \_TIM6\_SR)*
- `#define _TIM6_UG ((uint8_t) (0x01 << 0))`

- TIM6 Update generation [0] (in \_TIM6\_EGR)*
- #define `_TIM6_PSC` ((uint8\_t) (0x07 << 0))  
*TIM6 clock prescaler [2:0] (in \_TIM6\_PSCR)*
- #define `_TIM6_PSC0` ((uint8\_t) (0x01 << 0))  
*TIM6 clock prescaler [0] (in \_TIM6\_PSCR)*
- #define `_TIM6_PSC1` ((uint8\_t) (0x01 << 1))  
*TIM6 clock prescaler [1] (in \_TIM6\_PSCR)*
- #define `_TIM6_PSC2` ((uint8\_t) (0x01 << 2))  
*TIM6 clock prescaler [2] (in \_TIM6\_PSCR)*
- #define `_ADC1_SFR(ADC1_t, ADC1_AddressBase)`  
*ADC1 struct/bit access.*
- #define `_ADC1_DB0RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x00`)  
*ADC1 10-bit Data Buffer Register 0.*
- #define `_ADC1_DB0RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x01`)  
*ADC1 10-bit Data Buffer Register 0.*
- #define `_ADC1_DB1RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x02`)  
*ADC1 10-bit Data Buffer Register 1.*
- #define `_ADC1_DB1RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x03`)  
*ADC1 10-bit Data Buffer Register 1.*
- #define `_ADC1_DB2RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x04`)  
*ADC1 10-bit Data Buffer Register 2.*
- #define `_ADC1_DB2RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x05`)  
*ADC1 10-bit Data Buffer Register 2.*
- #define `_ADC1_DB3RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x06`)  
*ADC1 10-bit Data Buffer Register 3.*
- #define `_ADC1_DB3RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x07`)  
*ADC1 10-bit Data Buffer Register 3.*
- #define `_ADC1_DB4RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x08`)  
*ADC1 10-bit Data Buffer Register 4.*
- #define `_ADC1_DB4RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x09`)  
*ADC1 10-bit Data Buffer Register 4.*
- #define `_ADC1_DB5RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x0A`)  
*ADC1 10-bit Data Buffer Register 5.*
- #define `_ADC1_DB5RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x0B`)  
*ADC1 10-bit Data Buffer Register 5.*
- #define `_ADC1_DB6RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x0C`)  
*ADC1 10-bit Data Buffer Register 6.*
- #define `_ADC1_DB6RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x0D`)  
*ADC1 10-bit Data Buffer Register 6.*
- #define `_ADC1_DB7RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x0E`)  
*ADC1 10-bit Data Buffer Register 7.*
- #define `_ADC1_DB7RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x0F`)  
*ADC1 10-bit Data Buffer Register 7.*
- #define `_ADC1_DB8RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x10`)  
*ADC1 10-bit Data Buffer Register 8.*
- #define `_ADC1_DB8RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x11`)  
*ADC1 10-bit Data Buffer Register 8.*
- #define `_ADC1_DB9RH_SFR`(uint8\_t, `_ADC1_AddressBase+0x12`)  
*ADC1 10-bit Data Buffer Register 9.*
- #define `_ADC1_DB9RL_SFR`(uint8\_t, `_ADC1_AddressBase+0x13`)  
*ADC1 10-bit Data Buffer Register 9.*

- `#define _ADC1_CSR_SFR(uint8_t, ADC1_AddressBase+0x20)`  
*ADC1 control/status register.*
- `#define _ADC1_CR1_SFR(uint8_t, ADC1_AddressBase+0x21)`  
*ADC1 Configuration Register 1.*
- `#define _ADC1_CR2_SFR(uint8_t, ADC1_AddressBase+0x22)`  
*ADC1 Configuration Register 2.*
- `#define _ADC1_CR3_SFR(uint8_t, ADC1_AddressBase+0x23)`  
*ADC1 Configuration Register 3.*
- `#define _ADC1_DRH_SFR(uint8_t, ADC1_AddressBase+0x24)`  
*ADC1 (unbuffered) 10-bit measurement result.*
- `#define _ADC1_DRL_SFR(uint8_t, ADC1_AddressBase+0x25)`  
*ADC1 (unbuffered) 10-bit measurement result.*
- `#define _ADC1_TDRH_SFR(uint8_t, ADC1_AddressBase+0x26)`  
*ADC1 Schmitt trigger disable register.*
- `#define _ADC1_TDRL_SFR(uint8_t, ADC1_AddressBase+0x27)`  
*ADC1 Schmitt trigger disable register.*
- `#define _ADC1_HTRH_SFR(uint8_t, ADC1_AddressBase+0x28)`  
*ADC1 watchdog high threshold register.*
- `#define _ADC1_HTRL_SFR(uint8_t, ADC1_AddressBase+0x29)`  
*ADC1 watchdog high threshold register.*
- `#define _ADC1_LTRH_SFR(uint8_t, ADC1_AddressBase+0x2A)`  
*ADC1 watchdog low threshold register.*
- `#define _ADC1_LTRL_SFR(uint8_t, ADC1_AddressBase+0x2B)`  
*ADC1 watchdog low threshold register.*
- `#define _ADC1_AWSRH_SFR(uint8_t, ADC1_AddressBase+0x2C)`  
*ADC1 watchdog status register.*
- `#define _ADC1_AWSRL_SFR(uint8_t, ADC1_AddressBase+0x2D)`  
*ADC1 watchdog status register.*
- `#define _ADC1_AWCRH_SFR(uint8_t, ADC1_AddressBase+0x2E)`  
*ADC1 watchdog control register.*
- `#define _ADC1_AWCRL_SFR(uint8_t, ADC1_AddressBase+0x2F)`  
*ADC1 watchdog control register.*
- `#define _ADC1_CSR_RESET_VALUE ((uint8_t) 0x00)`  
*ADC1 control/status register reset value.*
- `#define _ADC1_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*ADC1 Configuration Register 1 reset value.*
- `#define _ADC1_CR2_RESET_VALUE ((uint8_t) 0x00)`  
*ADC1 Configuration Register 2 reset value.*
- `#define _ADC1_CR3_RESET_VALUE ((uint8_t) 0x00)`  
*ADC1 Configuration Register 3 reset value.*
- `#define _ADC1_TDRH_RESET_VALUE ((uint8_t) 0x00)`  
*ADC1 Schmitt trigger disable register reset value.*
- `#define _ADC1_TDRL_RESET_VALUE ((uint8_t) 0x00)`  
*ADC1 Schmitt trigger disable register reset value.*
- `#define _ADC1_HTRH_RESET_VALUE ((uint8_t) 0xFF)`  
*ADC1 watchdog high threshold register reset value.*
- `#define _ADC1_HTRL_RESET_VALUE ((uint8_t) 0x03)`  
*ADC1 watchdog high threshold register reset value.*
- `#define _ADC1_LTRH_RESET_VALUE ((uint8_t) 0x00)`  
*ADC1 watchdog low threshold register reset value.*
- `#define _ADC1_LTRL_RESET_VALUE ((uint8_t) 0x00)`

- ADC1 watchdog low threshold register reset value.*

  - #define `_ADC1_AWCRH_RESET_VALUE` ((uint8\_t) 0x00)
- ADC1 watchdog control register reset value.*

  - #define `_ADC1_AWCRL_RESET_VALUE` ((uint8\_t) 0x00)
- ADC1 watchdog control register reset value.*

  - #define `_ADC1_CH` ((uint8\_t) (0x0F << 0))
- ADC1 Channel selection bits [3:0] (in \_ADC1\_CSR)*

  - #define `_ADC1_CH0` ((uint8\_t) (0x01 << 0))
- ADC1 Channel selection bits [0] (in \_ADC1\_CSR)*

  - #define `_ADC1_CH1` ((uint8\_t) (0x01 << 1))
- ADC1 Channel selection bits [1] (in \_ADC1\_CSR)*

  - #define `_ADC1_CH2` ((uint8\_t) (0x01 << 2))
- ADC1 Channel selection bits [2] (in \_ADC1\_CSR)*

  - #define `_ADC1_CH3` ((uint8\_t) (0x01 << 3))
- ADC1 Channel selection bits [3] (in \_ADC1\_CSR)*

  - #define `_ADC1_AWDIE` ((uint8\_t) (0x01 << 4))
- ADC1 Analog watchdog interrupt enable [0] (in \_ADC1\_CSR)*

  - #define `_ADC1_EOCIE` ((uint8\_t) (0x01 << 5))
- ADC1 Interrupt enable for EOC [0] (in \_ADC1\_CSR)*

  - #define `_ADC1_AWD` ((uint8\_t) (0x01 << 6))
- ADC1 Analog Watchdog flag [0] (in \_ADC1\_CSR)*

  - #define `_ADC1_EOC` ((uint8\_t) (0x01 << 7))
- ADC1 End of conversion [0] (in \_ADC1\_CSR)*

  - #define `_ADC1_ADON` ((uint8\_t) (0x01 << 0))
- ADC1 Conversion on/off [0] (in \_ADC1\_CR1)*

  - #define `_ADC1_CONT` ((uint8\_t) (0x01 << 1))
- ADC1 Continuous conversion [0] (in \_ADC1\_CR1)*

  - #define `_ADC1_SPSEL` ((uint8\_t) (0x07 << 4))
- ADC1 clock prescaler selection [2:0] (in \_ADC1\_CR1)*

  - #define `_ADC1_SPSEL0` ((uint8\_t) (0x01 << 4))
- ADC1 clock prescaler selection [0] (in \_ADC1\_CR1)*

  - #define `_ADC1_SPSEL1` ((uint8\_t) (0x01 << 5))
- ADC1 clock prescaler selection [1] (in \_ADC1\_CR1)*

  - #define `_ADC1_SPSEL2` ((uint8\_t) (0x01 << 6))
- ADC1 clock prescaler selection [2] (in \_ADC1\_CR1)*

  - #define `_ADC1_SCAN` ((uint8\_t) (0x01 << 1))
- ADC1 Scan mode enable [0] (in \_ADC1\_CR2)*

  - #define `_ADC1_ALIGN` ((uint8\_t) (0x01 << 3))
- ADC1 Data alignment [0] (in \_ADC1\_CR2)*

  - #define `_ADC1_EXTSEL` ((uint8\_t) (0x03 << 4))
- ADC1 External event selection [1:0] (in \_ADC1\_CR2)*

  - #define `_ADC1_EXTSEL0` ((uint8\_t) (0x01 << 4))
- ADC1 External event selection [0] (in \_ADC1\_CR2)*

  - #define `_ADC1_EXTSEL1` ((uint8\_t) (0x01 << 5))
- ADC1 External event selection [1] (in \_ADC1\_CR2)*

  - #define `_ADC1_EXTTRIG` ((uint8\_t) (0x01 << 6))
- ADC1 External trigger enable [0] (in \_ADC1\_CR2)*

  - #define `_ADC1_OVR` ((uint8\_t) (0x01 << 6))
- ADC1 Overrun flag [0] (in \_ADC1\_CR3)*

  - #define `_ADC1_DBUF` ((uint8\_t) (0x01 << 7))
- ADC1 Data buffer enable [0] (in \_ADC1\_CR3)*



- `#define _ADC2_SFR(ADC2_t, ADC2_AddressBase)`  
*ADC2 struct/bit access.*
- `#define _ADC2_CSR_SFR(uint8_t, ADC2_AddressBase+0x00)`  
*ADC2 control/status register.*
- `#define _ADC2_CR1_SFR(uint8_t, ADC2_AddressBase+0x01)`  
*ADC2 Configuration Register 1.*
- `#define _ADC2_CR2_SFR(uint8_t, ADC2_AddressBase+0x02)`  
*ADC2 Configuration Register 2.*
- `#define _ADC2_DRH_SFR(uint8_t, ADC2_AddressBase+0x04)`  
*ADC2 (unbuffered) 10-bit measurement result.*
- `#define _ADC2_DRL_SFR(uint8_t, ADC2_AddressBase+0x05)`  
*ADC2 (unbuffered) 10-bit measurement result.*
- `#define _ADC2_TDRH_SFR(uint8_t, ADC2_AddressBase+0x06)`  
*ADC2 Schmitt trigger disable register.*
- `#define _ADC2_TDRL_SFR(uint8_t, ADC2_AddressBase+0x07)`  
*ADC2 Schmitt trigger disable register.*
- `#define _ADC2_CSR_RESET_VALUE ((uint8_t) 0x00)`  
*ADC2 control/status register reset value.*
- `#define _ADC2_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*ADC2 Configuration Register 1 reset value.*
- `#define _ADC2_CR2_RESET_VALUE ((uint8_t) 0x00)`  
*ADC2 Configuration Register 2 reset value.*
- `#define _ADC2_TDRL_RESET_VALUE ((uint8_t) 0x00)`  
*ADC2 Schmitt trigger disable register reset value.*
- `#define _ADC2_TDRH_RESET_VALUE ((uint8_t) 0x00)`  
*ADC2 Schmitt trigger disable register reset value.*
- `#define _ADC2_CH ((uint8_t) (0x0F << 0))`  
*ADC2 Channel selection bits [3:0] (in \_ADC2\_CSR)*
- `#define _ADC2_CH0 ((uint8_t) (0x01 << 0))`  
*ADC2 Channel selection bits [0] (in \_ADC2\_CSR)*
- `#define _ADC2_CH1 ((uint8_t) (0x01 << 1))`  
*ADC2 Channel selection bits [1] (in \_ADC2\_CSR)*
- `#define _ADC2_CH2 ((uint8_t) (0x01 << 2))`  
*ADC2 Channel selection bits [2] (in \_ADC2\_CSR)*
- `#define _ADC2_CH3 ((uint8_t) (0x01 << 3))`  
*ADC2 Channel selection bits [3] (in \_ADC2\_CSR)*
- `#define _ADC2_EOCIE ((uint8_t) (0x01 << 5))`  
*ADC2 Interrupt enable for EOC [0] (in \_ADC2\_CSR)*
- `#define _ADC2_EOC ((uint8_t) (0x01 << 7))`  
*ADC2 End of conversion [0] (in \_ADC2\_CSR)*
- `#define _ADC2_ADON ((uint8_t) (0x01 << 0))`  
*ADC2 Conversion on/off [0] (in \_ADC2\_CR1)*
- `#define _ADC2_CONT ((uint8_t) (0x01 << 1))`  
*ADC2 Continuous conversion [0] (in \_ADC2\_CR1)*
- `#define _ADC2_SPSEL ((uint8_t) (0x07 << 4))`  
*ADC2 clock prescaler selection [2:0] (in \_ADC2\_CR1)*
- `#define _ADC2_SPSEL0 ((uint8_t) (0x01 << 4))`  
*ADC2 clock prescaler selection [0] (in \_ADC2\_CR1)*
- `#define _ADC2_SPSEL1 ((uint8_t) (0x01 << 5))`  
*ADC2 clock prescaler selection [1] (in \_ADC2\_CR1)*
- `#define _ADC2_SPSEL2 ((uint8_t) (0x01 << 6))`



- ADC2 clock prescaler selection [2] (in \_ADC2\_CR1)*
- #define `_ADC2_ALIGN` ((uint8\_t) (0x01 << 3))
- ADC2 Data alignment [0] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTSEL` ((uint8\_t) (0x03 << 4))
- ADC2 External event selection [1:0] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTSEL0` ((uint8\_t) (0x01 << 4))
- ADC2 External event selection [0] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTSEL1` ((uint8\_t) (0x01 << 5))
- ADC2 External event selection [1] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTTRIG` ((uint8\_t) (0x01 << 6))
- ADC2 External trigger enable [0] (in \_ADC2\_CR2)*
- #define `_CAN_SFR`(CAN\_t, CAN\_AddressBase)
- CAN struct/bit access.*
- #define `_CAN_MCR_SFR`(uint8\_t, CAN\_AddressBase+0x00)
- CAN master control register.*
- #define `_CAN_MSR_SFR`(uint8\_t, CAN\_AddressBase+0x01)
- CAN master status register.*
- #define `_CAN_TSR_SFR`(uint8\_t, CAN\_AddressBase+0x02)
- CAN transmit status register.*
- #define `_CAN_TPR_SFR`(uint8\_t, CAN\_AddressBase+0x03)
- CAN transmit priority register.*
- #define `_CAN_RFR_SFR`(uint8\_t, CAN\_AddressBase+0x04)
- CAN receive FIFO register.*
- #define `_CAN_IER_SFR`(uint8\_t, CAN\_AddressBase+0x05)
- CAN interrupt enable register.*
- #define `_CAN_DGR_SFR`(uint8\_t, CAN\_AddressBase+0x06)
- CAN diagnosis register.*
- #define `_CAN_PSR_SFR`(uint8\_t, CAN\_AddressBase+0x07)
- CAN page selection for below paged registers.*
- #define `_CAN_MCSR_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x00)
- CAN message control/status register (page 0,1,5)*
- #define `_CAN_MDLCR_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x01)
- CAN mailbox data length control register (page 0,1,5,7)*
- #define `_CAN_MIDR1_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x02)
- CAN mailbox identifier register 1 (page 0,1,5,7)*
- #define `_CAN_MIDR2_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x03)
- CAN mailbox identifier register 2 (page 0,1,5,7)*
- #define `_CAN_MIDR3_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x04)
- CAN mailbox identifier register 3 (page 0,1,5,7)*
- #define `_CAN_MIDR4_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x05)
- CAN mailbox identifier register 4 (page 0,1,5,7)*
- #define `_CAN_MDAR1_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x06)
- CAN mailbox data register 1 (page 0,1,5,7) \*/.*
- #define `_CAN_MDAR2_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x07)
- CAN mailbox data register 2 (page 0,1,5,7) \*/.*
- #define `_CAN_MDAR3_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x08)
- CAN mailbox data register 3 (page 0,1,5,7) \*/.*
- #define `_CAN_MDAR4_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x09)
- CAN mailbox data register 4 (page 0,1,5,7) \*/.*
- #define `_CAN_MDAR5_SFR`(uint8\_t, CAN\_AddressBase+0x08+0x0A)
- CAN mailbox data register 5 (page 0,1,5,7) \*/.*

- `#define _CAN_MDAR6_SFR(uint8_t, CAN_AddressBase+0x08+0x0B)`  
*CAN mailbox data register 6 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR7_SFR(uint8_t, CAN_AddressBase+0x08+0x0C)`  
*CAN mailbox data register 7 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR8_SFR(uint8_t, CAN_AddressBase+0x08+0x0D)`  
*CAN mailbox data register 8 (page 0,1,5,7) \*/.*
- `#define _CAN_MTSRL_SFR(uint8_t, CAN_AddressBase+0x08+0x0E)`  
*CAN mailbox time stamp register low byte (page 0,1,5,7) \*/.*
- `#define _CAN_MTSRH_SFR(uint8_t, CAN_AddressBase+0x08+0x0F)`  
*CAN mailbox time stamp register high byte (page 0,1,5,7) \*/.*
- `#define _CAN_F0R1_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`  
*CAN acceptance filter 0/1 (page 2)*
- `#define _CAN_F0R2_SFR(uint8_t, CAN_AddressBase+0x08+0x01)`  
*CAN acceptance filter 0/2 (page 2)*
- `#define _CAN_F0R3_SFR(uint8_t, CAN_AddressBase+0x08+0x02)`  
*CAN acceptance filter 0/3 (page 2)*
- `#define _CAN_F0R4_SFR(uint8_t, CAN_AddressBase+0x08+0x03)`  
*CAN acceptance filter 0/4 (page 2)*
- `#define _CAN_F0R5_SFR(uint8_t, CAN_AddressBase+0x08+0x04)`  
*CAN acceptance filter 0/5 (page 2)*
- `#define _CAN_F0R6_SFR(uint8_t, CAN_AddressBase+0x08+0x05)`  
*CAN acceptance filter 0/6 (page 2)*
- `#define _CAN_F0R7_SFR(uint8_t, CAN_AddressBase+0x08+0x06)`  
*CAN acceptance filter 0/7 (page 2)*
- `#define _CAN_F0R8_SFR(uint8_t, CAN_AddressBase+0x08+0x07)`  
*CAN acceptance filter 0/8 (page 2)*
- `#define _CAN_F1R1_SFR(uint8_t, CAN_AddressBase+0x08+0x08)`  
*CAN acceptance filter 1/1 (page 2)*
- `#define _CAN_F1R2_SFR(uint8_t, CAN_AddressBase+0x08+0x09)`  
*CAN acceptance filter 1/2 (page 2)*
- `#define _CAN_F1R3_SFR(uint8_t, CAN_AddressBase+0x08+0x0A)`  
*CAN acceptance filter 1/3 (page 2)*
- `#define _CAN_F1R4_SFR(uint8_t, CAN_AddressBase+0x08+0x0B)`  
*CAN acceptance filter 1/4 (page 2)*
- `#define _CAN_F1R5_SFR(uint8_t, CAN_AddressBase+0x08+0x0C)`  
*CAN acceptance filter 1/5 (page 2)*
- `#define _CAN_F1R6_SFR(uint8_t, CAN_AddressBase+0x08+0x0D)`  
*CAN acceptance filter 1/6 (page 2)*
- `#define _CAN_F1R7_SFR(uint8_t, CAN_AddressBase+0x08+0x0E)`  
*CAN acceptance filter 1/7 (page 2)*
- `#define _CAN_F1R8_SFR(uint8_t, CAN_AddressBase+0x08+0x0F)`  
*CAN acceptance filter 1/8 (page 2)*
- `#define _CAN_F2R1_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`  
*CAN acceptance filter 2/1 (page 3)*
- `#define _CAN_F2R2_SFR(uint8_t, CAN_AddressBase+0x08+0x01)`  
*CAN acceptance filter 2/2 (page 3)*
- `#define _CAN_F2R3_SFR(uint8_t, CAN_AddressBase+0x08+0x02)`  
*CAN acceptance filter 2/3 (page 3)*
- `#define _CAN_F2R4_SFR(uint8_t, CAN_AddressBase+0x08+0x03)`  
*CAN acceptance filter 2/4 (page 3)*
- `#define _CAN_F2R5_SFR(uint8_t, CAN_AddressBase+0x08+0x04)`

- CAN acceptance filter 2/5 (page 3)*
- #define `_CAN_F2R6_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x05`)
- CAN acceptance filter 2/6 (page 3)*
- #define `_CAN_F2R7_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x06`)
- CAN acceptance filter 2/7 (page 3)*
- #define `_CAN_F2R8_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x07`)
- CAN acceptance filter 2/8 (page 3)*
- #define `_CAN_F3R1_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x08`)
- CAN acceptance filter 3/1 (page 3)*
- #define `_CAN_F3R2_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x09`)
- CAN acceptance filter 3/2 (page 3)*
- #define `_CAN_F3R3_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0A`)
- CAN acceptance filter 3/3 (page 3)*
- #define `_CAN_F3R4_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0B`)
- CAN acceptance filter 3/4 (page 3)*
- #define `_CAN_F3R5_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0C`)
- CAN acceptance filter 3/5 (page 3)*
- #define `_CAN_F3R6_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0D`)
- CAN acceptance filter 3/6 (page 3)*
- #define `_CAN_F3R7_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0E`)
- CAN acceptance filter 3/7 (page 3)*
- #define `_CAN_F3R8_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0F`)
- CAN acceptance filter 3/8 (page 3)*
- #define `_CAN_F4R1_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x00`)
- CAN acceptance filter 4/1 (page 4)*
- #define `_CAN_F4R2_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x01`)
- CAN acceptance filter 4/2 (page 4)*
- #define `_CAN_F4R3_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x02`)
- CAN acceptance filter 4/3 (page 4)*
- #define `_CAN_F4R4_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x03`)
- CAN acceptance filter 4/4 (page 4)*
- #define `_CAN_F4R5_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x04`)
- CAN acceptance filter 4/5 (page 4)*
- #define `_CAN_F4R6_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x05`)
- CAN acceptance filter 4/6 (page 4)*
- #define `_CAN_F4R7_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x06`)
- CAN acceptance filter 4/7 (page 4)*
- #define `_CAN_F4R8_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x07`)
- CAN acceptance filter 4/8 (page 4)*
- #define `_CAN_F5R1_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x08`)
- CAN acceptance filter 5/1 (page 4)*
- #define `_CAN_F5R2_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x09`)
- CAN acceptance filter 5/2 (page 4)*
- #define `_CAN_F5R3_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0A`)
- CAN acceptance filter 5/3 (page 4)*
- #define `_CAN_F5R4_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0B`)
- CAN acceptance filter 5/4 (page 4)*
- #define `_CAN_F5R5_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0C`)
- CAN acceptance filter 5/5 (page 4)*
- #define `_CAN_F5R6_SFR`(uint8\_t, `CAN_AddressBase+0x08+0x0D`)
- CAN acceptance filter 5/6 (page 4)*

- `#define _CAN_F5R7_SFR(uint8_t, CAN_AddressBase+0x08+0x0E)`  
*CAN acceptance filter 5/7 (page 4)*
- `#define _CAN_F5R8_SFR(uint8_t, CAN_AddressBase+0x08+0x0F)`  
*CAN acceptance filter 5/8 (page 4)*
- `#define _CAN_ESR_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`  
*CAN error status register (page 6)*
- `#define _CAN_EIER_SFR(uint8_t, CAN_AddressBase+0x08+0x01)`  
*CAN error interrupt enable register (page 6)*
- `#define _CAN_TECR_SFR(uint8_t, CAN_AddressBase+0x08+0x02)`  
*CAN transmit error counter register (page 6)*
- `#define _CAN_RECR_SFR(uint8_t, CAN_AddressBase+0x08+0x03)`  
*CAN receive error counter register (page 6)*
- `#define _CAN_BTR1_SFR(uint8_t, CAN_AddressBase+0x08+0x04)`  
*CAN bit timing register 1 (page 6)*
- `#define _CAN_BTR2_SFR(uint8_t, CAN_AddressBase+0x08+0x05)`  
*CAN bit timing register 2 (page 6)*
- `#define _CAN_FMR1_SFR(uint8_t, CAN_AddressBase+0x08+0x08)`  
*CAN filter mode register 1 (page 6)*
- `#define _CAN_FMR2_SFR(uint8_t, CAN_AddressBase+0x08+0x09)`  
*CAN filter mode register 2 (page 6)*
- `#define _CAN_FCR1_SFR(uint8_t, CAN_AddressBase+0x08+0x0A)`  
*CAN filter configuration register 1 (page 6)*
- `#define _CAN_FCR2_SFR(uint8_t, CAN_AddressBase+0x08+0x0B)`  
*CAN filter configuration register 2 (page 6)*
- `#define _CAN_FCR3_SFR(uint8_t, CAN_AddressBase+0x08+0x0C)`  
*CAN filter configuration register 3 (page 6)*
- `#define _CAN_MFMIR_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`  
*CAN mailbox filter match index register (page 7)*
- `#define _CAN_MCR_RESET_VALUE ((uint8_t) 0x02)`  
*CAN master control register reset value.*
- `#define _CAN_MSR_RESET_VALUE ((uint8_t) 0x02)`  
*CAN master status register reset value.*
- `#define _CAN_TSR_RESET_VALUE ((uint8_t) 0x00)`  
*CAN transmit status register reset value.*
- `#define _CAN_TPR_RESET_VALUE ((uint8_t) 0x0C)`  
*CAN transmit priority register reset value.*
- `#define _CAN_RFR_RESET_VALUE ((uint8_t) 0x00)`  
*CAN receive FIFO register reset value.*
- `#define _CAN_IER_RESET_VALUE ((uint8_t) 0x00)`  
*CAN interrupt enable register reset value.*
- `#define _CAN_DGR_RESET_VALUE ((uint8_t) 0x0C)`  
*CAN diagnosis register reset value.*
- `#define _CAN_PSR_RESET_VALUE ((uint8_t) 0x00)`  
*CAN page selection reset value.*
- `#define _CAN_MCSR_RESET_VALUE ((uint8_t) 0x00)`  
*CAN message control/status register (page 0,1,5) reset value.*
- `#define _CAN_MDLCR_RESET_VALUE ((uint8_t) 0x00)`  
*CAN mailbox data length control register (page 0,1,5,7) reset value.*
- `#define _CAN_ESR_RESET_VALUE ((uint8_t) 0x00)`  
*CAN error status register (page 6) reset value.*
- `#define _CAN_EIER_RESET_VALUE ((uint8_t) 0x00)`

- *CAN error interrupt enable register (page 6) reset value.*  
• #define `_CAN_TECR_RESET_VALUE` ((uint8\_t) 0x00)
- *CAN transmit error counter register (page 6) reset value.*  
• #define `_CAN_RECR_RESET_VALUE` ((uint8\_t) 0x00)
- *CAN receive error counter register (page 6) reset value.*  
• #define `_CAN_BTR1_RESET_VALUE` ((uint8\_t) 0x40)
- *CAN bit timing register 1 (page 6) reset value.*  
• #define `_CAN_BTR2_RESET_VALUE` ((uint8\_t) 0x23)
- *CAN bit timing register 2 (page 6) reset value.*  
• #define `_CAN_FMR1_RESET_VALUE` ((uint8\_t) 0x00)
- *CAN filter mode register 1 (page 6) reset value.*  
• #define `_CAN_FMR2_RESET_VALUE` ((uint8\_t) 0x00)
- *CAN filter mode register 2 (page 6) reset value.*  
• #define `_CAN_FCR_RESET_VALUE` ((uint8\_t) 0x00)
- *CAN filter configuration register reset value.*  
• #define `_CAN_MFMIR_RESET_VALUE` ((uint8\_t) 0x00)
- *CAN mailbox filter match index register reset value.*  
• #define `_CAN_INRQ` ((uint8\_t) (0x01 << 0))  
*CAN Channel Initialization Request [0] (in \_CAN\_MCR)*
- #define `_CAN_SLEEP` ((uint8\_t) (0x01 << 1))  
*CAN Channel Sleep Mode Request [0] (in \_CAN\_MCR)*
- #define `_CAN_TXFP` ((uint8\_t) (0x01 << 2))  
*CAN Channel Transmit FIFO Priority [0] (in \_CAN\_MCR)*
- #define `_CAN_RFLM` ((uint8\_t) (0x01 << 3))  
*CAN Channel Receive FIFO Locked Mode [0] (in \_CAN\_MCR)*
- #define `_CAN_NART` ((uint8\_t) (0x01 << 4))  
*CAN Channel No Automatic Retransmission [0] (in \_CAN\_MCR)*
- #define `_CAN_AWUM` ((uint8\_t) (0x01 << 5))  
*CAN Channel Automatic Wakeup Mode [0] (in \_CAN\_MCR)*
- #define `_CAN_ABOM` ((uint8\_t) (0x01 << 6))  
*CAN Channel Automatic Bus-Off Management [0] (in \_CAN\_MCR)*
- #define `_CAN_TTCM` ((uint8\_t) (0x01 << 7))  
*CAN Channel Time Triggered Communication Mode [0] (in \_CAN\_MCR)*
- #define `_CAN_INAK` ((uint8\_t) (0x01 << 0))  
*CAN Initialization Acknowledge [0] (in \_CAN\_MSR)*
- #define `_CAN_SLAK` ((uint8\_t) (0x01 << 1))  
*CAN Sleep Acknowledge [0] (in \_CAN\_MSR)*
- #define `_CAN_ERRI` ((uint8\_t) (0x01 << 2))  
*CAN Error Interrupt [0] (in \_CAN\_MSR)*
- #define `_CAN_WKUI` ((uint8\_t) (0x01 << 3))  
*CAN Wakeup Interrupt [0] (in \_CAN\_MSR)*
- #define `_CAN_TX` ((uint8\_t) (0x01 << 4))  
*CAN Transmit [0] (in \_CAN\_MSR)*
- #define `_CAN_RX` ((uint8\_t) (0x01 << 5))  
*CAN Receive [0] (in \_CAN\_MSR)*
- #define `_CAN_RQCP0` ((uint8\_t) (0x01 << 0))  
*CAN Request Completed for Mailbox 0 [0] (in \_CAN\_TSR)*
- #define `_CAN_RQCP1` ((uint8\_t) (0x01 << 1))  
*CAN Request Completed for Mailbox 1 [0] (in \_CAN\_TSR)*
- #define `_CAN_RQCP2` ((uint8\_t) (0x01 << 2))  
*CAN Request Completed for Mailbox 2 [0] (in \_CAN\_TSR)*

- `#define _CAN_TXOK0 ((uint8_t) (0x01 << 4))`  
*CAN Transmission ok for Mailbox 0 [0] (in \_CAN\_TSR)*
- `#define _CAN_TXOK1 ((uint8_t) (0x01 << 5))`  
*CAN Transmission ok for Mailbox 1 [0] (in \_CAN\_TSR)*
- `#define _CAN_TXOK2 ((uint8_t) (0x01 << 6))`  
*CAN Transmission ok for Mailbox 2 [0] (in \_CAN\_TSR)*
- `#define _CAN_CODE ((uint8_t) (0x03 << 0))`  
*CAN Mailbox Code [1:0] (in \_CAN\_TPR)*
- `#define _CAN_CODE0 ((uint8_t) (0x01 << 0))`  
*CAN Mailbox Code [0] (in \_CAN\_TPR)*
- `#define _CAN_CODE1 ((uint8_t) (0x01 << 1))`  
*CAN Mailbox Code [1] (in \_CAN\_TPR)*
- `#define _CAN_TME0 ((uint8_t) (0x01 << 2))`  
*CAN Transmit Mailbox 0 Empty [0] (in \_CAN\_TPR)*
- `#define _CAN_TME1 ((uint8_t) (0x01 << 3))`  
*CAN Transmit Mailbox 1 Empty [0] (in \_CAN\_TPR)*
- `#define _CAN_TME2 ((uint8_t) (0x01 << 4))`  
*CAN Transmit Mailbox 2 Empty [0] (in \_CAN\_TPR)*
- `#define _CAN_LOW0 ((uint8_t) (0x01 << 5))`  
*CAN Lowest Priority Flag for Mailbox 0 [0] (in \_CAN\_TPR)*
- `#define _CAN_LOW1 ((uint8_t) (0x01 << 6))`  
*CAN Lowest Priority Flag for Mailbox 1 [0] (in \_CAN\_TPR)*
- `#define _CAN_LOW2 ((uint8_t) (0x01 << 7))`  
*CAN Lowest Priority Flag for Mailbox 2 [0] (in \_CAN\_TPR)*
- `#define _CAN_FMP ((uint8_t) (0x03 << 0))`  
*CAN FIFO Message Pending [1:0] (in \_CAN\_RFR)*
- `#define _CAN_FMP0 ((uint8_t) (0x01 << 0))`  
*CAN FIFO Message Pending [0] (in \_CAN\_RFR)*
- `#define _CAN_FMP1 ((uint8_t) (0x01 << 1))`  
*CAN FIFO Message Pending [1] (in \_CAN\_RFR)*
- `#define _CAN_FULL ((uint8_t) (0x01 << 3))`  
*CAN FIFO Full [0] (in \_CAN\_RFR)*
- `#define _CAN_FOVR ((uint8_t) (0x01 << 4))`  
*CAN FIFO Overrun [0] (in \_CAN\_RFR)*
- `#define _CAN_RFOM ((uint8_t) (0x01 << 5))`  
*CAN Release FIFO Output Mailbox [0] (in \_CAN\_RFR)*
- `#define _CAN_TMEIE ((uint8_t) (0x01 << 0))`  
*CAN Transmit Mailbox Empty Interrupt Enable [0] (in \_CAN\_IER)*
- `#define _CAN_FMPIE ((uint8_t) (0x01 << 1))`  
*CAN FIFO Message Pending Interrupt Enable [0] (in \_CAN\_IER)*
- `#define _CAN_FFIE ((uint8_t) (0x01 << 2))`  
*CAN FIFO Full Interrupt Enable [0] (in \_CAN\_IER)*
- `#define _CAN_FOVIE ((uint8_t) (0x01 << 3))`  
*CAN FIFO Overrun Interrupt Enable [0] (in \_CAN\_IER)*
- `#define _CAN_WKUIE ((uint8_t) (0x01 << 7))`  
*CAN Wakeup Interrupt Enable [0] (in \_CAN\_IER)*
- `#define _CAN_LBKM ((uint8_t) (0x01 << 0))`  
*CAN Loop back mode [0] (in \_CAN\_DGR)*
- `#define _CAN_SILM ((uint8_t) (0x01 << 1))`  
*CAN Silent mode [0] (in \_CAN\_DGR)*
- `#define _CAN_SAMP ((uint8_t) (0x01 << 2))`



- *CAN Last sample point [0] (in \_CAN\_DGR)*  
• #define `_CAN_RXS` ((uint8\_t) (0x01 << 3))
- *CAN Rx Signal (=pin status) [0] (in \_CAN\_DGR)*  
• #define `_CAN_TXM2E` ((uint8\_t) (0x01 << 4))
- *CAN TX Mailbox 2 enable [0] (in \_CAN\_DGR)*  
• #define `_CAN_PS` ((uint8\_t) (0x07 << 0))
- *CAN Page select [2:0] (in \_CAN\_PSR)*  
• #define `_CAN_PS0` ((uint8\_t) (0x01 << 0))
- *CAN Page select [0] (in \_CAN\_PSR)*  
• #define `_CAN_PS1` ((uint8\_t) (0x01 << 1))
- *CAN Page select [1] (in \_CAN\_PSR)*  
• #define `_CAN_PS2` ((uint8\_t) (0x01 << 2))
- *CAN Page select [2] (in \_CAN\_PSR)*  
• #define `_CAN_TXRQ` ((uint8\_t) (0x01 << 0))
- *CAN Transmission mailbox request [0] (in \_CAN\_MCSR, page 0,1,5)*  
• #define `_CAN_ABRQ` ((uint8\_t) (0x01 << 1))
- *CAN Abort request for mailbox [0] (in \_CAN\_MCSR, page 0,1,5)*  
• #define `_CAN_RQCP` ((uint8\_t) (0x01 << 2))
- *CAN Request completed [0] (in \_CAN\_MCSR, page 0,1,5)*  
• #define `_CAN_TXOK` ((uint8\_t) (0x01 << 3))
- *CAN Transmission OK [0] (in \_CAN\_MCSR, page 0,1,5)*  
• #define `_CAN_ALST` ((uint8\_t) (0x01 << 4))
- *CAN Arbitration lost [0] (in \_CAN\_MCSR, page 0,1,5)*  
• #define `_CAN_TERR` ((uint8\_t) (0x01 << 5))
- *CAN Transmission error [0] (in \_CAN\_MCSR, page 0,1,5)*  
• #define `_CAN_DLC` ((uint8\_t) (0x0F << 0))
- *CAN Data length code [3:0] (in \_CAN\_MDLCR, page 0,1,5,7)*  
• #define `_CAN_DLC0` ((uint8\_t) (0x01 << 0))
- *CAN Data length code [0] (in \_CAN\_MDLCR, page 0,1,5,7)*  
• #define `_CAN_DLC1` ((uint8\_t) (0x01 << 1))
- *CAN Data length code [1] (in \_CAN\_MDLCR, page 0,1,5,7)*  
• #define `_CAN_DLC2` ((uint8\_t) (0x01 << 2))
- *CAN Data length code [2] (in \_CAN\_MDLCR, page 0,1,5,7)*  
• #define `_CAN_DLC3` ((uint8\_t) (0x01 << 3))
- *CAN Data length code [3] (in \_CAN\_MDLCR, page 0,1,5,7)*  
• #define `_CAN_TGT` ((uint8\_t) (0x01 << 7))
- *CAN Transmit global time [0] (in \_CAN\_MDLCR, page 0,1,5,7)*  
• #define `_CAN_RTR` ((uint8\_t) (0x01 << 5))
- *CAN Remote transmission request [0] (in \_CAN\_MIDR1, page 0,1,5)*  
• #define `_CAN_IDE` ((uint8\_t) (0x01 << 6))
- *CAN Extended identifier [0] (in \_CAN\_MIDR1, page 0,1,5)*  
• #define `_CAN_EWGF` ((uint8\_t) (0x01 << 0))
- *CAN Error warning flag [0] (in \_CAN\_ESR, page 6)*  
• #define `_CAN_EPVF` ((uint8\_t) (0x01 << 1))
- *CAN Error passive flag [0] (in \_CAN\_ESR, page 6)*  
• #define `_CAN_BOFF` ((uint8\_t) (0x01 << 2))
- *CAN Bus off flag [0] (in \_CAN\_ESR, page 6)*  
• #define `_CAN_LEC` ((uint8\_t) (0x07 << 4))
- *CAN Last error code [2:0] (in \_CAN\_ESR, page 6)*  
• #define `_CAN_LEC0` ((uint8\_t) (0x01 << 4))
- *CAN Last error code [0] (in \_CAN\_ESR, page 6)*

- `#define _CAN_LEC1 ((uint8_t) (0x01 << 5))`  
CAN Last error code [1] (in `_CAN_ESR`, page 6)
- `#define _CAN_LEC2 ((uint8_t) (0x01 << 6))`  
CAN Last error code [3] (in `_CAN_ESR`, page 6)
- `#define _CAN_EWGIE ((uint8_t) (0x01 << 0))`  
CAN Error warning interrupt enable [0] (in `_CAN_EIER`, page 6)
- `#define _CAN_EPVIE ((uint8_t) (0x01 << 1))`  
CAN Error passive interrupt enable [0] (in `_CAN_EIER`, page 6)
- `#define _CAN_BOFIE ((uint8_t) (0x01 << 2))`  
CAN Bus-Off interrupt enable [0] (in `_CAN_EIER`, page 6)
- `#define _CAN_LECIE ((uint8_t) (0x01 << 4))`  
CAN Last error code interrupt enable [0] (in `_CAN_EIER`, page 6)
- `#define _CAN_ERRIE ((uint8_t) (0x01 << 6))`  
CAN Error interrupt enable [0] (in `_CAN_EIER`, page 6)
- `#define _CAN_BRP ((uint8_t) (0x3F << 0))`  
CAN Baud rate prescaler [5:0] (in `_CAN_BTR1`, page 6)
- `#define _CAN_BRP0 ((uint8_t) (0x01 << 0))`  
CAN Baud rate prescaler [0] (in `_CAN_BTR1`, page 6)
- `#define _CAN_BRP1 ((uint8_t) (0x01 << 1))`  
CAN Baud rate prescaler [1] (in `_CAN_BTR1`, page 6)
- `#define _CAN_BRP2 ((uint8_t) (0x01 << 2))`  
CAN Baud rate prescaler [2] (in `_CAN_BTR1`, page 6)
- `#define _CAN_BRP3 ((uint8_t) (0x01 << 3))`  
CAN Baud rate prescaler [3] (in `_CAN_BTR1`, page 6)
- `#define _CAN_BRP4 ((uint8_t) (0x01 << 4))`  
CAN Baud rate prescaler [4] (in `_CAN_BTR1`, page 6)
- `#define _CAN_BRP5 ((uint8_t) (0x01 << 5))`  
CAN Baud rate prescaler [5] (in `_CAN_BTR1`, page 6)
- `#define _CAN_SJW ((uint8_t) (0x03 << 6))`  
CAN Resynchronization jump width [1:0] (in `_CAN_EIER`, page 6)
- `#define _CAN_SJW0 ((uint8_t) (0x01 << 6))`  
CAN Resynchronization jump width [0] (in `_CAN_EIER`, page 6)
- `#define _CAN_SJW1 ((uint8_t) (0x01 << 7))`  
CAN Resynchronization jump width [1] (in `_CAN_EIER`, page 6)
- `#define _CAN_BS1 ((uint8_t) (0x0F << 0))`  
CAN Bit segment 1 [3:0] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS10 ((uint8_t) (0x01 << 0))`  
CAN Bit segment 1 [0] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS11 ((uint8_t) (0x01 << 1))`  
CAN Bit segment 1 [1] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS12 ((uint8_t) (0x01 << 2))`  
CAN Bit segment 1 [2] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS13 ((uint8_t) (0x01 << 3))`  
CAN Bit segment 1 [3] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS2 ((uint8_t) (0x07 << 4))`  
CAN Bit segment 2 [2:0] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS20 ((uint8_t) (0x01 << 4))`  
CAN Bit segment 2 [0] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS21 ((uint8_t) (0x01 << 5))`  
CAN Bit segment 2 [1] (in `_CAN_BTR2`, page 6)
- `#define _CAN_BS22 ((uint8_t) (0x01 << 6))`



- CAN Bit segment 2 [2] (in \_CAN\_BTR2, page 6)*
- #define `_CAN_FML0` ((uint8\_t) (0x01 << 0))
- CAN Filter 0 mode low [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FMH0` ((uint8\_t) (0x01 << 1))
- CAN Filter 0 mode high [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FML1` ((uint8\_t) (0x01 << 2))
- CAN Filter 1 mode low [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FMH1` ((uint8\_t) (0x01 << 3))
- CAN Filter 1 mode high [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FML2` ((uint8\_t) (0x01 << 4))
- CAN Filter 2 mode low [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FMH2` ((uint8\_t) (0x01 << 5))
- CAN Filter 2 mode high [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FML3` ((uint8\_t) (0x01 << 6))
- CAN Filter 3 mode low [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FMH3` ((uint8\_t) (0x01 << 7))
- CAN Filter 3 mode high [0] (in \_CAN\_FMR1, page 6)*
- #define `_CAN_FML4` ((uint8\_t) (0x01 << 0))
- CAN Filter 4 mode low [0] (in \_CAN\_FMR2, page 6)*
- #define `_CAN_FMH4` ((uint8\_t) (0x01 << 1))
- CAN Filter 4 mode high [0] (in \_CAN\_FMR2, page 6)*
- #define `_CAN_FML5` ((uint8\_t) (0x01 << 2))
- CAN Filter 5 mode low [0] (in \_CAN\_FMR2, page 6)*
- #define `_CAN_FMH5` ((uint8\_t) (0x01 << 3))
- CAN Filter 5 mode high [0] (in \_CAN\_FMR2, page 6)*
- #define `_CAN_FACT0` ((uint8\_t) (0x01 << 0))
- CAN Filter 0 active [0] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FSC0` ((uint8\_t) (0x03 << 1))
- CAN Filter 0 scale configuration [1:0] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FSC00` ((uint8\_t) (0x01 << 1))
- CAN Filter 0 scale configuration [0] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FSC01` ((uint8\_t) (0x01 << 2))
- CAN Filter 0 scale configuration [1] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FACT1` ((uint8\_t) (0x01 << 4))
- CAN Filter 1 active [0] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FSC1` ((uint8\_t) (0x03 << 5))
- CAN Filter 1 scale configuration [1:0] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FSC10` ((uint8\_t) (0x01 << 5))
- CAN Filter 1 scale configuration [0] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FSC11` ((uint8\_t) (0x01 << 6))
- CAN Filter 1 scale configuration [1] (in \_CAN\_FCR1, page 6)*
- #define `_CAN_FACT2` ((uint8\_t) (0x01 << 0))
- CAN Filter 2 active [0] (in \_CAN\_FCR2, page 6)*
- #define `_CAN_FSC2` ((uint8\_t) (0x03 << 1))
- CAN Filter 2 scale configuration [1:0] (in \_CAN\_FCR2, page 6)*
- #define `_CAN_FSC20` ((uint8\_t) (0x01 << 1))
- CAN Filter 2 scale configuration [0] (in \_CAN\_FCR2, page 6)*
- #define `_CAN_FSC21` ((uint8\_t) (0x01 << 2))
- CAN Filter 2 scale configuration [1] (in \_CAN\_FCR2, page 6)*
- #define `_CAN_FACT3` ((uint8\_t) (0x01 << 4))
- CAN Filter 3 active [0] (in \_CAN\_FCR2, page 6)*

- `#define _CAN_FSC3 ((uint8_t) (0x03 << 5))`  
CAN Filter 3 scale configuration [1:0] (in `_CAN_FCR2`, page 6)
- `#define _CAN_FSC30 ((uint8_t) (0x01 << 5))`  
CAN Filter 3 scale configuration [0] (in `_CAN_FCR2`, page 6)
- `#define _CAN_FSC31 ((uint8_t) (0x01 << 6))`  
CAN Filter 3 scale configuration [1] (in `_CAN_FCR2`, page 6)
- `#define _CAN_FACT4 ((uint8_t) (0x01 << 0))`  
CAN Filter 4 active [0] (in `_CAN_FCR3`, page 6)
- `#define _CAN_FSC4 ((uint8_t) (0x03 << 1))`  
CAN Filter 4 scale configuration [1:0] (in `_CAN_FCR3`, page 6)
- `#define _CAN_FSC40 ((uint8_t) (0x01 << 1))`  
CAN Filter 4 scale configuration [0] (in `_CAN_FCR3`, page 6)
- `#define _CAN_FSC41 ((uint8_t) (0x01 << 2))`  
CAN Filter 4 scale configuration [1] (in `_CAN_FCR3`, page 6)
- `#define _CAN_FACT5 ((uint8_t) (0x01 << 4))`  
CAN Filter 5 active [0] (in `_CAN_FCR2`, page 6)
- `#define _CAN_FSC5 ((uint8_t) (0x03 << 5))`  
CAN Filter 5 scale configuration [1:0] (in `_CAN_FCR3`, page 6)
- `#define _CAN_FSC50 ((uint8_t) (0x01 << 5))`  
CAN Filter 5 scale configuration [0] (in `_CAN_FCR3`, page 6)
- `#define _CAN_FSC51 ((uint8_t) (0x01 << 6))`  
CAN Filter 5 scale configuration [1] (in `_CAN_FCR3`, page 6)
- `#define _CFG_SFR(CFG_t, CFG_AddressBase)`  
CFG struct/bit access.
- `#define _CFG_GCR_SFR(uint8_t, CFG_AddressBase+0x00)`  
Global configuration register (`CFG_GCR`)
- `#define _CFG_GCR_RESET_VALUE ((uint8_t)0x00)`
- `#define _CFG_SWD ((uint8_t) (0x01 << 0))`  
SWIM disable [0].
- `#define _CFG_AL ((uint8_t) (0x01 << 1))`  
Activation level [0].
- `#define _ITC_SFR(ITC_t, ITC_AddressBase)`  
ITC struct/bit access.
- `#define _ITC_SPR1_SFR(uint8_t, ITC_AddressBase+0x00)`  
Interrupt priority register 1/8.
- `#define _ITC_SPR2_SFR(uint8_t, ITC_AddressBase+0x01)`  
Interrupt priority register 2/8.
- `#define _ITC_SPR3_SFR(uint8_t, ITC_AddressBase+0x02)`  
Interrupt priority register 3/8.
- `#define _ITC_SPR4_SFR(uint8_t, ITC_AddressBase+0x03)`  
Interrupt priority register 4/8.
- `#define _ITC_SPR5_SFR(uint8_t, ITC_AddressBase+0x04)`  
Interrupt priority register 5/8.
- `#define _ITC_SPR6_SFR(uint8_t, ITC_AddressBase+0x05)`  
Interrupt priority register 6/8.
- `#define _ITC_SPR7_SFR(uint8_t, ITC_AddressBase+0x06)`  
Interrupt priority register 7/8.
- `#define _ITC_SPR8_SFR(uint8_t, ITC_AddressBase+0x07)`  
Interrupt priority register 8/8.
- `#define _ITC_SPR1_RESET_VALUE ((uint8_t) 0xFF)`  
Interrupt priority register 1/8 reset value.

- `#define _ITC_SPR2_RESET_VALUE ((uint8_t) 0xFF)`  
*Interrupt priority register 2/8 reset value.*
- `#define _ITC_SPR3_RESET_VALUE ((uint8_t) 0xFF)`  
*Interrupt priority register 3/8 reset value.*
- `#define _ITC_SPR4_RESET_VALUE ((uint8_t) 0xFF)`  
*Interrupt priority register 4/8 reset value.*
- `#define _ITC_SPR5_RESET_VALUE ((uint8_t) 0xFF)`  
*Interrupt priority register 5/8 reset value.*
- `#define _ITC_SPR6_RESET_VALUE ((uint8_t) 0xFF)`  
*Interrupt priority register 6/8 reset value.*
- `#define _ITC_SPR7_RESET_VALUE ((uint8_t) 0xFF)`  
*Interrupt priority register 7/8 reset value.*
- `#define _ITC_SPR8_RESET_VALUE ((uint8_t) 0x0F)`  
*Interrupt priority register 8/8 reset value.*
- `#define _ITC_VECT1SPR ((uint8_t) (0x03 << 2))`  
*ITC interrupt priority vector 1 [1:0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT1SPR0 ((uint8_t) (0x01 << 2))`  
*ITC interrupt priority vector 1 [0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT1SPR1 ((uint8_t) (0x01 << 3))`  
*ITC interrupt priority vector 1 [1] (in \_ITC\_SPR1)*
- `#define _ITC_VECT2SPR ((uint8_t) (0x03 << 4))`  
*ITC interrupt priority vector 2 [1:0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT2SPR0 ((uint8_t) (0x01 << 4))`  
*ITC interrupt priority vector 2 [0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT2SPR1 ((uint8_t) (0x01 << 5))`  
*ITC interrupt priority vector 2 [1] (in \_ITC\_SPR1)*
- `#define _ITC_VECT3SPR ((uint8_t) (0x03 << 6))`  
*ITC interrupt priority vector 3 [1:0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT3SPR0 ((uint8_t) (0x01 << 6))`  
*ITC interrupt priority vector 3 [0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT3SPR1 ((uint8_t) (0x01 << 7))`  
*ITC interrupt priority vector 3 [1] (in \_ITC\_SPR1)*
- `#define _ITC_VECT4SPR ((uint8_t) (0x03 << 0))`  
*ITC interrupt priority vector 4 [1:0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT4SPR0 ((uint8_t) (0x01 << 0))`  
*ITC interrupt priority vector 4 [0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT4SPR1 ((uint8_t) (0x01 << 1))`  
*ITC interrupt priority vector 4 [1] (in \_ITC\_SPR2)*
- `#define _ITC_VECT5SPR ((uint8_t) (0x03 << 2))`  
*ITC interrupt priority vector 5 [1:0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT5SPR0 ((uint8_t) (0x01 << 2))`  
*ITC interrupt priority vector 5 [0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT5SPR1 ((uint8_t) (0x01 << 3))`  
*ITC interrupt priority vector 5 [1] (in \_ITC\_SPR2)*
- `#define _ITC_VECT6SPR ((uint8_t) (0x03 << 4))`  
*ITC interrupt priority vector 6 [1:0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT6SPR0 ((uint8_t) (0x01 << 4))`  
*ITC interrupt priority vector 6 [0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT6SPR1 ((uint8_t) (0x01 << 5))`  
*ITC interrupt priority vector 6 [1] (in \_ITC\_SPR2)*
- `#define _ITC_VECT7SPR ((uint8_t) (0x03 << 6))`

```

    ITC interrupt priority vector 7 [1:0] (in _ITC_SPR2)
    • #define _ITC_VECT7SPR0 ((uint8_t) (0x01 << 6))
    ITC interrupt priority vector 7 [0] (in _ITC_SPR2)
    • #define _ITC_VECT7SPR1 ((uint8_t) (0x01 << 7))
    ITC interrupt priority vector 7 [1] (in _ITC_SPR2)
    • #define _ITC_VECT8SPR ((uint8_t) (0x03 << 0))
    ITC interrupt priority vector 8 [1:0] (in _ITC_SPR3)
    • #define _ITC_VECT8SPR0 ((uint8_t) (0x01 << 0))
    ITC interrupt priority vector 8 [0] (in _ITC_SPR3)
    • #define _ITC_VECT8SPR1 ((uint8_t) (0x01 << 1))
    ITC interrupt priority vector 8 [1] (in _ITC_SPR3)
    • #define _ITC_VECT9SPR ((uint8_t) (0x03 << 2))
    ITC interrupt priority vector 9 [1:0] (in _ITC_SPR3)
    • #define _ITC_VECT9SPR0 ((uint8_t) (0x01 << 2))
    ITC interrupt priority vector 9 [0] (in _ITC_SPR3)
    • #define _ITC_VECT9SPR1 ((uint8_t) (0x01 << 3))
    ITC interrupt priority vector 9 [1] (in _ITC_SPR3)
    • #define _ITC_VECT10SPR ((uint8_t) (0x03 << 4))
    ITC interrupt priority vector 10 [1:0] (in _ITC_SPR3)
    • #define _ITC_VECT10SPR0 ((uint8_t) (0x01 << 4))
    ITC interrupt priority vector 10 [0] (in _ITC_SPR3)
    • #define _ITC_VECT10SPR1 ((uint8_t) (0x01 << 5))
    ITC interrupt priority vector 10 [1] (in _ITC_SPR3)
    • #define _ITC_VECT11SPR ((uint8_t) (0x03 << 6))
    ITC interrupt priority vector 11 [1:0] (in _ITC_SPR3)
    • #define _ITC_VECT11SPR0 ((uint8_t) (0x01 << 6))
    ITC interrupt priority vector 11 [0] (in _ITC_SPR3)
    • #define _ITC_VECT11SPR1 ((uint8_t) (0x01 << 7))
    ITC interrupt priority vector 11 [1] (in _ITC_SPR3)
    • #define _ITC_VECT12SPR ((uint8_t) (0x03 << 0))
    ITC interrupt priority vector 12 [1:0] (in _ITC_SPR4)
    • #define _ITC_VECT12SPR0 ((uint8_t) (0x01 << 0))
    ITC interrupt priority vector 12 [0] (in _ITC_SPR4)
    • #define _ITC_VECT12SPR1 ((uint8_t) (0x01 << 1))
    ITC interrupt priority vector 12 [1] (in _ITC_SPR4)
    • #define _ITC_VECT13SPR ((uint8_t) (0x03 << 2))
    ITC interrupt priority vector 13 [1:0] (in _ITC_SPR4)
    • #define _ITC_VECT13SPR0 ((uint8_t) (0x01 << 2))
    ITC interrupt priority vector 13 [0] (in _ITC_SPR4)
    • #define _ITC_VECT13SPR1 ((uint8_t) (0x01 << 3))
    ITC interrupt priority vector 13 [1] (in _ITC_SPR4)
    • #define _ITC_VECT14SPR ((uint8_t) (0x03 << 4))
    ITC interrupt priority vector 14 [1:0] (in _ITC_SPR4)
    • #define _ITC_VECT14SPR0 ((uint8_t) (0x01 << 4))
    ITC interrupt priority vector 14 [0] (in _ITC_SPR4)
    • #define _ITC_VECT14SPR1 ((uint8_t) (0x01 << 5))
    ITC interrupt priority vector 14 [1] (in _ITC_SPR4)
    • #define _ITC_VECT15SPR ((uint8_t) (0x03 << 6))
    ITC interrupt priority vector 15 [1:0] (in _ITC_SPR4)
    • #define _ITC_VECT15SPR0 ((uint8_t) (0x01 << 6))
    ITC interrupt priority vector 15 [0] (in _ITC_SPR4)

```

- `#define _ITC_VECT15SPR1 ((uint8_t) (0x01 << 7))`  
*ITC interrupt priority vector 15 [1] (in \_ITC\_SPR4)*
- `#define _ITC_VECT16SPR ((uint8_t) (0x03 << 0))`  
*ITC interrupt priority vector 16 [1:0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT16SPR0 ((uint8_t) (0x01 << 0))`  
*ITC interrupt priority vector 16 [0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT16SPR1 ((uint8_t) (0x01 << 1))`  
*ITC interrupt priority vector 16 [1] (in \_ITC\_SPR5)*
- `#define _ITC_VECT17SPR ((uint8_t) (0x03 << 2))`  
*ITC interrupt priority vector 17 [1:0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT17SPR0 ((uint8_t) (0x01 << 2))`  
*ITC interrupt priority vector 17 [0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT17SPR1 ((uint8_t) (0x01 << 3))`  
*ITC interrupt priority vector 17 [1] (in \_ITC\_SPR5)*
- `#define _ITC_VECT18SPR ((uint8_t) (0x03 << 4))`  
*ITC interrupt priority vector 18 [1:0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT18SPR0 ((uint8_t) (0x01 << 4))`  
*ITC interrupt priority vector 18 [0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT18SPR1 ((uint8_t) (0x01 << 5))`  
*ITC interrupt priority vector 18 [1] (in \_ITC\_SPR5)*
- `#define _ITC_VECT19SPR ((uint8_t) (0x03 << 6))`  
*ITC interrupt priority vector 19 [1:0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT19SPR0 ((uint8_t) (0x01 << 6))`  
*ITC interrupt priority vector 19 [0] (in \_ITC\_SPR5)*
- `#define _ITC_VECT19SPR1 ((uint8_t) (0x01 << 7))`  
*ITC interrupt priority vector 19 [1] (in \_ITC\_SPR5)*
- `#define _ITC_VECT20SPR ((uint8_t) (0x03 << 0))`  
*ITC interrupt priority vector 20 [1:0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT20SPR0 ((uint8_t) (0x01 << 0))`  
*ITC interrupt priority vector 20 [0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT20SPR1 ((uint8_t) (0x01 << 1))`  
*ITC interrupt priority vector 20 [1] (in \_ITC\_SPR6)*
- `#define _ITC_VECT21SPR ((uint8_t) (0x03 << 2))`  
*ITC interrupt priority vector 21 [1:0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT21SPR0 ((uint8_t) (0x01 << 2))`  
*ITC interrupt priority vector 21 [0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT21SPR1 ((uint8_t) (0x01 << 3))`  
*ITC interrupt priority vector 21 [1] (in \_ITC\_SPR6)*
- `#define _ITC_VECT22SPR ((uint8_t) (0x03 << 4))`  
*ITC interrupt priority vector 22 [1:0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT22SPR0 ((uint8_t) (0x01 << 4))`  
*ITC interrupt priority vector 22 [0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT22SPR1 ((uint8_t) (0x01 << 5))`  
*ITC interrupt priority vector 22 [1] (in \_ITC\_SPR6)*
- `#define _ITC_VECT23SPR ((uint8_t) (0x03 << 6))`  
*ITC interrupt priority vector 23 [1:0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT23SPR0 ((uint8_t) (0x01 << 6))`  
*ITC interrupt priority vector 23 [0] (in \_ITC\_SPR6)*
- `#define _ITC_VECT23SPR1 ((uint8_t) (0x01 << 7))`  
*ITC interrupt priority vector 23 [1] (in \_ITC\_SPR6)*
- `#define _ITC_VECT24SPR ((uint8_t) (0x03 << 0))`

```

        ITC interrupt priority vector 24 [1:0] (in _ITC_SPR7)
    • #define _ITC_VECT24SPR0 ((uint8_t) (0x01 << 0))
        ITC interrupt priority vector 24 [0] (in _ITC_SPR7)
    • #define _ITC_VECT24SPR1 ((uint8_t) (0x01 << 1))
        ITC interrupt priority vector 24 [1] (in _ITC_SPR7)
    • #define _ITC_VECT25SPR ((uint8_t) (0x03 << 2))
        ITC interrupt priority vector 25 [1:0] (in _ITC_SPR7)
    • #define _ITC_VECT25SPR0 ((uint8_t) (0x01 << 2))
        ITC interrupt priority vector 25 [0] (in _ITC_SPR7)
    • #define _ITC_VECT25SPR1 ((uint8_t) (0x01 << 3))
        ITC interrupt priority vector 25 [1] (in _ITC_SPR7)
    • #define _ITC_VECT26SPR ((uint8_t) (0x03 << 4))
        ITC interrupt priority vector 26 [1:0] (in _ITC_SPR7)
    • #define _ITC_VECT26SPR0 ((uint8_t) (0x01 << 4))
        ITC interrupt priority vector 26 [0] (in _ITC_SPR7)
    • #define _ITC_VECT26SPR1 ((uint8_t) (0x01 << 5))
        ITC interrupt priority vector 26 [1] (in _ITC_SPR7)
    • #define _ITC_VECT27SPR ((uint8_t) (0x03 << 6))
        ITC interrupt priority vector 27 [1:0] (in _ITC_SPR7)
    • #define _ITC_VECT27SPR0 ((uint8_t) (0x01 << 6))
        ITC interrupt priority vector 27 [0] (in _ITC_SPR7)
    • #define _ITC_VECT27SPR1 ((uint8_t) (0x01 << 7))
        ITC interrupt priority vector 27 [1] (in _ITC_SPR7)
    • #define _ITC_VECT28SPR ((uint8_t) (0x03 << 0))
        ITC interrupt priority vector 28 [1:0] (in _ITC_SPR8)
    • #define _ITC_VECT28SPR0 ((uint8_t) (0x01 << 0))
        ITC interrupt priority vector 28 [0] (in _ITC_SPR8)
    • #define _ITC_VECT28SPR1 ((uint8_t) (0x01 << 1))
        ITC interrupt priority vector 28 [1] (in _ITC_SPR8)
    • #define _ITC_VECT29SPR ((uint8_t) (0x03 << 2))
        ITC interrupt priority vector 29 [1:0] (in _ITC_SPR8)
    • #define _ITC_VECT29SPR0 ((uint8_t) (0x01 << 2))
        ITC interrupt priority vector 29 [0] (in _ITC_SPR8)
    • #define _ITC_VECT29SPR1 ((uint8_t) (0x01 << 3))
        ITC interrupt priority vector 29 [1] (in _ITC_SPR8)
    • #define STM8S001J3
    • #define STM8S001
    • #define STM8_PFLASH_SIZE 8*1024
    • #define STM8_RAM_SIZE 1*1024
    • #define STM8_EEPROM_SIZE 128
    • #define OPT_AddressBase 0x4800
    • #define PORTA_AddressBase 0x5000
    • #define PORTB_AddressBase 0x5005
    • #define PORTC_AddressBase 0x500A
    • #define PORTD_AddressBase 0x500F
    • #define PORTE_AddressBase 0x5014
    • #define PORTF_AddressBase 0x5019
    • #define FLASH_AddressBase 0x505A
    • #define EXTI_AddressBase 0x50A0
    • #define RST_AddressBase 0x50B3
    • #define CLK_AddressBase 0x50C0
    • #define WWDG_AddressBase 0x50D1

```

- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8S003F3
- #define STM8S003
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 128
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8S003K3
- #define STM8S003
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 128
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019



- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8S005C6
- #define STM8S005
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 128
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8S005K6
- #define STM8S005
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 128



- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8S007C8
- #define STM8S007
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 128
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250

- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define STM8S103F2
- #define STM8S103
- #define STM8\_PFLASH\_SIZE 4\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x4865
- #define STM8S103F3
- #define STM8S103
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1

- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x4865
- #define STM8S103K3
- #define STM8S103
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x4865
- #define STM8S105C4
- #define STM8S105
- #define STM8\_PFLASH\_SIZE 16\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F

- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD
- #define [STM8S105C6](#)
- #define [STM8S105](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 32\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 1024
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

- #define UID\_AddressBase 0x48CD
- #define STM8S105K4
- #define STM8S105
- #define STM8\_PFLASH\_SIZE 16\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART2\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S105K6
- #define STM8S105
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3

- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`
- `#define STM8S105S4`
- `#define STM8S105`
- `#define STM8_PFLASH_SIZE 16*1024`
- `#define STM8_RAM_SIZE 2*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`
- `#define STM8S105S6`
- `#define STM8S105`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 2*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`

- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD
- #define [STM8S207C6](#)
- #define [STM8S207](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 32\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 1024
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400

- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`
- `#define STM8S207C8`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1536`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`
- `#define STM8S207CB`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`



- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S207K6
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70

- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD
- #define [STM8S207K8](#)
- #define [STM8S207](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 64\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 1024
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD
- #define [STM8S207M8](#)
- #define [STM8S207](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 64\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0

- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S207MB
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

- `#define STM8S207R6`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`
- `#define STM8S207R8`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1536`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`

- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S207RB
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S207S6
- #define STM8S207

- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S207S8
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0

- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S207SB
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S208C6
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024

- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD
- #define [STM8S208C8](#)
- #define [STM8S208](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 64\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0



- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S208CB
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S208MB
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 128\*1024

- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`
- `#define STM8S208R6`
- `#define STM8S208`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`

- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S208R8
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S208RB
- #define STM8S208

- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`
- `#define STM8S208S6`
- `#define STM8S208`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1536`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`

- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S208S8
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S208SB

- #define STM8S208
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD
- #define STM8S903F3
- #define STM8S903
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0

- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM5\_AddressBase 0x5300
- #define TIM6\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x4865
- #define STM8S903K3
- #define STM8S903
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM5\_AddressBase 0x5300
- #define TIM6\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x4865

### 5.1.1 Detailed Description

### 5.1.2 Macro Definition Documentation

#### 5.1.2.1 \_\_ADC1\_VECTOR\_\_

```
#define __ADC1_VECTOR__ 22
```

irq22 - ADC1 end of conversion (shared with \_\_ADC2\_VECTOR\_\_)

Definition at line 301 of file STM8AF\_STM8S.h.

#### 5.1.2.2 \_\_AWU\_VECTOR\_\_

```
#define __AWU_VECTOR__ 1
```

irq1 - Auto Wake Up from Halt interrupt (AWU)

Definition at line 243 of file STM8AF\_STM8S.h.

#### 5.1.2.3 \_\_CAN\_RX\_VECTOR\_\_

```
#define __CAN_RX_VECTOR__ 8
```

irq8 - CAN receive interrupt (shared with \_\_PORTF\_VECTOR\_\_)

Definition at line 251 of file STM8AF\_STM8S.h.

#### 5.1.2.4 \_\_CAN\_TX\_VECTOR\_\_

```
#define __CAN_TX_VECTOR__ 9
```

irq9 - CAN transmit interrupt

Definition at line 257 of file STM8AF\_STM8S.h.

#### 5.1.2.5 \_\_CLK\_VECTOR\_\_

```
#define __CLK_VECTOR__ 2
```

Definition at line 244 of file STM8AF\_STM8S.h.



#### 5.1.2.6 \_\_FLASH\_VECTOR\_\_

```
#define __FLASH_VECTOR__ 24
```

Definition at line 310 of file STM8AF\_STM8S.h.

#### 5.1.2.7 \_\_I2C\_VECTOR\_\_

```
#define __I2C_VECTOR__ 19
```

irq19 - I2C interrupt

Definition at line 285 of file STM8AF\_STM8S.h.

#### 5.1.2.8 \_\_PORTA\_VECTOR\_\_

```
#define __PORTA_VECTOR__ 3
```

Definition at line 245 of file STM8AF\_STM8S.h.

#### 5.1.2.9 \_\_PORTB\_VECTOR\_\_

```
#define __PORTB_VECTOR__ 4
```

Definition at line 246 of file STM8AF\_STM8S.h.

#### 5.1.2.10 \_\_PORTC\_VECTOR\_\_

```
#define __PORTC_VECTOR__ 5
```

Definition at line 247 of file STM8AF\_STM8S.h.

#### 5.1.2.11 \_\_PORTD\_VECTOR\_\_

```
#define __PORTD_VECTOR__ 6
```

Definition at line 248 of file STM8AF\_STM8S.h.

#### 5.1.2.12 \_\_PORTE\_VECTOR\_\_

```
#define __PORTE_VECTOR__ 7
```

Definition at line 249 of file STM8AF\_STM8S.h.

#### 5.1.2.13 \_\_PORTF\_VECTOR\_\_

```
#define __PORTF_VECTOR__ 8
```

irq8 - External interrupt 5 (GPIOF, shared with \_\_CAN\_RX\_VECTOR\_\_)

Definition at line 254 of file STM8AF\_STM8S.h.

#### 5.1.2.14 \_\_SPI\_VECTOR\_\_

```
#define __SPI_VECTOR__ 10
```

Definition at line 259 of file STM8AF\_STM8S.h.

#### 5.1.2.15 \_\_TIM1\_CAPCOM\_VECTOR\_\_

```
#define __TIM1_CAPCOM_VECTOR__ 12
```

Definition at line 261 of file STM8AF\_STM8S.h.

#### 5.1.2.16 \_\_TIM1\_UPD\_OVF\_VECTOR\_\_

```
#define __TIM1_UPD_OVF_VECTOR__ 11
```

Definition at line 260 of file STM8AF\_STM8S.h.

#### 5.1.2.17 \_\_TIM2\_CAPCOM\_VECTOR\_\_

```
#define __TIM2_CAPCOM_VECTOR__ 14
```

irq14 - TIM2 Capture/Compare interrupt (shared with \_\_TIM5\_CAPCOM\_VECTOR\_\_)

Definition at line 269 of file STM8AF\_STM8S.h.

**5.1.2.18 \_\_TIM2\_UPD\_OVF\_VECTOR\_\_**

```
#define __TIM2_UPD_OVF_VECTOR__ 13
```

irq13 - TIM2 Update/overflow interrupt (shared with \_\_TIM5\_UPD\_OVF\_VECTOR\_\_)

Definition at line 263 of file STM8AF\_STM8S.h.

**5.1.2.19 \_\_TIM3\_CAPCOM\_VECTOR\_\_**

```
#define __TIM3_CAPCOM_VECTOR__ 16
```

irq16 - TIM3 Capture/Compare interrupt

Definition at line 277 of file STM8AF\_STM8S.h.

**5.1.2.20 \_\_TIM3\_UPD\_OVF\_VECTOR\_\_**

```
#define __TIM3_UPD_OVF_VECTOR__ 15
```

irq15 - TIM3 Update/overflow interrupt

Definition at line 274 of file STM8AF\_STM8S.h.

**5.1.2.21 \_\_TIM4\_UPD\_OVF\_VECTOR\_\_**

```
#define __TIM4_UPD_OVF_VECTOR__ 23
```

irq23 - TIM4 Update/Overflow interrupt (shared with \_\_TIM6\_UPD\_OVF\_VECTOR\_\_)

Definition at line 306 of file STM8AF\_STM8S.h.

**5.1.2.22 \_\_TIM5\_UPD\_OVF\_VECTOR\_\_**

```
#define __TIM5_UPD_OVF_VECTOR__ 13
```

irq13 - TIM5 Update/overflow interrupt (shared with \_\_TIM2\_UPD\_OVF\_VECTOR\_\_)

Definition at line 266 of file STM8AF\_STM8S.h.

#### 5.1.2.23 \_\_TLI\_VECTOR\_\_

```
#define __TLI_VECTOR__ 0
```

irq0 - External Top Level interrupt (TLI) for pin PD7

Definition at line 242 of file STM8AF\_STM8S.h.

#### 5.1.2.24 \_\_UART1\_RXF\_VECTOR\_\_

```
#define __UART1_RXF_VECTOR__ 18
```

irq18 - USART/UART1 receive (RX full) interrupt

Definition at line 283 of file STM8AF\_STM8S.h.

#### 5.1.2.25 \_\_UART1\_TXE\_VECTOR\_\_

```
#define __UART1_TXE_VECTOR__ 17
```

irq17 - USART/UART1 send (TX empty) interrupt

Definition at line 280 of file STM8AF\_STM8S.h.

#### 5.1.2.26 \_\_UART2\_RXF\_VECTOR\_\_

```
#define __UART2_RXF_VECTOR__ 21
```

irq21 - UART2 receive (RX full) interrupt (shared with \_\_UART3\_RXF\_VECTOR\_\_ and \_\_UART4\_RXF\_VECTOR\_\_)

Definition at line 294 of file STM8AF\_STM8S.h.

#### 5.1.2.27 \_\_UART2\_TXE\_VECTOR\_\_

```
#define __UART2_TXE_VECTOR__ 20
```

irq20 - UART2 send (TX empty) interrupt (shared with \_\_UART3\_TXE\_VECTOR\_\_ and \_\_UART4\_TXE\_VECTOR\_\_)

Definition at line 287 of file STM8AF\_STM8S.h.

#### 5.1.2.28 \_ADC1

```
#define _ADC1 _SFR(ADC1_t, ADC1_AddressBase)
```

ADC1 struct/bit access.

Definition at line 4794 of file STM8AF\_STM8S.h.

#### 5.1.2.29 \_ADC1\_ADON

```
#define _ADC1_ADON ((uint8_t) (0x01 << 0))
```

ADC1 Conversion on/off [0] (in \_ADC1\_CR1)

Definition at line 4859 of file STM8AF\_STM8S.h.

#### 5.1.2.30 \_ADC1\_ALIGN

```
#define _ADC1_ALIGN ((uint8_t) (0x01 << 3))
```

ADC1 Data alignment [0] (in \_ADC1\_CR2)

Definition at line 4872 of file STM8AF\_STM8S.h.

#### 5.1.2.31 \_ADC1\_AWCRH

```
#define _ADC1_AWCRH _SFR(uint8_t, ADC1_AddressBase+0x2E)
```

ADC1 watchdog control register.

Definition at line 4830 of file STM8AF\_STM8S.h.

#### 5.1.2.32 \_ADC1\_AWCRH\_RESET\_VALUE

```
#define _ADC1_AWCRH_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 watchdog control register reset value.

Definition at line 4844 of file STM8AF\_STM8S.h.

#### 5.1.2.33 `_ADC1_AWCRL`

```
#define _ADC1_AWCRL _SFR(uint8_t, ADC1_AddressBase+0x2F)
```

ADC1 watchdog control register.

Definition at line 4831 of file STM8AF\_STM8S.h.

#### 5.1.2.34 `_ADC1_AWCRL_RESET_VALUE`

```
#define _ADC1_AWCRL_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 watchdog control register reset value.

Definition at line 4845 of file STM8AF\_STM8S.h.

#### 5.1.2.35 `_ADC1_AWD`

```
#define _ADC1_AWD ((uint8_t) (0x01 << 6))
```

ADC1 Analog Watchdog flag [0] (in `_ADC1_CSR`)

Definition at line 4855 of file STM8AF\_STM8S.h.

#### 5.1.2.36 `_ADC1_AWDIE`

```
#define _ADC1_AWDIE ((uint8_t) (0x01 << 4))
```

ADC1 Analog watchdog interrupt enable [0] (in `_ADC1_CSR`)

Definition at line 4853 of file STM8AF\_STM8S.h.

#### 5.1.2.37 `_ADC1_AWSRH`

```
#define _ADC1_AWSRH _SFR(uint8_t, ADC1_AddressBase+0x2C)
```

ADC1 watchdog status register.

Definition at line 4828 of file STM8AF\_STM8S.h.

#### 5.1.2.38 \_ADC1\_AWSRL

```
#define _ADC1_AWSRL _SFR(uint8_t, ADC1_AddressBase+0x2D)
```

ADC1 watchdog status register.

Definition at line 4829 of file STM8AF\_STM8S.h.

#### 5.1.2.39 \_ADC1\_CH

```
#define _ADC1_CH ((uint8_t) (0x0F << 0))
```

ADC1 Channel selection bits [3:0] (in \_ADC1\_CSR)

Definition at line 4848 of file STM8AF\_STM8S.h.

#### 5.1.2.40 \_ADC1\_CH0

```
#define _ADC1_CH0 ((uint8_t) (0x01 << 0))
```

ADC1 Channel selection bits [0] (in \_ADC1\_CSR)

Definition at line 4849 of file STM8AF\_STM8S.h.

#### 5.1.2.41 \_ADC1\_CH1

```
#define _ADC1_CH1 ((uint8_t) (0x01 << 1))
```

ADC1 Channel selection bits [1] (in \_ADC1\_CSR)

Definition at line 4850 of file STM8AF\_STM8S.h.

#### 5.1.2.42 \_ADC1\_CH2

```
#define _ADC1_CH2 ((uint8_t) (0x01 << 2))
```

ADC1 Channel selection bits [2] (in \_ADC1\_CSR)

Definition at line 4851 of file STM8AF\_STM8S.h.

#### 5.1.2.43 \_ADC1\_CH3

```
#define _ADC1_CH3 ((uint8_t) (0x01 << 3))
```

ADC1 Channel selection bits [3] (in \_ADC1\_CSR)

Definition at line 4852 of file STM8AF\_STM8S.h.

#### 5.1.2.44 \_ADC1\_CONT

```
#define _ADC1_CONT ((uint8_t) (0x01 << 1))
```

ADC1 Continuous conversion [0] (in \_ADC1\_CR1)

Definition at line 4860 of file STM8AF\_STM8S.h.

#### 5.1.2.45 \_ADC1\_CR1

```
#define _ADC1_CR1 _SFR(uint8_t, ADC1_AddressBase+0x21)
```

ADC1 Configuration Register 1.

Definition at line 4817 of file STM8AF\_STM8S.h.

#### 5.1.2.46 \_ADC1\_CR1\_RESET\_VALUE

```
#define _ADC1_CR1_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 Configuration Register 1 reset value.

Definition at line 4835 of file STM8AF\_STM8S.h.

#### 5.1.2.47 \_ADC1\_CR2

```
#define _ADC1_CR2 _SFR(uint8_t, ADC1_AddressBase+0x22)
```

ADC1 Configuration Register 2.

Definition at line 4818 of file STM8AF\_STM8S.h.



#### 5.1.2.48 \_ADC1\_CR2\_RESET\_VALUE

```
#define _ADC1_CR2_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 Configuration Register 2 reset value.

Definition at line 4836 of file STM8AF\_STM8S.h.

#### 5.1.2.49 \_ADC1\_CR3

```
#define _ADC1_CR3 _SFR(uint8_t, ADC1_AddressBase+0x23)
```

ADC1 Configuration Register 3.

Definition at line 4819 of file STM8AF\_STM8S.h.

#### 5.1.2.50 \_ADC1\_CR3\_RESET\_VALUE

```
#define _ADC1_CR3_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 Configuration Register 3 reset value.

Definition at line 4837 of file STM8AF\_STM8S.h.

#### 5.1.2.51 \_ADC1\_CSR

```
#define _ADC1_CSR _SFR(uint8_t, ADC1_AddressBase+0x20)
```

ADC1 control/status register.

Definition at line 4816 of file STM8AF\_STM8S.h.

#### 5.1.2.52 \_ADC1\_CSR\_RESET\_VALUE

```
#define _ADC1_CSR_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 control/status register reset value.

Definition at line 4834 of file STM8AF\_STM8S.h.

#### 5.1.2.53 `_ADC1_DB0RH`

```
#define _ADC1_DB0RH _SFR(uint8_t, ADC1_AddressBase+0x00)
```

ADC1 10-bit Data Buffer Register 0.

Definition at line 4795 of file STM8AF\_STM8S.h.

#### 5.1.2.54 `_ADC1_DB0RL`

```
#define _ADC1_DB0RL _SFR(uint8_t, ADC1_AddressBase+0x01)
```

ADC1 10-bit Data Buffer Register 0.

Definition at line 4796 of file STM8AF\_STM8S.h.

#### 5.1.2.55 `_ADC1_DB1RH`

```
#define _ADC1_DB1RH _SFR(uint8_t, ADC1_AddressBase+0x02)
```

ADC1 10-bit Data Buffer Register 1.

Definition at line 4797 of file STM8AF\_STM8S.h.

#### 5.1.2.56 `_ADC1_DB1RL`

```
#define _ADC1_DB1RL _SFR(uint8_t, ADC1_AddressBase+0x03)
```

ADC1 10-bit Data Buffer Register 1.

Definition at line 4798 of file STM8AF\_STM8S.h.

#### 5.1.2.57 `_ADC1_DB2RH`

```
#define _ADC1_DB2RH _SFR(uint8_t, ADC1_AddressBase+0x04)
```

ADC1 10-bit Data Buffer Register 2.

Definition at line 4799 of file STM8AF\_STM8S.h.

#### 5.1.2.58 \_ADC1\_DB2RL

```
#define _ADC1_DB2RL _SFR(uint8_t, ADC1_AddressBase+0x05)
```

ADC1 10-bit Data Buffer Register 2.

Definition at line 4800 of file STM8AF\_STM8S.h.

#### 5.1.2.59 \_ADC1\_DB3RH

```
#define _ADC1_DB3RH _SFR(uint8_t, ADC1_AddressBase+0x06)
```

ADC1 10-bit Data Buffer Register 3.

Definition at line 4801 of file STM8AF\_STM8S.h.

#### 5.1.2.60 \_ADC1\_DB3RL

```
#define _ADC1_DB3RL _SFR(uint8_t, ADC1_AddressBase+0x07)
```

ADC1 10-bit Data Buffer Register 3.

Definition at line 4802 of file STM8AF\_STM8S.h.

#### 5.1.2.61 \_ADC1\_DB4RH

```
#define _ADC1_DB4RH _SFR(uint8_t, ADC1_AddressBase+0x08)
```

ADC1 10-bit Data Buffer Register 4.

Definition at line 4803 of file STM8AF\_STM8S.h.

#### 5.1.2.62 \_ADC1\_DB4RL

```
#define _ADC1_DB4RL _SFR(uint8_t, ADC1_AddressBase+0x09)
```

ADC1 10-bit Data Buffer Register 4.

Definition at line 4804 of file STM8AF\_STM8S.h.

#### 5.1.2.63 `_ADC1_DB5RH`

```
#define _ADC1_DB5RH _SFR(uint8_t, ADC1_AddressBase+0x0A)
```

ADC1 10-bit Data Buffer Register 5.

Definition at line 4805 of file STM8AF\_STM8S.h.

#### 5.1.2.64 `_ADC1_DB5RL`

```
#define _ADC1_DB5RL _SFR(uint8_t, ADC1_AddressBase+0x0B)
```

ADC1 10-bit Data Buffer Register 5.

Definition at line 4806 of file STM8AF\_STM8S.h.

#### 5.1.2.65 `_ADC1_DB6RH`

```
#define _ADC1_DB6RH _SFR(uint8_t, ADC1_AddressBase+0x0C)
```

ADC1 10-bit Data Buffer Register 6.

Definition at line 4807 of file STM8AF\_STM8S.h.

#### 5.1.2.66 `_ADC1_DB6RL`

```
#define _ADC1_DB6RL _SFR(uint8_t, ADC1_AddressBase+0x0D)
```

ADC1 10-bit Data Buffer Register 6.

Definition at line 4808 of file STM8AF\_STM8S.h.

#### 5.1.2.67 `_ADC1_DB7RH`

```
#define _ADC1_DB7RH _SFR(uint8_t, ADC1_AddressBase+0x0E)
```

ADC1 10-bit Data Buffer Register 7.

Definition at line 4809 of file STM8AF\_STM8S.h.

#### 5.1.2.68 \_ADC1\_DB7RL

```
#define _ADC1_DB7RL _SFR(uint8_t, ADC1_AddressBase+0x0F)
```

ADC1 10-bit Data Buffer Register 7.

Definition at line 4810 of file STM8AF\_STM8S.h.

#### 5.1.2.69 \_ADC1\_DB8RH

```
#define _ADC1_DB8RH _SFR(uint8_t, ADC1_AddressBase+0x10)
```

ADC1 10-bit Data Buffer Register 8.

Definition at line 4811 of file STM8AF\_STM8S.h.

#### 5.1.2.70 \_ADC1\_DB8RL

```
#define _ADC1_DB8RL _SFR(uint8_t, ADC1_AddressBase+0x11)
```

ADC1 10-bit Data Buffer Register 8.

Definition at line 4812 of file STM8AF\_STM8S.h.

#### 5.1.2.71 \_ADC1\_DB9RH

```
#define _ADC1_DB9RH _SFR(uint8_t, ADC1_AddressBase+0x12)
```

ADC1 10-bit Data Buffer Register 9.

Definition at line 4813 of file STM8AF\_STM8S.h.

#### 5.1.2.72 \_ADC1\_DB9RL

```
#define _ADC1_DB9RL _SFR(uint8_t, ADC1_AddressBase+0x13)
```

ADC1 10-bit Data Buffer Register 9.

Definition at line 4814 of file STM8AF\_STM8S.h.

#### 5.1.2.73 `_ADC1_DBUF`

```
#define _ADC1_DBUF ((uint8_t) (0x01 << 7))
```

ADC1 Data buffer enable [0] (in `_ADC1_CR3`)

Definition at line 4882 of file `STM8AF_STM8S.h`.

#### 5.1.2.74 `_ADC1_DRH`

```
#define _ADC1_DRH _SFR(uint8_t, ADC1_AddressBase+0x24)
```

ADC1 (unbuffered) 10-bit measurement result.

Definition at line 4820 of file `STM8AF_STM8S.h`.

#### 5.1.2.75 `_ADC1_DRL`

```
#define _ADC1_DRL _SFR(uint8_t, ADC1_AddressBase+0x25)
```

ADC1 (unbuffered) 10-bit measurement result.

Definition at line 4821 of file `STM8AF_STM8S.h`.

#### 5.1.2.76 `_ADC1_EOC`

```
#define _ADC1_EOC ((uint8_t) (0x01 << 7))
```

ADC1 End of conversion [0] (in `_ADC1_CSR`)

Definition at line 4856 of file `STM8AF_STM8S.h`.

#### 5.1.2.77 `_ADC1_EOCIE`

```
#define _ADC1_EOCIE ((uint8_t) (0x01 << 5))
```

ADC1 Interrupt enable for EOC [0] (in `_ADC1_CSR`)

Definition at line 4854 of file `STM8AF_STM8S.h`.

#### 5.1.2.78 \_ADC1\_EXTSEL

```
#define _ADC1_EXTSEL ((uint8_t) (0x03 << 4))
```

ADC1 External event selection [1:0] (in \_ADC1\_CR2)

Definition at line 4873 of file STM8AF\_STM8S.h.

#### 5.1.2.79 \_ADC1\_EXTSEL0

```
#define _ADC1_EXTSEL0 ((uint8_t) (0x01 << 4))
```

ADC1 External event selection [0] (in \_ADC1\_CR2)

Definition at line 4874 of file STM8AF\_STM8S.h.

#### 5.1.2.80 \_ADC1\_EXTSEL1

```
#define _ADC1_EXTSEL1 ((uint8_t) (0x01 << 5))
```

ADC1 External event selection [1] (in \_ADC1\_CR2)

Definition at line 4875 of file STM8AF\_STM8S.h.

#### 5.1.2.81 \_ADC1\_EXTTRIG

```
#define _ADC1_EXTTRIG ((uint8_t) (0x01 << 6))
```

ADC1 External trigger enable [0] (in \_ADC1\_CR2)

Definition at line 4876 of file STM8AF\_STM8S.h.

#### 5.1.2.82 \_ADC1\_HTRH

```
#define _ADC1_HTRH _SFR(uint8_t, ADC1_AddressBase+0x28)
```

ADC1 watchdog high threshold register.

Definition at line 4824 of file STM8AF\_STM8S.h.

#### 5.1.2.83 `_ADC1_HTRH_RESET_VALUE`

```
#define _ADC1_HTRH_RESET_VALUE ((uint8_t) 0xFF)
```

ADC1 watchdog high threshold register reset value.

Definition at line 4840 of file STM8AF\_STM8S.h.

#### 5.1.2.84 `_ADC1_HTRL`

```
#define _ADC1_HTRL _SFR(uint8_t, ADC1_AddressBase+0x29)
```

ADC1 watchdog high threshold register.

Definition at line 4825 of file STM8AF\_STM8S.h.

#### 5.1.2.85 `_ADC1_HTRL_RESET_VALUE`

```
#define _ADC1_HTRL_RESET_VALUE ((uint8_t) 0x03)
```

ADC1 watchdog high threshold register reset value.

Definition at line 4841 of file STM8AF\_STM8S.h.

#### 5.1.2.86 `_ADC1_LTRH`

```
#define _ADC1_LTRH _SFR(uint8_t, ADC1_AddressBase+0x2A)
```

ADC1 watchdog low threshold register.

Definition at line 4826 of file STM8AF\_STM8S.h.

#### 5.1.2.87 `_ADC1_LTRH_RESET_VALUE`

```
#define _ADC1_LTRH_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 watchdog low threshold register reset value.

Definition at line 4842 of file STM8AF\_STM8S.h.



#### 5.1.2.88 \_ADC1\_LTRL

```
#define _ADC1_LTRL _SFR(uint8_t, ADC1_AddressBase+0x2B)
```

ADC1 watchdog low threshold register.

Definition at line 4827 of file STM8AF\_STM8S.h.

#### 5.1.2.89 \_ADC1\_LTRL\_RESET\_VALUE

```
#define _ADC1_LTRL_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 watchdog low threshold register reset value.

Definition at line 4843 of file STM8AF\_STM8S.h.

#### 5.1.2.90 \_ADC1\_OVR

```
#define _ADC1_OVR ((uint8_t) (0x01 << 6))
```

ADC1 Overrun flag [0] (in \_ADC1\_CR3)

Definition at line 4881 of file STM8AF\_STM8S.h.

#### 5.1.2.91 \_ADC1\_SCAN

```
#define _ADC1_SCAN ((uint8_t) (0x01 << 1))
```

ADC1 Scan mode enable [0] (in \_ADC1\_CR2)

Definition at line 4870 of file STM8AF\_STM8S.h.

#### 5.1.2.92 \_ADC1\_SPSEL

```
#define _ADC1_SPSEL ((uint8_t) (0x07 << 4))
```

ADC1 clock prescaler selection [2:0] (in \_ADC1\_CR1)

Definition at line 4862 of file STM8AF\_STM8S.h.

#### 5.1.2.93 `_ADC1_SPSEL0`

```
#define _ADC1_SPSEL0 ((uint8_t) (0x01 << 4))
```

ADC1 clock prescaler selection [0] (in `_ADC1_CR1`)

Definition at line 4863 of file `STM8AF_STM8S.h`.

#### 5.1.2.94 `_ADC1_SPSEL1`

```
#define _ADC1_SPSEL1 ((uint8_t) (0x01 << 5))
```

ADC1 clock prescaler selection [1] (in `_ADC1_CR1`)

Definition at line 4864 of file `STM8AF_STM8S.h`.

#### 5.1.2.95 `_ADC1_SPSEL2`

```
#define _ADC1_SPSEL2 ((uint8_t) (0x01 << 6))
```

ADC1 clock prescaler selection [2] (in `_ADC1_CR1`)

Definition at line 4865 of file `STM8AF_STM8S.h`.

#### 5.1.2.96 `_ADC1_TDRH`

```
#define _ADC1_TDRH \_SFR(uint8_t, ADC1\_AddressBase+0x26)
```

ADC1 Schmitt trigger disable register.

Definition at line 4822 of file `STM8AF_STM8S.h`.

#### 5.1.2.97 `_ADC1_TDRH_RESET_VALUE`

```
#define _ADC1_TDRH_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 Schmitt trigger disable register reset value.

Definition at line 4838 of file `STM8AF_STM8S.h`.

#### 5.1.2.98 \_ADC1\_TDRL

```
#define _ADC1_TDRL _SFR(uint8_t, ADC1_AddressBase+0x27)
```

ADC1 Schmitt trigger disable register.

Definition at line 4823 of file STM8AF\_STM8S.h.

#### 5.1.2.99 \_ADC1\_TDRL\_RESET\_VALUE

```
#define _ADC1_TDRL_RESET_VALUE ((uint8_t) 0x00)
```

ADC1 Schmitt trigger disable register reset value.

Definition at line 4839 of file STM8AF\_STM8S.h.

#### 5.1.2.100 \_ADC2

```
#define _ADC2 _SFR(ADC2_t, ADC2_AddressBase)
```

ADC2 struct/bit access.

Definition at line 4957 of file STM8AF\_STM8S.h.

#### 5.1.2.101 \_ADC2\_ADON

```
#define _ADC2_ADON ((uint8_t) (0x01 << 0))
```

ADC2 Conversion on/off [0] (in \_ADC2\_CR1)

Definition at line 4986 of file STM8AF\_STM8S.h.

#### 5.1.2.102 \_ADC2\_ALIGN

```
#define _ADC2_ALIGN ((uint8_t) (0x01 << 3))
```

ADC2 Data alignment [0] (in \_ADC2\_CR2)

Definition at line 4997 of file STM8AF\_STM8S.h.

#### 5.1.2.103 \_ADC2\_CH

```
#define _ADC2_CH ((uint8_t) (0x0F << 0))
```

ADC2 Channel selection bits [3:0] (in \_ADC2\_CSR)

Definition at line 4975 of file STM8AF\_STM8S.h.

#### 5.1.2.104 \_ADC2\_CH0

```
#define _ADC2_CH0 ((uint8_t) (0x01 << 0))
```

ADC2 Channel selection bits [0] (in \_ADC2\_CSR)

Definition at line 4976 of file STM8AF\_STM8S.h.

#### 5.1.2.105 \_ADC2\_CH1

```
#define _ADC2_CH1 ((uint8_t) (0x01 << 1))
```

ADC2 Channel selection bits [1] (in \_ADC2\_CSR)

Definition at line 4977 of file STM8AF\_STM8S.h.

#### 5.1.2.106 \_ADC2\_CH2

```
#define _ADC2_CH2 ((uint8_t) (0x01 << 2))
```

ADC2 Channel selection bits [2] (in \_ADC2\_CSR)

Definition at line 4978 of file STM8AF\_STM8S.h.

#### 5.1.2.107 \_ADC2\_CH3

```
#define _ADC2_CH3 ((uint8_t) (0x01 << 3))
```

ADC2 Channel selection bits [3] (in \_ADC2\_CSR)

Definition at line 4979 of file STM8AF\_STM8S.h.

#### 5.1.2.108 \_ADC2\_CONT

```
#define _ADC2_CONT ((uint8_t) (0x01 << 1))
```

ADC2 Continuous conversion [0] (in \_ADC2\_CR1)

Definition at line 4987 of file STM8AF\_STM8S.h.

#### 5.1.2.109 \_ADC2\_CR1

```
#define _ADC2_CR1 _SFR(uint8_t, ADC2_AddressBase+0x01)
```

ADC2 Configuration Register 1.

Definition at line 4959 of file STM8AF\_STM8S.h.

#### 5.1.2.110 \_ADC2\_CR1\_RESET\_VALUE

```
#define _ADC2_CR1_RESET_VALUE ((uint8_t) 0x00)
```

ADC2 Configuration Register 1 reset value.

Definition at line 4969 of file STM8AF\_STM8S.h.

#### 5.1.2.111 \_ADC2\_CR2

```
#define _ADC2_CR2 _SFR(uint8_t, ADC2_AddressBase+0x02)
```

ADC2 Configuration Register 2.

Definition at line 4960 of file STM8AF\_STM8S.h.

#### 5.1.2.112 \_ADC2\_CR2\_RESET\_VALUE

```
#define _ADC2_CR2_RESET_VALUE ((uint8_t) 0x00)
```

ADC2 Configuration Register 2 reset value.

Definition at line 4970 of file STM8AF\_STM8S.h.

#### 5.1.2.113 `_ADC2_CSR`

```
#define _ADC2_CSR _SFR(uint8_t, ADC2_AddressBase+0x00)
```

ADC2 control/status register.

Definition at line 4958 of file STM8AF\_STM8S.h.

#### 5.1.2.114 `_ADC2_CSR_RESET_VALUE`

```
#define _ADC2_CSR_RESET_VALUE ((uint8_t) 0x00)
```

ADC2 control/status register reset value.

Definition at line 4968 of file STM8AF\_STM8S.h.

#### 5.1.2.115 `_ADC2_DRH`

```
#define _ADC2_DRH _SFR(uint8_t, ADC2_AddressBase+0x04)
```

ADC2 (unbuffered) 10-bit measurement result.

Definition at line 4962 of file STM8AF\_STM8S.h.

#### 5.1.2.116 `_ADC2_DRL`

```
#define _ADC2_DRL _SFR(uint8_t, ADC2_AddressBase+0x05)
```

ADC2 (unbuffered) 10-bit measurement result.

Definition at line 4963 of file STM8AF\_STM8S.h.

#### 5.1.2.117 `_ADC2_EOC`

```
#define _ADC2_EOC ((uint8_t) (0x01 << 7))
```

ADC2 End of conversion [0] (in `_ADC2_CSR`)

Definition at line 4983 of file STM8AF\_STM8S.h.

**5.1.2.118 \_ADC2\_EOCIE**

```
#define _ADC2_EOCIE ((uint8_t) (0x01 << 5))
```

ADC2 Interrupt enable for EOC [0] (in \_ADC2\_CSR)

Definition at line 4981 of file STM8AF\_STM8S.h.

**5.1.2.119 \_ADC2\_EXTSEL**

```
#define _ADC2_EXTSEL ((uint8_t) (0x03 << 4))
```

ADC2 External event selection [1:0] (in \_ADC2\_CR2)

Definition at line 4998 of file STM8AF\_STM8S.h.

**5.1.2.120 \_ADC2\_EXTSEL0**

```
#define _ADC2_EXTSEL0 ((uint8_t) (0x01 << 4))
```

ADC2 External event selection [0] (in \_ADC2\_CR2)

Definition at line 4999 of file STM8AF\_STM8S.h.

**5.1.2.121 \_ADC2\_EXTSEL1**

```
#define _ADC2_EXTSEL1 ((uint8_t) (0x01 << 5))
```

ADC2 External event selection [1] (in \_ADC2\_CR2)

Definition at line 5000 of file STM8AF\_STM8S.h.

**5.1.2.122 \_ADC2\_EXTTRIG**

```
#define _ADC2_EXTTRIG ((uint8_t) (0x01 << 6))
```

ADC2 External trigger enable [0] (in \_ADC2\_CR2)

Definition at line 5001 of file STM8AF\_STM8S.h.

#### 5.1.2.123 `_ADC2_SPSEL`

```
#define _ADC2_SPSEL ((uint8_t) (0x07 << 4))
```

ADC2 clock prescaler selection [2:0] (in `_ADC2_CR1`)

Definition at line 4989 of file `STM8AF_STM8S.h`.

#### 5.1.2.124 `_ADC2_SPSEL0`

```
#define _ADC2_SPSEL0 ((uint8_t) (0x01 << 4))
```

ADC2 clock prescaler selection [0] (in `_ADC2_CR1`)

Definition at line 4990 of file `STM8AF_STM8S.h`.

#### 5.1.2.125 `_ADC2_SPSEL1`

```
#define _ADC2_SPSEL1 ((uint8_t) (0x01 << 5))
```

ADC2 clock prescaler selection [1] (in `_ADC2_CR1`)

Definition at line 4991 of file `STM8AF_STM8S.h`.

#### 5.1.2.126 `_ADC2_SPSEL2`

```
#define _ADC2_SPSEL2 ((uint8_t) (0x01 << 6))
```

ADC2 clock prescaler selection [2] (in `_ADC2_CR1`)

Definition at line 4992 of file `STM8AF_STM8S.h`.

#### 5.1.2.127 `_ADC2_TDRH`

```
#define _ADC2_TDRH _SFR(uint8_t, ADC2_AddressBase+0x06)
```

ADC2 Schmitt trigger disable register.

Definition at line 4964 of file `STM8AF_STM8S.h`.



**5.1.2.128 \_ADC2\_TDRH\_RESET\_VALUE**

```
#define _ADC2_TDRH_RESET_VALUE ((uint8_t) 0x00)
```

ADC2 Schmitt trigger disable register reset value.

Definition at line 4972 of file STM8AF\_STM8S.h.

**5.1.2.129 \_ADC2\_TDRL**

```
#define _ADC2_TDRL _SFR(uint8_t, ADC2_AddressBase+0x07)
```

ADC2 Schmitt trigger disable register.

Definition at line 4965 of file STM8AF\_STM8S.h.

**5.1.2.130 \_ADC2\_TDRL\_RESET\_VALUE**

```
#define _ADC2_TDRL_RESET_VALUE ((uint8_t) 0x00)
```

ADC2 Schmitt trigger disable register reset value.

Definition at line 4971 of file STM8AF\_STM8S.h.

**5.1.2.131 \_AWU**

```
#define _AWU _SFR(AWU_t, AWU_AddressBase)
```

Auto Wake-Up struct/bit access.

Definition at line 1128 of file STM8AF\_STM8S.h.

**5.1.2.132 \_AWU\_APR**

```
#define _AWU_APR _SFR(uint8_t, AWU_AddressBase+0x01)
```

Auto Wake-Up Asynchronous prescaler register (AWU\_APR)

Definition at line 1130 of file STM8AF\_STM8S.h.

#### 5.1.2.133 `_AWU_APR_RESET_VALUE`

```
#define _AWU_APR_RESET_VALUE ((uint8_t) 0x3F)
```

Auto Wake-Up Asynchronous prescaler register reset value.

Definition at line 1135 of file STM8AF\_STM8S.h.

#### 5.1.2.134 `_AWU_APRE`

```
#define _AWU_APRE ((uint8_t) (0x3F << 0))
```

Auto-wakeup asynchronous prescaler divider [5:0] (in `_AWU_APR`)

Definition at line 1146 of file STM8AF\_STM8S.h.

#### 5.1.2.135 `_AWU_APRE0`

```
#define _AWU_APRE0 ((uint8_t) (0x01 << 0))
```

Auto-wakeup asynchronous prescaler divider [0] (in `_AWU_APR`)

Definition at line 1147 of file STM8AF\_STM8S.h.

#### 5.1.2.136 `_AWU_APRE1`

```
#define _AWU_APRE1 ((uint8_t) (0x01 << 1))
```

Auto-wakeup asynchronous prescaler divider [1] (in `_AWU_APR`)

Definition at line 1148 of file STM8AF\_STM8S.h.

#### 5.1.2.137 `_AWU_APRE2`

```
#define _AWU_APRE2 ((uint8_t) (0x01 << 2))
```

Auto-wakeup asynchronous prescaler divider [2] (in `_AWU_APR`)

Definition at line 1149 of file STM8AF\_STM8S.h.

**5.1.2.138 \_AWU\_APRE3**

```
#define _AWU_APRE3 ((uint8_t) (0x01 << 3))
```

Auto-wakeup asynchronous prescaler divider [3] (in \_AWU\_APR)

Definition at line 1150 of file STM8AF\_STM8S.h.

**5.1.2.139 \_AWU\_APRE4**

```
#define _AWU_APRE4 ((uint8_t) (0x01 << 4))
```

Auto-wakeup asynchronous prescaler divider [4] (in \_AWU\_APR)

Definition at line 1151 of file STM8AF\_STM8S.h.

**5.1.2.140 \_AWU\_APRE5**

```
#define _AWU_APRE5 ((uint8_t) (0x01 << 5))
```

Auto-wakeup asynchronous prescaler divider [5] (in \_AWU\_APR)

Definition at line 1152 of file STM8AF\_STM8S.h.

**5.1.2.141 \_AWU\_AWUEN**

```
#define _AWU_AWUEN ((uint8_t) (0x01 << 4))
```

Auto-wakeup enable [0] (in \_AWU\_CSR)

Definition at line 1141 of file STM8AF\_STM8S.h.

**5.1.2.142 \_AWU\_AWUF**

```
#define _AWU_AWUF ((uint8_t) (0x01 << 5))
```

Auto-wakeup status flag [0] (in \_AWU\_CSR)

Definition at line 1142 of file STM8AF\_STM8S.h.

#### 5.1.2.143 \_AWU\_AWUTB

```
#define _AWU_AWUTB ((uint8_t) (0x0F << 0))
```

Auto-wakeup timebase selection [3:0] (in \_AWU\_APR)

Definition at line 1156 of file STM8AF\_STM8S.h.

#### 5.1.2.144 \_AWU\_AWUTB0

```
#define _AWU_AWUTB0 ((uint8_t) (0x01 << 0))
```

Auto-wakeup timebase selection [0] (in \_AWU\_APR)

Definition at line 1157 of file STM8AF\_STM8S.h.

#### 5.1.2.145 \_AWU\_AWUTB1

```
#define _AWU_AWUTB1 ((uint8_t) (0x01 << 1))
```

Auto-wakeup timebase selection [1] (in \_AWU\_APR)

Definition at line 1158 of file STM8AF\_STM8S.h.

#### 5.1.2.146 \_AWU\_AWUTB2

```
#define _AWU_AWUTB2 ((uint8_t) (0x01 << 2))
```

Auto-wakeup timebase selection [2] (in \_AWU\_APR)

Definition at line 1159 of file STM8AF\_STM8S.h.

#### 5.1.2.147 \_AWU\_AWUTB3

```
#define _AWU_AWUTB3 ((uint8_t) (0x01 << 3))
```

Auto-wakeup timebase selection [3] (in \_AWU\_APR)

Definition at line 1160 of file STM8AF\_STM8S.h.

#### 5.1.2.148 \_AWU\_CSR

```
#define _AWU_CSR _SFR(uint8_t, AWU_AddressBase+0x00)
```

Auto Wake-Up Control/status register (AWU\_CSR)

Definition at line 1129 of file STM8AF\_STM8S.h.

#### 5.1.2.149 \_AWU\_CSR\_RESET\_VALUE

```
#define _AWU_CSR_RESET_VALUE ((uint8_t) 0x00)
```

Auto Wake-Up Control/status register reset value.

Definition at line 1134 of file STM8AF\_STM8S.h.

#### 5.1.2.150 \_AWU\_MSR

```
#define _AWU_MSR ((uint8_t) (0x01 << 0))
```

Auto Wake-Up LSI measurement enable [0] (in \_AWU\_CSR)

Definition at line 1139 of file STM8AF\_STM8S.h.

#### 5.1.2.151 \_AWU\_TBR

```
#define _AWU_TBR _SFR(uint8_t, AWU_AddressBase+0x02)
```

Auto Wake-Up Timebase selection register (AWU\_TBR)

Definition at line 1131 of file STM8AF\_STM8S.h.

#### 5.1.2.152 \_AWU\_TBR\_RESET\_VALUE

```
#define _AWU_TBR_RESET_VALUE ((uint8_t) 0x00)
```

Auto Wake-Up Timebase selection register reset value.

Definition at line 1136 of file STM8AF\_STM8S.h.

#### 5.1.2.153 `_BEEP`

```
#define _BEEP _SFR(BEEP_t, BEEP_AddressBase)
```

Beeper struct/bit access.

Definition at line 1185 of file STM8AF\_STM8S.h.

#### 5.1.2.154 `_BEEP_BEEPDIV`

```
#define _BEEP_BEEPDIV ((uint8_t) (0x1F << 0))
```

Beeper clock prescaler divider [4:0] (in `_BEEP_CSR`)

Definition at line 1192 of file STM8AF\_STM8S.h.

#### 5.1.2.155 `_BEEP_BEEPDIV0`

```
#define _BEEP_BEEPDIV0 ((uint8_t) (0x01 << 0))
```

Beeper clock prescaler divider [0] (in `_BEEP_CSR`)

Definition at line 1193 of file STM8AF\_STM8S.h.

#### 5.1.2.156 `_BEEP_BEEPDIV1`

```
#define _BEEP_BEEPDIV1 ((uint8_t) (0x01 << 1))
```

Beeper clock prescaler divider [1] (in `_BEEP_CSR`)

Definition at line 1194 of file STM8AF\_STM8S.h.

#### 5.1.2.157 `_BEEP_BEEPDIV2`

```
#define _BEEP_BEEPDIV2 ((uint8_t) (0x01 << 2))
```

Beeper clock prescaler divider [2] (in `_BEEP_CSR`)

Definition at line 1195 of file STM8AF\_STM8S.h.

**5.1.2.158 \_BEEP\_BEEPDIV3**

```
#define _BEEP_BEEPDIV3 ((uint8_t) (0x01 << 3))
```

Beeper clock prescaler divider [3] (in \_BEEP\_CSR)

Definition at line 1196 of file STM8AF\_STM8S.h.

**5.1.2.159 \_BEEP\_BEEPDIV4**

```
#define _BEEP_BEEPDIV4 ((uint8_t) (0x01 << 4))
```

Beeper clock prescaler divider [4] (in \_BEEP\_CSR)

Definition at line 1197 of file STM8AF\_STM8S.h.

**5.1.2.160 \_BEEP\_BEEPEN**

```
#define _BEEP_BEEPEN ((uint8_t) (0x01 << 5))
```

Beeper enable [0] (in \_BEEP\_CSR)

Definition at line 1198 of file STM8AF\_STM8S.h.

**5.1.2.161 \_BEEP\_BEEPSEL**

```
#define _BEEP_BEEPSEL ((uint8_t) (0x03 << 6))
```

Beeper frequency selection [1:0] (in \_BEEP\_CSR)

Definition at line 1199 of file STM8AF\_STM8S.h.

**5.1.2.162 \_BEEP\_BEEPSEL0**

```
#define _BEEP_BEEPSEL0 ((uint8_t) (0x01 << 6))
```

Beeper frequency selection [0] (in \_BEEP\_CSR)

Definition at line 1200 of file STM8AF\_STM8S.h.

#### 5.1.2.163 `_BEEP_BEEPSEL1`

```
#define _BEEP_BEEPSEL1 ((uint8_t) (0x01 << 7))
```

Beeper frequency selection [1] (in `_BEEP_CSR`)

Definition at line 1201 of file `STM8AF_STM8S.h`.

#### 5.1.2.164 `_BEEP_CSR`

```
#define _BEEP_CSR _SFR(uint8_t, BEEP_AddressBase+0x00)
```

Beeper control/status register (`BEEP_CSR`)

Definition at line 1186 of file `STM8AF_STM8S.h`.

#### 5.1.2.165 `_BEEP_CSR_RESET_VALUE`

```
#define _BEEP_CSR_RESET_VALUE ((uint8_t) 0x1F)
```

Beeper control/status register reset value.

Definition at line 1189 of file `STM8AF_STM8S.h`.

#### 5.1.2.166 `_BITS`

```
#define _BITS unsigned int
```

data type in bit structs (follow C90 standard)

Definition at line 177 of file `STM8AF_STM8S.h`.

#### 5.1.2.167 `_CAN`

```
#define _CAN _SFR(CAN_t, CAN_AddressBase)
```

CAN struct/bit access.

Definition at line 5946 of file `STM8AF_STM8S.h`.



**5.1.2.168 \_CAN\_ABOM**

```
#define _CAN_ABOM ((uint8_t) (0x01 << 6))
```

CAN Channel Automatic Bus-Off Management [0] (in \_CAN\_MCR)

Definition at line 6095 of file STM8AF\_STM8S.h.

**5.1.2.169 \_CAN\_ABRQ**

```
#define _CAN_ABRQ ((uint8_t) (0x01 << 1))
```

CAN Abort request for mailbox [0] (in \_CAN\_MCSR, page 0,1,5)

Definition at line 6163 of file STM8AF\_STM8S.h.

**5.1.2.170 \_CAN\_ALST**

```
#define _CAN_ALST ((uint8_t) (0x01 << 4))
```

CAN Arbitration lost [0] (in \_CAN\_MCSR, page 0,1,5)

Definition at line 6166 of file STM8AF\_STM8S.h.

**5.1.2.171 \_CAN\_AWUM**

```
#define _CAN_AWUM ((uint8_t) (0x01 << 5))
```

CAN Channel Automatic Wakeup Mode [0] (in \_CAN\_MCR)

Definition at line 6094 of file STM8AF\_STM8S.h.

**5.1.2.172 \_CAN\_BOFF**

```
#define _CAN_BOFF ((uint8_t) (0x01 << 2))
```

CAN Bus off flag [0] (in \_CAN\_ESR, page 6)

Definition at line 6188 of file STM8AF\_STM8S.h.

#### 5.1.2.173 `_CAN_BOFIE`

```
#define _CAN_BOFIE ((uint8_t) (0x01 << 2))
```

CAN Bus-Off interrupt enable [0] (in `_CAN_EIER`, page 6)

Definition at line 6199 of file `STM8AF_STM8S.h`.

#### 5.1.2.174 `_CAN_BRP`

```
#define _CAN_BRP ((uint8_t) (0x3F << 0))
```

CAN Baud rate prescaler [5:0] (in `_CAN_BTR1`, page 6)

Definition at line 6207 of file `STM8AF_STM8S.h`.

#### 5.1.2.175 `_CAN_BRP0`

```
#define _CAN_BRP0 ((uint8_t) (0x01 << 0))
```

CAN Baud rate prescaler [0] (in `_CAN_BTR1`, page 6)

Definition at line 6208 of file `STM8AF_STM8S.h`.

#### 5.1.2.176 `_CAN_BRP1`

```
#define _CAN_BRP1 ((uint8_t) (0x01 << 1))
```

CAN Baud rate prescaler [1] (in `_CAN_BTR1`, page 6)

Definition at line 6209 of file `STM8AF_STM8S.h`.

#### 5.1.2.177 `_CAN_BRP2`

```
#define _CAN_BRP2 ((uint8_t) (0x01 << 2))
```

CAN Baud rate prescaler [2] (in `_CAN_BTR1`, page 6)

Definition at line 6210 of file `STM8AF_STM8S.h`.

**5.1.2.178 \_CAN\_BRP3**

```
#define _CAN_BRP3 ((uint8_t) (0x01 << 3))
```

CAN Baud rate prescaler [3] (in \_CAN\_BTR1, page 6)

Definition at line 6211 of file STM8AF\_STM8S.h.

**5.1.2.179 \_CAN\_BRP4**

```
#define _CAN_BRP4 ((uint8_t) (0x01 << 4))
```

CAN Baud rate prescaler [4] (in \_CAN\_BTR1, page 6)

Definition at line 6212 of file STM8AF\_STM8S.h.

**5.1.2.180 \_CAN\_BRP5**

```
#define _CAN_BRP5 ((uint8_t) (0x01 << 5))
```

CAN Baud rate prescaler [5] (in \_CAN\_BTR1, page 6)

Definition at line 6213 of file STM8AF\_STM8S.h.

**5.1.2.181 \_CAN\_BS1**

```
#define _CAN_BS1 ((uint8_t) (0x0F << 0))
```

CAN Bit segment 1 [3:0] (in \_CAN\_BTR2, page 6)

Definition at line 6219 of file STM8AF\_STM8S.h.

**5.1.2.182 \_CAN\_BS10**

```
#define _CAN_BS10 ((uint8_t) (0x01 << 0))
```

CAN Bit segment 1 [0] (in \_CAN\_BTR2, page 6)

Definition at line 6220 of file STM8AF\_STM8S.h.

**5.1.2.183 \_CAN\_BS11**

```
#define _CAN_BS11 ((uint8_t) (0x01 << 1))
```

CAN Bit segment 1 [1] (in \_CAN\_BTR2, page 6)

Definition at line 6221 of file STM8AF\_STM8S.h.

**5.1.2.184 \_CAN\_BS12**

```
#define _CAN_BS12 ((uint8_t) (0x01 << 2))
```

CAN Bit segment 1 [2] (in \_CAN\_BTR2, page 6)

Definition at line 6222 of file STM8AF\_STM8S.h.

**5.1.2.185 \_CAN\_BS13**

```
#define _CAN_BS13 ((uint8_t) (0x01 << 3))
```

CAN Bit segment 1 [3] (in \_CAN\_BTR2, page 6)

Definition at line 6223 of file STM8AF\_STM8S.h.

**5.1.2.186 \_CAN\_BS2**

```
#define _CAN_BS2 ((uint8_t) (0x07 << 4))
```

CAN Bit segment 2 [2:0] (in \_CAN\_BTR2, page 6)

Definition at line 6224 of file STM8AF\_STM8S.h.

**5.1.2.187 \_CAN\_BS20**

```
#define _CAN_BS20 ((uint8_t) (0x01 << 4))
```

CAN Bit segment 2 [0] (in \_CAN\_BTR2, page 6)

Definition at line 6225 of file STM8AF\_STM8S.h.

**5.1.2.188 \_CAN\_BS21**

```
#define _CAN_BS21 ((uint8_t) (0x01 << 5))
```

CAN Bit segment 2 [1] (in \_CAN\_BTR2, page 6)

Definition at line 6226 of file STM8AF\_STM8S.h.

**5.1.2.189 \_CAN\_BS22**

```
#define _CAN_BS22 ((uint8_t) (0x01 << 6))
```

CAN Bit segment 2 [2] (in \_CAN\_BTR2, page 6)

Definition at line 6227 of file STM8AF\_STM8S.h.

**5.1.2.190 \_CAN\_BTR1**

```
#define _CAN_BTR1 _SFR(uint8_t, CAN_AddressBase+0x08+0x04)
```

CAN bit timing register 1 (page 6)

Definition at line 6037 of file STM8AF\_STM8S.h.

**5.1.2.191 \_CAN\_BTR1\_RESET\_VALUE**

```
#define _CAN_BTR1_RESET_VALUE ((uint8_t) 0x40)
```

CAN bit timing register 1 (page 6) reset value.

Definition at line 6081 of file STM8AF\_STM8S.h.

**5.1.2.192 \_CAN\_BTR2**

```
#define _CAN_BTR2 _SFR(uint8_t, CAN_AddressBase+0x08+0x05)
```

CAN bit timing register 2 (page 6)

Definition at line 6038 of file STM8AF\_STM8S.h.

#### 5.1.2.193 `_CAN_BTR2_RESET_VALUE`

```
#define _CAN_BTR2_RESET_VALUE ((uint8_t) 0x23)
```

CAN bit timing register 2 (page 6) reset value.

Definition at line 6082 of file STM8AF\_STM8S.h.

#### 5.1.2.194 `_CAN_CODE`

```
#define _CAN_CODE ((uint8_t) (0x03 << 0))
```

CAN Mailbox Code [1:0] (in `_CAN_TPR`)

Definition at line 6118 of file STM8AF\_STM8S.h.

#### 5.1.2.195 `_CAN_CODE0`

```
#define _CAN_CODE0 ((uint8_t) (0x01 << 0))
```

CAN Mailbox Code [0] (in `_CAN_TPR`)

Definition at line 6119 of file STM8AF\_STM8S.h.

#### 5.1.2.196 `_CAN_CODE1`

```
#define _CAN_CODE1 ((uint8_t) (0x01 << 1))
```

CAN Mailbox Code [1] (in `_CAN_TPR`)

Definition at line 6120 of file STM8AF\_STM8S.h.

#### 5.1.2.197 `_CAN_DGR`

```
#define _CAN_DGR \_SFR(uint8_t, CAN\_AddressBase+0x06)
```

CAN diagnosis register.

Definition at line 5953 of file STM8AF\_STM8S.h.

**5.1.2.198 \_CAN\_DGR\_RESET\_VALUE**

```
#define _CAN_DGR_RESET_VALUE ((uint8_t) 0x0C)
```

CAN diagnosis register reset value.

Definition at line 6072 of file STM8AF\_STM8S.h.

**5.1.2.199 \_CAN\_DLC**

```
#define _CAN_DLC ((uint8_t) (0x0F << 0))
```

CAN Data length code [3:0] (in \_CAN\_MDLCR, page 0,1,5,7)

Definition at line 6171 of file STM8AF\_STM8S.h.

**5.1.2.200 \_CAN\_DLC0**

```
#define _CAN_DLC0 ((uint8_t) (0x01 << 0))
```

CAN Data length code [0] (in \_CAN\_MDLCR, page 0,1,5,7)

Definition at line 6172 of file STM8AF\_STM8S.h.

**5.1.2.201 \_CAN\_DLC1**

```
#define _CAN_DLC1 ((uint8_t) (0x01 << 1))
```

CAN Data length code [1] (in \_CAN\_MDLCR, page 0,1,5,7)

Definition at line 6173 of file STM8AF\_STM8S.h.

**5.1.2.202 \_CAN\_DLC2**

```
#define _CAN_DLC2 ((uint8_t) (0x01 << 2))
```

CAN Data length code [2] (in \_CAN\_MDLCR, page 0,1,5,7)

Definition at line 6174 of file STM8AF\_STM8S.h.

#### 5.1.2.203 `_CAN_DLC3`

```
#define _CAN_DLC3 ((uint8_t) (0x01 << 3))
```

CAN Data length code [3] (in `_CAN_MDLCR`, page 0,1,5,7)

Definition at line 6175 of file `STM8AF_STM8S.h`.

#### 5.1.2.204 `_CAN_EIER`

```
#define _CAN_EIER _SFR(uint8_t, CAN_AddressBase+0x08+0x01)
```

CAN error interrupt enable register (page 6)

Definition at line 6034 of file `STM8AF_STM8S.h`.

#### 5.1.2.205 `_CAN_EIER_RESET_VALUE`

```
#define _CAN_EIER_RESET_VALUE ((uint8_t) 0x00)
```

CAN error interrupt enable register (page 6) reset value.

Definition at line 6078 of file `STM8AF_STM8S.h`.

#### 5.1.2.206 `_CAN_EPVF`

```
#define _CAN_EPVF ((uint8_t) (0x01 << 1))
```

CAN Error passive flag [0] (in `_CAN_ESR`, page 6)

Definition at line 6187 of file `STM8AF_STM8S.h`.

#### 5.1.2.207 `_CAN_EPVIE`

```
#define _CAN_EPVIE ((uint8_t) (0x01 << 1))
```

CAN Error passive interrupt enable [0] (in `_CAN_EIER`, page 6)

Definition at line 6198 of file `STM8AF_STM8S.h`.



**5.1.2.208 \_CAN\_ERRI**

```
#define _CAN_ERRI ((uint8_t) (0x01 << 2))
```

CAN Error Interrupt [0] (in \_CAN\_MSR)

Definition at line 6101 of file STM8AF\_STM8S.h.

**5.1.2.209 \_CAN\_ERRIE**

```
#define _CAN_ERRIE ((uint8_t) (0x01 << 6))
```

CAN Error interrupt enable [0] (in \_CAN\_EIER, page 6)

Definition at line 6203 of file STM8AF\_STM8S.h.

**5.1.2.210 \_CAN\_ESR**

```
#define _CAN_ESR _SFR(uint8_t, CAN_AddressBase+0x08+0x00)
```

CAN error status register (page 6)

Definition at line 6033 of file STM8AF\_STM8S.h.

**5.1.2.211 \_CAN\_ESR\_RESET\_VALUE**

```
#define _CAN_ESR_RESET_VALUE ((uint8_t) 0x00)
```

CAN error status register (page 6) reset value.

Definition at line 6077 of file STM8AF\_STM8S.h.

**5.1.2.212 \_CAN\_EWGF**

```
#define _CAN_EWGF ((uint8_t) (0x01 << 0))
```

CAN Error warning flag [0] (in \_CAN\_ESR, page 6)

Definition at line 6186 of file STM8AF\_STM8S.h.

#### 5.1.2.213 `_CAN_EWGIE`

```
#define _CAN_EWGIE ((uint8_t) (0x01 << 0))
```

CAN Error warning interrupt enable [0] (in `_CAN_EIER`, page 6)

Definition at line 6197 of file `STM8AF_STM8S.h`.

#### 5.1.2.214 `_CAN_F0R1`

```
#define _CAN_F0R1 _SFR(uint8_t, CAN_AddressBase+0x08+0x00)
```

CAN acceptance filter 0/1 (page 2)

Definition at line 5977 of file `STM8AF_STM8S.h`.

#### 5.1.2.215 `_CAN_F0R2`

```
#define _CAN_F0R2 _SFR(uint8_t, CAN_AddressBase+0x08+0x01)
```

CAN acceptance filter 0/2 (page 2)

Definition at line 5978 of file `STM8AF_STM8S.h`.

#### 5.1.2.216 `_CAN_F0R3`

```
#define _CAN_F0R3 _SFR(uint8_t, CAN_AddressBase+0x08+0x02)
```

CAN acceptance filter 0/3 (page 2)

Definition at line 5979 of file `STM8AF_STM8S.h`.

#### 5.1.2.217 `_CAN_F0R4`

```
#define _CAN_F0R4 _SFR(uint8_t, CAN_AddressBase+0x08+0x03)
```

CAN acceptance filter 0/4 (page 2)

Definition at line 5980 of file `STM8AF_STM8S.h`.

**5.1.2.218 \_CAN\_F0R5**

```
#define _CAN_F0R5 _SFR(uint8_t, CAN_AddressBase+0x08+0x04)
```

CAN acceptance filter 0/5 (page 2)

Definition at line 5981 of file STM8AF\_STM8S.h.

**5.1.2.219 \_CAN\_F0R6**

```
#define _CAN_F0R6 _SFR(uint8_t, CAN_AddressBase+0x08+0x05)
```

CAN acceptance filter 0/6 (page 2)

Definition at line 5982 of file STM8AF\_STM8S.h.

**5.1.2.220 \_CAN\_F0R7**

```
#define _CAN_F0R7 _SFR(uint8_t, CAN_AddressBase+0x08+0x06)
```

CAN acceptance filter 0/7 (page 2)

Definition at line 5983 of file STM8AF\_STM8S.h.

**5.1.2.221 \_CAN\_F0R8**

```
#define _CAN_F0R8 _SFR(uint8_t, CAN_AddressBase+0x08+0x07)
```

CAN acceptance filter 0/8 (page 2)

Definition at line 5984 of file STM8AF\_STM8S.h.

**5.1.2.222 \_CAN\_F1R1**

```
#define _CAN_F1R1 _SFR(uint8_t, CAN_AddressBase+0x08+0x08)
```

CAN acceptance filter 1/1 (page 2)

Definition at line 5985 of file STM8AF\_STM8S.h.

#### 5.1.2.223 `_CAN_F1R2`

```
#define _CAN_F1R2 _SFR(uint8_t, CAN_AddressBase+0x08+0x09)
```

CAN acceptance filter 1/2 (page 2)

Definition at line 5986 of file STM8AF\_STM8S.h.

#### 5.1.2.224 `_CAN_F1R3`

```
#define _CAN_F1R3 _SFR(uint8_t, CAN_AddressBase+0x08+0x0A)
```

CAN acceptance filter 1/3 (page 2)

Definition at line 5987 of file STM8AF\_STM8S.h.

#### 5.1.2.225 `_CAN_F1R4`

```
#define _CAN_F1R4 _SFR(uint8_t, CAN_AddressBase+0x08+0x0B)
```

CAN acceptance filter 1/4 (page 2)

Definition at line 5988 of file STM8AF\_STM8S.h.

#### 5.1.2.226 `_CAN_F1R5`

```
#define _CAN_F1R5 _SFR(uint8_t, CAN_AddressBase+0x08+0x0C)
```

CAN acceptance filter 1/5 (page 2)

Definition at line 5989 of file STM8AF\_STM8S.h.

#### 5.1.2.227 `_CAN_F1R6`

```
#define _CAN_F1R6 _SFR(uint8_t, CAN_AddressBase+0x08+0x0D)
```

CAN acceptance filter 1/6 (page 2)

Definition at line 5990 of file STM8AF\_STM8S.h.

#### 5.1.2.228 \_CAN\_F1R7

```
#define _CAN_F1R7 _SFR(uint8_t, CAN_AddressBase+0x08+0x0E)
```

CAN acceptance filter 1/7 (page 2)

Definition at line 5991 of file STM8AF\_STM8S.h.

#### 5.1.2.229 \_CAN\_F1R8

```
#define _CAN_F1R8 _SFR(uint8_t, CAN_AddressBase+0x08+0x0F)
```

CAN acceptance filter 1/8 (page 2)

Definition at line 5992 of file STM8AF\_STM8S.h.

#### 5.1.2.230 \_CAN\_F2R1

```
#define _CAN_F2R1 _SFR(uint8_t, CAN_AddressBase+0x08+0x00)
```

CAN acceptance filter 2/1 (page 3)

Definition at line 5995 of file STM8AF\_STM8S.h.

#### 5.1.2.231 \_CAN\_F2R2

```
#define _CAN_F2R2 _SFR(uint8_t, CAN_AddressBase+0x08+0x01)
```

CAN acceptance filter 2/2 (page 3)

Definition at line 5996 of file STM8AF\_STM8S.h.

#### 5.1.2.232 \_CAN\_F2R3

```
#define _CAN_F2R3 _SFR(uint8_t, CAN_AddressBase+0x08+0x02)
```

CAN acceptance filter 2/3 (page 3)

Definition at line 5997 of file STM8AF\_STM8S.h.

#### 5.1.2.233 `_CAN_F2R4`

```
#define _CAN_F2R4 _SFR(uint8_t, CAN_AddressBase+0x08+0x03)
```

CAN acceptance filter 2/4 (page 3)

Definition at line 5998 of file STM8AF\_STM8S.h.

#### 5.1.2.234 `_CAN_F2R5`

```
#define _CAN_F2R5 _SFR(uint8_t, CAN_AddressBase+0x08+0x04)
```

CAN acceptance filter 2/5 (page 3)

Definition at line 5999 of file STM8AF\_STM8S.h.

#### 5.1.2.235 `_CAN_F2R6`

```
#define _CAN_F2R6 _SFR(uint8_t, CAN_AddressBase+0x08+0x05)
```

CAN acceptance filter 2/6 (page 3)

Definition at line 6000 of file STM8AF\_STM8S.h.

#### 5.1.2.236 `_CAN_F2R7`

```
#define _CAN_F2R7 _SFR(uint8_t, CAN_AddressBase+0x08+0x06)
```

CAN acceptance filter 2/7 (page 3)

Definition at line 6001 of file STM8AF\_STM8S.h.

#### 5.1.2.237 `_CAN_F2R8`

```
#define _CAN_F2R8 _SFR(uint8_t, CAN_AddressBase+0x08+0x07)
```

CAN acceptance filter 2/8 (page 3)

Definition at line 6002 of file STM8AF\_STM8S.h.

#### 5.1.2.238 \_CAN\_F3R1

```
#define _CAN_F3R1 _SFR(uint8_t, CAN_AddressBase+0x08+0x08)
```

CAN acceptance filter 3/1 (page 3)

Definition at line 6003 of file STM8AF\_STM8S.h.

#### 5.1.2.239 \_CAN\_F3R2

```
#define _CAN_F3R2 _SFR(uint8_t, CAN_AddressBase+0x08+0x09)
```

CAN acceptance filter 3/2 (page 3)

Definition at line 6004 of file STM8AF\_STM8S.h.

#### 5.1.2.240 \_CAN\_F3R3

```
#define _CAN_F3R3 _SFR(uint8_t, CAN_AddressBase+0x08+0x0A)
```

CAN acceptance filter 3/3 (page 3)

Definition at line 6005 of file STM8AF\_STM8S.h.

#### 5.1.2.241 \_CAN\_F3R4

```
#define _CAN_F3R4 _SFR(uint8_t, CAN_AddressBase+0x08+0x0B)
```

CAN acceptance filter 3/4 (page 3)

Definition at line 6006 of file STM8AF\_STM8S.h.

#### 5.1.2.242 \_CAN\_F3R5

```
#define _CAN_F3R5 _SFR(uint8_t, CAN_AddressBase+0x08+0x0C)
```

CAN acceptance filter 3/5 (page 3)

Definition at line 6007 of file STM8AF\_STM8S.h.

#### 5.1.2.243 `_CAN_F3R6`

```
#define _CAN_F3R6 _SFR(uint8_t, CAN_AddressBase+0x08+0x0D)
```

CAN acceptance filter 3/6 (page 3)

Definition at line 6008 of file STM8AF\_STM8S.h.

#### 5.1.2.244 `_CAN_F3R7`

```
#define _CAN_F3R7 _SFR(uint8_t, CAN_AddressBase+0x08+0x0E)
```

CAN acceptance filter 3/7 (page 3)

Definition at line 6009 of file STM8AF\_STM8S.h.

#### 5.1.2.245 `_CAN_F3R8`

```
#define _CAN_F3R8 _SFR(uint8_t, CAN_AddressBase+0x08+0x0F)
```

CAN acceptance filter 3/8 (page 3)

Definition at line 6010 of file STM8AF\_STM8S.h.

#### 5.1.2.246 `_CAN_F4R1`

```
#define _CAN_F4R1 _SFR(uint8_t, CAN_AddressBase+0x08+0x00)
```

CAN acceptance filter 4/1 (page 4)

Definition at line 6013 of file STM8AF\_STM8S.h.

#### 5.1.2.247 `_CAN_F4R2`

```
#define _CAN_F4R2 _SFR(uint8_t, CAN_AddressBase+0x08+0x01)
```

CAN acceptance filter 4/2 (page 4)

Definition at line 6014 of file STM8AF\_STM8S.h.



**5.1.2.248 \_CAN\_F4R3**

```
#define _CAN_F4R3 _SFR(uint8_t, CAN_AddressBase+0x08+0x02)
```

CAN acceptance filter 4/3 (page 4)

Definition at line 6015 of file STM8AF\_STM8S.h.

**5.1.2.249 \_CAN\_F4R4**

```
#define _CAN_F4R4 _SFR(uint8_t, CAN_AddressBase+0x08+0x03)
```

CAN acceptance filter 4/4 (page 4)

Definition at line 6016 of file STM8AF\_STM8S.h.

**5.1.2.250 \_CAN\_F4R5**

```
#define _CAN_F4R5 _SFR(uint8_t, CAN_AddressBase+0x08+0x04)
```

CAN acceptance filter 4/5 (page 4)

Definition at line 6017 of file STM8AF\_STM8S.h.

**5.1.2.251 \_CAN\_F4R6**

```
#define _CAN_F4R6 _SFR(uint8_t, CAN_AddressBase+0x08+0x05)
```

CAN acceptance filter 4/6 (page 4)

Definition at line 6018 of file STM8AF\_STM8S.h.

**5.1.2.252 \_CAN\_F4R7**

```
#define _CAN_F4R7 _SFR(uint8_t, CAN_AddressBase+0x08+0x06)
```

CAN acceptance filter 4/7 (page 4)

Definition at line 6019 of file STM8AF\_STM8S.h.

#### 5.1.2.253 `_CAN_F4R8`

```
#define _CAN_F4R8 _SFR(uint8_t, CAN_AddressBase+0x08+0x07)
```

CAN acceptance filter 4/8 (page 4)

Definition at line 6020 of file STM8AF\_STM8S.h.

#### 5.1.2.254 `_CAN_F5R1`

```
#define _CAN_F5R1 _SFR(uint8_t, CAN_AddressBase+0x08+0x08)
```

CAN acceptance filter 5/1 (page 4)

Definition at line 6021 of file STM8AF\_STM8S.h.

#### 5.1.2.255 `_CAN_F5R2`

```
#define _CAN_F5R2 _SFR(uint8_t, CAN_AddressBase+0x08+0x09)
```

CAN acceptance filter 5/2 (page 4)

Definition at line 6022 of file STM8AF\_STM8S.h.

#### 5.1.2.256 `_CAN_F5R3`

```
#define _CAN_F5R3 _SFR(uint8_t, CAN_AddressBase+0x08+0x0A)
```

CAN acceptance filter 5/3 (page 4)

Definition at line 6023 of file STM8AF\_STM8S.h.

#### 5.1.2.257 `_CAN_F5R4`

```
#define _CAN_F5R4 _SFR(uint8_t, CAN_AddressBase+0x08+0x0B)
```

CAN acceptance filter 5/4 (page 4)

Definition at line 6024 of file STM8AF\_STM8S.h.

**5.1.2.258 \_CAN\_F5R5**

```
#define _CAN_F5R5 _SFR(uint8_t, CAN_AddressBase+0x08+0x0C)
```

CAN acceptance filter 5/5 (page 4)

Definition at line 6025 of file STM8AF\_STM8S.h.

**5.1.2.259 \_CAN\_F5R6**

```
#define _CAN_F5R6 _SFR(uint8_t, CAN_AddressBase+0x08+0x0D)
```

CAN acceptance filter 5/6 (page 4)

Definition at line 6026 of file STM8AF\_STM8S.h.

**5.1.2.260 \_CAN\_F5R7**

```
#define _CAN_F5R7 _SFR(uint8_t, CAN_AddressBase+0x08+0x0E)
```

CAN acceptance filter 5/7 (page 4)

Definition at line 6027 of file STM8AF\_STM8S.h.

**5.1.2.261 \_CAN\_F5R8**

```
#define _CAN_F5R8 _SFR(uint8_t, CAN_AddressBase+0x08+0x0F)
```

CAN acceptance filter 5/8 (page 4)

Definition at line 6028 of file STM8AF\_STM8S.h.

**5.1.2.262 \_CAN\_FACT0**

```
#define _CAN_FACT0 ((uint8_t) (0x01 << 0))
```

CAN Filter 0 active [0] (in \_CAN\_FCR1, page 6)

Definition at line 6248 of file STM8AF\_STM8S.h.

#### 5.1.2.263 `_CAN_FACT1`

```
#define _CAN_FACT1 ((uint8_t) (0x01 << 4))
```

CAN Filter 1 active [0] (in `_CAN_FCR1`, page 6)

Definition at line 6253 of file `STM8AF_STM8S.h`.

#### 5.1.2.264 `_CAN_FACT2`

```
#define _CAN_FACT2 ((uint8_t) (0x01 << 0))
```

CAN Filter 2 active [0] (in `_CAN_FCR2`, page 6)

Definition at line 6260 of file `STM8AF_STM8S.h`.

#### 5.1.2.265 `_CAN_FACT3`

```
#define _CAN_FACT3 ((uint8_t) (0x01 << 4))
```

CAN Filter 3 active [0] (in `_CAN_FCR2`, page 6)

Definition at line 6265 of file `STM8AF_STM8S.h`.

#### 5.1.2.266 `_CAN_FACT4`

```
#define _CAN_FACT4 ((uint8_t) (0x01 << 0))
```

CAN Filter 4 active [0] (in `_CAN_FCR3`, page 6)

Definition at line 6272 of file `STM8AF_STM8S.h`.

#### 5.1.2.267 `_CAN_FACT5`

```
#define _CAN_FACT5 ((uint8_t) (0x01 << 4))
```

CAN Filter 5 active [0] (in `_CAN_FCR2`, page 6)

Definition at line 6277 of file `STM8AF_STM8S.h`.

**5.1.2.268 \_CAN\_FCR1**

```
#define _CAN_FCR1 _SFR(uint8_t, CAN_AddressBase+0x08+0x0A)
```

CAN filter configuration register 1 (page 6)

Definition at line 6042 of file STM8AF\_STM8S.h.

**5.1.2.269 \_CAN\_FCR2**

```
#define _CAN_FCR2 _SFR(uint8_t, CAN_AddressBase+0x08+0x0B)
```

CAN filter configuration register 2 (page 6)

Definition at line 6043 of file STM8AF\_STM8S.h.

**5.1.2.270 \_CAN\_FCR3**

```
#define _CAN_FCR3 _SFR(uint8_t, CAN_AddressBase+0x08+0x0C)
```

CAN filter configuration register 3 (page 6)

Definition at line 6044 of file STM8AF\_STM8S.h.

**5.1.2.271 \_CAN\_FCR\_RESET\_VALUE**

```
#define _CAN_FCR_RESET_VALUE ((uint8_t) 0x00)
```

CAN filter configuration register reset value.

Definition at line 6085 of file STM8AF\_STM8S.h.

**5.1.2.272 \_CAN\_FFIE**

```
#define _CAN_FFIE ((uint8_t) (0x01 << 2))
```

CAN FIFO Full Interrupt Enable [0] (in \_CAN\_IER)

Definition at line 6141 of file STM8AF\_STM8S.h.

**5.1.2.273 \_CAN\_FMH0**

```
#define _CAN_FMH0 ((uint8_t) (0x01 << 1))
```

CAN Filter 0 mode high [0] (in \_CAN\_FMR1, page 6)

Definition at line 6232 of file STM8AF\_STM8S.h.

**5.1.2.274 \_CAN\_FMH1**

```
#define _CAN_FMH1 ((uint8_t) (0x01 << 3))
```

CAN Filter 1 mode high [0] (in \_CAN\_FMR1, page 6)

Definition at line 6234 of file STM8AF\_STM8S.h.

**5.1.2.275 \_CAN\_FMH2**

```
#define _CAN_FMH2 ((uint8_t) (0x01 << 5))
```

CAN Filter 2 mode high [0] (in \_CAN\_FMR1, page 6)

Definition at line 6236 of file STM8AF\_STM8S.h.

**5.1.2.276 \_CAN\_FMH3**

```
#define _CAN_FMH3 ((uint8_t) (0x01 << 7))
```

CAN Filter 3 mode high [0] (in \_CAN\_FMR1, page 6)

Definition at line 6238 of file STM8AF\_STM8S.h.

**5.1.2.277 \_CAN\_FMH4**

```
#define _CAN_FMH4 ((uint8_t) (0x01 << 1))
```

CAN Filter 4 mode high [0] (in \_CAN\_FMR2, page 6)

Definition at line 6242 of file STM8AF\_STM8S.h.

**5.1.2.278 \_CAN\_FMH5**

```
#define _CAN_FMH5 ((uint8_t) (0x01 << 3))
```

CAN Filter 5 mode high [0] (in \_CAN\_FMR2, page 6)

Definition at line 6244 of file STM8AF\_STM8S.h.

**5.1.2.279 \_CAN\_FML0**

```
#define _CAN_FML0 ((uint8_t) (0x01 << 0))
```

CAN Filter 0 mode low [0] (in \_CAN\_FMR1, page 6)

Definition at line 6231 of file STM8AF\_STM8S.h.

**5.1.2.280 \_CAN\_FML1**

```
#define _CAN_FML1 ((uint8_t) (0x01 << 2))
```

CAN Filter 1 mode low [0] (in \_CAN\_FMR1, page 6)

Definition at line 6233 of file STM8AF\_STM8S.h.

**5.1.2.281 \_CAN\_FML2**

```
#define _CAN_FML2 ((uint8_t) (0x01 << 4))
```

CAN Filter 2 mode low [0] (in \_CAN\_FMR1, page 6)

Definition at line 6235 of file STM8AF\_STM8S.h.

**5.1.2.282 \_CAN\_FML3**

```
#define _CAN_FML3 ((uint8_t) (0x01 << 6))
```

CAN Filter 3 mode low [0] (in \_CAN\_FMR1, page 6)

Definition at line 6237 of file STM8AF\_STM8S.h.

#### 5.1.2.283 `_CAN_FML4`

```
#define _CAN_FML4 ((uint8_t) (0x01 << 0))
```

CAN Filter 4 mode low [0] (in `_CAN_FMR2`, page 6)

Definition at line 6241 of file `STM8AF_STM8S.h`.

#### 5.1.2.284 `_CAN_FML5`

```
#define _CAN_FML5 ((uint8_t) (0x01 << 2))
```

CAN Filter 5 mode low [0] (in `_CAN_FMR2`, page 6)

Definition at line 6243 of file `STM8AF_STM8S.h`.

#### 5.1.2.285 `_CAN_FMP`

```
#define _CAN_FMP ((uint8_t) (0x03 << 0))
```

CAN FIFO Message Pending [1:0] (in `_CAN_RFR`)

Definition at line 6129 of file `STM8AF_STM8S.h`.

#### 5.1.2.286 `_CAN_FMP0`

```
#define _CAN_FMP0 ((uint8_t) (0x01 << 0))
```

CAN FIFO Message Pending [0] (in `_CAN_RFR`)

Definition at line 6130 of file `STM8AF_STM8S.h`.

#### 5.1.2.287 `_CAN_FMP1`

```
#define _CAN_FMP1 ((uint8_t) (0x01 << 1))
```

CAN FIFO Message Pending [1] (in `_CAN_RFR`)

Definition at line 6131 of file `STM8AF_STM8S.h`.



**5.1.2.288 \_CAN\_FMPIE**

```
#define _CAN_FMPIE ((uint8_t) (0x01 << 1))
```

CAN FIFO Message Pending Interrupt Enable [0] (in \_CAN\_IER)

Definition at line 6140 of file STM8AF\_STM8S.h.

**5.1.2.289 \_CAN\_FMR1**

```
#define _CAN_FMR1 _SFR(uint8_t, CAN_AddressBase+0x08+0x08)
```

CAN filter mode register 1 (page 6)

Definition at line 6040 of file STM8AF\_STM8S.h.

**5.1.2.290 \_CAN\_FMR1\_RESET\_VALUE**

```
#define _CAN_FMR1_RESET_VALUE ((uint8_t) 0x00)
```

CAN filter mode register 1 (page 6) reset value.

Definition at line 6083 of file STM8AF\_STM8S.h.

**5.1.2.291 \_CAN\_FMR2**

```
#define _CAN_FMR2 _SFR(uint8_t, CAN_AddressBase+0x08+0x09)
```

CAN filter mode register 2 (page 6)

Definition at line 6041 of file STM8AF\_STM8S.h.

**5.1.2.292 \_CAN\_FMR2\_RESET\_VALUE**

```
#define _CAN_FMR2_RESET_VALUE ((uint8_t) 0x00)
```

CAN filter mode register 2 (page 6) reset value.

Definition at line 6084 of file STM8AF\_STM8S.h.

#### 5.1.2.293 `_CAN_FOVIE`

```
#define _CAN_FOVIE ((uint8_t) (0x01 << 3))
```

CAN FIFO Overrun Interrupt Enable [0] (in `_CAN_IER`)

Definition at line 6142 of file `STM8AF_STM8S.h`.

#### 5.1.2.294 `_CAN_FOVR`

```
#define _CAN_FOVR ((uint8_t) (0x01 << 4))
```

CAN FIFO Overrun [0] (in `_CAN_RFR`)

Definition at line 6134 of file `STM8AF_STM8S.h`.

#### 5.1.2.295 `_CAN_FSC0`

```
#define _CAN_FSC0 ((uint8_t) (0x03 << 1))
```

CAN Filter 0 scale configuration [1:0] (in `_CAN_FCR1`, page 6)

Definition at line 6249 of file `STM8AF_STM8S.h`.

#### 5.1.2.296 `_CAN_FSC00`

```
#define _CAN_FSC00 ((uint8_t) (0x01 << 1))
```

CAN Filter 0 scale configuration [0] (in `_CAN_FCR1`, page 6)

Definition at line 6250 of file `STM8AF_STM8S.h`.

#### 5.1.2.297 `_CAN_FSC01`

```
#define _CAN_FSC01 ((uint8_t) (0x01 << 2))
```

CAN Filter 0 scale configuration [1] (in `_CAN_FCR1`, page 6)

Definition at line 6251 of file `STM8AF_STM8S.h`.

**5.1.2.298 \_CAN\_FSC1**

```
#define _CAN_FSC1 ((uint8_t) (0x03 << 5))
```

CAN Filter 1 scale configuration [1:0] (in \_CAN\_FCR1, page 6)

Definition at line 6254 of file STM8AF\_STM8S.h.

**5.1.2.299 \_CAN\_FSC10**

```
#define _CAN_FSC10 ((uint8_t) (0x01 << 5))
```

CAN Filter 1 scale configuration [0] (in \_CAN\_FCR1, page 6)

Definition at line 6255 of file STM8AF\_STM8S.h.

**5.1.2.300 \_CAN\_FSC11**

```
#define _CAN_FSC11 ((uint8_t) (0x01 << 6))
```

CAN Filter 1 scale configuration [1] (in \_CAN\_FCR1, page 6)

Definition at line 6256 of file STM8AF\_STM8S.h.

**5.1.2.301 \_CAN\_FSC2**

```
#define _CAN_FSC2 ((uint8_t) (0x03 << 1))
```

CAN Filter 2 scale configuration [1:0] (in \_CAN\_FCR2, page 6)

Definition at line 6261 of file STM8AF\_STM8S.h.

**5.1.2.302 \_CAN\_FSC20**

```
#define _CAN_FSC20 ((uint8_t) (0x01 << 1))
```

CAN Filter 2 scale configuration [0] (in \_CAN\_FCR2, page 6)

Definition at line 6262 of file STM8AF\_STM8S.h.

#### 5.1.2.303 `_CAN_FSC21`

```
#define _CAN_FSC21 ((uint8_t) (0x01 << 2))
```

CAN Filter 2 scale configuration [1] (in `_CAN_FCR2`, page 6)

Definition at line 6263 of file `STM8AF_STM8S.h`.

#### 5.1.2.304 `_CAN_FSC3`

```
#define _CAN_FSC3 ((uint8_t) (0x03 << 5))
```

CAN Filter 3 scale configuration [1:0] (in `_CAN_FCR2`, page 6)

Definition at line 6266 of file `STM8AF_STM8S.h`.

#### 5.1.2.305 `_CAN_FSC30`

```
#define _CAN_FSC30 ((uint8_t) (0x01 << 5))
```

CAN Filter 3 scale configuration [0] (in `_CAN_FCR2`, page 6)

Definition at line 6267 of file `STM8AF_STM8S.h`.

#### 5.1.2.306 `_CAN_FSC31`

```
#define _CAN_FSC31 ((uint8_t) (0x01 << 6))
```

CAN Filter 3 scale configuration [1] (in `_CAN_FCR2`, page 6)

Definition at line 6268 of file `STM8AF_STM8S.h`.

#### 5.1.2.307 `_CAN_FSC4`

```
#define _CAN_FSC4 ((uint8_t) (0x03 << 1))
```

CAN Filter 4 scale configuration [1:0] (in `_CAN_FCR3`, page 6)

Definition at line 6273 of file `STM8AF_STM8S.h`.

**5.1.2.308 \_CAN\_FSC40**

```
#define _CAN_FSC40 ((uint8_t) (0x01 << 1))
```

CAN Filter 4 scale configuration [0] (in \_CAN\_FCR3, page 6)

Definition at line 6274 of file STM8AF\_STM8S.h.

**5.1.2.309 \_CAN\_FSC41**

```
#define _CAN_FSC41 ((uint8_t) (0x01 << 2))
```

CAN Filter 4 scale configuration [1] (in \_CAN\_FCR3, page 6)

Definition at line 6275 of file STM8AF\_STM8S.h.

**5.1.2.310 \_CAN\_FSC5**

```
#define _CAN_FSC5 ((uint8_t) (0x03 << 5))
```

CAN Filter 5 scale configuration [1:0] (in \_CAN\_FCR3, page 6)

Definition at line 6278 of file STM8AF\_STM8S.h.

**5.1.2.311 \_CAN\_FSC50**

```
#define _CAN_FSC50 ((uint8_t) (0x01 << 5))
```

CAN Filter 5 scale configuration [0] (in \_CAN\_FCR3, page 6)

Definition at line 6279 of file STM8AF\_STM8S.h.

**5.1.2.312 \_CAN\_FSC51**

```
#define _CAN_FSC51 ((uint8_t) (0x01 << 6))
```

CAN Filter 5 scale configuration [1] (in \_CAN\_FCR3, page 6)

Definition at line 6280 of file STM8AF\_STM8S.h.

#### 5.1.2.313 `_CAN_FULL`

```
#define _CAN_FULL ((uint8_t) (0x01 << 3))
```

CAN FIFO Full [0] (in `_CAN_RFR`)

Definition at line 6133 of file `STM8AF_STM8S.h`.

#### 5.1.2.314 `_CAN_IDE`

```
#define _CAN_IDE ((uint8_t) (0x01 << 6))
```

CAN Extended identifier [0] (in `_CAN_MIDR1`, page 0,1,5)

Definition at line 6182 of file `STM8AF_STM8S.h`.

#### 5.1.2.315 `_CAN_IER`

```
#define _CAN_IER _SFR(uint8_t, CAN_AddressBase+0x05)
```

CAN interrupt enable register.

Definition at line 5952 of file `STM8AF_STM8S.h`.

#### 5.1.2.316 `_CAN_IER_RESET_VALUE`

```
#define _CAN_IER_RESET_VALUE ((uint8_t) 0x00)
```

CAN interrupt enable register reset value.

Definition at line 6071 of file `STM8AF_STM8S.h`.

#### 5.1.2.317 `_CAN_INAK`

```
#define _CAN_INAK ((uint8_t) (0x01 << 0))
```

CAN Initialization Acknowledge [0] (in `_CAN_MSR`)

Definition at line 6099 of file `STM8AF_STM8S.h`.

**5.1.2.318 \_CAN\_INRQ**

```
#define _CAN_INRQ ((uint8_t) (0x01 << 0))
```

CAN Channel Initialization Request [0] (in \_CAN\_MCR)

Definition at line 6089 of file STM8AF\_STM8S.h.

**5.1.2.319 \_CAN\_LBKM**

```
#define _CAN_LBKM ((uint8_t) (0x01 << 0))
```

CAN Loop back mode [0] (in \_CAN\_DGR)

Definition at line 6147 of file STM8AF\_STM8S.h.

**5.1.2.320 \_CAN\_LEC**

```
#define _CAN_LEC ((uint8_t) (0x07 << 4))
```

CAN Last error code [2:0] (in \_CAN\_ESR, page 6)

Definition at line 6190 of file STM8AF\_STM8S.h.

**5.1.2.321 \_CAN\_LEC0**

```
#define _CAN_LEC0 ((uint8_t) (0x01 << 4))
```

CAN Last error code [0] (in \_CAN\_ESR, page 6)

Definition at line 6191 of file STM8AF\_STM8S.h.

**5.1.2.322 \_CAN\_LEC1**

```
#define _CAN_LEC1 ((uint8_t) (0x01 << 5))
```

CAN Last error code [1] (in \_CAN\_ESR, page 6)

Definition at line 6192 of file STM8AF\_STM8S.h.

#### 5.1.2.323 `_CAN_LEC2`

```
#define _CAN_LEC2 ((uint8_t) (0x01 << 6))
```

CAN Last error code [3] (in `_CAN_ESR`, page 6)

Definition at line 6193 of file `STM8AF_STM8S.h`.

#### 5.1.2.324 `_CAN_LECIE`

```
#define _CAN_LECIE ((uint8_t) (0x01 << 4))
```

CAN Last error code interrupt enable [0] (in `_CAN_EIER`, page 6)

Definition at line 6201 of file `STM8AF_STM8S.h`.

#### 5.1.2.325 `_CAN_LOW0`

```
#define _CAN_LOW0 ((uint8_t) (0x01 << 5))
```

CAN Lowest Priority Flag for Mailbox 0 [0] (in `_CAN_TPR`)

Definition at line 6124 of file `STM8AF_STM8S.h`.

#### 5.1.2.326 `_CAN_LOW1`

```
#define _CAN_LOW1 ((uint8_t) (0x01 << 6))
```

CAN Lowest Priority Flag for Mailbox 1 [0] (in `_CAN_TPR`)

Definition at line 6125 of file `STM8AF_STM8S.h`.

#### 5.1.2.327 `_CAN_LOW2`

```
#define _CAN_LOW2 ((uint8_t) (0x01 << 7))
```

CAN Lowest Priority Flag for Mailbox 2 [0] (in `_CAN_TPR`)

Definition at line 6126 of file `STM8AF_STM8S.h`.



**5.1.2.328 \_CAN\_MCR**

```
#define _CAN_MCR _SFR(uint8_t, CAN_AddressBase+0x00)
```

CAN master control register.

Definition at line 5947 of file STM8AF\_STM8S.h.

**5.1.2.329 \_CAN\_MCR\_RESET\_VALUE**

```
#define _CAN_MCR_RESET_VALUE ((uint8_t) 0x02)
```

CAN master control register reset value.

Definition at line 6066 of file STM8AF\_STM8S.h.

**5.1.2.330 \_CAN\_MCSR**

```
#define _CAN_MCSR _SFR(uint8_t, CAN_AddressBase+0x08+0x00)
```

CAN message control/status register (page 0,1,5)

Definition at line 5957 of file STM8AF\_STM8S.h.

**5.1.2.331 \_CAN\_MCSR\_RESET\_VALUE**

```
#define _CAN_MCSR_RESET_VALUE ((uint8_t) 0x00)
```

CAN message control/status register (page 0,1,5) reset value.

Definition at line 6075 of file STM8AF\_STM8S.h.

**5.1.2.332 \_CAN\_MDAR1**

```
#define _CAN_MDAR1 _SFR(uint8_t, CAN_AddressBase+0x08+0x06)
```

CAN mailbox data register 1 (page 0,1,5,7) \*/.

Definition at line 5963 of file STM8AF\_STM8S.h.

#### 5.1.2.333 `_CAN_MDAR2`

```
#define _CAN_MDAR2 __SFR(uint8_t, CAN_AddressBase+0x08+0x07)
```

CAN mailbox data register 2 (page 0,1,5,7) \*/.

Definition at line 5964 of file STM8AF\_STM8S.h.

#### 5.1.2.334 `_CAN_MDAR3`

```
#define _CAN_MDAR3 __SFR(uint8_t, CAN_AddressBase+0x08+0x08)
```

CAN mailbox data register 3 (page 0,1,5,7) \*/.

Definition at line 5965 of file STM8AF\_STM8S.h.

#### 5.1.2.335 `_CAN_MDAR4`

```
#define _CAN_MDAR4 __SFR(uint8_t, CAN_AddressBase+0x08+0x09)
```

CAN mailbox data register 4 (page 0,1,5,7) \*/.

Definition at line 5966 of file STM8AF\_STM8S.h.

#### 5.1.2.336 `_CAN_MDAR5`

```
#define _CAN_MDAR5 __SFR(uint8_t, CAN_AddressBase+0x08+0x0A)
```

CAN mailbox data register 5 (page 0,1,5,7) \*/.

Definition at line 5967 of file STM8AF\_STM8S.h.

#### 5.1.2.337 `_CAN_MDAR6`

```
#define _CAN_MDAR6 __SFR(uint8_t, CAN_AddressBase+0x08+0x0B)
```

CAN mailbox data register 6 (page 0,1,5,7) \*/.

Definition at line 5968 of file STM8AF\_STM8S.h.

**5.1.2.338 \_CAN\_MDAR7**

```
#define _CAN_MDAR7 _SFR(uint8_t, CAN_AddressBase+0x08+0x0C)
```

CAN mailbox data register 7 (page 0,1,5,7) \*/.

Definition at line 5969 of file STM8AF\_STM8S.h.

**5.1.2.339 \_CAN\_MDAR8**

```
#define _CAN_MDAR8 _SFR(uint8_t, CAN_AddressBase+0x08+0x0D)
```

CAN mailbox data register 8 (page 0,1,5,7) \*/.

Definition at line 5970 of file STM8AF\_STM8S.h.

**5.1.2.340 \_CAN\_MDLCR**

```
#define _CAN_MDLCR _SFR(uint8_t, CAN_AddressBase+0x08+0x01)
```

CAN mailbox data length control register (page 0,1,5,7)

Definition at line 5958 of file STM8AF\_STM8S.h.

**5.1.2.341 \_CAN\_MDLCR\_RESET\_VALUE**

```
#define _CAN_MDLCR_RESET_VALUE ((uint8_t) 0x00)
```

CAN mailbox data length control register (page 0,1,5,7) reset value.

Definition at line 6076 of file STM8AF\_STM8S.h.

**5.1.2.342 \_CAN\_MFMIR**

```
#define _CAN_MFMIR _SFR(uint8_t, CAN_AddressBase+0x08+0x00)
```

CAN mailbox filter match index register (page 7)

Definition at line 6048 of file STM8AF\_STM8S.h.

#### 5.1.2.343 `_CAN_MFMIR_RESET_VALUE`

```
#define _CAN_MFMIR_RESET_VALUE ((uint8_t) 0x00)
```

CAN mailbox filter match index register reset value.

Definition at line 6086 of file STM8AF\_STM8S.h.

#### 5.1.2.344 `_CAN_MIDR1`

```
#define _CAN_MIDR1 _SFR(uint8_t, CAN_AddressBase+0x08+0x02)
```

CAN mailbox identifier register 1 (page 0,1,5,7)

Definition at line 5959 of file STM8AF\_STM8S.h.

#### 5.1.2.345 `_CAN_MIDR2`

```
#define _CAN_MIDR2 _SFR(uint8_t, CAN_AddressBase+0x08+0x03)
```

CAN mailbox identifier register 2 (page 0,1,5,7)

Definition at line 5960 of file STM8AF\_STM8S.h.

#### 5.1.2.346 `_CAN_MIDR3`

```
#define _CAN_MIDR3 _SFR(uint8_t, CAN_AddressBase+0x08+0x04)
```

CAN mailbox identifier register 3 (page 0,1,5,7)

Definition at line 5961 of file STM8AF\_STM8S.h.

#### 5.1.2.347 `_CAN_MIDR4`

```
#define _CAN_MIDR4 _SFR(uint8_t, CAN_AddressBase+0x08+0x05)
```

CAN mailbox identifier register 4 (page 0,1,5,7)

Definition at line 5962 of file STM8AF\_STM8S.h.

#### 5.1.2.348 \_CAN\_MSR

```
#define _CAN_MSR _SFR(uint8_t, CAN_AddressBase+0x01)
```

CAN master status register.

Definition at line 5948 of file STM8AF\_STM8S.h.

#### 5.1.2.349 \_CAN\_MSR\_RESET\_VALUE

```
#define _CAN_MSR_RESET_VALUE ((uint8_t) 0x02)
```

CAN master status register reset value.

Definition at line 6067 of file STM8AF\_STM8S.h.

#### 5.1.2.350 \_CAN\_MTSRH

```
#define _CAN_MTSRH _SFR(uint8_t, CAN_AddressBase+0x08+0x0F)
```

CAN mailbox time stamp register high byte (page 0,1,5,7) \*/.

Definition at line 5972 of file STM8AF\_STM8S.h.

#### 5.1.2.351 \_CAN\_MTSRL

```
#define _CAN_MTSRL _SFR(uint8_t, CAN_AddressBase+0x08+0x0E)
```

CAN mailbox time stamp register low byte (page 0,1,5,7) \*/.

Definition at line 5971 of file STM8AF\_STM8S.h.

#### 5.1.2.352 \_CAN\_NART

```
#define _CAN_NART ((uint8_t) (0x01 << 4))
```

CAN Channel No Automatic Retransmission [0] (in \_CAN\_MCR)

Definition at line 6093 of file STM8AF\_STM8S.h.

#### 5.1.2.353 `_CAN_PS`

```
#define _CAN_PS ((uint8_t) (0x07 << 0))
```

CAN Page select [2:0] (in `_CAN_PSR`)

Definition at line 6155 of file STM8AF\_STM8S.h.

#### 5.1.2.354 `_CAN_PS0`

```
#define _CAN_PS0 ((uint8_t) (0x01 << 0))
```

CAN Page select [0] (in `_CAN_PSR`)

Definition at line 6156 of file STM8AF\_STM8S.h.

#### 5.1.2.355 `_CAN_PS1`

```
#define _CAN_PS1 ((uint8_t) (0x01 << 1))
```

CAN Page select [1] (in `_CAN_PSR`)

Definition at line 6157 of file STM8AF\_STM8S.h.

#### 5.1.2.356 `_CAN_PS2`

```
#define _CAN_PS2 ((uint8_t) (0x01 << 2))
```

CAN Page select [2] (in `_CAN_PSR`)

Definition at line 6158 of file STM8AF\_STM8S.h.

#### 5.1.2.357 `_CAN_PSR`

```
#define _CAN_PSR \_SFR(uint8_t, CAN\_AddressBase+0x07)
```

CAN page selection for below paged registers.

Definition at line 5954 of file STM8AF\_STM8S.h.

**5.1.2.358 \_CAN\_PSR\_RESET\_VALUE**

```
#define _CAN_PSR_RESET_VALUE ((uint8_t) 0x00)
```

CAN page selection reset value.

Definition at line 6073 of file STM8AF\_STM8S.h.

**5.1.2.359 \_CAN\_RECR**

```
#define _CAN_RECR _SFR(uint8_t, CAN_AddressBase+0x08+0x03)
```

CAN receive error counter register (page 6)

Definition at line 6036 of file STM8AF\_STM8S.h.

**5.1.2.360 \_CAN\_RECR\_RESET\_VALUE**

```
#define _CAN_RECR_RESET_VALUE ((uint8_t) 0x00)
```

CAN receive error counter register (page 6) reset value.

Definition at line 6080 of file STM8AF\_STM8S.h.

**5.1.2.361 \_CAN\_RFLM**

```
#define _CAN_RFLM ((uint8_t) (0x01 << 3))
```

CAN Channel Receive FIFO Locked Mode [0] (in \_CAN\_MCR)

Definition at line 6092 of file STM8AF\_STM8S.h.

**5.1.2.362 \_CAN\_RFOM**

```
#define _CAN_RFOM ((uint8_t) (0x01 << 5))
```

CAN Release FIFO Output Mailbox [0] (in \_CAN\_RFR)

Definition at line 6135 of file STM8AF\_STM8S.h.

#### 5.1.2.363 `_CAN_RFR`

```
#define _CAN_RFR _SFR(uint8_t, CAN_AddressBase+0x04)
```

CAN receive FIFO register.

Definition at line 5951 of file STM8AF\_STM8S.h.

#### 5.1.2.364 `_CAN_RFR_RESET_VALUE`

```
#define _CAN_RFR_RESET_VALUE ((uint8_t) 0x00)
```

CAN receive FIFO register reset value.

Definition at line 6070 of file STM8AF\_STM8S.h.

#### 5.1.2.365 `_CAN_RQCP`

```
#define _CAN_RQCP ((uint8_t) (0x01 << 2))
```

CAN Request completed [0] (in `_CAN_MCSR`, page 0,1,5)

Definition at line 6164 of file STM8AF\_STM8S.h.

#### 5.1.2.366 `_CAN_RQCP0`

```
#define _CAN_RQCP0 ((uint8_t) (0x01 << 0))
```

CAN Request Completed for Mailbox 0 [0] (in `_CAN_TSR`)

Definition at line 6108 of file STM8AF\_STM8S.h.

#### 5.1.2.367 `_CAN_RQCP1`

```
#define _CAN_RQCP1 ((uint8_t) (0x01 << 1))
```

CAN Request Completed for Mailbox 1 [0] (in `_CAN_TSR`)

Definition at line 6109 of file STM8AF\_STM8S.h.



**5.1.2.368 \_CAN\_RQCP2**

```
#define _CAN_RQCP2 ((uint8_t) (0x01 << 2))
```

CAN Request Completed for Mailbox 2 [0] (in \_CAN\_TSR)

Definition at line 6110 of file STM8AF\_STM8S.h.

**5.1.2.369 \_CAN\_RTR**

```
#define _CAN_RTR ((uint8_t) (0x01 << 5))
```

CAN Remote transmission request [0] (in \_CAN\_MIDR1, page 0,1,5)

Definition at line 6181 of file STM8AF\_STM8S.h.

**5.1.2.370 \_CAN\_RX**

```
#define _CAN_RX ((uint8_t) (0x01 << 5))
```

CAN Receive [0] (in \_CAN\_MSR)

Definition at line 6104 of file STM8AF\_STM8S.h.

**5.1.2.371 \_CAN\_RXS**

```
#define _CAN_RXS ((uint8_t) (0x01 << 3))
```

CAN Rx Signal (=pin status) [0] (in \_CAN\_DGR)

Definition at line 6150 of file STM8AF\_STM8S.h.

**5.1.2.372 \_CAN\_SAMP**

```
#define _CAN_SAMP ((uint8_t) (0x01 << 2))
```

CAN Last sample point [0] (in \_CAN\_DGR)

Definition at line 6149 of file STM8AF\_STM8S.h.

#### 5.1.2.373 `_CAN_SILM`

```
#define _CAN_SILM ((uint8_t) (0x01 << 1))
```

CAN Silent mode [0] (in `_CAN_DGR`)

Definition at line 6148 of file `STM8AF_STM8S.h`.

#### 5.1.2.374 `_CAN_SJW`

```
#define _CAN_SJW ((uint8_t) (0x03 << 6))
```

CAN Resynchronization jump width [1:0] (in `_CAN_EIER`, page 6)

Definition at line 6214 of file `STM8AF_STM8S.h`.

#### 5.1.2.375 `_CAN_SJW0`

```
#define _CAN_SJW0 ((uint8_t) (0x01 << 6))
```

CAN Resynchronization jump width [0] (in `_CAN_EIER`, page 6)

Definition at line 6215 of file `STM8AF_STM8S.h`.

#### 5.1.2.376 `_CAN_SJW1`

```
#define _CAN_SJW1 ((uint8_t) (0x01 << 7))
```

CAN Resynchronization jump width [1] (in `_CAN_EIER`, page 6)

Definition at line 6216 of file `STM8AF_STM8S.h`.

#### 5.1.2.377 `_CAN_SLAK`

```
#define _CAN_SLAK ((uint8_t) (0x01 << 1))
```

CAN Sleep Acknowledge [0] (in `_CAN_MSR`)

Definition at line 6100 of file `STM8AF_STM8S.h`.

**5.1.2.378 \_CAN\_SLEEP**

```
#define _CAN_SLEEP ((uint8_t) (0x01 << 1))
```

CAN Channel Sleep Mode Request [0] (in \_CAN\_MCR)

Definition at line 6090 of file STM8AF\_STM8S.h.

**5.1.2.379 \_CAN\_TECR**

```
#define _CAN_TECR _SFR(uint8_t, CAN_AddressBase+0x08+0x02)
```

CAN transmit error counter register (page 6)

Definition at line 6035 of file STM8AF\_STM8S.h.

**5.1.2.380 \_CAN\_TECR\_RESET\_VALUE**

```
#define _CAN_TECR_RESET_VALUE ((uint8_t) 0x00)
```

CAN transmit error counter register (page 6) reset value.

Definition at line 6079 of file STM8AF\_STM8S.h.

**5.1.2.381 \_CAN\_TERR**

```
#define _CAN_TERR ((uint8_t) (0x01 << 5))
```

CAN Transmission error [0] (in \_CAN\_MCSR, page 0,1,5)

Definition at line 6167 of file STM8AF\_STM8S.h.

**5.1.2.382 \_CAN\_TGT**

```
#define _CAN_TGT ((uint8_t) (0x01 << 7))
```

CAN Transmit global time [0] (in \_CAN\_MDLCR, page 0,1,5,7)

Definition at line 6177 of file STM8AF\_STM8S.h.

#### 5.1.2.383 `_CAN_TME0`

```
#define _CAN_TME0 ((uint8_t) (0x01 << 2))
```

CAN Transmit Mailbox 0 Empty [0] (in `_CAN_TPR`)

Definition at line 6121 of file `STM8AF_STM8S.h`.

#### 5.1.2.384 `_CAN_TME1`

```
#define _CAN_TME1 ((uint8_t) (0x01 << 3))
```

CAN Transmit Mailbox 1 Empty [0] (in `_CAN_TPR`)

Definition at line 6122 of file `STM8AF_STM8S.h`.

#### 5.1.2.385 `_CAN_TME2`

```
#define _CAN_TME2 ((uint8_t) (0x01 << 4))
```

CAN Transmit Mailbox 2 Empty [0] (in `_CAN_TPR`)

Definition at line 6123 of file `STM8AF_STM8S.h`.

#### 5.1.2.386 `_CAN_TMEIE`

```
#define _CAN_TMEIE ((uint8_t) (0x01 << 0))
```

CAN Transmit Mailbox Empty Interrupt Enable [0] (in `_CAN_IER`)

Definition at line 6139 of file `STM8AF_STM8S.h`.

#### 5.1.2.387 `_CAN_TPR`

```
#define _CAN_TPR \_SFR(uint8_t, CAN\_AddressBase+0x03)
```

CAN transmit priority register.

Definition at line 5950 of file `STM8AF_STM8S.h`.

**5.1.2.388 \_CAN\_TPR\_RESET\_VALUE**

```
#define _CAN_TPR_RESET_VALUE ((uint8_t) 0x0C)
```

CAN transmit priority register reset value.

Definition at line 6069 of file STM8AF\_STM8S.h.

**5.1.2.389 \_CAN\_TSR**

```
#define _CAN_TSR _SFR(uint8_t, CAN_AddressBase+0x02)
```

CAN transmit status register.

Definition at line 5949 of file STM8AF\_STM8S.h.

**5.1.2.390 \_CAN\_TSR\_RESET\_VALUE**

```
#define _CAN_TSR_RESET_VALUE ((uint8_t) 0x00)
```

CAN transmit status register reset value.

Definition at line 6068 of file STM8AF\_STM8S.h.

**5.1.2.391 \_CAN\_TTCM**

```
#define _CAN_TTCM ((uint8_t) (0x01 << 7))
```

CAN Channel Time Triggered Communication Mode [0] (in \_CAN\_MCR)

Definition at line 6096 of file STM8AF\_STM8S.h.

**5.1.2.392 \_CAN\_TX**

```
#define _CAN_TX ((uint8_t) (0x01 << 4))
```

CAN Transmit [0] (in \_CAN\_MSR)

Definition at line 6103 of file STM8AF\_STM8S.h.

#### 5.1.2.393 `_CAN_TXFP`

```
#define _CAN_TXFP ((uint8_t) (0x01 << 2))
```

CAN Channel Transmit FIFO Priority [0] (in `_CAN_MCR`)

Definition at line 6091 of file `STM8AF_STM8S.h`.

#### 5.1.2.394 `_CAN_TXM2E`

```
#define _CAN_TXM2E ((uint8_t) (0x01 << 4))
```

CAN TX Mailbox 2 enable [0] (in `_CAN_DGR`)

Definition at line 6151 of file `STM8AF_STM8S.h`.

#### 5.1.2.395 `_CAN_TXOK`

```
#define _CAN_TXOK ((uint8_t) (0x01 << 3))
```

CAN Transmission OK [0] (in `_CAN_MCSR`, page 0,1,5)

Definition at line 6165 of file `STM8AF_STM8S.h`.

#### 5.1.2.396 `_CAN_TXOK0`

```
#define _CAN_TXOK0 ((uint8_t) (0x01 << 4))
```

CAN Transmission ok for Mailbox 0 [0] (in `_CAN_TSR`)

Definition at line 6112 of file `STM8AF_STM8S.h`.

#### 5.1.2.397 `_CAN_TXOK1`

```
#define _CAN_TXOK1 ((uint8_t) (0x01 << 5))
```

CAN Transmission ok for Mailbox 1 [0] (in `_CAN_TSR`)

Definition at line 6113 of file `STM8AF_STM8S.h`.

**5.1.2.398 \_CAN\_TXOK2**

```
#define _CAN_TXOK2 ((uint8_t) (0x01 << 6))
```

CAN Transmission ok for Mailbox 2 [0] (in \_CAN\_TSR)

Definition at line 6114 of file STM8AF\_STM8S.h.

**5.1.2.399 \_CAN\_TXRQ**

```
#define _CAN_TXRQ ((uint8_t) (0x01 << 0))
```

CAN Transmission mailbox request [0] (in \_CAN\_MCSR, page 0,1,5)

Definition at line 6162 of file STM8AF\_STM8S.h.

**5.1.2.400 \_CAN\_WKUI**

```
#define _CAN_WKUI ((uint8_t) (0x01 << 3))
```

CAN Wakeup Interrupt [0] (in \_CAN\_MSR)

Definition at line 6102 of file STM8AF\_STM8S.h.

**5.1.2.401 \_CAN\_WKUIE**

```
#define _CAN_WKUIE ((uint8_t) (0x01 << 7))
```

CAN Wakeup Interrupt Enable [0] (in \_CAN\_IER)

Definition at line 6144 of file STM8AF\_STM8S.h.

**5.1.2.402 \_CFG**

```
#define _CFG _SFR(CFG_t, CFG_AddressBase)
```

CFG struct/bit access.

Definition at line 6305 of file STM8AF\_STM8S.h.

#### 5.1.2.403 \_CFG\_AL

```
#define _CFG_AL ((uint8_t) (0x01 << 1))
```

Activation level [0].

Definition at line 6313 of file STM8AF\_STM8S.h.

#### 5.1.2.404 \_CFG\_GCR

```
#define _CFG_GCR _SFR(uint8_t, CFG_AddressBase+0x00)
```

Global configuration register (CFG\_GCR)

Definition at line 6306 of file STM8AF\_STM8S.h.

#### 5.1.2.405 \_CFG\_GCR\_RESET\_VALUE

```
#define _CFG_GCR_RESET_VALUE ((uint8_t)0x00)
```

Definition at line 6309 of file STM8AF\_STM8S.h.

#### 5.1.2.406 \_CFG\_SWD

```
#define _CFG_SWD ((uint8_t) (0x01 << 0))
```

SWIM disable [0].

Definition at line 6312 of file STM8AF\_STM8S.h.

#### 5.1.2.407 \_CLK

```
#define _CLK _SFR(CLK_t, CLK_AddressBase)
```

Clock module struct/bit access.

Definition at line 865 of file STM8AF\_STM8S.h.



**5.1.2.408 \_CLK\_ADC**

```
#define _CLK_ADC ((uint8_t) (0x01 << 3))
```

clock enable ADC [0] (in \_CLK\_PCKENR2)

Definition at line 962 of file STM8AF\_STM8S.h.

**5.1.2.409 \_CLK\_AUX**

```
#define _CLK_AUX ((uint8_t) (0x01 << 1))
```

Auxiliary oscillator connected to master clock [0] (in \_CLK\_CSSR)

Definition at line 943 of file STM8AF\_STM8S.h.

**5.1.2.410 \_CLK\_AWU**

```
#define _CLK_AWU ((uint8_t) (0x01 << 2))
```

clock enable AWU [0] (in \_CLK\_PCKENR2)

Definition at line 961 of file STM8AF\_STM8S.h.

**5.1.2.411 \_CLK\_CAN**

```
#define _CLK_CAN ((uint8_t) (0x01 << 7))
```

clock enable CAN [0] (in \_CLK\_PCKENR2)

Definition at line 964 of file STM8AF\_STM8S.h.

**5.1.2.412 \_CLK\_CCBSY**

```
#define _CLK_CCBSY ((uint8_t) (0x01 << 6))
```

Configurable clock output busy [0] (in \_CLK\_CCOR)

Definition at line 956 of file STM8AF\_STM8S.h.

#### 5.1.2.413 \_CLK\_CCOEN

```
#define _CLK_CCOEN ((uint8_t) (0x01 << 0))
```

Configurable clock output enable [0] (in \_CLK\_CCOR)

Definition at line 949 of file STM8AF\_STM8S.h.

#### 5.1.2.414 \_CLK\_CCOR

```
#define _CLK_CCOR _SFR(uint8_t, CLK_AddressBase+0x09)
```

Configurable clock output register.

Definition at line 875 of file STM8AF\_STM8S.h.

#### 5.1.2.415 \_CLK\_CCOR\_RESET\_VALUE

```
#define _CLK_CCOR_RESET_VALUE ((uint8_t) 0x00)
```

Configurable clock output register reset value.

Definition at line 891 of file STM8AF\_STM8S.h.

#### 5.1.2.416 \_CLK\_CCORDY

```
#define _CLK_CCORDY ((uint8_t) (0x01 << 5))
```

Configurable clock output ready [0] (in \_CLK\_CCOR)

Definition at line 955 of file STM8AF\_STM8S.h.

#### 5.1.2.417 \_CLK\_CCOSEL

```
#define _CLK_CCOSEL ((uint8_t) (0x0F << 1))
```

Configurable clock output selection [3:0] (in \_CLK\_CCOR)

Definition at line 950 of file STM8AF\_STM8S.h.

**5.1.2.418 \_CLK\_CCOSSEL0**

```
#define _CLK_CCOSSEL0 ((uint8_t) (0x01 << 1))
```

Configurable clock output selection [0] (in \_CLK\_CCOR)

Definition at line 951 of file STM8AF\_STM8S.h.

**5.1.2.419 \_CLK\_CCOSSEL1**

```
#define _CLK_CCOSSEL1 ((uint8_t) (0x01 << 2))
```

Configurable clock output selection [1] (in \_CLK\_CCOR)

Definition at line 952 of file STM8AF\_STM8S.h.

**5.1.2.420 \_CLK\_CCOSSEL2**

```
#define _CLK_CCOSSEL2 ((uint8_t) (0x01 << 3))
```

Configurable clock output selection [2] (in \_CLK\_CCOR)

Definition at line 953 of file STM8AF\_STM8S.h.

**5.1.2.421 \_CLK\_CCOSSEL3**

```
#define _CLK_CCOSSEL3 ((uint8_t) (0x01 << 4))
```

Configurable clock output selection [3] (in \_CLK\_CCOR)

Definition at line 954 of file STM8AF\_STM8S.h.

**5.1.2.422 \_CLK\_CKDIVR**

```
#define _CLK_CKDIVR _SFR(uint8_t, CLK_AddressBase+0x06)
```

Clock divider register.

Definition at line 872 of file STM8AF\_STM8S.h.

#### 5.1.2.423 \_CLK\_CKDIVR\_RESET\_VALUE

```
#define _CLK_CKDIVR_RESET_VALUE ((uint8_t) 0x18)
```

Clock divider register reset value.

Definition at line 887 of file STM8AF\_STM8S.h.

#### 5.1.2.424 \_CLK\_CMSR

```
#define _CLK_CMSR _SFR(uint8_t, CLK_AddressBase+0x03)
```

Clock master status register.

Definition at line 869 of file STM8AF\_STM8S.h.

#### 5.1.2.425 \_CLK\_CMSR\_RESET\_VALUE

```
#define _CLK_CMSR_RESET_VALUE ((uint8_t) 0xE1)
```

Clock master status reset value.

Definition at line 884 of file STM8AF\_STM8S.h.

#### 5.1.2.426 \_CLK\_CPUDIV

```
#define _CLK_CPUDIV ((uint8_t) (0x07 << 0))
```

CPU clock prescaler [2:0] (in \_CLK\_CKDIVR)

Definition at line 922 of file STM8AF\_STM8S.h.

#### 5.1.2.427 \_CLK\_CPUDIV0

```
#define _CLK_CPUDIV0 ((uint8_t) (0x01 << 0))
```

CPU clock prescaler [0] (in \_CLK\_CKDIVR)

Definition at line 923 of file STM8AF\_STM8S.h.

**5.1.2.428 \_CLK\_CPUDIV1**

```
#define _CLK_CPUDIV1 ((uint8_t) (0x01 << 1))
```

CPU clock prescaler [1] (in \_CLK\_CKDIVR)

Definition at line 924 of file STM8AF\_STM8S.h.

**5.1.2.429 \_CLK\_CPUDIV2**

```
#define _CLK_CPUDIV2 ((uint8_t) (0x01 << 2))
```

CPU clock prescaler [2] (in \_CLK\_CKDIVR)

Definition at line 925 of file STM8AF\_STM8S.h.

**5.1.2.430 \_CLK\_CSSD**

```
#define _CLK_CSSD ((uint8_t) (0x01 << 3))
```

Clock security system detection [0] (in \_CLK\_CSSR)

Definition at line 945 of file STM8AF\_STM8S.h.

**5.1.2.431 \_CLK\_CSSDIE**

```
#define _CLK_CSSDIE ((uint8_t) (0x01 << 2))
```

Clock security system detection interrupt enable [0] (in \_CLK\_CSSR)

Definition at line 944 of file STM8AF\_STM8S.h.

**5.1.2.432 \_CLK\_CSSEN**

```
#define _CLK_CSSEN ((uint8_t) (0x01 << 0))
```

Clock security system enable [0] (in \_CLK\_CSSR)

Definition at line 942 of file STM8AF\_STM8S.h.

#### 5.1.2.433 \_CLK\_CSSR

```
#define _CLK_CSSR _SFR(uint8_t, CLK_AddressBase+0x08)
```

Clock security system register.

Definition at line 874 of file STM8AF\_STM8S.h.

#### 5.1.2.434 \_CLK\_CSSR\_RESET\_VALUE

```
#define _CLK_CSSR_RESET_VALUE ((uint8_t) 0x00)
```

Clock security system register reset value.

Definition at line 890 of file STM8AF\_STM8S.h.

#### 5.1.2.435 \_CLK\_ECKR

```
#define _CLK_ECKR _SFR(uint8_t, CLK_AddressBase+0x01)
```

External clock register.

Definition at line 867 of file STM8AF\_STM8S.h.

#### 5.1.2.436 \_CLK\_ECKR\_HSERDY

```
#define _CLK_ECKR_HSERDY ((uint8_t) (0x01 << 1))
```

High speed external crystal oscillator ready [0] (in \_CLK\_ECKR)

Definition at line 906 of file STM8AF\_STM8S.h.

#### 5.1.2.437 \_CLK\_ECKR\_RESET\_VALUE

```
#define _CLK_ECKR_RESET_VALUE ((uint8_t) 0x00)
```

External clock register reset value.

Definition at line 883 of file STM8AF\_STM8S.h.

**5.1.2.438 \_CLK\_FHWU**

```
#define _CLK_FHWU ((uint8_t) (0x01 << 2))
```

Fast wakeup from Halt/Active-halt modes [0] (in \_CLK\_ICKR)

Definition at line 898 of file STM8AF\_STM8S.h.

**5.1.2.439 \_CLK\_HSEEN**

```
#define _CLK_HSEEN ((uint8_t) (0x01 << 0))
```

High speed external crystal oscillator enable [0] (in \_CLK\_ECKR)

Definition at line 905 of file STM8AF\_STM8S.h.

**5.1.2.440 \_CLK\_HSIDIV**

```
#define _CLK_HSIDIV ((uint8_t) (0x03 << 3))
```

High speed internal clock prescaler [1:0] (in \_CLK\_CKDIVR)

Definition at line 926 of file STM8AF\_STM8S.h.

**5.1.2.441 \_CLK\_HSIDIV0**

```
#define _CLK_HSIDIV0 ((uint8_t) (0x01 << 3))
```

High speed internal clock prescaler [0] (in \_CLK\_CKDIVR)

Definition at line 927 of file STM8AF\_STM8S.h.

**5.1.2.442 \_CLK\_HSIDIV1**

```
#define _CLK_HSIDIV1 ((uint8_t) (0x01 << 4))
```

High speed internal clock prescaler [1] (in \_CLK\_CKDIVR)

Definition at line 928 of file STM8AF\_STM8S.h.

#### 5.1.2.443 \_CLK\_HSIEN

```
#define _CLK_HSIEN ((uint8_t) (0x01 << 0))
```

High speed internal RC oscillator enable [0] (in \_CLK\_ICKR)

Definition at line 896 of file STM8AF\_STM8S.h.

#### 5.1.2.444 \_CLK\_HSIIRDY

```
#define _CLK_HSIIRDY ((uint8_t) (0x01 << 1))
```

High speed internal oscillator ready [0] (in \_CLK\_ICKR)

Definition at line 897 of file STM8AF\_STM8S.h.

#### 5.1.2.445 \_CLK\_HSITRIM

```
#define _CLK_HSITRIM ((uint8_t) (0x0F << 0))
```

HSI trimming value (some devices only support 3 bits, see DS!) [3:0] (in \_CLK\_HSITRIMR)

Definition at line 967 of file STM8AF\_STM8S.h.

#### 5.1.2.446 \_CLK\_HSITRIM0

```
#define _CLK_HSITRIM0 ((uint8_t) (0x01 << 0))
```

HSI trimming value [0] (in \_CLK\_HSITRIMR)

Definition at line 968 of file STM8AF\_STM8S.h.

#### 5.1.2.447 \_CLK\_HSITRIM1

```
#define _CLK_HSITRIM1 ((uint8_t) (0x01 << 1))
```

HSI trimming value [1] (in \_CLK\_HSITRIMR)

Definition at line 969 of file STM8AF\_STM8S.h.



**5.1.2.448 \_CLK\_HSITRIM2**

```
#define _CLK_HSITRIM2 ((uint8_t) (0x01 << 2))
```

HSI trimming value [2] (in \_CLK\_HSITRIMR)

Definition at line 970 of file STM8AF\_STM8S.h.

**5.1.2.449 \_CLK\_HSITRIM3**

```
#define _CLK_HSITRIM3 ((uint8_t) (0x01 << 3))
```

HSI trimming value [3] (in \_CLK\_HSITRIMR)

Definition at line 971 of file STM8AF\_STM8S.h.

**5.1.2.450 \_CLK\_HSITRIMR**

```
#define _CLK_HSITRIMR \_SFR(uint8_t, CLK\_AddressBase+0x0C)
```

HSI clock calibration trimming register.

Definition at line 878 of file STM8AF\_STM8S.h.

**5.1.2.451 \_CLK\_HSITRIMR\_RESET\_VALUE**

```
#define _CLK_HSITRIMR_RESET_VALUE ((uint8_t) 0x00)
```

HSI clock calibration trimming register reset value.

Definition at line 892 of file STM8AF\_STM8S.h.

**5.1.2.452 \_CLK\_I2C**

```
#define _CLK_I2C ((uint8_t) (0x01 << 0))
```

clock enable I2C [0] (in \_CLK\_PCKENR1)

Definition at line 932 of file STM8AF\_STM8S.h.

#### 5.1.2.453 \_CLK\_ICKR

```
#define _CLK_ICKR _SFR(uint8_t, CLK_AddressBase+0x00)
```

Internal clock register.

Definition at line 866 of file STM8AF\_STM8S.h.

#### 5.1.2.454 \_CLK\_ICKR\_RESET\_VALUE

```
#define _CLK_ICKR_RESET_VALUE ((uint8_t) 0x01)
```

Internal clock register reset value.

Definition at line 882 of file STM8AF\_STM8S.h.

#### 5.1.2.455 \_CLK\_LSIEN

```
#define _CLK_LSIEN ((uint8_t) (0x01 << 3))
```

Low speed internal RC oscillator enable [0] (in \_CLK\_ICKR)

Definition at line 899 of file STM8AF\_STM8S.h.

#### 5.1.2.456 \_CLK\_LSIRDY

```
#define _CLK_LSIRDY ((uint8_t) (0x01 << 4))
```

Low speed internal oscillator ready [0] (in \_CLK\_ICKR)

Definition at line 900 of file STM8AF\_STM8S.h.

#### 5.1.2.457 \_CLK\_PCKENR1

```
#define _CLK_PCKENR1 _SFR(uint8_t, CLK_AddressBase+0x07)
```

Peripheral clock gating register 1.

Definition at line 873 of file STM8AF\_STM8S.h.

**5.1.2.458 \_CLK\_PCKENR1\_RESET\_VALUE**

```
#define _CLK_PCKENR1_RESET_VALUE ((uint8_t) 0xFF)
```

Peripheral clock gating register 1 reset value.

Definition at line 888 of file STM8AF\_STM8S.h.

**5.1.2.459 \_CLK\_PCKENR2**

```
#define _CLK_PCKENR2 _SFR(uint8_t, CLK_AddressBase+0x0A)
```

Peripheral clock gating register 2.

Definition at line 876 of file STM8AF\_STM8S.h.

**5.1.2.460 \_CLK\_PCKENR2\_RESET\_VALUE**

```
#define _CLK_PCKENR2_RESET_VALUE ((uint8_t) 0xFF)
```

Peripheral clock gating register 2 reset value.

Definition at line 889 of file STM8AF\_STM8S.h.

**5.1.2.461 \_CLK\_REGAH**

```
#define _CLK_REGAH ((uint8_t) (0x01 << 5))
```

Regulator power off in Active-halt mode [0] (in \_CLK\_ICKR)

Definition at line 901 of file STM8AF\_STM8S.h.

**5.1.2.462 \_CLK\_SPI**

```
#define _CLK_SPI ((uint8_t) (0x01 << 1))
```

clock enable SPI [0] (in \_CLK\_PCKENR1)

Definition at line 933 of file STM8AF\_STM8S.h.

#### 5.1.2.463 \_CLK\_SWBSY

```
#define _CLK_SWBSY ((uint8_t) (0x01 << 0))
```

Switch busy flag [0] (in \_CLK\_SWCR)

Definition at line 915 of file STM8AF\_STM8S.h.

#### 5.1.2.464 \_CLK\_SWCR

```
#define _CLK_SWCR _SFR(uint8_t, CLK_AddressBase+0x05)
```

Clock switch control register.

Definition at line 871 of file STM8AF\_STM8S.h.

#### 5.1.2.465 \_CLK\_SWCR\_RESET\_VALUE

```
#define _CLK_SWCR_RESET_VALUE ((uint8_t) 0x00)
```

Clock switch control reset value.

Definition at line 886 of file STM8AF\_STM8S.h.

#### 5.1.2.466 \_CLK\_SWEN

```
#define _CLK_SWEN ((uint8_t) (0x01 << 1))
```

Switch start/stop enable [0] (in \_CLK\_SWCR)

Definition at line 916 of file STM8AF\_STM8S.h.

#### 5.1.2.467 \_CLK\_SWI\_HSE

```
#define _CLK_SWI_HSE ((uint8_t) 0xB4)
```

write to CLK\_SWR for HSE clock (in \_CLK\_SWR)

Definition at line 912 of file STM8AF\_STM8S.h.

**5.1.2.468 \_CLK\_SWI\_HSI**

```
#define _CLK_SWI_HSI ((uint8_t) 0xE1)
```

write to CLK\_SWR for HSI clock (in \_CLK\_SWR)

Definition at line 910 of file STM8AF\_STM8S.h.

**5.1.2.469 \_CLK\_SWI\_LSI**

```
#define _CLK_SWI_LSI ((uint8_t) 0xD2)
```

write to CLK\_SWR for LSI clock (in \_CLK\_SWR)

Definition at line 911 of file STM8AF\_STM8S.h.

**5.1.2.470 \_CLK\_SWIEN**

```
#define _CLK_SWIEN ((uint8_t) (0x01 << 2))
```

Clock switch interrupt enable [0] (in \_CLK\_SWCR)

Definition at line 917 of file STM8AF\_STM8S.h.

**5.1.2.471 \_CLK\_SWIF**

```
#define _CLK_SWIF ((uint8_t) (0x01 << 3))
```

Clock switch interrupt flag [0] (in \_CLK\_SWCR)

Definition at line 918 of file STM8AF\_STM8S.h.

**5.1.2.472 \_CLK\_SWIMCCR**

```
#define _CLK_SWIMCCR _SFR(uint8_t, CLK_AddressBase+0x0D)
```

SWIM clock control register.

Definition at line 879 of file STM8AF\_STM8S.h.

#### 5.1.2.473 `_CLK_SWIMCCR_RESET_VALUE`

```
#define _CLK_SWIMCCR_RESET_VALUE ((uint8_t) 0x00)
```

SWIM clock control register reset value.

Definition at line 893 of file STM8AF\_STM8S.h.

#### 5.1.2.474 `_CLK_SWIMCLK`

```
#define _CLK_SWIMCLK ((uint8_t) (0x01 << 0))
```

SWIM clock divider [0] (in `_CLK_SWIMCCR`)

Definition at line 975 of file STM8AF\_STM8S.h.

#### 5.1.2.475 `_CLK_SWR`

```
#define _CLK_SWR \_SFR(uint8_t, CLK\_AddressBase+0x04)
```

Clock master switch register.

Definition at line 870 of file STM8AF\_STM8S.h.

#### 5.1.2.476 `_CLK_SWR_RESET_VALUE`

```
#define _CLK_SWR_RESET_VALUE ((uint8_t) 0xE1)
```

Clock master switch reset value.

Definition at line 885 of file STM8AF\_STM8S.h.

#### 5.1.2.477 `_CLK_TIM1`

```
#define _CLK_TIM1 ((uint8_t) (0x01 << 7))
```

clock enable TIM1 [0] (in `_CLK_PCKENR1`)

Definition at line 939 of file STM8AF\_STM8S.h.

**5.1.2.478 \_CLK\_TIM2\_TIM5**

```
#define _CLK_TIM2_TIM5 ((uint8_t) (0x01 << 5))
```

clock enable TIM2/TIM5 [0] (in \_CLK\_PCKENR1)

Definition at line 937 of file STM8AF\_STM8S.h.

**5.1.2.479 \_CLK\_TIM3**

```
#define _CLK_TIM3 ((uint8_t) (0x01 << 6))
```

clock enable TIM3 [0] (in \_CLK\_PCKENR1)

Definition at line 938 of file STM8AF\_STM8S.h.

**5.1.2.480 \_CLK\_TIM4\_TIM6**

```
#define _CLK_TIM4_TIM6 ((uint8_t) (0x01 << 4))
```

clock enable TIM4/TIM6 [0] (in \_CLK\_PCKENR1)

Definition at line 936 of file STM8AF\_STM8S.h.

**5.1.2.481 \_CLK\_UART1**

```
#define _CLK_UART1 ((uint8_t) (0x01 << 2))
```

clock enable UART1 [0] (in \_CLK\_PCKENR1)

Definition at line 934 of file STM8AF\_STM8S.h.

**5.1.2.482 \_CLK\_UART2**

```
#define _CLK_UART2 ((uint8_t) (0x01 << 3))
```

clock enable UART2 [0] (in \_CLK\_PCKENR1)

Definition at line 935 of file STM8AF\_STM8S.h.

#### 5.1.2.483 `_EXTI`

```
#define _EXTI _SFR(EXTI_t, EXTI_AddressBase)
```

External interrupt struct/bit access.

Definition at line 671 of file STM8AF\_STM8S.h.

#### 5.1.2.484 `_EXTI_CR1`

```
#define _EXTI_CR1 _SFR(uint8_t, EXTI_AddressBase+0x00)
```

External interrupt control register 1 (EXTI\_CR1)

Definition at line 672 of file STM8AF\_STM8S.h.

#### 5.1.2.485 `_EXTI_CR1_RESET_VALUE`

```
#define _EXTI_CR1_RESET_VALUE ((uint8_t) 0x00)
```

External interrupt control register 1 reset value.

Definition at line 676 of file STM8AF\_STM8S.h.

#### 5.1.2.486 `_EXTI_CR2`

```
#define _EXTI_CR2 _SFR(uint8_t, EXTI_AddressBase+0x01)
```

External interrupt control register 2 (EXTI\_CR2)

Definition at line 673 of file STM8AF\_STM8S.h.

#### 5.1.2.487 `_EXTI_CR2_RESET_VALUE`

```
#define _EXTI_CR2_RESET_VALUE ((uint8_t) 0x00)
```

External interrupt control register 2 reset value.

Definition at line 677 of file STM8AF\_STM8S.h.



**5.1.2.488 \_EXTI\_PAIS**

```
#define _EXTI_PAIS ((uint8_t) (0x03 << 0))
```

External interrupt sensitivity for Port A [1:0] (in \_EXTI\_CR1)

Definition at line 680 of file STM8AF\_STM8S.h.

**5.1.2.489 \_EXTI\_PAIS0**

```
#define _EXTI_PAIS0 ((uint8_t) (0x01 << 0))
```

External interrupt sensitivity for Port A [0] (in \_EXTI\_CR1)

Definition at line 681 of file STM8AF\_STM8S.h.

**5.1.2.490 \_EXTI\_PAIS1**

```
#define _EXTI_PAIS1 ((uint8_t) (0x01 << 1))
```

External interrupt sensitivity for Port A [1] (in \_EXTI\_CR1)

Definition at line 682 of file STM8AF\_STM8S.h.

**5.1.2.491 \_EXTI\_PBIS**

```
#define _EXTI_PBIS ((uint8_t) (0x03 << 2))
```

External interrupt sensitivity for Port B [1:0] (in \_EXTI\_CR1)

Definition at line 683 of file STM8AF\_STM8S.h.

**5.1.2.492 \_EXTI\_PBIS0**

```
#define _EXTI_PBIS0 ((uint8_t) (0x01 << 2))
```

External interrupt sensitivity for Port B [0] (in \_EXTI\_CR1)

Definition at line 684 of file STM8AF\_STM8S.h.

**5.1.2.493 \_EXTI\_PBS1**

```
#define _EXTI_PBS1 ((uint8_t) (0x01 << 3))
```

External interrupt sensitivity for Port B [1] (in \_EXTI\_CR1)

Definition at line 685 of file STM8AF\_STM8S.h.

**5.1.2.494 \_EXTI\_PCIS**

```
#define _EXTI_PCIS ((uint8_t) (0x03 << 4))
```

External interrupt sensitivity for Port C [1:0] (in \_EXTI\_CR1)

Definition at line 686 of file STM8AF\_STM8S.h.

**5.1.2.495 \_EXTI\_PCIS0**

```
#define _EXTI_PCIS0 ((uint8_t) (0x01 << 4))
```

External interrupt sensitivity for Port C [0] (in \_EXTI\_CR1)

Definition at line 687 of file STM8AF\_STM8S.h.

**5.1.2.496 \_EXTI\_PCIS1**

```
#define _EXTI_PCIS1 ((uint8_t) (0x01 << 5))
```

External interrupt sensitivity for Port C [1] (in \_EXTI\_CR1)

Definition at line 688 of file STM8AF\_STM8S.h.

**5.1.2.497 \_EXTI\_PDIS**

```
#define _EXTI_PDIS ((uint8_t) (0x03 << 6))
```

External interrupt sensitivity for Port D [1:0] (in \_EXTI\_CR1)

Definition at line 689 of file STM8AF\_STM8S.h.

**5.1.2.498 \_EXTI\_PDIS0**

```
#define _EXTI_PDIS0 ((uint8_t) (0x01 << 6))
```

External interrupt sensitivity for Port D [0] (in \_EXTI\_CR1)

Definition at line 690 of file STM8AF\_STM8S.h.

**5.1.2.499 \_EXTI\_PDIS1**

```
#define _EXTI_PDIS1 ((uint8_t) (0x01 << 7))
```

External interrupt sensitivity for Port D [1] (in \_EXTI\_CR1)

Definition at line 691 of file STM8AF\_STM8S.h.

**5.1.2.500 \_EXTI\_PEIS**

```
#define _EXTI_PEIS ((uint8_t) (0x03 << 0))
```

Port E external interrupt sensitivity bits [1:0] (in \_EXTI\_CR2)

Definition at line 694 of file STM8AF\_STM8S.h.

**5.1.2.501 \_EXTI\_PEIS0**

```
#define _EXTI_PEIS0 ((uint8_t) (0x01 << 0))
```

Port E external interrupt sensitivity bits [0] (in \_EXTI\_CR2)

Definition at line 695 of file STM8AF\_STM8S.h.

**5.1.2.502 \_EXTI\_PEIS1**

```
#define _EXTI_PEIS1 ((uint8_t) (0x01 << 1))
```

Port E external interrupt sensitivity bits [1] (in \_EXTI\_CR2)

Definition at line 696 of file STM8AF\_STM8S.h.

#### 5.1.2.503 \_EXTI\_TLIS

```
#define _EXTI_TLIS ((uint8_t) (0x01 << 2))
```

Top level interrupt sensitivity [0] (in \_EXTI\_CR2)

Definition at line 697 of file STM8AF\_STM8S.h.

#### 5.1.2.504 \_FLASH

```
#define _FLASH __SFR(FLASH_t, FLASH_AddressBase)
```

Flash struct/bit access.

Definition at line 586 of file STM8AF\_STM8S.h.

#### 5.1.2.505 \_FLASH\_AHALT

```
#define _FLASH_AHALT ((uint8_t) (0x01 << 2))
```

Power-down in Active-halt mode [0] (in \_FLASH\_CR1)

Definition at line 609 of file STM8AF\_STM8S.h.

#### 5.1.2.506 \_FLASH\_CR1

```
#define _FLASH_CR1 __SFR(uint8_t, FLASH_AddressBase+0x00)
```

Flash control register 1 (FLASH\_CR1)

Definition at line 587 of file STM8AF\_STM8S.h.

#### 5.1.2.507 \_FLASH\_CR1\_RESET\_VALUE

```
#define _FLASH_CR1_RESET_VALUE ((uint8_t) 0x00)
```

Flash control register 1 reset value.

Definition at line 599 of file STM8AF\_STM8S.h.

#### 5.1.2.508 \_FLASH\_CR2

```
#define _FLASH_CR2 _SFR(uint8_t, FLASH_AddressBase+0x01)
```

Flash control register 2 (FLASH\_CR2)

Definition at line 588 of file STM8AF\_STM8S.h.

#### 5.1.2.509 \_FLASH\_CR2\_RESET\_VALUE

```
#define _FLASH_CR2_RESET_VALUE ((uint8_t) 0x00)
```

Flash control register 2 reset value.

Definition at line 600 of file STM8AF\_STM8S.h.

#### 5.1.2.510 \_FLASH\_DUKR

```
#define _FLASH_DUKR _SFR(uint8_t, FLASH_AddressBase+0x0A)
```

Data EEPROM unprotection key register (FLASH\_DUKR)

Definition at line 596 of file STM8AF\_STM8S.h.

#### 5.1.2.511 \_FLASH\_DUKR\_RESET\_VALUE

```
#define _FLASH_DUKR_RESET_VALUE ((uint8_t) 0x00)
```

Data EEPROM unprotection key reset value.

Definition at line 604 of file STM8AF\_STM8S.h.

#### 5.1.2.512 \_FLASH\_DUL

```
#define _FLASH_DUL ((uint8_t) (0x01 << 3))
```

Data EEPROM area unlocked flag [0] (in \_FLASH\_IAPSR)

Definition at line 635 of file STM8AF\_STM8S.h.

#### 5.1.2.513 \_FLASH\_EOP

```
#define _FLASH_EOP ((uint8_t) (0x01 << 2))
```

End of programming (write or erase operation) flag [0] (in \_FLASH\_IAPSR)

Definition at line 634 of file STM8AF\_STM8S.h.

#### 5.1.2.514 \_FLASH\_ERASE

```
#define _FLASH_ERASE ((uint8_t) (0x01 << 5))
```

Block erasing [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)

Definition at line 617 of file STM8AF\_STM8S.h.

#### 5.1.2.515 \_FLASH\_FIX

```
#define _FLASH_FIX ((uint8_t) (0x01 << 0))
```

Fixed Byte programming time [0] (in \_FLASH\_CR1)

Definition at line 607 of file STM8AF\_STM8S.h.

#### 5.1.2.516 \_FLASH\_FPR

```
#define _FLASH_FPR _SFR(uint8_t, FLASH_AddressBase+0x03)
```

Flash protection register (FLASH\_FPR)

Definition at line 590 of file STM8AF\_STM8S.h.

#### 5.1.2.517 \_FLASH\_FPRG

```
#define _FLASH_FPRG ((uint8_t) (0x01 << 4))
```

Fast block programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)

Definition at line 616 of file STM8AF\_STM8S.h.

**5.1.2.518 \_FLASH\_HALT**

```
#define _FLASH_HALT ((uint8_t) (0x01 << 3))
```

Power-down in Halt mode [0] (in \_FLASH\_CR1)

Definition at line 610 of file STM8AF\_STM8S.h.

**5.1.2.519 \_FLASH\_HVOFF**

```
#define _FLASH_HVOFF ((uint8_t) (0x01 << 5))
```

End of high voltage flag [0] (in \_FLASH\_IAPSR)

Definition at line 637 of file STM8AF\_STM8S.h.

**5.1.2.520 \_FLASH\_IAPSR**

```
#define _FLASH_IAPSR _SFR(uint8_t, FLASH_AddressBase+0x05)
```

Flash status register (FLASH\_IAPSR)

Definition at line 592 of file STM8AF\_STM8S.h.

**5.1.2.521 \_FLASH\_IAPSR\_RESET\_VALUE**

```
#define _FLASH_IAPSR_RESET_VALUE ((uint8_t) 0x40)
```

Flash status register reset value.

Definition at line 602 of file STM8AF\_STM8S.h.

**5.1.2.522 \_FLASH\_IE**

```
#define _FLASH_IE ((uint8_t) (0x01 << 1))
```

Flash Interrupt enable [0] (in \_FLASH\_CR1)

Definition at line 608 of file STM8AF\_STM8S.h.

#### 5.1.2.523 \_FLASH\_NCR2

```
#define _FLASH_NCR2 _SFR(uint8_t, FLASH_AddressBase+0x02)
```

complementary Flash control register 2 (FLASH\_NCR2)

Definition at line 589 of file STM8AF\_STM8S.h.

#### 5.1.2.524 \_FLASH\_NCR2\_RESET\_VALUE

```
#define _FLASH_NCR2_RESET_VALUE ((uint8_t) 0xFF)
```

complementary Flash control register 2 reset value

Definition at line 601 of file STM8AF\_STM8S.h.

#### 5.1.2.525 \_FLASH\_NFPR

```
#define _FLASH_NFPR _SFR(uint8_t, FLASH_AddressBase+0x04)
```

complementary Flash protection register (FLASH\_NFPR)

Definition at line 591 of file STM8AF\_STM8S.h.

#### 5.1.2.526 \_FLASH\_OPT

```
#define _FLASH_OPT ((uint8_t) (0x01 << 7))
```

Write option bytes [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)

Definition at line 619 of file STM8AF\_STM8S.h.

#### 5.1.2.527 \_FLASH\_PRG

```
#define _FLASH_PRG ((uint8_t) (0x01 << 0))
```

Standard block programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)

Definition at line 614 of file STM8AF\_STM8S.h.



**5.1.2.528 \_FLASH\_PUKR**

```
#define _FLASH_PUKR _SFR(uint8_t, FLASH_AddressBase+0x08)
```

Flash program memory unprotecting key register (FLASH\_PUKR)

Definition at line 594 of file STM8AF\_STM8S.h.

**5.1.2.529 \_FLASH\_PUKR\_RESET\_VALUE**

```
#define _FLASH_PUKR_RESET_VALUE ((uint8_t) 0x00)
```

Flash program memory unprotecting key reset value.

Definition at line 603 of file STM8AF\_STM8S.h.

**5.1.2.530 \_FLASH\_PUL**

```
#define _FLASH_PUL ((uint8_t) (0x01 << 1))
```

Flash Program memory unlocked flag [0] (in \_FLASH\_IAPSR)

Definition at line 633 of file STM8AF\_STM8S.h.

**5.1.2.531 \_FLASH\_WPB**

```
#define _FLASH_WPB ((uint8_t) (0x3F << 0))
```

User boot code area protection bits [5:0] (in \_FLASH\_FPR and \_FLASH\_NFPR)

Definition at line 622 of file STM8AF\_STM8S.h.

**5.1.2.532 \_FLASH\_WPB0**

```
#define _FLASH_WPB0 ((uint8_t) (0x01 << 0))
```

User boot code area protection bit [0] (in \_FLASH\_FPR and \_FLASH\_NFPR)

Definition at line 623 of file STM8AF\_STM8S.h.

#### 5.1.2.533 \_FLASH\_WPB1

```
#define _FLASH_WPB1 ((uint8_t) (0x01 << 1))
```

User boot code area protection bit [1] (in \_FLASH\_FPR and \_FLASH\_NFPR)

Definition at line 624 of file STM8AF\_STM8S.h.

#### 5.1.2.534 \_FLASH\_WPB2

```
#define _FLASH_WPB2 ((uint8_t) (0x01 << 2))
```

User boot code area protection bit [2] (in \_FLASH\_FPR and \_FLASH\_NFPR)

Definition at line 625 of file STM8AF\_STM8S.h.

#### 5.1.2.535 \_FLASH\_WPB3

```
#define _FLASH_WPB3 ((uint8_t) (0x01 << 3))
```

User boot code area protection bit [3] (in \_FLASH\_FPR and \_FLASH\_NFPR)

Definition at line 626 of file STM8AF\_STM8S.h.

#### 5.1.2.536 \_FLASH\_WPB4

```
#define _FLASH_WPB4 ((uint8_t) (0x01 << 4))
```

User boot code area protection bit [4] (in \_FLASH\_FPR and \_FLASH\_NFPR)

Definition at line 627 of file STM8AF\_STM8S.h.

#### 5.1.2.537 \_FLASH\_WPB5

```
#define _FLASH_WPB5 ((uint8_t) (0x01 << 5))
```

User boot code area protection bit [5] (in \_FLASH\_FPR and \_FLASH\_NFPR)

Definition at line 628 of file STM8AF\_STM8S.h.

**5.1.2.538 \_FLASH\_WPRG**

```
#define _FLASH_WPRG ((uint8_t) (0x01 << 6))
```

Word programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)

Definition at line 618 of file STM8AF\_STM8S.h.

**5.1.2.539 \_FLASH\_WR\_PG\_DIS**

```
#define _FLASH_WR_PG_DIS ((uint8_t) (0x01 << 0))
```

Write attempted to protected page flag [0] (in \_FLASH\_IAPSR)

Definition at line 632 of file STM8AF\_STM8S.h.

**5.1.2.540 \_GPIO\_CR1\_RESET\_VALUE**

```
#define _GPIO_CR1_RESET_VALUE ((uint8_t) 0x00)
```

port control register 1 reset value

Definition at line 481 of file STM8AF\_STM8S.h.

**5.1.2.541 \_GPIO\_CR2\_RESET\_VALUE**

```
#define _GPIO_CR2_RESET_VALUE ((uint8_t) 0x00)
```

port control register 2 reset value

Definition at line 482 of file STM8AF\_STM8S.h.

**5.1.2.542 \_GPIO\_DDR\_RESET\_VALUE**

```
#define _GPIO_DDR_RESET_VALUE ((uint8_t) 0x00)
```

port direction register reset value

Definition at line 480 of file STM8AF\_STM8S.h.

#### 5.1.2.543 \_GPIO\_ODR\_RESET\_VALUE

```
#define _GPIO_ODR_RESET_VALUE ((uint8_t) 0x00)
```

port output register reset value

Definition at line 479 of file STM8AF\_STM8S.h.

#### 5.1.2.544 \_GPIO\_PIN0

```
#define _GPIO_PIN0 ((uint8_t) (0x01 << 0))
```

port bit mask for pin 0 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 485 of file STM8AF\_STM8S.h.

#### 5.1.2.545 \_GPIO\_PIN1

```
#define _GPIO_PIN1 ((uint8_t) (0x01 << 1))
```

port bit mask for pin 1 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 486 of file STM8AF\_STM8S.h.

#### 5.1.2.546 \_GPIO\_PIN2

```
#define _GPIO_PIN2 ((uint8_t) (0x01 << 2))
```

port bit mask for pin 2 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 487 of file STM8AF\_STM8S.h.

#### 5.1.2.547 \_GPIO\_PIN3

```
#define _GPIO_PIN3 ((uint8_t) (0x01 << 3))
```

port bit mask for pin 3 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 488 of file STM8AF\_STM8S.h.

**5.1.2.548 \_GPIO\_PIN4**

```
#define _GPIO_PIN4 ((uint8_t) (0x01 << 4))
```

port bit mask for pin 4 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 489 of file STM8AF\_STM8S.h.

**5.1.2.549 \_GPIO\_PIN5**

```
#define _GPIO_PIN5 ((uint8_t) (0x01 << 5))
```

port bit mask for pin 5 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 490 of file STM8AF\_STM8S.h.

**5.1.2.550 \_GPIO\_PIN6**

```
#define _GPIO_PIN6 ((uint8_t) (0x01 << 6))
```

port bit mask for pin 6 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 491 of file STM8AF\_STM8S.h.

**5.1.2.551 \_GPIO\_PIN7**

```
#define _GPIO_PIN7 ((uint8_t) (0x01 << 7))
```

port bit mask for pin 7 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)

Definition at line 492 of file STM8AF\_STM8S.h.

**5.1.2.552 \_GPIOA**

```
#define _GPIOA _SFR(PORT_t, PORTA_AddressBase)
```

port A struct/bit access

Definition at line 390 of file STM8AF\_STM8S.h.

#### 5.1.2.553 \_GPIOA\_CR1

```
#define _GPIOA_CR1 _SFR(uint8_t, PORTA_AddressBase+0x03)
```

port A control register 1

Definition at line 394 of file STM8AF\_STM8S.h.

#### 5.1.2.554 \_GPIOA\_CR2

```
#define _GPIOA_CR2 _SFR(uint8_t, PORTA_AddressBase+0x04)
```

port A control register 2

Definition at line 395 of file STM8AF\_STM8S.h.

#### 5.1.2.555 \_GPIOA\_DDR

```
#define _GPIOA_DDR _SFR(uint8_t, PORTA_AddressBase+0x02)
```

port A direction register

Definition at line 393 of file STM8AF\_STM8S.h.

#### 5.1.2.556 \_GPIOA\_IDR

```
#define _GPIOA_IDR _SFR(uint8_t, PORTA_AddressBase+0x01)
```

port A input register

Definition at line 392 of file STM8AF\_STM8S.h.

#### 5.1.2.557 \_GPIOA\_ODR

```
#define _GPIOA_ODR _SFR(uint8_t, PORTA_AddressBase+0x00)
```

port A output register

Definition at line 391 of file STM8AF\_STM8S.h.

#### 5.1.2.558 \_GPIOB

```
#define _GPIOB _SFR(PORT_t, PORTB_AddressBase)
```

port B struct/bit access

Definition at line 400 of file STM8AF\_STM8S.h.

#### 5.1.2.559 \_GPIOB\_CR1

```
#define _GPIOB_CR1 _SFR(uint8_t, PORTB_AddressBase+0x03)
```

port B control register 1

Definition at line 404 of file STM8AF\_STM8S.h.

#### 5.1.2.560 \_GPIOB\_CR2

```
#define _GPIOB_CR2 _SFR(uint8_t, PORTB_AddressBase+0x04)
```

port B control register 2

Definition at line 405 of file STM8AF\_STM8S.h.

#### 5.1.2.561 \_GPIOB\_DDR

```
#define _GPIOB_DDR _SFR(uint8_t, PORTB_AddressBase+0x02)
```

port B direction register

Definition at line 403 of file STM8AF\_STM8S.h.

#### 5.1.2.562 \_GPIOB\_IDR

```
#define _GPIOB_IDR _SFR(uint8_t, PORTB_AddressBase+0x01)
```

port B input register

Definition at line 402 of file STM8AF\_STM8S.h.

#### 5.1.2.563 \_GPIOB\_ODR

```
#define _GPIOB_ODR _SFR(uint8_t, PORTB_AddressBase+0x00)
```

port B output register

Definition at line 401 of file STM8AF\_STM8S.h.

#### 5.1.2.564 \_GPIOC

```
#define _GPIOC _SFR(PORT_t, PORTC_AddressBase)
```

port C struct/bit access

Definition at line 410 of file STM8AF\_STM8S.h.

#### 5.1.2.565 \_GPIOC\_CR1

```
#define _GPIOC_CR1 _SFR(uint8_t, PORTC_AddressBase+0x03)
```

port C control register 1

Definition at line 414 of file STM8AF\_STM8S.h.

#### 5.1.2.566 \_GPIOC\_CR2

```
#define _GPIOC_CR2 _SFR(uint8_t, PORTC_AddressBase+0x04)
```

port C control register 2

Definition at line 415 of file STM8AF\_STM8S.h.

#### 5.1.2.567 \_GPIOC\_DDR

```
#define _GPIOC_DDR _SFR(uint8_t, PORTC_AddressBase+0x02)
```

port C direction register

Definition at line 413 of file STM8AF\_STM8S.h.



**5.1.2.568 \_GPIOC\_IDR**

```
#define _GPIOC_IDR _SFR(uint8_t, PORTC_AddressBase+0x01)
```

port C input register

Definition at line 412 of file STM8AF\_STM8S.h.

**5.1.2.569 \_GPIOC\_ODR**

```
#define _GPIOC_ODR _SFR(uint8_t, PORTC_AddressBase+0x00)
```

port C output register

Definition at line 411 of file STM8AF\_STM8S.h.

**5.1.2.570 \_GPIOD**

```
#define _GPIOD _SFR(PORT_t, PORTD_AddressBase)
```

port D struct/bit access

Definition at line 420 of file STM8AF\_STM8S.h.

**5.1.2.571 \_GPIOD\_CR1**

```
#define _GPIOD_CR1 _SFR(uint8_t, PORTD_AddressBase+0x03)
```

port D control register 1

Definition at line 424 of file STM8AF\_STM8S.h.

**5.1.2.572 \_GPIOD\_CR2**

```
#define _GPIOD_CR2 _SFR(uint8_t, PORTD_AddressBase+0x04)
```

port D control register 2

Definition at line 425 of file STM8AF\_STM8S.h.

#### 5.1.2.573 \_GPIOD\_DDR

```
#define _GPIOD_DDR _SFR(uint8_t, PORTD_AddressBase+0x02)
```

port D direction register

Definition at line 423 of file STM8AF\_STM8S.h.

#### 5.1.2.574 \_GPIOD\_IDR

```
#define _GPIOD_IDR _SFR(uint8_t, PORTD_AddressBase+0x01)
```

port D input register

Definition at line 422 of file STM8AF\_STM8S.h.

#### 5.1.2.575 \_GPIOD\_ODR

```
#define _GPIOD_ODR _SFR(uint8_t, PORTD_AddressBase+0x00)
```

port D output register

Definition at line 421 of file STM8AF\_STM8S.h.

#### 5.1.2.576 \_GPIOE

```
#define _GPIOE _SFR(PORT_t, PORTE_AddressBase)
```

port E struct/bit access

Definition at line 430 of file STM8AF\_STM8S.h.

#### 5.1.2.577 \_GPIOE\_CR1

```
#define _GPIOE_CR1 _SFR(uint8_t, PORTE_AddressBase+0x03)
```

port E control register 1

Definition at line 434 of file STM8AF\_STM8S.h.

**5.1.2.578 \_GPIOE\_CR2**

```
#define _GPIOE_CR2 _SFR(uint8_t, PORTE_AddressBase+0x04)
```

port E control register 2

Definition at line 435 of file STM8AF\_STM8S.h.

**5.1.2.579 \_GPIOE\_DDR**

```
#define _GPIOE_DDR _SFR(uint8_t, PORTE_AddressBase+0x02)
```

port E direction register

Definition at line 433 of file STM8AF\_STM8S.h.

**5.1.2.580 \_GPIOE\_IDR**

```
#define _GPIOE_IDR _SFR(uint8_t, PORTE_AddressBase+0x01)
```

port E input register

Definition at line 432 of file STM8AF\_STM8S.h.

**5.1.2.581 \_GPIOE\_ODR**

```
#define _GPIOE_ODR _SFR(uint8_t, PORTE_AddressBase+0x00)
```

port E output register

Definition at line 431 of file STM8AF\_STM8S.h.

**5.1.2.582 \_GPIOF**

```
#define _GPIOF _SFR(PORT_t, PORTF_AddressBase)
```

port F struct/bit access

Definition at line 440 of file STM8AF\_STM8S.h.

#### 5.1.2.583 \_GPIOF\_CR1

```
#define _GPIOF_CR1 _SFR(uint8_t, PORTF_AddressBase+0x03)
```

port F control register 1

Definition at line 444 of file STM8AF\_STM8S.h.

#### 5.1.2.584 \_GPIOF\_CR2

```
#define _GPIOF_CR2 _SFR(uint8_t, PORTF_AddressBase+0x04)
```

port F control register 2

Definition at line 445 of file STM8AF\_STM8S.h.

#### 5.1.2.585 \_GPIOF\_DDR

```
#define _GPIOF_DDR _SFR(uint8_t, PORTF_AddressBase+0x02)
```

port F direction register

Definition at line 443 of file STM8AF\_STM8S.h.

#### 5.1.2.586 \_GPIOF\_IDR

```
#define _GPIOF_IDR _SFR(uint8_t, PORTF_AddressBase+0x01)
```

port F input register

Definition at line 442 of file STM8AF\_STM8S.h.

#### 5.1.2.587 \_GPIOF\_ODR

```
#define _GPIOF_ODR _SFR(uint8_t, PORTF_AddressBase+0x00)
```

port F output register

Definition at line 441 of file STM8AF\_STM8S.h.

**5.1.2.588 \_GPIOG**

```
#define _GPIOG _SFR(PORT_t, PORTG_AddressBase)
```

port G struct/bit access

Definition at line 450 of file STM8AF\_STM8S.h.

**5.1.2.589 \_GPIOG\_CR1**

```
#define _GPIOG_CR1 _SFR(uint8_t, PORTG_AddressBase+0x03)
```

port G control register 1

Definition at line 454 of file STM8AF\_STM8S.h.

**5.1.2.590 \_GPIOG\_CR2**

```
#define _GPIOG_CR2 _SFR(uint8_t, PORTG_AddressBase+0x04)
```

port G control register 2

Definition at line 455 of file STM8AF\_STM8S.h.

**5.1.2.591 \_GPIOG\_DDR**

```
#define _GPIOG_DDR _SFR(uint8_t, PORTG_AddressBase+0x02)
```

port G direction register

Definition at line 453 of file STM8AF\_STM8S.h.

**5.1.2.592 \_GPIOG\_IDR**

```
#define _GPIOG_IDR _SFR(uint8_t, PORTG_AddressBase+0x01)
```

port G input register

Definition at line 452 of file STM8AF\_STM8S.h.

#### 5.1.2.593 \_GPIOG\_ODR

```
#define _GPIOG_ODR _SFR(uint8_t, PORTG_AddressBase+0x00)
```

port G output register

Definition at line 451 of file STM8AF\_STM8S.h.

#### 5.1.2.594 \_GPIOH

```
#define _GPIOH _SFR(PORT_t, PORTH_AddressBase)
```

port H struct/bit access

Definition at line 460 of file STM8AF\_STM8S.h.

#### 5.1.2.595 \_GPIOH\_CR1

```
#define _GPIOH_CR1 _SFR(uint8_t, PORTH_AddressBase+0x03)
```

port H control register 1

Definition at line 464 of file STM8AF\_STM8S.h.

#### 5.1.2.596 \_GPIOH\_CR2

```
#define _GPIOH_CR2 _SFR(uint8_t, PORTH_AddressBase+0x04)
```

port H control register 2

Definition at line 465 of file STM8AF\_STM8S.h.

#### 5.1.2.597 \_GPIOH\_DDR

```
#define _GPIOH_DDR _SFR(uint8_t, PORTH_AddressBase+0x02)
```

port H direction register

Definition at line 463 of file STM8AF\_STM8S.h.

**5.1.2.598 \_GPIOH\_IDR**

```
#define _GPIOH_IDR _SFR(uint8_t, PORTH_AddressBase+0x01)
```

port H input register

Definition at line 462 of file STM8AF\_STM8S.h.

**5.1.2.599 \_GPIOH\_ODR**

```
#define _GPIOH_ODR _SFR(uint8_t, PORTH_AddressBase+0x00)
```

port H output register

Definition at line 461 of file STM8AF\_STM8S.h.

**5.1.2.600 \_GPIOI**

```
#define _GPIOI _SFR(PORT_t, PORTI_AddressBase)
```

port I struct/bit access

Definition at line 470 of file STM8AF\_STM8S.h.

**5.1.2.601 \_GPIOI\_CR1**

```
#define _GPIOI_CR1 _SFR(uint8_t, PORTI_AddressBase+0x03)
```

port I control register 1

Definition at line 474 of file STM8AF\_STM8S.h.

**5.1.2.602 \_GPIOI\_CR2**

```
#define _GPIOI_CR2 _SFR(uint8_t, PORTI_AddressBase+0x04)
```

port I control register 2

Definition at line 475 of file STM8AF\_STM8S.h.

#### 5.1.2.603 \_GPIOI\_DDR

```
#define _GPIOI_DDR _SFR(uint8_t, PORTI_AddressBase+0x02)
```

port I direction register

Definition at line 473 of file STM8AF\_STM8S.h.

#### 5.1.2.604 \_GPIOI\_IDR

```
#define _GPIOI_IDR _SFR(uint8_t, PORTI_AddressBase+0x01)
```

port I input register

Definition at line 472 of file STM8AF\_STM8S.h.

#### 5.1.2.605 \_GPIOI\_ODR

```
#define _GPIOI_ODR _SFR(uint8_t, PORTI_AddressBase+0x00)
```

port I output register

Definition at line 471 of file STM8AF\_STM8S.h.

#### 5.1.2.606 \_I2C

```
#define _I2C _SFR(I2C_t, I2C_AddressBase)
```

register for SPI control

I2C struct/bit access

Definition at line 1485 of file STM8AF\_STM8S.h.

#### 5.1.2.607 \_I2C\_ACK

```
#define _I2C_ACK ((uint8_t) (0x01 << 2))
```

I2C Acknowledge enable [0] (in \_I2C\_CR2)

Definition at line 1526 of file STM8AF\_STM8S.h.



**5.1.2.608 \_I2C\_ADD0**

```
#define _I2C_ADD0 ((uint8_t) (0x01 << 0))
```

I2C Interface address [0] (in 10-bit address mode) (in \_I2C\_OARL)

Definition at line 1542 of file STM8AF\_STM8S.h.

**5.1.2.609 \_I2C\_ADD1**

```
#define _I2C_ADD1 ((uint8_t) (0x01 << 1))
```

I2C Interface address [1] (in \_I2C\_OARL)

Definition at line 1543 of file STM8AF\_STM8S.h.

**5.1.2.610 \_I2C\_ADD10**

```
#define _I2C_ADD10 ((uint8_t) (0x01 << 3))
```

I2C 10-bit header sent (Master mode) [0] (in \_I2C\_SR1)

Definition at line 1564 of file STM8AF\_STM8S.h.

**5.1.2.611 \_I2C\_ADD2**

```
#define _I2C_ADD2 ((uint8_t) (0x01 << 2))
```

I2C Interface address [2] (in \_I2C\_OARL)

Definition at line 1544 of file STM8AF\_STM8S.h.

**5.1.2.612 \_I2C\_ADD3**

```
#define _I2C_ADD3 ((uint8_t) (0x01 << 3))
```

I2C Interface address [3] (in \_I2C\_OARL)

Definition at line 1545 of file STM8AF\_STM8S.h.

**5.1.2.613   \_I2C\_ADD4**

```
#define _I2C_ADD4 ((uint8_t) (0x01 << 4))
```

I2C Interface address [4] (in \_I2C\_OARL)

Definition at line 1546 of file STM8AF\_STM8S.h.

**5.1.2.614   \_I2C\_ADD5**

```
#define _I2C_ADD5 ((uint8_t) (0x01 << 5))
```

I2C Interface address [5] (in \_I2C\_OARL)

Definition at line 1547 of file STM8AF\_STM8S.h.

**5.1.2.615   \_I2C\_ADD6**

```
#define _I2C_ADD6 ((uint8_t) (0x01 << 6))
```

I2C Interface address [6] (in \_I2C\_OARL)

Definition at line 1548 of file STM8AF\_STM8S.h.

**5.1.2.616   \_I2C\_ADD7**

```
#define _I2C_ADD7 ((uint8_t) (0x01 << 7))
```

I2C Interface address [7] (in \_I2C\_OARL)

Definition at line 1549 of file STM8AF\_STM8S.h.

**5.1.2.617   \_I2C\_ADD8**

```
#define _I2C_ADD8 ((uint8_t) (0x01 << 1))
```

I2C Interface address [8] (in \_I2C\_OARH)

Definition at line 1554 of file STM8AF\_STM8S.h.

**5.1.2.618 \_I2C\_ADD9**

```
#define _I2C_ADD9 ((uint8_t) (0x01 << 2))
```

I2C Interface address [9] (in \_I2C\_OARH)

Definition at line 1555 of file STM8AF\_STM8S.h.

**5.1.2.619 \_I2C\_ADD\_8\_9**

```
#define _I2C_ADD_8_9 ((uint8_t) (0x03 << 1))
```

I2C Interface address [9:8] (in 10-bit address mode) (in \_I2C\_OARH)

Definition at line 1553 of file STM8AF\_STM8S.h.

**5.1.2.620 \_I2C\_ADDCONF**

```
#define _I2C_ADDCONF ((uint8_t) (0x01 << 6))
```

I2C Address mode configuration [0] (in \_I2C\_OARH)

Definition at line 1557 of file STM8AF\_STM8S.h.

**5.1.2.621 \_I2C\_ADDMODE**

```
#define _I2C_ADDMODE ((uint8_t) (0x01 << 7))
```

I2C 7-/10-bit addressing mode (Slave mode) [0] (in \_I2C\_OARH)

Definition at line 1558 of file STM8AF\_STM8S.h.

**5.1.2.622 \_I2C\_ADDR**

```
#define _I2C_ADDR ((uint8_t) (0x01 << 1))
```

I2C Address sent (master mode) / matched (slave mode) [0] (in \_I2C\_SR1)

Definition at line 1562 of file STM8AF\_STM8S.h.

#### 5.1.2.623 `_I2C_AF`

```
#define _I2C_AF ((uint8_t) (0x01 << 2))
```

I2C Acknowledge failure [0] (in `_I2C_SR2`)

Definition at line 1573 of file STM8AF\_STM8S.h.

#### 5.1.2.624 `_I2C_ARLO`

```
#define _I2C_ARLO ((uint8_t) (0x01 << 1))
```

I2C Arbitration lost (master mode) [0] (in `_I2C_SR2`)

Definition at line 1572 of file STM8AF\_STM8S.h.

#### 5.1.2.625 `_I2C_BERR`

```
#define _I2C_BERR ((uint8_t) (0x01 << 0))
```

I2C Bus error [0] (in `_I2C_SR2`)

Definition at line 1571 of file STM8AF\_STM8S.h.

#### 5.1.2.626 `_I2C_BTF`

```
#define _I2C_BTF ((uint8_t) (0x01 << 2))
```

I2C Byte transfer finished [0] (in `_I2C_SR1`)

Definition at line 1563 of file STM8AF\_STM8S.h.

#### 5.1.2.627 `_I2C_BUSY`

```
#define _I2C_BUSY ((uint8_t) (0x01 << 1))
```

I2C Bus busy [0] (in `_I2C_SR3`)

Definition at line 1581 of file STM8AF\_STM8S.h.

**5.1.2.628 \_I2C\_CCR**

```
#define _I2C_CCR ((uint8_t) (0x0F << 0))
```

I2C Clock control register (Master mode) [3:0] (in \_I2C\_CCRH)

Definition at line 1594 of file STM8AF\_STM8S.h.

**5.1.2.629 \_I2C\_CCR0**

```
#define _I2C_CCR0 ((uint8_t) (0x01 << 0))
```

I2C Clock control register (Master mode) [0] (in \_I2C\_CCRH)

Definition at line 1595 of file STM8AF\_STM8S.h.

**5.1.2.630 \_I2C\_CCR1**

```
#define _I2C_CCR1 ((uint8_t) (0x01 << 1))
```

I2C Clock control register (Master mode) [1] (in \_I2C\_CCRH)

Definition at line 1596 of file STM8AF\_STM8S.h.

**5.1.2.631 \_I2C\_CCR2**

```
#define _I2C_CCR2 ((uint8_t) (0x01 << 2))
```

I2C Clock control register (Master mode) [2] (in \_I2C\_CCRH)

Definition at line 1597 of file STM8AF\_STM8S.h.

**5.1.2.632 \_I2C\_CCR3**

```
#define _I2C_CCR3 ((uint8_t) (0x01 << 3))
```

I2C Clock control register (Master mode) [3] (in \_I2C\_CCRH)

Definition at line 1598 of file STM8AF\_STM8S.h.

#### 5.1.2.633 `_I2C_CCRH`

```
#define _I2C_CCRH _SFR(uint8_t, I2C_AddressBase+0x0C)
```

I2C Clock control register high byte.

Definition at line 1498 of file STM8AF\_STM8S.h.

#### 5.1.2.634 `_I2C_CCRH_RESET_VALUE`

```
#define _I2C_CCRH_RESET_VALUE ((uint8_t) 0x00)
```

I2C Clock control register high byte reset value.

Definition at line 1514 of file STM8AF\_STM8S.h.

#### 5.1.2.635 `_I2C_CCRL`

```
#define _I2C_CCRL _SFR(uint8_t, I2C_AddressBase+0x0B)
```

I2C Clock control register low byte.

Definition at line 1497 of file STM8AF\_STM8S.h.

#### 5.1.2.636 `_I2C_CCRL_RESET_VALUE`

```
#define _I2C_CCRL_RESET_VALUE ((uint8_t) 0x00)
```

I2C Clock control register low byte reset value.

Definition at line 1513 of file STM8AF\_STM8S.h.

#### 5.1.2.637 `_I2C_CR1`

```
#define _I2C_CR1 _SFR(uint8_t, I2C_AddressBase+0x00)
```

I2C Control register 1.

Definition at line 1486 of file STM8AF\_STM8S.h.

**5.1.2.638 \_I2C\_CR1\_RESET\_VALUE**

```
#define _I2C_CR1_RESET_VALUE ((uint8_t) 0x00)
```

I2C Control register 1 reset value.

Definition at line 1503 of file STM8AF\_STM8S.h.

**5.1.2.639 \_I2C\_CR2**

```
#define _I2C_CR2 _SFR(uint8_t, I2C_AddressBase+0x01)
```

I2C Control register 2.

Definition at line 1487 of file STM8AF\_STM8S.h.

**5.1.2.640 \_I2C\_CR2\_RESET\_VALUE**

```
#define _I2C_CR2_RESET_VALUE ((uint8_t) 0x00)
```

I2C Control register 2 reset value.

Definition at line 1504 of file STM8AF\_STM8S.h.

**5.1.2.641 \_I2C\_DR**

```
#define _I2C_DR _SFR(uint8_t, I2C_AddressBase+0x06)
```

I2C data register.

Definition at line 1492 of file STM8AF\_STM8S.h.

**5.1.2.642 \_I2C\_DR\_RESET\_VALUE**

```
#define _I2C_DR_RESET_VALUE ((uint8_t) 0x00)
```

I2C data register reset value.

Definition at line 1508 of file STM8AF\_STM8S.h.

#### 5.1.2.643 `_I2C_DUTY`

```
#define _I2C_DUTY ((uint8_t) (0x01 << 6))
```

I2C Fast mode duty cycle [0] (in `_I2C_CCRH`)

Definition at line 1600 of file `STM8AF_STM8S.h`.

#### 5.1.2.644 `_I2C_ENGC`

```
#define _I2C_ENGC ((uint8_t) (0x01 << 6))
```

I2C General call enable [0] (in `_I2C_CR1`)

Definition at line 1520 of file `STM8AF_STM8S.h`.

#### 5.1.2.645 `_I2C_FREQ`

```
#define _I2C_FREQ ((uint8_t) (0x3F << 0))
```

I2C Peripheral clock frequency [5:0] (in `_I2C_FREQR`)

Definition at line 1532 of file `STM8AF_STM8S.h`.

#### 5.1.2.646 `_I2C_FREQ0`

```
#define _I2C_FREQ0 ((uint8_t) (0x01 << 0))
```

I2C Peripheral clock frequency [0] (in `_I2C_FREQR`)

Definition at line 1533 of file `STM8AF_STM8S.h`.

#### 5.1.2.647 `_I2C_FREQ1`

```
#define _I2C_FREQ1 ((uint8_t) (0x01 << 1))
```

I2C Peripheral clock frequency [1] (in `_I2C_FREQR`)

Definition at line 1534 of file `STM8AF_STM8S.h`.



**5.1.2.648 \_I2C\_FREQ2**

```
#define _I2C_FREQ2 ((uint8_t) (0x01 << 2))
```

I2C Peripheral clock frequency [2] (in \_I2C\_FREQR)

Definition at line 1535 of file STM8AF\_STM8S.h.

**5.1.2.649 \_I2C\_FREQ3**

```
#define _I2C_FREQ3 ((uint8_t) (0x01 << 3))
```

I2C Peripheral clock frequency [3] (in \_I2C\_FREQR)

Definition at line 1536 of file STM8AF\_STM8S.h.

**5.1.2.650 \_I2C\_FREQ4**

```
#define _I2C_FREQ4 ((uint8_t) (0x01 << 4))
```

I2C Peripheral clock frequency [4] (in \_I2C\_FREQR)

Definition at line 1537 of file STM8AF\_STM8S.h.

**5.1.2.651 \_I2C\_FREQ5**

```
#define _I2C_FREQ5 ((uint8_t) (0x01 << 5))
```

I2C Peripheral clock frequency [5] (in \_I2C\_FREQR)

Definition at line 1538 of file STM8AF\_STM8S.h.

**5.1.2.652 \_I2C\_FREQR**

```
#define _I2C_FREQR _SFR(uint8_t, I2C_AddressBase+0x02)
```

I2C Frequency register.

Definition at line 1488 of file STM8AF\_STM8S.h.

#### 5.1.2.653 `_I2C_FREQR_RESET_VALUE`

```
#define _I2C_FREQR_RESET_VALUE ((uint8_t) 0x00)
```

I2C Frequency register reset value.

Definition at line 1505 of file STM8AF\_STM8S.h.

#### 5.1.2.654 `_I2C_FS`

```
#define _I2C_FS ((uint8_t) (0x01 << 7))
```

I2C master mode selection [0] (in `_I2C_CCRH`)

Definition at line 1601 of file STM8AF\_STM8S.h.

#### 5.1.2.655 `_I2C_GENCALL`

```
#define _I2C_GENCALL ((uint8_t) (0x01 << 4))
```

I2C General call header (Slavemode) [0] (in `_I2C_SR3`)

Definition at line 1584 of file STM8AF\_STM8S.h.

#### 5.1.2.656 `_I2C_ITBUFEN`

```
#define _I2C_ITBUFEN ((uint8_t) (0x01 << 2))
```

I2C Buffer interrupt enable [0] (in `_I2C_ITR`)

Definition at line 1590 of file STM8AF\_STM8S.h.

#### 5.1.2.657 `_I2C_ITERREN`

```
#define _I2C_ITERREN ((uint8_t) (0x01 << 0))
```

I2C Error interrupt enable [0] (in `_I2C_ITR`)

Definition at line 1588 of file STM8AF\_STM8S.h.

**5.1.2.658 \_I2C\_ITEVTEN**

```
#define _I2C_ITEVTEN ((uint8_t) (0x01 << 1))
```

I2C Event interrupt enable [0] (in \_I2C\_ITR)

Definition at line 1589 of file STM8AF\_STM8S.h.

**5.1.2.659 \_I2C\_ITR**

```
#define _I2C_ITR _SFR(uint8_t, I2C_AddressBase+0x0A)
```

I2C Interrupt register.

Definition at line 1496 of file STM8AF\_STM8S.h.

**5.1.2.660 \_I2C\_ITR\_RESET\_VALUE**

```
#define _I2C_ITR_RESET_VALUE ((uint8_t) 0x00)
```

I2C Interrupt register reset value.

Definition at line 1512 of file STM8AF\_STM8S.h.

**5.1.2.661 \_I2C\_MSL**

```
#define _I2C_MSL ((uint8_t) (0x01 << 0))
```

I2C Master/Slave [0] (in \_I2C\_SR3)

Definition at line 1580 of file STM8AF\_STM8S.h.

**5.1.2.662 \_I2C\_NOSTRETCH**

```
#define _I2C_NOSTRETCH ((uint8_t) (0x01 << 7))
```

I2C Clock stretching disable (Slave mode) [0] (in \_I2C\_CR1)

Definition at line 1521 of file STM8AF\_STM8S.h.

#### 5.1.2.663 `_I2C_OARH`

```
#define _I2C_OARH _SFR(uint8_t, I2C_AddressBase+0x04)
```

I2C own address register high byte.

Definition at line 1490 of file STM8AF\_STM8S.h.

#### 5.1.2.664 `_I2C_OARH_RESET_VALUE`

```
#define _I2C_OARH_RESET_VALUE ((uint8_t) 0x00)
```

I2C own address register high byte reset value.

Definition at line 1507 of file STM8AF\_STM8S.h.

#### 5.1.2.665 `_I2C_OARL`

```
#define _I2C_OARL _SFR(uint8_t, I2C_AddressBase+0x03)
```

I2C own address register low byte.

Definition at line 1489 of file STM8AF\_STM8S.h.

#### 5.1.2.666 `_I2C_OARL_RESET_VALUE`

```
#define _I2C_OARL_RESET_VALUE ((uint8_t) 0x00)
```

I2C own address register low byte reset value.

Definition at line 1506 of file STM8AF\_STM8S.h.

#### 5.1.2.667 `_I2C_OVR`

```
#define _I2C_OVR ((uint8_t) (0x01 << 3))
```

I2C Overrun/underrun [0] (in `_I2C_SR2`)

Definition at line 1574 of file STM8AF\_STM8S.h.

**5.1.2.668 \_I2C\_PE**

```
#define _I2C_PE ((uint8_t) (0x01 << 0))
```

I2C Peripheral enable [0] (in \_I2C\_CR1)

Definition at line 1518 of file STM8AF\_STM8S.h.

**5.1.2.669 \_I2C\_POS**

```
#define _I2C_POS ((uint8_t) (0x01 << 3))
```

I2C Acknowledge position (for data reception) [0] (in \_I2C\_CR2)

Definition at line 1527 of file STM8AF\_STM8S.h.

**5.1.2.670 \_I2C\_RXNE**

```
#define _I2C_RXNE ((uint8_t) (0x01 << 6))
```

I2C Data register not empty (receivers) [0] (in \_I2C\_SR1)

Definition at line 1567 of file STM8AF\_STM8S.h.

**5.1.2.671 \_I2C\_SB**

```
#define _I2C_SB ((uint8_t) (0x01 << 0))
```

I2C Start bit (Mastermode) [0] (in \_I2C\_SR1)

Definition at line 1561 of file STM8AF\_STM8S.h.

**5.1.2.672 \_I2C\_SR1**

```
#define _I2C_SR1 _SFR(uint8_t, I2C_AddressBase+0x07)
```

I2C Status register 1.

Definition at line 1493 of file STM8AF\_STM8S.h.

**5.1.2.673 \_I2C\_SR1\_RESET\_VALUE**

```
#define _I2C_SR1_RESET_VALUE ((uint8_t) 0x00)
```

I2C Status register 1 reset value.

Definition at line 1509 of file STM8AF\_STM8S.h.

**5.1.2.674 \_I2C\_SR2**

```
#define _I2C_SR2 _SFR(uint8_t, I2C_AddressBase+0x08)
```

I2C Status register 2.

Definition at line 1494 of file STM8AF\_STM8S.h.

**5.1.2.675 \_I2C\_SR2\_RESET\_VALUE**

```
#define _I2C_SR2_RESET_VALUE ((uint8_t) 0x00)
```

I2C Status register 2 reset value.

Definition at line 1510 of file STM8AF\_STM8S.h.

**5.1.2.676 \_I2C\_SR3**

```
#define _I2C_SR3 _SFR(uint8_t, I2C_AddressBase+0x09)
```

I2C Status register 3.

Definition at line 1495 of file STM8AF\_STM8S.h.

**5.1.2.677 \_I2C\_SR3\_RESET\_VALUE**

```
#define _I2C_SR3_RESET_VALUE ((uint8_t) 0x00)
```

I2C Status register 3 reset value.

Definition at line 1511 of file STM8AF\_STM8S.h.

**5.1.2.678 \_I2C\_START**

```
#define _I2C_START ((uint8_t) (0x01 << 0))
```

I2C Start generation [0] (in \_I2C\_CR2)

Definition at line 1524 of file STM8AF\_STM8S.h.

**5.1.2.679 \_I2C\_STOP**

```
#define _I2C_STOP ((uint8_t) (0x01 << 1))
```

I2C Stop generation [0] (in \_I2C\_CR2)

Definition at line 1525 of file STM8AF\_STM8S.h.

**5.1.2.680 \_I2C\_STOPF**

```
#define _I2C_STOPF ((uint8_t) (0x01 << 4))
```

I2C Stop detection (Slave mode) [0] (in \_I2C\_SR1)

Definition at line 1565 of file STM8AF\_STM8S.h.

**5.1.2.681 \_I2C\_SWRST**

```
#define _I2C_SWRST ((uint8_t) (0x01 << 7))
```

I2C Software reset [0] (in \_I2C\_CR2)

Definition at line 1529 of file STM8AF\_STM8S.h.

**5.1.2.682 \_I2C\_TRA**

```
#define _I2C_TRA ((uint8_t) (0x01 << 2))
```

I2C Transmitter/Receiver [0] (in \_I2C\_SR3)

Definition at line 1582 of file STM8AF\_STM8S.h.

**5.1.2.683 \_I2C\_TRISE**

```
#define _I2C_TRISE ((uint8_t) (0x3F << 0))
```

I2C Maximum rise time (Master mode) [5:0] (in \_I2C\_TRISER)

Definition at line 1604 of file STM8AF\_STM8S.h.

**5.1.2.684 \_I2C\_TRISE0**

```
#define _I2C_TRISE0 ((uint8_t) (0x01 << 0))
```

I2C Maximum rise time (Master mode) [0] (in \_I2C\_TRISER)

Definition at line 1605 of file STM8AF\_STM8S.h.

**5.1.2.685 \_I2C\_TRISE1**

```
#define _I2C_TRISE1 ((uint8_t) (0x01 << 1))
```

I2C Maximum rise time (Master mode) [1] (in \_I2C\_TRISER)

Definition at line 1606 of file STM8AF\_STM8S.h.

**5.1.2.686 \_I2C\_TRISE2**

```
#define _I2C_TRISE2 ((uint8_t) (0x01 << 2))
```

I2C Maximum rise time (Master mode) [2] (in \_I2C\_TRISER)

Definition at line 1607 of file STM8AF\_STM8S.h.

**5.1.2.687 \_I2C\_TRISE3**

```
#define _I2C_TRISE3 ((uint8_t) (0x01 << 3))
```

I2C Maximum rise time (Master mode) [3] (in \_I2C\_TRISER)

Definition at line 1608 of file STM8AF\_STM8S.h.



**5.1.2.688 \_I2C\_TRISE4**

```
#define _I2C_TRISE4 ((uint8_t) (0x01 << 4))
```

I2C Maximum rise time (Master mode) [4] (in \_I2C\_TRISER)

Definition at line 1609 of file STM8AF\_STM8S.h.

**5.1.2.689 \_I2C\_TRISE5**

```
#define _I2C_TRISE5 ((uint8_t) (0x01 << 5))
```

I2C Maximum rise time (Master mode) [5] (in \_I2C\_TRISER)

Definition at line 1610 of file STM8AF\_STM8S.h.

**5.1.2.690 \_I2C\_TRISER**

```
#define _I2C_TRISER _SFR(uint8_t, I2C_AddressBase+0x0D)
```

I2C rise time register.

Definition at line 1499 of file STM8AF\_STM8S.h.

**5.1.2.691 \_I2C\_TRISER\_RESET\_VALUE**

```
#define _I2C_TRISER_RESET_VALUE ((uint8_t) 0x02)
```

I2C rise time register reset value.

Definition at line 1515 of file STM8AF\_STM8S.h.

**5.1.2.692 \_I2C\_TXE**

```
#define _I2C_TXE ((uint8_t) (0x01 << 7))
```

I2C Data register empty (transmitters) [0] (in \_I2C\_SR1)

Definition at line 1568 of file STM8AF\_STM8S.h.

#### 5.1.2.693 `_I2C_WUFH`

```
#define _I2C_WUFH ((uint8_t) (0x01 << 5))
```

I2C Wakeup from Halt [0] (in `_I2C_SR2`)

Definition at line 1576 of file `STM8AF_STM8S.h`.

#### 5.1.2.694 `_ITC`

```
#define _ITC _SFR(ITC_t, ITC_AddressBase)
```

ITC struct/bit access.

Definition at line 6401 of file `STM8AF_STM8S.h`.

#### 5.1.2.695 `_ITC_SPR1`

```
#define _ITC_SPR1 _SFR(uint8_t, ITC_AddressBase+0x00)
```

Interrupt priority register 1/8.

Definition at line 6402 of file `STM8AF_STM8S.h`.

#### 5.1.2.696 `_ITC_SPR1_RESET_VALUE`

```
#define _ITC_SPR1_RESET_VALUE ((uint8_t) 0xFF)
```

Interrupt priority register 1/8 reset value.

Definition at line 6412 of file `STM8AF_STM8S.h`.

#### 5.1.2.697 `_ITC_SPR2`

```
#define _ITC_SPR2 _SFR(uint8_t, ITC_AddressBase+0x01)
```

Interrupt priority register 2/8.

Definition at line 6403 of file `STM8AF_STM8S.h`.

**5.1.2.698 \_ITC\_SPR2\_RESET\_VALUE**

```
#define _ITC_SPR2_RESET_VALUE ((uint8_t) 0xFF)
```

Interrupt priority register 2/8 reset value.

Definition at line 6413 of file STM8AF\_STM8S.h.

**5.1.2.699 \_ITC\_SPR3**

```
#define _ITC_SPR3 _SFR(uint8_t, ITC_AddressBase+0x02)
```

Interrupt priority register 3/8.

Definition at line 6404 of file STM8AF\_STM8S.h.

**5.1.2.700 \_ITC\_SPR3\_RESET\_VALUE**

```
#define _ITC_SPR3_RESET_VALUE ((uint8_t) 0xFF)
```

Interrupt priority register 3/8 reset value.

Definition at line 6414 of file STM8AF\_STM8S.h.

**5.1.2.701 \_ITC\_SPR4**

```
#define _ITC_SPR4 _SFR(uint8_t, ITC_AddressBase+0x03)
```

Interrupt priority register 4/8.

Definition at line 6405 of file STM8AF\_STM8S.h.

**5.1.2.702 \_ITC\_SPR4\_RESET\_VALUE**

```
#define _ITC_SPR4_RESET_VALUE ((uint8_t) 0xFF)
```

Interrupt priority register 4/8 reset value.

Definition at line 6415 of file STM8AF\_STM8S.h.

#### 5.1.2.703 `_ITC_SPR5`

```
#define _ITC_SPR5 _SFR(uint8_t, ITC_AddressBase+0x04)
```

Interrupt priority register 5/8.

Definition at line 6406 of file STM8AF\_STM8S.h.

#### 5.1.2.704 `_ITC_SPR5_RESET_VALUE`

```
#define _ITC_SPR5_RESET_VALUE ((uint8_t) 0xFF)
```

Interrupt priority register 5/8 reset value.

Definition at line 6416 of file STM8AF\_STM8S.h.

#### 5.1.2.705 `_ITC_SPR6`

```
#define _ITC_SPR6 _SFR(uint8_t, ITC_AddressBase+0x05)
```

Interrupt priority register 6/8.

Definition at line 6407 of file STM8AF\_STM8S.h.

#### 5.1.2.706 `_ITC_SPR6_RESET_VALUE`

```
#define _ITC_SPR6_RESET_VALUE ((uint8_t) 0xFF)
```

Interrupt priority register 6/8 reset value.

Definition at line 6417 of file STM8AF\_STM8S.h.

#### 5.1.2.707 `_ITC_SPR7`

```
#define _ITC_SPR7 _SFR(uint8_t, ITC_AddressBase+0x06)
```

Interrupt priority register 7/8.

Definition at line 6408 of file STM8AF\_STM8S.h.

**5.1.2.708 \_ITC\_SPR7\_RESET\_VALUE**

```
#define _ITC_SPR7_RESET_VALUE ((uint8_t) 0xFF)
```

Interrupt priority register 7/8 reset value.

Definition at line 6418 of file STM8AF\_STM8S.h.

**5.1.2.709 \_ITC\_SPR8**

```
#define _ITC_SPR8 _SFR(uint8_t, ITC_AddressBase+0x07)
```

Interrupt priority register 8/8.

Definition at line 6409 of file STM8AF\_STM8S.h.

**5.1.2.710 \_ITC\_SPR8\_RESET\_VALUE**

```
#define _ITC_SPR8_RESET_VALUE ((uint8_t) 0x0F)
```

Interrupt priority register 8/8 reset value.

Definition at line 6419 of file STM8AF\_STM8S.h.

**5.1.2.711 \_ITC\_VECT10SPR**

```
#define _ITC_VECT10SPR ((uint8_t) (0x03 << 4))
```

ITC interrupt priority vector 10 [1:0] (in \_ITC\_SPR3)

Definition at line 6454 of file STM8AF\_STM8S.h.

**5.1.2.712 \_ITC\_VECT10SPR0**

```
#define _ITC_VECT10SPR0 ((uint8_t) (0x01 << 4))
```

ITC interrupt priority vector 10 [0] (in \_ITC\_SPR3)

Definition at line 6455 of file STM8AF\_STM8S.h.

**5.1.2.713 \_ITC\_VECT10SPR1**

```
#define _ITC_VECT10SPR1 ((uint8_t) (0x01 << 5))
```

ITC interrupt priority vector 10 [1] (in \_ITC\_SPR3)

Definition at line 6456 of file STM8AF\_STM8S.h.

**5.1.2.714 \_ITC\_VECT11SPR**

```
#define _ITC_VECT11SPR ((uint8_t) (0x03 << 6))
```

ITC interrupt priority vector 11 [1:0] (in \_ITC\_SPR3)

Definition at line 6457 of file STM8AF\_STM8S.h.

**5.1.2.715 \_ITC\_VECT11SPR0**

```
#define _ITC_VECT11SPR0 ((uint8_t) (0x01 << 6))
```

ITC interrupt priority vector 11 [0] (in \_ITC\_SPR3)

Definition at line 6458 of file STM8AF\_STM8S.h.

**5.1.2.716 \_ITC\_VECT11SPR1**

```
#define _ITC_VECT11SPR1 ((uint8_t) (0x01 << 7))
```

ITC interrupt priority vector 11 [1] (in \_ITC\_SPR3)

Definition at line 6459 of file STM8AF\_STM8S.h.

**5.1.2.717 \_ITC\_VECT12SPR**

```
#define _ITC_VECT12SPR ((uint8_t) (0x03 << 0))
```

ITC interrupt priority vector 12 [1:0] (in \_ITC\_SPR4)

Definition at line 6462 of file STM8AF\_STM8S.h.

**5.1.2.718 \_ITC\_VECT12SPR0**

```
#define _ITC_VECT12SPR0 ((uint8_t) (0x01 << 0))
```

ITC interrupt priority vector 12 [0] (in \_ITC\_SPR4)

Definition at line 6463 of file STM8AF\_STM8S.h.

**5.1.2.719 \_ITC\_VECT12SPR1**

```
#define _ITC_VECT12SPR1 ((uint8_t) (0x01 << 1))
```

ITC interrupt priority vector 12 [1] (in \_ITC\_SPR4)

Definition at line 6464 of file STM8AF\_STM8S.h.

**5.1.2.720 \_ITC\_VECT13SPR**

```
#define _ITC_VECT13SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 13 [1:0] (in \_ITC\_SPR4)

Definition at line 6465 of file STM8AF\_STM8S.h.

**5.1.2.721 \_ITC\_VECT13SPR0**

```
#define _ITC_VECT13SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 13 [0] (in \_ITC\_SPR4)

Definition at line 6466 of file STM8AF\_STM8S.h.

**5.1.2.722 \_ITC\_VECT13SPR1**

```
#define _ITC_VECT13SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 13 [1] (in \_ITC\_SPR4)

Definition at line 6467 of file STM8AF\_STM8S.h.

**5.1.2.723 \_ITC\_VECT14SPR**

```
#define _ITC_VECT14SPR ((uint8_t) (0x03 << 4))
```

ITC interrupt priority vector 14 [1:0] (in \_ITC\_SPR4)

Definition at line 6468 of file STM8AF\_STM8S.h.

**5.1.2.724 \_ITC\_VECT14SPR0**

```
#define _ITC_VECT14SPR0 ((uint8_t) (0x01 << 4))
```

ITC interrupt priority vector 14 [0] (in \_ITC\_SPR4)

Definition at line 6469 of file STM8AF\_STM8S.h.

**5.1.2.725 \_ITC\_VECT14SPR1**

```
#define _ITC_VECT14SPR1 ((uint8_t) (0x01 << 5))
```

ITC interrupt priority vector 14 [1] (in \_ITC\_SPR4)

Definition at line 6470 of file STM8AF\_STM8S.h.

**5.1.2.726 \_ITC\_VECT15SPR**

```
#define _ITC_VECT15SPR ((uint8_t) (0x03 << 6))
```

ITC interrupt priority vector 15 [1:0] (in \_ITC\_SPR4)

Definition at line 6471 of file STM8AF\_STM8S.h.

**5.1.2.727 \_ITC\_VECT15SPR0**

```
#define _ITC_VECT15SPR0 ((uint8_t) (0x01 << 6))
```

ITC interrupt priority vector 15 [0] (in \_ITC\_SPR4)

Definition at line 6472 of file STM8AF\_STM8S.h.



**5.1.2.728 \_ITC\_VECT15SPR1**

```
#define _ITC_VECT15SPR1 ((uint8_t) (0x01 << 7))
```

ITC interrupt priority vector 15 [1] (in \_ITC\_SPR4)

Definition at line 6473 of file STM8AF\_STM8S.h.

**5.1.2.729 \_ITC\_VECT16SPR**

```
#define _ITC_VECT16SPR ((uint8_t) (0x03 << 0))
```

ITC interrupt priority vector 16 [1:0] (in \_ITC\_SPR5)

Definition at line 6476 of file STM8AF\_STM8S.h.

**5.1.2.730 \_ITC\_VECT16SPR0**

```
#define _ITC_VECT16SPR0 ((uint8_t) (0x01 << 0))
```

ITC interrupt priority vector 16 [0] (in \_ITC\_SPR5)

Definition at line 6477 of file STM8AF\_STM8S.h.

**5.1.2.731 \_ITC\_VECT16SPR1**

```
#define _ITC_VECT16SPR1 ((uint8_t) (0x01 << 1))
```

ITC interrupt priority vector 16 [1] (in \_ITC\_SPR5)

Definition at line 6478 of file STM8AF\_STM8S.h.

**5.1.2.732 \_ITC\_VECT17SPR**

```
#define _ITC_VECT17SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 17 [1:0] (in \_ITC\_SPR5)

Definition at line 6479 of file STM8AF\_STM8S.h.

**5.1.2.733 \_ITC\_VECT17SPR0**

```
#define _ITC_VECT17SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 17 [0] (in \_ITC\_SPR5)

Definition at line 6480 of file STM8AF\_STM8S.h.

**5.1.2.734 \_ITC\_VECT17SPR1**

```
#define _ITC_VECT17SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 17 [1] (in \_ITC\_SPR5)

Definition at line 6481 of file STM8AF\_STM8S.h.

**5.1.2.735 \_ITC\_VECT18SPR**

```
#define _ITC_VECT18SPR ((uint8_t) (0x03 << 4))
```

ITC interrupt priority vector 18 [1:0] (in \_ITC\_SPR5)

Definition at line 6482 of file STM8AF\_STM8S.h.

**5.1.2.736 \_ITC\_VECT18SPR0**

```
#define _ITC_VECT18SPR0 ((uint8_t) (0x01 << 4))
```

ITC interrupt priority vector 18 [0] (in \_ITC\_SPR5)

Definition at line 6483 of file STM8AF\_STM8S.h.

**5.1.2.737 \_ITC\_VECT18SPR1**

```
#define _ITC_VECT18SPR1 ((uint8_t) (0x01 << 5))
```

ITC interrupt priority vector 18 [1] (in \_ITC\_SPR5)

Definition at line 6484 of file STM8AF\_STM8S.h.

**5.1.2.738 \_ITC\_VECT19SPR**

```
#define _ITC_VECT19SPR ((uint8_t) (0x03 << 6))
```

ITC interrupt priority vector 19 [1:0] (in \_ITC\_SPR5)

Definition at line 6485 of file STM8AF\_STM8S.h.

**5.1.2.739 \_ITC\_VECT19SPR0**

```
#define _ITC_VECT19SPR0 ((uint8_t) (0x01 << 6))
```

ITC interrupt priority vector 19 [0] (in \_ITC\_SPR5)

Definition at line 6486 of file STM8AF\_STM8S.h.

**5.1.2.740 \_ITC\_VECT19SPR1**

```
#define _ITC_VECT19SPR1 ((uint8_t) (0x01 << 7))
```

ITC interrupt priority vector 19 [1] (in \_ITC\_SPR5)

Definition at line 6487 of file STM8AF\_STM8S.h.

**5.1.2.741 \_ITC\_VECT1SPR**

```
#define _ITC_VECT1SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 1 [1:0] (in \_ITC\_SPR1)

Definition at line 6423 of file STM8AF\_STM8S.h.

**5.1.2.742 \_ITC\_VECT1SPR0**

```
#define _ITC_VECT1SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 1 [0] (in \_ITC\_SPR1)

Definition at line 6424 of file STM8AF\_STM8S.h.

**5.1.2.743 \_ITC\_VECT1SPR1**

```
#define _ITC_VECT1SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 1 [1] (in \_ITC\_SPR1)

Definition at line 6425 of file STM8AF\_STM8S.h.

**5.1.2.744 \_ITC\_VECT20SPR**

```
#define _ITC_VECT20SPR ((uint8_t) (0x03 << 0))
```

ITC interrupt priority vector 20 [1:0] (in \_ITC\_SPR6)

Definition at line 6490 of file STM8AF\_STM8S.h.

**5.1.2.745 \_ITC\_VECT20SPR0**

```
#define _ITC_VECT20SPR0 ((uint8_t) (0x01 << 0))
```

ITC interrupt priority vector 20 [0] (in \_ITC\_SPR6)

Definition at line 6491 of file STM8AF\_STM8S.h.

**5.1.2.746 \_ITC\_VECT20SPR1**

```
#define _ITC_VECT20SPR1 ((uint8_t) (0x01 << 1))
```

ITC interrupt priority vector 20 [1] (in \_ITC\_SPR6)

Definition at line 6492 of file STM8AF\_STM8S.h.

**5.1.2.747 \_ITC\_VECT21SPR**

```
#define _ITC_VECT21SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 21 [1:0] (in \_ITC\_SPR6)

Definition at line 6493 of file STM8AF\_STM8S.h.

**5.1.2.748 \_ITC\_VECT21SPR0**

```
#define _ITC_VECT21SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 21 [0] (in \_ITC\_SPR6)

Definition at line 6494 of file STM8AF\_STM8S.h.

**5.1.2.749 \_ITC\_VECT21SPR1**

```
#define _ITC_VECT21SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 21 [1] (in \_ITC\_SPR6)

Definition at line 6495 of file STM8AF\_STM8S.h.

**5.1.2.750 \_ITC\_VECT22SPR**

```
#define _ITC_VECT22SPR ((uint8_t) (0x03 << 4))
```

ITC interrupt priority vector 22 [1:0] (in \_ITC\_SPR6)

Definition at line 6496 of file STM8AF\_STM8S.h.

**5.1.2.751 \_ITC\_VECT22SPR0**

```
#define _ITC_VECT22SPR0 ((uint8_t) (0x01 << 4))
```

ITC interrupt priority vector 22 [0] (in \_ITC\_SPR6)

Definition at line 6497 of file STM8AF\_STM8S.h.

**5.1.2.752 \_ITC\_VECT22SPR1**

```
#define _ITC_VECT22SPR1 ((uint8_t) (0x01 << 5))
```

ITC interrupt priority vector 22 [1] (in \_ITC\_SPR6)

Definition at line 6498 of file STM8AF\_STM8S.h.

**5.1.2.753 \_ITC\_VECT23SPR**

```
#define _ITC_VECT23SPR ((uint8_t) (0x03 << 6))
```

ITC interrupt priority vector 23 [1:0] (in \_ITC\_SPR6)

Definition at line 6499 of file STM8AF\_STM8S.h.

**5.1.2.754 \_ITC\_VECT23SPR0**

```
#define _ITC_VECT23SPR0 ((uint8_t) (0x01 << 6))
```

ITC interrupt priority vector 23 [0] (in \_ITC\_SPR6)

Definition at line 6500 of file STM8AF\_STM8S.h.

**5.1.2.755 \_ITC\_VECT23SPR1**

```
#define _ITC_VECT23SPR1 ((uint8_t) (0x01 << 7))
```

ITC interrupt priority vector 23 [1] (in \_ITC\_SPR6)

Definition at line 6501 of file STM8AF\_STM8S.h.

**5.1.2.756 \_ITC\_VECT24SPR**

```
#define _ITC_VECT24SPR ((uint8_t) (0x03 << 0))
```

ITC interrupt priority vector 24 [1:0] (in \_ITC\_SPR7)

Definition at line 6504 of file STM8AF\_STM8S.h.

**5.1.2.757 \_ITC\_VECT24SPR0**

```
#define _ITC_VECT24SPR0 ((uint8_t) (0x01 << 0))
```

ITC interrupt priority vector 24 [0] (in \_ITC\_SPR7)

Definition at line 6505 of file STM8AF\_STM8S.h.

**5.1.2.758 \_ITC\_VECT24SPR1**

```
#define _ITC_VECT24SPR1 ((uint8_t) (0x01 << 1))
```

ITC interrupt priority vector 24 [1] (in \_ITC\_SPR7)

Definition at line 6506 of file STM8AF\_STM8S.h.

**5.1.2.759 \_ITC\_VECT25SPR**

```
#define _ITC_VECT25SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 25 [1:0] (in \_ITC\_SPR7)

Definition at line 6507 of file STM8AF\_STM8S.h.

**5.1.2.760 \_ITC\_VECT25SPR0**

```
#define _ITC_VECT25SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 25 [0] (in \_ITC\_SPR7)

Definition at line 6508 of file STM8AF\_STM8S.h.

**5.1.2.761 \_ITC\_VECT25SPR1**

```
#define _ITC_VECT25SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 25 [1] (in \_ITC\_SPR7)

Definition at line 6509 of file STM8AF\_STM8S.h.

**5.1.2.762 \_ITC\_VECT26SPR**

```
#define _ITC_VECT26SPR ((uint8_t) (0x03 << 4))
```

ITC interrupt priority vector 26 [1:0] (in \_ITC\_SPR7)

Definition at line 6510 of file STM8AF\_STM8S.h.

**5.1.2.763 \_ITC\_VECT26SPR0**

```
#define _ITC_VECT26SPR0 ((uint8_t) (0x01 << 4))
```

ITC interrupt priority vector 26 [0] (in \_ITC\_SPR7)

Definition at line 6511 of file STM8AF\_STM8S.h.

**5.1.2.764 \_ITC\_VECT26SPR1**

```
#define _ITC_VECT26SPR1 ((uint8_t) (0x01 << 5))
```

ITC interrupt priority vector 26 [1] (in \_ITC\_SPR7)

Definition at line 6512 of file STM8AF\_STM8S.h.

**5.1.2.765 \_ITC\_VECT27SPR**

```
#define _ITC_VECT27SPR ((uint8_t) (0x03 << 6))
```

ITC interrupt priority vector 27 [1:0] (in \_ITC\_SPR7)

Definition at line 6513 of file STM8AF\_STM8S.h.

**5.1.2.766 \_ITC\_VECT27SPR0**

```
#define _ITC_VECT27SPR0 ((uint8_t) (0x01 << 6))
```

ITC interrupt priority vector 27 [0] (in \_ITC\_SPR7)

Definition at line 6514 of file STM8AF\_STM8S.h.

**5.1.2.767 \_ITC\_VECT27SPR1**

```
#define _ITC_VECT27SPR1 ((uint8_t) (0x01 << 7))
```

ITC interrupt priority vector 27 [1] (in \_ITC\_SPR7)

Definition at line 6515 of file STM8AF\_STM8S.h.



**5.1.2.768 \_ITC\_VECT28SPR**

```
#define _ITC_VECT28SPR ((uint8_t) (0x03 << 0))
```

ITC interrupt priority vector 28 [1:0] (in \_ITC\_SPR8)

Definition at line 6518 of file STM8AF\_STM8S.h.

**5.1.2.769 \_ITC\_VECT28SPR0**

```
#define _ITC_VECT28SPR0 ((uint8_t) (0x01 << 0))
```

ITC interrupt priority vector 28 [0] (in \_ITC\_SPR8)

Definition at line 6519 of file STM8AF\_STM8S.h.

**5.1.2.770 \_ITC\_VECT28SPR1**

```
#define _ITC_VECT28SPR1 ((uint8_t) (0x01 << 1))
```

ITC interrupt priority vector 28 [1] (in \_ITC\_SPR8)

Definition at line 6520 of file STM8AF\_STM8S.h.

**5.1.2.771 \_ITC\_VECT29SPR**

```
#define _ITC_VECT29SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 29 [1:0] (in \_ITC\_SPR8)

Definition at line 6521 of file STM8AF\_STM8S.h.

**5.1.2.772 \_ITC\_VECT29SPR0**

```
#define _ITC_VECT29SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 29 [0] (in \_ITC\_SPR8)

Definition at line 6522 of file STM8AF\_STM8S.h.

**5.1.2.773 \_ITC\_VECT29SPR1**

```
#define _ITC_VECT29SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 29 [1] (in \_ITC\_SPR8)

Definition at line 6523 of file STM8AF\_STM8S.h.

**5.1.2.774 \_ITC\_VECT2SPR**

```
#define _ITC_VECT2SPR ((uint8_t) (0x03 << 4))
```

ITC interrupt priority vector 2 [1:0] (in \_ITC\_SPR1)

Definition at line 6426 of file STM8AF\_STM8S.h.

**5.1.2.775 \_ITC\_VECT2SPR0**

```
#define _ITC_VECT2SPR0 ((uint8_t) (0x01 << 4))
```

ITC interrupt priority vector 2 [0] (in \_ITC\_SPR1)

Definition at line 6427 of file STM8AF\_STM8S.h.

**5.1.2.776 \_ITC\_VECT2SPR1**

```
#define _ITC_VECT2SPR1 ((uint8_t) (0x01 << 5))
```

ITC interrupt priority vector 2 [1] (in \_ITC\_SPR1)

Definition at line 6428 of file STM8AF\_STM8S.h.

**5.1.2.777 \_ITC\_VECT3SPR**

```
#define _ITC_VECT3SPR ((uint8_t) (0x03 << 6))
```

ITC interrupt priority vector 3 [1:0] (in \_ITC\_SPR1)

Definition at line 6429 of file STM8AF\_STM8S.h.

**5.1.2.778 \_ITC\_VECT3SPR0**

```
#define _ITC_VECT3SPR0 ((uint8_t) (0x01 << 6))
```

ITC interrupt priority vector 3 [0] (in \_ITC\_SPR1)

Definition at line 6430 of file STM8AF\_STM8S.h.

**5.1.2.779 \_ITC\_VECT3SPR1**

```
#define _ITC_VECT3SPR1 ((uint8_t) (0x01 << 7))
```

ITC interrupt priority vector 3 [1] (in \_ITC\_SPR1)

Definition at line 6431 of file STM8AF\_STM8S.h.

**5.1.2.780 \_ITC\_VECT4SPR**

```
#define _ITC_VECT4SPR ((uint8_t) (0x03 << 0))
```

ITC interrupt priority vector 4 [1:0] (in \_ITC\_SPR2)

Definition at line 6434 of file STM8AF\_STM8S.h.

**5.1.2.781 \_ITC\_VECT4SPR0**

```
#define _ITC_VECT4SPR0 ((uint8_t) (0x01 << 0))
```

ITC interrupt priority vector 4 [0] (in \_ITC\_SPR2)

Definition at line 6435 of file STM8AF\_STM8S.h.

**5.1.2.782 \_ITC\_VECT4SPR1**

```
#define _ITC_VECT4SPR1 ((uint8_t) (0x01 << 1))
```

ITC interrupt priority vector 4 [1] (in \_ITC\_SPR2)

Definition at line 6436 of file STM8AF\_STM8S.h.

**5.1.2.783 \_ITC\_VECT5SPR**

```
#define _ITC_VECT5SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 5 [1:0] (in \_ITC\_SPR2)

Definition at line 6437 of file STM8AF\_STM8S.h.

**5.1.2.784 \_ITC\_VECT5SPR0**

```
#define _ITC_VECT5SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 5 [0] (in \_ITC\_SPR2)

Definition at line 6438 of file STM8AF\_STM8S.h.

**5.1.2.785 \_ITC\_VECT5SPR1**

```
#define _ITC_VECT5SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 5 [1] (in \_ITC\_SPR2)

Definition at line 6439 of file STM8AF\_STM8S.h.

**5.1.2.786 \_ITC\_VECT6SPR**

```
#define _ITC_VECT6SPR ((uint8_t) (0x03 << 4))
```

ITC interrupt priority vector 6 [1:0] (in \_ITC\_SPR2)

Definition at line 6440 of file STM8AF\_STM8S.h.

**5.1.2.787 \_ITC\_VECT6SPR0**

```
#define _ITC_VECT6SPR0 ((uint8_t) (0x01 << 4))
```

ITC interrupt priority vector 6 [0] (in \_ITC\_SPR2)

Definition at line 6441 of file STM8AF\_STM8S.h.

**5.1.2.788 \_ITC\_VECT6SPR1**

```
#define _ITC_VECT6SPR1 ((uint8_t) (0x01 << 5))
```

ITC interrupt priority vector 6 [1] (in \_ITC\_SPR2)

Definition at line 6442 of file STM8AF\_STM8S.h.

**5.1.2.789 \_ITC\_VECT7SPR**

```
#define _ITC_VECT7SPR ((uint8_t) (0x03 << 6))
```

ITC interrupt priority vector 7 [1:0] (in \_ITC\_SPR2)

Definition at line 6443 of file STM8AF\_STM8S.h.

**5.1.2.790 \_ITC\_VECT7SPR0**

```
#define _ITC_VECT7SPR0 ((uint8_t) (0x01 << 6))
```

ITC interrupt priority vector 7 [0] (in \_ITC\_SPR2)

Definition at line 6444 of file STM8AF\_STM8S.h.

**5.1.2.791 \_ITC\_VECT7SPR1**

```
#define _ITC_VECT7SPR1 ((uint8_t) (0x01 << 7))
```

ITC interrupt priority vector 7 [1] (in \_ITC\_SPR2)

Definition at line 6445 of file STM8AF\_STM8S.h.

**5.1.2.792 \_ITC\_VECT8SPR**

```
#define _ITC_VECT8SPR ((uint8_t) (0x03 << 0))
```

ITC interrupt priority vector 8 [1:0] (in \_ITC\_SPR3)

Definition at line 6448 of file STM8AF\_STM8S.h.

**5.1.2.793 \_ITC\_VECT8SPR0**

```
#define _ITC_VECT8SPR0 ((uint8_t) (0x01 << 0))
```

ITC interrupt priority vector 8 [0] (in \_ITC\_SPR3)

Definition at line 6449 of file STM8AF\_STM8S.h.

**5.1.2.794 \_ITC\_VECT8SPR1**

```
#define _ITC_VECT8SPR1 ((uint8_t) (0x01 << 1))
```

ITC interrupt priority vector 8 [1] (in \_ITC\_SPR3)

Definition at line 6450 of file STM8AF\_STM8S.h.

**5.1.2.795 \_ITC\_VECT9SPR**

```
#define _ITC_VECT9SPR ((uint8_t) (0x03 << 2))
```

ITC interrupt priority vector 9 [1:0] (in \_ITC\_SPR3)

Definition at line 6451 of file STM8AF\_STM8S.h.

**5.1.2.796 \_ITC\_VECT9SPR0**

```
#define _ITC_VECT9SPR0 ((uint8_t) (0x01 << 2))
```

ITC interrupt priority vector 9 [0] (in \_ITC\_SPR3)

Definition at line 6452 of file STM8AF\_STM8S.h.

**5.1.2.797 \_ITC\_VECT9SPR1**

```
#define _ITC_VECT9SPR1 ((uint8_t) (0x01 << 3))
```

ITC interrupt priority vector 9 [1] (in \_ITC\_SPR3)

Definition at line 6453 of file STM8AF\_STM8S.h.

#### 5.1.2.798 \_IWDG

```
#define _IWDG _SFR(IWDG_t, IWDG_AddressBase)
```

Independent Timeout Watchdog struct/bit access.

Definition at line 1069 of file STM8AF\_STM8S.h.

#### 5.1.2.799 \_IWDG\_KEY\_ACCESS

```
#define _IWDG_KEY_ACCESS ((uint8_t) 0x55)
```

Independent Timeout Watchdog unlock write to IWDG\_PR and IWDG\_RLR (in \_IWDG\_KR)

Definition at line 1081 of file STM8AF\_STM8S.h.

#### 5.1.2.800 \_IWDG\_KEY\_ENABLE

```
#define _IWDG_KEY_ENABLE ((uint8_t) 0xCC)
```

Independent Timeout Watchdog enable (in \_IWDG\_KR)

Definition at line 1079 of file STM8AF\_STM8S.h.

#### 5.1.2.801 \_IWDG\_KEY\_REFRESH

```
#define _IWDG_KEY_REFRESH ((uint8_t) 0xAA)
```

Independent Timeout Watchdog refresh (in \_IWDG\_KR)

Definition at line 1080 of file STM8AF\_STM8S.h.

#### 5.1.2.802 \_IWDG\_KR

```
#define _IWDG_KR _SFR(uint8_t, IWDG_AddressBase+0x00)
```

Independent Timeout Watchdog Key register (IWDG\_KR)

Definition at line 1070 of file STM8AF\_STM8S.h.

#### 5.1.2.803 \_IWDG\_PR

```
#define _IWDG_PR _SFR(uint8_t, IWDG_AddressBase+0x01)
```

Independent Timeout Watchdog Prescaler register (IWDG\_PR)

Definition at line 1071 of file STM8AF\_STM8S.h.

#### 5.1.2.804 \_IWDG\_PR\_RESET\_VALUE

```
#define _IWDG_PR_RESET_VALUE ((uint8_t) 0x00)
```

Independent Timeout Watchdog Prescaler register reset value.

Definition at line 1075 of file STM8AF\_STM8S.h.

#### 5.1.2.805 \_IWDG\_PRE

```
#define _IWDG_PRE ((uint8_t) (0x07 << 0))
```

Independent Timeout Watchdog Prescaler divider [2:0] (in \_IWDG\_PR)

Definition at line 1084 of file STM8AF\_STM8S.h.

#### 5.1.2.806 \_IWDG\_PRE0

```
#define _IWDG_PRE0 ((uint8_t) (0x01 << 0))
```

Independent Timeout Watchdog Prescaler divider [0] (in \_IWDG\_PR)

Definition at line 1085 of file STM8AF\_STM8S.h.

#### 5.1.2.807 \_IWDG\_PRE1

```
#define _IWDG_PRE1 ((uint8_t) (0x01 << 1))
```

Independent Timeout Watchdog Prescaler divider [1] (in \_IWDG\_PR)

Definition at line 1086 of file STM8AF\_STM8S.h.



**5.1.2.808 \_IWDG\_PRE2**

```
#define _IWDG_PRE2 ((uint8_t) (0x01 << 2))
```

Independent Timeout Watchdog Prescaler divider [2] (in \_IWDG\_PR)

Definition at line 1087 of file STM8AF\_STM8S.h.

**5.1.2.809 \_IWDG\_RLR**

```
#define _IWDG_RLR _SFR(uint8_t, IWDG_AddressBase+0x02)
```

Independent Timeout Watchdog Reload register (IWDG\_RLR)

Definition at line 1072 of file STM8AF\_STM8S.h.

**5.1.2.810 \_IWDG\_RLR\_RESET\_VALUE**

```
#define _IWDG_RLR_RESET_VALUE ((uint8_t) 0xFF)
```

Independent Timeout Watchdog Reload register reset value.

Definition at line 1076 of file STM8AF\_STM8S.h.

**5.1.2.811 \_RST**

```
#define _RST _SFR(RST_t, RST_AddressBase)
```

Reset module struct/bit access.

Definition at line 725 of file STM8AF\_STM8S.h.

**5.1.2.812 \_RST\_EMCF**

```
#define _RST_EMCF ((uint8_t) (0x01 << 4))
```

EMC reset flag [0] (in \_RST\_SR)

Definition at line 733 of file STM8AF\_STM8S.h.

#### 5.1.2.813 `_RST_ILLOPF`

```
#define _RST_ILLOPF ((uint8_t) (0x01 << 2))
```

Illegal opcode reset flag [0] (in `_RST_SR`)

Definition at line 731 of file `STM8AF_STM8S.h`.

#### 5.1.2.814 `_RST_IWDGF`

```
#define _RST_IWDGF ((uint8_t) (0x01 << 1))
```

Independent Watchdog reset flag [0] (in `_RST_SR`)

Definition at line 730 of file `STM8AF_STM8S.h`.

#### 5.1.2.815 `_RST_SR`

```
#define _RST_SR _SFR(uint8_t, RST_AddressBase+0x00)
```

Reset module status register (`RST_SR`)

Definition at line 726 of file `STM8AF_STM8S.h`.

#### 5.1.2.816 `_RST_SWIMF`

```
#define _RST_SWIMF ((uint8_t) (0x01 << 3))
```

SWIM reset flag [0] (in `_RST_SR`)

Definition at line 732 of file `STM8AF_STM8S.h`.

#### 5.1.2.817 `_RST_WWDGF`

```
#define _RST_WWDGF ((uint8_t) (0x01 << 0))
```

Window Watchdog reset flag [0] (in `_RST_SR`)

Definition at line 729 of file `STM8AF_STM8S.h`.

**5.1.2.818 \_SFR**

```
#define _SFR(  
    type,  
    addr ) (*((volatile type*) (addr)))
```

peripheral register

Definition at line 187 of file STM8AF\_STM8S.h.

**5.1.2.819 \_SPI**

```
#define _SPI _SFR(SPI_t, SPI_AddressBase)
```

register for SPI control

SPI struct/bit access

Definition at line 1288 of file STM8AF\_STM8S.h.

**5.1.2.820 \_SPI\_BDM**

```
#define _SPI_BDM ((uint8_t) (0x01 << 7))
```

SPI Bidirectional data mode enable [0] (in \_SPI\_CR2)

Definition at line 1327 of file STM8AF\_STM8S.h.

**5.1.2.821 \_SPI\_BDOE**

```
#define _SPI_BDOE ((uint8_t) (0x01 << 6))
```

SPI Input/Output enable in bidirectional mode [0] (in \_SPI\_CR2)

Definition at line 1326 of file STM8AF\_STM8S.h.

**5.1.2.822 \_SPI\_BR**

```
#define _SPI_BR ((uint8_t) (0x07 << 3))
```

SPI Baudrate control [2:0] (in \_SPI\_CR1)

Definition at line 1312 of file STM8AF\_STM8S.h.

**5.1.2.823 \_SPI\_BR0**

```
#define _SPI_BR0 ((uint8_t) (0x01 << 3))
```

SPI Baudrate control [0] (in \_SPI\_CR1)

Definition at line 1313 of file STM8AF\_STM8S.h.

**5.1.2.824 \_SPI\_BR1**

```
#define _SPI_BR1 ((uint8_t) (0x01 << 4))
```

SPI Baudrate control [1] (in \_SPI\_CR1)

Definition at line 1314 of file STM8AF\_STM8S.h.

**5.1.2.825 \_SPI\_BR2**

```
#define _SPI_BR2 ((uint8_t) (0x01 << 5))
```

SPI Baudrate control [2] (in \_SPI\_CR1)

Definition at line 1315 of file STM8AF\_STM8S.h.

**5.1.2.826 \_SPI\_BSY**

```
#define _SPI_BSY ((uint8_t) (0x01 << 7))
```

SPI Busy flag [0] (in \_SPI\_SR)

Definition at line 1344 of file STM8AF\_STM8S.h.

**5.1.2.827 \_SPI\_CPHA**

```
#define _SPI_CPHA ((uint8_t) (0x01 << 0))
```

SPI Clock phase [0] (in \_SPI\_CR1)

Definition at line 1309 of file STM8AF\_STM8S.h.

**5.1.2.828 \_SPI\_CPOL**

```
#define _SPI_CPOL ((uint8_t) (0x01 << 1))
```

SPI Clock polarity [0] (in \_SPI\_CR1)

Definition at line 1310 of file STM8AF\_STM8S.h.

**5.1.2.829 \_SPI\_CR1**

```
#define _SPI_CR1 _SFR(uint8_t, SPI_AddressBase+0x00)
```

SPI control register 1.

Definition at line 1289 of file STM8AF\_STM8S.h.

**5.1.2.830 \_SPI\_CR1\_RESET\_VALUE**

```
#define _SPI_CR1_RESET_VALUE ((uint8_t) 0x00)
```

SPI Control Register 1 reset value.

Definition at line 1299 of file STM8AF\_STM8S.h.

**5.1.2.831 \_SPI\_CR2**

```
#define _SPI_CR2 _SFR(uint8_t, SPI_AddressBase+0x01)
```

SPI control register 2.

Definition at line 1290 of file STM8AF\_STM8S.h.

**5.1.2.832 \_SPI\_CR2\_RESET\_VALUE**

```
#define _SPI_CR2_RESET_VALUE ((uint8_t) 0x00)
```

SPI Control Register 2 reset value.

Definition at line 1300 of file STM8AF\_STM8S.h.

#### 5.1.2.833 `_SPI_CRCEN`

```
#define _SPI_CRCEN ((uint8_t) (0x01 << 5))
```

SPI Hardware CRC calculation enable [0] (in `_SPI_CR2`)

Definition at line 1325 of file `STM8AF_STM8S.h`.

#### 5.1.2.834 `_SPI_CRCERR`

```
#define _SPI_CRCERR ((uint8_t) (0x01 << 4))
```

SPI CRC error flag [0] (in `_SPI_SR`)

Definition at line 1341 of file `STM8AF_STM8S.h`.

#### 5.1.2.835 `_SPI_CRCNEXT`

```
#define _SPI_CRCNEXT ((uint8_t) (0x01 << 4))
```

SPI Transmit CRC next [0] (in `_SPI_CR2`)

Definition at line 1324 of file `STM8AF_STM8S.h`.

#### 5.1.2.836 `_SPI_CRCPR`

```
#define _SPI_CRCPR \_SFR(uint8_t, SPI\_AddressBase+0x05)
```

SPI CRC polynomial register.

Definition at line 1294 of file `STM8AF_STM8S.h`.

#### 5.1.2.837 `_SPI_CRCPR_RESET_VALUE`

```
#define _SPI_CRCPR_RESET_VALUE ((uint8_t) 0x07)
```

SPI Polynomial Register reset value.

Definition at line 1304 of file `STM8AF_STM8S.h`.

### 5.1.2.838 \_SPI\_DR

```
#define _SPI_DR _SFR(uint8_t, SPI_AddressBase+0x04)
```

SPI data register.

Definition at line 1293 of file STM8AF\_STM8S.h.

### 5.1.2.839 \_SPI\_DR\_RESET\_VALUE

```
#define _SPI_DR_RESET_VALUE ((uint8_t) 0x00)
```

SPI Data Register reset value.

Definition at line 1303 of file STM8AF\_STM8S.h.

### 5.1.2.840 \_SPI\_ERRIE

```
#define _SPI_ERRIE ((uint8_t) (0x01 << 5))
```

SPI Error interrupt enable [0] (in \_SPI\_ICR)

Definition at line 1332 of file STM8AF\_STM8S.h.

### 5.1.2.841 \_SPI\_ICR

```
#define _SPI_ICR _SFR(uint8_t, SPI_AddressBase+0x02)
```

SPI interrupt control register.

Definition at line 1291 of file STM8AF\_STM8S.h.

### 5.1.2.842 \_SPI\_ICR\_RESET\_VALUE

```
#define _SPI_ICR_RESET_VALUE ((uint8_t) 0x00)
```

SPI Interrupt Control Register reset value.

Definition at line 1301 of file STM8AF\_STM8S.h.

#### 5.1.2.843 `_SPI_LSBFIRST`

```
#define _SPI_LSBFIRST ((uint8_t) (0x01 << 7))
```

SPI Frame format [0] (in `_SPI_CR1`)

Definition at line 1317 of file `STM8AF_STM8S.h`.

#### 5.1.2.844 `_SPI_MODF`

```
#define _SPI_MODF ((uint8_t) (0x01 << 5))
```

SPI Mode fault [0] (in `_SPI_SR`)

Definition at line 1342 of file `STM8AF_STM8S.h`.

#### 5.1.2.845 `_SPI_MSTR`

```
#define _SPI_MSTR ((uint8_t) (0x01 << 2))
```

SPI Master/slave selection [0] (in `_SPI_CR1`)

Definition at line 1311 of file `STM8AF_STM8S.h`.

#### 5.1.2.846 `_SPI_OVR`

```
#define _SPI_OVR ((uint8_t) (0x01 << 6))
```

SPI Overrun flag [0] (in `_SPI_SR`)

Definition at line 1343 of file `STM8AF_STM8S.h`.

#### 5.1.2.847 `_SPI_RXCR`

```
#define _SPI_RXCR \_SFR(uint8_t, SPI\_AddressBase+0x06)
```

SPI Rx CRC register.

Definition at line 1295 of file `STM8AF_STM8S.h`.



**5.1.2.848 \_SPI\_RXCRCR\_RESET\_VALUE**

```
#define _SPI_RXCRCR_RESET_VALUE ((uint8_t) 0x00)
```

SPI RX CRC Register reset value.

Definition at line 1305 of file STM8AF\_STM8S.h.

**5.1.2.849 \_SPI\_RXIE**

```
#define _SPI_RXIE ((uint8_t) (0x01 << 6))
```

SPI Rx buffer not empty interrupt enable [0] (in \_SPI\_ICR)

Definition at line 1333 of file STM8AF\_STM8S.h.

**5.1.2.850 \_SPI\_RXNE**

```
#define _SPI_RXNE ((uint8_t) (0x01 << 0))
```

SPI Receive buffer not empty [0] (in \_SPI\_SR)

Definition at line 1337 of file STM8AF\_STM8S.h.

**5.1.2.851 \_SPI\_RXONLY**

```
#define _SPI_RXONLY ((uint8_t) (0x01 << 2))
```

SPI Receive only [0] (in \_SPI\_CR2)

Definition at line 1322 of file STM8AF\_STM8S.h.

**5.1.2.852 \_SPI\_SPE**

```
#define _SPI_SPE ((uint8_t) (0x01 << 6))
```

SPI enable [0] (in \_SPI\_CR1)

Definition at line 1316 of file STM8AF\_STM8S.h.

#### 5.1.2.853 `_SPI_SR`

```
#define _SPI_SR \_SFR(uint8_t, SPI\_AddressBase+0x03)
```

SPI status register.

Definition at line 1292 of file STM8AF\_STM8S.h.

#### 5.1.2.854 `_SPI_SR_RESET_VALUE`

```
#define _SPI_SR_RESET_VALUE ((uint8_t) 0x02)
```

SPI Status Register reset value.

Definition at line 1302 of file STM8AF\_STM8S.h.

#### 5.1.2.855 `_SPI_SSI`

```
#define _SPI_SSI ((uint8_t) (0x01 << 0))
```

SPI Internal slave select [0] (in `_SPI_CR2`)

Definition at line 1320 of file STM8AF\_STM8S.h.

#### 5.1.2.856 `_SPI_SSM`

```
#define _SPI_SSM ((uint8_t) (0x01 << 1))
```

SPI Software slave management [0] (in `_SPI_CR2`)

Definition at line 1321 of file STM8AF\_STM8S.h.

#### 5.1.2.857 `_SPI_TXCRCR`

```
#define _SPI_TXCRCR \_SFR(uint8_t, SPI\_AddressBase+0x07)
```

SPI Tx CRC register.

Definition at line 1296 of file STM8AF\_STM8S.h.

**5.1.2.858 \_SPI\_TXCRCR\_RESET\_VALUE**

```
#define _SPI_TXCRCR_RESET_VALUE ((uint8_t) 0x00)
```

SPI TX CRC Register reset value.

Definition at line 1306 of file STM8AF\_STM8S.h.

**5.1.2.859 \_SPI\_TXE**

```
#define _SPI_TXE ((uint8_t) (0x01 << 1))
```

SPI Transmit buffer empty [0] (in \_SPI\_SR)

Definition at line 1338 of file STM8AF\_STM8S.h.

**5.1.2.860 \_SPI\_TXIE**

```
#define _SPI_TXIE ((uint8_t) (0x01 << 7))
```

SPI Tx buffer empty interrupt enable [0] (in \_SPI\_ICR)

Definition at line 1334 of file STM8AF\_STM8S.h.

**5.1.2.861 \_SPI\_WKIE**

```
#define _SPI_WKIE ((uint8_t) (0x01 << 4))
```

SPI Wakeup interrupt enable [0] (in \_SPI\_ICR)

Definition at line 1331 of file STM8AF\_STM8S.h.

**5.1.2.862 \_SPI\_WKUP**

```
#define _SPI_WKUP ((uint8_t) (0x01 << 3))
```

SPI Wakeup flag [0] (in \_SPI\_SR)

Definition at line 1340 of file STM8AF\_STM8S.h.

### 5.1.2.863 `_TIM1`

```
#define _TIM1 _SFR(TIM1_t, TIM1_AddressBase)
```

TIM1 struct/bit access.

Definition at line 2803 of file STM8AF\_STM8S.h.

### 5.1.2.864 `_TIM1_AOE`

```
#define _TIM1_AOE ((uint8_t) (0x01 << 6))
```

TIM1 Automatic output enable [0] (in `_TIM1_BKR`)

Definition at line 3074 of file STM8AF\_STM8S.h.

### 5.1.2.865 `_TIM1_ARPE`

```
#define _TIM1_ARPE ((uint8_t) (0x01 << 7))
```

TIM1 Auto-reload preload enable [0] (in `_TIM1_CR1`)

Definition at line 2880 of file STM8AF\_STM8S.h.

### 5.1.2.866 `_TIM1_ARRH`

```
#define _TIM1_ARRH _SFR(uint8_t, TIM1_AddressBase+0x12)
```

TIM1 auto-reload register high byte.

Definition at line 2822 of file STM8AF\_STM8S.h.

### 5.1.2.867 `_TIM1_ARRH_RESET_VALUE`

```
#define _TIM1_ARRH_RESET_VALUE ((uint8_t) 0xFF)
```

TIM1 auto-reload register high byte reset value.

Definition at line 2856 of file STM8AF\_STM8S.h.

**5.1.2.868 \_TIM1\_ARRL**

```
#define _TIM1_ARRL _SFR(uint8_t, TIM1_AddressBase+0x13)
```

TIM1 auto-reload register low byte.

Definition at line 2823 of file STM8AF\_STM8S.h.

**5.1.2.869 \_TIM1\_ARRL\_RESET\_VALUE**

```
#define _TIM1_ARRL_RESET_VALUE ((uint8_t) 0xFF)
```

TIM1 auto-reload register low byte reset value.

Definition at line 2857 of file STM8AF\_STM8S.h.

**5.1.2.870 \_TIM1\_BG**

```
#define _TIM1_BG ((uint8_t) (0x01 << 7))
```

TIM1 Break generation [0] (in \_TIM1\_EGR)

Definition at line 2953 of file STM8AF\_STM8S.h.

**5.1.2.871 \_TIM1\_BIE**

```
#define _TIM1_BIE ((uint8_t) (0x01 << 7))
```

TIM1 Break interrupt enable [0] (in \_TIM1\_IER)

Definition at line 2925 of file STM8AF\_STM8S.h.

**5.1.2.872 \_TIM1\_BIF**

```
#define _TIM1_BIF ((uint8_t) (0x01 << 7))
```

TIM1 Break interrupt flag [0] (in \_TIM1\_SR1)

Definition at line 2935 of file STM8AF\_STM8S.h.

#### 5.1.2.873 `_TIM1_BKE`

```
#define _TIM1_BKE ((uint8_t) (0x01 << 4))
```

TIM1 Break enable [0] (in `_TIM1_BKR`)

Definition at line 3072 of file `STM8AF_STM8S.h`.

#### 5.1.2.874 `_TIM1_BKP`

```
#define _TIM1_BKP ((uint8_t) (0x01 << 5))
```

TIM1 Break polarity [0] (in `_TIM1_BKR`)

Definition at line 3073 of file `STM8AF_STM8S.h`.

#### 5.1.2.875 `_TIM1_BKR`

```
#define _TIM1_BKR _SFR(uint8_t, TIM1_AddressBase+0x1D)
```

TIM1 Break register.

Definition at line 2833 of file `STM8AF_STM8S.h`.

#### 5.1.2.876 `_TIM1_BKR_RESET_VALUE`

```
#define _TIM1_BKR_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Break register reset value.

Definition at line 2867 of file `STM8AF_STM8S.h`.

#### 5.1.2.877 `_TIM1_CC1E`

```
#define _TIM1_CC1E ((uint8_t) (0x01 << 0))
```

TIM1 Capture/compare 1 output enable [0] (in `_TIM1_CCER1`)

Definition at line 3048 of file `STM8AF_STM8S.h`.

**5.1.2.878 \_TIM1\_CC1G**

```
#define _TIM1_CC1G ((uint8_t) (0x01 << 1))
```

TIM1 Capture/compare 1 generation [0] (in \_TIM1\_EGR)

Definition at line 2947 of file STM8AF\_STM8S.h.

**5.1.2.879 \_TIM1\_CC1IE**

```
#define _TIM1_CC1IE ((uint8_t) (0x01 << 1))
```

TIM1 Capture/compare 1 interrupt enable [0] (in \_TIM1\_IER)

Definition at line 2919 of file STM8AF\_STM8S.h.

**5.1.2.880 \_TIM1\_CC1IF**

```
#define _TIM1_CC1IF ((uint8_t) (0x01 << 1))
```

TIM1 Capture/compare 1 interrupt flag [0] (in \_TIM1\_SR1)

Definition at line 2929 of file STM8AF\_STM8S.h.

**5.1.2.881 \_TIM1\_CC1NE**

```
#define _TIM1_CC1NE ((uint8_t) (0x01 << 2))
```

TIM1 Capture/compare 1 complementary output enable [0] (in \_TIM1\_CCER1)

Definition at line 3050 of file STM8AF\_STM8S.h.

**5.1.2.882 \_TIM1\_CC1NP**

```
#define _TIM1_CC1NP ((uint8_t) (0x01 << 3))
```

TIM1 Capture/compare 1 complementary output polarity [0] (in \_TIM1\_CCER1)

Definition at line 3051 of file STM8AF\_STM8S.h.

**5.1.2.883 \_TIM1\_CC1OF**

```
#define _TIM1_CC1OF ((uint8_t) (0x01 << 1))
```

TIM1 Capture/compare 1 overcapture flag [0] (in \_TIM1\_SR2)

Definition at line 2939 of file STM8AF\_STM8S.h.

**5.1.2.884 \_TIM1\_CC1P**

```
#define _TIM1_CC1P ((uint8_t) (0x01 << 1))
```

TIM1 Capture/compare 1 output polarity [0] (in \_TIM1\_CCER1)

Definition at line 3049 of file STM8AF\_STM8S.h.

**5.1.2.885 \_TIM1\_CC1S**

```
#define _TIM1_CC1S ((uint8_t) (0x03 << 0))
```

TIM1 Compare 1 selection [1:0] (in \_TIM1\_CCMR1)

Definition at line 2956 of file STM8AF\_STM8S.h.

**5.1.2.886 \_TIM1\_CC1S0**

```
#define _TIM1_CC1S0 ((uint8_t) (0x01 << 0))
```

TIM1 Compare 1 selection [0] (in \_TIM1\_CCMR1)

Definition at line 2957 of file STM8AF\_STM8S.h.

**5.1.2.887 \_TIM1\_CC1S1**

```
#define _TIM1_CC1S1 ((uint8_t) (0x01 << 1))
```

TIM1 Compare 1 selection [1] (in \_TIM1\_CCMR1)

Definition at line 2958 of file STM8AF\_STM8S.h.



**5.1.2.888 \_TIM1\_CC2E**

```
#define _TIM1_CC2E ((uint8_t) (0x01 << 4))
```

TIM1 Capture/compare 2 output enable [0] (in \_TIM1\_CCER1)

Definition at line 3052 of file STM8AF\_STM8S.h.

**5.1.2.889 \_TIM1\_CC2G**

```
#define _TIM1_CC2G ((uint8_t) (0x01 << 2))
```

TIM1 Capture/compare 2 generation [0] (in \_TIM1\_EGR)

Definition at line 2948 of file STM8AF\_STM8S.h.

**5.1.2.890 \_TIM1\_CC2IE**

```
#define _TIM1_CC2IE ((uint8_t) (0x01 << 2))
```

TIM1 Capture/compare 2 interrupt enable [0] (in \_TIM1\_IER)

Definition at line 2920 of file STM8AF\_STM8S.h.

**5.1.2.891 \_TIM1\_CC2IF**

```
#define _TIM1_CC2IF ((uint8_t) (0x01 << 2))
```

TIM1 Capture/compare 2 interrupt flag [0] (in \_TIM1\_SR1)

Definition at line 2930 of file STM8AF\_STM8S.h.

**5.1.2.892 \_TIM1\_CC2NE**

```
#define _TIM1_CC2NE ((uint8_t) (0x01 << 6))
```

TIM1 Capture/compare 2 complementary output enable [0] (in \_TIM1\_CCER1)

Definition at line 3054 of file STM8AF\_STM8S.h.

**5.1.2.893 \_TIM1\_CC2NP**

```
#define _TIM1_CC2NP ((uint8_t) (0x01 << 7))
```

TIM1 Capture/compare 2 complementary output polarity [0] (in \_TIM1\_CCER1)

Definition at line 3055 of file STM8AF\_STM8S.h.

**5.1.2.894 \_TIM1\_CC2OF**

```
#define _TIM1_CC2OF ((uint8_t) (0x01 << 2))
```

TIM1 Capture/compare 2 overcapture flag [0] (in \_TIM1\_SR2)

Definition at line 2940 of file STM8AF\_STM8S.h.

**5.1.2.895 \_TIM1\_CC2P**

```
#define _TIM1_CC2P ((uint8_t) (0x01 << 5))
```

TIM1 Capture/compare 2 output polarity [0] (in \_TIM1\_CCER1)

Definition at line 3053 of file STM8AF\_STM8S.h.

**5.1.2.896 \_TIM1\_CC2S**

```
#define _TIM1_CC2S ((uint8_t) (0x03 << 0))
```

TIM1 Compare 2 selection [1:0] (in \_TIM1\_CCMR2)

Definition at line 2979 of file STM8AF\_STM8S.h.

**5.1.2.897 \_TIM1\_CC2S0**

```
#define _TIM1_CC2S0 ((uint8_t) (0x01 << 0))
```

TIM1 Compare 2 selection [0] (in \_TIM1\_CCMR2)

Definition at line 2980 of file STM8AF\_STM8S.h.

**5.1.2.898 \_TIM1\_CC2S1**

```
#define _TIM1_CC2S1 ((uint8_t) (0x01 << 1))
```

TIM1 Compare 2 selection [1] (in \_TIM1\_CCMR2)

Definition at line 2981 of file STM8AF\_STM8S.h.

**5.1.2.899 \_TIM1\_CC3E**

```
#define _TIM1_CC3E ((uint8_t) (0x01 << 0))
```

TIM1 Capture/compare 3 output enable [0] (in \_TIM1\_CCER2)

Definition at line 3058 of file STM8AF\_STM8S.h.

**5.1.2.900 \_TIM1\_CC3G**

```
#define _TIM1_CC3G ((uint8_t) (0x01 << 3))
```

TIM1 Capture/compare 3 generation [0] (in \_TIM1\_EGR)

Definition at line 2949 of file STM8AF\_STM8S.h.

**5.1.2.901 \_TIM1\_CC3IE**

```
#define _TIM1_CC3IE ((uint8_t) (0x01 << 3))
```

TIM1 Capture/compare 3 interrupt enable [0] (in \_TIM1\_IER)

Definition at line 2921 of file STM8AF\_STM8S.h.

**5.1.2.902 \_TIM1\_CC3IF**

```
#define _TIM1_CC3IF ((uint8_t) (0x01 << 3))
```

TIM1 Capture/compare 3 interrupt flag [0] (in \_TIM1\_SR1)

Definition at line 2931 of file STM8AF\_STM8S.h.

#### 5.1.2.903 `_TIM1_CC3NE`

```
#define _TIM1_CC3NE ((uint8_t) (0x01 << 2))
```

TIM1 Capture/compare 3 complementary output enable [0] (in `_TIM1_CCER2`)

Definition at line 3060 of file `STM8AF_STM8S.h`.

#### 5.1.2.904 `_TIM1_CC3NP`

```
#define _TIM1_CC3NP ((uint8_t) (0x01 << 3))
```

TIM1 Capture/compare 3 complementary output polarity [0] (in `_TIM1_CCER2`)

Definition at line 3061 of file `STM8AF_STM8S.h`.

#### 5.1.2.905 `_TIM1_CC3OF`

```
#define _TIM1_CC3OF ((uint8_t) (0x01 << 3))
```

TIM1 Capture/compare 3 overcapture flag [0] (in `_TIM1_SR2`)

Definition at line 2941 of file `STM8AF_STM8S.h`.

#### 5.1.2.906 `_TIM1_CC3P`

```
#define _TIM1_CC3P ((uint8_t) (0x01 << 1))
```

TIM1 Capture/compare 3 output polarity [0] (in `_TIM1_CCER2`)

Definition at line 3059 of file `STM8AF_STM8S.h`.

#### 5.1.2.907 `_TIM1_CC3S`

```
#define _TIM1_CC3S ((uint8_t) (0x03 << 0))
```

TIM1 Compare 3 selection [1:0] (in `_TIM1_CCMR3`)

Definition at line 3002 of file `STM8AF_STM8S.h`.

**5.1.2.908 \_TIM1\_CC3S0**

```
#define _TIM1_CC3S0 ((uint8_t) (0x01 << 0))
```

TIM1 Compare 3 selection [0] (in \_TIM1\_CCMR3)

Definition at line 3003 of file STM8AF\_STM8S.h.

**5.1.2.909 \_TIM1\_CC3S1**

```
#define _TIM1_CC3S1 ((uint8_t) (0x01 << 1))
```

TIM1 Compare 3 selection [1] (in \_TIM1\_CCMR3)

Definition at line 3004 of file STM8AF\_STM8S.h.

**5.1.2.910 \_TIM1\_CC4E**

```
#define _TIM1_CC4E ((uint8_t) (0x01 << 4))
```

TIM1 Capture/compare 4 output enable [0] (in \_TIM1\_CCER2)

Definition at line 3062 of file STM8AF\_STM8S.h.

**5.1.2.911 \_TIM1\_CC4G**

```
#define _TIM1_CC4G ((uint8_t) (0x01 << 4))
```

TIM1 Capture/compare 4 generation [0] (in \_TIM1\_EGR)

Definition at line 2950 of file STM8AF\_STM8S.h.

**5.1.2.912 \_TIM1\_CC4IE**

```
#define _TIM1_CC4IE ((uint8_t) (0x01 << 4))
```

TIM1 Capture/compare 4 interrupt enable [0] (in \_TIM1\_IER)

Definition at line 2922 of file STM8AF\_STM8S.h.

#### 5.1.2.913 `_TIM1_CC4IF`

```
#define _TIM1_CC4IF ((uint8_t) (0x01 << 4))
```

TIM1 Capture/compare 4 interrupt flag [0] (in `_TIM1_SR1`)

Definition at line 2932 of file `STM8AF_STM8S.h`.

#### 5.1.2.914 `_TIM1_CC4OF`

```
#define _TIM1_CC4OF ((uint8_t) (0x01 << 4))
```

TIM1 Capture/compare 4 overcapture flag [0] (in `_TIM1_SR2`)

Definition at line 2942 of file `STM8AF_STM8S.h`.

#### 5.1.2.915 `_TIM1_CC4P`

```
#define _TIM1_CC4P ((uint8_t) (0x01 << 5))
```

TIM1 Capture/compare 4 output polarity [0] (in `_TIM1_CCER2`)

Definition at line 3063 of file `STM8AF_STM8S.h`.

#### 5.1.2.916 `_TIM1_CC4S`

```
#define _TIM1_CC4S ((uint8_t) (0x03 << 0))
```

TIM1 Compare 4 selection [1:0] (in `_TIM1_CCMR4`)

Definition at line 3025 of file `STM8AF_STM8S.h`.

#### 5.1.2.917 `_TIM1_CC4S0`

```
#define _TIM1_CC4S0 ((uint8_t) (0x01 << 0))
```

TIM1 Compare 4 selection [0] (in `_TIM1_CCMR4`)

Definition at line 3026 of file `STM8AF_STM8S.h`.

**5.1.2.918 \_TIM1\_CC4S1**

```
#define _TIM1_CC4S1 ((uint8_t) (0x01 << 1))
```

TIM1 Compare 4 selection [1] (in \_TIM1\_CCMR4)

Definition at line 3027 of file STM8AF\_STM8S.h.

**5.1.2.919 \_TIM1\_CCER1**

```
#define _TIM1_CCER1 _SFR(uint8_t, TIM1_AddressBase+0x0C)
```

TIM1 Capture/compare enable register 1.

Definition at line 2816 of file STM8AF\_STM8S.h.

**5.1.2.920 \_TIM1\_CCER1\_RESET\_VALUE**

```
#define _TIM1_CCER1_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Capture/compare enable register 1 reset value.

Definition at line 2850 of file STM8AF\_STM8S.h.

**5.1.2.921 \_TIM1\_CCER2**

```
#define _TIM1_CCER2 _SFR(uint8_t, TIM1_AddressBase+0x0D)
```

TIM1 Capture/compare enable register 2.

Definition at line 2817 of file STM8AF\_STM8S.h.

**5.1.2.922 \_TIM1\_CCER2\_RESET\_VALUE**

```
#define _TIM1_CCER2_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Capture/compare enable register 2 reset value.

Definition at line 2851 of file STM8AF\_STM8S.h.

#### 5.1.2.923 `_TIM1_CCMR1`

```
#define _TIM1_CCMR1 __SFR(uint8_t, TIM1_AddressBase+0x08)
```

TIM1 Capture/compare mode register 1.

Definition at line 2812 of file STM8AF\_STM8S.h.

#### 5.1.2.924 `_TIM1_CCMR1_RESET_VALUE`

```
#define _TIM1_CCMR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Capture/compare mode register 1 reset value.

Definition at line 2846 of file STM8AF\_STM8S.h.

#### 5.1.2.925 `_TIM1_CCMR2`

```
#define _TIM1_CCMR2 __SFR(uint8_t, TIM1_AddressBase+0x09)
```

TIM1 Capture/compare mode register 2.

Definition at line 2813 of file STM8AF\_STM8S.h.

#### 5.1.2.926 `_TIM1_CCMR2_RESET_VALUE`

```
#define _TIM1_CCMR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Capture/compare mode register 2 reset value.

Definition at line 2847 of file STM8AF\_STM8S.h.

#### 5.1.2.927 `_TIM1_CCMR3`

```
#define _TIM1_CCMR3 __SFR(uint8_t, TIM1_AddressBase+0x0A)
```

TIM1 Capture/compare mode register 3.

Definition at line 2814 of file STM8AF\_STM8S.h.



**5.1.2.928 \_TIM1\_CCMR3\_RESET\_VALUE**

```
#define _TIM1_CCMR3_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Capture/compare mode register 3 reset value.

Definition at line 2848 of file STM8AF\_STM8S.h.

**5.1.2.929 \_TIM1\_CCMR4**

```
#define _TIM1_CCMR4 __SFR(uint8_t, TIM1_AddressBase+0x0B)
```

TIM1 Capture/compare mode register 4.

Definition at line 2815 of file STM8AF\_STM8S.h.

**5.1.2.930 \_TIM1\_CCMR4\_RESET\_VALUE**

```
#define _TIM1_CCMR4_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Capture/compare mode register 4 reset value.

Definition at line 2849 of file STM8AF\_STM8S.h.

**5.1.2.931 \_TIM1\_CCPC**

```
#define _TIM1_CCPC ((uint8_t) (0x01 << 0))
```

TIM1 Capture/compare preloaded control [0] (in \_TIM1\_CR2)

Definition at line 2883 of file STM8AF\_STM8S.h.

**5.1.2.932 \_TIM1\_CCR1H**

```
#define _TIM1_CCR1H __SFR(uint8_t, TIM1_AddressBase+0x15)
```

TIM1 16-bit capture/compare value 1 high byte.

Definition at line 2825 of file STM8AF\_STM8S.h.

#### 5.1.2.933 `_TIM1_CCR1H_RESET_VALUE`

```
#define _TIM1_CCR1H_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 1 high byte reset value.

Definition at line 2859 of file STM8AF\_STM8S.h.

#### 5.1.2.934 `_TIM1_CCR1L`

```
#define _TIM1_CCR1L \_SFR(uint8_t, TIM1\_AddressBase+0x16)
```

TIM1 16-bit capture/compare value 1 low byte.

Definition at line 2826 of file STM8AF\_STM8S.h.

#### 5.1.2.935 `_TIM1_CCR1L_RESET_VALUE`

```
#define _TIM1_CCR1L_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 1 low byte reset value.

Definition at line 2860 of file STM8AF\_STM8S.h.

#### 5.1.2.936 `_TIM1_CCR2H`

```
#define _TIM1_CCR2H \_SFR(uint8_t, TIM1\_AddressBase+0x17)
```

TIM1 16-bit capture/compare value 2 high byte.

Definition at line 2827 of file STM8AF\_STM8S.h.

#### 5.1.2.937 `_TIM1_CCR2H_RESET_VALUE`

```
#define _TIM1_CCR2H_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 2 high byte reset value.

Definition at line 2861 of file STM8AF\_STM8S.h.

**5.1.2.938 \_TIM1\_CCR2L**

```
#define _TIM1_CCR2L _SFR(uint8_t, TIM1_AddressBase+0x18)
```

TIM1 16-bit capture/compare value 2 low byte.

Definition at line 2828 of file STM8AF\_STM8S.h.

**5.1.2.939 \_TIM1\_CCR2L\_RESET\_VALUE**

```
#define _TIM1_CCR2L_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 2 low byte reset value.

Definition at line 2862 of file STM8AF\_STM8S.h.

**5.1.2.940 \_TIM1\_CCR3H**

```
#define _TIM1_CCR3H _SFR(uint8_t, TIM1_AddressBase+0x19)
```

TIM1 16-bit capture/compare value 3 high byte.

Definition at line 2829 of file STM8AF\_STM8S.h.

**5.1.2.941 \_TIM1\_CCR3H\_RESET\_VALUE**

```
#define _TIM1_CCR3H_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 3 high byte reset value.

Definition at line 2863 of file STM8AF\_STM8S.h.

**5.1.2.942 \_TIM1\_CCR3L**

```
#define _TIM1_CCR3L _SFR(uint8_t, TIM1_AddressBase+0x1A)
```

TIM1 16-bit capture/compare value 3 low byte.

Definition at line 2830 of file STM8AF\_STM8S.h.

#### 5.1.2.943 `_TIM1_CCR3L_RESET_VALUE`

```
#define _TIM1_CCR3L_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 3 low byte reset value.

Definition at line 2864 of file STM8AF\_STM8S.h.

#### 5.1.2.944 `_TIM1_CCR4H`

```
#define _TIM1_CCR4H \_SFR(uint8_t, TIM1\_AddressBase+0x1B)
```

TIM1 16-bit capture/compare value 4 high byte.

Definition at line 2831 of file STM8AF\_STM8S.h.

#### 5.1.2.945 `_TIM1_CCR4H_RESET_VALUE`

```
#define _TIM1_CCR4H_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 4 high byte reset value.

Definition at line 2865 of file STM8AF\_STM8S.h.

#### 5.1.2.946 `_TIM1_CCR4L`

```
#define _TIM1_CCR4L \_SFR(uint8_t, TIM1\_AddressBase+0x1C)
```

TIM1 16-bit capture/compare value 4 low byte.

Definition at line 2832 of file STM8AF\_STM8S.h.

#### 5.1.2.947 `_TIM1_CCR4L_RESET_VALUE`

```
#define _TIM1_CCR4L_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 16-bit capture/compare value 4 low byte reset value.

Definition at line 2866 of file STM8AF\_STM8S.h.

**5.1.2.948 \_TIM1\_CEN**

```
#define _TIM1_CEN ((uint8_t) (0x01 << 0))
```

TIM1 Counter enable [0] (in \_TIM1\_CR1)

Definition at line 2872 of file STM8AF\_STM8S.h.

**5.1.2.949 \_TIM1\_CMS**

```
#define _TIM1_CMS ((uint8_t) (0x03 << 5))
```

TIM1 Center-aligned mode selection [1:0] (in \_TIM1\_CR1)

Definition at line 2877 of file STM8AF\_STM8S.h.

**5.1.2.950 \_TIM1\_CMS0**

```
#define _TIM1_CMS0 ((uint8_t) (0x01 << 5))
```

TIM1 Center-aligned mode selection [0] (in \_TIM1\_CR1)

Definition at line 2878 of file STM8AF\_STM8S.h.

**5.1.2.951 \_TIM1\_CMS1**

```
#define _TIM1_CMS1 ((uint8_t) (0x01 << 6))
```

TIM1 Center-aligned mode selection [1] (in \_TIM1\_CR1)

Definition at line 2879 of file STM8AF\_STM8S.h.

**5.1.2.952 \_TIM1\_CNTRH**

```
#define _TIM1_CNTRH _SFR(uint8_t, TIM1_AddressBase+0x0E)
```

TIM1 counter register high byte.

Definition at line 2818 of file STM8AF\_STM8S.h.

#### 5.1.2.953 `_TIM1_CNTRH_RESET_VALUE`

```
#define _TIM1_CNTRH_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 counter register high byte reset value.

Definition at line 2852 of file STM8AF\_STM8S.h.

#### 5.1.2.954 `_TIM1_CNTRL`

```
#define _TIM1_CNTRL __SFR(uint8_t, TIM1_AddressBase+0x0F)
```

TIM1 counter register low byte.

Definition at line 2819 of file STM8AF\_STM8S.h.

#### 5.1.2.955 `_TIM1_CNTRL_RESET_VALUE`

```
#define _TIM1_CNTRL_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 counter register low byte reset value.

Definition at line 2853 of file STM8AF\_STM8S.h.

#### 5.1.2.956 `_TIM1_COMG`

```
#define _TIM1_COMG ((uint8_t) (0x01 << 5))
```

TIM1 Capture/compare control update generation [0] (in `_TIM1_EGR`)

Definition at line 2951 of file STM8AF\_STM8S.h.

#### 5.1.2.957 `_TIM1_COMIE`

```
#define _TIM1_COMIE ((uint8_t) (0x01 << 5))
```

TIM1 Commutation interrupt enable [0] (in `_TIM1_IER`)

Definition at line 2923 of file STM8AF\_STM8S.h.

**5.1.2.958 \_TIM1\_COMIF**

```
#define _TIM1_COMIF ((uint8_t) (0x01 << 5))
```

TIM1 Commutation interrupt flag [0] (in \_TIM1\_SR1)

Definition at line 2933 of file STM8AF\_STM8S.h.

**5.1.2.959 \_TIM1\_COMS**

```
#define _TIM1_COMS ((uint8_t) (0x01 << 2))
```

TIM1 Capture/compare control update selection [0] (in \_TIM1\_CR2)

Definition at line 2885 of file STM8AF\_STM8S.h.

**5.1.2.960 \_TIM1\_CR1**

```
#define _TIM1_CR1 _SFR(uint8_t, TIM1_AddressBase+0x00)
```

TIM1 control register 1.

Definition at line 2804 of file STM8AF\_STM8S.h.

**5.1.2.961 \_TIM1\_CR1\_RESET\_VALUE**

```
#define _TIM1_CR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 control register 1 reset value.

Definition at line 2838 of file STM8AF\_STM8S.h.

**5.1.2.962 \_TIM1\_CR2**

```
#define _TIM1_CR2 _SFR(uint8_t, TIM1_AddressBase+0x01)
```

TIM1 control register 2.

Definition at line 2805 of file STM8AF\_STM8S.h.

#### 5.1.2.963 `_TIM1_CR2_RESET_VALUE`

```
#define _TIM1_CR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 control register 2 reset value.

Definition at line 2839 of file STM8AF\_STM8S.h.

#### 5.1.2.964 `_TIM1_DIR`

```
#define _TIM1_DIR ((uint8_t) (0x01 << 4))
```

TIM1 Direction [0] (in `_TIM1_CR1`)

Definition at line 2876 of file STM8AF\_STM8S.h.

#### 5.1.2.965 `_TIM1_DTR`

```
#define _TIM1_DTR \_SFR(uint8_t, TIM1\_AddressBase+0x1E)
```

TIM1 Dead-time register.

Definition at line 2834 of file STM8AF\_STM8S.h.

#### 5.1.2.966 `_TIM1_DTR_RESET_VALUE`

```
#define _TIM1_DTR_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Dead-time register reset value.

Definition at line 2868 of file STM8AF\_STM8S.h.

#### 5.1.2.967 `_TIM1_ECE`

```
#define _TIM1_ECE ((uint8_t) (0x01 << 6))
```

TIM1 External clock enable [0] (in `_TIM1_ETR`)

Definition at line 2914 of file STM8AF\_STM8S.h.



**5.1.2.968 \_TIM1\_EGR**

```
#define _TIM1_EGR _SFR(uint8_t, TIM1_AddressBase+0x07)
```

TIM1 Event generation register.

Definition at line 2811 of file STM8AF\_STM8S.h.

**5.1.2.969 \_TIM1\_EGR\_RESET\_VALUE**

```
#define _TIM1_EGR_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Event generation register reset value.

Definition at line 2845 of file STM8AF\_STM8S.h.

**5.1.2.970 \_TIM1 ETF**

```
#define _TIM1 ETF ((uint8_t) (0x0F << 0))
```

TIM1 External trigger filter [3:0] (in \_TIM1\_ETR)

Definition at line 2906 of file STM8AF\_STM8S.h.

**5.1.2.971 \_TIM1 ETF0**

```
#define _TIM1 ETF0 ((uint8_t) (0x01 << 0))
```

TIM1 External trigger filter [0] (in \_TIM1\_ETR)

Definition at line 2907 of file STM8AF\_STM8S.h.

**5.1.2.972 \_TIM1 ETF1**

```
#define _TIM1 ETF1 ((uint8_t) (0x01 << 1))
```

TIM1 External trigger filter [1] (in \_TIM1\_ETR)

Definition at line 2908 of file STM8AF\_STM8S.h.

**5.1.2.973 \_TIM1 ETF2**

```
#define _TIM1 ETF2 ((uint8_t) (0x01 << 2))
```

TIM1 External trigger filter [2] (in \_TIM1\_ETR)

Definition at line 2909 of file STM8AF\_STM8S.h.

**5.1.2.974 \_TIM1 ETF3**

```
#define _TIM1 ETF3 ((uint8_t) (0x01 << 3))
```

TIM1 External trigger filter [3] (in \_TIM1\_ETR)

Definition at line 2910 of file STM8AF\_STM8S.h.

**5.1.2.975 \_TIM1 ETP**

```
#define _TIM1 ETP ((uint8_t) (0x01 << 7))
```

TIM1 External trigger polarity [0] (in \_TIM1\_ETR)

Definition at line 2915 of file STM8AF\_STM8S.h.

**5.1.2.976 \_TIM1 ETPS**

```
#define _TIM1 ETPS ((uint8_t) (0x03 << 4))
```

TIM1 External trigger prescaler [1:0] (in \_TIM1\_ETR)

Definition at line 2911 of file STM8AF\_STM8S.h.

**5.1.2.977 \_TIM1 ETPS0**

```
#define _TIM1 ETPS0 ((uint8_t) (0x01 << 4))
```

TIM1 External trigger prescaler [0] (in \_TIM1\_ETR)

Definition at line 2912 of file STM8AF\_STM8S.h.

**5.1.2.978 \_TIM1\_ETPS1**

```
#define _TIM1_ETPS1 ((uint8_t) (0x01 << 5))
```

TIM1 External trigger prescaler [1] (in \_TIM1\_ETR)

Definition at line 2913 of file STM8AF\_STM8S.h.

**5.1.2.979 \_TIM1\_ETR**

```
#define _TIM1_ETR _SFR(uint8_t, TIM1_AddressBase+0x03)
```

TIM1 External trigger register.

Definition at line 2807 of file STM8AF\_STM8S.h.

**5.1.2.980 \_TIM1\_ETR\_RESET\_VALUE**

```
#define _TIM1_ETR_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 External trigger register reset value.

Definition at line 2841 of file STM8AF\_STM8S.h.

**5.1.2.981 \_TIM1\_IC1F**

```
#define _TIM1_IC1F ((uint8_t) (0x0F << 4))
```

TIM1 Output compare 1 mode [3:0] (in \_TIM1\_CCMR1)

Definition at line 2972 of file STM8AF\_STM8S.h.

**5.1.2.982 \_TIM1\_IC1F0**

```
#define _TIM1_IC1F0 ((uint8_t) (0x01 << 4))
```

TIM1 Input capture 1 filter [0] (in \_TIM1\_CCMR1)

Definition at line 2973 of file STM8AF\_STM8S.h.

**5.1.2.983 \_TIM1\_IC1F1**

```
#define _TIM1_IC1F1 ((uint8_t) (0x01 << 5))
```

TIM1 Input capture 1 filter [1] (in \_TIM1\_CCMR1)

Definition at line 2974 of file STM8AF\_STM8S.h.

**5.1.2.984 \_TIM1\_IC1F2**

```
#define _TIM1_IC1F2 ((uint8_t) (0x01 << 6))
```

TIM1 Input capture 1 filter [2] (in \_TIM1\_CCMR1)

Definition at line 2975 of file STM8AF\_STM8S.h.

**5.1.2.985 \_TIM1\_IC1F3**

```
#define _TIM1_IC1F3 ((uint8_t) (0x01 << 7))
```

TIM1 Input capture 1 filter [3] (in \_TIM1\_CCMR1)

Definition at line 2976 of file STM8AF\_STM8S.h.

**5.1.2.986 \_TIM1\_IC1PSC**

```
#define _TIM1_IC1PSC ((uint8_t) (0x03 << 2))
```

TIM1 Input capture 1 prescaler [1:0] (in \_TIM1\_CCMR1)

Definition at line 2969 of file STM8AF\_STM8S.h.

**5.1.2.987 \_TIM1\_IC1PSC0**

```
#define _TIM1_IC1PSC0 ((uint8_t) (0x01 << 2))
```

TIM1 Input capture 1 prescaler [0] (in \_TIM1\_CCMR1)

Definition at line 2970 of file STM8AF\_STM8S.h.

**5.1.2.988 \_TIM1\_IC1PSC1**

```
#define _TIM1_IC1PSC1 ((uint8_t) (0x01 << 3))
```

TIM1 Input capture 1 prescaler [1] (in \_TIM1\_CCMR1)

Definition at line 2971 of file STM8AF\_STM8S.h.

**5.1.2.989 \_TIM1\_IC2F**

```
#define _TIM1_IC2F ((uint8_t) (0x0F << 4))
```

TIM1 Output compare 2 mode [3:0] (in \_TIM1\_CCMR2)

Definition at line 2995 of file STM8AF\_STM8S.h.

**5.1.2.990 \_TIM1\_IC2F0**

```
#define _TIM1_IC2F0 ((uint8_t) (0x01 << 4))
```

TIM1 Input capture 2 filter [0] (in \_TIM1\_CCMR2)

Definition at line 2996 of file STM8AF\_STM8S.h.

**5.1.2.991 \_TIM1\_IC2F1**

```
#define _TIM1_IC2F1 ((uint8_t) (0x01 << 5))
```

TIM1 Input capture 2 filter [1] (in \_TIM1\_CCMR2)

Definition at line 2997 of file STM8AF\_STM8S.h.

**5.1.2.992 \_TIM1\_IC2F2**

```
#define _TIM1_IC2F2 ((uint8_t) (0x01 << 6))
```

TIM1 Input capture 2 filter [2] (in \_TIM1\_CCMR2)

Definition at line 2998 of file STM8AF\_STM8S.h.

#### 5.1.2.993 `_TIM1_IC2F3`

```
#define _TIM1_IC2F3 ((uint8_t) (0x01 << 7))
```

TIM1 Input capture 2 filter [3] (in `_TIM1_CCMR2`)

Definition at line 2999 of file `STM8AF_STM8S.h`.

#### 5.1.2.994 `_TIM1_IC2PSC`

```
#define _TIM1_IC2PSC ((uint8_t) (0x03 << 2))
```

TIM1 Input capture 2 prescaler [1:0] (in `_TIM1_CCMR2`)

Definition at line 2992 of file `STM8AF_STM8S.h`.

#### 5.1.2.995 `_TIM1_IC2PSC0`

```
#define _TIM1_IC2PSC0 ((uint8_t) (0x01 << 2))
```

TIM1 Input capture 2 prescaler [0] (in `_TIM1_CCMR2`)

Definition at line 2993 of file `STM8AF_STM8S.h`.

#### 5.1.2.996 `_TIM1_IC2PSC1`

```
#define _TIM1_IC2PSC1 ((uint8_t) (0x01 << 3))
```

TIM1 Input capture 2 prescaler [1] (in `_TIM1_CCMR2`)

Definition at line 2994 of file `STM8AF_STM8S.h`.

#### 5.1.2.997 `_TIM1_IC3F`

```
#define _TIM1_IC3F ((uint8_t) (0x0F << 4))
```

TIM1 Output compare 3 mode [3:0] (in `_TIM1_CCMR3`)

Definition at line 3018 of file `STM8AF_STM8S.h`.

**5.1.2.998 \_TIM1\_IC3F0**

```
#define _TIM1_IC3F0 ((uint8_t) (0x01 << 4))
```

TIM1 Input capture 3 filter [0] (in \_TIM1\_CCMR3)

Definition at line 3019 of file STM8AF\_STM8S.h.

**5.1.2.999 \_TIM1\_IC3F1**

```
#define _TIM1_IC3F1 ((uint8_t) (0x01 << 5))
```

TIM1 Input capture 3 filter [1] (in \_TIM1\_CCMR3)

Definition at line 3020 of file STM8AF\_STM8S.h.

**5.1.2.1000 \_TIM1\_IC3F2**

```
#define _TIM1_IC3F2 ((uint8_t) (0x01 << 6))
```

TIM1 Input capture 3 filter [2] (in \_TIM1\_CCMR3)

Definition at line 3021 of file STM8AF\_STM8S.h.

**5.1.2.1001 \_TIM1\_IC3F3**

```
#define _TIM1_IC3F3 ((uint8_t) (0x01 << 7))
```

TIM1 Input capture 3 filter [3] (in \_TIM1\_CCMR3)

Definition at line 3022 of file STM8AF\_STM8S.h.

**5.1.2.1002 \_TIM1\_IC3PSC**

```
#define _TIM1_IC3PSC ((uint8_t) (0x03 << 2))
```

TIM1 Input capture 3 prescaler [1:0] (in \_TIM1\_CCMR3)

Definition at line 3015 of file STM8AF\_STM8S.h.

#### 5.1.2.1003 \_TIM1\_IC3PSC0

```
#define _TIM1_IC3PSC0 ((uint8_t) (0x01 << 2))
```

TIM1 Input capture 3 prescaler [0] (in \_TIM1\_CCMR3)

Definition at line 3016 of file STM8AF\_STM8S.h.

#### 5.1.2.1004 \_TIM1\_IC3PSC1

```
#define _TIM1_IC3PSC1 ((uint8_t) (0x01 << 3))
```

TIM1 Input capture 3 prescaler [1] (in \_TIM1\_CCMR3)

Definition at line 3017 of file STM8AF\_STM8S.h.

#### 5.1.2.1005 \_TIM1\_IC4F

```
#define _TIM1_IC4F ((uint8_t) (0x0F << 4))
```

TIM1 Output compare 4 mode [3:0] (in \_TIM1\_CCMR4)

Definition at line 3041 of file STM8AF\_STM8S.h.

#### 5.1.2.1006 \_TIM1\_IC4F0

```
#define _TIM1_IC4F0 ((uint8_t) (0x01 << 4))
```

TIM1 Input capture 4 filter [0] (in \_TIM1\_CCMR4)

Definition at line 3042 of file STM8AF\_STM8S.h.

#### 5.1.2.1007 \_TIM1\_IC4F1

```
#define _TIM1_IC4F1 ((uint8_t) (0x01 << 5))
```

TIM1 Input capture 4 filter [1] (in \_TIM1\_CCMR4)

Definition at line 3043 of file STM8AF\_STM8S.h.



**5.1.2.1008 \_TIM1\_IC4F2**

```
#define _TIM1_IC4F2 ((uint8_t) (0x01 << 6))
```

TIM1 Input capture 4 filter [2] (in \_TIM1\_CCMR4)

Definition at line 3044 of file STM8AF\_STM8S.h.

**5.1.2.1009 \_TIM1\_IC4F3**

```
#define _TIM1_IC4F3 ((uint8_t) (0x01 << 7))
```

TIM1 Input capture 4 filter [3] (in \_TIM1\_CCMR4)

Definition at line 3045 of file STM8AF\_STM8S.h.

**5.1.2.1010 \_TIM1\_IC4PSC**

```
#define _TIM1_IC4PSC ((uint8_t) (0x03 << 2))
```

TIM1 Input capture 4 prescaler [1:0] (in \_TIM1\_CCMR4)

Definition at line 3038 of file STM8AF\_STM8S.h.

**5.1.2.1011 \_TIM1\_IC4PSC0**

```
#define _TIM1_IC4PSC0 ((uint8_t) (0x01 << 2))
```

TIM1 Input capture 4 prescaler [0] (in \_TIM1\_CCMR4)

Definition at line 3039 of file STM8AF\_STM8S.h.

**5.1.2.1012 \_TIM1\_IC4PSC1**

```
#define _TIM1_IC4PSC1 ((uint8_t) (0x01 << 3))
```

TIM1 Input capture 4 prescaler [1] (in \_TIM1\_CCMR4)

Definition at line 3040 of file STM8AF\_STM8S.h.

#### 5.1.2.1013 \_TIM1\_IER

```
#define _TIM1_IER _SFR(uint8_t, TIM1_AddressBase+0x04)
```

TIM1 interrupt enable register.

Definition at line 2808 of file STM8AF\_STM8S.h.

#### 5.1.2.1014 \_TIM1\_IER\_RESET\_VALUE

```
#define _TIM1_IER_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 interrupt enable register reset value.

Definition at line 2842 of file STM8AF\_STM8S.h.

#### 5.1.2.1015 \_TIM1\_LOCK

```
#define _TIM1_LOCK ((int8_t) (0x03 << 0))
```

TIM1 Lock configuration [1:0] (in \_TIM1\_BKR)

Definition at line 3067 of file STM8AF\_STM8S.h.

#### 5.1.2.1016 \_TIM1\_LOCK0

```
#define _TIM1_LOCK0 ((uint8_t) (0x01 << 0))
```

TIM1 Lock configuration [0] (in \_TIM1\_BKR)

Definition at line 3068 of file STM8AF\_STM8S.h.

#### 5.1.2.1017 \_TIM1\_LOCK1

```
#define _TIM1_LOCK1 ((uint8_t) (0x01 << 1))
```

TIM1 Lock configuration [1] (in \_TIM1\_BKR)

Definition at line 3069 of file STM8AF\_STM8S.h.

**5.1.2.1018 \_TIM1\_MMS**

```
#define _TIM1_MMS ((uint8_t) (0x07 << 4))
```

TIM1 Master mode selection [2:0] (in \_TIM1\_CR2)

Definition at line 2887 of file STM8AF\_STM8S.h.

**5.1.2.1019 \_TIM1\_MMS0**

```
#define _TIM1_MMS0 ((uint8_t) (0x01 << 4))
```

TIM1 Master mode selection [0] (in \_TIM1\_CR2)

Definition at line 2888 of file STM8AF\_STM8S.h.

**5.1.2.1020 \_TIM1\_MMS1**

```
#define _TIM1_MMS1 ((uint8_t) (0x01 << 5))
```

TIM1 Master mode selection [1] (in \_TIM1\_CR2)

Definition at line 2889 of file STM8AF\_STM8S.h.

**5.1.2.1021 \_TIM1\_MMS2**

```
#define _TIM1_MMS2 ((uint8_t) (0x01 << 6))
```

TIM1 Master mode selection [2] (in \_TIM1\_CR2)

Definition at line 2890 of file STM8AF\_STM8S.h.

**5.1.2.1022 \_TIM1\_MOE**

```
#define _TIM1_MOE ((uint8_t) (0x01 << 7))
```

TIM1 Main output enable [0] (in \_TIM1\_BKR)

Definition at line 3075 of file STM8AF\_STM8S.h.

#### 5.1.2.1023 \_TIM1\_MSM

```
#define _TIM1_MSM ((uint8_t) (0x01 << 7))
```

TIM1 Master/slave mode [0] (in \_TIM1\_SMCR)

Definition at line 2903 of file STM8AF\_STM8S.h.

#### 5.1.2.1024 \_TIM1\_OC1CE

```
#define _TIM1_OC1CE ((uint8_t) (0x01 << 7))
```

TIM1 Output compare 1 clear enable [0] (in \_TIM1\_CCMR1)

Definition at line 2965 of file STM8AF\_STM8S.h.

#### 5.1.2.1025 \_TIM1\_OC1FE

```
#define _TIM1_OC1FE ((uint8_t) (0x01 << 2))
```

TIM1 Output compare 1 fast enable [0] (in \_TIM1\_CCMR1)

Definition at line 2959 of file STM8AF\_STM8S.h.

#### 5.1.2.1026 \_TIM1\_OC1M

```
#define _TIM1_OC1M ((uint8_t) (0x07 << 4))
```

TIM1 Output compare 1 mode [2:0] (in \_TIM1\_CCMR1)

Definition at line 2961 of file STM8AF\_STM8S.h.

#### 5.1.2.1027 \_TIM1\_OC1M0

```
#define _TIM1_OC1M0 ((uint8_t) (0x01 << 4))
```

TIM1 Output compare 1 mode [0] (in \_TIM1\_CCMR1)

Definition at line 2962 of file STM8AF\_STM8S.h.

**5.1.2.1028 \_TIM1\_OC1M1**

```
#define _TIM1_OC1M1 ((uint8_t) (0x01 << 5))
```

TIM1 Output compare 1 mode [1] (in \_TIM1\_CCMR1)

Definition at line 2963 of file STM8AF\_STM8S.h.

**5.1.2.1029 \_TIM1\_OC1M2**

```
#define _TIM1_OC1M2 ((uint8_t) (0x01 << 6))
```

TIM1 Output compare 1 mode [2] (in \_TIM1\_CCMR1)

Definition at line 2964 of file STM8AF\_STM8S.h.

**5.1.2.1030 \_TIM1\_OC1PE**

```
#define _TIM1_OC1PE ((uint8_t) (0x01 << 3))
```

TIM1 Output compare 1 preload enable [0] (in \_TIM1\_CCMR1)

Definition at line 2960 of file STM8AF\_STM8S.h.

**5.1.2.1031 \_TIM1\_OC2CE**

```
#define _TIM1_OC2CE ((uint8_t) (0x01 << 7))
```

TIM1 Output compare 2 clear enable [0] (in \_TIM1\_CCMR2)

Definition at line 2988 of file STM8AF\_STM8S.h.

**5.1.2.1032 \_TIM1\_OC2FE**

```
#define _TIM1_OC2FE ((uint8_t) (0x01 << 2))
```

TIM1 Output compare 2 fast enable [0] (in \_TIM1\_CCMR2)

Definition at line 2982 of file STM8AF\_STM8S.h.

#### 5.1.2.1033 \_TIM1\_OC2M

```
#define _TIM1_OC2M ((uint8_t) (0x07 << 4))
```

TIM1 Output compare 2 mode [2:0] (in \_TIM1\_CCMR2)

Definition at line 2984 of file STM8AF\_STM8S.h.

#### 5.1.2.1034 \_TIM1\_OC2M0

```
#define _TIM1_OC2M0 ((uint8_t) (0x01 << 4))
```

TIM1 Output compare 2 mode [0] (in \_TIM1\_CCMR2)

Definition at line 2985 of file STM8AF\_STM8S.h.

#### 5.1.2.1035 \_TIM1\_OC2M1

```
#define _TIM1_OC2M1 ((uint8_t) (0x01 << 5))
```

TIM1 Output compare 2 mode [1] (in \_TIM1\_CCMR2)

Definition at line 2986 of file STM8AF\_STM8S.h.

#### 5.1.2.1036 \_TIM1\_OC2M2

```
#define _TIM1_OC2M2 ((uint8_t) (0x01 << 6))
```

TIM1 Output compare 2 mode [2] (in \_TIM1\_CCMR2)

Definition at line 2987 of file STM8AF\_STM8S.h.

#### 5.1.2.1037 \_TIM1\_OC2PE

```
#define _TIM1_OC2PE ((uint8_t) (0x01 << 3))
```

TIM1 Output compare 2 preload enable [0] (in \_TIM1\_CCMR2)

Definition at line 2983 of file STM8AF\_STM8S.h.

**5.1.2.1038 \_TIM1\_OC3CE**

```
#define _TIM1_OC3CE ((uint8_t) (0x01 << 7))
```

TIM1 Output compare 3 clear enable [0] (in \_TIM1\_CCMR3)

Definition at line 3011 of file STM8AF\_STM8S.h.

**5.1.2.1039 \_TIM1\_OC3FE**

```
#define _TIM1_OC3FE ((uint8_t) (0x01 << 2))
```

TIM1 Output compare 3 fast enable [0] (in \_TIM1\_CCMR3)

Definition at line 3005 of file STM8AF\_STM8S.h.

**5.1.2.1040 \_TIM1\_OC3M**

```
#define _TIM1_OC3M ((uint8_t) (0x07 << 4))
```

TIM1 Output compare 3 mode [2:0] (in \_TIM1\_CCMR3)

Definition at line 3007 of file STM8AF\_STM8S.h.

**5.1.2.1041 \_TIM1\_OC3M0**

```
#define _TIM1_OC3M0 ((uint8_t) (0x01 << 4))
```

TIM1 Output compare 3 mode [0] (in \_TIM1\_CCMR3)

Definition at line 3008 of file STM8AF\_STM8S.h.

**5.1.2.1042 \_TIM1\_OC3M1**

```
#define _TIM1_OC3M1 ((uint8_t) (0x01 << 5))
```

TIM1 Output compare 3 mode [1] (in \_TIM1\_CCMR3)

Definition at line 3009 of file STM8AF\_STM8S.h.

#### 5.1.2.1043 \_TIM1\_OC3M2

```
#define _TIM1_OC3M2 ((uint8_t) (0x01 << 6))
```

TIM1 Output compare 3 mode [2] (in \_TIM1\_CCMR3)

Definition at line 3010 of file STM8AF\_STM8S.h.

#### 5.1.2.1044 \_TIM1\_OC3PE

```
#define _TIM1_OC3PE ((uint8_t) (0x01 << 3))
```

TIM1 Output compare 3 preload enable [0] (in \_TIM1\_CCMR3)

Definition at line 3006 of file STM8AF\_STM8S.h.

#### 5.1.2.1045 \_TIM1\_OC4CE

```
#define _TIM1_OC4CE ((uint8_t) (0x01 << 7))
```

TIM1 Output compare 4 clear enable [0] (in \_TIM1\_CCMR4)

Definition at line 3034 of file STM8AF\_STM8S.h.

#### 5.1.2.1046 \_TIM1\_OC4FE

```
#define _TIM1_OC4FE ((uint8_t) (0x01 << 2))
```

TIM1 Output compare 4 fast enable [0] (in \_TIM1\_CCMR4)

Definition at line 3028 of file STM8AF\_STM8S.h.

#### 5.1.2.1047 \_TIM1\_OC4M

```
#define _TIM1_OC4M ((uint8_t) (0x07 << 4))
```

TIM1 Output compare 4 mode [2:0] (in \_TIM1\_CCMR4)

Definition at line 3030 of file STM8AF\_STM8S.h.



**5.1.2.1048 \_TIM1\_OC4M0**

```
#define _TIM1_OC4M0 ((uint8_t) (0x01 << 4))
```

TIM1 Output compare 4 mode [0] (in \_TIM1\_CCMR4)

Definition at line 3031 of file STM8AF\_STM8S.h.

**5.1.2.1049 \_TIM1\_OC4M1**

```
#define _TIM1_OC4M1 ((uint8_t) (0x01 << 5))
```

TIM1 Output compare 4 mode [1] (in \_TIM1\_CCMR4)

Definition at line 3032 of file STM8AF\_STM8S.h.

**5.1.2.1050 \_TIM1\_OC4M2**

```
#define _TIM1_OC4M2 ((uint8_t) (0x01 << 6))
```

TIM1 Output compare 4 mode [2] (in \_TIM1\_CCMR4)

Definition at line 3033 of file STM8AF\_STM8S.h.

**5.1.2.1051 \_TIM1\_OC4PE**

```
#define _TIM1_OC4PE ((uint8_t) (0x01 << 3))
```

TIM1 Output compare 4 preload enable [0] (in \_TIM1\_CCMR4)

Definition at line 3029 of file STM8AF\_STM8S.h.

**5.1.2.1052 \_TIM1\_OIS1**

```
#define _TIM1_OIS1 ((uint8_t) (0x01 << 0))
```

TIM1 Output idle state 1 (OC1 output) [0] (in \_TIM1\_OISR)

Definition at line 3078 of file STM8AF\_STM8S.h.

#### 5.1.2.1053 \_TIM1\_OIS1N

```
#define _TIM1_OIS1N ((uint8_t) (0x01 << 1))
```

TIM1 Output idle state 1 (OC1N output) [0] (in \_TIM1\_OISR)

Definition at line 3079 of file STM8AF\_STM8S.h.

#### 5.1.2.1054 \_TIM1\_OIS2

```
#define _TIM1_OIS2 ((uint8_t) (0x01 << 2))
```

TIM1 Output idle state 2 (OC2 output) [0] (in \_TIM1\_OISR)

Definition at line 3080 of file STM8AF\_STM8S.h.

#### 5.1.2.1055 \_TIM1\_OIS2N

```
#define _TIM1_OIS2N ((uint8_t) (0x01 << 3))
```

TIM1 Output idle state 2 (OC2N output) [0] (in \_TIM1\_OISR)

Definition at line 3081 of file STM8AF\_STM8S.h.

#### 5.1.2.1056 \_TIM1\_OIS3

```
#define _TIM1_OIS3 ((uint8_t) (0x01 << 4))
```

TIM1 Output idle state 3 (OC3 output) [0] (in \_TIM1\_OISR)

Definition at line 3082 of file STM8AF\_STM8S.h.

#### 5.1.2.1057 \_TIM1\_OIS3N

```
#define _TIM1_OIS3N ((uint8_t) (0x01 << 5))
```

TIM1 Output idle state 3 (OC3N output) [0] (in \_TIM1\_OISR)

Definition at line 3083 of file STM8AF\_STM8S.h.

**5.1.2.1058 \_TIM1\_OIS4**

```
#define _TIM1_OIS4 ((uint8_t) (0x01 << 6))
```

TIM1 Output idle state 4 (OC4 output) [0] (in \_TIM1\_OISR)

Definition at line 3084 of file STM8AF\_STM8S.h.

**5.1.2.1059 \_TIM1\_OISR**

```
#define _TIM1_OISR _SFR(uint8_t, TIM1_AddressBase+0x1F)
```

TIM1 Output idle state register.

Definition at line 2835 of file STM8AF\_STM8S.h.

**5.1.2.1060 \_TIM1\_OISR\_RESET\_VALUE**

```
#define _TIM1_OISR_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Output idle state register reset value.

Definition at line 2869 of file STM8AF\_STM8S.h.

**5.1.2.1061 \_TIM1\_OPM**

```
#define _TIM1_OPM ((uint8_t) (0x01 << 3))
```

TIM1 One-pulse mode [0] (in \_TIM1\_CR1)

Definition at line 2875 of file STM8AF\_STM8S.h.

**5.1.2.1062 \_TIM1\_OSSI**

```
#define _TIM1_OSSI ((uint8_t) (0x01 << 2))
```

TIM1 Off state selection for idle mode [0] (in \_TIM1\_BKR)

Definition at line 3070 of file STM8AF\_STM8S.h.

#### 5.1.2.1063 \_TIM1\_OSSR

```
#define _TIM1_OSSR ((uint8_t) (0x01 << 3))
```

TIM1 Off state selection for Run mode [0] (in \_TIM1\_BKR)

Definition at line 3071 of file STM8AF\_STM8S.h.

#### 5.1.2.1064 \_TIM1\_PSCRH

```
#define _TIM1_PSCRH _SFR(uint8_t, TIM1_AddressBase+0x10)
```

TIM1 clock prescaler register high byte.

Definition at line 2820 of file STM8AF\_STM8S.h.

#### 5.1.2.1065 \_TIM1\_PSCRH\_RESET\_VALUE

```
#define _TIM1_PSCRH_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 clock prescaler register high byte reset value.

Definition at line 2854 of file STM8AF\_STM8S.h.

#### 5.1.2.1066 \_TIM1\_PSCRL

```
#define _TIM1_PSCRL _SFR(uint8_t, TIM1_AddressBase+0x11)
```

TIM1 clock prescaler register low byte.

Definition at line 2821 of file STM8AF\_STM8S.h.

#### 5.1.2.1067 \_TIM1\_PSCRL\_RESET\_VALUE

```
#define _TIM1_PSCRL_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 clock prescaler register low byte reset value.

Definition at line 2855 of file STM8AF\_STM8S.h.

**5.1.2.1068 \_TIM1\_RCR**

```
#define _TIM1_RCR _SFR(uint8_t, TIM1_AddressBase+0x14)
```

TIM1 Repetition counter.

Definition at line 2824 of file STM8AF\_STM8S.h.

**5.1.2.1069 \_TIM1\_RCR\_RESET\_VALUE**

```
#define _TIM1_RCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Repetition counter reset value.

Definition at line 2858 of file STM8AF\_STM8S.h.

**5.1.2.1070 \_TIM1\_SMCR**

```
#define _TIM1_SMCR _SFR(uint8_t, TIM1_AddressBase+0x02)
```

TIM1 Slave mode control register.

Definition at line 2806 of file STM8AF\_STM8S.h.

**5.1.2.1071 \_TIM1\_SMCR\_RESET\_VALUE**

```
#define _TIM1_SMCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 Slave mode control register reset value.

Definition at line 2840 of file STM8AF\_STM8S.h.

**5.1.2.1072 \_TIM1\_SMS**

```
#define _TIM1_SMS ((uint8_t) (0x07 << 0))
```

TIM1 Clock/trigger/slave mode selection [2:0] (in \_TIM1\_SMCR)

Definition at line 2894 of file STM8AF\_STM8S.h.

#### 5.1.2.1073 \_TIM1\_SMS0

```
#define _TIM1_SMS0 ((uint8_t) (0x01 << 0))
```

TIM1 Clock/trigger/slave mode selection [0] (in \_TIM1\_SMCR)

Definition at line 2895 of file STM8AF\_STM8S.h.

#### 5.1.2.1074 \_TIM1\_SMS1

```
#define _TIM1_SMS1 ((uint8_t) (0x01 << 1))
```

TIM1 Clock/trigger/slave mode selection [1] (in \_TIM1\_SMCR)

Definition at line 2896 of file STM8AF\_STM8S.h.

#### 5.1.2.1075 \_TIM1\_SMS2

```
#define _TIM1_SMS2 ((uint8_t) (0x01 << 2))
```

TIM1 Clock/trigger/slave mode selection [2] (in \_TIM1\_SMCR)

Definition at line 2897 of file STM8AF\_STM8S.h.

#### 5.1.2.1076 \_TIM1\_SR1

```
#define _TIM1_SR1 \_SFR(uint8_t, TIM1\_AddressBase+0x05)
```

TIM1 status register 1.

Definition at line 2809 of file STM8AF\_STM8S.h.

#### 5.1.2.1077 \_TIM1\_SR1\_RESET\_VALUE

```
#define _TIM1_SR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 status register 1 reset value.

Definition at line 2843 of file STM8AF\_STM8S.h.

### 5.1.2.1078 \_TIM1\_SR2

```
#define _TIM1_SR2 _SFR(uint8_t, TIM1_AddressBase+0x06)
```

TIM1 status register 2.

Definition at line 2810 of file STM8AF\_STM8S.h.

### 5.1.2.1079 \_TIM1\_SR2\_RESET\_VALUE

```
#define _TIM1_SR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM1 status register 2 reset value.

Definition at line 2844 of file STM8AF\_STM8S.h.

### 5.1.2.1080 \_TIM1\_TG

```
#define _TIM1_TG ((uint8_t) (0x01 << 6))
```

TIM1 Trigger generation [0] (in \_TIM1\_EGR)

Definition at line 2952 of file STM8AF\_STM8S.h.

### 5.1.2.1081 \_TIM1\_TIE

```
#define _TIM1_TIE ((uint8_t) (0x01 << 6))
```

TIM1 Trigger interrupt enable [0] (in \_TIM1\_IER)

Definition at line 2924 of file STM8AF\_STM8S.h.

### 5.1.2.1082 \_TIM1\_TIF

```
#define _TIM1_TIF ((uint8_t) (0x01 << 6))
```

TIM1 Trigger interrupt flag [0] (in \_TIM1\_SR1)

Definition at line 2934 of file STM8AF\_STM8S.h.

#### 5.1.2.1083 \_TIM1\_TS

```
#define _TIM1_TS ((uint8_t) (0x07 << 4))
```

TIM1 Trigger selection [2:0] (in \_TIM1\_SMCR)

Definition at line 2899 of file STM8AF\_STM8S.h.

#### 5.1.2.1084 \_TIM1\_TS0

```
#define _TIM1_TS0 ((uint8_t) (0x01 << 4))
```

TIM1 Trigger selection [0] (in \_TIM1\_SMCR)

Definition at line 2900 of file STM8AF\_STM8S.h.

#### 5.1.2.1085 \_TIM1\_TS1

```
#define _TIM1_TS1 ((uint8_t) (0x01 << 5))
```

TIM1 Trigger selection [1] (in \_TIM1\_SMCR)

Definition at line 2901 of file STM8AF\_STM8S.h.

#### 5.1.2.1086 \_TIM1\_TS2

```
#define _TIM1_TS2 ((uint8_t) (0x01 << 6))
```

TIM1 Trigger selection [2] (in \_TIM1\_SMCR)

Definition at line 2902 of file STM8AF\_STM8S.h.

#### 5.1.2.1087 \_TIM1\_UDIS

```
#define _TIM1_UDIS ((uint8_t) (0x01 << 1))
```

TIM1 Update disable [0] (in \_TIM1\_CR1)

Definition at line 2873 of file STM8AF\_STM8S.h.



**5.1.2.1088 \_TIM1\_UG**

```
#define _TIM1_UG ((uint8_t) (0x01 << 0))
```

TIM1 Update generation [0] (in \_TIM1\_EGR)

Definition at line 2946 of file STM8AF\_STM8S.h.

**5.1.2.1089 \_TIM1\_UIE**

```
#define _TIM1_UIE ((uint8_t) (0x01 << 0))
```

TIM1 Update interrupt enable [0] (in \_TIM1\_IER)

Definition at line 2918 of file STM8AF\_STM8S.h.

**5.1.2.1090 \_TIM1\_UIF**

```
#define _TIM1_UIF ((uint8_t) (0x01 << 0))
```

TIM1 Update interrupt flag [0] (in \_TIM1\_SR1)

Definition at line 2928 of file STM8AF\_STM8S.h.

**5.1.2.1091 \_TIM1\_URS**

```
#define _TIM1_URS ((uint8_t) (0x01 << 2))
```

TIM1 Update request source [0] (in \_TIM1\_CR1)

Definition at line 2874 of file STM8AF\_STM8S.h.

**5.1.2.1092 \_TIM2**

```
#define _TIM2 _SFR(TIM2_t, TIM2_AddressBase)
```

TIM2 struct/bit access.

Definition at line 3310 of file STM8AF\_STM8S.h.

#### 5.1.2.1093 \_TIM2\_ARPE

```
#define _TIM2_ARPE ((uint8_t) (0x01 << 7))
```

TIM2 Auto-reload preload enable [0] (in \_TIM2\_CR1)

Definition at line 3386 of file STM8AF\_STM8S.h.

#### 5.1.2.1094 \_TIM2\_ARRH

```
#define _TIM2_ARRH _SFR(uint8_t, TIM2_AddressBase+0x0D)
```

TIM2 auto-reload register high byte.

Definition at line 3347 of file STM8AF\_STM8S.h.

#### 5.1.2.1095 \_TIM2\_ARRH\_RESET\_VALUE

```
#define _TIM2_ARRH_RESET_VALUE ((uint8_t) 0xFF)
```

TIM2 auto-reload register high byte reset value.

Definition at line 3371 of file STM8AF\_STM8S.h.

#### 5.1.2.1096 \_TIM2\_ARRL

```
#define _TIM2_ARRL _SFR(uint8_t, TIM2_AddressBase+0x0E)
```

TIM2 auto-reload register low byte.

Definition at line 3348 of file STM8AF\_STM8S.h.

#### 5.1.2.1097 \_TIM2\_ARRL\_RESET\_VALUE

```
#define _TIM2_ARRL_RESET_VALUE ((uint8_t) 0xFF)
```

TIM2 auto-reload register low byte reset value.

Definition at line 3372 of file STM8AF\_STM8S.h.

**5.1.2.1098 \_TIM2\_CC1E**

```
#define _TIM2_CC1E ((uint8_t) (0x01 << 0))
```

TIM2 Capture/compare 1 output enable [0] (in \_TIM2\_CCER1)

Definition at line 3486 of file STM8AF\_STM8S.h.

**5.1.2.1099 \_TIM2\_CC1G**

```
#define _TIM2_CC1G ((uint8_t) (0x01 << 1))
```

TIM2 Capture/compare 1 generation [0] (in \_TIM2\_EGR)

Definition at line 3411 of file STM8AF\_STM8S.h.

**5.1.2.1100 \_TIM2\_CC1IE**

```
#define _TIM2_CC1IE ((uint8_t) (0x01 << 1))
```

TIM2 Capture/compare 1 interrupt enable [0] (in \_TIM2\_IER)

Definition at line 3390 of file STM8AF\_STM8S.h.

**5.1.2.1101 \_TIM2\_CC1IF**

```
#define _TIM2_CC1IF ((uint8_t) (0x01 << 1))
```

TIM2 Capture/compare 1 interrupt flag [0] (in \_TIM2\_SR1)

Definition at line 3397 of file STM8AF\_STM8S.h.

**5.1.2.1102 \_TIM2\_CC1OF**

```
#define _TIM2_CC1OF ((uint8_t) (0x01 << 1))
```

TIM2 Capture/compare 1 overcapture flag [0] (in \_TIM2\_SR2)

Definition at line 3404 of file STM8AF\_STM8S.h.

#### 5.1.2.1103 \_TIM2\_CC1P

```
#define _TIM2_CC1P ((uint8_t) (0x01 << 1))
```

TIM2 Capture/compare 1 output polarity [0] (in \_TIM2\_CCER1)

Definition at line 3487 of file STM8AF\_STM8S.h.

#### 5.1.2.1104 \_TIM2\_CC1S

```
#define _TIM2_CC1S ((uint8_t) (0x03 << 0))
```

TIM2 Compare 1 selection [1:0] (in \_TIM2\_CCMR1)

Definition at line 3417 of file STM8AF\_STM8S.h.

#### 5.1.2.1105 \_TIM2\_CC1S0

```
#define _TIM2_CC1S0 ((uint8_t) (0x01 << 0))
```

TIM2 Compare 1 selection [0] (in \_TIM2\_CCMR1)

Definition at line 3418 of file STM8AF\_STM8S.h.

#### 5.1.2.1106 \_TIM2\_CC1S1

```
#define _TIM2_CC1S1 ((uint8_t) (0x01 << 1))
```

TIM2 Compare 1 selection [1] (in \_TIM2\_CCMR1)

Definition at line 3419 of file STM8AF\_STM8S.h.

#### 5.1.2.1107 \_TIM2\_CC2E

```
#define _TIM2_CC2E ((uint8_t) (0x01 << 4))
```

TIM2 Capture/compare 2 output enable [0] (in \_TIM2\_CCER1)

Definition at line 3489 of file STM8AF\_STM8S.h.

**5.1.2.1108 \_TIM2\_CC2G**

```
#define _TIM2_CC2G ((uint8_t) (0x01 << 2))
```

TIM2 Capture/compare 2 generation [0] (in \_TIM2\_EGR)

Definition at line 3412 of file STM8AF\_STM8S.h.

**5.1.2.1109 \_TIM2\_CC2IE**

```
#define _TIM2_CC2IE ((uint8_t) (0x01 << 2))
```

TIM2 Capture/compare 2 interrupt enable [0] (in \_TIM2\_IER)

Definition at line 3391 of file STM8AF\_STM8S.h.

**5.1.2.1110 \_TIM2\_CC2IF**

```
#define _TIM2_CC2IF ((uint8_t) (0x01 << 2))
```

TIM2 Capture/compare 2 interrupt flag [0] (in \_TIM2\_SR1)

Definition at line 3398 of file STM8AF\_STM8S.h.

**5.1.2.1111 \_TIM2\_CC2OF**

```
#define _TIM2_CC2OF ((uint8_t) (0x01 << 2))
```

TIM2 Capture/compare 2 overcapture flag [0] (in \_TIM2\_SR2)

Definition at line 3405 of file STM8AF\_STM8S.h.

**5.1.2.1112 \_TIM2\_CC2P**

```
#define _TIM2_CC2P ((uint8_t) (0x01 << 5))
```

TIM2 Capture/compare 2 output polarity [0] (in \_TIM2\_CCER1)

Definition at line 3490 of file STM8AF\_STM8S.h.

**5.1.2.1113 \_TIM2\_CC2S**

```
#define _TIM2_CC2S ((uint8_t) (0x03 << 0))
```

TIM2 Compare 2 selection [1:0] (in \_TIM2\_CCMR2)

Definition at line 3440 of file STM8AF\_STM8S.h.

**5.1.2.1114 \_TIM2\_CC2S0**

```
#define _TIM2_CC2S0 ((uint8_t) (0x01 << 0))
```

TIM2 Compare 2 selection [0] (in \_TIM2\_CCMR2)

Definition at line 3441 of file STM8AF\_STM8S.h.

**5.1.2.1115 \_TIM2\_CC2S1**

```
#define _TIM2_CC2S1 ((uint8_t) (0x01 << 1))
```

TIM2 Compare 2 selection [1] (in \_TIM2\_CCMR2)

Definition at line 3442 of file STM8AF\_STM8S.h.

**5.1.2.1116 \_TIM2\_CC3E**

```
#define _TIM2_CC3E ((uint8_t) (0x01 << 0))
```

TIM2 Capture/compare 3 output enable [0] (in \_TIM2\_CCER2)

Definition at line 3494 of file STM8AF\_STM8S.h.

**5.1.2.1117 \_TIM2\_CC3G**

```
#define _TIM2_CC3G ((uint8_t) (0x01 << 3))
```

TIM2 Capture/compare 3 generation [0] (in \_TIM2\_EGR)

Definition at line 3413 of file STM8AF\_STM8S.h.

**5.1.2.1118 \_TIM2\_CC3IE**

```
#define _TIM2_CC3IE ((uint8_t) (0x01 << 3))
```

TIM2 Capture/compare 3 interrupt enable [0] (in \_TIM2\_IER)

Definition at line 3392 of file STM8AF\_STM8S.h.

**5.1.2.1119 \_TIM2\_CC3IF**

```
#define _TIM2_CC3IF ((uint8_t) (0x01 << 3))
```

TIM2 Capture/compare 3 interrupt flag [0] (in \_TIM2\_SR1)

Definition at line 3399 of file STM8AF\_STM8S.h.

**5.1.2.1120 \_TIM2\_CC3OF**

```
#define _TIM2_CC3OF ((uint8_t) (0x01 << 3))
```

TIM2 Capture/compare 3 overcapture flag [0] (in \_TIM2\_SR2)

Definition at line 3406 of file STM8AF\_STM8S.h.

**5.1.2.1121 \_TIM2\_CC3P**

```
#define _TIM2_CC3P ((uint8_t) (0x01 << 1))
```

TIM2 Capture/compare 3 output polarity [0] (in \_TIM2\_CCER2)

Definition at line 3495 of file STM8AF\_STM8S.h.

**5.1.2.1122 \_TIM2\_CC3S**

```
#define _TIM2_CC3S ((uint8_t) (0x03 << 0))
```

TIM2 Compare 3 selection [1:0] (in \_TIM2\_CCMR3)

Definition at line 3463 of file STM8AF\_STM8S.h.

#### 5.1.2.1123 \_TIM2\_CC3S0

```
#define _TIM2_CC3S0 ((uint8_t) (0x01 << 0))
```

TIM2 Compare 3 selection [0] (in \_TIM2\_CCMR3)

Definition at line 3464 of file STM8AF\_STM8S.h.

#### 5.1.2.1124 \_TIM2\_CC3S1

```
#define _TIM2_CC3S1 ((uint8_t) (0x01 << 1))
```

TIM2 Compare 3 selection [1] (in \_TIM2\_CCMR3)

Definition at line 3465 of file STM8AF\_STM8S.h.

#### 5.1.2.1125 \_TIM2\_CCER1

```
#define _TIM2_CCER1 _SFR(uint8_t, TIM2_AddressBase+0x08)
```

TIM2 Capture/compare enable register 1.

Definition at line 3342 of file STM8AF\_STM8S.h.

#### 5.1.2.1126 \_TIM2\_CCER1\_RESET\_VALUE

```
#define _TIM2_CCER1_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 Capture/compare enable register 1 reset value.

Definition at line 3366 of file STM8AF\_STM8S.h.

#### 5.1.2.1127 \_TIM2\_CCER2

```
#define _TIM2_CCER2 _SFR(uint8_t, TIM2_AddressBase+0x09)
```

TIM2 Capture/compare enable register 2.

Definition at line 3343 of file STM8AF\_STM8S.h.



**5.1.2.1128 \_TIM2\_CCER2\_RESET\_VALUE**

```
#define _TIM2_CCER2_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 Capture/compare enable register 2 reset value.

Definition at line 3367 of file STM8AF\_STM8S.h.

**5.1.2.1129 \_TIM2\_CCMR1**

```
#define _TIM2_CCMR1 _SFR(uint8_t, TIM2_AddressBase+0x05)
```

TIM2 Capture/compare mode register 1.

Definition at line 3339 of file STM8AF\_STM8S.h.

**5.1.2.1130 \_TIM2\_CCMR1\_RESET\_VALUE**

```
#define _TIM2_CCMR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 Capture/compare mode register 1 reset value.

Definition at line 3363 of file STM8AF\_STM8S.h.

**5.1.2.1131 \_TIM2\_CCMR2**

```
#define _TIM2_CCMR2 _SFR(uint8_t, TIM2_AddressBase+0x06)
```

TIM2 Capture/compare mode register 2.

Definition at line 3340 of file STM8AF\_STM8S.h.

**5.1.2.1132 \_TIM2\_CCMR2\_RESET\_VALUE**

```
#define _TIM2_CCMR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 Capture/compare mode register 2 reset value.

Definition at line 3364 of file STM8AF\_STM8S.h.

#### 5.1.2.1133 \_TIM2\_CCMR3

```
#define _TIM2_CCMR3 _SFR(uint8_t, TIM2_AddressBase+0x07)
```

TIM2 Capture/compare mode register 3.

Definition at line 3341 of file STM8AF\_STM8S.h.

#### 5.1.2.1134 \_TIM2\_CCMR3\_RESET\_VALUE

```
#define _TIM2_CCMR3_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 Capture/compare mode register 3 reset value.

Definition at line 3365 of file STM8AF\_STM8S.h.

#### 5.1.2.1135 \_TIM2\_CCR1H

```
#define _TIM2_CCR1H _SFR(uint8_t, TIM2_AddressBase+0x0F)
```

TIM2 16-bit capture/compare value 1 high byte.

Definition at line 3349 of file STM8AF\_STM8S.h.

#### 5.1.2.1136 \_TIM2\_CCR1H\_RESET\_VALUE

```
#define _TIM2_CCR1H_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 16-bit capture/compare value 1 high byte reset value.

Definition at line 3373 of file STM8AF\_STM8S.h.

#### 5.1.2.1137 \_TIM2\_CCR1L

```
#define _TIM2_CCR1L _SFR(uint8_t, TIM2_AddressBase+0x10)
```

TIM2 16-bit capture/compare value 1 low byte.

Definition at line 3350 of file STM8AF\_STM8S.h.

**5.1.2.1138 \_TIM2\_CCR1L\_RESET\_VALUE**

```
#define _TIM2_CCR1L_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 16-bit capture/compare value 1 low byte reset value.

Definition at line 3374 of file STM8AF\_STM8S.h.

**5.1.2.1139 \_TIM2\_CCR2H**

```
#define _TIM2_CCR2H _SFR(uint8_t, TIM2_AddressBase+0x11)
```

TIM2 16-bit capture/compare value 2 high byte.

Definition at line 3351 of file STM8AF\_STM8S.h.

**5.1.2.1140 \_TIM2\_CCR2H\_RESET\_VALUE**

```
#define _TIM2_CCR2H_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 16-bit capture/compare value 2 high byte reset value.

Definition at line 3375 of file STM8AF\_STM8S.h.

**5.1.2.1141 \_TIM2\_CCR2L**

```
#define _TIM2_CCR2L _SFR(uint8_t, TIM2_AddressBase+0x12)
```

TIM2 16-bit capture/compare value 2 low byte.

Definition at line 3352 of file STM8AF\_STM8S.h.

**5.1.2.1142 \_TIM2\_CCR2L\_RESET\_VALUE**

```
#define _TIM2_CCR2L_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 16-bit capture/compare value 2 low byte reset value.

Definition at line 3376 of file STM8AF\_STM8S.h.

#### 5.1.2.1143 \_TIM2\_CCR3H

```
#define _TIM2_CCR3H _SFR(uint8_t, TIM2_AddressBase+0x13)
```

TIM2 16-bit capture/compare value 3 high byte.

Definition at line 3353 of file STM8AF\_STM8S.h.

#### 5.1.2.1144 \_TIM2\_CCR3H\_RESET\_VALUE

```
#define _TIM2_CCR3H_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 16-bit capture/compare value 3 high byte reset value.

Definition at line 3377 of file STM8AF\_STM8S.h.

#### 5.1.2.1145 \_TIM2\_CCR3L

```
#define _TIM2_CCR3L _SFR(uint8_t, TIM2_AddressBase+0x14)
```

TIM2 16-bit capture/compare value 3 low byte.

Definition at line 3354 of file STM8AF\_STM8S.h.

#### 5.1.2.1146 \_TIM2\_CCR3L\_RESET\_VALUE

```
#define _TIM2_CCR3L_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 16-bit capture/compare value 3 low byte reset value.

Definition at line 3378 of file STM8AF\_STM8S.h.

#### 5.1.2.1147 \_TIM2\_CEN

```
#define _TIM2_CEN ((uint8_t) (0x01 << 0))
```

TIM2 Counter enable [0] (in \_TIM2\_CR1)

Definition at line 3381 of file STM8AF\_STM8S.h.

**5.1.2.1148 \_TIM2\_CNTRH**

```
#define _TIM2_CNTRH _SFR(uint8_t, TIM2_AddressBase+0x0A)
```

TIM2 counter register high byte.

Definition at line 3344 of file STM8AF\_STM8S.h.

**5.1.2.1149 \_TIM2\_CNTRH\_RESET\_VALUE**

```
#define _TIM2_CNTRH_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 counter register high byte reset value.

Definition at line 3368 of file STM8AF\_STM8S.h.

**5.1.2.1150 \_TIM2\_CNTRL**

```
#define _TIM2_CNTRL _SFR(uint8_t, TIM2_AddressBase+0x0B)
```

TIM2 counter register low byte.

Definition at line 3345 of file STM8AF\_STM8S.h.

**5.1.2.1151 \_TIM2\_CNTRL\_RESET\_VALUE**

```
#define _TIM2_CNTRL_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 counter register low byte reset value.

Definition at line 3369 of file STM8AF\_STM8S.h.

**5.1.2.1152 \_TIM2\_CR1**

```
#define _TIM2_CR1 _SFR(uint8_t, TIM2_AddressBase+0x00)
```

TIM2 control register 1.

Definition at line 3311 of file STM8AF\_STM8S.h.

#### 5.1.2.1153 \_TIM2\_CR1\_RESET\_VALUE

```
#define _TIM2_CR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 control register 1 reset value.

Definition at line 3358 of file STM8AF\_STM8S.h.

#### 5.1.2.1154 \_TIM2\_EGR

```
#define _TIM2_EGR _SFR(uint8_t, TIM2_AddressBase+0x04)
```

TIM2 Event generation register.

Definition at line 3338 of file STM8AF\_STM8S.h.

#### 5.1.2.1155 \_TIM2\_EGR\_RESET\_VALUE

```
#define _TIM2_EGR_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 Event generation register reset value.

Definition at line 3362 of file STM8AF\_STM8S.h.

#### 5.1.2.1156 \_TIM2\_IC1F

```
#define _TIM2_IC1F ((uint8_t) (0x0F << 4))
```

TIM2 Output compare 1 mode [3:0] (in \_TIM2\_CCMR1)

Definition at line 3433 of file STM8AF\_STM8S.h.

#### 5.1.2.1157 \_TIM2\_IC1F0

```
#define _TIM2_IC1F0 ((uint8_t) (0x01 << 4))
```

TIM2 Output compare 1 mode [0] (in \_TIM2\_CCMR1)

Definition at line 3434 of file STM8AF\_STM8S.h.

**5.1.2.1158 \_TIM2\_IC1F1**

```
#define _TIM2_IC1F1 ((uint8_t) (0x01 << 5))
```

TIM2 Output compare 1 mode [1] (in \_TIM2\_CCMR1)

Definition at line 3435 of file STM8AF\_STM8S.h.

**5.1.2.1159 \_TIM2\_IC1F2**

```
#define _TIM2_IC1F2 ((uint8_t) (0x01 << 6))
```

TIM2 Output compare 1 mode [2] (in \_TIM2\_CCMR1)

Definition at line 3436 of file STM8AF\_STM8S.h.

**5.1.2.1160 \_TIM2\_IC1F3**

```
#define _TIM2_IC1F3 ((uint8_t) (0x01 << 7))
```

TIM2 Output compare 1 mode [3] (in \_TIM2\_CCMR1)

Definition at line 3437 of file STM8AF\_STM8S.h.

**5.1.2.1161 \_TIM2\_IC1PSC**

```
#define _TIM2_IC1PSC ((uint8_t) (0x03 << 2))
```

TIM2 Input capture 1 prescaler [1:0] (in \_TIM2\_CCMR1)

Definition at line 3430 of file STM8AF\_STM8S.h.

**5.1.2.1162 \_TIM2\_IC1PSC0**

```
#define _TIM2_IC1PSC0 ((uint8_t) (0x01 << 2))
```

TIM2 Input capture 1 prescaler [0] (in \_TIM2\_CCMR1)

Definition at line 3431 of file STM8AF\_STM8S.h.

#### 5.1.2.1163 \_TIM2\_IC1PSC1

```
#define _TIM2_IC1PSC1 ((uint8_t) (0x01 << 3))
```

TIM2 Input capture 1 prescaler [1] (in \_TIM2\_CCMR1)

Definition at line 3432 of file STM8AF\_STM8S.h.

#### 5.1.2.1164 \_TIM2\_IC2F

```
#define _TIM2_IC2F ((uint8_t) (0x0F << 4))
```

TIM2 Output compare 2 mode [3:0] (in \_TIM2\_CCMR2)

Definition at line 3456 of file STM8AF\_STM8S.h.

#### 5.1.2.1165 \_TIM2\_IC2F0

```
#define _TIM2_IC2F0 ((uint8_t) (0x01 << 4))
```

TIM2 Output compare 2 mode [0] (in \_TIM2\_CCMR2)

Definition at line 3457 of file STM8AF\_STM8S.h.

#### 5.1.2.1166 \_TIM2\_IC2F1

```
#define _TIM2_IC2F1 ((uint8_t) (0x01 << 5))
```

TIM2 Output compare 2 mode [1] (in \_TIM2\_CCMR2)

Definition at line 3458 of file STM8AF\_STM8S.h.

#### 5.1.2.1167 \_TIM2\_IC2F2

```
#define _TIM2_IC2F2 ((uint8_t) (0x01 << 6))
```

TIM2 Output compare 2 mode [2] (in \_TIM2\_CCMR2)

Definition at line 3459 of file STM8AF\_STM8S.h.



**5.1.2.1168 \_TIM2\_IC2F3**

```
#define _TIM2_IC2F3 ((uint8_t) (0x01 << 7))
```

TIM2 Output compare 2 mode [3] (in \_TIM2\_CCMR2)

Definition at line 3460 of file STM8AF\_STM8S.h.

**5.1.2.1169 \_TIM2\_IC2PSC**

```
#define _TIM2_IC2PSC ((uint8_t) (0x03 << 2))
```

TIM2 Input capture 2 prescaler [1:0] (in \_TIM2\_CCMR2)

Definition at line 3453 of file STM8AF\_STM8S.h.

**5.1.2.1170 \_TIM2\_IC2PSC0**

```
#define _TIM2_IC2PSC0 ((uint8_t) (0x01 << 2))
```

TIM2 Input capture 2 prescaler [0] (in \_TIM2\_CCMR2)

Definition at line 3454 of file STM8AF\_STM8S.h.

**5.1.2.1171 \_TIM2\_IC2PSC1**

```
#define _TIM2_IC2PSC1 ((uint8_t) (0x01 << 3))
```

TIM2 Input capture 2 prescaler [1] (in \_TIM2\_CCMR2)

Definition at line 3455 of file STM8AF\_STM8S.h.

**5.1.2.1172 \_TIM2\_IC3F**

```
#define _TIM2_IC3F ((uint8_t) (0x0F << 4))
```

TIM2 Output compare 3 mode [3:0] (in \_TIM2\_CCMR3)

Definition at line 3479 of file STM8AF\_STM8S.h.

**5.1.2.1173 \_TIM2\_IC3F0**

```
#define _TIM2_IC3F0 ((uint8_t) (0x01 << 4))
```

TIM2 Output compare 3 mode [0] (in \_TIM2\_CCMR3)

Definition at line 3480 of file STM8AF\_STM8S.h.

**5.1.2.1174 \_TIM2\_IC3F1**

```
#define _TIM2_IC3F1 ((uint8_t) (0x01 << 5))
```

TIM2 Output compare 3 mode [1] (in \_TIM2\_CCMR3)

Definition at line 3481 of file STM8AF\_STM8S.h.

**5.1.2.1175 \_TIM2\_IC3F2**

```
#define _TIM2_IC3F2 ((uint8_t) (0x01 << 6))
```

TIM2 Output compare 3 mode [2] (in \_TIM2\_CCMR3)

Definition at line 3482 of file STM8AF\_STM8S.h.

**5.1.2.1176 \_TIM2\_IC3F3**

```
#define _TIM2_IC3F3 ((uint8_t) (0x01 << 7))
```

TIM2 Output compare 3 mode [3] (in \_TIM2\_CCMR3)

Definition at line 3483 of file STM8AF\_STM8S.h.

**5.1.2.1177 \_TIM2\_IC3PSC**

```
#define _TIM2_IC3PSC ((uint8_t) (0x03 << 2))
```

TIM2 Input capture 3 prescaler [1:0] (in \_TIM2\_CCMR3)

Definition at line 3476 of file STM8AF\_STM8S.h.

**5.1.2.1178 \_TIM2\_IC3PSC0**

```
#define _TIM2_IC3PSC0 ((uint8_t) (0x01 << 2))
```

TIM2 Input capture 3 prescaler [0] (in \_TIM2\_CCMR3)

Definition at line 3477 of file STM8AF\_STM8S.h.

**5.1.2.1179 \_TIM2\_IC3PSC1**

```
#define _TIM2_IC3PSC1 ((uint8_t) (0x01 << 3))
```

TIM2 Input capture 3 prescaler [1] (in \_TIM2\_CCMR3)

Definition at line 3478 of file STM8AF\_STM8S.h.

**5.1.2.1180 \_TIM2\_IER**

```
#define _TIM2_IER _SFR(uint8_t, TIM2_AddressBase+0x01)
```

TIM2 interrupt enable register.

Definition at line 3335 of file STM8AF\_STM8S.h.

**5.1.2.1181 \_TIM2\_IER\_RESET\_VALUE**

```
#define _TIM2_IER_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 interrupt enable register reset value.

Definition at line 3359 of file STM8AF\_STM8S.h.

**5.1.2.1182 \_TIM2\_OC1M**

```
#define _TIM2_OC1M ((uint8_t) (0x07 << 4))
```

TIM2 Output compare 1 mode [2:0] (in \_TIM2\_CCMR1)

Definition at line 3422 of file STM8AF\_STM8S.h.

**5.1.2.1183 \_TIM2\_OC1M0**

```
#define _TIM2_OC1M0 ((uint8_t) (0x01 << 4))
```

TIM2 Output compare 1 mode [0] (in \_TIM2\_CCMR1)

Definition at line 3423 of file STM8AF\_STM8S.h.

**5.1.2.1184 \_TIM2\_OC1M1**

```
#define _TIM2_OC1M1 ((uint8_t) (0x01 << 5))
```

TIM2 Output compare 1 mode [1] (in \_TIM2\_CCMR1)

Definition at line 3424 of file STM8AF\_STM8S.h.

**5.1.2.1185 \_TIM2\_OC1M2**

```
#define _TIM2_OC1M2 ((uint8_t) (0x01 << 6))
```

TIM2 Output compare 1 mode [2] (in \_TIM2\_CCMR1)

Definition at line 3425 of file STM8AF\_STM8S.h.

**5.1.2.1186 \_TIM2\_OC1PE**

```
#define _TIM2_OC1PE ((uint8_t) (0x01 << 3))
```

TIM2 Output compare 1 preload enable [0] (in \_TIM2\_CCMR1)

Definition at line 3421 of file STM8AF\_STM8S.h.

**5.1.2.1187 \_TIM2\_OC2M**

```
#define _TIM2_OC2M ((uint8_t) (0x07 << 4))
```

TIM2 Output compare 2 mode [2:0] (in \_TIM2\_CCMR2)

Definition at line 3445 of file STM8AF\_STM8S.h.

**5.1.2.1188 \_TIM2\_OC2M0**

```
#define _TIM2_OC2M0 ((uint8_t) (0x01 << 4))
```

TIM2 Output compare 2 mode [0] (in \_TIM2\_CCMR2)

Definition at line 3446 of file STM8AF\_STM8S.h.

**5.1.2.1189 \_TIM2\_OC2M1**

```
#define _TIM2_OC2M1 ((uint8_t) (0x01 << 5))
```

TIM2 Output compare 2 mode [1] (in \_TIM2\_CCMR2)

Definition at line 3447 of file STM8AF\_STM8S.h.

**5.1.2.1190 \_TIM2\_OC2M2**

```
#define _TIM2_OC2M2 ((uint8_t) (0x01 << 6))
```

TIM2 Output compare 2 mode [2] (in \_TIM2\_CCMR2)

Definition at line 3448 of file STM8AF\_STM8S.h.

**5.1.2.1191 \_TIM2\_OC2PE**

```
#define _TIM2_OC2PE ((uint8_t) (0x01 << 3))
```

TIM2 Output compare 2 preload enable [0] (in \_TIM2\_CCMR2)

Definition at line 3444 of file STM8AF\_STM8S.h.

**5.1.2.1192 \_TIM2\_OC3M**

```
#define _TIM2_OC3M ((uint8_t) (0x07 << 4))
```

TIM2 Output compare 3 mode [2:0] (in \_TIM2\_CCMR3)

Definition at line 3468 of file STM8AF\_STM8S.h.

**5.1.2.1193 \_TIM2\_OC3M0**

```
#define _TIM2_OC3M0 ((uint8_t) (0x01 << 4))
```

TIM2 Output compare 3 mode [0] (in \_TIM2\_CCMR3)

Definition at line 3469 of file STM8AF\_STM8S.h.

**5.1.2.1194 \_TIM2\_OC3M1**

```
#define _TIM2_OC3M1 ((uint8_t) (0x01 << 5))
```

TIM2 Output compare 3 mode [1] (in \_TIM2\_CCMR3)

Definition at line 3470 of file STM8AF\_STM8S.h.

**5.1.2.1195 \_TIM2\_OC3M2**

```
#define _TIM2_OC3M2 ((uint8_t) (0x01 << 6))
```

TIM2 Output compare 3 mode [2] (in \_TIM2\_CCMR3)

Definition at line 3471 of file STM8AF\_STM8S.h.

**5.1.2.1196 \_TIM2\_OC3PE**

```
#define _TIM2_OC3PE ((uint8_t) (0x01 << 3))
```

TIM2 Output compare 3 preload enable [0] (in \_TIM2\_CCMR3)

Definition at line 3467 of file STM8AF\_STM8S.h.

**5.1.2.1197 \_TIM2\_OPM**

```
#define _TIM2_OPM ((uint8_t) (0x01 << 3))
```

TIM2 One-pulse mode [0] (in \_TIM2\_CR1)

Definition at line 3384 of file STM8AF\_STM8S.h.

**5.1.2.1198 \_TIM2\_PSC**

```
#define _TIM2_PSC ((uint8_t) (0x0F << 0))
```

TIM2 prescaler [3:0] (in \_TIM2\_PSCR)

Definition at line 3499 of file STM8AF\_STM8S.h.

**5.1.2.1199 \_TIM2\_PSC0**

```
#define _TIM2_PSC0 ((uint8_t) (0x01 << 0))
```

TIM2 prescaler [0] (in \_TIM2\_PSCR)

Definition at line 3500 of file STM8AF\_STM8S.h.

**5.1.2.1200 \_TIM2\_PSC1**

```
#define _TIM2_PSC1 ((uint8_t) (0x01 << 1))
```

TIM2 prescaler [1] (in \_TIM2\_PSCR)

Definition at line 3501 of file STM8AF\_STM8S.h.

**5.1.2.1201 \_TIM2\_PSC2**

```
#define _TIM2_PSC2 ((uint8_t) (0x01 << 2))
```

TIM2 prescaler [2] (in \_TIM2\_PSCR)

Definition at line 3502 of file STM8AF\_STM8S.h.

**5.1.2.1202 \_TIM2\_PSC3**

```
#define _TIM2_PSC3 ((uint8_t) (0x01 << 3))
```

TIM2 prescaler [3] (in \_TIM2\_PSCR)

Definition at line 3503 of file STM8AF\_STM8S.h.

#### 5.1.2.1203 \_TIM2\_PSCR

```
#define _TIM2_PSCR _SFR(uint8_t, TIM2_AddressBase+0x0C)
```

TIM2 clock prescaler register.

Definition at line 3346 of file STM8AF\_STM8S.h.

#### 5.1.2.1204 \_TIM2\_PSCR\_RESET\_VALUE

```
#define _TIM2_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 clock prescaler register reset value.

Definition at line 3370 of file STM8AF\_STM8S.h.

#### 5.1.2.1205 \_TIM2\_SR1

```
#define _TIM2_SR1 _SFR(uint8_t, TIM2_AddressBase+0x02)
```

TIM2 status register 1.

Definition at line 3336 of file STM8AF\_STM8S.h.

#### 5.1.2.1206 \_TIM2\_SR1\_RESET\_VALUE

```
#define _TIM2_SR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 status register 1 reset value.

Definition at line 3360 of file STM8AF\_STM8S.h.

#### 5.1.2.1207 \_TIM2\_SR2

```
#define _TIM2_SR2 _SFR(uint8_t, TIM2_AddressBase+0x03)
```

TIM2 status register 2.

Definition at line 3337 of file STM8AF\_STM8S.h.



#### 5.1.2.1208 \_TIM2\_SR2\_RESET\_VALUE

```
#define _TIM2_SR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM2 status register 2 reset value.

Definition at line 3361 of file STM8AF\_STM8S.h.

#### 5.1.2.1209 \_TIM2\_UDIS

```
#define _TIM2_UDIS ((uint8_t) (0x01 << 1))
```

TIM2 Update disable [0] (in \_TIM2\_CR1)

Definition at line 3382 of file STM8AF\_STM8S.h.

#### 5.1.2.1210 \_TIM2\_UG

```
#define _TIM2_UG ((uint8_t) (0x01 << 0))
```

TIM2 Update generation [0] (in \_TIM2\_EGR)

Definition at line 3410 of file STM8AF\_STM8S.h.

#### 5.1.2.1211 \_TIM2\_UIE

```
#define _TIM2_UIE ((uint8_t) (0x01 << 0))
```

TIM2 Update interrupt enable [0] (in \_TIM2\_IER)

Definition at line 3389 of file STM8AF\_STM8S.h.

#### 5.1.2.1212 \_TIM2\_UIF

```
#define _TIM2_UIF ((uint8_t) (0x01 << 0))
```

TIM2 Update interrupt flag [0] (in \_TIM2\_SR1)

Definition at line 3396 of file STM8AF\_STM8S.h.

#### 5.1.2.1213 \_TIM2\_URS

```
#define _TIM2_URS ((uint8_t) (0x01 << 2))
```

TIM2 Update request source [0] (in \_TIM2\_CR1)

Definition at line 3383 of file STM8AF\_STM8S.h.

#### 5.1.2.1214 \_TIM3

```
#define _TIM3 _SFR(TIM3_t, TIM3_AddressBase)
```

TIM3 struct/bit access.

Definition at line 3677 of file STM8AF\_STM8S.h.

#### 5.1.2.1215 \_TIM3\_ARPE

```
#define _TIM3_ARPE ((uint8_t) (0x01 << 7))
```

TIM3 Auto-reload preload enable [0] (in \_TIM3\_CR1)

Definition at line 3721 of file STM8AF\_STM8S.h.

#### 5.1.2.1216 \_TIM3\_ARRH

```
#define _TIM3_ARRH _SFR(uint8_t, TIM3_AddressBase+0x0D)
```

TIM3 auto-reload register high byte.

Definition at line 3689 of file STM8AF\_STM8S.h.

#### 5.1.2.1217 \_TIM3\_ARRH\_RESET\_VALUE

```
#define _TIM3_ARRH_RESET_VALUE ((uint8_t) 0xFF)
```

TIM3 auto-reload register high byte reset value.

Definition at line 3708 of file STM8AF\_STM8S.h.

**5.1.2.1218 \_TIM3\_ARRL**

```
#define _TIM3_ARRL _SFR(uint8_t, TIM3_AddressBase+0x0E)
```

TIM3 auto-reload register low byte.

Definition at line 3690 of file STM8AF\_STM8S.h.

**5.1.2.1219 \_TIM3\_ARRL\_RESET\_VALUE**

```
#define _TIM3_ARRL_RESET_VALUE ((uint8_t) 0xFF)
```

TIM3 auto-reload register low byte reset value.

Definition at line 3709 of file STM8AF\_STM8S.h.

**5.1.2.1220 \_TIM3\_CC1E**

```
#define _TIM3_CC1E ((uint8_t) (0x01 << 0))
```

TIM3 Capture/compare 1 output enable [0] (in \_TIM3\_CCER1)

Definition at line 3794 of file STM8AF\_STM8S.h.

**5.1.2.1221 \_TIM3\_CC1G**

```
#define _TIM3_CC1G ((uint8_t) (0x01 << 1))
```

TIM3 Capture/compare 1 generation [0] (in \_TIM3\_EGR)

Definition at line 3743 of file STM8AF\_STM8S.h.

**5.1.2.1222 \_TIM3\_CC1IE**

```
#define _TIM3_CC1IE ((uint8_t) (0x01 << 1))
```

TIM3 Capture/compare 1 interrupt enable [0] (in \_TIM3\_IER)

Definition at line 3725 of file STM8AF\_STM8S.h.

**5.1.2.1223 \_TIM3\_CC1IF**

```
#define _TIM3_CC1IF ((uint8_t) (0x01 << 1))
```

TIM3 Capture/compare 1 interrupt flag [0] (in \_TIM3\_SR1)

Definition at line 3731 of file STM8AF\_STM8S.h.

**5.1.2.1224 \_TIM3\_CC1OF**

```
#define _TIM3_CC1OF ((uint8_t) (0x01 << 1))
```

TIM3 Capture/compare 1 overcapture flag [0] (in \_TIM3\_SR2)

Definition at line 3737 of file STM8AF\_STM8S.h.

**5.1.2.1225 \_TIM3\_CC1P**

```
#define _TIM3_CC1P ((uint8_t) (0x01 << 1))
```

TIM3 Capture/compare 1 output polarity [0] (in \_TIM3\_CCER1)

Definition at line 3795 of file STM8AF\_STM8S.h.

**5.1.2.1226 \_TIM3\_CC1S**

```
#define _TIM3_CC1S ((uint8_t) (0x03 << 0))
```

TIM3 Compare 1 selection [1:0] (in \_TIM3\_CCMR1)

Definition at line 3748 of file STM8AF\_STM8S.h.

**5.1.2.1227 \_TIM3\_CC1S0**

```
#define _TIM3_CC1S0 ((uint8_t) (0x01 << 0))
```

TIM3 Compare 1 selection [0] (in \_TIM3\_CCMR1)

Definition at line 3749 of file STM8AF\_STM8S.h.

**5.1.2.1228 \_TIM3\_CC1S1**

```
#define _TIM3_CC1S1 ((uint8_t) (0x01 << 1))
```

TIM3 Compare 1 selection [1] (in \_TIM3\_CCMR1)

Definition at line 3750 of file STM8AF\_STM8S.h.

**5.1.2.1229 \_TIM3\_CC2E**

```
#define _TIM3_CC2E ((uint8_t) (0x01 << 4))
```

TIM3 Capture/compare 2 output enable [0] (in \_TIM3\_CCER1)

Definition at line 3797 of file STM8AF\_STM8S.h.

**5.1.2.1230 \_TIM3\_CC2G**

```
#define _TIM3_CC2G ((uint8_t) (0x01 << 2))
```

TIM3 Capture/compare 2 generation [0] (in \_TIM3\_EGR)

Definition at line 3744 of file STM8AF\_STM8S.h.

**5.1.2.1231 \_TIM3\_CC2IE**

```
#define _TIM3_CC2IE ((uint8_t) (0x01 << 2))
```

TIM3 Capture/compare 2 interrupt enable [0] (in \_TIM3\_IER)

Definition at line 3726 of file STM8AF\_STM8S.h.

**5.1.2.1232 \_TIM3\_CC2IF**

```
#define _TIM3_CC2IF ((uint8_t) (0x01 << 2))
```

TIM3 Capture/compare 2 interrupt flag [0] (in \_TIM3\_SR1)

Definition at line 3732 of file STM8AF\_STM8S.h.

**5.1.2.1233 \_TIM3\_CC2OF**

```
#define _TIM3_CC2OF ((uint8_t) (0x01 << 2))
```

TIM3 Capture/compare 2 overcapture flag [0] (in \_TIM3\_SR2)

Definition at line 3738 of file STM8AF\_STM8S.h.

**5.1.2.1234 \_TIM3\_CC2P**

```
#define _TIM3_CC2P ((uint8_t) (0x01 << 5))
```

TIM3 Capture/compare 2 output polarity [0] (in \_TIM3\_CCER1)

Definition at line 3798 of file STM8AF\_STM8S.h.

**5.1.2.1235 \_TIM3\_CC2S**

```
#define _TIM3_CC2S ((uint8_t) (0x03 << 0))
```

TIM3 Compare 2 selection [1:0] (in \_TIM3\_CCMR2)

Definition at line 3771 of file STM8AF\_STM8S.h.

**5.1.2.1236 \_TIM3\_CC2S0**

```
#define _TIM3_CC2S0 ((uint8_t) (0x01 << 0))
```

TIM3 Compare 2 selection [0] (in \_TIM3\_CCMR2)

Definition at line 3772 of file STM8AF\_STM8S.h.

**5.1.2.1237 \_TIM3\_CC2S1**

```
#define _TIM3_CC2S1 ((uint8_t) (0x01 << 1))
```

TIM3 Compare 2 selection [1] (in \_TIM3\_CCMR2)

Definition at line 3773 of file STM8AF\_STM8S.h.

**5.1.2.1238 \_TIM3\_CCER1**

```
#define _TIM3_CCER1 _SFR(uint8_t, TIM3_AddressBase+0x08)
```

TIM3 Capture/compare enable register 1.

Definition at line 3685 of file STM8AF\_STM8S.h.

**5.1.2.1239 \_TIM3\_CCER1\_RESET\_VALUE**

```
#define _TIM3_CCER1_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 Capture/compare enable register 1 reset value.

Definition at line 3704 of file STM8AF\_STM8S.h.

**5.1.2.1240 \_TIM3\_CCMR1**

```
#define _TIM3_CCMR1 _SFR(uint8_t, TIM3_AddressBase+0x05)
```

TIM3 Capture/compare mode register 1.

Definition at line 3683 of file STM8AF\_STM8S.h.

**5.1.2.1241 \_TIM3\_CCMR1\_RESET\_VALUE**

```
#define _TIM3_CCMR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 Capture/compare mode register 1 reset value.

Definition at line 3702 of file STM8AF\_STM8S.h.

**5.1.2.1242 \_TIM3\_CCMR2**

```
#define _TIM3_CCMR2 _SFR(uint8_t, TIM3_AddressBase+0x06)
```

TIM3 Capture/compare mode register 2.

Definition at line 3684 of file STM8AF\_STM8S.h.

#### 5.1.2.1243 \_TIM3\_CCMR2\_RESET\_VALUE

```
#define _TIM3_CCMR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 Capture/compare mode register 2 reset value.

Definition at line 3703 of file STM8AF\_STM8S.h.

#### 5.1.2.1244 \_TIM3\_CCR1H

```
#define _TIM3_CCR1H _SFR(uint8_t, TIM3_AddressBase+0x0F)
```

TIM3 16-bit capture/compare value 1 high byte.

Definition at line 3691 of file STM8AF\_STM8S.h.

#### 5.1.2.1245 \_TIM3\_CCR1H\_RESET\_VALUE

```
#define _TIM3_CCR1H_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 16-bit capture/compare value 1 high byte reset value.

Definition at line 3710 of file STM8AF\_STM8S.h.

#### 5.1.2.1246 \_TIM3\_CCR1L

```
#define _TIM3_CCR1L _SFR(uint8_t, TIM3_AddressBase+0x10)
```

TIM3 16-bit capture/compare value 1 low byte.

Definition at line 3692 of file STM8AF\_STM8S.h.

#### 5.1.2.1247 \_TIM3\_CCR1L\_RESET\_VALUE

```
#define _TIM3_CCR1L_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 16-bit capture/compare value 1 low byte reset value.

Definition at line 3711 of file STM8AF\_STM8S.h.



**5.1.2.1248 \_TIM3\_CCR2H**

```
#define _TIM3_CCR2H _SFR(uint8_t, TIM3_AddressBase+0x11)
```

TIM3 16-bit capture/compare value 2 high byte.

Definition at line 3693 of file STM8AF\_STM8S.h.

**5.1.2.1249 \_TIM3\_CCR2H\_RESET\_VALUE**

```
#define _TIM3_CCR2H_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 16-bit capture/compare value 2 high byte reset value.

Definition at line 3712 of file STM8AF\_STM8S.h.

**5.1.2.1250 \_TIM3\_CCR2L**

```
#define _TIM3_CCR2L _SFR(uint8_t, TIM3_AddressBase+0x12)
```

TIM3 16-bit capture/compare value 2 low byte.

Definition at line 3694 of file STM8AF\_STM8S.h.

**5.1.2.1251 \_TIM3\_CCR2L\_RESET\_VALUE**

```
#define _TIM3_CCR2L_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 16-bit capture/compare value 2 low byte reset value.

Definition at line 3713 of file STM8AF\_STM8S.h.

**5.1.2.1252 \_TIM3\_CEN**

```
#define _TIM3_CEN ((uint8_t) (0x01 << 0))
```

TIM3 Counter enable [0] (in \_TIM3\_CR1)

Definition at line 3716 of file STM8AF\_STM8S.h.

#### 5.1.2.1253 \_TIM3\_CNTRH

```
#define _TIM3_CNTRH _SFR(uint8_t, TIM3_AddressBase+0x0A)
```

TIM3 counter register high byte.

Definition at line 3686 of file STM8AF\_STM8S.h.

#### 5.1.2.1254 \_TIM3\_CNTRH\_RESET\_VALUE

```
#define _TIM3_CNTRH_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 counter register high byte reset value.

Definition at line 3705 of file STM8AF\_STM8S.h.

#### 5.1.2.1255 \_TIM3\_CNTRL

```
#define _TIM3_CNTRL _SFR(uint8_t, TIM3_AddressBase+0x0B)
```

TIM3 counter register low byte.

Definition at line 3687 of file STM8AF\_STM8S.h.

#### 5.1.2.1256 \_TIM3\_CNTRL\_RESET\_VALUE

```
#define _TIM3_CNTRL_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 counter register low byte reset value.

Definition at line 3706 of file STM8AF\_STM8S.h.

#### 5.1.2.1257 \_TIM3\_CR1

```
#define _TIM3_CR1 _SFR(uint8_t, TIM3_AddressBase+0x00)
```

TIM3 control register 1.

Definition at line 3678 of file STM8AF\_STM8S.h.

**5.1.2.1258 \_TIM3\_CR1\_RESET\_VALUE**

```
#define _TIM3_CR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 control register 1 reset value.

Definition at line 3697 of file STM8AF\_STM8S.h.

**5.1.2.1259 \_TIM3\_EGR**

```
#define _TIM3_EGR _SFR(uint8_t, TIM3_AddressBase+0x04)
```

TIM3 Event generation register.

Definition at line 3682 of file STM8AF\_STM8S.h.

**5.1.2.1260 \_TIM3\_EGR\_RESET\_VALUE**

```
#define _TIM3_EGR_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 Event generation register reset value.

Definition at line 3701 of file STM8AF\_STM8S.h.

**5.1.2.1261 \_TIM3\_IC1F**

```
#define _TIM3_IC1F ((uint8_t) (0x0F << 4))
```

TIM3 Output compare 1 mode [3:0] (in \_TIM3\_CCMR1)

Definition at line 3764 of file STM8AF\_STM8S.h.

**5.1.2.1262 \_TIM3\_IC1F0**

```
#define _TIM3_IC1F0 ((uint8_t) (0x01 << 4))
```

TIM3 Output compare 1 mode [0] (in \_TIM3\_CCMR1)

Definition at line 3765 of file STM8AF\_STM8S.h.

**5.1.2.1263 \_TIM3\_IC1F1**

```
#define _TIM3_IC1F1 ((uint8_t) (0x01 << 5))
```

TIM3 Output compare 1 mode [1] (in \_TIM3\_CCMR1)

Definition at line 3766 of file STM8AF\_STM8S.h.

**5.1.2.1264 \_TIM3\_IC1F2**

```
#define _TIM3_IC1F2 ((uint8_t) (0x01 << 6))
```

TIM3 Output compare 1 mode [2] (in \_TIM3\_CCMR1)

Definition at line 3767 of file STM8AF\_STM8S.h.

**5.1.2.1265 \_TIM3\_IC1F3**

```
#define _TIM3_IC1F3 ((uint8_t) (0x01 << 7))
```

TIM3 Output compare 1 mode [3] (in \_TIM3\_CCMR1)

Definition at line 3768 of file STM8AF\_STM8S.h.

**5.1.2.1266 \_TIM3\_IC1PSC**

```
#define _TIM3_IC1PSC ((uint8_t) (0x03 << 2))
```

TIM3 Input capture 1 prescaler [1:0] (in \_TIM3\_CCMR1)

Definition at line 3761 of file STM8AF\_STM8S.h.

**5.1.2.1267 \_TIM3\_IC1PSC0**

```
#define _TIM3_IC1PSC0 ((uint8_t) (0x01 << 2))
```

TIM3 Input capture 1 prescaler [0] (in \_TIM3\_CCMR1)

Definition at line 3762 of file STM8AF\_STM8S.h.

**5.1.2.1268 \_TIM3\_IC1PSC1**

```
#define _TIM3_IC1PSC1 ((uint8_t) (0x01 << 3))
```

TIM3 Input capture 1 prescaler [1] (in \_TIM3\_CCMR1)

Definition at line 3763 of file STM8AF\_STM8S.h.

**5.1.2.1269 \_TIM3\_IC2F**

```
#define _TIM3_IC2F ((uint8_t) (0x0F << 4))
```

TIM3 Output compare 2 mode [3:0] (in \_TIM3\_CCMR2)

Definition at line 3787 of file STM8AF\_STM8S.h.

**5.1.2.1270 \_TIM3\_IC2F0**

```
#define _TIM3_IC2F0 ((uint8_t) (0x01 << 4))
```

TIM3 Output compare 2 mode [0] (in \_TIM3\_CCMR2)

Definition at line 3788 of file STM8AF\_STM8S.h.

**5.1.2.1271 \_TIM3\_IC2F1**

```
#define _TIM3_IC2F1 ((uint8_t) (0x01 << 5))
```

TIM3 Output compare 2 mode [1] (in \_TIM3\_CCMR2)

Definition at line 3789 of file STM8AF\_STM8S.h.

**5.1.2.1272 \_TIM3\_IC2F2**

```
#define _TIM3_IC2F2 ((uint8_t) (0x01 << 6))
```

TIM3 Output compare 2 mode [2] (in \_TIM3\_CCMR2)

Definition at line 3790 of file STM8AF\_STM8S.h.

#### 5.1.2.1273 `_TIM3_IC2F3`

```
#define _TIM3_IC2F3 ((uint8_t) (0x01 << 7))
```

TIM3 Output compare 2 mode [3] (in `_TIM3_CCMR2`)

Definition at line 3791 of file `STM8AF_STM8S.h`.

#### 5.1.2.1274 `_TIM3_IC2PSC`

```
#define _TIM3_IC2PSC ((uint8_t) (0x03 << 2))
```

TIM3 Input capture 2 prescaler [1:0] (in `_TIM3_CCMR2`)

Definition at line 3784 of file `STM8AF_STM8S.h`.

#### 5.1.2.1275 `_TIM3_IC2PSC0`

```
#define _TIM3_IC2PSC0 ((uint8_t) (0x01 << 2))
```

TIM3 Input capture 2 prescaler [0] (in `_TIM3_CCMR2`)

Definition at line 3785 of file `STM8AF_STM8S.h`.

#### 5.1.2.1276 `_TIM3_IC2PSC1`

```
#define _TIM3_IC2PSC1 ((uint8_t) (0x01 << 3))
```

TIM3 Input capture 2 prescaler [1] (in `_TIM3_CCMR2`)

Definition at line 3786 of file `STM8AF_STM8S.h`.

#### 5.1.2.1277 `_TIM3_IER`

```
#define _TIM3_IER _SFR(uint8_t, TIM3_AddressBase+0x01)
```

TIM3 interrupt enable register.

Definition at line 3679 of file `STM8AF_STM8S.h`.

**5.1.2.1278 \_TIM3\_IER\_RESET\_VALUE**

```
#define _TIM3_IER_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 interrupt enable register reset value.

Definition at line 3698 of file STM8AF\_STM8S.h.

**5.1.2.1279 \_TIM3\_OC1M**

```
#define _TIM3_OC1M ((uint8_t) (0x07 << 4))
```

TIM3 Output compare 1 mode [2:0] (in \_TIM3\_CCMR1)

Definition at line 3753 of file STM8AF\_STM8S.h.

**5.1.2.1280 \_TIM3\_OC1M0**

```
#define _TIM3_OC1M0 ((uint8_t) (0x01 << 4))
```

TIM3 Output compare 1 mode [0] (in \_TIM3\_CCMR1)

Definition at line 3754 of file STM8AF\_STM8S.h.

**5.1.2.1281 \_TIM3\_OC1M1**

```
#define _TIM3_OC1M1 ((uint8_t) (0x01 << 5))
```

TIM3 Output compare 1 mode [1] (in \_TIM3\_CCMR1)

Definition at line 3755 of file STM8AF\_STM8S.h.

**5.1.2.1282 \_TIM3\_OC1M2**

```
#define _TIM3_OC1M2 ((uint8_t) (0x01 << 6))
```

TIM3 Output compare 1 mode [2] (in \_TIM3\_CCMR1)

Definition at line 3756 of file STM8AF\_STM8S.h.

**5.1.2.1283 \_TIM3\_OC1PE**

```
#define _TIM3_OC1PE ((uint8_t) (0x01 << 3))
```

TIM3 Output compare 1 preload enable [0] (in \_TIM3\_CCMR1)

Definition at line 3752 of file STM8AF\_STM8S.h.

**5.1.2.1284 \_TIM3\_OC2M**

```
#define _TIM3_OC2M ((uint8_t) (0x07 << 4))
```

TIM3 Output compare 2 mode [2:0] (in \_TIM3\_CCMR2)

Definition at line 3776 of file STM8AF\_STM8S.h.

**5.1.2.1285 \_TIM3\_OC2M0**

```
#define _TIM3_OC2M0 ((uint8_t) (0x01 << 4))
```

TIM3 Output compare 2 mode [0] (in \_TIM3\_CCMR2)

Definition at line 3777 of file STM8AF\_STM8S.h.

**5.1.2.1286 \_TIM3\_OC2M1**

```
#define _TIM3_OC2M1 ((uint8_t) (0x01 << 5))
```

TIM3 Output compare 2 mode [1] (in \_TIM3\_CCMR2)

Definition at line 3778 of file STM8AF\_STM8S.h.

**5.1.2.1287 \_TIM3\_OC2M2**

```
#define _TIM3_OC2M2 ((uint8_t) (0x01 << 6))
```

TIM3 Output compare 2 mode [2] (in \_TIM3\_CCMR2)

Definition at line 3779 of file STM8AF\_STM8S.h.



**5.1.2.1288 \_TIM3\_OC2PE**

```
#define _TIM3_OC2PE ((uint8_t) (0x01 << 3))
```

TIM3 Output compare 2 preload enable [0] (in \_TIM3\_CCMR2)

Definition at line 3775 of file STM8AF\_STM8S.h.

**5.1.2.1289 \_TIM3\_OPM**

```
#define _TIM3_OPM ((uint8_t) (0x01 << 3))
```

TIM3 One-pulse mode [0] (in \_TIM3\_CR1)

Definition at line 3719 of file STM8AF\_STM8S.h.

**5.1.2.1290 \_TIM3\_PSC**

```
#define _TIM3_PSC ((uint8_t) (0x0F << 0))
```

TIM3 clock prescaler [3:0] (in \_TIM3\_PSCR)

Definition at line 3802 of file STM8AF\_STM8S.h.

**5.1.2.1291 \_TIM3\_PSC0**

```
#define _TIM3_PSC0 ((uint8_t) (0x01 << 0))
```

TIM3 clock prescaler [0] (in \_TIM3\_PSCR)

Definition at line 3803 of file STM8AF\_STM8S.h.

**5.1.2.1292 \_TIM3\_PSC1**

```
#define _TIM3_PSC1 ((uint8_t) (0x01 << 1))
```

TIM3 clock prescaler [1] (in \_TIM3\_PSCR)

Definition at line 3804 of file STM8AF\_STM8S.h.

#### 5.1.2.1293 \_TIM3\_PSC2

```
#define _TIM3_PSC2 ((uint8_t) (0x01 << 2))
```

TIM3 clock prescaler [2] (in \_TIM3\_PSCR)

Definition at line 3805 of file STM8AF\_STM8S.h.

#### 5.1.2.1294 \_TIM3\_PSC3

```
#define _TIM3_PSC3 ((uint8_t) (0x01 << 3))
```

TIM3 clock prescaler [3] (in \_TIM3\_PSCR)

Definition at line 3806 of file STM8AF\_STM8S.h.

#### 5.1.2.1295 \_TIM3\_PSCR

```
#define _TIM3_PSCR _SFR(uint8_t, TIM3_AddressBase+0x0C)
```

TIM3 clock prescaler register.

Definition at line 3688 of file STM8AF\_STM8S.h.

#### 5.1.2.1296 \_TIM3\_PSCR\_RESET\_VALUE

```
#define _TIM3_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 clock prescaler register reset value.

Definition at line 3707 of file STM8AF\_STM8S.h.

#### 5.1.2.1297 \_TIM3\_SR1

```
#define _TIM3_SR1 _SFR(uint8_t, TIM3_AddressBase+0x02)
```

TIM3 status register 1.

Definition at line 3680 of file STM8AF\_STM8S.h.

**5.1.2.1298 \_TIM3\_SR1\_RESET\_VALUE**

```
#define _TIM3_SR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 status register 1 reset value.

Definition at line 3699 of file STM8AF\_STM8S.h.

**5.1.2.1299 \_TIM3\_SR2**

```
#define _TIM3_SR2 _SFR(uint8_t, TIM3_AddressBase+0x03)
```

TIM3 status register 2.

Definition at line 3681 of file STM8AF\_STM8S.h.

**5.1.2.1300 \_TIM3\_SR2\_RESET\_VALUE**

```
#define _TIM3_SR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM3 status register 2 reset value.

Definition at line 3700 of file STM8AF\_STM8S.h.

**5.1.2.1301 \_TIM3\_UDIS**

```
#define _TIM3_UDIS ((uint8_t) (0x01 << 1))
```

TIM3 Update disable [0] (in \_TIM3\_CR1)

Definition at line 3717 of file STM8AF\_STM8S.h.

**5.1.2.1302 \_TIM3\_UG**

```
#define _TIM3_UG ((uint8_t) (0x01 << 0))
```

TIM3 Update generation [0] (in \_TIM3\_EGR)

Definition at line 3742 of file STM8AF\_STM8S.h.

#### 5.1.2.1303 \_TIM3\_UIE

```
#define _TIM3_UIE ((uint8_t) (0x01 << 0))
```

TIM3 Update interrupt enable [0] (in \_TIM3\_IER)

Definition at line 3724 of file STM8AF\_STM8S.h.

#### 5.1.2.1304 \_TIM3\_UIF

```
#define _TIM3_UIF ((uint8_t) (0x01 << 0))
```

TIM3 Update interrupt flag [0] (in \_TIM3\_SR1)

Definition at line 3730 of file STM8AF\_STM8S.h.

#### 5.1.2.1305 \_TIM3\_URS

```
#define _TIM3_URS ((uint8_t) (0x01 << 2))
```

TIM3 Update request source [0] (in \_TIM3\_CR1)

Definition at line 3718 of file STM8AF\_STM8S.h.

#### 5.1.2.1306 \_TIM4

```
#define _TIM4 _SFR(TIM4_t, TIM4_AddressBase)
```

TIM4 struct/bit access.

Definition at line 3879 of file STM8AF\_STM8S.h.

#### 5.1.2.1307 \_TIM4\_ARPE

```
#define _TIM4_ARPE ((uint8_t) (0x01 << 7))
```

TIM4 Auto-reload preload enable [0] (in \_TIM4\_CR)

Definition at line 3913 of file STM8AF\_STM8S.h.

### 5.1.2.1308 \_TIM4\_ARR

```
#define _TIM4_ARR _SFR(uint8_t, TIM4_AddressBase+0x06)
```

TIM4 auto-reload register.

Definition at line 3895 of file STM8AF\_STM8S.h.

### 5.1.2.1309 \_TIM4\_ARR\_RESET\_VALUE

```
#define _TIM4_ARR_RESET_VALUE ((uint8_t) 0xFF)
```

TIM4 auto-reload register reset value.

Definition at line 3905 of file STM8AF\_STM8S.h.

### 5.1.2.1310 \_TIM4\_CEN

```
#define _TIM4_CEN ((uint8_t) (0x01 << 0))
```

TIM4 Counter enable [0] (in \_TIM4\_CR)

Definition at line 3908 of file STM8AF\_STM8S.h.

### 5.1.2.1311 \_TIM4\_CNTR

```
#define _TIM4_CNTR _SFR(uint8_t, TIM4_AddressBase+0x04)
```

TIM4 counter register.

Definition at line 3893 of file STM8AF\_STM8S.h.

### 5.1.2.1312 \_TIM4\_CNTR\_RESET\_VALUE

```
#define _TIM4_CNTR_RESET_VALUE ((uint8_t) 0x00)
```

TIM4 counter register reset value.

Definition at line 3903 of file STM8AF\_STM8S.h.

#### 5.1.2.1313 \_TIM4\_CR

```
#define _TIM4_CR _SFR(uint8_t, TIM4_AddressBase+0x00)
```

TIM4 control register.

Definition at line 3880 of file STM8AF\_STM8S.h.

#### 5.1.2.1314 \_TIM4\_CR\_RESET\_VALUE

```
#define _TIM4_CR_RESET_VALUE ((uint8_t) 0x00)
```

TIM4 control register reset value.

Definition at line 3899 of file STM8AF\_STM8S.h.

#### 5.1.2.1315 \_TIM4\_EGR

```
#define _TIM4_EGR _SFR(uint8_t, TIM4_AddressBase+0x03)
```

TIM4 event generation register.

Definition at line 3892 of file STM8AF\_STM8S.h.

#### 5.1.2.1316 \_TIM4\_EGR\_RESET\_VALUE

```
#define _TIM4_EGR_RESET_VALUE ((uint8_t) 0x00)
```

TIM4 event generation register reset value.

Definition at line 3902 of file STM8AF\_STM8S.h.

#### 5.1.2.1317 \_TIM4\_IER

```
#define _TIM4_IER _SFR(uint8_t, TIM4_AddressBase+0x01)
```

TIM4 interrupt enable register.

Definition at line 3890 of file STM8AF\_STM8S.h.

**5.1.2.1318 \_TIM4\_IER\_RESET\_VALUE**

```
#define _TIM4_IER_RESET_VALUE ((uint8_t) 0x00)
```

TIM4 interrupt enable register reset value.

Definition at line 3900 of file STM8AF\_STM8S.h.

**5.1.2.1319 \_TIM4\_OPM**

```
#define _TIM4_OPM ((uint8_t) (0x01 << 3))
```

TIM4 One-pulse mode [0] (in \_TIM4\_CR)

Definition at line 3911 of file STM8AF\_STM8S.h.

**5.1.2.1320 \_TIM4\_PSC**

```
#define _TIM4_PSC ((uint8_t) (0x07 << 0))
```

TIM4 clock prescaler [2:0] (in \_TIM4\_PSCR)

Definition at line 3928 of file STM8AF\_STM8S.h.

**5.1.2.1321 \_TIM4\_PSC0**

```
#define _TIM4_PSC0 ((uint8_t) (0x01 << 0))
```

TIM4 clock prescaler [0] (in \_TIM4\_PSCR)

Definition at line 3929 of file STM8AF\_STM8S.h.

**5.1.2.1322 \_TIM4\_PSC1**

```
#define _TIM4_PSC1 ((uint8_t) (0x01 << 1))
```

TIM4 clock prescaler [1] (in \_TIM4\_PSCR)

Definition at line 3930 of file STM8AF\_STM8S.h.

#### 5.1.2.1323 \_TIM4\_PSC2

```
#define _TIM4_PSC2 ((uint8_t) (0x01 << 2))
```

TIM4 clock prescaler [2] (in \_TIM4\_PSCR)

Definition at line 3931 of file STM8AF\_STM8S.h.

#### 5.1.2.1324 \_TIM4\_PSCR

```
#define _TIM4_PSCR _SFR(uint8_t, TIM4_AddressBase+0x05)
```

TIM4 clock prescaler register.

Definition at line 3894 of file STM8AF\_STM8S.h.

#### 5.1.2.1325 \_TIM4\_PSCR\_RESET\_VALUE

```
#define _TIM4_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM4 clock prescaler register reset value.

Definition at line 3904 of file STM8AF\_STM8S.h.

#### 5.1.2.1326 \_TIM4\_SR

```
#define _TIM4_SR _SFR(uint8_t, TIM4_AddressBase+0x02)
```

TIM4 status register.

Definition at line 3891 of file STM8AF\_STM8S.h.

#### 5.1.2.1327 \_TIM4\_SR\_RESET\_VALUE

```
#define _TIM4_SR_RESET_VALUE ((uint8_t) 0x00)
```

TIM4 status register reset value.

Definition at line 3901 of file STM8AF\_STM8S.h.



**5.1.2.1328 \_TIM4\_UDIS**

```
#define _TIM4_UDIS ((uint8_t) (0x01 << 1))
```

TIM4 Update disable [0] (in \_TIM4\_CR)

Definition at line 3909 of file STM8AF\_STM8S.h.

**5.1.2.1329 \_TIM4\_UG**

```
#define _TIM4_UG ((uint8_t) (0x01 << 0))
```

TIM4 Update generation [0] (in \_TIM4\_EGR)

Definition at line 3924 of file STM8AF\_STM8S.h.

**5.1.2.1330 \_TIM4\_UIE**

```
#define _TIM4_UIE ((uint8_t) (0x01 << 0))
```

TIM4 Update interrupt enable [0] (in \_TIM4\_IER)

Definition at line 3916 of file STM8AF\_STM8S.h.

**5.1.2.1331 \_TIM4\_UIF**

```
#define _TIM4_UIF ((uint8_t) (0x01 << 0))
```

TIM4 Update interrupt flag [0] (in \_TIM4\_SR)

Definition at line 3920 of file STM8AF\_STM8S.h.

**5.1.2.1332 \_TIM4\_URS**

```
#define _TIM4_URS ((uint8_t) (0x01 << 2))
```

TIM4 Update request source [0] (in \_TIM4\_CR)

Definition at line 3910 of file STM8AF\_STM8S.h.

#### 5.1.2.1333 \_TIM5

```
#define _TIM5 _SFR(TIM5_t, TIM5_AddressBase)
```

TIM5 struct/bit access.

Definition at line 4177 of file STM8AF\_STM8S.h.

#### 5.1.2.1334 \_TIM5\_ARPE

```
#define _TIM5_ARPE ((uint8_t) (0x01 << 7))
```

TIM5 Auto-reload preload enable [0] (in \_TIM5\_CR1)

Definition at line 4233 of file STM8AF\_STM8S.h.

#### 5.1.2.1335 \_TIM5\_ARRH

```
#define _TIM5_ARRH _SFR(uint8_t, TIM5_AddressBase+0x0F)
```

TIM5 auto-reload register high byte.

Definition at line 4193 of file STM8AF\_STM8S.h.

#### 5.1.2.1336 \_TIM5\_ARRH\_RESET\_VALUE

```
#define _TIM5_ARRH_RESET_VALUE ((uint8_t) 0xFF)
```

TIM5 auto-reload register high byte reset value.

Definition at line 4218 of file STM8AF\_STM8S.h.

#### 5.1.2.1337 \_TIM5\_ARRL

```
#define _TIM5_ARRL _SFR(uint8_t, TIM5_AddressBase+0x10)
```

TIM5 auto-reload register low byte.

Definition at line 4194 of file STM8AF\_STM8S.h.

**5.1.2.1338 \_TIM5\_ARRL\_RESET\_VALUE**

```
#define _TIM5_ARRL_RESET_VALUE ((uint8_t) 0xFF)
```

TIM5 auto-reload register low byte reset value.

Definition at line 4219 of file STM8AF\_STM8S.h.

**5.1.2.1339 \_TIM5\_CC1E**

```
#define _TIM5_CC1E ((uint8_t) (0x01 << 0))
```

TIM5 Capture/compare 1 output enable [0] (in \_TIM5\_CCER1)

Definition at line 4362 of file STM8AF\_STM8S.h.

**5.1.2.1340 \_TIM5\_CC1G**

```
#define _TIM5_CC1G ((uint8_t) (0x01 << 1))
```

TIM5 Capture/compare 1 generation [0] (in \_TIM5\_EGR)

Definition at line 4285 of file STM8AF\_STM8S.h.

**5.1.2.1341 \_TIM5\_CC1IE**

```
#define _TIM5_CC1IE ((uint8_t) (0x01 << 1))
```

TIM5 Capture/compare 1 interrupt enable [0] (in \_TIM5\_IER)

Definition at line 4260 of file STM8AF\_STM8S.h.

**5.1.2.1342 \_TIM5\_CC1IF**

```
#define _TIM5_CC1IF ((uint8_t) (0x01 << 1))
```

TIM5 Capture/compare 1 interrupt flag [0] (in \_TIM5\_SR1)

Definition at line 4269 of file STM8AF\_STM8S.h.

**5.1.2.1343 \_TIM5\_CC1OF**

```
#define _TIM5_CC1OF ((uint8_t) (0x01 << 1))
```

TIM5 Capture/compare 1 overcapture flag [0] (in \_TIM5\_SR2)

Definition at line 4278 of file STM8AF\_STM8S.h.

**5.1.2.1344 \_TIM5\_CC1P**

```
#define _TIM5_CC1P ((uint8_t) (0x01 << 1))
```

TIM5 Capture/compare 1 output polarity [0] (in \_TIM5\_CCER1)

Definition at line 4363 of file STM8AF\_STM8S.h.

**5.1.2.1345 \_TIM5\_CC1S**

```
#define _TIM5_CC1S ((uint8_t) (0x03 << 0))
```

TIM5 Compare 1 selection [1:0] (in \_TIM5\_CCMR1)

Definition at line 4293 of file STM8AF\_STM8S.h.

**5.1.2.1346 \_TIM5\_CC1S0**

```
#define _TIM5_CC1S0 ((uint8_t) (0x01 << 0))
```

TIM5 Compare 1 selection [0] (in \_TIM5\_CCMR1)

Definition at line 4294 of file STM8AF\_STM8S.h.

**5.1.2.1347 \_TIM5\_CC1S1**

```
#define _TIM5_CC1S1 ((uint8_t) (0x01 << 1))
```

TIM5 Compare 1 selection [1] (in \_TIM5\_CCMR1)

Definition at line 4295 of file STM8AF\_STM8S.h.

**5.1.2.1348 \_TIM5\_CC2E**

```
#define _TIM5_CC2E ((uint8_t) (0x01 << 4))
```

TIM5 Capture/compare 2 output enable [0] (in \_TIM5\_CCER1)

Definition at line 4365 of file STM8AF\_STM8S.h.

**5.1.2.1349 \_TIM5\_CC2G**

```
#define _TIM5_CC2G ((uint8_t) (0x01 << 2))
```

TIM5 Capture/compare 2 generation [0] (in \_TIM5\_EGR)

Definition at line 4286 of file STM8AF\_STM8S.h.

**5.1.2.1350 \_TIM5\_CC2IE**

```
#define _TIM5_CC2IE ((uint8_t) (0x01 << 2))
```

TIM5 Capture/compare 2 interrupt enable [0] (in \_TIM5\_IER)

Definition at line 4261 of file STM8AF\_STM8S.h.

**5.1.2.1351 \_TIM5\_CC2IF**

```
#define _TIM5_CC2IF ((uint8_t) (0x01 << 2))
```

TIM5 Capture/compare 2 interrupt flag [0] (in \_TIM5\_SR1)

Definition at line 4270 of file STM8AF\_STM8S.h.

**5.1.2.1352 \_TIM5\_CC2OF**

```
#define _TIM5_CC2OF ((uint8_t) (0x01 << 2))
```

TIM5 Capture/compare 2 overcapture flag [0] (in \_TIM5\_SR2)

Definition at line 4279 of file STM8AF\_STM8S.h.

**5.1.2.1353 \_TIM5\_CC2P**

```
#define _TIM5_CC2P ((uint8_t) (0x01 << 5))
```

TIM5 Capture/compare 2 output polarity [0] (in \_TIM5\_CCER1)

Definition at line 4366 of file STM8AF\_STM8S.h.

**5.1.2.1354 \_TIM5\_CC2S**

```
#define _TIM5_CC2S ((uint8_t) (0x03 << 0))
```

TIM5 Compare 2 selection [1:0] (in \_TIM5\_CCMR2)

Definition at line 4316 of file STM8AF\_STM8S.h.

**5.1.2.1355 \_TIM5\_CC2S0**

```
#define _TIM5_CC2S0 ((uint8_t) (0x01 << 0))
```

TIM5 Compare 2 selection [0] (in \_TIM5\_CCMR2)

Definition at line 4317 of file STM8AF\_STM8S.h.

**5.1.2.1356 \_TIM5\_CC2S1**

```
#define _TIM5_CC2S1 ((uint8_t) (0x01 << 1))
```

TIM5 Compare 2 selection [1] (in \_TIM5\_CCMR2)

Definition at line 4318 of file STM8AF\_STM8S.h.

**5.1.2.1357 \_TIM5\_CC3E**

```
#define _TIM5_CC3E ((uint8_t) (0x01 << 0))
```

TIM5 Capture/compare 3 output enable [0] (in \_TIM5\_CCER2)

Definition at line 4370 of file STM8AF\_STM8S.h.

**5.1.2.1358 \_TIM5\_CC3G**

```
#define _TIM5_CC3G ((uint8_t) (0x01 << 3))
```

TIM5 Capture/compare 3 generation [0] (in \_TIM5\_EGR)

Definition at line 4287 of file STM8AF\_STM8S.h.

**5.1.2.1359 \_TIM5\_CC3IE**

```
#define _TIM5_CC3IE ((uint8_t) (0x01 << 3))
```

TIM5 Capture/compare 3 interrupt enable [0] (in \_TIM5\_IER)

Definition at line 4262 of file STM8AF\_STM8S.h.

**5.1.2.1360 \_TIM5\_CC3IF**

```
#define _TIM5_CC3IF ((uint8_t) (0x01 << 3))
```

TIM5 Capture/compare 3 interrupt flag [0] (in \_TIM5\_SR1)

Definition at line 4271 of file STM8AF\_STM8S.h.

**5.1.2.1361 \_TIM5\_CC3OF**

```
#define _TIM5_CC3OF ((uint8_t) (0x01 << 3))
```

TIM5 Capture/compare 3 overcapture flag [0] (in \_TIM5\_SR2)

Definition at line 4280 of file STM8AF\_STM8S.h.

**5.1.2.1362 \_TIM5\_CC3P**

```
#define _TIM5_CC3P ((uint8_t) (0x01 << 1))
```

TIM5 Capture/compare 3 output polarity [0] (in \_TIM5\_CCER2)

Definition at line 4371 of file STM8AF\_STM8S.h.

#### 5.1.2.1363 \_TIM5\_CC3S

```
#define _TIM5_CC3S ((uint8_t) (0x03 << 0))
```

TIM5 Compare 3 selection [1:0] (in \_TIM5\_CCMR3)

Definition at line 4339 of file STM8AF\_STM8S.h.

#### 5.1.2.1364 \_TIM5\_CC3S0

```
#define _TIM5_CC3S0 ((uint8_t) (0x01 << 0))
```

TIM5 Compare 3 selection [0] (in \_TIM5\_CCMR3)

Definition at line 4340 of file STM8AF\_STM8S.h.

#### 5.1.2.1365 \_TIM5\_CC3S1

```
#define _TIM5_CC3S1 ((uint8_t) (0x01 << 1))
```

TIM5 Compare 3 selection [1] (in \_TIM5\_CCMR3)

Definition at line 4341 of file STM8AF\_STM8S.h.

#### 5.1.2.1366 \_TIM5\_CCER1

```
#define _TIM5_CCER1 \_SFR(uint8_t, TIM5\_AddressBase+0x0A)
```

TIM5 Capture/compare enable register 1.

Definition at line 4188 of file STM8AF\_STM8S.h.

#### 5.1.2.1367 \_TIM5\_CCER1\_RESET\_VALUE

```
#define _TIM5_CCER1_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 Capture/compare enable register 1 reset value.

Definition at line 4213 of file STM8AF\_STM8S.h.



**5.1.2.1368 \_TIM5\_CCER2**

```
#define _TIM5_CCER2 _SFR(uint8_t, TIM5_AddressBase+0x0B)
```

TIM5 Capture/compare enable register 2.

Definition at line 4189 of file STM8AF\_STM8S.h.

**5.1.2.1369 \_TIM5\_CCER2\_RESET\_VALUE**

```
#define _TIM5_CCER2_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 Capture/compare enable register 2 reset value.

Definition at line 4214 of file STM8AF\_STM8S.h.

**5.1.2.1370 \_TIM5\_CCMR1**

```
#define _TIM5_CCMR1 _SFR(uint8_t, TIM5_AddressBase+0x07)
```

TIM5 Capture/compare mode register 1.

Definition at line 4185 of file STM8AF\_STM8S.h.

**5.1.2.1371 \_TIM5\_CCMR1\_RESET\_VALUE**

```
#define _TIM5_CCMR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 Capture/compare mode register 1 reset value.

Definition at line 4210 of file STM8AF\_STM8S.h.

**5.1.2.1372 \_TIM5\_CCMR2**

```
#define _TIM5_CCMR2 _SFR(uint8_t, TIM5_AddressBase+0x08)
```

TIM5 Capture/compare mode register 2.

Definition at line 4186 of file STM8AF\_STM8S.h.

#### 5.1.2.1373 \_TIM5\_CCMR2\_RESET\_VALUE

```
#define _TIM5_CCMR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 Capture/compare mode register 2 reset value.

Definition at line 4211 of file STM8AF\_STM8S.h.

#### 5.1.2.1374 \_TIM5\_CCMR3

```
#define _TIM5_CCMR3 __SFR(uint8_t, TIM5_AddressBase+0x09)
```

TIM5 Capture/compare mode register 3.

Definition at line 4187 of file STM8AF\_STM8S.h.

#### 5.1.2.1375 \_TIM5\_CCMR3\_RESET\_VALUE

```
#define _TIM5_CCMR3_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 Capture/compare mode register 3 reset value.

Definition at line 4212 of file STM8AF\_STM8S.h.

#### 5.1.2.1376 \_TIM5\_CCPC

```
#define _TIM5_CCPC ((uint8_t) (0x01 << 0))
```

TIM5 Capture/compare preloaded control [0] (in \_TIM5\_CR2)

Definition at line 4236 of file STM8AF\_STM8S.h.

#### 5.1.2.1377 \_TIM5\_CCR1H

```
#define _TIM5_CCR1H __SFR(uint8_t, TIM5_AddressBase+0x11)
```

TIM5 16-bit capture/compare value 1 high byte.

Definition at line 4195 of file STM8AF\_STM8S.h.

**5.1.2.1378 \_TIM5\_CCR1H\_RESET\_VALUE**

```
#define _TIM5_CCR1H_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 16-bit capture/compare value 1 high byte reset value.

Definition at line 4220 of file STM8AF\_STM8S.h.

**5.1.2.1379 \_TIM5\_CCR1L**

```
#define _TIM5_CCR1L _SFR(uint8_t, TIM5_AddressBase+0x12)
```

TIM5 16-bit capture/compare value 1 low byte.

Definition at line 4196 of file STM8AF\_STM8S.h.

**5.1.2.1380 \_TIM5\_CCR1L\_RESET\_VALUE**

```
#define _TIM5_CCR1L_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 16-bit capture/compare value 1 low byte reset value.

Definition at line 4221 of file STM8AF\_STM8S.h.

**5.1.2.1381 \_TIM5\_CCR2H**

```
#define _TIM5_CCR2H _SFR(uint8_t, TIM5_AddressBase+0x13)
```

TIM5 16-bit capture/compare value 2 high byte.

Definition at line 4197 of file STM8AF\_STM8S.h.

**5.1.2.1382 \_TIM5\_CCR2H\_RESET\_VALUE**

```
#define _TIM5_CCR2H_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 16-bit capture/compare value 2 high byte reset value.

Definition at line 4222 of file STM8AF\_STM8S.h.

#### 5.1.2.1383 \_TIM5\_CCR2L

```
#define _TIM5_CCR2L _SFR(uint8_t, TIM5_AddressBase+0x14)
```

TIM5 16-bit capture/compare value 2 low byte.

Definition at line 4198 of file STM8AF\_STM8S.h.

#### 5.1.2.1384 \_TIM5\_CCR2L\_RESET\_VALUE

```
#define _TIM5_CCR2L_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 16-bit capture/compare value 2 low byte reset value.

Definition at line 4223 of file STM8AF\_STM8S.h.

#### 5.1.2.1385 \_TIM5\_CCR3H

```
#define _TIM5_CCR3H _SFR(uint8_t, TIM5_AddressBase+0x15)
```

TIM5 16-bit capture/compare value 3 high byte.

Definition at line 4199 of file STM8AF\_STM8S.h.

#### 5.1.2.1386 \_TIM5\_CCR3H\_RESET\_VALUE

```
#define _TIM5_CCR3H_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 16-bit capture/compare value 3 high byte reset value.

Definition at line 4224 of file STM8AF\_STM8S.h.

#### 5.1.2.1387 \_TIM5\_CCR3L

```
#define _TIM5_CCR3L _SFR(uint8_t, TIM5_AddressBase+0x16)
```

TIM5 16-bit capture/compare value 3 low byte.

Definition at line 4200 of file STM8AF\_STM8S.h.

**5.1.2.1388 \_TIM5\_CCR3L\_RESET\_VALUE**

```
#define _TIM5_CCR3L_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 16-bit capture/compare value 3 low byte reset value.

Definition at line 4225 of file STM8AF\_STM8S.h.

**5.1.2.1389 \_TIM5\_CEN**

```
#define _TIM5_CEN ((uint8_t) (0x01 << 0))
```

TIM5 Counter enable [0] (in \_TIM5\_CR1)

Definition at line 4228 of file STM8AF\_STM8S.h.

**5.1.2.1390 \_TIM5\_CNTRH**

```
#define _TIM5_CNTRH _SFR(uint8_t, TIM5_AddressBase+0x0C)
```

TIM5 counter register high byte.

Definition at line 4190 of file STM8AF\_STM8S.h.

**5.1.2.1391 \_TIM5\_CNTRH\_RESET\_VALUE**

```
#define _TIM5_CNTRH_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 counter register high byte reset value.

Definition at line 4215 of file STM8AF\_STM8S.h.

**5.1.2.1392 \_TIM5\_CNTRL**

```
#define _TIM5_CNTRL _SFR(uint8_t, TIM5_AddressBase+0x0D)
```

TIM5 counter register low byte.

Definition at line 4191 of file STM8AF\_STM8S.h.

#### 5.1.2.1393 \_TIM5\_CNTRL\_RESET\_VALUE

```
#define _TIM5_CNTRL_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 counter register low byte reset value.

Definition at line 4216 of file STM8AF\_STM8S.h.

#### 5.1.2.1394 \_TIM5\_COMS

```
#define _TIM5_COMS ((uint8_t) (0x01 << 2))
```

TIM5 Capture/compare control update selection [0] (in \_TIM5\_CR2)

Definition at line 4238 of file STM8AF\_STM8S.h.

#### 5.1.2.1395 \_TIM5\_CR1

```
#define _TIM5_CR1 _SFR(uint8_t, TIM5_AddressBase+0x00)
```

TIM5 control register 1.

Definition at line 4178 of file STM8AF\_STM8S.h.

#### 5.1.2.1396 \_TIM5\_CR1\_RESET\_VALUE

```
#define _TIM5_CR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 control register 1 reset value.

Definition at line 4203 of file STM8AF\_STM8S.h.

#### 5.1.2.1397 \_TIM5\_CR2

```
#define _TIM5_CR2 _SFR(uint8_t, TIM5_AddressBase+0x01)
```

TIM5 control register 2.

Definition at line 4179 of file STM8AF\_STM8S.h.

#### 5.1.2.1398 \_TIM5\_CR2\_RESET\_VALUE

```
#define _TIM5_CR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 control register 2 reset value.

Definition at line 4204 of file STM8AF\_STM8S.h.

#### 5.1.2.1399 \_TIM5\_EGR

```
#define _TIM5_EGR _SFR(uint8_t, TIM5_AddressBase+0x06)
```

TIM5 Event generation register.

Definition at line 4184 of file STM8AF\_STM8S.h.

#### 5.1.2.1400 \_TIM5\_EGR\_RESET\_VALUE

```
#define _TIM5_EGR_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 Event generation register reset value.

Definition at line 4209 of file STM8AF\_STM8S.h.

#### 5.1.2.1401 \_TIM5\_IC1F

```
#define _TIM5_IC1F ((uint8_t) (0x0F << 4))
```

TIM5 Output compare 1 mode [3:0] (in \_TIM5\_CCMR1)

Definition at line 4309 of file STM8AF\_STM8S.h.

#### 5.1.2.1402 \_TIM5\_IC1F0

```
#define _TIM5_IC1F0 ((uint8_t) (0x01 << 4))
```

TIM5 Input capture 1 filter [0] (in \_TIM5\_CCMR1)

Definition at line 4310 of file STM8AF\_STM8S.h.

#### 5.1.2.1403 \_TIM5\_IC1F1

```
#define _TIM5_IC1F1 ((uint8_t) (0x01 << 5))
```

TIM5 Input capture 1 filter [1] (in \_TIM5\_CCMR1)

Definition at line 4311 of file STM8AF\_STM8S.h.

#### 5.1.2.1404 \_TIM5\_IC1F2

```
#define _TIM5_IC1F2 ((uint8_t) (0x01 << 6))
```

TIM5 Input capture 1 filter [2] (in \_TIM5\_CCMR1)

Definition at line 4312 of file STM8AF\_STM8S.h.

#### 5.1.2.1405 \_TIM5\_IC1F3

```
#define _TIM5_IC1F3 ((uint8_t) (0x01 << 7))
```

TIM5 Input capture 1 filter [3] (in \_TIM5\_CCMR1)

Definition at line 4313 of file STM8AF\_STM8S.h.

#### 5.1.2.1406 \_TIM5\_IC1PSC

```
#define _TIM5_IC1PSC ((uint8_t) (0x03 << 2))
```

TIM5 Input capture 1 prescaler [1:0] (in \_TIM5\_CCMR1)

Definition at line 4306 of file STM8AF\_STM8S.h.

#### 5.1.2.1407 \_TIM5\_IC1PSC0

```
#define _TIM5_IC1PSC0 ((uint8_t) (0x01 << 2))
```

TIM5 Input capture 1 prescaler [0] (in \_TIM5\_CCMR1)

Definition at line 4307 of file STM8AF\_STM8S.h.



**5.1.2.1408 \_TIM5\_IC1PSC1**

```
#define _TIM5_IC1PSC1 ((uint8_t) (0x01 << 3))
```

TIM5 Input capture 1 prescaler [1] (in \_TIM5\_CCMR1)

Definition at line 4308 of file STM8AF\_STM8S.h.

**5.1.2.1409 \_TIM5\_IC2F**

```
#define _TIM5_IC2F ((uint8_t) (0x0F << 4))
```

TIM5 Output compare 2 mode [3:0] (in \_TIM5\_CCMR2)

Definition at line 4332 of file STM8AF\_STM8S.h.

**5.1.2.1410 \_TIM5\_IC2F0**

```
#define _TIM5_IC2F0 ((uint8_t) (0x01 << 4))
```

TIM5 Input capture 2 filter [0] (in \_TIM5\_CCMR2)

Definition at line 4333 of file STM8AF\_STM8S.h.

**5.1.2.1411 \_TIM5\_IC2F1**

```
#define _TIM5_IC2F1 ((uint8_t) (0x01 << 5))
```

TIM5 Input capture 2 filter [1] (in \_TIM5\_CCMR2)

Definition at line 4334 of file STM8AF\_STM8S.h.

**5.1.2.1412 \_TIM5\_IC2F2**

```
#define _TIM5_IC2F2 ((uint8_t) (0x01 << 6))
```

TIM5 Input capture 2 filter [2] (in \_TIM5\_CCMR2)

Definition at line 4335 of file STM8AF\_STM8S.h.

**5.1.2.1413 \_TIM5\_IC2F3**

```
#define _TIM5_IC2F3 ((uint8_t) (0x01 << 7))
```

TIM5 Input capture 2 filter [3] (in \_TIM5\_CCMR2)

Definition at line 4336 of file STM8AF\_STM8S.h.

**5.1.2.1414 \_TIM5\_IC2PSC**

```
#define _TIM5_IC2PSC ((uint8_t) (0x03 << 2))
```

TIM5 Input capture 2 prescaler [1:0] (in \_TIM5\_CCMR2)

Definition at line 4329 of file STM8AF\_STM8S.h.

**5.1.2.1415 \_TIM5\_IC2PSC0**

```
#define _TIM5_IC2PSC0 ((uint8_t) (0x01 << 2))
```

TIM5 Input capture 2 prescaler [0] (in \_TIM5\_CCMR2)

Definition at line 4330 of file STM8AF\_STM8S.h.

**5.1.2.1416 \_TIM5\_IC2PSC1**

```
#define _TIM5_IC2PSC1 ((uint8_t) (0x01 << 3))
```

TIM5 Input capture 2 prescaler [1] (in \_TIM5\_CCMR2)

Definition at line 4331 of file STM8AF\_STM8S.h.

**5.1.2.1417 \_TIM5\_IC3F**

```
#define _TIM5_IC3F ((uint8_t) (0x0F << 4))
```

TIM5 Output compare 3 mode [3:0] (in \_TIM5\_CCMR3)

Definition at line 4355 of file STM8AF\_STM8S.h.

**5.1.2.1418 \_TIM5\_IC3F0**

```
#define _TIM5_IC3F0 ((uint8_t) (0x01 << 4))
```

TIM5 Input capture 3 filter [0] (in \_TIM5\_CCMR3)

Definition at line 4356 of file STM8AF\_STM8S.h.

**5.1.2.1419 \_TIM5\_IC3F1**

```
#define _TIM5_IC3F1 ((uint8_t) (0x01 << 5))
```

TIM5 Input capture 3 filter [1] (in \_TIM5\_CCMR3)

Definition at line 4357 of file STM8AF\_STM8S.h.

**5.1.2.1420 \_TIM5\_IC3F2**

```
#define _TIM5_IC3F2 ((uint8_t) (0x01 << 6))
```

TIM5 Input capture 3 filter [2] (in \_TIM5\_CCMR3)

Definition at line 4358 of file STM8AF\_STM8S.h.

**5.1.2.1421 \_TIM5\_IC3F3**

```
#define _TIM5_IC3F3 ((uint8_t) (0x01 << 7))
```

TIM5 Input capture 3 filter [3] (in \_TIM5\_CCMR3)

Definition at line 4359 of file STM8AF\_STM8S.h.

**5.1.2.1422 \_TIM5\_IC3PSC**

```
#define _TIM5_IC3PSC ((uint8_t) (0x03 << 2))
```

TIM5 Input capture 3 prescaler [1:0] (in \_TIM5\_CCMR3)

Definition at line 4352 of file STM8AF\_STM8S.h.

#### 5.1.2.1423 \_TIM5\_IC3PSC0

```
#define _TIM5_IC3PSC0 ((uint8_t) (0x01 << 2))
```

TIM5 Input capture 3 prescaler [0] (in \_TIM5\_CCMR3)

Definition at line 4353 of file STM8AF\_STM8S.h.

#### 5.1.2.1424 \_TIM5\_IC3PSC1

```
#define _TIM5_IC3PSC1 ((uint8_t) (0x01 << 3))
```

TIM5 Input capture 3 prescaler [1] (in \_TIM5\_CCMR3)

Definition at line 4354 of file STM8AF\_STM8S.h.

#### 5.1.2.1425 \_TIM5\_IER

```
#define _TIM5_IER _SFR(uint8_t, TIM5_AddressBase+0x03)
```

TIM5 interrupt enable register.

Definition at line 4181 of file STM8AF\_STM8S.h.

#### 5.1.2.1426 \_TIM5\_IER\_RESET\_VALUE

```
#define _TIM5_IER_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 interrupt enable register reset value.

Definition at line 4206 of file STM8AF\_STM8S.h.

#### 5.1.2.1427 \_TIM5\_MMS

```
#define _TIM5_MMS ((uint8_t) (0x07 << 4))
```

TIM5 Master mode selection [2:0] (in \_TIM5\_CR2)

Definition at line 4240 of file STM8AF\_STM8S.h.

**5.1.2.1428 \_TIM5\_MMS0**

```
#define _TIM5_MMS0 ((uint8_t) (0x01 << 4))
```

TIM5 Master mode selection [0] (in \_TIM5\_CR2)

Definition at line 4241 of file STM8AF\_STM8S.h.

**5.1.2.1429 \_TIM5\_MMS1**

```
#define _TIM5_MMS1 ((uint8_t) (0x01 << 5))
```

TIM5 Master mode selection [1] (in \_TIM5\_CR2)

Definition at line 4242 of file STM8AF\_STM8S.h.

**5.1.2.1430 \_TIM5\_MMS2**

```
#define _TIM5_MMS2 ((uint8_t) (0x01 << 6))
```

TIM5 Master mode selection [2] (in \_TIM5\_CR2)

Definition at line 4243 of file STM8AF\_STM8S.h.

**5.1.2.1431 \_TIM5\_MSM**

```
#define _TIM5_MSM ((uint8_t) (0x01 << 7))
```

TIM5 Master/slave mode [0] (in \_TIM5\_SMCR)

Definition at line 4256 of file STM8AF\_STM8S.h.

**5.1.2.1432 \_TIM5\_OC1M**

```
#define _TIM5_OC1M ((uint8_t) (0x07 << 4))
```

TIM5 Output compare 1 mode [2:0] (in \_TIM5\_CCMR1)

Definition at line 4298 of file STM8AF\_STM8S.h.

#### 5.1.2.1433 \_TIM5\_OC1M0

```
#define _TIM5_OC1M0 ((uint8_t) (0x01 << 4))
```

TIM5 Output compare 1 mode [0] (in \_TIM5\_CCMR1)

Definition at line 4299 of file STM8AF\_STM8S.h.

#### 5.1.2.1434 \_TIM5\_OC1M1

```
#define _TIM5_OC1M1 ((uint8_t) (0x01 << 5))
```

TIM5 Output compare 1 mode [1] (in \_TIM5\_CCMR1)

Definition at line 4300 of file STM8AF\_STM8S.h.

#### 5.1.2.1435 \_TIM5\_OC1M2

```
#define _TIM5_OC1M2 ((uint8_t) (0x01 << 6))
```

TIM5 Output compare 1 mode [2] (in \_TIM5\_CCMR1)

Definition at line 4301 of file STM8AF\_STM8S.h.

#### 5.1.2.1436 \_TIM5\_OC1PE

```
#define _TIM5_OC1PE ((uint8_t) (0x01 << 3))
```

TIM5 Output compare 1 preload enable [0] (in \_TIM5\_CCMR1)

Definition at line 4297 of file STM8AF\_STM8S.h.

#### 5.1.2.1437 \_TIM5\_OC2M

```
#define _TIM5_OC2M ((uint8_t) (0x07 << 4))
```

TIM5 Output compare 2 mode [2:0] (in \_TIM5\_CCMR2)

Definition at line 4321 of file STM8AF\_STM8S.h.

**5.1.2.1438 \_TIM5\_OC2M0**

```
#define _TIM5_OC2M0 ((uint8_t) (0x01 << 4))
```

TIM5 Output compare 2 mode [0] (in \_TIM5\_CCMR2)

Definition at line 4322 of file STM8AF\_STM8S.h.

**5.1.2.1439 \_TIM5\_OC2M1**

```
#define _TIM5_OC2M1 ((uint8_t) (0x01 << 5))
```

TIM5 Output compare 2 mode [1] (in \_TIM5\_CCMR2)

Definition at line 4323 of file STM8AF\_STM8S.h.

**5.1.2.1440 \_TIM5\_OC2M2**

```
#define _TIM5_OC2M2 ((uint8_t) (0x01 << 6))
```

TIM5 Output compare 2 mode [2] (in \_TIM5\_CCMR2)

Definition at line 4324 of file STM8AF\_STM8S.h.

**5.1.2.1441 \_TIM5\_OC2PE**

```
#define _TIM5_OC2PE ((uint8_t) (0x01 << 3))
```

TIM5 Output compare 2 preload enable [0] (in \_TIM5\_CCMR2)

Definition at line 4320 of file STM8AF\_STM8S.h.

**5.1.2.1442 \_TIM5\_OC3M**

```
#define _TIM5_OC3M ((uint8_t) (0x07 << 4))
```

TIM5 Output compare 3 mode [2:0] (in \_TIM5\_CCMR3)

Definition at line 4344 of file STM8AF\_STM8S.h.

**5.1.2.1443 \_TIM5\_OC3M0**

```
#define _TIM5_OC3M0 ((uint8_t) (0x01 << 4))
```

TIM5 Output compare 3 mode [0] (in \_TIM5\_CCMR3)

Definition at line 4345 of file STM8AF\_STM8S.h.

**5.1.2.1444 \_TIM5\_OC3M1**

```
#define _TIM5_OC3M1 ((uint8_t) (0x01 << 5))
```

TIM5 Output compare 3 mode [1] (in \_TIM5\_CCMR3)

Definition at line 4346 of file STM8AF\_STM8S.h.

**5.1.2.1445 \_TIM5\_OC3M2**

```
#define _TIM5_OC3M2 ((uint8_t) (0x01 << 6))
```

TIM5 Output compare 3 mode [2] (in \_TIM5\_CCMR3)

Definition at line 4347 of file STM8AF\_STM8S.h.

**5.1.2.1446 \_TIM5\_OC3PE**

```
#define _TIM5_OC3PE ((uint8_t) (0x01 << 3))
```

TIM5 Output compare 3 preload enable [0] (in \_TIM5\_CCMR3)

Definition at line 4343 of file STM8AF\_STM8S.h.

**5.1.2.1447 \_TIM5\_OPM**

```
#define _TIM5_OPM ((uint8_t) (0x01 << 3))
```

TIM5 One-pulse mode [0] (in \_TIM5\_CR1)

Definition at line 4231 of file STM8AF\_STM8S.h.



**5.1.2.1448 \_TIM5\_PSC**

```
#define _TIM5_PSC ((uint8_t) (0x0F << 0))
```

TIM5 clock prescaler [3:0] (in \_TIM5\_PSCR)

Definition at line 4375 of file STM8AF\_STM8S.h.

**5.1.2.1449 \_TIM5\_PSC0**

```
#define _TIM5_PSC0 ((uint8_t) (0x01 << 0))
```

TIM5 clock prescaler [0] (in \_TIM5\_PSCR)

Definition at line 4376 of file STM8AF\_STM8S.h.

**5.1.2.1450 \_TIM5\_PSC1**

```
#define _TIM5_PSC1 ((uint8_t) (0x01 << 1))
```

TIM5 clock prescaler [1] (in \_TIM5\_PSCR)

Definition at line 4377 of file STM8AF\_STM8S.h.

**5.1.2.1451 \_TIM5\_PSC2**

```
#define _TIM5_PSC2 ((uint8_t) (0x01 << 2))
```

TIM5 clock prescaler [2] (in \_TIM5\_PSCR)

Definition at line 4378 of file STM8AF\_STM8S.h.

**5.1.2.1452 \_TIM5\_PSC3**

```
#define _TIM5_PSC3 ((uint8_t) (0x01 << 3))
```

TIM5 clock prescaler [3] (in \_TIM5\_PSCR)

Definition at line 4379 of file STM8AF\_STM8S.h.

#### 5.1.2.1453 \_TIM5\_PSCR

```
#define _TIM5_PSCR _SFR(uint8_t, TIM5_AddressBase+0x0E)
```

TIM5 clock prescaler register.

Definition at line 4192 of file STM8AF\_STM8S.h.

#### 5.1.2.1454 \_TIM5\_PSCR\_RESET\_VALUE

```
#define _TIM5_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 clock prescaler register reset value.

Definition at line 4217 of file STM8AF\_STM8S.h.

#### 5.1.2.1455 \_TIM5\_SMCR

```
#define _TIM5_SMCR _SFR(uint8_t, TIM5_AddressBase+0x02)
```

TIM5 Slave mode control register.

Definition at line 4180 of file STM8AF\_STM8S.h.

#### 5.1.2.1456 \_TIM5\_SMCR\_RESET\_VALUE

```
#define _TIM5_SMCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 Slave mode control register reset value.

Definition at line 4205 of file STM8AF\_STM8S.h.

#### 5.1.2.1457 \_TIM5\_SMS

```
#define _TIM5_SMS ((uint8_t) (0x07 << 0))
```

TIM5 Clock/trigger/slave mode selection [2:0] (in \_TIM5\_SMCR)

Definition at line 4247 of file STM8AF\_STM8S.h.

**5.1.2.1458 \_TIM5\_SMS0**

```
#define _TIM5_SMS0 ((uint8_t) (0x01 << 0))
```

TIM5 Clock/trigger/slave mode selection [0] (in \_TIM5\_SMCR)

Definition at line 4248 of file STM8AF\_STM8S.h.

**5.1.2.1459 \_TIM5\_SMS1**

```
#define _TIM5_SMS1 ((uint8_t) (0x01 << 1))
```

TIM5 Clock/trigger/slave mode selection [1] (in \_TIM5\_SMCR)

Definition at line 4249 of file STM8AF\_STM8S.h.

**5.1.2.1460 \_TIM5\_SMS2**

```
#define _TIM5_SMS2 ((uint8_t) (0x01 << 2))
```

TIM5 Clock/trigger/slave mode selection [2] (in \_TIM5\_SMCR)

Definition at line 4250 of file STM8AF\_STM8S.h.

**5.1.2.1461 \_TIM5\_SR1**

```
#define _TIM5_SR1 \_SFR(uint8_t, TIM5\_AddressBase+0x04)
```

TIM5 status register 1.

Definition at line 4182 of file STM8AF\_STM8S.h.

**5.1.2.1462 \_TIM5\_SR1\_RESET\_VALUE**

```
#define _TIM5_SR1_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 status register 1 reset value.

Definition at line 4207 of file STM8AF\_STM8S.h.

#### 5.1.2.1463 \_TIM5\_SR2

```
#define _TIM5_SR2 _SFR(uint8_t, TIM5_AddressBase+0x05)
```

TIM5 status register 2.

Definition at line 4183 of file STM8AF\_STM8S.h.

#### 5.1.2.1464 \_TIM5\_SR2\_RESET\_VALUE

```
#define _TIM5_SR2_RESET_VALUE ((uint8_t) 0x00)
```

TIM5 status register 2 reset value.

Definition at line 4208 of file STM8AF\_STM8S.h.

#### 5.1.2.1465 \_TIM5\_TG

```
#define _TIM5_TG ((uint8_t) (0x01 << 6))
```

TIM5 Trigger generation [0] (in \_TIM5\_EGR)

Definition at line 4289 of file STM8AF\_STM8S.h.

#### 5.1.2.1466 \_TIM5\_TIE

```
#define _TIM5_TIE ((uint8_t) (0x01 << 6))
```

TIM5 Trigger interrupt enable [0] (in \_TIM5\_IER)

Definition at line 4264 of file STM8AF\_STM8S.h.

#### 5.1.2.1467 \_TIM5\_TIF

```
#define _TIM5_TIF ((uint8_t) (0x01 << 6))
```

TIM5 Trigger interrupt flag [0] (in \_TIM5\_SR1)

Definition at line 4273 of file STM8AF\_STM8S.h.

**5.1.2.1468 \_TIM5\_TS**

```
#define _TIM5_TS ((uint8_t) (0x07 << 4))
```

TIM5 Trigger selection [2:0] (in \_TIM5\_SMCR)

Definition at line 4252 of file STM8AF\_STM8S.h.

**5.1.2.1469 \_TIM5\_TS0**

```
#define _TIM5_TS0 ((uint8_t) (0x01 << 4))
```

TIM5 Trigger selection [0] (in \_TIM5\_SMCR)

Definition at line 4253 of file STM8AF\_STM8S.h.

**5.1.2.1470 \_TIM5\_TS1**

```
#define _TIM5_TS1 ((uint8_t) (0x01 << 5))
```

TIM5 Trigger selection [1] (in \_TIM5\_SMCR)

Definition at line 4254 of file STM8AF\_STM8S.h.

**5.1.2.1471 \_TIM5\_TS2**

```
#define _TIM5_TS2 ((uint8_t) (0x01 << 6))
```

TIM5 Trigger selection [2] (in \_TIM5\_SMCR)

Definition at line 4255 of file STM8AF\_STM8S.h.

**5.1.2.1472 \_TIM5\_UDIS**

```
#define _TIM5_UDIS ((uint8_t) (0x01 << 1))
```

TIM5 Update disable [0] (in \_TIM5\_CR1)

Definition at line 4229 of file STM8AF\_STM8S.h.

#### 5.1.2.1473 \_TIM5\_UG

```
#define _TIM5_UG ((uint8_t) (0x01 << 0))
```

TIM5 Update generation [0] (in \_TIM5\_EGR)

Definition at line 4284 of file STM8AF\_STM8S.h.

#### 5.1.2.1474 \_TIM5\_UIE

```
#define _TIM5_UIE ((uint8_t) (0x01 << 0))
```

TIM5 Update interrupt enable [0] (in \_TIM5\_IER)

Definition at line 4259 of file STM8AF\_STM8S.h.

#### 5.1.2.1475 \_TIM5\_UIF

```
#define _TIM5_UIF ((uint8_t) (0x01 << 0))
```

TIM5 Update interrupt flag [0] (in \_TIM5\_SR1)

Definition at line 4268 of file STM8AF\_STM8S.h.

#### 5.1.2.1476 \_TIM5\_URS

```
#define _TIM5_URS ((uint8_t) (0x01 << 2))
```

TIM5 Update request source [0] (in \_TIM5\_CR1)

Definition at line 4230 of file STM8AF\_STM8S.h.

#### 5.1.2.1477 \_TIM6

```
#define _TIM6 _SFR(TIM6_t, TIM6_AddressBase)
```

TIM6 struct/bit access.

Definition at line 4464 of file STM8AF\_STM8S.h.

**5.1.2.1478 \_TIM6\_ARPE**

```
#define _TIM6_ARPE ((uint8_t) (0x01 << 7))
```

TIM6 Auto-reload preload enable [0] (in \_TIM6\_CR1)

Definition at line 4488 of file STM8AF\_STM8S.h.

**5.1.2.1479 \_TIM6\_ARR**

```
#define _TIM6_ARR _SFR(uint8_t, TIM6_AddressBase+0x06)
```

TIM6 auto-reload register.

Definition at line 4471 of file STM8AF\_STM8S.h.

**5.1.2.1480 \_TIM6\_ARR\_RESET\_VALUE**

```
#define _TIM6_ARR_RESET_VALUE ((uint8_t) 0xFF)
```

TIM6 auto-reload register reset value.

Definition at line 4480 of file STM8AF\_STM8S.h.

**5.1.2.1481 \_TIM6\_CEN**

```
#define _TIM6_CEN ((uint8_t) (0x01 << 0))
```

TIM6 Counter enable [0] (in \_TIM6\_CR1)

Definition at line 4483 of file STM8AF\_STM8S.h.

**5.1.2.1482 \_TIM6\_CNTR**

```
#define _TIM6_CNTR _SFR(uint8_t, TIM6_AddressBase+0x04)
```

TIM6 counter register.

Definition at line 4469 of file STM8AF\_STM8S.h.

#### 5.1.2.1483 \_TIM6\_CNTR\_RESET\_VALUE

```
#define _TIM6_CNTR_RESET_VALUE ((uint8_t) 0x00)
```

TIM6 counter register reset value.

Definition at line 4478 of file STM8AF\_STM8S.h.

#### 5.1.2.1484 \_TIM6\_CR

```
#define _TIM6_CR _SFR(uint8_t, TIM6_AddressBase+0x00)
```

TIM6 control register.

Definition at line 4465 of file STM8AF\_STM8S.h.

#### 5.1.2.1485 \_TIM6\_CR\_RESET\_VALUE

```
#define _TIM6_CR_RESET_VALUE ((uint8_t) 0x00)
```

TIM6 control register reset value.

Definition at line 4474 of file STM8AF\_STM8S.h.

#### 5.1.2.1486 \_TIM6\_EGR

```
#define _TIM6_EGR _SFR(uint8_t, TIM6_AddressBase+0x03)
```

TIM6 event generation register.

Definition at line 4468 of file STM8AF\_STM8S.h.

#### 5.1.2.1487 \_TIM6\_EGR\_RESET\_VALUE

```
#define _TIM6_EGR_RESET_VALUE ((uint8_t) 0x00)
```

TIM6 event generation register reset value.

Definition at line 4477 of file STM8AF\_STM8S.h.



#### 5.1.2.1488 \_TIM6\_IER

```
#define _TIM6_IER _SFR(uint8_t, TIM6_AddressBase+0x01)
```

TIM6 interrupt enable register.

Definition at line 4466 of file STM8AF\_STM8S.h.

#### 5.1.2.1489 \_TIM6\_IER\_RESET\_VALUE

```
#define _TIM6_IER_RESET_VALUE ((uint8_t) 0x00)
```

TIM6 interrupt enable register reset value.

Definition at line 4475 of file STM8AF\_STM8S.h.

#### 5.1.2.1490 \_TIM6\_MMS

```
#define _TIM6_MMS ((uint8_t) (0x07 << 4))
```

TIM6 Master mode selection [2:0] (in \_TIM6\_CR2)

Definition at line 4492 of file STM8AF\_STM8S.h.

#### 5.1.2.1491 \_TIM6\_MMS0

```
#define _TIM6_MMS0 ((uint8_t) (0x01 << 4))
```

TIM6 Master mode selection [0] (in \_TIM6\_CR2)

Definition at line 4493 of file STM8AF\_STM8S.h.

#### 5.1.2.1492 \_TIM6\_MMS1

```
#define _TIM6_MMS1 ((uint8_t) (0x01 << 5))
```

TIM6 Master mode selection [1] (in \_TIM6\_CR2)

Definition at line 4494 of file STM8AF\_STM8S.h.

**5.1.2.1493 \_TIM6\_MMS2**

```
#define _TIM6_MMS2 ((uint8_t) (0x01 << 6))
```

TIM6 Master mode selection [2] (in \_TIM6\_CR2)

Definition at line 4495 of file STM8AF\_STM8S.h.

**5.1.2.1494 \_TIM6\_OPM**

```
#define _TIM6_OPM ((uint8_t) (0x01 << 3))
```

TIM6 One-pulse mode [0] (in \_TIM6\_CR1)

Definition at line 4486 of file STM8AF\_STM8S.h.

**5.1.2.1495 \_TIM6\_PSC**

```
#define _TIM6_PSC ((uint8_t) (0x07 << 0))
```

TIM6 clock prescaler [2:0] (in \_TIM6\_PSCR)

Definition at line 4523 of file STM8AF\_STM8S.h.

**5.1.2.1496 \_TIM6\_PSC0**

```
#define _TIM6_PSC0 ((uint8_t) (0x01 << 0))
```

TIM6 clock prescaler [0] (in \_TIM6\_PSCR)

Definition at line 4524 of file STM8AF\_STM8S.h.

**5.1.2.1497 \_TIM6\_PSC1**

```
#define _TIM6_PSC1 ((uint8_t) (0x01 << 1))
```

TIM6 clock prescaler [1] (in \_TIM6\_PSCR)

Definition at line 4525 of file STM8AF\_STM8S.h.

**5.1.2.1498 \_TIM6\_PSC2**

```
#define _TIM6_PSC2 ((uint8_t) (0x01 << 2))
```

TIM6 clock prescaler [2] (in \_TIM6\_PSCR)

Definition at line 4526 of file STM8AF\_STM8S.h.

**5.1.2.1499 \_TIM6\_PSCR**

```
#define _TIM6_PSCR _SFR(uint8_t, TIM6_AddressBase+0x05)
```

TIM6 clock prescaler register.

Definition at line 4470 of file STM8AF\_STM8S.h.

**5.1.2.1500 \_TIM6\_PSCR\_RESET\_VALUE**

```
#define _TIM6_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

TIM6 clock prescaler register reset value.

Definition at line 4479 of file STM8AF\_STM8S.h.

**5.1.2.1501 \_TIM6\_SMS**

```
#define _TIM6_SMS ((uint8_t) (0x07 << 0))
```

TIM6 Clock/trigger/slave mode selection [2:0] (in \_TIM6\_SMCR)

Definition at line 4499 of file STM8AF\_STM8S.h.

**5.1.2.1502 \_TIM6\_SMS0**

```
#define _TIM6_SMS0 ((uint8_t) (0x01 << 0))
```

TIM6 Clock/trigger/slave mode selection [0] (in \_TIM6\_SMCR)

Definition at line 4500 of file STM8AF\_STM8S.h.

#### 5.1.2.1503 \_TIM6\_SMS1

```
#define _TIM6_SMS1 ((uint8_t) (0x01 << 1))
```

TIM6 Clock/trigger/slave mode selection [1] (in \_TIM6\_SMCR)

Definition at line 4501 of file STM8AF\_STM8S.h.

#### 5.1.2.1504 \_TIM6\_SMS2

```
#define _TIM6_SMS2 ((uint8_t) (0x01 << 2))
```

TIM6 Clock/trigger/slave mode selection [2] (in \_TIM6\_SMCR)

Definition at line 4502 of file STM8AF\_STM8S.h.

#### 5.1.2.1505 \_TIM6\_SR

```
#define _TIM6_SR _SFR(uint8_t, TIM6_AddressBase+0x02)
```

TIM6 status register.

Definition at line 4467 of file STM8AF\_STM8S.h.

#### 5.1.2.1506 \_TIM6\_SR\_RESET\_VALUE

```
#define _TIM6_SR_RESET_VALUE ((uint8_t) 0x00)
```

TIM6 status register reset value.

Definition at line 4476 of file STM8AF\_STM8S.h.

#### 5.1.2.1507 \_TIM6\_TS

```
#define _TIM6_TS ((uint8_t) (0x07 << 4))
```

TIM6 Trigger selection [2:0] (in \_TIM6\_SMCR)

Definition at line 4504 of file STM8AF\_STM8S.h.

**5.1.2.1508 \_TIM6\_TS0**

```
#define _TIM6_TS0 ((uint8_t) (0x01 << 4))
```

TIM6 Trigger selection [0] (in \_TIM6\_SMCR)

Definition at line 4505 of file STM8AF\_STM8S.h.

**5.1.2.1509 \_TIM6\_TS1**

```
#define _TIM6_TS1 ((uint8_t) (0x01 << 5))
```

TIM6 Trigger selection [1] (in \_TIM6\_SMCR)

Definition at line 4506 of file STM8AF\_STM8S.h.

**5.1.2.1510 \_TIM6\_TS2**

```
#define _TIM6_TS2 ((uint8_t) (0x01 << 6))
```

TIM6 Trigger selection [2] (in \_TIM6\_SMCR)

Definition at line 4507 of file STM8AF\_STM8S.h.

**5.1.2.1511 \_TIM6\_UDIS**

```
#define _TIM6_UDIS ((uint8_t) (0x01 << 1))
```

TIM6 Update disable [0] (in \_TIM6\_CR1)

Definition at line 4484 of file STM8AF\_STM8S.h.

**5.1.2.1512 \_TIM6\_UG**

```
#define _TIM6_UG ((uint8_t) (0x01 << 0))
```

TIM6 Update generation [0] (in \_TIM6\_EGR)

Definition at line 4519 of file STM8AF\_STM8S.h.

#### 5.1.2.1513 \_TIM6\_UIE

```
#define _TIM6_UIE ((uint8_t) (0x01 << 0))
```

TIM6 Update interrupt enable [0] (in \_TIM6\_IER)

Definition at line 4511 of file STM8AF\_STM8S.h.

#### 5.1.2.1514 \_TIM6\_UIF

```
#define _TIM6_UIF ((uint8_t) (0x01 << 0))
```

TIM6 Update interrupt flag [0] (in \_TIM6\_SR)

Definition at line 4515 of file STM8AF\_STM8S.h.

#### 5.1.2.1515 \_TIM6\_URS

```
#define _TIM6_URS ((uint8_t) (0x01 << 2))
```

TIM6 Update request source [0] (in \_TIM6\_CR1)

Definition at line 4485 of file STM8AF\_STM8S.h.

#### 5.1.2.1516 \_UART1

```
#define _UART1 _SFR(UART1_t, UART1_AddressBase)
```

UART1 struct/bit access.

Definition at line 1731 of file STM8AF\_STM8S.h.

#### 5.1.2.1517 \_UART1\_ADD

```
#define _UART1_ADD ((uint8_t) (0x0F << 0))
```

UART1 Address of the UART node [3:0] (in \_UART1\_CR4)

Definition at line 1798 of file STM8AF\_STM8S.h.

**5.1.2.1518 \_UART1\_ADD0**

```
#define _UART1_ADD0 ((uint8_t) (0x01 << 0))
```

UART1 Address of the UART node [0] (in \_UART1\_CR4)

Definition at line 1799 of file STM8AF\_STM8S.h.

**5.1.2.1519 \_UART1\_ADD1**

```
#define _UART1_ADD1 ((uint8_t) (0x01 << 1))
```

UART1 Address of the UART node [1] (in \_UART1\_CR4)

Definition at line 1800 of file STM8AF\_STM8S.h.

**5.1.2.1520 \_UART1\_ADD2**

```
#define _UART1_ADD2 ((uint8_t) (0x01 << 2))
```

UART1 Address of the UART node [2] (in \_UART1\_CR4)

Definition at line 1801 of file STM8AF\_STM8S.h.

**5.1.2.1521 \_UART1\_ADD3**

```
#define _UART1_ADD3 ((uint8_t) (0x01 << 3))
```

UART1 Address of the UART node [3] (in \_UART1\_CR4)

Definition at line 1802 of file STM8AF\_STM8S.h.

**5.1.2.1522 \_UART1\_BRR1**

```
#define _UART1_BRR1 \_SFR(uint8_t, UART1\_AddressBase+0x02)
```

UART1 Baud rate register 1.

Definition at line 1734 of file STM8AF\_STM8S.h.

#### 5.1.2.1523 \_UART1\_BRR1\_RESET\_VALUE

```
#define _UART1_BRR1_RESET_VALUE ((uint8_t) 0x00)
```

UART1 Baud rate register 1 reset value.

Definition at line 1746 of file STM8AF\_STM8S.h.

#### 5.1.2.1524 \_UART1\_BRR2

```
#define _UART1_BRR2 __SFR(uint8_t, UART1_AddressBase+0x03)
```

UART1 Baud rate register 2.

Definition at line 1735 of file STM8AF\_STM8S.h.

#### 5.1.2.1525 \_UART1\_BRR2\_RESET\_VALUE

```
#define _UART1_BRR2_RESET_VALUE ((uint8_t) 0x00)
```

UART1 Baud rate register 2 reset value.

Definition at line 1747 of file STM8AF\_STM8S.h.

#### 5.1.2.1526 \_UART1\_CKEN

```
#define _UART1_CKEN ((uint8_t) (0x01 << 3))
```

UART1 Clock enable [0] (in \_UART1\_CR3)

Definition at line 1790 of file STM8AF\_STM8S.h.

#### 5.1.2.1527 \_UART1\_CPHA

```
#define _UART1_CPHA ((uint8_t) (0x01 << 1))
```

UART1 Clock phase [0] (in \_UART1\_CR3)

Definition at line 1788 of file STM8AF\_STM8S.h.



**5.1.2.1528 \_UART1\_CPOL**

```
#define _UART1_CPOL ((uint8_t) (0x01 << 2))
```

UART1 Clock polarity [0] (in \_UART1\_CR3)

Definition at line 1789 of file STM8AF\_STM8S.h.

**5.1.2.1529 \_UART1\_CR1**

```
#define _UART1_CR1 _SFR(uint8_t, UART1_AddressBase+0x04)
```

UART1 Control register 1.

Definition at line 1736 of file STM8AF\_STM8S.h.

**5.1.2.1530 \_UART1\_CR1\_RESET\_VALUE**

```
#define _UART1_CR1_RESET_VALUE ((uint8_t) 0x00)
```

UART1 Control register 1 reset value.

Definition at line 1748 of file STM8AF\_STM8S.h.

**5.1.2.1531 \_UART1\_CR2**

```
#define _UART1_CR2 _SFR(uint8_t, UART1_AddressBase+0x05)
```

UART1 Control register 2.

Definition at line 1737 of file STM8AF\_STM8S.h.

**5.1.2.1532 \_UART1\_CR2\_RESET\_VALUE**

```
#define _UART1_CR2_RESET_VALUE ((uint8_t) 0x00)
```

UART1 Control register 2 reset value.

Definition at line 1749 of file STM8AF\_STM8S.h.

#### 5.1.2.1533 \_UART1\_CR3

```
#define _UART1_CR3 _SFR(uint8_t, UART1_AddressBase+0x06)
```

UART1 Control register 3.

Definition at line 1738 of file STM8AF\_STM8S.h.

#### 5.1.2.1534 \_UART1\_CR3\_RESET\_VALUE

```
#define _UART1_CR3_RESET_VALUE ((uint8_t) 0x00)
```

UART1 Control register 3 reset value.

Definition at line 1750 of file STM8AF\_STM8S.h.

#### 5.1.2.1535 \_UART1\_CR4

```
#define _UART1_CR4 _SFR(uint8_t, UART1_AddressBase+0x07)
```

UART1 Control register 4.

Definition at line 1739 of file STM8AF\_STM8S.h.

#### 5.1.2.1536 \_UART1\_CR4\_RESET\_VALUE

```
#define _UART1_CR4_RESET_VALUE ((uint8_t) 0x00)
```

UART1 Control register 4 reset value.

Definition at line 1751 of file STM8AF\_STM8S.h.

#### 5.1.2.1537 \_UART1\_CR5

```
#define _UART1_CR5 _SFR(uint8_t, UART1_AddressBase+0x08)
```

UART1 Control register 5.

Definition at line 1740 of file STM8AF\_STM8S.h.

**5.1.2.1538 \_UART1\_CR5\_RESET\_VALUE**

```
#define _UART1_CR5_RESET_VALUE ((uint8_t) 0x00)
```

UART1 Control register 5 reset value.

Definition at line 1752 of file STM8AF\_STM8S.h.

**5.1.2.1539 \_UART1\_DR**

```
#define _UART1_DR _SFR(uint8_t, UART1_AddressBase+0x01)
```

UART1 data register.

Definition at line 1733 of file STM8AF\_STM8S.h.

**5.1.2.1540 \_UART1\_FE**

```
#define _UART1_FE ((uint8_t) (0x01 << 1))
```

UART1 Framing error [0] (in \_UART1\_SR)

Definition at line 1758 of file STM8AF\_STM8S.h.

**5.1.2.1541 \_UART1\_GTR**

```
#define _UART1_GTR _SFR(uint8_t, UART1_AddressBase+0x09)
```

UART1 guard time register.

Definition at line 1741 of file STM8AF\_STM8S.h.

**5.1.2.1542 \_UART1\_GTR\_RESET\_VALUE**

```
#define _UART1_GTR_RESET_VALUE ((uint8_t) 0x00)
```

UART1 guard time register reset value.

Definition at line 1753 of file STM8AF\_STM8S.h.

#### 5.1.2.1543 \_UART1\_HDSEL

```
#define _UART1_HDSEL ((uint8_t) (0x01 << 3))
```

UART1 Half-Duplex Selection [0] (in \_UART1\_CR5)

Definition at line 1812 of file STM8AF\_STM8S.h.

#### 5.1.2.1544 \_UART1\_IDLE

```
#define _UART1_IDLE ((uint8_t) (0x01 << 4))
```

UART1 IDLE line detected [0] (in \_UART1\_SR)

Definition at line 1761 of file STM8AF\_STM8S.h.

#### 5.1.2.1545 \_UART1\_ILIEN

```
#define _UART1_ILIEN ((uint8_t) (0x01 << 4))
```

UART1 IDLE Line interrupt enable [0] (in \_UART1\_CR2)

Definition at line 1781 of file STM8AF\_STM8S.h.

#### 5.1.2.1546 \_UART1\_IREN

```
#define _UART1_IREN ((uint8_t) (0x01 << 1))
```

UART1 IrDA mode Enable [0] (in \_UART1\_CR5)

Definition at line 1810 of file STM8AF\_STM8S.h.

#### 5.1.2.1547 \_UART1\_IRLP

```
#define _UART1_IRLP ((uint8_t) (0x01 << 2))
```

UART1 IrDA Low Power [0] (in \_UART1\_CR5)

Definition at line 1811 of file STM8AF\_STM8S.h.

**5.1.2.1548 \_UART1\_LBCL**

```
#define _UART1_LBCL ((uint8_t) (0x01 << 0))
```

UART1 Last bit clock pulse [0] (in \_UART1\_CR3)

Definition at line 1787 of file STM8AF\_STM8S.h.

**5.1.2.1549 \_UART1\_LBDF**

```
#define _UART1_LBDF ((uint8_t) (0x01 << 4))
```

UART1 LIN Break Detection Flag [0] (in \_UART1\_CR4)

Definition at line 1803 of file STM8AF\_STM8S.h.

**5.1.2.1550 \_UART1\_LBDIEN**

```
#define _UART1_LBDIEN ((uint8_t) (0x01 << 6))
```

UART1 LIN Break Detection Interrupt Enable [0] (in \_UART1\_CR4)

Definition at line 1805 of file STM8AF\_STM8S.h.

**5.1.2.1551 \_UART1\_LBDL**

```
#define _UART1_LBDL ((uint8_t) (0x01 << 5))
```

UART1 LIN Break Detection Length [0] (in \_UART1\_CR4)

Definition at line 1804 of file STM8AF\_STM8S.h.

**5.1.2.1552 \_UART1\_LINEN**

```
#define _UART1_LINEN ((uint8_t) (0x01 << 6))
```

UART1 LIN mode enable [0] (in \_UART1\_CR3)

Definition at line 1794 of file STM8AF\_STM8S.h.

**5.1.2.1553 \_UART1\_M**

```
#define _UART1_M ((uint8_t) (0x01 << 4))
```

UART1 word length [0] (in \_UART1\_CR1)

Definition at line 1771 of file STM8AF\_STM8S.h.

**5.1.2.1554 \_UART1\_NACK**

```
#define _UART1_NACK ((uint8_t) (0x01 << 4))
```

UART1 Smartcard NACK enable [0] (in \_UART1\_CR5)

Definition at line 1813 of file STM8AF\_STM8S.h.

**5.1.2.1555 \_UART1\_NF**

```
#define _UART1_NF ((uint8_t) (0x01 << 2))
```

UART1 Noise flag [0] (in \_UART1\_SR)

Definition at line 1759 of file STM8AF\_STM8S.h.

**5.1.2.1556 \_UART1\_OR\_LHE**

```
#define _UART1_OR_LHE ((uint8_t) (0x01 << 3))
```

UART1 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART1\_SR)

Definition at line 1760 of file STM8AF\_STM8S.h.

**5.1.2.1557 \_UART1\_PCEN**

```
#define _UART1_PCEN ((uint8_t) (0x01 << 2))
```

UART1 Parity control enable [0] (in \_UART1\_CR1)

Definition at line 1769 of file STM8AF\_STM8S.h.

**5.1.2.1558 \_UART1\_PE**

```
#define _UART1_PE ((uint8_t) (0x01 << 0))
```

UART1 Parity error [0] (in \_UART1\_SR)

Definition at line 1757 of file STM8AF\_STM8S.h.

**5.1.2.1559 \_UART1\_PIEN**

```
#define _UART1_PIEN ((uint8_t) (0x01 << 0))
```

UART1 Parity interrupt enable [0] (in \_UART1\_CR1)

Definition at line 1767 of file STM8AF\_STM8S.h.

**5.1.2.1560 \_UART1\_PS**

```
#define _UART1_PS ((uint8_t) (0x01 << 1))
```

UART1 Parity selection [0] (in \_UART1\_CR1)

Definition at line 1768 of file STM8AF\_STM8S.h.

**5.1.2.1561 \_UART1\_PSCR**

```
#define _UART1_PSCR _SFR(uint8_t, UART1_AddressBase+0x0A)
```

UART1 prescaler register.

Definition at line 1742 of file STM8AF\_STM8S.h.

**5.1.2.1562 \_UART1\_PSCR\_RESET\_VALUE**

```
#define _UART1_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

UART1 prescaler register reset value.

Definition at line 1754 of file STM8AF\_STM8S.h.

**5.1.2.1563 \_UART1\_R8**

```
#define _UART1_R8 ((uint8_t) (0x01 << 7))
```

UART1 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART1\_CR1)

Definition at line 1774 of file STM8AF\_STM8S.h.

**5.1.2.1564 \_UART1\_REN**

```
#define _UART1_REN ((uint8_t) (0x01 << 2))
```

UART1 Receiver enable [0] (in \_UART1\_CR2)

Definition at line 1779 of file STM8AF\_STM8S.h.

**5.1.2.1565 \_UART1\_RIEN**

```
#define _UART1_RIEN ((uint8_t) (0x01 << 5))
```

UART1 Receiver interrupt enable [0] (in \_UART1\_CR2)

Definition at line 1782 of file STM8AF\_STM8S.h.

**5.1.2.1566 \_UART1\_RWU**

```
#define _UART1_RWU ((uint8_t) (0x01 << 1))
```

UART1 Receiver wakeup [0] (in \_UART1\_CR2)

Definition at line 1778 of file STM8AF\_STM8S.h.

**5.1.2.1567 \_UART1\_RXNE**

```
#define _UART1_RXNE ((uint8_t) (0x01 << 5))
```

UART1 Read data register not empty [0] (in \_UART1\_SR)

Definition at line 1762 of file STM8AF\_STM8S.h.



**5.1.2.1568 \_UART1\_SBK**

```
#define _UART1_SBK ((uint8_t) (0x01 << 0))
```

UART1 Send break [0] (in \_UART1\_CR2)

Definition at line 1777 of file STM8AF\_STM8S.h.

**5.1.2.1569 \_UART1\_SCEN**

```
#define _UART1_SCEN ((uint8_t) (0x01 << 5))
```

UART1 Smartcard mode enable [0] (in \_UART1\_CR5)

Definition at line 1814 of file STM8AF\_STM8S.h.

**5.1.2.1570 \_UART1\_SR**

```
#define _UART1_SR _SFR(uint8_t, UART1_AddressBase+0x00)
```

UART1 Status register.

Definition at line 1732 of file STM8AF\_STM8S.h.

**5.1.2.1571 \_UART1\_SR\_RESET\_VALUE**

```
#define _UART1_SR_RESET_VALUE ((uint8_t) 0xC0)
```

UART1 Status register reset value.

Definition at line 1745 of file STM8AF\_STM8S.h.

**5.1.2.1572 \_UART1\_STOP**

```
#define _UART1_STOP ((uint8_t) (0x03 << 4))
```

UART1 STOP bits [1:0] (in \_UART1\_CR3)

Definition at line 1791 of file STM8AF\_STM8S.h.

**5.1.2.1573 \_UART1\_STOP0**

```
#define _UART1_STOP0 ((uint8_t) (0x01 << 4))
```

UART1 STOP bits [0] (in \_UART1\_CR3)

Definition at line 1792 of file STM8AF\_STM8S.h.

**5.1.2.1574 \_UART1\_STOP1**

```
#define _UART1_STOP1 ((uint8_t) (0x01 << 5))
```

UART1 STOP bits [1] (in \_UART1\_CR3)

Definition at line 1793 of file STM8AF\_STM8S.h.

**5.1.2.1575 \_UART1\_T8**

```
#define _UART1_T8 ((uint8_t) (0x01 << 6))
```

UART1 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART1\_CR1)

Definition at line 1773 of file STM8AF\_STM8S.h.

**5.1.2.1576 \_UART1\_TC**

```
#define _UART1_TC ((uint8_t) (0x01 << 6))
```

UART1 Transmission complete [0] (in \_UART1\_SR)

Definition at line 1763 of file STM8AF\_STM8S.h.

**5.1.2.1577 \_UART1\_TCIEN**

```
#define _UART1_TCIEN ((uint8_t) (0x01 << 6))
```

UART1 Transmission complete interrupt enable [0] (in \_UART1\_CR2)

Definition at line 1783 of file STM8AF\_STM8S.h.

**5.1.2.1578 \_UART1\_TEN**

```
#define _UART1_TEN ((uint8_t) (0x01 << 3))
```

UART1 Transmitter enable [0] (in \_UART1\_CR2)

Definition at line 1780 of file STM8AF\_STM8S.h.

**5.1.2.1579 \_UART1\_TIEN**

```
#define _UART1_TIEN ((uint8_t) (0x01 << 7))
```

UART1 Transmitter interrupt enable [0] (in \_UART1\_CR2)

Definition at line 1784 of file STM8AF\_STM8S.h.

**5.1.2.1580 \_UART1\_TXE**

```
#define _UART1_TXE ((uint8_t) (0x01 << 7))
```

UART1 Transmit data register empty [0] (in \_UART1\_SR)

Definition at line 1764 of file STM8AF\_STM8S.h.

**5.1.2.1581 \_UART1\_UARTD**

```
#define _UART1_UARTD ((uint8_t) (0x01 << 5))
```

UART1 Disable (for low power consumption) [0] (in \_UART1\_CR1)

Definition at line 1772 of file STM8AF\_STM8S.h.

**5.1.2.1582 \_UART1\_WAKE**

```
#define _UART1_WAKE ((uint8_t) (0x01 << 3))
```

UART1 Wakeup method [0] (in \_UART1\_CR1)

Definition at line 1770 of file STM8AF\_STM8S.h.

#### 5.1.2.1583 \_UART2

```
#define _UART2 _SFR(UART2_t, UART2_AddressBase)
```

UART2 struct/bit access.

Definition at line 1948 of file STM8AF\_STM8S.h.

#### 5.1.2.1584 \_UART2\_ADD

```
#define _UART2_ADD ((uint8_t) (0x0F << 0))
```

UART2 Address of the UART node [3:0] (in \_UART2\_CR4)

Definition at line 2017 of file STM8AF\_STM8S.h.

#### 5.1.2.1585 \_UART2\_ADD0

```
#define _UART2_ADD0 ((uint8_t) (0x01 << 0))
```

UART2 Address of the UART node [0] (in \_UART2\_CR4)

Definition at line 2018 of file STM8AF\_STM8S.h.

#### 5.1.2.1586 \_UART2\_ADD1

```
#define _UART2_ADD1 ((uint8_t) (0x01 << 1))
```

UART2 Address of the UART node [1] (in \_UART2\_CR4)

Definition at line 2019 of file STM8AF\_STM8S.h.

#### 5.1.2.1587 \_UART2\_ADD2

```
#define _UART2_ADD2 ((uint8_t) (0x01 << 2))
```

UART2 Address of the UART node [2] (in \_UART2\_CR4)

Definition at line 2020 of file STM8AF\_STM8S.h.

**5.1.2.1588 \_UART2\_ADD3**

```
#define _UART2_ADD3 ((uint8_t) (0x01 << 3))
```

UART2 Address of the UART node [3] (in \_UART2\_CR4)

Definition at line 2021 of file STM8AF\_STM8S.h.

**5.1.2.1589 \_UART2\_BRR1**

```
#define _UART2_BRR1 _SFR(uint8_t, UART2_AddressBase+0x02)
```

UART2 Baud rate register 1.

Definition at line 1951 of file STM8AF\_STM8S.h.

**5.1.2.1590 \_UART2\_BRR1\_RESET\_VALUE**

```
#define _UART2_BRR1_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Baud rate register 1 reset value.

Definition at line 1964 of file STM8AF\_STM8S.h.

**5.1.2.1591 \_UART2\_BRR2**

```
#define _UART2_BRR2 _SFR(uint8_t, UART2_AddressBase+0x03)
```

UART2 Baud rate register 2.

Definition at line 1952 of file STM8AF\_STM8S.h.

**5.1.2.1592 \_UART2\_BRR2\_RESET\_VALUE**

```
#define _UART2_BRR2_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Baud rate register 2 reset value.

Definition at line 1965 of file STM8AF\_STM8S.h.

#### 5.1.2.1593 \_UART2\_CKEN

```
#define _UART2_CKEN ((uint8_t) (0x01 << 3))
```

UART2 Clock enable [0] (in \_UART2\_CR3)

Definition at line 2009 of file STM8AF\_STM8S.h.

#### 5.1.2.1594 \_UART2\_CPHA

```
#define _UART2_CPHA ((uint8_t) (0x01 << 1))
```

UART2 Clock phase [0] (in \_UART2\_CR3)

Definition at line 2007 of file STM8AF\_STM8S.h.

#### 5.1.2.1595 \_UART2\_CPOL

```
#define _UART2_CPOL ((uint8_t) (0x01 << 2))
```

UART2 Clock polarity [0] (in \_UART2\_CR3)

Definition at line 2008 of file STM8AF\_STM8S.h.

#### 5.1.2.1596 \_UART2\_CR1

```
#define _UART2_CR1 \_SFR(uint8_t, UART2\_AddressBase+0x04)
```

UART2 Control register 1.

Definition at line 1953 of file STM8AF\_STM8S.h.

#### 5.1.2.1597 \_UART2\_CR1\_RESET\_VALUE

```
#define _UART2_CR1_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Control register 1 reset value.

Definition at line 1966 of file STM8AF\_STM8S.h.

**5.1.2.1598 \_UART2\_CR2**

```
#define _UART2_CR2 _SFR(uint8_t, UART2_AddressBase+0x05)
```

UART2 Control register 2.

Definition at line 1954 of file STM8AF\_STM8S.h.

**5.1.2.1599 \_UART2\_CR2\_RESET\_VALUE**

```
#define _UART2_CR2_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Control register 2 reset value.

Definition at line 1967 of file STM8AF\_STM8S.h.

**5.1.2.1600 \_UART2\_CR3**

```
#define _UART2_CR3 _SFR(uint8_t, UART2_AddressBase+0x06)
```

UART2 Control register 3.

Definition at line 1955 of file STM8AF\_STM8S.h.

**5.1.2.1601 \_UART2\_CR3\_RESET\_VALUE**

```
#define _UART2_CR3_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Control register 3 reset value.

Definition at line 1968 of file STM8AF\_STM8S.h.

**5.1.2.1602 \_UART2\_CR4**

```
#define _UART2_CR4 _SFR(uint8_t, UART2_AddressBase+0x07)
```

UART2 Control register 4.

Definition at line 1956 of file STM8AF\_STM8S.h.

#### 5.1.2.1603 \_UART2\_CR4\_RESET\_VALUE

```
#define _UART2_CR4_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Control register 4 reset value.

Definition at line 1969 of file STM8AF\_STM8S.h.

#### 5.1.2.1604 \_UART2\_CR5

```
#define _UART2_CR5 _SFR(uint8_t, UART2_AddressBase+0x08)
```

UART2 Control register 5.

Definition at line 1957 of file STM8AF\_STM8S.h.

#### 5.1.2.1605 \_UART2\_CR5\_RESET\_VALUE

```
#define _UART2_CR5_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Control register 5 reset value.

Definition at line 1970 of file STM8AF\_STM8S.h.

#### 5.1.2.1606 \_UART2\_CR6

```
#define _UART2_CR6 _SFR(uint8_t, UART2_AddressBase+0x09)
```

UART2 Control register 6.

Definition at line 1958 of file STM8AF\_STM8S.h.

#### 5.1.2.1607 \_UART2\_CR6\_RESET\_VALUE

```
#define _UART2_CR6_RESET_VALUE ((uint8_t) 0x00)
```

UART2 Control register 6 reset value.

Definition at line 1971 of file STM8AF\_STM8S.h.



**5.1.2.1608 \_UART2\_DR**

```
#define _UART2_DR _SFR(uint8_t, UART2_AddressBase+0x01)
```

UART2 data register.

Definition at line 1950 of file STM8AF\_STM8S.h.

**5.1.2.1609 \_UART2\_FE**

```
#define _UART2_FE ((uint8_t) (0x01 << 1))
```

UART2 Framing error [0] (in \_UART2\_SR)

Definition at line 1977 of file STM8AF\_STM8S.h.

**5.1.2.1610 \_UART2\_GTR**

```
#define _UART2_GTR _SFR(uint8_t, UART2_AddressBase+0x0A)
```

UART2 guard time register.

Definition at line 1959 of file STM8AF\_STM8S.h.

**5.1.2.1611 \_UART2\_GTR\_RESET\_VALUE**

```
#define _UART2_GTR_RESET_VALUE ((uint8_t) 0x00)
```

UART2 guard time register reset value.

Definition at line 1972 of file STM8AF\_STM8S.h.

**5.1.2.1612 \_UART2\_IDLE**

```
#define _UART2_IDLE ((uint8_t) (0x01 << 4))
```

UART2 IDLE line detected [0] (in \_UART2\_SR)

Definition at line 1980 of file STM8AF\_STM8S.h.

**5.1.2.1613 \_UART2\_ILIEN**

```
#define _UART2_ILIEN ((uint8_t) (0x01 << 4))
```

UART2 IDLE Line interrupt enable [0] (in \_UART2\_CR2)

Definition at line 2000 of file STM8AF\_STM8S.h.

**5.1.2.1614 \_UART2\_IREN**

```
#define _UART2_IREN ((uint8_t) (0x01 << 1))
```

UART2 IrDA mode Enable [0] (in \_UART2\_CR5)

Definition at line 2029 of file STM8AF\_STM8S.h.

**5.1.2.1615 \_UART2\_IRLP**

```
#define _UART2_IRLP ((uint8_t) (0x01 << 2))
```

UART2 IrDA Low Power [0] (in \_UART2\_CR5)

Definition at line 2030 of file STM8AF\_STM8S.h.

**5.1.2.1616 \_UART2\_LASE**

```
#define _UART2_LASE ((uint8_t) (0x01 << 4))
```

UART2 LIN automatic resynchronisation enable [0] (in \_UART2\_CR6)

Definition at line 2041 of file STM8AF\_STM8S.h.

**5.1.2.1617 \_UART2\_LBCL**

```
#define _UART2_LBCL ((uint8_t) (0x01 << 0))
```

UART2 Last bit clock pulse [0] (in \_UART2\_CR3)

Definition at line 2006 of file STM8AF\_STM8S.h.

**5.1.2.1618 \_UART2\_LBDF**

```
#define _UART2_LBDF ((uint8_t) (0x01 << 4))
```

UART2 LIN Break Detection Flag [0] (in \_UART2\_CR4)

Definition at line 2022 of file STM8AF\_STM8S.h.

**5.1.2.1619 \_UART2\_LBDIEN**

```
#define _UART2_LBDIEN ((uint8_t) (0x01 << 6))
```

UART2 LIN Break Detection Interrupt Enable [0] (in \_UART2\_CR4)

Definition at line 2024 of file STM8AF\_STM8S.h.

**5.1.2.1620 \_UART2\_LBDL**

```
#define _UART2_LBDL ((uint8_t) (0x01 << 5))
```

UART2 LIN Break Detection Length [0] (in \_UART2\_CR4)

Definition at line 2023 of file STM8AF\_STM8S.h.

**5.1.2.1621 \_UART2\_LDUM**

```
#define _UART2_LDUM ((uint8_t) (0x01 << 7))
```

UART2 LIN Divider Update Method [0] (in \_UART2\_CR6)

Definition at line 2044 of file STM8AF\_STM8S.h.

**5.1.2.1622 \_UART2\_LHDF**

```
#define _UART2_LHDF ((uint8_t) (0x01 << 1))
```

UART2 LIN Header Detection Flag [0] (in \_UART2\_CR6)

Definition at line 2038 of file STM8AF\_STM8S.h.

**5.1.2.1623 \_UART2\_LHDIEN**

```
#define _UART2_LHDIEN ((uint8_t) (0x01 << 2))
```

UART2 LIN Header Detection Interrupt Enable [0] (in \_UART2\_CR6)

Definition at line 2039 of file STM8AF\_STM8S.h.

**5.1.2.1624 \_UART2\_LINEN**

```
#define _UART2_LINEN ((uint8_t) (0x01 << 6))
```

UART2 LIN mode enable [0] (in \_UART2\_CR3)

Definition at line 2013 of file STM8AF\_STM8S.h.

**5.1.2.1625 \_UART2\_LSF**

```
#define _UART2_LSF ((uint8_t) (0x01 << 0))
```

UART2 LIN Sync Field [0] (in \_UART2\_CR6)

Definition at line 2037 of file STM8AF\_STM8S.h.

**5.1.2.1626 \_UART2\_LSLV**

```
#define _UART2_LSLV ((uint8_t) (0x01 << 5))
```

UART2 LIN Slave Enable [0] (in \_UART2\_CR6)

Definition at line 2042 of file STM8AF\_STM8S.h.

**5.1.2.1627 \_UART2\_M**

```
#define _UART2_M ((uint8_t) (0x01 << 4))
```

UART2 word length [0] (in \_UART2\_CR1)

Definition at line 1990 of file STM8AF\_STM8S.h.

**5.1.2.1628 \_UART2\_NACK**

```
#define _UART2_NACK ((uint8_t) (0x01 << 4))
```

UART2 Smartcard NACK enable [0] (in \_UART2\_CR5)

Definition at line 2032 of file STM8AF\_STM8S.h.

**5.1.2.1629 \_UART2\_NF**

```
#define _UART2_NF ((uint8_t) (0x01 << 2))
```

UART2 Noise flag [0] (in \_UART2\_SR)

Definition at line 1978 of file STM8AF\_STM8S.h.

**5.1.2.1630 \_UART2\_OR\_LHE**

```
#define _UART2_OR_LHE ((uint8_t) (0x01 << 3))
```

UART2 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART2\_SR)

Definition at line 1979 of file STM8AF\_STM8S.h.

**5.1.2.1631 \_UART2\_PCEN**

```
#define _UART2_PCEN ((uint8_t) (0x01 << 2))
```

UART2 Parity control enable [0] (in \_UART2\_CR1)

Definition at line 1988 of file STM8AF\_STM8S.h.

**5.1.2.1632 \_UART2\_PE**

```
#define _UART2_PE ((uint8_t) (0x01 << 0))
```

UART2 Parity error [0] (in \_UART2\_SR)

Definition at line 1976 of file STM8AF\_STM8S.h.

#### 5.1.2.1633 \_UART2\_PIEN

```
#define _UART2_PIEN ((uint8_t) (0x01 << 0))
```

UART2 Parity interrupt enable [0] (in \_UART2\_CR1)

Definition at line 1986 of file STM8AF\_STM8S.h.

#### 5.1.2.1634 \_UART2\_PS

```
#define _UART2_PS ((uint8_t) (0x01 << 1))
```

UART2 Parity selection [0] (in \_UART2\_CR1)

Definition at line 1987 of file STM8AF\_STM8S.h.

#### 5.1.2.1635 \_UART2\_PSCR

```
#define _UART2_PSCR _SFR(uint8_t, UART2_AddressBase+0x0B)
```

UART2 prescaler register.

Definition at line 1960 of file STM8AF\_STM8S.h.

#### 5.1.2.1636 \_UART2\_PSCR\_RESET\_VALUE

```
#define _UART2_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

UART2 prescaler register reset value.

Definition at line 1973 of file STM8AF\_STM8S.h.

#### 5.1.2.1637 \_UART2\_R8

```
#define _UART2_R8 ((uint8_t) (0x01 << 7))
```

UART2 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART2\_CR1)

Definition at line 1993 of file STM8AF\_STM8S.h.

**5.1.2.1638 \_UART2\_REN**

```
#define _UART2_REN ((uint8_t) (0x01 << 2))
```

UART2 Receiver enable [0] (in \_UART2\_CR2)

Definition at line 1998 of file STM8AF\_STM8S.h.

**5.1.2.1639 \_UART2\_RIEN**

```
#define _UART2_RIEN ((uint8_t) (0x01 << 5))
```

UART2 Receiver interrupt enable [0] (in \_UART2\_CR2)

Definition at line 2001 of file STM8AF\_STM8S.h.

**5.1.2.1640 \_UART2\_RWU**

```
#define _UART2_RWU ((uint8_t) (0x01 << 1))
```

UART2 Receiver wakeup [0] (in \_UART2\_CR2)

Definition at line 1997 of file STM8AF\_STM8S.h.

**5.1.2.1641 \_UART2\_RXNE**

```
#define _UART2_RXNE ((uint8_t) (0x01 << 5))
```

UART2 Read data register not empty [0] (in \_UART2\_SR)

Definition at line 1981 of file STM8AF\_STM8S.h.

**5.1.2.1642 \_UART2\_SBK**

```
#define _UART2_SBK ((uint8_t) (0x01 << 0))
```

UART2 Send break [0] (in \_UART2\_CR2)

Definition at line 1996 of file STM8AF\_STM8S.h.

#### 5.1.2.1643 \_UART2\_SCEN

```
#define _UART2_SCEN ((uint8_t) (0x01 << 5))
```

UART2 Smartcard mode enable [0] (in \_UART2\_CR5)

Definition at line 2033 of file STM8AF\_STM8S.h.

#### 5.1.2.1644 \_UART2\_SR

```
#define _UART2_SR _SFR(uint8_t, UART2_AddressBase+0x00)
```

UART2 Status register.

Definition at line 1949 of file STM8AF\_STM8S.h.

#### 5.1.2.1645 \_UART2\_SR\_RESET\_VALUE

```
#define _UART2_SR_RESET_VALUE ((uint8_t) 0xC0)
```

UART2 Status register reset value.

Definition at line 1963 of file STM8AF\_STM8S.h.

#### 5.1.2.1646 \_UART2\_STOP

```
#define _UART2_STOP ((uint8_t) (0x03 << 4))
```

UART2 STOP bits [1:0] (in \_UART2\_CR3)

Definition at line 2010 of file STM8AF\_STM8S.h.

#### 5.1.2.1647 \_UART2\_STOP0

```
#define _UART2_STOP0 ((uint8_t) (0x01 << 4))
```

UART2 STOP bits [0] (in \_UART2\_CR3)

Definition at line 2011 of file STM8AF\_STM8S.h.



**5.1.2.1648 \_UART2\_STOP1**

```
#define _UART2_STOP1 ((uint8_t) (0x01 << 5))
```

UART2 STOP bits [1] (in \_UART2\_CR3)

Definition at line 2012 of file STM8AF\_STM8S.h.

**5.1.2.1649 \_UART2\_T8**

```
#define _UART2_T8 ((uint8_t) (0x01 << 6))
```

UART2 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART2\_CR1)

Definition at line 1992 of file STM8AF\_STM8S.h.

**5.1.2.1650 \_UART2\_TC**

```
#define _UART2_TC ((uint8_t) (0x01 << 6))
```

UART2 Transmission complete [0] (in \_UART2\_SR)

Definition at line 1982 of file STM8AF\_STM8S.h.

**5.1.2.1651 \_UART2\_TCIEN**

```
#define _UART2_TCIEN ((uint8_t) (0x01 << 6))
```

UART2 Transmission complete interrupt enable [0] (in \_UART2\_CR2)

Definition at line 2002 of file STM8AF\_STM8S.h.

**5.1.2.1652 \_UART2\_TEN**

```
#define _UART2_TEN ((uint8_t) (0x01 << 3))
```

UART2 Transmitter enable [0] (in \_UART2\_CR2)

Definition at line 1999 of file STM8AF\_STM8S.h.

#### 5.1.2.1653 \_UART2\_TIEN

```
#define _UART2_TIEN ((uint8_t) (0x01 << 7))
```

UART2 Transmitter interrupt enable [0] (in \_UART2\_CR2)

Definition at line 2003 of file STM8AF\_STM8S.h.

#### 5.1.2.1654 \_UART2\_TXE

```
#define _UART2_TXE ((uint8_t) (0x01 << 7))
```

UART2 Transmit data register empty [0] (in \_UART2\_SR)

Definition at line 1983 of file STM8AF\_STM8S.h.

#### 5.1.2.1655 \_UART2\_UARTD

```
#define _UART2_UARTD ((uint8_t) (0x01 << 5))
```

UART2 Disable (for low power consumption) [0] (in \_UART2\_CR1)

Definition at line 1991 of file STM8AF\_STM8S.h.

#### 5.1.2.1656 \_UART2\_WAKE

```
#define _UART2_WAKE ((uint8_t) (0x01 << 3))
```

UART2 Wakeup method [0] (in \_UART2\_CR1)

Definition at line 1989 of file STM8AF\_STM8S.h.

#### 5.1.2.1657 \_UART3

```
#define _UART3 \_SFR(UART3\_t, UART3\_AddressBase)
```

UART3 struct/bit access.

Definition at line 2154 of file STM8AF\_STM8S.h.

**5.1.2.1658 \_UART3\_ADD**

```
#define _UART3_ADD ((uint8_t) (0x0F << 0))
```

UART3 Address of the UART node [3:0] (in \_UART3\_CR4)

Definition at line 2215 of file STM8AF\_STM8S.h.

**5.1.2.1659 \_UART3\_ADD0**

```
#define _UART3_ADD0 ((uint8_t) (0x01 << 0))
```

UART3 Address of the UART node [0] (in \_UART3\_CR4)

Definition at line 2216 of file STM8AF\_STM8S.h.

**5.1.2.1660 \_UART3\_ADD1**

```
#define _UART3_ADD1 ((uint8_t) (0x01 << 1))
```

UART3 Address of the UART node [1] (in \_UART3\_CR4)

Definition at line 2217 of file STM8AF\_STM8S.h.

**5.1.2.1661 \_UART3\_ADD2**

```
#define _UART3_ADD2 ((uint8_t) (0x01 << 2))
```

UART3 Address of the UART node [2] (in \_UART3\_CR4)

Definition at line 2218 of file STM8AF\_STM8S.h.

**5.1.2.1662 \_UART3\_ADD3**

```
#define _UART3_ADD3 ((uint8_t) (0x01 << 3))
```

UART3 Address of the UART node [3] (in \_UART3\_CR4)

Definition at line 2219 of file STM8AF\_STM8S.h.

#### 5.1.2.1663 \_UART3\_BRR1

```
#define _UART3_BRR1 _SFR(uint8_t, UART3_AddressBase+0x02)
```

UART3 Baud rate register 1.

Definition at line 2157 of file STM8AF\_STM8S.h.

#### 5.1.2.1664 \_UART3\_BRR1\_RESET\_VALUE

```
#define _UART3_BRR1_RESET_VALUE ((uint8_t) 0x00)
```

UART3 Baud rate register 1 reset value.

Definition at line 2168 of file STM8AF\_STM8S.h.

#### 5.1.2.1665 \_UART3\_BRR2

```
#define _UART3_BRR2 _SFR(uint8_t, UART3_AddressBase+0x03)
```

UART3 Baud rate register 2.

Definition at line 2158 of file STM8AF\_STM8S.h.

#### 5.1.2.1666 \_UART3\_BRR2\_RESET\_VALUE

```
#define _UART3_BRR2_RESET_VALUE ((uint8_t) 0x00)
```

UART3 Baud rate register 2 reset value.

Definition at line 2169 of file STM8AF\_STM8S.h.

#### 5.1.2.1667 \_UART3\_CR1

```
#define _UART3_CR1 _SFR(uint8_t, UART3_AddressBase+0x04)
```

UART3 Control register 1.

Definition at line 2159 of file STM8AF\_STM8S.h.

**5.1.2.1668 \_UART3\_CR1\_RESET\_VALUE**

```
#define _UART3_CR1_RESET_VALUE ((uint8_t) 0x00)
```

UART3 Control register 1 reset value.

Definition at line 2170 of file STM8AF\_STM8S.h.

**5.1.2.1669 \_UART3\_CR2**

```
#define _UART3_CR2 _SFR(uint8_t, UART3_AddressBase+0x05)
```

UART3 Control register 2.

Definition at line 2160 of file STM8AF\_STM8S.h.

**5.1.2.1670 \_UART3\_CR2\_RESET\_VALUE**

```
#define _UART3_CR2_RESET_VALUE ((uint8_t) 0x00)
```

UART3 Control register 2 reset value.

Definition at line 2171 of file STM8AF\_STM8S.h.

**5.1.2.1671 \_UART3\_CR3**

```
#define _UART3_CR3 _SFR(uint8_t, UART3_AddressBase+0x06)
```

UART3 Control register 3.

Definition at line 2161 of file STM8AF\_STM8S.h.

**5.1.2.1672 \_UART3\_CR3\_RESET\_VALUE**

```
#define _UART3_CR3_RESET_VALUE ((uint8_t) 0x00)
```

UART3 Control register 3 reset value.

Definition at line 2172 of file STM8AF\_STM8S.h.

**5.1.2.1673 \_UART3\_CR4**

```
#define _UART3_CR4 _SFR(uint8_t, UART3_AddressBase+0x07)
```

UART3 Control register 4.

Definition at line 2162 of file STM8AF\_STM8S.h.

**5.1.2.1674 \_UART3\_CR4\_RESET\_VALUE**

```
#define _UART3_CR4_RESET_VALUE ((uint8_t) 0x00)
```

UART3 Control register 4 reset value.

Definition at line 2173 of file STM8AF\_STM8S.h.

**5.1.2.1675 \_UART3\_CR6**

```
#define _UART3_CR6 _SFR(uint8_t, UART3_AddressBase+0x09)
```

UART3 Control register 6.

Definition at line 2164 of file STM8AF\_STM8S.h.

**5.1.2.1676 \_UART3\_CR6\_RESET\_VALUE**

```
#define _UART3_CR6_RESET_VALUE ((uint8_t) 0x00)
```

UART3 Control register 6 reset value.

Definition at line 2174 of file STM8AF\_STM8S.h.

**5.1.2.1677 \_UART3\_DR**

```
#define _UART3_DR _SFR(uint8_t, UART3_AddressBase+0x01)
```

UART3 data register.

Definition at line 2156 of file STM8AF\_STM8S.h.

**5.1.2.1678 \_UART3\_FE**

```
#define _UART3_FE ((uint8_t) (0x01 << 1))
```

UART3 Framing error [0] (in \_UART3\_SR)

Definition at line 2178 of file STM8AF\_STM8S.h.

**5.1.2.1679 \_UART3\_IDLE**

```
#define _UART3_IDLE ((uint8_t) (0x01 << 4))
```

UART3 IDLE line detected [0] (in \_UART3\_SR)

Definition at line 2181 of file STM8AF\_STM8S.h.

**5.1.2.1680 \_UART3\_ILIEN**

```
#define _UART3_ILIEN ((uint8_t) (0x01 << 4))
```

UART3 IDLE Line interrupt enable [0] (in \_UART3\_CR2)

Definition at line 2201 of file STM8AF\_STM8S.h.

**5.1.2.1681 \_UART3\_LASE**

```
#define _UART3_LASE ((uint8_t) (0x01 << 4))
```

UART3 LIN automatic resynchronisation enable [0] (in \_UART3\_CR6)

Definition at line 2230 of file STM8AF\_STM8S.h.

**5.1.2.1682 \_UART3\_LBDF**

```
#define _UART3_LBDF ((uint8_t) (0x01 << 4))
```

UART3 LIN Break Detection Flag [0] (in \_UART3\_CR4)

Definition at line 2220 of file STM8AF\_STM8S.h.

**5.1.2.1683 \_UART3\_LBDIEN**

```
#define _UART3_LBDIEN ((uint8_t) (0x01 << 6))
```

UART3 LIN Break Detection Interrupt Enable [0] (in \_UART3\_CR4)

Definition at line 2222 of file STM8AF\_STM8S.h.

**5.1.2.1684 \_UART3\_LBDL**

```
#define _UART3_LBDL ((uint8_t) (0x01 << 5))
```

UART3 LIN Break Detection Length [0] (in \_UART3\_CR4)

Definition at line 2221 of file STM8AF\_STM8S.h.

**5.1.2.1685 \_UART3\_LDUM**

```
#define _UART3_LDUM ((uint8_t) (0x01 << 7))
```

UART3 LIN Divider Update Method [0] (in \_UART3\_CR6)

Definition at line 2233 of file STM8AF\_STM8S.h.

**5.1.2.1686 \_UART3\_LHDF**

```
#define _UART3_LHDF ((uint8_t) (0x01 << 1))
```

UART3 LIN Header Detection Flag [0] (in \_UART3\_CR6)

Definition at line 2227 of file STM8AF\_STM8S.h.

**5.1.2.1687 \_UART3\_LHDIEN**

```
#define _UART3_LHDIEN ((uint8_t) (0x01 << 2))
```

UART3 LIN Header Detection Interrupt Enable [0] (in \_UART3\_CR6)

Definition at line 2228 of file STM8AF\_STM8S.h.



**5.1.2.1688 \_UART3\_LINEN**

```
#define _UART3_LINEN ((uint8_t) (0x01 << 6))
```

UART3 LIN mode enable [0] (in \_UART3\_CR3)

Definition at line 2211 of file STM8AF\_STM8S.h.

**5.1.2.1689 \_UART3\_LSF**

```
#define _UART3_LSF ((uint8_t) (0x01 << 0))
```

UART3 LIN Sync Field [0] (in \_UART3\_CR6)

Definition at line 2226 of file STM8AF\_STM8S.h.

**5.1.2.1690 \_UART3\_LSLV**

```
#define _UART3_LSLV ((uint8_t) (0x01 << 5))
```

UART3 LIN Slave Enable [0] (in \_UART3\_CR6)

Definition at line 2231 of file STM8AF\_STM8S.h.

**5.1.2.1691 \_UART3\_M**

```
#define _UART3_M ((uint8_t) (0x01 << 4))
```

UART3 word length [0] (in \_UART3\_CR1)

Definition at line 2191 of file STM8AF\_STM8S.h.

**5.1.2.1692 \_UART3\_NF**

```
#define _UART3_NF ((uint8_t) (0x01 << 2))
```

UART3 Noise flag [0] (in \_UART3\_SR)

Definition at line 2179 of file STM8AF\_STM8S.h.

**5.1.2.1693 \_UART3\_OR\_LHE**

```
#define _UART3_OR_LHE ((uint8_t) (0x01 << 3))
```

UART3 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART3\_SR)

Definition at line 2180 of file STM8AF\_STM8S.h.

**5.1.2.1694 \_UART3\_PCEN**

```
#define _UART3_PCEN ((uint8_t) (0x01 << 2))
```

UART3 Parity control enable [0] (in \_UART3\_CR1)

Definition at line 2189 of file STM8AF\_STM8S.h.

**5.1.2.1695 \_UART3\_PE**

```
#define _UART3_PE ((uint8_t) (0x01 << 0))
```

UART3 Parity error [0] (in \_UART3\_SR)

Definition at line 2177 of file STM8AF\_STM8S.h.

**5.1.2.1696 \_UART3\_PIEN**

```
#define _UART3_PIEN ((uint8_t) (0x01 << 0))
```

UART3 Parity interrupt enable [0] (in \_UART3\_CR1)

Definition at line 2187 of file STM8AF\_STM8S.h.

**5.1.2.1697 \_UART3\_PS**

```
#define _UART3_PS ((uint8_t) (0x01 << 1))
```

UART3 Parity selection [0] (in \_UART3\_CR1)

Definition at line 2188 of file STM8AF\_STM8S.h.

**5.1.2.1698 \_UART3\_R8**

```
#define _UART3_R8 ((uint8_t) (0x01 << 7))
```

UART3 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART3\_CR1)

Definition at line 2194 of file STM8AF\_STM8S.h.

**5.1.2.1699 \_UART3\_REN**

```
#define _UART3_REN ((uint8_t) (0x01 << 2))
```

UART3 Receiver enable [0] (in \_UART3\_CR2)

Definition at line 2199 of file STM8AF\_STM8S.h.

**5.1.2.1700 \_UART3\_RIEN**

```
#define _UART3_RIEN ((uint8_t) (0x01 << 5))
```

UART3 Receiver interrupt enable [0] (in \_UART3\_CR2)

Definition at line 2202 of file STM8AF\_STM8S.h.

**5.1.2.1701 \_UART3\_RWU**

```
#define _UART3_RWU ((uint8_t) (0x01 << 1))
```

UART3 Receiver wakeup [0] (in \_UART3\_CR2)

Definition at line 2198 of file STM8AF\_STM8S.h.

**5.1.2.1702 \_UART3\_RXNE**

```
#define _UART3_RXNE ((uint8_t) (0x01 << 5))
```

UART3 Read data register not empty [0] (in \_UART3\_SR)

Definition at line 2182 of file STM8AF\_STM8S.h.

#### 5.1.2.1703 \_UART3\_SBK

```
#define _UART3_SBK ((uint8_t) (0x01 << 0))
```

UART3 Send break [0] (in \_UART3\_CR2)

Definition at line 2197 of file STM8AF\_STM8S.h.

#### 5.1.2.1704 \_UART3\_SR

```
#define _UART3_SR _SFR(uint8_t, UART3_AddressBase+0x00)
```

UART3 Status register.

Definition at line 2155 of file STM8AF\_STM8S.h.

#### 5.1.2.1705 \_UART3\_SR\_RESET\_VALUE

```
#define _UART3_SR_RESET_VALUE ((uint8_t) 0xC0)
```

UART3 Status register reset value.

Definition at line 2167 of file STM8AF\_STM8S.h.

#### 5.1.2.1706 \_UART3\_STOP

```
#define _UART3_STOP ((uint8_t) (0x03 << 4))
```

UART3 STOP bits [1:0] (in \_UART3\_CR3)

Definition at line 2208 of file STM8AF\_STM8S.h.

#### 5.1.2.1707 \_UART3\_STOP0

```
#define _UART3_STOP0 ((uint8_t) (0x01 << 4))
```

UART3 STOP bits [0] (in \_UART3\_CR3)

Definition at line 2209 of file STM8AF\_STM8S.h.

**5.1.2.1708 \_UART3\_STOP1**

```
#define _UART3_STOP1 ((uint8_t) (0x01 << 5))
```

UART3 STOP bits [1] (in \_UART3\_CR3)

Definition at line 2210 of file STM8AF\_STM8S.h.

**5.1.2.1709 \_UART3\_T8**

```
#define _UART3_T8 ((uint8_t) (0x01 << 6))
```

UART3 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART3\_CR1)

Definition at line 2193 of file STM8AF\_STM8S.h.

**5.1.2.1710 \_UART3\_TC**

```
#define _UART3_TC ((uint8_t) (0x01 << 6))
```

UART3 Transmission complete [0] (in \_UART3\_SR)

Definition at line 2183 of file STM8AF\_STM8S.h.

**5.1.2.1711 \_UART3\_TCIEN**

```
#define _UART3_TCIEN ((uint8_t) (0x01 << 6))
```

UART3 Transmission complete interrupt enable [0] (in \_UART3\_CR2)

Definition at line 2203 of file STM8AF\_STM8S.h.

**5.1.2.1712 \_UART3\_TEN**

```
#define _UART3_TEN ((uint8_t) (0x01 << 3))
```

UART3 Transmitter enable [0] (in \_UART3\_CR2)

Definition at line 2200 of file STM8AF\_STM8S.h.

#### 5.1.2.1713 \_UART3\_TIEN

```
#define _UART3_TIEN ((uint8_t) (0x01 << 7))
```

UART3 Transmitter interrupt enable [0] (in \_UART3\_CR2)

Definition at line 2204 of file STM8AF\_STM8S.h.

#### 5.1.2.1714 \_UART3\_TXE

```
#define _UART3_TXE ((uint8_t) (0x01 << 7))
```

UART3 Transmit data register empty [0] (in \_UART3\_SR)

Definition at line 2184 of file STM8AF\_STM8S.h.

#### 5.1.2.1715 \_UART3\_UARTD

```
#define _UART3_UARTD ((uint8_t) (0x01 << 5))
```

UART3 Disable (for low power consumption) [0] (in \_UART3\_CR1)

Definition at line 2192 of file STM8AF\_STM8S.h.

#### 5.1.2.1716 \_UART3\_WAKE

```
#define _UART3_WAKE ((uint8_t) (0x01 << 3))
```

UART3 Wakeup method [0] (in \_UART3\_CR1)

Definition at line 2190 of file STM8AF\_STM8S.h.

#### 5.1.2.1717 \_UART4

```
#define _UART4 _SFR(UART4_t, UART4_AddressBase)
```

UART4 struct/bit access.

Definition at line 2366 of file STM8AF\_STM8S.h.

**5.1.2.1718 \_UART4\_ADD**

```
#define _UART4_ADD ((uint8_t) (0x0F << 0))
```

UART4 Address of the UART node [3:0] (in \_UART4\_CR4)

Definition at line 2435 of file STM8AF\_STM8S.h.

**5.1.2.1719 \_UART4\_ADD0**

```
#define _UART4_ADD0 ((uint8_t) (0x01 << 0))
```

UART4 Address of the UART node [0] (in \_UART4\_CR4)

Definition at line 2436 of file STM8AF\_STM8S.h.

**5.1.2.1720 \_UART4\_ADD1**

```
#define _UART4_ADD1 ((uint8_t) (0x01 << 1))
```

UART4 Address of the UART node [1] (in \_UART4\_CR4)

Definition at line 2437 of file STM8AF\_STM8S.h.

**5.1.2.1721 \_UART4\_ADD2**

```
#define _UART4_ADD2 ((uint8_t) (0x01 << 2))
```

UART4 Address of the UART node [2] (in \_UART4\_CR4)

Definition at line 2438 of file STM8AF\_STM8S.h.

**5.1.2.1722 \_UART4\_ADD3**

```
#define _UART4_ADD3 ((uint8_t) (0x01 << 3))
```

UART4 Address of the UART node [3] (in \_UART4\_CR4)

Definition at line 2439 of file STM8AF\_STM8S.h.

**5.1.2.1723 \_UART4\_BRR1**

```
#define _UART4_BRR1 _SFR(uint8_t, UART4_AddressBase+0x02)
```

UART4 Baud rate register 1.

Definition at line 2369 of file STM8AF\_STM8S.h.

**5.1.2.1724 \_UART4\_BRR1\_RESET\_VALUE**

```
#define _UART4_BRR1_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Baud rate register 1 reset value.

Definition at line 2382 of file STM8AF\_STM8S.h.

**5.1.2.1725 \_UART4\_BRR2**

```
#define _UART4_BRR2 _SFR(uint8_t, UART4_AddressBase+0x03)
```

UART4 Baud rate register 2.

Definition at line 2370 of file STM8AF\_STM8S.h.

**5.1.2.1726 \_UART4\_BRR2\_RESET\_VALUE**

```
#define _UART4_BRR2_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Baud rate register 2 reset value.

Definition at line 2383 of file STM8AF\_STM8S.h.

**5.1.2.1727 \_UART4\_CKEN**

```
#define _UART4_CKEN ((uint8_t) (0x01 << 3))
```

UART4 Clock enable [0] (in \_UART4\_CR3)

Definition at line 2427 of file STM8AF\_STM8S.h.



**5.1.2.1728 \_UART4\_CPHA**

```
#define _UART4_CPHA ((uint8_t) (0x01 << 1))
```

UART4 Clock phase [0] (in \_UART4\_CR3)

Definition at line 2425 of file STM8AF\_STM8S.h.

**5.1.2.1729 \_UART4\_CPOL**

```
#define _UART4_CPOL ((uint8_t) (0x01 << 2))
```

UART4 Clock polarity [0] (in \_UART4\_CR3)

Definition at line 2426 of file STM8AF\_STM8S.h.

**5.1.2.1730 \_UART4\_CR1**

```
#define _UART4_CR1 \_SFR(uint8_t, UART4\_AddressBase+0x04)
```

UART4 Control register 1.

Definition at line 2371 of file STM8AF\_STM8S.h.

**5.1.2.1731 \_UART4\_CR1\_RESET\_VALUE**

```
#define _UART4_CR1_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Control register 1 reset value.

Definition at line 2384 of file STM8AF\_STM8S.h.

**5.1.2.1732 \_UART4\_CR2**

```
#define _UART4_CR2 \_SFR(uint8_t, UART4\_AddressBase+0x05)
```

UART4 Control register 2.

Definition at line 2372 of file STM8AF\_STM8S.h.

**5.1.2.1733 \_UART4\_CR2\_RESET\_VALUE**

```
#define _UART4_CR2_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Control register 2 reset value.

Definition at line 2385 of file STM8AF\_STM8S.h.

**5.1.2.1734 \_UART4\_CR3**

```
#define _UART4_CR3 _SFR(uint8_t, UART4_AddressBase+0x06)
```

UART4 Control register 3.

Definition at line 2373 of file STM8AF\_STM8S.h.

**5.1.2.1735 \_UART4\_CR3\_RESET\_VALUE**

```
#define _UART4_CR3_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Control register 3 reset value.

Definition at line 2386 of file STM8AF\_STM8S.h.

**5.1.2.1736 \_UART4\_CR4**

```
#define _UART4_CR4 _SFR(uint8_t, UART4_AddressBase+0x07)
```

UART4 Control register 4.

Definition at line 2374 of file STM8AF\_STM8S.h.

**5.1.2.1737 \_UART4\_CR4\_RESET\_VALUE**

```
#define _UART4_CR4_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Control register 4 reset value.

Definition at line 2387 of file STM8AF\_STM8S.h.

**5.1.2.1738 \_UART4\_CR5**

```
#define _UART4_CR5 _SFR(uint8_t, UART4_AddressBase+0x08)
```

UART4 Control register 5.

Definition at line 2375 of file STM8AF\_STM8S.h.

**5.1.2.1739 \_UART4\_CR5\_RESET\_VALUE**

```
#define _UART4_CR5_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Control register 5 reset value.

Definition at line 2388 of file STM8AF\_STM8S.h.

**5.1.2.1740 \_UART4\_CR6**

```
#define _UART4_CR6 _SFR(uint8_t, UART4_AddressBase+0x09)
```

UART4 Control register 6.

Definition at line 2376 of file STM8AF\_STM8S.h.

**5.1.2.1741 \_UART4\_CR6\_RESET\_VALUE**

```
#define _UART4_CR6_RESET_VALUE ((uint8_t) 0x00)
```

UART4 Control register 6 reset value.

Definition at line 2389 of file STM8AF\_STM8S.h.

**5.1.2.1742 \_UART4\_DR**

```
#define _UART4_DR _SFR(uint8_t, UART4_AddressBase+0x01)
```

UART4 data register.

Definition at line 2368 of file STM8AF\_STM8S.h.

#### 5.1.2.1743 \_UART4\_FE

```
#define _UART4_FE ((uint8_t) (0x01 << 1))
```

UART4 Framing error [0] (in \_UART4\_SR)

Definition at line 2395 of file STM8AF\_STM8S.h.

#### 5.1.2.1744 \_UART4\_GTR

```
#define _UART4_GTR _SFR(uint8_t, UART4_AddressBase+0x0A)
```

UART4 guard time register.

Definition at line 2377 of file STM8AF\_STM8S.h.

#### 5.1.2.1745 \_UART4\_GTR\_RESET\_VALUE

```
#define _UART4_GTR_RESET_VALUE ((uint8_t) 0x00)
```

UART4 guard time register reset value.

Definition at line 2390 of file STM8AF\_STM8S.h.

#### 5.1.2.1746 \_UART4\_HDSEL

```
#define _UART4_HDSEL ((uint8_t) (0x01 << 3))
```

UART4 Half-Duplex Selection [0] (in \_UART4\_CR5)

Definition at line 2449 of file STM8AF\_STM8S.h.

#### 5.1.2.1747 \_UART4\_IDLE

```
#define _UART4_IDLE ((uint8_t) (0x01 << 4))
```

UART4 IDLE line detected [0] (in \_UART4\_SR)

Definition at line 2398 of file STM8AF\_STM8S.h.

**5.1.2.1748 \_UART4\_ILIEN**

```
#define _UART4_ILIEN ((uint8_t) (0x01 << 4))
```

UART4 IDLE Line interrupt enable [0] (in \_UART4\_CR2)

Definition at line 2418 of file STM8AF\_STM8S.h.

**5.1.2.1749 \_UART4\_IREN**

```
#define _UART4_IREN ((uint8_t) (0x01 << 1))
```

UART4 IrDA mode Enable [0] (in \_UART4\_CR5)

Definition at line 2447 of file STM8AF\_STM8S.h.

**5.1.2.1750 \_UART4\_IRLP**

```
#define _UART4_IRLP ((uint8_t) (0x01 << 2))
```

UART4 IrDA Low Power [0] (in \_UART4\_CR5)

Definition at line 2448 of file STM8AF\_STM8S.h.

**5.1.2.1751 \_UART4\_LASE**

```
#define _UART4_LASE ((uint8_t) (0x01 << 4))
```

UART4 LIN automatic resynchronisation enable [0] (in \_UART4\_CR6)

Definition at line 2459 of file STM8AF\_STM8S.h.

**5.1.2.1752 \_UART4\_LBCL**

```
#define _UART4_LBCL ((uint8_t) (0x01 << 0))
```

UART4 Last bit clock pulse [0] (in \_UART4\_CR3)

Definition at line 2424 of file STM8AF\_STM8S.h.

**5.1.2.1753 \_UART4\_LBDF**

```
#define _UART4_LBDF ((uint8_t) (0x01 << 4))
```

UART4 LIN Break Detection Flag [0] (in \_UART4\_CR4)

Definition at line 2440 of file STM8AF\_STM8S.h.

**5.1.2.1754 \_UART4\_LBDIEN**

```
#define _UART4_LBDIEN ((uint8_t) (0x01 << 6))
```

UART4 LIN Break Detection Interrupt Enable [0] (in \_UART4\_CR4)

Definition at line 2442 of file STM8AF\_STM8S.h.

**5.1.2.1755 \_UART4\_LBDL**

```
#define _UART4_LBDL ((uint8_t) (0x01 << 5))
```

UART4 LIN Break Detection Length [0] (in \_UART4\_CR4)

Definition at line 2441 of file STM8AF\_STM8S.h.

**5.1.2.1756 \_UART4\_LDUM**

```
#define _UART4_LDUM ((uint8_t) (0x01 << 7))
```

UART4 LIN Divider Update Method [0] (in \_UART4\_CR6)

Definition at line 2462 of file STM8AF\_STM8S.h.

**5.1.2.1757 \_UART4\_LHDF**

```
#define _UART4_LHDF ((uint8_t) (0x01 << 1))
```

UART4 LIN Header Detection Flag [0] (in \_UART4\_CR6)

Definition at line 2456 of file STM8AF\_STM8S.h.

**5.1.2.1758 \_UART4\_LHDIEN**

```
#define _UART4_LHDIEN ((uint8_t) (0x01 << 2))
```

UART4 LIN Header Detection Interrupt Enable [0] (in \_UART4\_CR6)

Definition at line 2457 of file STM8AF\_STM8S.h.

**5.1.2.1759 \_UART4\_LINEN**

```
#define _UART4_LINEN ((uint8_t) (0x01 << 6))
```

UART4 LIN mode enable [0] (in \_UART4\_CR3)

Definition at line 2431 of file STM8AF\_STM8S.h.

**5.1.2.1760 \_UART4\_LSF**

```
#define _UART4_LSF ((uint8_t) (0x01 << 0))
```

UART4 LIN Sync Field [0] (in \_UART4\_CR6)

Definition at line 2455 of file STM8AF\_STM8S.h.

**5.1.2.1761 \_UART4\_LSLV**

```
#define _UART4_LSLV ((uint8_t) (0x01 << 5))
```

UART4 LIN Slave Enable [0] (in \_UART4\_CR6)

Definition at line 2460 of file STM8AF\_STM8S.h.

**5.1.2.1762 \_UART4\_M**

```
#define _UART4_M ((uint8_t) (0x01 << 4))
```

UART4 word length [0] (in \_UART4\_CR1)

Definition at line 2408 of file STM8AF\_STM8S.h.

**5.1.2.1763 \_UART4\_NACK**

```
#define _UART4_NACK ((uint8_t) (0x01 << 4))
```

UART4 Smartcard NACK enable [0] (in \_UART4\_CR5)

Definition at line 2450 of file STM8AF\_STM8S.h.

**5.1.2.1764 \_UART4\_NF**

```
#define _UART4_NF ((uint8_t) (0x01 << 2))
```

UART4 Noise flag [0] (in \_UART4\_SR)

Definition at line 2396 of file STM8AF\_STM8S.h.

**5.1.2.1765 \_UART4\_OR\_LHE**

```
#define _UART4_OR_LHE ((uint8_t) (0x01 << 3))
```

UART4 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART4\_SR)

Definition at line 2397 of file STM8AF\_STM8S.h.

**5.1.2.1766 \_UART4\_PCEN**

```
#define _UART4_PCEN ((uint8_t) (0x01 << 2))
```

UART4 Parity control enable [0] (in \_UART4\_CR1)

Definition at line 2406 of file STM8AF\_STM8S.h.

**5.1.2.1767 \_UART4\_PE**

```
#define _UART4_PE ((uint8_t) (0x01 << 0))
```

UART4 Parity error [0] (in \_UART4\_SR)

Definition at line 2394 of file STM8AF\_STM8S.h.



**5.1.2.1768 \_UART4\_PIEN**

```
#define _UART4_PIEN ((uint8_t) (0x01 << 0))
```

UART4 Parity interrupt enable [0] (in \_UART4\_CR1)

Definition at line 2404 of file STM8AF\_STM8S.h.

**5.1.2.1769 \_UART4\_PS**

```
#define _UART4_PS ((uint8_t) (0x01 << 1))
```

UART4 Parity selection [0] (in \_UART4\_CR1)

Definition at line 2405 of file STM8AF\_STM8S.h.

**5.1.2.1770 \_UART4\_PSCR**

```
#define _UART4_PSCR _SFR(uint8_t, UART4_AddressBase+0x0B)
```

UART4 prescaler register.

Definition at line 2378 of file STM8AF\_STM8S.h.

**5.1.2.1771 \_UART4\_PSCR\_RESET\_VALUE**

```
#define _UART4_PSCR_RESET_VALUE ((uint8_t) 0x00)
```

UART4 prescaler register reset value.

Definition at line 2391 of file STM8AF\_STM8S.h.

**5.1.2.1772 \_UART4\_R8**

```
#define _UART4_R8 ((uint8_t) (0x01 << 7))
```

UART4 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART4\_CR1)

Definition at line 2411 of file STM8AF\_STM8S.h.

**5.1.2.1773 \_UART4\_REN**

```
#define _UART4_REN ((uint8_t) (0x01 << 2))
```

UART4 Receiver enable [0] (in \_UART4\_CR2)

Definition at line 2416 of file STM8AF\_STM8S.h.

**5.1.2.1774 \_UART4\_RIEN**

```
#define _UART4_RIEN ((uint8_t) (0x01 << 5))
```

UART4 Receiver interrupt enable [0] (in \_UART4\_CR2)

Definition at line 2419 of file STM8AF\_STM8S.h.

**5.1.2.1775 \_UART4\_RWU**

```
#define _UART4_RWU ((uint8_t) (0x01 << 1))
```

UART4 Receiver wakeup [0] (in \_UART4\_CR2)

Definition at line 2415 of file STM8AF\_STM8S.h.

**5.1.2.1776 \_UART4\_RXNE**

```
#define _UART4_RXNE ((uint8_t) (0x01 << 5))
```

UART4 Read data register not empty [0] (in \_UART4\_SR)

Definition at line 2399 of file STM8AF\_STM8S.h.

**5.1.2.1777 \_UART4\_SBK**

```
#define _UART4_SBK ((uint8_t) (0x01 << 0))
```

UART4 Send break [0] (in \_UART4\_CR2)

Definition at line 2414 of file STM8AF\_STM8S.h.

**5.1.2.1778 \_UART4\_SCEN**

```
#define _UART4_SCEN ((uint8_t) (0x01 << 5))
```

UART4 Smartcard mode enable [0] (in \_UART4\_CR5)

Definition at line 2451 of file STM8AF\_STM8S.h.

**5.1.2.1779 \_UART4\_SR**

```
#define _UART4_SR _SFR(uint8_t, UART4_AddressBase+0x00)
```

UART4 Status register.

Definition at line 2367 of file STM8AF\_STM8S.h.

**5.1.2.1780 \_UART4\_SR\_RESET\_VALUE**

```
#define _UART4_SR_RESET_VALUE ((uint8_t) 0xC0)
```

UART4 Status register reset value.

Definition at line 2381 of file STM8AF\_STM8S.h.

**5.1.2.1781 \_UART4\_STOP**

```
#define _UART4_STOP ((uint8_t) (0x03 << 4))
```

UART4 STOP bits [1:0] (in \_UART4\_CR3)

Definition at line 2428 of file STM8AF\_STM8S.h.

**5.1.2.1782 \_UART4\_STOP0**

```
#define _UART4_STOP0 ((uint8_t) (0x01 << 4))
```

UART4 STOP bits [0] (in \_UART4\_CR3)

Definition at line 2429 of file STM8AF\_STM8S.h.

**5.1.2.1783 \_UART4\_STOP1**

```
#define _UART4_STOP1 ((uint8_t) (0x01 << 5))
```

UART4 STOP bits [1] (in \_UART4\_CR3)

Definition at line 2430 of file STM8AF\_STM8S.h.

**5.1.2.1784 \_UART4\_T8**

```
#define _UART4_T8 ((uint8_t) (0x01 << 6))
```

UART4 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART4\_CR1)

Definition at line 2410 of file STM8AF\_STM8S.h.

**5.1.2.1785 \_UART4\_TC**

```
#define _UART4_TC ((uint8_t) (0x01 << 6))
```

UART4 Transmission complete [0] (in \_UART4\_SR)

Definition at line 2400 of file STM8AF\_STM8S.h.

**5.1.2.1786 \_UART4\_TCIEN**

```
#define _UART4_TCIEN ((uint8_t) (0x01 << 6))
```

UART4 Transmission complete interrupt enable [0] (in \_UART4\_CR2)

Definition at line 2420 of file STM8AF\_STM8S.h.

**5.1.2.1787 \_UART4\_TEN**

```
#define _UART4_TEN ((uint8_t) (0x01 << 3))
```

UART4 Transmitter enable [0] (in \_UART4\_CR2)

Definition at line 2417 of file STM8AF\_STM8S.h.

**5.1.2.1788 \_UART4\_TIEN**

```
#define _UART4_TIEN ((uint8_t) (0x01 << 7))
```

UART4 Transmitter interrupt enable [0] (in \_UART4\_CR2)

Definition at line 2421 of file STM8AF\_STM8S.h.

**5.1.2.1789 \_UART4\_TXE**

```
#define _UART4_TXE ((uint8_t) (0x01 << 7))
```

UART4 Transmit data register empty [0] (in \_UART4\_SR)

Definition at line 2401 of file STM8AF\_STM8S.h.

**5.1.2.1790 \_UART4\_UARTD**

```
#define _UART4_UARTD ((uint8_t) (0x01 << 5))
```

UART4 Disable (for low power consumption) [0] (in \_UART4\_CR1)

Definition at line 2409 of file STM8AF\_STM8S.h.

**5.1.2.1791 \_UART4\_WAKE**

```
#define _UART4_WAKE ((uint8_t) (0x01 << 3))
```

UART4 Wakeup method [0] (in \_UART4\_CR1)

Definition at line 2407 of file STM8AF\_STM8S.h.

**5.1.2.1792 \_WWDG**

```
#define _WWDG _SFR(WWDG_t, WWDG_AddressBase)
```

Window Watchdog struct/bit access.

Definition at line 1006 of file STM8AF\_STM8S.h.

**5.1.2.1793 \_WWDG\_CR**

```
#define _WWDG_CR _SFR(uint8_t, WWDG_AddressBase+0x00)
```

Window Watchdog Control register (WWDG\_CR)

Definition at line 1007 of file STM8AF\_STM8S.h.

**5.1.2.1794 \_WWDG\_CR\_RESET\_VALUE**

```
#define _WWDG_CR_RESET_VALUE ((uint8_t) 0x7F)
```

Window Watchdog Control register reset value.

Definition at line 1011 of file STM8AF\_STM8S.h.

**5.1.2.1795 \_WWDG\_T**

```
#define _WWDG_T ((uint8_t) (0x7F << 0))
```

Window Watchdog 7-bit counter [6:0] (in \_WWDG\_CR)

Definition at line 1015 of file STM8AF\_STM8S.h.

**5.1.2.1796 \_WWDG\_T0**

```
#define _WWDG_T0 ((uint8_t) (0x01 << 0))
```

Window Watchdog 7-bit counter [0] (in \_WWDG\_CR)

Definition at line 1016 of file STM8AF\_STM8S.h.

**5.1.2.1797 \_WWDG\_T1**

```
#define _WWDG_T1 ((uint8_t) (0x01 << 1))
```

Window Watchdog 7-bit counter [1] (in \_WWDG\_CR)

Definition at line 1017 of file STM8AF\_STM8S.h.

**5.1.2.1798 \_WWDG\_T2**

```
#define _WWDG_T2 ((uint8_t) (0x01 << 2))
```

Window Watchdog 7-bit counter [2] (in \_WWDG\_CR)

Definition at line 1018 of file STM8AF\_STM8S.h.

**5.1.2.1799 \_WWDG\_T3**

```
#define _WWDG_T3 ((uint8_t) (0x01 << 3))
```

Window Watchdog 7-bit counter [3] (in \_WWDG\_CR)

Definition at line 1019 of file STM8AF\_STM8S.h.

**5.1.2.1800 \_WWDG\_T4**

```
#define _WWDG_T4 ((uint8_t) (0x01 << 4))
```

Window Watchdog 7-bit counter [4] (in \_WWDG\_CR)

Definition at line 1020 of file STM8AF\_STM8S.h.

**5.1.2.1801 \_WWDG\_T5**

```
#define _WWDG_T5 ((uint8_t) (0x01 << 5))
```

Window Watchdog 7-bit counter [5] (in \_WWDG\_CR)

Definition at line 1021 of file STM8AF\_STM8S.h.

**5.1.2.1802 \_WWDG\_T6**

```
#define _WWDG_T6 ((uint8_t) (0x01 << 6))
```

Window Watchdog 7-bit counter [6] (in \_WWDG\_CR)

Definition at line 1022 of file STM8AF\_STM8S.h.

**5.1.2.1803 \_WWDG\_W**

```
#define _WWDG_W ((uint8_t) (0x7F << 0))
```

Window Watchdog 7-bit window value [6:0] (in \_WWDG\_WR)

Definition at line 1026 of file STM8AF\_STM8S.h.

**5.1.2.1804 \_WWDG\_W0**

```
#define _WWDG_W0 ((uint8_t) (0x01 << 0))
```

Window Watchdog 7-bit window value [0] (in \_WWDG\_WR)

Definition at line 1027 of file STM8AF\_STM8S.h.

**5.1.2.1805 \_WWDG\_W1**

```
#define _WWDG_W1 ((uint8_t) (0x01 << 1))
```

Window Watchdog 7-bit window value [1] (in \_WWDG\_WR)

Definition at line 1028 of file STM8AF\_STM8S.h.

**5.1.2.1806 \_WWDG\_W2**

```
#define _WWDG_W2 ((uint8_t) (0x01 << 2))
```

Window Watchdog 7-bit window value [2] (in \_WWDG\_WR)

Definition at line 1029 of file STM8AF\_STM8S.h.

**5.1.2.1807 \_WWDG\_W3**

```
#define _WWDG_W3 ((uint8_t) (0x01 << 3))
```

Window Watchdog 7-bit window value [3] (in \_WWDG\_WR)

Definition at line 1030 of file STM8AF\_STM8S.h.



**5.1.2.1808 \_WWDG\_W4**

```
#define _WWDG_W4 ((uint8_t) (0x01 << 4))
```

Window Watchdog 7-bit window value [4] (in \_WWDG\_WR)

Definition at line 1031 of file STM8AF\_STM8S.h.

**5.1.2.1809 \_WWDG\_W5**

```
#define _WWDG_W5 ((uint8_t) (0x01 << 5))
```

Window Watchdog 7-bit window value [5] (in \_WWDG\_WR)

Definition at line 1032 of file STM8AF\_STM8S.h.

**5.1.2.1810 \_WWDG\_W6**

```
#define _WWDG_W6 ((uint8_t) (0x01 << 6))
```

Window Watchdog 7-bit window value [6] (in \_WWDG\_WR)

Definition at line 1033 of file STM8AF\_STM8S.h.

**5.1.2.1811 \_WWDG\_WDGA**

```
#define _WWDG_WDGA ((uint8_t) (0x01 << 7))
```

Window Watchdog activation bit (n/a if WWDG enabled by option byte) [0] (in \_WWDG\_CR)

Definition at line 1023 of file STM8AF\_STM8S.h.

**5.1.2.1812 \_WWDG\_WR**

```
#define _WWDG_WR _SFR(uint8_t, WWDG_AddressBase+0x01)
```

Window Watchdog Window register (WWDG\_WR)

Definition at line 1008 of file STM8AF\_STM8S.h.

**5.1.2.1813   \_WWDG\_WR\_RESET\_VALUE**

```
#define _WWDG_WR_RESET_VALUE ((uint8_t) 0x7F)
```

Window Watchdog Window register reset value.

Definition at line 1012 of file STM8AF\_STM8S.h.

**5.1.2.1814   ADC1\_AddressBase** [1/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8AF6213.h.

**5.1.2.1815   ADC1\_AddressBase** [2/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S903F3.h.

**5.1.2.1816   ADC1\_AddressBase** [3/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S903K3.h.

**5.1.2.1817   ADC1\_AddressBase** [4/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S001J3.h.

**5.1.2.1818   ADC1\_AddressBase** [5/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S003F3.h.

**5.1.2.1819 ADC1\_AddressBase** [6/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S003K3.h.

**5.1.2.1820 ADC1\_AddressBase** [7/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S103F2.h.

**5.1.2.1821 ADC1\_AddressBase** [8/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8AF6213A.h.

**5.1.2.1822 ADC1\_AddressBase** [9/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8AF6223.h.

**5.1.2.1823 ADC1\_AddressBase** [10/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S103F3.h.

**5.1.2.1824 ADC1\_AddressBase** [11/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8AF6223A.h.

**5.1.2.1825 ADC1\_AddressBase** [12/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8AF6226.h.

**5.1.2.1826 ADC1\_AddressBase** [13/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 73 of file STM8S103K3.h.

**5.1.2.1827 ADC1\_AddressBase** [14/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 74 of file STM8AF6366.h.

**5.1.2.1828 ADC1\_AddressBase** [15/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8AF6246.h.

**5.1.2.1829 ADC1\_AddressBase** [16/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8AF6248.h.

**5.1.2.1830 ADC1\_AddressBase** [17/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8AF6266.h.

**5.1.2.1831** ADC1\_AddressBase [18/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8AF6268.h.

**5.1.2.1832** ADC1\_AddressBase [19/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8AF6269.h.

**5.1.2.1833** ADC1\_AddressBase [20/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S005C6.h.

**5.1.2.1834** ADC1\_AddressBase [21/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S005K6.h.

**5.1.2.1835** ADC1\_AddressBase [22/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S105C4.h.

**5.1.2.1836** ADC1\_AddressBase [23/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S105C6.h.

**5.1.2.1837 ADC1\_AddressBase** [24/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S105K4.h.

**5.1.2.1838 ADC1\_AddressBase** [25/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S105K6.h.

**5.1.2.1839 ADC1\_AddressBase** [26/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S105S4.h.

**5.1.2.1840 ADC1\_AddressBase** [27/27]

```
#define ADC1_AddressBase 0x53E0
```

Definition at line 75 of file STM8S105S6.h.

**5.1.2.1841 ADC2\_AddressBase** [1/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF5269.h.

**5.1.2.1842 ADC2\_AddressBase** [2/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208S8.h.

**5.1.2.1843 ADC2\_AddressBase** [3/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF52A8.h.

**5.1.2.1844 ADC2\_AddressBase** [4/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208SB.h.

**5.1.2.1845 ADC2\_AddressBase** [5/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF5288.h.

**5.1.2.1846 ADC2\_AddressBase** [6/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208R8.h.

**5.1.2.1847 ADC2\_AddressBase** [7/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF52A9.h.

**5.1.2.1848 ADC2\_AddressBase** [8/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF5289.h.

**5.1.2.1849** **ADC2\_AddressBase** [9/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208CB.h.

**5.1.2.1850** **ADC2\_AddressBase** [10/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF6286.h.

**5.1.2.1851** **ADC2\_AddressBase** [11/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF52AA.h.

**5.1.2.1852** **ADC2\_AddressBase** [12/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF6288.h.

**5.1.2.1853** **ADC2\_AddressBase** [13/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S007C8.h.

**5.1.2.1854** **ADC2\_AddressBase** [14/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF6289.h.



**5.1.2.1855** ADC2\_AddressBase [15/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF528A.h.

**5.1.2.1856** ADC2\_AddressBase [16/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF628A.h.

**5.1.2.1857** ADC2\_AddressBase [17/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF62A6.h.

**5.1.2.1858** ADC2\_AddressBase [18/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF5268.h.

**5.1.2.1859** ADC2\_AddressBase [19/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207C6.h.

**5.1.2.1860** ADC2\_AddressBase [20/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207C8.h.

**5.1.2.1861** **ADC2\_AddressBase** [21/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207CB.h.

**5.1.2.1862** **ADC2\_AddressBase** [22/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF5286.h.

**5.1.2.1863** **ADC2\_AddressBase** [23/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF62A8.h.

**5.1.2.1864** **ADC2\_AddressBase** [24/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207K6.h.

**5.1.2.1865** **ADC2\_AddressBase** [25/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207K8.h.

**5.1.2.1866** **ADC2\_AddressBase** [26/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207M8.h.

**5.1.2.1867 ADC2\_AddressBase** [27/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207MB.h.

**5.1.2.1868 ADC2\_AddressBase** [28/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF62A9.h.

**5.1.2.1869 ADC2\_AddressBase** [29/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF52A6.h.

**5.1.2.1870 ADC2\_AddressBase** [30/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207R6.h.

**5.1.2.1871 ADC2\_AddressBase** [31/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207R8.h.

**5.1.2.1872 ADC2\_AddressBase** [32/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207RB.h.

**5.1.2.1873** **ADC2\_AddressBase** [33/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF62AA.h.

**5.1.2.1874** **ADC2\_AddressBase** [34/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207S6.h.

**5.1.2.1875** **ADC2\_AddressBase** [35/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207S8.h.

**5.1.2.1876** **ADC2\_AddressBase** [36/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S207SB.h.

**5.1.2.1877** **ADC2\_AddressBase** [37/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208C6.h.

**5.1.2.1878** **ADC2\_AddressBase** [38/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208C8.h.

**5.1.2.1879** ADC2\_AddressBase [39/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208MB.h.

**5.1.2.1880** ADC2\_AddressBase [40/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8AF6388.h.

**5.1.2.1881** ADC2\_AddressBase [41/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208R6.h.

**5.1.2.1882** ADC2\_AddressBase [42/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208RB.h.

**5.1.2.1883** ADC2\_AddressBase [43/43]

```
#define ADC2_AddressBase 0x5400
```

Definition at line 78 of file STM8S208S6.h.

**5.1.2.1884** AWU\_AddressBase [1/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S103K3.h.

**5.1.2.1885 AWU\_AddressBase** [2/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S903F3.h.

**5.1.2.1886 AWU\_AddressBase** [3/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S903K3.h.

**5.1.2.1887 AWU\_AddressBase** [4/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S001J3.h.

**5.1.2.1888 AWU\_AddressBase** [5/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S003F3.h.

**5.1.2.1889 AWU\_AddressBase** [6/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S003K3.h.

**5.1.2.1890 AWU\_AddressBase** [7/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8AF6213.h.

**5.1.2.1891 AWU\_AddressBase** [8/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S103F2.h.

**5.1.2.1892 AWU\_AddressBase** [9/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8S103F3.h.

**5.1.2.1893 AWU\_AddressBase** [10/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8AF6213A.h.

**5.1.2.1894 AWU\_AddressBase** [11/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8AF6223.h.

**5.1.2.1895 AWU\_AddressBase** [12/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8AF6223A.h.

**5.1.2.1896 AWU\_AddressBase** [13/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8AF6366.h.

**5.1.2.1897 AWU\_AddressBase** [14/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 65 of file STM8AF6226.h.

**5.1.2.1898 AWU\_AddressBase** [15/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8AF6246.h.

**5.1.2.1899 AWU\_AddressBase** [16/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S005K6.h.

**5.1.2.1900 AWU\_AddressBase** [17/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8AF6248.h.

**5.1.2.1901 AWU\_AddressBase** [18/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8AF6266.h.

**5.1.2.1902 AWU\_AddressBase** [19/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8AF6268.h.



**5.1.2.1903 AWU\_AddressBase** [20/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8AF6269.h.

**5.1.2.1904 AWU\_AddressBase** [21/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S005C6.h.

**5.1.2.1905 AWU\_AddressBase** [22/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S105C6.h.

**5.1.2.1906 AWU\_AddressBase** [23/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S105K4.h.

**5.1.2.1907 AWU\_AddressBase** [24/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S105K6.h.

**5.1.2.1908 AWU\_AddressBase** [25/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S105S4.h.

**5.1.2.1909 AWU\_AddressBase** [26/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S105S6.h.

**5.1.2.1910 AWU\_AddressBase** [27/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 66 of file STM8S105C4.h.

**5.1.2.1911 AWU\_AddressBase** [28/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208S8.h.

**5.1.2.1912 AWU\_AddressBase** [29/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208SB.h.

**5.1.2.1913 AWU\_AddressBase** [30/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF52A8.h.

**5.1.2.1914 AWU\_AddressBase** [31/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207R8.h.

**5.1.2.1915 AWU\_AddressBase** [32/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF5288.h.

**5.1.2.1916 AWU\_AddressBase** [33/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF5268.h.

**5.1.2.1917 AWU\_AddressBase** [34/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207S6.h.

**5.1.2.1918 AWU\_AddressBase** [35/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF52A9.h.

**5.1.2.1919 AWU\_AddressBase** [36/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207CB.h.

**5.1.2.1920 AWU\_AddressBase** [37/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF5289.h.

**5.1.2.1921 AWU\_AddressBase** [38/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208S6.h.

**5.1.2.1922 AWU\_AddressBase** [39/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208R6.h.

**5.1.2.1923 AWU\_AddressBase** [40/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF52AA.h.

**5.1.2.1924 AWU\_AddressBase** [41/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF6286.h.

**5.1.2.1925 AWU\_AddressBase** [42/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207MB.h.

**5.1.2.1926 AWU\_AddressBase** [43/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF5286.h.

**5.1.2.1927 AWU\_AddressBase** [44/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207K8.h.

**5.1.2.1928 AWU\_AddressBase** [45/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF6288.h.

**5.1.2.1929 AWU\_AddressBase** [46/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF6289.h.

**5.1.2.1930 AWU\_AddressBase** [47/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S007C8.h.

**5.1.2.1931 AWU\_AddressBase** [48/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF528A.h.

**5.1.2.1932 AWU\_AddressBase** [49/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF628A.h.

**5.1.2.1933 AWU\_AddressBase** [50/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF5269.h.

**5.1.2.1934 AWU\_AddressBase** [51/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF62A6.h.

**5.1.2.1935 AWU\_AddressBase** [52/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207C8.h.

**5.1.2.1936 AWU\_AddressBase** [53/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207K6.h.

**5.1.2.1937 AWU\_AddressBase** [54/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF62A8.h.

**5.1.2.1938 AWU\_AddressBase** [55/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207M8.h.

**5.1.2.1939 AWU\_AddressBase** [56/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207R6.h.

**5.1.2.1940 AWU\_AddressBase** [57/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF62A9.h.

**5.1.2.1941 AWU\_AddressBase** [58/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207RB.h.

**5.1.2.1942 AWU\_AddressBase** [59/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF62AA.h.

**5.1.2.1943 AWU\_AddressBase** [60/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207S8.h.

**5.1.2.1944 AWU\_AddressBase** [61/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207SB.h.

**5.1.2.1945 AWU\_AddressBase** [62/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208C6.h.

**5.1.2.1946 AWU\_AddressBase** [63/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208C8.h.

**5.1.2.1947 AWU\_AddressBase** [64/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208CB.h.

**5.1.2.1948 AWU\_AddressBase** [65/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S207C6.h.

**5.1.2.1949 AWU\_AddressBase** [66/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208MB.h.

**5.1.2.1950 AWU\_AddressBase** [67/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF6388.h.



**5.1.2.1951 AWU\_AddressBase** [68/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208R8.h.

**5.1.2.1952 AWU\_AddressBase** [69/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8AF52A6.h.

**5.1.2.1953 AWU\_AddressBase** [70/70]

```
#define AWU_AddressBase 0x50F0
```

Definition at line 68 of file STM8S208RB.h.

**5.1.2.1954 BEEP\_AddressBase** [1/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S903F3.h.

**5.1.2.1955 BEEP\_AddressBase** [2/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S903K3.h.

**5.1.2.1956 BEEP\_AddressBase** [3/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8AF6223A.h.

**5.1.2.1957 BEEP\_AddressBase** [4/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S001J3.h.

**5.1.2.1958 BEEP\_AddressBase** [5/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S003F3.h.

**5.1.2.1959 BEEP\_AddressBase** [6/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S003K3.h.

**5.1.2.1960 BEEP\_AddressBase** [7/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8AF6213.h.

**5.1.2.1961 BEEP\_AddressBase** [8/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S103F2.h.

**5.1.2.1962 BEEP\_AddressBase** [9/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S103F3.h.

**5.1.2.1963 BEEP\_AddressBase** [10/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8S103K3.h.

**5.1.2.1964 BEEP\_AddressBase** [11/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8AF6213A.h.

**5.1.2.1965 BEEP\_AddressBase** [12/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8AF6223.h.

**5.1.2.1966 BEEP\_AddressBase** [13/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8AF6366.h.

**5.1.2.1967 BEEP\_AddressBase** [14/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 66 of file STM8AF6226.h.

**5.1.2.1968 BEEP\_AddressBase** [15/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8AF6246.h.

**5.1.2.1969 BEEP\_AddressBase** [16/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8AF6248.h.

**5.1.2.1970 BEEP\_AddressBase** [17/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8AF6266.h.

**5.1.2.1971 BEEP\_AddressBase** [18/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8AF6268.h.

**5.1.2.1972 BEEP\_AddressBase** [19/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8AF6269.h.

**5.1.2.1973 BEEP\_AddressBase** [20/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S005C6.h.

**5.1.2.1974 BEEP\_AddressBase** [21/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S005K6.h.

**5.1.2.1975 BEEP\_AddressBase** [22/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S105C4.h.

**5.1.2.1976 BEEP\_AddressBase** [23/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S105C6.h.

**5.1.2.1977 BEEP\_AddressBase** [24/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S105K4.h.

**5.1.2.1978 BEEP\_AddressBase** [25/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S105K6.h.

**5.1.2.1979 BEEP\_AddressBase** [26/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S105S4.h.

**5.1.2.1980 BEEP\_AddressBase** [27/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 67 of file STM8S105S6.h.

**5.1.2.1981 BEEP\_AddressBase** [28/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208S8.h.

**5.1.2.1982 BEEP\_AddressBase** [29/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF5288.h.

**5.1.2.1983 BEEP\_AddressBase** [30/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF5268.h.

**5.1.2.1984 BEEP\_AddressBase** [31/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF52A8.h.

**5.1.2.1985 BEEP\_AddressBase** [32/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208SB.h.

**5.1.2.1986 BEEP\_AddressBase** [33/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF52A9.h.

**5.1.2.1987 BEEP\_AddressBase** [34/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208CB.h.

**5.1.2.1988 BEEP\_AddressBase** [35/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208R8.h.

**5.1.2.1989 BEEP\_AddressBase** [36/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF5289.h.

**5.1.2.1990 BEEP\_AddressBase** [37/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208S6.h.

**5.1.2.1991 BEEP\_AddressBase** [38/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208MB.h.

**5.1.2.1992 BEEP\_AddressBase** [39/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF52AA.h.

**5.1.2.1993 BEEP\_AddressBase** [40/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF6286.h.

**5.1.2.1994 BEEP\_AddressBase** [41/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF5286.h.

**5.1.2.1995 BEEP\_AddressBase** [42/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF6288.h.

**5.1.2.1996 BEEP\_AddressBase** [43/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S007C8.h.

**5.1.2.1997 BEEP\_AddressBase** [44/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF6289.h.

**5.1.2.1998 BEEP\_AddressBase** [45/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF628A.h.



**5.1.2.1999 BEEP\_AddressBase** [46/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF528A.h.

**5.1.2.2000 BEEP\_AddressBase** [47/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF62A6.h.

**5.1.2.2001 BEEP\_AddressBase** [48/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207C6.h.

**5.1.2.2002 BEEP\_AddressBase** [49/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207C8.h.

**5.1.2.2003 BEEP\_AddressBase** [50/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207CB.h.

**5.1.2.2004 BEEP\_AddressBase** [51/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207K6.h.

**5.1.2.2005 BEEP\_AddressBase** [52/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF62A8.h.

**5.1.2.2006 BEEP\_AddressBase** [53/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207K8.h.

**5.1.2.2007 BEEP\_AddressBase** [54/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207M8.h.

**5.1.2.2008 BEEP\_AddressBase** [55/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207MB.h.

**5.1.2.2009 BEEP\_AddressBase** [56/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF62A9.h.

**5.1.2.2010 BEEP\_AddressBase** [57/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207R6.h.

**5.1.2.2011 BEEP\_AddressBase** [58/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207R8.h.

**5.1.2.2012 BEEP\_AddressBase** [59/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207RB.h.

**5.1.2.2013 BEEP\_AddressBase** [60/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207S6.h.

**5.1.2.2014 BEEP\_AddressBase** [61/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF52A6.h.

**5.1.2.2015 BEEP\_AddressBase** [62/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207S8.h.

**5.1.2.2016 BEEP\_AddressBase** [63/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF5269.h.

**5.1.2.2017 BEEP\_AddressBase** [64/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF62AA.h.

**5.1.2.2018 BEEP\_AddressBase** [65/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S207SB.h.

**5.1.2.2019 BEEP\_AddressBase** [66/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208C6.h.

**5.1.2.2020 BEEP\_AddressBase** [67/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208C8.h.

**5.1.2.2021 BEEP\_AddressBase** [68/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208R6.h.

**5.1.2.2022 BEEP\_AddressBase** [69/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8AF6388.h.

**5.1.2.2023 BEEP\_AddressBase** [70/70]

```
#define BEEP_AddressBase 0x50F3
```

Definition at line 69 of file STM8S208RB.h.

**5.1.2.2024 CAN\_AddressBase** [1/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF5269.h.

**5.1.2.2025 CAN\_AddressBase** [2/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF5288.h.

**5.1.2.2026 CAN\_AddressBase** [3/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF5268.h.

**5.1.2.2027 CAN\_AddressBase** [4/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF52A8.h.

**5.1.2.2028 CAN\_AddressBase** [5/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208SB.h.

**5.1.2.2029 CAN\_AddressBase** [6/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208S8.h.

**5.1.2.2030 CAN\_AddressBase** [7/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208R8.h.

**5.1.2.2031 CAN\_AddressBase** [8/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF52A9.h.

**5.1.2.2032 CAN\_AddressBase** [9/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208MB.h.

**5.1.2.2033 CAN\_AddressBase** [10/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208C8.h.

**5.1.2.2034 CAN\_AddressBase** [11/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208S6.h.

**5.1.2.2035 CAN\_AddressBase** [12/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208C6.h.

**5.1.2.2036 CAN\_AddressBase** [13/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF52AA.h.

**5.1.2.2037 CAN\_AddressBase** [14/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF5289.h.

**5.1.2.2038 CAN\_AddressBase** [15/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF5286.h.

**5.1.2.2039 CAN\_AddressBase** [16/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF528A.h.

**5.1.2.2040 CAN\_AddressBase** [17/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208CB.h.

**5.1.2.2041 CAN\_AddressBase** [18/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208R6.h.

**5.1.2.2042 CAN\_AddressBase** [19/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8AF52A6.h.

**5.1.2.2043 CAN\_AddressBase** [20/20]

```
#define CAN_AddressBase 0x5420
```

Definition at line 79 of file STM8S208RB.h.

**5.1.2.2044 CFG\_AddressBase** [1/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S903F3.h.

**5.1.2.2045 CFG\_AddressBase** [2/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S903K3.h.

**5.1.2.2046 CFG\_AddressBase** [3/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S001J3.h.



**5.1.2.2047 CFG\_AddressBase** [4/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S003F3.h.

**5.1.2.2048 CFG\_AddressBase** [5/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S003K3.h.

**5.1.2.2049 CFG\_AddressBase** [6/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S103F2.h.

**5.1.2.2050 CFG\_AddressBase** [7/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8AF6213.h.

**5.1.2.2051 CFG\_AddressBase** [8/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S103K3.h.

**5.1.2.2052 CFG\_AddressBase** [9/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8AF6213A.h.

**5.1.2.2053** CFG\_AddressBase [10/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8AF6223.h.

**5.1.2.2054** CFG\_AddressBase [11/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8AF6223A.h.

**5.1.2.2055** CFG\_AddressBase [12/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8S103F3.h.

**5.1.2.2056** CFG\_AddressBase [13/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 74 of file STM8AF6226.h.

**5.1.2.2057** CFG\_AddressBase [14/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 75 of file STM8AF6366.h.

**5.1.2.2058** CFG\_AddressBase [15/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8AF6246.h.

**5.1.2.2059** CFG\_AddressBase [16/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S005K6.h.

**5.1.2.2060** CFG\_AddressBase [17/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S005C6.h.

**5.1.2.2061** CFG\_AddressBase [18/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8AF6248.h.

**5.1.2.2062** CFG\_AddressBase [19/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8AF6266.h.

**5.1.2.2063** CFG\_AddressBase [20/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8AF6268.h.

**5.1.2.2064** CFG\_AddressBase [21/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8AF6269.h.

**5.1.2.2065** **CFG\_AddressBase** [22/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S105C6.h.

**5.1.2.2066** **CFG\_AddressBase** [23/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S105K6.h.

**5.1.2.2067** **CFG\_AddressBase** [24/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S105C4.h.

**5.1.2.2068** **CFG\_AddressBase** [25/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S105S4.h.

**5.1.2.2069** **CFG\_AddressBase** [26/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S105S6.h.

**5.1.2.2070** **CFG\_AddressBase** [27/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 76 of file STM8S105K4.h.

**5.1.2.2071** CFG\_AddressBase [28/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207S8.h.

**5.1.2.2072** CFG\_AddressBase [29/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF6286.h.

**5.1.2.2073** CFG\_AddressBase [30/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207K6.h.

**5.1.2.2074** CFG\_AddressBase [31/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207M8.h.

**5.1.2.2075** CFG\_AddressBase [32/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207R6.h.

**5.1.2.2076** CFG\_AddressBase [33/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF6288.h.

**5.1.2.2077** CFG\_AddressBase [34/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S007C8.h.

**5.1.2.2078** CFG\_AddressBase [35/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF628A.h.

**5.1.2.2079** CFG\_AddressBase [36/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF62A6.h.

**5.1.2.2080** CFG\_AddressBase [37/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207C6.h.

**5.1.2.2081** CFG\_AddressBase [38/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207C8.h.

**5.1.2.2082** CFG\_AddressBase [39/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207CB.h.

**5.1.2.2083** CFG\_AddressBase [40/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF62A8.h.

**5.1.2.2084** CFG\_AddressBase [41/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207K8.h.

**5.1.2.2085** CFG\_AddressBase [42/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207MB.h.

**5.1.2.2086** CFG\_AddressBase [43/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207R8.h.

**5.1.2.2087** CFG\_AddressBase [44/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF62A9.h.

**5.1.2.2088** CFG\_AddressBase [45/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207RB.h.

**5.1.2.2089** **CFG\_AddressBase** [46/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207S6.h.

**5.1.2.2090** **CFG\_AddressBase** [47/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF62AA.h.

**5.1.2.2091** **CFG\_AddressBase** [48/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8S207SB.h.

**5.1.2.2092** **CFG\_AddressBase** [49/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF6388.h.

**5.1.2.2093** **CFG\_AddressBase** [50/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 79 of file STM8AF6289.h.

**5.1.2.2094** **CFG\_AddressBase** [51/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208SB.h.



**5.1.2.2095** CFG\_AddressBase [52/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF5268.h.

**5.1.2.2096** CFG\_AddressBase [53/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF5269.h.

**5.1.2.2097** CFG\_AddressBase [54/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF5288.h.

**5.1.2.2098** CFG\_AddressBase [55/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF52A8.h.

**5.1.2.2099** CFG\_AddressBase [56/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208S8.h.

**5.1.2.2100** CFG\_AddressBase [57/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208R6.h.

**5.1.2.2101** **CFG\_AddressBase** [58/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF52A9.h.

**5.1.2.2102** **CFG\_AddressBase** [59/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208RB.h.

**5.1.2.2103** **CFG\_AddressBase** [60/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF5286.h.

**5.1.2.2104** **CFG\_AddressBase** [61/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208C6.h.

**5.1.2.2105** **CFG\_AddressBase** [62/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF5289.h.

**5.1.2.2106** **CFG\_AddressBase** [63/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF52AA.h.

**5.1.2.2107 CFG\_AddressBase** [64/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF528A.h.

**5.1.2.2108 CFG\_AddressBase** [65/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208C8.h.

**5.1.2.2109 CFG\_AddressBase** [66/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208CB.h.

**5.1.2.2110 CFG\_AddressBase** [67/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208MB.h.

**5.1.2.2111 CFG\_AddressBase** [68/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8AF52A6.h.

**5.1.2.2112 CFG\_AddressBase** [69/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208R8.h.

**5.1.2.2113** CFG\_AddressBase [70/70]

```
#define CFG_AddressBase 0x7F60
```

Definition at line 80 of file STM8S208S6.h.

**5.1.2.2114** CLK\_AddressBase [1/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S103F2.h.

**5.1.2.2115** CLK\_AddressBase [2/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S903F3.h.

**5.1.2.2116** CLK\_AddressBase [3/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S903K3.h.

**5.1.2.2117** CLK\_AddressBase [4/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S001J3.h.

**5.1.2.2118** CLK\_AddressBase [5/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S003F3.h.

**5.1.2.2119 CLK\_AddressBase** [6/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S003K3.h.

**5.1.2.2120 CLK\_AddressBase** [7/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8AF6213.h.

**5.1.2.2121 CLK\_AddressBase** [8/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S103F3.h.

**5.1.2.2122 CLK\_AddressBase** [9/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8S103K3.h.

**5.1.2.2123 CLK\_AddressBase** [10/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8AF6213A.h.

**5.1.2.2124 CLK\_AddressBase** [11/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8AF6223.h.

**5.1.2.2125 CLK\_AddressBase** [12/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8AF6223A.h.

**5.1.2.2126 CLK\_AddressBase** [13/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8AF6366.h.

**5.1.2.2127 CLK\_AddressBase** [14/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 62 of file STM8AF6226.h.

**5.1.2.2128 CLK\_AddressBase** [15/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8AF6246.h.

**5.1.2.2129 CLK\_AddressBase** [16/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8AF6248.h.

**5.1.2.2130 CLK\_AddressBase** [17/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8AF6266.h.

**5.1.2.2131 CLK\_AddressBase** [18/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8AF6268.h.

**5.1.2.2132 CLK\_AddressBase** [19/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8AF6269.h.

**5.1.2.2133 CLK\_AddressBase** [20/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S005C6.h.

**5.1.2.2134 CLK\_AddressBase** [21/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S005K6.h.

**5.1.2.2135 CLK\_AddressBase** [22/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S105C4.h.

**5.1.2.2136 CLK\_AddressBase** [23/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S105C6.h.

**5.1.2.2137 CLK\_AddressBase** [24/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S105S4.h.

**5.1.2.2138 CLK\_AddressBase** [25/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S105S6.h.

**5.1.2.2139 CLK\_AddressBase** [26/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S105K4.h.

**5.1.2.2140 CLK\_AddressBase** [27/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 63 of file STM8S105K6.h.

**5.1.2.2141 CLK\_AddressBase** [28/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208S8.h.

**5.1.2.2142 CLK\_AddressBase** [29/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208SB.h.



**5.1.2.2143 CLK\_AddressBase** [30/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF52A8.h.

**5.1.2.2144 CLK\_AddressBase** [31/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF6289.h.

**5.1.2.2145 CLK\_AddressBase** [32/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF52A9.h.

**5.1.2.2146 CLK\_AddressBase** [33/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF5289.h.

**5.1.2.2147 CLK\_AddressBase** [34/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208MB.h.

**5.1.2.2148 CLK\_AddressBase** [35/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF6286.h.

**5.1.2.2149 CLK\_AddressBase** [36/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF52AA.h.

**5.1.2.2150 CLK\_AddressBase** [37/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF6288.h.

**5.1.2.2151 CLK\_AddressBase** [38/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF5268.h.

**5.1.2.2152 CLK\_AddressBase** [39/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF5286.h.

**5.1.2.2153 CLK\_AddressBase** [40/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208RB.h.

**5.1.2.2154 CLK\_AddressBase** [41/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF528A.h.

**5.1.2.2155 CLK\_AddressBase** [42/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF62A6.h.

**5.1.2.2156 CLK\_AddressBase** [43/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207C6.h.

**5.1.2.2157 CLK\_AddressBase** [44/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207C8.h.

**5.1.2.2158 CLK\_AddressBase** [45/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207CB.h.

**5.1.2.2159 CLK\_AddressBase** [46/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207K6.h.

**5.1.2.2160 CLK\_AddressBase** [47/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF62A8.h.

**5.1.2.2161 CLK\_AddressBase** [48/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207K8.h.

**5.1.2.2162 CLK\_AddressBase** [49/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207M8.h.

**5.1.2.2163 CLK\_AddressBase** [50/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207MB.h.

**5.1.2.2164 CLK\_AddressBase** [51/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207R6.h.

**5.1.2.2165 CLK\_AddressBase** [52/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF62A9.h.

**5.1.2.2166 CLK\_AddressBase** [53/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207R8.h.

**5.1.2.2167 CLK\_AddressBase** [54/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207RB.h.

**5.1.2.2168 CLK\_AddressBase** [55/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207S6.h.

**5.1.2.2169 CLK\_AddressBase** [56/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF62AA.h.

**5.1.2.2170 CLK\_AddressBase** [57/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207S8.h.

**5.1.2.2171 CLK\_AddressBase** [58/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S207SB.h.

**5.1.2.2172 CLK\_AddressBase** [59/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF52A6.h.

**5.1.2.2173 CLK\_AddressBase** [60/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208C6.h.

**5.1.2.2174 CLK\_AddressBase** [61/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF5269.h.

**5.1.2.2175 CLK\_AddressBase** [62/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF628A.h.

**5.1.2.2176 CLK\_AddressBase** [63/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208C8.h.

**5.1.2.2177 CLK\_AddressBase** [64/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208CB.h.

**5.1.2.2178 CLK\_AddressBase** [65/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208R6.h.

**5.1.2.2179 CLK\_AddressBase** [66/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208R8.h.

**5.1.2.2180 CLK\_AddressBase** [67/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF6388.h.

**5.1.2.2181 CLK\_AddressBase** [68/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8AF5288.h.

**5.1.2.2182 CLK\_AddressBase** [69/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S208S6.h.

**5.1.2.2183 CLK\_AddressBase** [70/70]

```
#define CLK_AddressBase 0x50C0
```

Definition at line 65 of file STM8S007C8.h.

**5.1.2.2184 DISABLE\_INTERRUPTS**

```
#define DISABLE_INTERRUPTS( ) __asm__("sim")
```

disable interrupt handling

Definition at line 169 of file STM8AF\_STM8S.h.

**5.1.2.2185 DM\_AddressBase** [1/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S903F3.h.

**5.1.2.2186 DM\_AddressBase** [2/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S903K3.h.

**5.1.2.2187 DM\_AddressBase** [3/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S001J3.h.

**5.1.2.2188 DM\_AddressBase** [4/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S003F3.h.

**5.1.2.2189 DM\_AddressBase** [5/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S003K3.h.

**5.1.2.2190 DM\_AddressBase** [6/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S103F2.h.



**5.1.2.2191 DM\_AddressBase** [7/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S103F3.h.

**5.1.2.2192 DM\_AddressBase** [8/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8AF6213.h.

**5.1.2.2193 DM\_AddressBase** [9/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8S103K3.h.

**5.1.2.2194 DM\_AddressBase** [10/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8AF6213A.h.

**5.1.2.2195 DM\_AddressBase** [11/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8AF6223.h.

**5.1.2.2196 DM\_AddressBase** [12/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8AF6223A.h.

**5.1.2.2197 DM\_AddressBase** [13/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 76 of file STM8AF6226.h.

**5.1.2.2198 DM\_AddressBase** [14/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 77 of file STM8AF6366.h.

**5.1.2.2199 DM\_AddressBase** [15/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8AF6246.h.

**5.1.2.2200 DM\_AddressBase** [16/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8AF6248.h.

**5.1.2.2201 DM\_AddressBase** [17/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8AF6266.h.

**5.1.2.2202 DM\_AddressBase** [18/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8AF6268.h.

**5.1.2.2203 DM\_AddressBase** [19/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8AF6269.h.

**5.1.2.2204 DM\_AddressBase** [20/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S005C6.h.

**5.1.2.2205 DM\_AddressBase** [21/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S005K6.h.

**5.1.2.2206 DM\_AddressBase** [22/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S105C4.h.

**5.1.2.2207 DM\_AddressBase** [23/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S105C6.h.

**5.1.2.2208 DM\_AddressBase** [24/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S105K4.h.

**5.1.2.2209 DM\_AddressBase** [25/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S105S6.h.

**5.1.2.2210 DM\_AddressBase** [26/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S105S4.h.

**5.1.2.2211 DM\_AddressBase** [27/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 78 of file STM8S105K6.h.

**5.1.2.2212 DM\_AddressBase** [28/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S007C8.h.

**5.1.2.2213 DM\_AddressBase** [29/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF6286.h.

**5.1.2.2214 DM\_AddressBase** [30/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF6288.h.

**5.1.2.2215 DM\_AddressBase** [31/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF628A.h.

**5.1.2.2216 DM\_AddressBase** [32/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207C6.h.

**5.1.2.2217 DM\_AddressBase** [33/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF62A6.h.

**5.1.2.2218 DM\_AddressBase** [34/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207C8.h.

**5.1.2.2219 DM\_AddressBase** [35/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207CB.h.

**5.1.2.2220 DM\_AddressBase** [36/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207K6.h.

**5.1.2.2221 DM\_AddressBase** [37/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207K8.h.

**5.1.2.2222 DM\_AddressBase** [38/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF62A8.h.

**5.1.2.2223 DM\_AddressBase** [39/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207M8.h.

**5.1.2.2224 DM\_AddressBase** [40/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207MB.h.

**5.1.2.2225 DM\_AddressBase** [41/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207R6.h.

**5.1.2.2226 DM\_AddressBase** [42/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF62A9.h.

**5.1.2.2227 DM\_AddressBase** [43/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207R8.h.

**5.1.2.2228 DM\_AddressBase** [44/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207RB.h.

**5.1.2.2229 DM\_AddressBase** [45/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207S6.h.

**5.1.2.2230 DM\_AddressBase** [46/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207S8.h.

**5.1.2.2231 DM\_AddressBase** [47/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8S207SB.h.

**5.1.2.2232 DM\_AddressBase** [48/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF62AA.h.

**5.1.2.2233 DM\_AddressBase** [49/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF6289.h.

**5.1.2.2234 DM\_AddressBase** [50/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 81 of file STM8AF6388.h.

**5.1.2.2235 DM\_AddressBase** [51/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208S8.h.

**5.1.2.2236 DM\_AddressBase** [52/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF5269.h.

**5.1.2.2237 DM\_AddressBase** [53/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF52A8.h.

**5.1.2.2238 DM\_AddressBase** [54/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF5288.h.



**5.1.2.2239 DM\_AddressBase** [55/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208SB.h.

**5.1.2.2240 DM\_AddressBase** [56/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF5268.h.

**5.1.2.2241 DM\_AddressBase** [57/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF52A9.h.

**5.1.2.2242 DM\_AddressBase** [58/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208C8.h.

**5.1.2.2243 DM\_AddressBase** [59/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208RB.h.

**5.1.2.2244 DM\_AddressBase** [60/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF528A.h.

**5.1.2.2245 DM\_AddressBase** [61/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF5289.h.

**5.1.2.2246 DM\_AddressBase** [62/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF52AA.h.

**5.1.2.2247 DM\_AddressBase** [63/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF5286.h.

**5.1.2.2248 DM\_AddressBase** [64/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208C6.h.

**5.1.2.2249 DM\_AddressBase** [65/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208CB.h.

**5.1.2.2250 DM\_AddressBase** [66/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208MB.h.

**5.1.2.2251 DM\_AddressBase** [67/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208R6.h.

**5.1.2.2252 DM\_AddressBase** [68/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208R8.h.

**5.1.2.2253 DM\_AddressBase** [69/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8AF52A6.h.

**5.1.2.2254 DM\_AddressBase** [70/70]

```
#define DM_AddressBase 0x7F90
```

Definition at line 82 of file STM8S208S6.h.

**5.1.2.2255 ENABLE\_INTERRUPTS**

```
#define ENABLE_INTERRUPTS( ) __asm__("rim")
```

enable interrupt handling

Definition at line 170 of file STM8AF\_STM8S.h.

**5.1.2.2256 ENTER\_HALT**

```
#define ENTER_HALT( ) __asm__("halt")
```

put controller to HALT mode

Definition at line 173 of file STM8AF\_STM8S.h.

**5.1.2.2257** **EXTI\_AddressBase** [1/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S903F3.h.

**5.1.2.2258** **EXTI\_AddressBase** [2/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S903K3.h.

**5.1.2.2259** **EXTI\_AddressBase** [3/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8AF6226.h.

**5.1.2.2260** **EXTI\_AddressBase** [4/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S001J3.h.

**5.1.2.2261** **EXTI\_AddressBase** [5/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S003F3.h.

**5.1.2.2262** **EXTI\_AddressBase** [6/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S003K3.h.

**5.1.2.2263** EXTI\_AddressBase [7/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8AF6213.h.

**5.1.2.2264** EXTI\_AddressBase [8/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S103F2.h.

**5.1.2.2265** EXTI\_AddressBase [9/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S103F3.h.

**5.1.2.2266** EXTI\_AddressBase [10/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8S103K3.h.

**5.1.2.2267** EXTI\_AddressBase [11/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8AF6223.h.

**5.1.2.2268** EXTI\_AddressBase [12/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8AF6223A.h.

**5.1.2.2269** **EXTI\_AddressBase** [13/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8AF6213A.h.

**5.1.2.2270** **EXTI\_AddressBase** [14/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 60 of file STM8AF6366.h.

**5.1.2.2271** **EXTI\_AddressBase** [15/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8AF6248.h.

**5.1.2.2272** **EXTI\_AddressBase** [16/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8AF6266.h.

**5.1.2.2273** **EXTI\_AddressBase** [17/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8AF6246.h.

**5.1.2.2274** **EXTI\_AddressBase** [18/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8AF6268.h.

**5.1.2.2275** EXTI\_AddressBase [19/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8AF6269.h.

**5.1.2.2276** EXTI\_AddressBase [20/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S005C6.h.

**5.1.2.2277** EXTI\_AddressBase [21/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S005K6.h.

**5.1.2.2278** EXTI\_AddressBase [22/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S105C4.h.

**5.1.2.2279** EXTI\_AddressBase [23/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S105C6.h.

**5.1.2.2280** EXTI\_AddressBase [24/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S105K4.h.

**5.1.2.2281** **EXTI\_AddressBase** [25/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S105K6.h.

**5.1.2.2282** **EXTI\_AddressBase** [26/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S105S6.h.

**5.1.2.2283** **EXTI\_AddressBase** [27/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 61 of file STM8S105S4.h.

**5.1.2.2284** **EXTI\_AddressBase** [28/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208SB.h.

**5.1.2.2285** **EXTI\_AddressBase** [29/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF52A8.h.

**5.1.2.2286** **EXTI\_AddressBase** [30/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207RB.h.



**5.1.2.2287** EXTI\_AddressBase [31/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207S8.h.

**5.1.2.2288** EXTI\_AddressBase [32/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF52A9.h.

**5.1.2.2289** EXTI\_AddressBase [33/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208C6.h.

**5.1.2.2290** EXTI\_AddressBase [34/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208S8.h.

**5.1.2.2291** EXTI\_AddressBase [35/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208R8.h.

**5.1.2.2292** EXTI\_AddressBase [36/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF5289.h.

**5.1.2.2293** **EXTI\_AddressBase** [37/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207C8.h.

**5.1.2.2294** **EXTI\_AddressBase** [38/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF52AA.h.

**5.1.2.2295** **EXTI\_AddressBase** [39/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF6286.h.

**5.1.2.2296** **EXTI\_AddressBase** [40/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207M8.h.

**5.1.2.2297** **EXTI\_AddressBase** [41/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207K6.h.

**5.1.2.2298** **EXTI\_AddressBase** [42/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207R6.h.

**5.1.2.2299** EXTI\_AddressBase [43/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF5268.h.

**5.1.2.2300** EXTI\_AddressBase [44/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF5286.h.

**5.1.2.2301** EXTI\_AddressBase [45/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF6288.h.

**5.1.2.2302** EXTI\_AddressBase [46/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S007C8.h.

**5.1.2.2303** EXTI\_AddressBase [47/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF528A.h.

**5.1.2.2304** EXTI\_AddressBase [48/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF62A6.h.

**5.1.2.2305** **EXTI\_AddressBase** [49/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207C6.h.

**5.1.2.2306** **EXTI\_AddressBase** [50/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF5269.h.

**5.1.2.2307** **EXTI\_AddressBase** [51/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207CB.h.

**5.1.2.2308** **EXTI\_AddressBase** [52/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF62A8.h.

**5.1.2.2309** **EXTI\_AddressBase** [53/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207K8.h.

**5.1.2.2310** **EXTI\_AddressBase** [54/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207MB.h.

**5.1.2.2311** EXTI\_AddressBase [55/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF62A9.h.

**5.1.2.2312** EXTI\_AddressBase [56/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207R8.h.

**5.1.2.2313** EXTI\_AddressBase [57/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207S6.h.

**5.1.2.2314** EXTI\_AddressBase [58/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF62AA.h.

**5.1.2.2315** EXTI\_AddressBase [59/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF52A6.h.

**5.1.2.2316** EXTI\_AddressBase [60/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S207SB.h.

**5.1.2.2317** **EXTI\_AddressBase** [61/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF628A.h.

**5.1.2.2318** **EXTI\_AddressBase** [62/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208C8.h.

**5.1.2.2319** **EXTI\_AddressBase** [63/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208CB.h.

**5.1.2.2320** **EXTI\_AddressBase** [64/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208MB.h.

**5.1.2.2321** **EXTI\_AddressBase** [65/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208R6.h.

**5.1.2.2322** **EXTI\_AddressBase** [66/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF5288.h.

**5.1.2.2323** EXTI\_AddressBase [67/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF6388.h.

**5.1.2.2324** EXTI\_AddressBase [68/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208RB.h.

**5.1.2.2325** EXTI\_AddressBase [69/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8AF6289.h.

**5.1.2.2326** EXTI\_AddressBase [70/70]

```
#define EXTI_AddressBase 0x50A0
```

Definition at line 63 of file STM8S208S6.h.

**5.1.2.2327** FLASH\_AddressBase [1/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S903K3.h.

**5.1.2.2328** FLASH\_AddressBase [2/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S903F3.h.

**5.1.2.2329 FLASH\_AddressBase** [3/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8AF6366.h.

**5.1.2.2330 FLASH\_AddressBase** [4/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8AF6226.h.

**5.1.2.2331 FLASH\_AddressBase** [5/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S001J3.h.

**5.1.2.2332 FLASH\_AddressBase** [6/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S003F3.h.

**5.1.2.2333 FLASH\_AddressBase** [7/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S003K3.h.

**5.1.2.2334 FLASH\_AddressBase** [8/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8AF6213.h.



**5.1.2.2335 FLASH\_AddressBase** [9/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S103K3.h.

**5.1.2.2336 FLASH\_AddressBase** [10/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8AF6213A.h.

**5.1.2.2337 FLASH\_AddressBase** [11/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8AF6223.h.

**5.1.2.2338 FLASH\_AddressBase** [12/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8AF6223A.h.

**5.1.2.2339 FLASH\_AddressBase** [13/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S103F3.h.

**5.1.2.2340 FLASH\_AddressBase** [14/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 59 of file STM8S103F2.h.

**5.1.2.2341 FLASH\_AddressBase** [15/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8AF6246.h.

**5.1.2.2342 FLASH\_AddressBase** [16/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8AF6248.h.

**5.1.2.2343 FLASH\_AddressBase** [17/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8AF6266.h.

**5.1.2.2344 FLASH\_AddressBase** [18/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8AF6268.h.

**5.1.2.2345 FLASH\_AddressBase** [19/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8AF6269.h.

**5.1.2.2346 FLASH\_AddressBase** [20/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S005C6.h.

**5.1.2.2347 FLASH\_AddressBase** [21/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S005K6.h.

**5.1.2.2348 FLASH\_AddressBase** [22/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S105C4.h.

**5.1.2.2349 FLASH\_AddressBase** [23/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S105K4.h.

**5.1.2.2350 FLASH\_AddressBase** [24/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S105S4.h.

**5.1.2.2351 FLASH\_AddressBase** [25/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S105S6.h.

**5.1.2.2352 FLASH\_AddressBase** [26/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S105K6.h.

**5.1.2.2353 FLASH\_AddressBase** [27/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 60 of file STM8S105C6.h.

**5.1.2.2354 FLASH\_AddressBase** [28/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208S8.h.

**5.1.2.2355 FLASH\_AddressBase** [29/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF52A8.h.

**5.1.2.2356 FLASH\_AddressBase** [30/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF52A6.h.

**5.1.2.2357 FLASH\_AddressBase** [31/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF52A9.h.

**5.1.2.2358 FLASH\_AddressBase** [32/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208SB.h.

**5.1.2.2359 FLASH\_AddressBase** [33/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF5268.h.

**5.1.2.2360 FLASH\_AddressBase** [34/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208CB.h.

**5.1.2.2361 FLASH\_AddressBase** [35/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208R8.h.

**5.1.2.2362 FLASH\_AddressBase** [36/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208C8.h.

**5.1.2.2363 FLASH\_AddressBase** [37/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208S6.h.

**5.1.2.2364 FLASH\_AddressBase** [38/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF5288.h.

**5.1.2.2365 FLASH\_AddressBase** [39/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF52AA.h.

**5.1.2.2366 FLASH\_AddressBase** [40/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208MB.h.

**5.1.2.2367 FLASH\_AddressBase** [41/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF6286.h.

**5.1.2.2368 FLASH\_AddressBase** [42/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF5286.h.

**5.1.2.2369 FLASH\_AddressBase** [43/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF6288.h.

**5.1.2.2370 FLASH\_AddressBase** [44/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S007C8.h.

**5.1.2.2371 FLASH\_AddressBase** [45/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF6289.h.

**5.1.2.2372 FLASH\_AddressBase** [46/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF628A.h.

**5.1.2.2373 FLASH\_AddressBase** [47/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF528A.h.

**5.1.2.2374 FLASH\_AddressBase** [48/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF5289.h.

**5.1.2.2375 FLASH\_AddressBase** [49/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF62A6.h.

**5.1.2.2376 FLASH\_AddressBase** [50/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207C6.h.

**5.1.2.2377 FLASH\_AddressBase** [51/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF5269.h.

**5.1.2.2378 FLASH\_AddressBase** [52/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207C8.h.

**5.1.2.2379 FLASH\_AddressBase** [53/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207CB.h.

**5.1.2.2380 FLASH\_AddressBase** [54/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF62A8.h.

**5.1.2.2381 FLASH\_AddressBase** [55/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207K6.h.

**5.1.2.2382 FLASH\_AddressBase** [56/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207K8.h.



**5.1.2.2383 FLASH\_AddressBase** [57/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207M8.h.

**5.1.2.2384 FLASH\_AddressBase** [58/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207MB.h.

**5.1.2.2385 FLASH\_AddressBase** [59/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF62A9.h.

**5.1.2.2386 FLASH\_AddressBase** [60/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207R6.h.

**5.1.2.2387 FLASH\_AddressBase** [61/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207R8.h.

**5.1.2.2388 FLASH\_AddressBase** [62/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207RB.h.

**5.1.2.2389 FLASH\_AddressBase** [63/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207S6.h.

**5.1.2.2390 FLASH\_AddressBase** [64/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF62AA.h.

**5.1.2.2391 FLASH\_AddressBase** [65/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207S8.h.

**5.1.2.2392 FLASH\_AddressBase** [66/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S207SB.h.

**5.1.2.2393 FLASH\_AddressBase** [67/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208C6.h.

**5.1.2.2394 FLASH\_AddressBase** [68/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208R6.h.

**5.1.2.2395 FLASH\_AddressBase** [69/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8AF6388.h.

**5.1.2.2396 FLASH\_AddressBase** [70/70]

```
#define FLASH_AddressBase 0x505A
```

Definition at line 62 of file STM8S208RB.h.

**5.1.2.2397 I2C\_AddressBase** [1/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S903F3.h.

**5.1.2.2398 I2C\_AddressBase** [2/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S903K3.h.

**5.1.2.2399 I2C\_AddressBase** [3/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8AF6226.h.

**5.1.2.2400 I2C\_AddressBase** [4/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S001J3.h.

**5.1.2.2401 I2C\_AddressBase** [5/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S003F3.h.

**5.1.2.2402 I2C\_AddressBase** [6/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S003K3.h.

**5.1.2.2403 I2C\_AddressBase** [7/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8AF6213.h.

**5.1.2.2404 I2C\_AddressBase** [8/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S103F2.h.

**5.1.2.2405 I2C\_AddressBase** [9/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S103F3.h.

**5.1.2.2406 I2C\_AddressBase** [10/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8S103K3.h.

**5.1.2.2407 I2C\_AddressBase** [11/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8AF6213A.h.

**5.1.2.2408 I2C\_AddressBase** [12/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8AF6223.h.

**5.1.2.2409 I2C\_AddressBase** [13/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8AF6223A.h.

**5.1.2.2410 I2C\_AddressBase** [14/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 68 of file STM8AF6366.h.

**5.1.2.2411 I2C\_AddressBase** [15/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8AF6246.h.

**5.1.2.2412 I2C\_AddressBase** [16/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8AF6248.h.

**5.1.2.2413 I2C\_AddressBase** [17/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8AF6266.h.

**5.1.2.2414 I2C\_AddressBase** [18/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8AF6268.h.

**5.1.2.2415 I2C\_AddressBase** [19/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8AF6269.h.

**5.1.2.2416 I2C\_AddressBase** [20/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S005C6.h.

**5.1.2.2417 I2C\_AddressBase** [21/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S005K6.h.

**5.1.2.2418 I2C\_AddressBase** [22/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S105C4.h.

**5.1.2.2419 I2C\_AddressBase** [23/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S105C6.h.

**5.1.2.2420 I2C\_AddressBase** [24/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S105K4.h.

**5.1.2.2421 I2C\_AddressBase** [25/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S105K6.h.

**5.1.2.2422 I2C\_AddressBase** [26/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S105S4.h.

**5.1.2.2423 I2C\_AddressBase** [27/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 69 of file STM8S105S6.h.

**5.1.2.2424 I2C\_AddressBase** [28/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF6289.h.

**5.1.2.2425 I2C\_AddressBase** [29/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207RB.h.

**5.1.2.2426 I2C\_AddressBase** [30/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF5288.h.

**5.1.2.2427 I2C\_AddressBase** [31/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207R6.h.

**5.1.2.2428 I2C\_AddressBase** [32/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF52A8.h.

**5.1.2.2429 I2C\_AddressBase** [33/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207S8.h.

**5.1.2.2430 I2C\_AddressBase** [34/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208SB.h.



**5.1.2.2431 I2C\_AddressBase** [35/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208S8.h.

**5.1.2.2432 I2C\_AddressBase** [36/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF52A9.h.

**5.1.2.2433 I2C\_AddressBase** [37/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208C6.h.

**5.1.2.2434 I2C\_AddressBase** [38/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208R8.h.

**5.1.2.2435 I2C\_AddressBase** [39/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF52A6.h.

**5.1.2.2436 I2C\_AddressBase** [40/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207C8.h.

**5.1.2.2437 I2C\_AddressBase** [41/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207M8.h.

**5.1.2.2438 I2C\_AddressBase** [42/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF5289.h.

**5.1.2.2439 I2C\_AddressBase** [43/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF5268.h.

**5.1.2.2440 I2C\_AddressBase** [44/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF6286.h.

**5.1.2.2441 I2C\_AddressBase** [45/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF52AA.h.

**5.1.2.2442 I2C\_AddressBase** [46/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207K6.h.

**5.1.2.2443 I2C\_AddressBase** [47/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF6288.h.

**5.1.2.2444 I2C\_AddressBase** [48/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF628A.h.

**5.1.2.2445 I2C\_AddressBase** [49/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF528A.h.

**5.1.2.2446 I2C\_AddressBase** [50/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF62A6.h.

**5.1.2.2447 I2C\_AddressBase** [51/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207C6.h.

**5.1.2.2448 I2C\_AddressBase** [52/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207CB.h.

**5.1.2.2449 I2C\_AddressBase** [53/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF62A8.h.

**5.1.2.2450 I2C\_AddressBase** [54/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207K8.h.

**5.1.2.2451 I2C\_AddressBase** [55/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207MB.h.

**5.1.2.2452 I2C\_AddressBase** [56/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF62A9.h.

**5.1.2.2453 I2C\_AddressBase** [57/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207R8.h.

**5.1.2.2454 I2C\_AddressBase** [58/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207S6.h.

**5.1.2.2455 I2C\_AddressBase** [59/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF62AA.h.

**5.1.2.2456 I2C\_AddressBase** [60/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S207SB.h.

**5.1.2.2457 I2C\_AddressBase** [61/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208C8.h.

**5.1.2.2458 I2C\_AddressBase** [62/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208CB.h.

**5.1.2.2459 I2C\_AddressBase** [63/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208MB.h.

**5.1.2.2460 I2C\_AddressBase** [64/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF5286.h.

**5.1.2.2461 I2C\_AddressBase** [65/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S007C8.h.

**5.1.2.2462 I2C\_AddressBase** [66/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208R6.h.

**5.1.2.2463 I2C\_AddressBase** [67/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF5269.h.

**5.1.2.2464 I2C\_AddressBase** [68/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8AF6388.h.

**5.1.2.2465 I2C\_AddressBase** [69/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208RB.h.

**5.1.2.2466 I2C\_AddressBase** [70/70]

```
#define I2C_AddressBase 0x5210
```

Definition at line 71 of file STM8S208S6.h.

**5.1.2.2467 ISR\_HANDLER**

```
#define ISR_HANDLER(  
    func,  
    irq ) void func(void) __interrupt(irq)
```

handler for interrupt service routine

Definition at line 160 of file STM8AF\_STM8S.h.

**5.1.2.2468 ISR\_HANDLER\_TRAP**

```
#define ISR_HANDLER_TRAP(  
    func ) void func() __trap
```

handler for trap service routine

Definition at line 162 of file STM8AF\_STM8S.h.

**5.1.2.2469 ITC\_AddressBase [1/70]**

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S103F2.h.

**5.1.2.2470 ITC\_AddressBase [2/70]**

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S903K3.h.

**5.1.2.2471 ITC\_AddressBase [3/70]**

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S903F3.h.

**5.1.2.2472 ITC\_AddressBase** [4/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S001J3.h.

**5.1.2.2473 ITC\_AddressBase** [5/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S003F3.h.

**5.1.2.2474 ITC\_AddressBase** [6/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S003K3.h.

**5.1.2.2475 ITC\_AddressBase** [7/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8AF6213.h.

**5.1.2.2476 ITC\_AddressBase** [8/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8AF6223.h.

**5.1.2.2477 ITC\_AddressBase** [9/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8AF6213A.h.



**5.1.2.2478 ITC\_AddressBase** [10/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8AF6223A.h.

**5.1.2.2479 ITC\_AddressBase** [11/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S103F3.h.

**5.1.2.2480 ITC\_AddressBase** [12/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8S103K3.h.

**5.1.2.2481 ITC\_AddressBase** [13/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 75 of file STM8AF6226.h.

**5.1.2.2482 ITC\_AddressBase** [14/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 76 of file STM8AF6366.h.

**5.1.2.2483 ITC\_AddressBase** [15/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S005C6.h.

**5.1.2.2484 ITC\_AddressBase** [16/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8AF6246.h.

**5.1.2.2485 ITC\_AddressBase** [17/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S005K6.h.

**5.1.2.2486 ITC\_AddressBase** [18/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8AF6248.h.

**5.1.2.2487 ITC\_AddressBase** [19/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8AF6266.h.

**5.1.2.2488 ITC\_AddressBase** [20/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8AF6268.h.

**5.1.2.2489 ITC\_AddressBase** [21/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8AF6269.h.

**5.1.2.2490 ITC\_AddressBase** [22/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S105S4.h.

**5.1.2.2491 ITC\_AddressBase** [23/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S105C4.h.

**5.1.2.2492 ITC\_AddressBase** [24/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S105K6.h.

**5.1.2.2493 ITC\_AddressBase** [25/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S105C6.h.

**5.1.2.2494 ITC\_AddressBase** [26/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S105K4.h.

**5.1.2.2495 ITC\_AddressBase** [27/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 77 of file STM8S105S6.h.

**5.1.2.2496 ITC\_AddressBase** [28/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S007C8.h.

**5.1.2.2497 ITC\_AddressBase** [29/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF6288.h.

**5.1.2.2498 ITC\_AddressBase** [30/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207R8.h.

**5.1.2.2499 ITC\_AddressBase** [31/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207RB.h.

**5.1.2.2500 ITC\_AddressBase** [32/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207S6.h.

**5.1.2.2501 ITC\_AddressBase** [33/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207S8.h.

**5.1.2.2502 ITC\_AddressBase** [34/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207SB.h.

**5.1.2.2503 ITC\_AddressBase** [35/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF62AA.h.

**5.1.2.2504 ITC\_AddressBase** [36/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207CB.h.

**5.1.2.2505 ITC\_AddressBase** [37/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207C8.h.

**5.1.2.2506 ITC\_AddressBase** [38/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207M8.h.

**5.1.2.2507 ITC\_AddressBase** [39/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207K8.h.

**5.1.2.2508 ITC\_AddressBase** [40/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF6286.h.

**5.1.2.2509 ITC\_AddressBase** [41/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207K6.h.

**5.1.2.2510 ITC\_AddressBase** [42/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207R6.h.

**5.1.2.2511 ITC\_AddressBase** [43/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207MB.h.

**5.1.2.2512 ITC\_AddressBase** [44/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF6289.h.

**5.1.2.2513 ITC\_AddressBase** [45/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF628A.h.

**5.1.2.2514 ITC\_AddressBase** [46/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF62A6.h.

**5.1.2.2515 ITC\_AddressBase** [47/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF62A8.h.

**5.1.2.2516 ITC\_AddressBase** [48/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF62A9.h.

**5.1.2.2517 ITC\_AddressBase** [49/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8S207C6.h.

**5.1.2.2518 ITC\_AddressBase** [50/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 80 of file STM8AF6388.h.

**5.1.2.2519 ITC\_AddressBase** [51/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF5268.h.

**5.1.2.2520 ITC\_AddressBase** [52/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF5269.h.

**5.1.2.2521 ITC\_AddressBase** [53/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF5288.h.

**5.1.2.2522 ITC\_AddressBase** [54/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF52A8.h.

**5.1.2.2523 ITC\_AddressBase** [55/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208SB.h.

**5.1.2.2524 ITC\_AddressBase** [56/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208C8.h.

**5.1.2.2525 ITC\_AddressBase** [57/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208S8.h.



**5.1.2.2526 ITC\_AddressBase** [58/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208R8.h.

**5.1.2.2527 ITC\_AddressBase** [59/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF52A6.h.

**5.1.2.2528 ITC\_AddressBase** [60/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208R6.h.

**5.1.2.2529 ITC\_AddressBase** [61/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF52A9.h.

**5.1.2.2530 ITC\_AddressBase** [62/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208CB.h.

**5.1.2.2531 ITC\_AddressBase** [63/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208MB.h.

**5.1.2.2532 ITC\_AddressBase** [64/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208C6.h.

**5.1.2.2533 ITC\_AddressBase** [65/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208S6.h.

**5.1.2.2534 ITC\_AddressBase** [66/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8S208RB.h.

**5.1.2.2535 ITC\_AddressBase** [67/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF5289.h.

**5.1.2.2536 ITC\_AddressBase** [68/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF52AA.h.

**5.1.2.2537 ITC\_AddressBase** [69/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF5286.h.

**5.1.2.2538 ITC\_AddressBase** [70/70]

```
#define ITC_AddressBase 0x7F70
```

Definition at line 81 of file STM8AF528A.h.

**5.1.2.2539 IWDG\_AddressBase** [1/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S103K3.h.

**5.1.2.2540 IWDG\_AddressBase** [2/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S103F2.h.

**5.1.2.2541 IWDG\_AddressBase** [3/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S903F3.h.

**5.1.2.2542 IWDG\_AddressBase** [4/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8AF6213.h.

**5.1.2.2543 IWDG\_AddressBase** [5/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S903K3.h.

**5.1.2.2544 IWDG\_AddressBase** [6/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S001J3.h.

**5.1.2.2545 IWDG\_AddressBase** [7/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S003F3.h.

**5.1.2.2546 IWDG\_AddressBase** [8/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S003K3.h.

**5.1.2.2547 IWDG\_AddressBase** [9/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8AF6213A.h.

**5.1.2.2548 IWDG\_AddressBase** [10/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8AF6223.h.

**5.1.2.2549 IWDG\_AddressBase** [11/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8AF6223A.h.

**5.1.2.2550 IWDG\_AddressBase** [12/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8AF6366.h.

**5.1.2.2551 IWDG\_AddressBase** [13/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8AF6226.h.

**5.1.2.2552 IWDG\_AddressBase** [14/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 64 of file STM8S103F3.h.

**5.1.2.2553 IWDG\_AddressBase** [15/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8AF6246.h.

**5.1.2.2554 IWDG\_AddressBase** [16/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S005K6.h.

**5.1.2.2555 IWDG\_AddressBase** [17/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S005C6.h.

**5.1.2.2556 IWDG\_AddressBase** [18/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8AF6248.h.

**5.1.2.2557 IWDG\_AddressBase** [19/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8AF6266.h.

**5.1.2.2558 IWDG\_AddressBase** [20/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8AF6268.h.

**5.1.2.2559 IWDG\_AddressBase** [21/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8AF6269.h.

**5.1.2.2560 IWDG\_AddressBase** [22/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S105K6.h.

**5.1.2.2561 IWDG\_AddressBase** [23/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S105S6.h.

**5.1.2.2562 IWDG\_AddressBase** [24/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S105S4.h.

**5.1.2.2563 IWDG\_AddressBase** [25/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S105C4.h.

**5.1.2.2564 IWDG\_AddressBase** [26/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S105C6.h.

**5.1.2.2565 IWDG\_AddressBase** [27/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 65 of file STM8S105K4.h.

**5.1.2.2566 IWDG\_AddressBase** [28/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF5268.h.

**5.1.2.2567 IWDG\_AddressBase** [29/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF5288.h.

**5.1.2.2568 IWDG\_AddressBase** [30/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207RB.h.

**5.1.2.2569 IWDG\_AddressBase** [31/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF52A8.h.

**5.1.2.2570 IWDG\_AddressBase** [32/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207R8.h.

**5.1.2.2571 IWDG\_AddressBase** [33/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF6288.h.

**5.1.2.2572 IWDG\_AddressBase** [34/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207S6.h.

**5.1.2.2573 IWDG\_AddressBase** [35/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF5269.h.



**5.1.2.2574 IWDG\_AddressBase** [36/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF62AA.h.

**5.1.2.2575 IWDG\_AddressBase** [37/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207S8.h.

**5.1.2.2576 IWDG\_AddressBase** [38/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208SB.h.

**5.1.2.2577 IWDG\_AddressBase** [39/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207SB.h.

**5.1.2.2578 IWDG\_AddressBase** [40/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF52A9.h.

**5.1.2.2579 IWDG\_AddressBase** [41/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208S8.h.

**5.1.2.2580 IWDG\_AddressBase** [42/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208CB.h.

**5.1.2.2581 IWDG\_AddressBase** [43/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208C6.h.

**5.1.2.2582 IWDG\_AddressBase** [44/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208R8.h.

**5.1.2.2583 IWDG\_AddressBase** [45/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208R6.h.

**5.1.2.2584 IWDG\_AddressBase** [46/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208C8.h.

**5.1.2.2585 IWDG\_AddressBase** [47/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208RB.h.

**5.1.2.2586 IWDG\_AddressBase** [48/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF5289.h.

**5.1.2.2587 IWDG\_AddressBase** [49/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207C8.h.

**5.1.2.2588 IWDG\_AddressBase** [50/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207CB.h.

**5.1.2.2589 IWDG\_AddressBase** [51/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207M8.h.

**5.1.2.2590 IWDG\_AddressBase** [52/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF6286.h.

**5.1.2.2591 IWDG\_AddressBase** [53/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF52AA.h.

**5.1.2.2592 IWDG\_AddressBase** [54/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207K6.h.

**5.1.2.2593 IWDG\_AddressBase** [55/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207K8.h.

**5.1.2.2594 IWDG\_AddressBase** [56/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207MB.h.

**5.1.2.2595 IWDG\_AddressBase** [57/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF62A8.h.

**5.1.2.2596 IWDG\_AddressBase** [58/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF62A9.h.

**5.1.2.2597 IWDG\_AddressBase** [59/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207R6.h.

**5.1.2.2598 IWDG\_AddressBase** [60/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF5286.h.

**5.1.2.2599 IWDG\_AddressBase** [61/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF6289.h.

**5.1.2.2600 IWDG\_AddressBase** [62/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF528A.h.

**5.1.2.2601 IWDG\_AddressBase** [63/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF62A6.h.

**5.1.2.2602 IWDG\_AddressBase** [64/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S207C6.h.

**5.1.2.2603 IWDG\_AddressBase** [65/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF52A6.h.

**5.1.2.2604 IWDG\_AddressBase** [66/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF628A.h.

**5.1.2.2605 IWDG\_AddressBase** [67/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208MB.h.

**5.1.2.2606 IWDG\_AddressBase** [68/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8AF6388.h.

**5.1.2.2607 IWDG\_AddressBase** [69/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S007C8.h.

**5.1.2.2608 IWDG\_AddressBase** [70/70]

```
#define IWDG_AddressBase 0x50E0
```

Definition at line 67 of file STM8S208S6.h.

**5.1.2.2609 NOP**

```
#define NOP( ) __asm__("nop")
```

perform a nop() operation (=minimum delay)

Definition at line 168 of file STM8AF\_STM8S.h.

**5.1.2.2610** OPT\_AddressBase [1/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208S8.h.

**5.1.2.2611** OPT\_AddressBase [2/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208SB.h.

**5.1.2.2612** OPT\_AddressBase [3/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6246.h.

**5.1.2.2613** OPT\_AddressBase [4/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207RB.h.

**5.1.2.2614** OPT\_AddressBase [5/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6289.h.

**5.1.2.2615** OPT\_AddressBase [6/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6248.h.

**5.1.2.2616** `OPT_AddressBase` [7/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6213.h.

**5.1.2.2617** `OPT_AddressBase` [8/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6223A.h.

**5.1.2.2618** `OPT_AddressBase` [9/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF5268.h.

**5.1.2.2619** `OPT_AddressBase` [10/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6266.h.

**5.1.2.2620** `OPT_AddressBase` [11/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207S8.h.

**5.1.2.2621** `OPT_AddressBase` [12/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF52A9.h.



**5.1.2.2622** OPT\_AddressBase [13/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF5289.h.

**5.1.2.2623** OPT\_AddressBase [14/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207CB.h.

**5.1.2.2624** OPT\_AddressBase [15/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207S6.h.

**5.1.2.2625** OPT\_AddressBase [16/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S903F3.h.

**5.1.2.2626** OPT\_AddressBase [17/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207SB.h.

**5.1.2.2627** OPT\_AddressBase [18/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF5286.h.

**5.1.2.2628** `OPT_AddressBase` [19/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6268.h.

**5.1.2.2629** `OPT_AddressBase` [20/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S903K3.h.

**5.1.2.2630** `OPT_AddressBase` [21/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF52A8.h.

**5.1.2.2631** `OPT_AddressBase` [22/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208C6.h.

**5.1.2.2632** `OPT_AddressBase` [23/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208R8.h.

**5.1.2.2633** `OPT_AddressBase` [24/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6269.h.

**5.1.2.2634** OPT\_AddressBase [25/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207C8.h.

**5.1.2.2635** OPT\_AddressBase [26/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF5269.h.

**5.1.2.2636** OPT\_AddressBase [27/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF52AA.h.

**5.1.2.2637** OPT\_AddressBase [28/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6286.h.

**5.1.2.2638** OPT\_AddressBase [29/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208R6.h.

**5.1.2.2639** OPT\_AddressBase [30/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207K6.h.

**5.1.2.2640 OPT\_AddressBase** [31/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207M8.h.

**5.1.2.2641 OPT\_AddressBase** [32/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207K8.h.

**5.1.2.2642 OPT\_AddressBase** [33/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6288.h.

**5.1.2.2643 OPT\_AddressBase** [34/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207R6.h.

**5.1.2.2644 OPT\_AddressBase** [35/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S001J3.h.

**5.1.2.2645 OPT\_AddressBase** [36/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF5288.h.

**5.1.2.2646** OPT\_AddressBase [37/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207R8.h.

**5.1.2.2647** OPT\_AddressBase [38/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S003F3.h.

**5.1.2.2648** OPT\_AddressBase [39/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S003K3.h.

**5.1.2.2649** OPT\_AddressBase [40/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207MB.h.

**5.1.2.2650** OPT\_AddressBase [41/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S005C6.h.

**5.1.2.2651** OPT\_AddressBase [42/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF528A.h.

**5.1.2.2652** **OPT\_AddressBase** [43/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S005K6.h.

**5.1.2.2653** **OPT\_AddressBase** [44/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208S6.h.

**5.1.2.2654** **OPT\_AddressBase** [45/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S007C8.h.

**5.1.2.2655** **OPT\_AddressBase** [46/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S103F2.h.

**5.1.2.2656** **OPT\_AddressBase** [47/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S103F3.h.

**5.1.2.2657** **OPT\_AddressBase** [48/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S103K3.h.

**5.1.2.2658** OPT\_AddressBase [49/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S105C4.h.

**5.1.2.2659** OPT\_AddressBase [50/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6213A.h.

**5.1.2.2660** OPT\_AddressBase [51/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S105K4.h.

**5.1.2.2661** OPT\_AddressBase [52/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF62A6.h.

**5.1.2.2662** OPT\_AddressBase [53/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S105S4.h.

**5.1.2.2663** OPT\_AddressBase [54/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF62A8.h.

**5.1.2.2664 OPT\_AddressBase** [55/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6223.h.

**5.1.2.2665 OPT\_AddressBase** [56/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF52A6.h.

**5.1.2.2666 OPT\_AddressBase** [57/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF62A9.h.

**5.1.2.2667 OPT\_AddressBase** [58/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S105S6.h.

**5.1.2.2668 OPT\_AddressBase** [59/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF62AA.h.

**5.1.2.2669 OPT\_AddressBase** [60/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S207C6.h.



**5.1.2.2670 OPT\_AddressBase** [61/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6366.h.

**5.1.2.2671 OPT\_AddressBase** [62/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208C8.h.

**5.1.2.2672 OPT\_AddressBase** [63/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S105K6.h.

**5.1.2.2673 OPT\_AddressBase** [64/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6226.h.

**5.1.2.2674 OPT\_AddressBase** [65/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208CB.h.

**5.1.2.2675 OPT\_AddressBase** [66/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208MB.h.

**5.1.2.2676 OPT\_AddressBase** [67/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF6388.h.

**5.1.2.2677 OPT\_AddressBase** [68/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S105C6.h.

**5.1.2.2678 OPT\_AddressBase** [69/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8S208RB.h.

**5.1.2.2679 OPT\_AddressBase** [70/70]

```
#define OPT_AddressBase 0x4800
```

Definition at line 52 of file STM8AF628A.h.

**5.1.2.2680 PORTA\_AddressBase** [1/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208S8.h.

**5.1.2.2681 PORTA\_AddressBase** [2/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S103K3.h.

**5.1.2.2682** PORTA\_AddressBase [3/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF52A8.h.

**5.1.2.2683** PORTA\_AddressBase [4/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6246.h.

**5.1.2.2684** PORTA\_AddressBase [5/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6213.h.

**5.1.2.2685** PORTA\_AddressBase [6/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207RB.h.

**5.1.2.2686** PORTA\_AddressBase [7/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF528A.h.

**5.1.2.2687** PORTA\_AddressBase [8/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6289.h.

**5.1.2.2688** **PORTA\_AddressBase** [9/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6248.h.

**5.1.2.2689** **PORTA\_AddressBase** [10/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6223A.h.

**5.1.2.2690** **PORTA\_AddressBase** [11/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF5288.h.

**5.1.2.2691** **PORTA\_AddressBase** [12/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6266.h.

**5.1.2.2692** **PORTA\_AddressBase** [13/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF52A9.h.

**5.1.2.2693** **PORTA\_AddressBase** [14/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6223.h.

**5.1.2.2694** PORTA\_AddressBase [15/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF5289.h.

**5.1.2.2695** PORTA\_AddressBase [16/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S903F3.h.

**5.1.2.2696** PORTA\_AddressBase [17/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6268.h.

**5.1.2.2697** PORTA\_AddressBase [18/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S903K3.h.

**5.1.2.2698** PORTA\_AddressBase [19/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208SB.h.

**5.1.2.2699** PORTA\_AddressBase [20/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF5286.h.

**5.1.2.2700** **PORTA\_AddressBase** [21/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208R8.h.

**5.1.2.2701** **PORTA\_AddressBase** [22/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6269.h.

**5.1.2.2702** **PORTA\_AddressBase** [23/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6226.h.

**5.1.2.2703** **PORTA\_AddressBase** [24/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF52AA.h.

**5.1.2.2704** **PORTA\_AddressBase** [25/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208RB.h.

**5.1.2.2705** **PORTA\_AddressBase** [26/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208S6.h.

**5.1.2.2706** PORTA\_AddressBase [27/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208MB.h.

**5.1.2.2707** PORTA\_AddressBase [28/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6366.h.

**5.1.2.2708** PORTA\_AddressBase [29/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207C8.h.

**5.1.2.2709** PORTA\_AddressBase [30/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6286.h.

**5.1.2.2710** PORTA\_AddressBase [31/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF5269.h.

**5.1.2.2711** PORTA\_AddressBase [32/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207CB.h.

**5.1.2.2712** PORTA\_AddressBase [33/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207M8.h.

**5.1.2.2713** PORTA\_AddressBase [34/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6288.h.

**5.1.2.2714** PORTA\_AddressBase [35/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S001J3.h.

**5.1.2.2715** PORTA\_AddressBase [36/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S003F3.h.

**5.1.2.2716** PORTA\_AddressBase [37/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S003K3.h.

**5.1.2.2717** PORTA\_AddressBase [38/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S005C6.h.



**5.1.2.2718** PORTA\_AddressBase [39/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S005K6.h.

**5.1.2.2719** PORTA\_AddressBase [40/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S007C8.h.

**5.1.2.2720** PORTA\_AddressBase [41/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S103F3.h.

**5.1.2.2721** PORTA\_AddressBase [42/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S105C4.h.

**5.1.2.2722** PORTA\_AddressBase [43/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6213A.h.

**5.1.2.2723** PORTA\_AddressBase [44/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S105K4.h.

**5.1.2.2724** **PORTA\_AddressBase** [45/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208CB.h.

**5.1.2.2725** **PORTA\_AddressBase** [46/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207C6.h.

**5.1.2.2726** **PORTA\_AddressBase** [47/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF5268.h.

**5.1.2.2727** **PORTA\_AddressBase** [48/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF62A8.h.

**5.1.2.2728** **PORTA\_AddressBase** [49/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207K6.h.

**5.1.2.2729** **PORTA\_AddressBase** [50/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207K8.h.

**5.1.2.2730** PORTA\_AddressBase [51/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S105S4.h.

**5.1.2.2731** PORTA\_AddressBase [52/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S105S6.h.

**5.1.2.2732** PORTA\_AddressBase [53/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF62A9.h.

**5.1.2.2733** PORTA\_AddressBase [54/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207MB.h.

**5.1.2.2734** PORTA\_AddressBase [55/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF52A6.h.

**5.1.2.2735** PORTA\_AddressBase [56/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF62A6.h.

**5.1.2.2736** **PORTA\_AddressBase** [57/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207R6.h.

**5.1.2.2737** **PORTA\_AddressBase** [58/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207R8.h.

**5.1.2.2738** **PORTA\_AddressBase** [59/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S105K6.h.

**5.1.2.2739** **PORTA\_AddressBase** [60/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF62AA.h.

**5.1.2.2740** **PORTA\_AddressBase** [61/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207S6.h.

**5.1.2.2741** **PORTA\_AddressBase** [62/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207S8.h.

**5.1.2.2742** PORTA\_AddressBase [63/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S207SB.h.

**5.1.2.2743** PORTA\_AddressBase [64/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208C6.h.

**5.1.2.2744** PORTA\_AddressBase [65/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208C8.h.

**5.1.2.2745** PORTA\_AddressBase [66/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF6388.h.

**5.1.2.2746** PORTA\_AddressBase [67/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S208R6.h.

**5.1.2.2747** PORTA\_AddressBase [68/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S103F2.h.

**5.1.2.2748** **PORTA\_AddressBase** [69/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8S105C6.h.

**5.1.2.2749** **PORTA\_AddressBase** [70/70]

```
#define PORTA_AddressBase 0x5000
```

Definition at line 53 of file STM8AF628A.h.

**5.1.2.2750** **PORTB\_AddressBase** [1/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S103F2.h.

**5.1.2.2751** **PORTB\_AddressBase** [2/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S005C6.h.

**5.1.2.2752** **PORTB\_AddressBase** [3/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S005K6.h.

**5.1.2.2753** **PORTB\_AddressBase** [4/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF5268.h.

**5.1.2.2754** PORTB\_AddressBase [5/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6289.h.

**5.1.2.2755** PORTB\_AddressBase [6/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S903K3.h.

**5.1.2.2756** PORTB\_AddressBase [7/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207RB.h.

**5.1.2.2757** PORTB\_AddressBase [8/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6213.h.

**5.1.2.2758** PORTB\_AddressBase [9/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S007C8.h.

**5.1.2.2759** PORTB\_AddressBase [10/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6248.h.

**5.1.2.2760** **PORTB\_AddressBase** [11/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S001J3.h.

**5.1.2.2761** **PORTB\_AddressBase** [12/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6266.h.

**5.1.2.2762** **PORTB\_AddressBase** [13/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S003F3.h.

**5.1.2.2763** **PORTB\_AddressBase** [14/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF52A9.h.

**5.1.2.2764** **PORTB\_AddressBase** [15/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207S8.h.

**5.1.2.2765** **PORTB\_AddressBase** [16/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208SB.h.



**5.1.2.2766** PORTB\_AddressBase [17/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S903F3.h.

**5.1.2.2767** PORTB\_AddressBase [18/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207SB.h.

**5.1.2.2768** PORTB\_AddressBase [19/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF52A8.h.

**5.1.2.2769** PORTB\_AddressBase [20/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF5289.h.

**5.1.2.2770** PORTB\_AddressBase [21/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6268.h.

**5.1.2.2771** PORTB\_AddressBase [22/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6246.h.

**5.1.2.2772** PORTB\_AddressBase [23/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208C6.h.

**5.1.2.2773** PORTB\_AddressBase [24/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208R8.h.

**5.1.2.2774** PORTB\_AddressBase [25/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF5286.h.

**5.1.2.2775** PORTB\_AddressBase [26/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6269.h.

**5.1.2.2776** PORTB\_AddressBase [27/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6223.h.

**5.1.2.2777** PORTB\_AddressBase [28/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208CB.h.

**5.1.2.2778** PORTB\_AddressBase [29/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208S8.h.

**5.1.2.2779** PORTB\_AddressBase [30/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF52AA.h.

**5.1.2.2780** PORTB\_AddressBase [31/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6286.h.

**5.1.2.2781** PORTB\_AddressBase [32/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208R6.h.

**5.1.2.2782** PORTB\_AddressBase [33/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6366.h.

**5.1.2.2783** PORTB\_AddressBase [34/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207C8.h.

**5.1.2.2784** PORTB\_AddressBase [35/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207CB.h.

**5.1.2.2785** PORTB\_AddressBase [36/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207K6.h.

**5.1.2.2786** PORTB\_AddressBase [37/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207M8.h.

**5.1.2.2787** PORTB\_AddressBase [38/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207K8.h.

**5.1.2.2788** PORTB\_AddressBase [39/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207R6.h.

**5.1.2.2789** PORTB\_AddressBase [40/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF5269.h.

**5.1.2.2790** PORTB\_AddressBase [41/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6288.h.

**5.1.2.2791** PORTB\_AddressBase [42/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF52A6.h.

**5.1.2.2792** PORTB\_AddressBase [43/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207R8.h.

**5.1.2.2793** PORTB\_AddressBase [44/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S003K3.h.

**5.1.2.2794** PORTB\_AddressBase [45/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207MB.h.

**5.1.2.2795** PORTB\_AddressBase [46/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF528A.h.

**5.1.2.2796 PORTB\_AddressBase** [47/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S103K3.h.

**5.1.2.2797 PORTB\_AddressBase** [48/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF628A.h.

**5.1.2.2798 PORTB\_AddressBase** [49/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6226.h.

**5.1.2.2799 PORTB\_AddressBase** [50/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208RB.h.

**5.1.2.2800 PORTB\_AddressBase** [51/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S105C6.h.

**5.1.2.2801 PORTB\_AddressBase** [52/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6213A.h.

**5.1.2.2802** PORTB\_AddressBase [53/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF62A6.h.

**5.1.2.2803** PORTB\_AddressBase [54/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S105S6.h.

**5.1.2.2804** PORTB\_AddressBase [55/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF62A8.h.

**5.1.2.2805** PORTB\_AddressBase [56/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF62A9.h.

**5.1.2.2806** PORTB\_AddressBase [57/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S105S4.h.

**5.1.2.2807** PORTB\_AddressBase [58/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6223A.h.

**5.1.2.2808 PORTB\_AddressBase** [59/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF62AA.h.

**5.1.2.2809 PORTB\_AddressBase** [60/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207S6.h.

**5.1.2.2810 PORTB\_AddressBase** [61/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF5288.h.

**5.1.2.2811 PORTB\_AddressBase** [62/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S105K4.h.

**5.1.2.2812 PORTB\_AddressBase** [63/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S105K6.h.

**5.1.2.2813 PORTB\_AddressBase** [64/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S207C6.h.



**5.1.2.2814** PORTB\_AddressBase [65/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208C8.h.

**5.1.2.2815** PORTB\_AddressBase [66/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208MB.h.

**5.1.2.2816** PORTB\_AddressBase [67/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8AF6388.h.

**5.1.2.2817** PORTB\_AddressBase [68/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S105C4.h.

**5.1.2.2818** PORTB\_AddressBase [69/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S208S6.h.

**5.1.2.2819** PORTB\_AddressBase [70/70]

```
#define PORTB_AddressBase 0x5005
```

Definition at line 54 of file STM8S103F3.h.

**5.1.2.2820 PORTC\_AddressBase** [1/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S103F2.h.

**5.1.2.2821 PORTC\_AddressBase** [2/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S103K3.h.

**5.1.2.2822 PORTC\_AddressBase** [3/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF5269.h.

**5.1.2.2823 PORTC\_AddressBase** [4/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S903F3.h.

**5.1.2.2824 PORTC\_AddressBase** [5/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6213.h.

**5.1.2.2825 PORTC\_AddressBase** [6/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6223A.h.

**5.1.2.2826** PORTC\_AddressBase [7/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF5268.h.

**5.1.2.2827** PORTC\_AddressBase [8/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207RB.h.

**5.1.2.2828** PORTC\_AddressBase [9/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6248.h.

**5.1.2.2829** PORTC\_AddressBase [10/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6288.h.

**5.1.2.2830** PORTC\_AddressBase [11/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S001J3.h.

**5.1.2.2831** PORTC\_AddressBase [12/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207K6.h.

**5.1.2.2832 PORTC\_AddressBase** [13/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208S8.h.

**5.1.2.2833 PORTC\_AddressBase** [14/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF62AA.h.

**5.1.2.2834 PORTC\_AddressBase** [15/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6266.h.

**5.1.2.2835 PORTC\_AddressBase** [16/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207MB.h.

**5.1.2.2836 PORTC\_AddressBase** [17/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF62A8.h.

**5.1.2.2837 PORTC\_AddressBase** [18/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF52A9.h.

**5.1.2.2838 PORTC\_AddressBase** [19/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF52A8.h.

**5.1.2.2839 PORTC\_AddressBase** [20/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S903K3.h.

**5.1.2.2840 PORTC\_AddressBase** [21/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208SB.h.

**5.1.2.2841 PORTC\_AddressBase** [22/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF5289.h.

**5.1.2.2842 PORTC\_AddressBase** [23/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6268.h.

**5.1.2.2843 PORTC\_AddressBase** [24/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF5288.h.

**5.1.2.2844 PORTC\_AddressBase** [25/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208R8.h.

**5.1.2.2845 PORTC\_AddressBase** [26/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208MB.h.

**5.1.2.2846 PORTC\_AddressBase** [27/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6246.h.

**5.1.2.2847 PORTC\_AddressBase** [28/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208S6.h.

**5.1.2.2848 PORTC\_AddressBase** [29/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208C8.h.

**5.1.2.2849 PORTC\_AddressBase** [30/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6269.h.

**5.1.2.2850 PORTC\_AddressBase** [31/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6223.h.

**5.1.2.2851 PORTC\_AddressBase** [32/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6388.h.

**5.1.2.2852 PORTC\_AddressBase** [33/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF52AA.h.

**5.1.2.2853 PORTC\_AddressBase** [34/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207S8.h.

**5.1.2.2854 PORTC\_AddressBase** [35/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6226.h.

**5.1.2.2855 PORTC\_AddressBase** [36/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207C8.h.

**5.1.2.2856 PORTC\_AddressBase** [37/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6286.h.

**5.1.2.2857 PORTC\_AddressBase** [38/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208R6.h.

**5.1.2.2858 PORTC\_AddressBase** [39/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207CB.h.

**5.1.2.2859 PORTC\_AddressBase** [40/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207M8.h.

**5.1.2.2860 PORTC\_AddressBase** [41/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207K8.h.

**5.1.2.2861 PORTC\_AddressBase** [42/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207R6.h.



**5.1.2.2862 PORTC\_AddressBase** [43/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF52A6.h.

**5.1.2.2863 PORTC\_AddressBase** [44/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S003F3.h.

**5.1.2.2864 PORTC\_AddressBase** [45/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S003K3.h.

**5.1.2.2865 PORTC\_AddressBase** [46/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S005C6.h.

**5.1.2.2866 PORTC\_AddressBase** [47/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S005K6.h.

**5.1.2.2867 PORTC\_AddressBase** [48/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6289.h.

**5.1.2.2868 PORTC\_AddressBase** [49/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S007C8.h.

**5.1.2.2869 PORTC\_AddressBase** [50/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S105C4.h.

**5.1.2.2870 PORTC\_AddressBase** [51/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF628A.h.

**5.1.2.2871 PORTC\_AddressBase** [52/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S105S4.h.

**5.1.2.2872 PORTC\_AddressBase** [53/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF5286.h.

**5.1.2.2873 PORTC\_AddressBase** [54/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208C6.h.

**5.1.2.2874 PORTC\_AddressBase** [55/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208RB.h.

**5.1.2.2875 PORTC\_AddressBase** [56/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S105S6.h.

**5.1.2.2876 PORTC\_AddressBase** [57/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF62A9.h.

**5.1.2.2877 PORTC\_AddressBase** [58/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S105C6.h.

**5.1.2.2878 PORTC\_AddressBase** [59/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207R8.h.

**5.1.2.2879 PORTC\_AddressBase** [60/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF62A6.h.

**5.1.2.2880 PORTC\_AddressBase** [61/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207S6.h.

**5.1.2.2881 PORTC\_AddressBase** [62/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S105K4.h.

**5.1.2.2882 PORTC\_AddressBase** [63/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207SB.h.

**5.1.2.2883 PORTC\_AddressBase** [64/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6213A.h.

**5.1.2.2884 PORTC\_AddressBase** [65/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S105K6.h.

**5.1.2.2885 PORTC\_AddressBase** [66/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF6366.h.

**5.1.2.2886 PORTC\_AddressBase** [67/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S208CB.h.

**5.1.2.2887 PORTC\_AddressBase** [68/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S207C6.h.

**5.1.2.2888 PORTC\_AddressBase** [69/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8S103F3.h.

**5.1.2.2889 PORTC\_AddressBase** [70/70]

```
#define PORTC_AddressBase 0x500A
```

Definition at line 55 of file STM8AF528A.h.

**5.1.2.2890 PORTD\_AddressBase** [1/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S103K3.h.

**5.1.2.2891 PORTD\_AddressBase** [2/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF52A8.h.

**5.1.2.2892** PORTD\_AddressBase [3/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S903K3.h.

**5.1.2.2893** PORTD\_AddressBase [4/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S005K6.h.

**5.1.2.2894** PORTD\_AddressBase [5/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF5269.h.

**5.1.2.2895** PORTD\_AddressBase [6/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S007C8.h.

**5.1.2.2896** PORTD\_AddressBase [7/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S005C6.h.

**5.1.2.2897** PORTD\_AddressBase [8/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6289.h.

**5.1.2.2898 PORTD\_AddressBase** [9/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6213.h.

**5.1.2.2899 PORTD\_AddressBase** [10/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207RB.h.

**5.1.2.2900 PORTD\_AddressBase** [11/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S003K3.h.

**5.1.2.2901 PORTD\_AddressBase** [12/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6248.h.

**5.1.2.2902 PORTD\_AddressBase** [13/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF5268.h.

**5.1.2.2903 PORTD\_AddressBase** [14/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S001J3.h.

**5.1.2.2904 PORTD\_AddressBase** [15/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF62A9.h.

**5.1.2.2905 PORTD\_AddressBase** [16/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207S6.h.

**5.1.2.2906 PORTD\_AddressBase** [17/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S003F3.h.

**5.1.2.2907 PORTD\_AddressBase** [18/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6223A.h.

**5.1.2.2908 PORTD\_AddressBase** [19/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6266.h.

**5.1.2.2909 PORTD\_AddressBase** [20/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6226.h.



**5.1.2.2910 PORTD\_AddressBase** [21/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF62A8.h.

**5.1.2.2911 PORTD\_AddressBase** [22/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF52A9.h.

**5.1.2.2912 PORTD\_AddressBase** [23/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S903F3.h.

**5.1.2.2913 PORTD\_AddressBase** [24/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6246.h.

**5.1.2.2914 PORTD\_AddressBase** [25/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208SB.h.

**5.1.2.2915 PORTD\_AddressBase** [26/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207SB.h.

**5.1.2.2916 PORTD\_AddressBase** [27/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF5289.h.

**5.1.2.2917 PORTD\_AddressBase** [28/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6268.h.

**5.1.2.2918 PORTD\_AddressBase** [29/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208C6.h.

**5.1.2.2919 PORTD\_AddressBase** [30/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208CB.h.

**5.1.2.2920 PORTD\_AddressBase** [31/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208R8.h.

**5.1.2.2921 PORTD\_AddressBase** [32/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF5288.h.

**5.1.2.2922** PORTD\_AddressBase [33/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6366.h.

**5.1.2.2923** PORTD\_AddressBase [34/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF52AA.h.

**5.1.2.2924** PORTD\_AddressBase [35/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208R6.h.

**5.1.2.2925** PORTD\_AddressBase [36/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6286.h.

**5.1.2.2926** PORTD\_AddressBase [37/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207CB.h.

**5.1.2.2927** PORTD\_AddressBase [38/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207C8.h.

**5.1.2.2928** PORTD\_AddressBase [39/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207K6.h.

**5.1.2.2929** PORTD\_AddressBase [40/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6223.h.

**5.1.2.2930** PORTD\_AddressBase [41/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207M8.h.

**5.1.2.2931** PORTD\_AddressBase [42/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207R6.h.

**5.1.2.2932** PORTD\_AddressBase [43/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207K8.h.

**5.1.2.2933** PORTD\_AddressBase [44/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6288.h.

**5.1.2.2934** PORTD\_AddressBase [45/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207MB.h.

**5.1.2.2935** PORTD\_AddressBase [46/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207S8.h.

**5.1.2.2936** PORTD\_AddressBase [47/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208RB.h.

**5.1.2.2937** PORTD\_AddressBase [48/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207R8.h.

**5.1.2.2938** PORTD\_AddressBase [49/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S103F2.h.

**5.1.2.2939** PORTD\_AddressBase [50/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF5286.h.

**5.1.2.2940** PORTD\_AddressBase [51/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF628A.h.

**5.1.2.2941** PORTD\_AddressBase [52/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208MB.h.

**5.1.2.2942** PORTD\_AddressBase [53/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6269.h.

**5.1.2.2943** PORTD\_AddressBase [54/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6388.h.

**5.1.2.2944** PORTD\_AddressBase [55/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S105K6.h.

**5.1.2.2945** PORTD\_AddressBase [56/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208S6.h.

**5.1.2.2946** PORTD\_AddressBase [57/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208S8.h.

**5.1.2.2947** PORTD\_AddressBase [58/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF528A.h.

**5.1.2.2948** PORTD\_AddressBase [59/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S105C4.h.

**5.1.2.2949** PORTD\_AddressBase [60/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF62AA.h.

**5.1.2.2950** PORTD\_AddressBase [61/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S105C6.h.

**5.1.2.2951** PORTD\_AddressBase [62/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF52A6.h.

**5.1.2.2952 PORTD\_AddressBase** [63/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF62A6.h.

**5.1.2.2953 PORTD\_AddressBase** [64/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S105S4.h.

**5.1.2.2954 PORTD\_AddressBase** [65/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S105S6.h.

**5.1.2.2955 PORTD\_AddressBase** [66/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S105K4.h.

**5.1.2.2956 PORTD\_AddressBase** [67/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S207C6.h.

**5.1.2.2957 PORTD\_AddressBase** [68/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S208C8.h.



**5.1.2.2958 PORTD\_AddressBase** [69/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8AF6213A.h.

**5.1.2.2959 PORTD\_AddressBase** [70/70]

```
#define PORTD_AddressBase 0x500F
```

Definition at line 56 of file STM8S103F3.h.

**5.1.2.2960 PORTE\_AddressBase** [1/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF528A.h.

**5.1.2.2961 PORTE\_AddressBase** [2/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S005K6.h.

**5.1.2.2962 PORTE\_AddressBase** [3/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF52A8.h.

**5.1.2.2963 PORTE\_AddressBase** [4/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S903K3.h.

**5.1.2.2964** **PORTE\_AddressBase** [5/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF5269.h.

**5.1.2.2965** **PORTE\_AddressBase** [6/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6213.h.

**5.1.2.2966** **PORTE\_AddressBase** [7/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S005C6.h.

**5.1.2.2967** **PORTE\_AddressBase** [8/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S007C8.h.

**5.1.2.2968** **PORTE\_AddressBase** [9/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6248.h.

**5.1.2.2969** **PORTE\_AddressBase** [10/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6288.h.

**5.1.2.2970** **PORTE\_AddressBase** [11/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF62AA.h.

**5.1.2.2971** **PORTE\_AddressBase** [12/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S001J3.h.

**5.1.2.2972** **PORTE\_AddressBase** [13/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6266.h.

**5.1.2.2973** **PORTE\_AddressBase** [14/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF52A9.h.

**5.1.2.2974** **PORTE\_AddressBase** [15/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208C8.h.

**5.1.2.2975** **PORTE\_AddressBase** [16/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S903F3.h.

**5.1.2.2976** `PORTE_AddressBase` [17/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208S8.h.

**5.1.2.2977** `PORTE_AddressBase` [18/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6268.h.

**5.1.2.2978** `PORTE_AddressBase` [19/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208SB.h.

**5.1.2.2979** `PORTE_AddressBase` [20/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF5289.h.

**5.1.2.2980** `PORTE_AddressBase` [21/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208CB.h.

**5.1.2.2981** `PORTE_AddressBase` [22/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF5288.h.

**5.1.2.2982** **PORTE\_AddressBase** [23/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6246.h.

**5.1.2.2983** **PORTE\_AddressBase** [24/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6269.h.

**5.1.2.2984** **PORTE\_AddressBase** [25/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208RB.h.

**5.1.2.2985** **PORTE\_AddressBase** [26/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208MB.h.

**5.1.2.2986** **PORTE\_AddressBase** [27/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6366.h.

**5.1.2.2987** **PORTE\_AddressBase** [28/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF5286.h.

**5.1.2.2988** **PORTE\_AddressBase** [29/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF62A8.h.

**5.1.2.2989** **PORTE\_AddressBase** [30/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF52AA.h.

**5.1.2.2990** **PORTE\_AddressBase** [31/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6286.h.

**5.1.2.2991** **PORTE\_AddressBase** [32/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207CB.h.

**5.1.2.2992** **PORTE\_AddressBase** [33/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6226.h.

**5.1.2.2993** **PORTE\_AddressBase** [34/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6388.h.

**5.1.2.2994**   **PORTE\_AddressBase** [35/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207K6.h.

**5.1.2.2995**   **PORTE\_AddressBase** [36/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF5268.h.

**5.1.2.2996**   **PORTE\_AddressBase** [37/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S003F3.h.

**5.1.2.2997**   **PORTE\_AddressBase** [38/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF62A9.h.

**5.1.2.2998**   **PORTE\_AddressBase** [39/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S003K3.h.

**5.1.2.2999**   **PORTE\_AddressBase** [40/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208R6.h.

**5.1.2.3000** **PORTE\_AddressBase** [41/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6289.h.

**5.1.2.3001** **PORTE\_AddressBase** [42/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208S6.h.

**5.1.2.3002** **PORTE\_AddressBase** [43/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S103F3.h.

**5.1.2.3003** **PORTE\_AddressBase** [44/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207S6.h.

**5.1.2.3004** **PORTE\_AddressBase** [45/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S105K4.h.

**5.1.2.3005** **PORTE\_AddressBase** [46/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207C6.h.



**5.1.2.3006** **PORTE\_AddressBase** [47/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207C8.h.

**5.1.2.3007** **PORTE\_AddressBase** [48/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208R8.h.

**5.1.2.3008** **PORTE\_AddressBase** [49/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6223.h.

**5.1.2.3009** **PORTE\_AddressBase** [50/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207K8.h.

**5.1.2.3010** **PORTE\_AddressBase** [51/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207M8.h.

**5.1.2.3011** **PORTE\_AddressBase** [52/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S105S6.h.

**5.1.2.3012** **PORTE\_AddressBase** [53/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207MB.h.

**5.1.2.3013** **PORTE\_AddressBase** [54/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207R6.h.

**5.1.2.3014** **PORTE\_AddressBase** [55/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF62A6.h.

**5.1.2.3015** **PORTE\_AddressBase** [56/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S105C4.h.

**5.1.2.3016** **PORTE\_AddressBase** [57/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207R8.h.

**5.1.2.3017** **PORTE\_AddressBase** [58/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF52A6.h.

**5.1.2.3018**   **PORTE\_AddressBase** [59/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S105S4.h.

**5.1.2.3019**   **PORTE\_AddressBase** [60/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207RB.h.

**5.1.2.3020**   **PORTE\_AddressBase** [61/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6223A.h.

**5.1.2.3021**   **PORTE\_AddressBase** [62/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF628A.h.

**5.1.2.3022**   **PORTE\_AddressBase** [63/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207S8.h.

**5.1.2.3023**   **PORTE\_AddressBase** [64/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S207SB.h.

**5.1.2.3024** **PORTE\_AddressBase** [65/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8AF6213A.h.

**5.1.2.3025** **PORTE\_AddressBase** [66/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S208C6.h.

**5.1.2.3026** **PORTE\_AddressBase** [67/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S105K6.h.

**5.1.2.3027** **PORTE\_AddressBase** [68/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S103K3.h.

**5.1.2.3028** **PORTE\_AddressBase** [69/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S105C6.h.

**5.1.2.3029** **PORTE\_AddressBase** [70/70]

```
#define PORTE_AddressBase 0x5014
```

Definition at line 57 of file STM8S103F2.h.

**5.1.2.3030 PORTF\_AddressBase** [1/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6266.h.

**5.1.2.3031 PORTF\_AddressBase** [2/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6213.h.

**5.1.2.3032 PORTF\_AddressBase** [3/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S903K3.h.

**5.1.2.3033 PORTF\_AddressBase** [4/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF52A8.h.

**5.1.2.3034 PORTF\_AddressBase** [5/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S007C8.h.

**5.1.2.3035 PORTF\_AddressBase** [6/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S005K6.h.

**5.1.2.3036** PORTF\_AddressBase [7/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207RB.h.

**5.1.2.3037** PORTF\_AddressBase [8/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S005C6.h.

**5.1.2.3038** PORTF\_AddressBase [9/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6289.h.

**5.1.2.3039** PORTF\_AddressBase [10/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207R8.h.

**5.1.2.3040** PORTF\_AddressBase [11/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6248.h.

**5.1.2.3041** PORTF\_AddressBase [12/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S001J3.h.

**5.1.2.3042** PORTF\_AddressBase [13/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6223A.h.

**5.1.2.3043** PORTF\_AddressBase [14/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF62AA.h.

**5.1.2.3044** PORTF\_AddressBase [15/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S003F3.h.

**5.1.2.3045** PORTF\_AddressBase [16/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6288.h.

**5.1.2.3046** PORTF\_AddressBase [17/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207S6.h.

**5.1.2.3047** PORTF\_AddressBase [18/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF5268.h.

**5.1.2.3048 PORTF\_AddressBase** [19/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207K6.h.

**5.1.2.3049 PORTF\_AddressBase** [20/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207MB.h.

**5.1.2.3050 PORTF\_AddressBase** [21/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S903F3.h.

**5.1.2.3051 PORTF\_AddressBase** [22/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF52A9.h.

**5.1.2.3052 PORTF\_AddressBase** [23/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208C6.h.

**5.1.2.3053 PORTF\_AddressBase** [24/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207SB.h.



**5.1.2.3054** PORTF\_AddressBase [25/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208SB.h.

**5.1.2.3055** PORTF\_AddressBase [26/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S103K3.h.

**5.1.2.3056** PORTF\_AddressBase [27/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208R8.h.

**5.1.2.3057** PORTF\_AddressBase [28/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208S8.h.

**5.1.2.3058** PORTF\_AddressBase [29/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6269.h.

**5.1.2.3059** PORTF\_AddressBase [30/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208C8.h.

**5.1.2.3060 PORTF\_AddressBase** [31/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208RB.h.

**5.1.2.3061 PORTF\_AddressBase** [32/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF5288.h.

**5.1.2.3062 PORTF\_AddressBase** [33/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6366.h.

**5.1.2.3063 PORTF\_AddressBase** [34/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208CB.h.

**5.1.2.3064 PORTF\_AddressBase** [35/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF52AA.h.

**5.1.2.3065 PORTF\_AddressBase** [36/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF5286.h.

**5.1.2.3066** PORTF\_AddressBase [37/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208R6.h.

**5.1.2.3067** PORTF\_AddressBase [38/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207C8.h.

**5.1.2.3068** PORTF\_AddressBase [39/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207CB.h.

**5.1.2.3069** PORTF\_AddressBase [40/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207R6.h.

**5.1.2.3070** PORTF\_AddressBase [41/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF62A9.h.

**5.1.2.3071** PORTF\_AddressBase [42/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207M8.h.

**5.1.2.3072 PORTF\_AddressBase** [43/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6286.h.

**5.1.2.3073 PORTF\_AddressBase** [44/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207K8.h.

**5.1.2.3074 PORTF\_AddressBase** [45/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S003K3.h.

**5.1.2.3075 PORTF\_AddressBase** [46/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF528A.h.

**5.1.2.3076 PORTF\_AddressBase** [47/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF62A6.h.

**5.1.2.3077 PORTF\_AddressBase** [48/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S105S6.h.

**5.1.2.3078** PORTF\_AddressBase [49/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF5289.h.

**5.1.2.3079** PORTF\_AddressBase [50/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207S8.h.

**5.1.2.3080** PORTF\_AddressBase [51/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF62A8.h.

**5.1.2.3081** PORTF\_AddressBase [52/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6223.h.

**5.1.2.3082** PORTF\_AddressBase [53/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6268.h.

**5.1.2.3083** PORTF\_AddressBase [54/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6246.h.

**5.1.2.3084** PORTF\_AddressBase [55/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S105C4.h.

**5.1.2.3085** PORTF\_AddressBase [56/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S105S4.h.

**5.1.2.3086** PORTF\_AddressBase [57/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF628A.h.

**5.1.2.3087** PORTF\_AddressBase [58/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF52A6.h.

**5.1.2.3088** PORTF\_AddressBase [59/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S105K4.h.

**5.1.2.3089** PORTF\_AddressBase [60/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S105K6.h.

**5.1.2.3090** PORTF\_AddressBase [61/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6213A.h.

**5.1.2.3091** PORTF\_AddressBase [62/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S105C6.h.

**5.1.2.3092** PORTF\_AddressBase [63/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S207C6.h.

**5.1.2.3093** PORTF\_AddressBase [64/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208MB.h.

**5.1.2.3094** PORTF\_AddressBase [65/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6226.h.

**5.1.2.3095** PORTF\_AddressBase [66/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF6388.h.

**5.1.2.3096 PORTF\_AddressBase** [67/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S103F3.h.

**5.1.2.3097 PORTF\_AddressBase** [68/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8AF5269.h.

**5.1.2.3098 PORTF\_AddressBase** [69/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S103F2.h.

**5.1.2.3099 PORTF\_AddressBase** [70/70]

```
#define PORTF_AddressBase 0x5019
```

Definition at line 58 of file STM8S208S6.h.

**5.1.2.3100 PORTG\_AddressBase** [1/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207CB.h.

**5.1.2.3101 PORTG\_AddressBase** [2/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S005K6.h.



**5.1.2.3102 PORTG\_AddressBase** [3/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207RB.h.

**5.1.2.3103 PORTG\_AddressBase** [4/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S005C6.h.

**5.1.2.3104 PORTG\_AddressBase** [5/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207R8.h.

**5.1.2.3105 PORTG\_AddressBase** [6/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6248.h.

**5.1.2.3106 PORTG\_AddressBase** [7/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6289.h.

**5.1.2.3107 PORTG\_AddressBase** [8/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207K6.h.

**5.1.2.3108 PORTG\_AddressBase** [9/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF62AA.h.

**5.1.2.3109 PORTG\_AddressBase** [10/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207S6.h.

**5.1.2.3110 PORTG\_AddressBase** [11/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207S8.h.

**5.1.2.3111 PORTG\_AddressBase** [12/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208CB.h.

**5.1.2.3112 PORTG\_AddressBase** [13/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF528A.h.

**5.1.2.3113 PORTG\_AddressBase** [14/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6266.h.

**5.1.2.3114 PORTG\_AddressBase** [15/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF5286.h.

**5.1.2.3115 PORTG\_AddressBase** [16/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6268.h.

**5.1.2.3116 PORTG\_AddressBase** [17/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF5289.h.

**5.1.2.3117 PORTG\_AddressBase** [18/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208R8.h.

**5.1.2.3118 PORTG\_AddressBase** [19/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6246.h.

**5.1.2.3119 PORTG\_AddressBase** [20/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF52A9.h.

**5.1.2.3120 PORTG\_AddressBase** [21/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6269.h.

**5.1.2.3121 PORTG\_AddressBase** [22/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF5288.h.

**5.1.2.3122 PORTG\_AddressBase** [23/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207C8.h.

**5.1.2.3123 PORTG\_AddressBase** [24/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF52AA.h.

**5.1.2.3124 PORTG\_AddressBase** [25/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208R6.h.

**5.1.2.3125 PORTG\_AddressBase** [26/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF62A8.h.

**5.1.2.3126 PORTG\_AddressBase** [27/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6286.h.

**5.1.2.3127 PORTG\_AddressBase** [28/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207MB.h.

**5.1.2.3128 PORTG\_AddressBase** [29/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207R6.h.

**5.1.2.3129 PORTG\_AddressBase** [30/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207K8.h.

**5.1.2.3130 PORTG\_AddressBase** [31/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6288.h.

**5.1.2.3131 PORTG\_AddressBase** [32/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S007C8.h.

**5.1.2.3132 PORTG\_AddressBase** [33/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF6388.h.

**5.1.2.3133 PORTG\_AddressBase** [34/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208MB.h.

**5.1.2.3134 PORTG\_AddressBase** [35/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208S6.h.

**5.1.2.3135 PORTG\_AddressBase** [36/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208C8.h.

**5.1.2.3136 PORTG\_AddressBase** [37/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF62A9.h.

**5.1.2.3137 PORTG\_AddressBase** [38/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF5268.h.

**5.1.2.3138 PORTG\_AddressBase** [39/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208SB.h.

**5.1.2.3139 PORTG\_AddressBase** [40/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207SB.h.

**5.1.2.3140 PORTG\_AddressBase** [41/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF628A.h.

**5.1.2.3141 PORTG\_AddressBase** [42/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207M8.h.

**5.1.2.3142 PORTG\_AddressBase** [43/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF62A6.h.

**5.1.2.3143 PORTG\_AddressBase** [44/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S105S6.h.

**5.1.2.3144 PORTG\_AddressBase** [45/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF5269.h.

**5.1.2.3145 PORTG\_AddressBase** [46/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S105C4.h.

**5.1.2.3146 PORTG\_AddressBase** [47/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF52A6.h.

**5.1.2.3147 PORTG\_AddressBase** [48/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S105S4.h.

**5.1.2.3148 PORTG\_AddressBase** [49/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S207C6.h.

**5.1.2.3149 PORTG\_AddressBase** [50/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208C6.h.



**5.1.2.3150 PORTG\_AddressBase** [51/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S105K6.h.

**5.1.2.3151 PORTG\_AddressBase** [52/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8AF52A8.h.

**5.1.2.3152 PORTG\_AddressBase** [53/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S105K4.h.

**5.1.2.3153 PORTG\_AddressBase** [54/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S105C6.h.

**5.1.2.3154 PORTG\_AddressBase** [55/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208RB.h.

**5.1.2.3155 PORTG\_AddressBase** [56/56]

```
#define PORTG_AddressBase 0x501E
```

Definition at line 59 of file STM8S208S8.h.

**5.1.2.3156** **PORTH\_AddressBase** [1/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208SB.h.

**5.1.2.3157** **PORTH\_AddressBase** [2/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF52A8.h.

**5.1.2.3158** **PORTH\_AddressBase** [3/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF62A9.h.

**5.1.2.3159** **PORTH\_AddressBase** [4/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207R8.h.

**5.1.2.3160** **PORTH\_AddressBase** [5/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207RB.h.

**5.1.2.3161** **PORTH\_AddressBase** [6/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF52A6.h.

**5.1.2.3162** PORTH\_AddressBase [7/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207S6.h.

**5.1.2.3163** PORTH\_AddressBase [8/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207S8.h.

**5.1.2.3164** PORTH\_AddressBase [9/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207K6.h.

**5.1.2.3165** PORTH\_AddressBase [10/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF5286.h.

**5.1.2.3166** PORTH\_AddressBase [11/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF52A9.h.

**5.1.2.3167** PORTH\_AddressBase [12/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF5268.h.

**5.1.2.3168** **PORTH\_AddressBase** [13/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF5289.h.

**5.1.2.3169** **PORTH\_AddressBase** [14/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF6388.h.

**5.1.2.3170** **PORTH\_AddressBase** [15/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207SB.h.

**5.1.2.3171** **PORTH\_AddressBase** [16/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208S6.h.

**5.1.2.3172** **PORTH\_AddressBase** [17/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208MB.h.

**5.1.2.3173** **PORTH\_AddressBase** [18/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207C8.h.

**5.1.2.3174** PORTH\_AddressBase [19/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207MB.h.

**5.1.2.3175** PORTH\_AddressBase [20/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207M8.h.

**5.1.2.3176** PORTH\_AddressBase [21/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207R6.h.

**5.1.2.3177** PORTH\_AddressBase [22/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF628A.h.

**5.1.2.3178** PORTH\_AddressBase [23/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF6288.h.

**5.1.2.3179** PORTH\_AddressBase [24/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208C6.h.

**5.1.2.3180** **PORTH\_AddressBase** [25/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208R6.h.

**5.1.2.3181** **PORTH\_AddressBase** [26/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF5288.h.

**5.1.2.3182** **PORTH\_AddressBase** [27/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF52AA.h.

**5.1.2.3183** **PORTH\_AddressBase** [28/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF6286.h.

**5.1.2.3184** **PORTH\_AddressBase** [29/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF62A8.h.

**5.1.2.3185** **PORTH\_AddressBase** [30/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208RB.h.

**5.1.2.3186** PORTH\_AddressBase [31/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208CB.h.

**5.1.2.3187** PORTH\_AddressBase [32/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208S8.h.

**5.1.2.3188** PORTH\_AddressBase [33/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF6289.h.

**5.1.2.3189** PORTH\_AddressBase [34/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207CB.h.

**5.1.2.3190** PORTH\_AddressBase [35/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF62AA.h.

**5.1.2.3191** PORTH\_AddressBase [36/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207C6.h.

**5.1.2.3192** **PORTH\_AddressBase** [37/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF5269.h.

**5.1.2.3193** **PORTH\_AddressBase** [38/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208C8.h.

**5.1.2.3194** **PORTH\_AddressBase** [39/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF62A6.h.

**5.1.2.3195** **PORTH\_AddressBase** [40/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S207K8.h.

**5.1.2.3196** **PORTH\_AddressBase** [41/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S007C8.h.

**5.1.2.3197** **PORTH\_AddressBase** [42/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8S208R8.h.



**5.1.2.3198** PORTH\_AddressBase [43/43]

```
#define PORTH_AddressBase 0x5023
```

Definition at line 60 of file STM8AF528A.h.

**5.1.2.3199** PORTI\_AddressBase [1/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF6388.h.

**5.1.2.3200** PORTI\_AddressBase [2/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF5288.h.

**5.1.2.3201** PORTI\_AddressBase [3/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF6288.h.

**5.1.2.3202** PORTI\_AddressBase [4/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207R8.h.

**5.1.2.3203** PORTI\_AddressBase [5/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207S6.h.

**5.1.2.3204 PORTI\_AddressBase** [6/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207K6.h.

**5.1.2.3205 PORTI\_AddressBase** [7/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF62AA.h.

**5.1.2.3206 PORTI\_AddressBase** [8/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208R6.h.

**5.1.2.3207 PORTI\_AddressBase** [9/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207CB.h.

**5.1.2.3208 PORTI\_AddressBase** [10/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF52A6.h.

**5.1.2.3209 PORTI\_AddressBase** [11/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207S8.h.

**5.1.2.3210** PORTI\_AddressBase [12/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207MB.h.

**5.1.2.3211** PORTI\_AddressBase [13/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207M8.h.

**5.1.2.3212** PORTI\_AddressBase [14/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207SB.h.

**5.1.2.3213** PORTI\_AddressBase [15/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF5289.h.

**5.1.2.3214** PORTI\_AddressBase [16/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208R8.h.

**5.1.2.3215** PORTI\_AddressBase [17/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF52A9.h.

**5.1.2.3216 PORTI\_AddressBase** [18/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207R6.h.

**5.1.2.3217 PORTI\_AddressBase** [19/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208C8.h.

**5.1.2.3218 PORTI\_AddressBase** [20/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208MB.h.

**5.1.2.3219 PORTI\_AddressBase** [21/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF6286.h.

**5.1.2.3220 PORTI\_AddressBase** [22/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF62A9.h.

**5.1.2.3221 PORTI\_AddressBase** [23/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF52AA.h.

**5.1.2.3222** PORTI\_AddressBase [24/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208S6.h.

**5.1.2.3223** PORTI\_AddressBase [25/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207C8.h.

**5.1.2.3224** PORTI\_AddressBase [26/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF528A.h.

**5.1.2.3225** PORTI\_AddressBase [27/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF628A.h.

**5.1.2.3226** PORTI\_AddressBase [28/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF52A8.h.

**5.1.2.3227** PORTI\_AddressBase [29/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207RB.h.

**5.1.2.3228** **PORTI\_AddressBase** [30/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208RB.h.

**5.1.2.3229** **PORTI\_AddressBase** [31/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF62A6.h.

**5.1.2.3230** **PORTI\_AddressBase** [32/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208CB.h.

**5.1.2.3231** **PORTI\_AddressBase** [33/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208SB.h.

**5.1.2.3232** **PORTI\_AddressBase** [34/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF5269.h.

**5.1.2.3233** **PORTI\_AddressBase** [35/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF62A8.h.

**5.1.2.3234** PORTI\_AddressBase [36/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208S8.h.

**5.1.2.3235** PORTI\_AddressBase [37/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207K8.h.

**5.1.2.3236** PORTI\_AddressBase [38/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S208C6.h.

**5.1.2.3237** PORTI\_AddressBase [39/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S007C8.h.

**5.1.2.3238** PORTI\_AddressBase [40/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF5286.h.

**5.1.2.3239** PORTI\_AddressBase [41/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF6289.h.

**5.1.2.3240** **PORTI\_AddressBase** [42/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8AF5268.h.

**5.1.2.3241** **PORTI\_AddressBase** [43/43]

```
#define PORTI_AddressBase 0x5028
```

Definition at line 61 of file STM8S207C6.h.

**5.1.2.3242** **RST\_AddressBase** [1/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S103K3.h.

**5.1.2.3243** **RST\_AddressBase** [2/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8AF6213.h.

**5.1.2.3244** **RST\_AddressBase** [3/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S001J3.h.

**5.1.2.3245** **RST\_AddressBase** [4/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S003F3.h.



**5.1.2.3246 RST\_AddressBase** [5/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8AF6223A.h.

**5.1.2.3247 RST\_AddressBase** [6/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8AF6366.h.

**5.1.2.3248 RST\_AddressBase** [7/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S903K3.h.

**5.1.2.3249 RST\_AddressBase** [8/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8AF6226.h.

**5.1.2.3250 RST\_AddressBase** [9/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8AF6223.h.

**5.1.2.3251 RST\_AddressBase** [10/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S003K3.h.

**5.1.2.3252 RST\_AddressBase** [11/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S903F3.h.

**5.1.2.3253 RST\_AddressBase** [12/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S103F2.h.

**5.1.2.3254 RST\_AddressBase** [13/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8AF6213A.h.

**5.1.2.3255 RST\_AddressBase** [14/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 61 of file STM8S103F3.h.

**5.1.2.3256 RST\_AddressBase** [15/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8AF6246.h.

**5.1.2.3257 RST\_AddressBase** [16/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8AF6248.h.

**5.1.2.3258 RST\_AddressBase** [17/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8AF6268.h.

**5.1.2.3259 RST\_AddressBase** [18/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S005C6.h.

**5.1.2.3260 RST\_AddressBase** [19/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8AF6269.h.

**5.1.2.3261 RST\_AddressBase** [20/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S105S4.h.

**5.1.2.3262 RST\_AddressBase** [21/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S105S6.h.

**5.1.2.3263 RST\_AddressBase** [22/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8AF6266.h.

**5.1.2.3264 RST\_AddressBase** [23/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S105K4.h.

**5.1.2.3265 RST\_AddressBase** [24/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S105C4.h.

**5.1.2.3266 RST\_AddressBase** [25/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S105C6.h.

**5.1.2.3267 RST\_AddressBase** [26/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S005K6.h.

**5.1.2.3268 RST\_AddressBase** [27/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 62 of file STM8S105K6.h.

**5.1.2.3269 RST\_AddressBase** [28/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF62A8.h.

**5.1.2.3270 RST\_AddressBase** [29/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207R8.h.

**5.1.2.3271 RST\_AddressBase** [30/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207MB.h.

**5.1.2.3272 RST\_AddressBase** [31/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207S6.h.

**5.1.2.3273 RST\_AddressBase** [32/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF6288.h.

**5.1.2.3274 RST\_AddressBase** [33/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207S8.h.

**5.1.2.3275 RST\_AddressBase** [34/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF62AA.h.

**5.1.2.3276 RST\_AddressBase** [35/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF52AA.h.

**5.1.2.3277 RST\_AddressBase** [36/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF5269.h.

**5.1.2.3278 RST\_AddressBase** [37/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208SB.h.

**5.1.2.3279 RST\_AddressBase** [38/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF52A9.h.

**5.1.2.3280 RST\_AddressBase** [39/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207SB.h.

**5.1.2.3281 RST\_AddressBase** [40/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207R6.h.

**5.1.2.3282 RST\_AddressBase** [41/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208S8.h.

**5.1.2.3283 RST\_AddressBase** [42/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF5268.h.

**5.1.2.3284 RST\_AddressBase** [43/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF6388.h.

**5.1.2.3285 RST\_AddressBase** [44/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF52A8.h.

**5.1.2.3286 RST\_AddressBase** [45/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208C8.h.

**5.1.2.3287 RST\_AddressBase** [46/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208R6.h.

**5.1.2.3288 RST\_AddressBase** [47/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208CB.h.

**5.1.2.3289 RST\_AddressBase** [48/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208RB.h.

**5.1.2.3290 RST\_AddressBase** [49/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF5288.h.

**5.1.2.3291 RST\_AddressBase** [50/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208MB.h.

**5.1.2.3292 RST\_AddressBase** [51/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF6286.h.

**5.1.2.3293 RST\_AddressBase** [52/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207M8.h.



**5.1.2.3294 RST\_AddressBase** [53/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207K6.h.

**5.1.2.3295 RST\_AddressBase** [54/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207K8.h.

**5.1.2.3296 RST\_AddressBase** [55/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF62A9.h.

**5.1.2.3297 RST\_AddressBase** [56/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF5289.h.

**5.1.2.3298 RST\_AddressBase** [57/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207C6.h.

**5.1.2.3299 RST\_AddressBase** [58/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF5286.h.

**5.1.2.3300 RST\_AddressBase** [59/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF6289.h.

**5.1.2.3301 RST\_AddressBase** [60/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208S6.h.

**5.1.2.3302 RST\_AddressBase** [61/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207C8.h.

**5.1.2.3303 RST\_AddressBase** [62/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207RB.h.

**5.1.2.3304 RST\_AddressBase** [63/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208R8.h.

**5.1.2.3305 RST\_AddressBase** [64/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S208C6.h.

**5.1.2.3306 RST\_AddressBase** [65/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF528A.h.

**5.1.2.3307 RST\_AddressBase** [66/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S207CB.h.

**5.1.2.3308 RST\_AddressBase** [67/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF628A.h.

**5.1.2.3309 RST\_AddressBase** [68/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF62A6.h.

**5.1.2.3310 RST\_AddressBase** [69/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8AF52A6.h.

**5.1.2.3311 RST\_AddressBase** [70/70]

```
#define RST_AddressBase 0x50B3
```

Definition at line 64 of file STM8S007C8.h.

**5.1.2.3312 SPI\_AddressBase** [1/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S003F3.h.

**5.1.2.3313 SPI\_AddressBase** [2/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S903K3.h.

**5.1.2.3314 SPI\_AddressBase** [3/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8AF6223A.h.

**5.1.2.3315 SPI\_AddressBase** [4/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S001J3.h.

**5.1.2.3316 SPI\_AddressBase** [5/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S003K3.h.

**5.1.2.3317 SPI\_AddressBase** [6/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S103K3.h.

**5.1.2.3318 SPI\_AddressBase** [7/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S103F2.h.

**5.1.2.3319 SPI\_AddressBase** [8/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8AF6223.h.

**5.1.2.3320 SPI\_AddressBase** [9/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8AF6213A.h.

**5.1.2.3321 SPI\_AddressBase** [10/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8AF6366.h.

**5.1.2.3322 SPI\_AddressBase** [11/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S903F3.h.

**5.1.2.3323 SPI\_AddressBase** [12/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8AF6226.h.

**5.1.2.3324 SPI\_AddressBase** [13/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8S103F3.h.

**5.1.2.3325 SPI\_AddressBase** [14/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 67 of file STM8AF6213.h.

**5.1.2.3326 SPI\_AddressBase** [15/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S005C6.h.

**5.1.2.3327 SPI\_AddressBase** [16/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8AF6248.h.

**5.1.2.3328 SPI\_AddressBase** [17/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8AF6266.h.

**5.1.2.3329 SPI\_AddressBase** [18/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S005K6.h.

**5.1.2.3330 SPI\_AddressBase** [19/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8AF6268.h.

**5.1.2.3331 SPI\_AddressBase** [20/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8AF6246.h.

**5.1.2.3332 SPI\_AddressBase** [21/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8AF6269.h.

**5.1.2.3333 SPI\_AddressBase** [22/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S105S6.h.

**5.1.2.3334 SPI\_AddressBase** [23/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S105K4.h.

**5.1.2.3335 SPI\_AddressBase** [24/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S105K6.h.

**5.1.2.3336 SPI\_AddressBase** [25/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S105S4.h.

**5.1.2.3337 SPI\_AddressBase** [26/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S105C4.h.

**5.1.2.3338 SPI\_AddressBase** [27/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 68 of file STM8S105C6.h.

**5.1.2.3339 SPI\_AddressBase** [28/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF6289.h.

**5.1.2.3340 SPI\_AddressBase** [29/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208MB.h.

**5.1.2.3341 SPI\_AddressBase** [30/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF5268.h.



**5.1.2.3342 SPI\_AddressBase** [31/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207RB.h.

**5.1.2.3343 SPI\_AddressBase** [32/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF52A8.h.

**5.1.2.3344 SPI\_AddressBase** [33/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF6288.h.

**5.1.2.3345 SPI\_AddressBase** [34/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207R8.h.

**5.1.2.3346 SPI\_AddressBase** [35/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207S6.h.

**5.1.2.3347 SPI\_AddressBase** [36/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207M8.h.

**5.1.2.3348 SPI\_AddressBase** [37/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF6286.h.

**5.1.2.3349 SPI\_AddressBase** [38/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF52AA.h.

**5.1.2.3350 SPI\_AddressBase** [39/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208R6.h.

**5.1.2.3351 SPI\_AddressBase** [40/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207C6.h.

**5.1.2.3352 SPI\_AddressBase** [41/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207SB.h.

**5.1.2.3353 SPI\_AddressBase** [42/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF62AA.h.

**5.1.2.3354 SPI\_AddressBase** [43/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208C6.h.

**5.1.2.3355 SPI\_AddressBase** [44/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207S8.h.

**5.1.2.3356 SPI\_AddressBase** [45/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208SB.h.

**5.1.2.3357 SPI\_AddressBase** [46/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207C8.h.

**5.1.2.3358 SPI\_AddressBase** [47/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208C8.h.

**5.1.2.3359 SPI\_AddressBase** [48/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207K8.h.

**5.1.2.3360 SPI\_AddressBase** [49/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207MB.h.

**5.1.2.3361 SPI\_AddressBase** [50/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207K6.h.

**5.1.2.3362 SPI\_AddressBase** [51/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF5288.h.

**5.1.2.3363 SPI\_AddressBase** [52/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF5269.h.

**5.1.2.3364 SPI\_AddressBase** [53/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207R6.h.

**5.1.2.3365 SPI\_AddressBase** [54/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF5289.h.

**5.1.2.3366 SPI\_AddressBase** [55/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208S6.h.

**5.1.2.3367 SPI\_AddressBase** [56/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208RB.h.

**5.1.2.3368 SPI\_AddressBase** [57/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S207CB.h.

**5.1.2.3369 SPI\_AddressBase** [58/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF62A6.h.

**5.1.2.3370 SPI\_AddressBase** [59/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208R8.h.

**5.1.2.3371 SPI\_AddressBase** [60/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF52A9.h.

**5.1.2.3372 SPI\_AddressBase** [61/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208S8.h.

**5.1.2.3373 SPI\_AddressBase** [62/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF62A8.h.

**5.1.2.3374 SPI\_AddressBase** [63/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S208CB.h.

**5.1.2.3375 SPI\_AddressBase** [64/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF528A.h.

**5.1.2.3376 SPI\_AddressBase** [65/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF62A9.h.

**5.1.2.3377 SPI\_AddressBase** [66/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF5286.h.

**5.1.2.3378 SPI\_AddressBase** [67/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8S007C8.h.

**5.1.2.3379 SPI\_AddressBase** [68/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF52A6.h.

**5.1.2.3380 SPI\_AddressBase** [69/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF628A.h.

**5.1.2.3381 SPI\_AddressBase** [70/70]

```
#define SPI_AddressBase 0x5200
```

Definition at line 70 of file STM8AF6388.h.

**5.1.2.3382 STM8\_ADDR\_WIDTH**

```
#define STM8_ADDR_WIDTH 16
```

width of address space

Definition at line 81 of file STM8AF\_STM8S.h.

**5.1.2.3383 STM8\_EEPROM\_END**

```
#define STM8_EEPROM_END (STM8_EEPROM_START + STM8_EEPROM_SIZE - 1)
```

last address in EEPROM

Definition at line 77 of file STM8AF\_STM8S.h.

**5.1.2.3384 STM8\_EEPROM\_SIZE** [1/71]

```
#define STM8_EEPROM_SIZE 1536
```

Definition at line 49 of file STM8S207S8.h.

**5.1.2.3385 STM8\_EEPROM\_SIZE** [2/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF6289.h.

**5.1.2.3386 STM8\_EEPROM\_SIZE** [3/71]

```
#define STM8_EEPROM_SIZE 1536
```

Definition at line 49 of file STM8S208SB.h.

**5.1.2.3387 STM8\_EEPROM\_SIZE** [4/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8S903F3.h.

**5.1.2.3388 STM8\_EEPROM\_SIZE** [5/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S207MB.h.

**5.1.2.3389 STM8\_EEPROM\_SIZE** [6/71]

```
#define STM8_EEPROM_SIZE 1536
```

Definition at line 49 of file STM8S207R8.h.



**5.1.2.3390 STM8\_EEPROM\_SIZE** [7/71]

```
#define STM8_EEPROM_SIZE 128
```

Definition at line 49 of file STM8S007C8.h.

**5.1.2.3391 STM8\_EEPROM\_SIZE** [8/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S207RB.h.

**5.1.2.3392 STM8\_EEPROM\_SIZE** [9/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8AF6213.h.

**5.1.2.3393 STM8\_EEPROM\_SIZE** [10/71]

```
#define STM8_EEPROM_SIZE 1536
```

Definition at line 49 of file STM8S208S6.h.

**5.1.2.3394 STM8\_EEPROM\_SIZE** [11/71]

```
#define STM8_EEPROM_SIZE 128
```

Definition at line 49 of file STM8S001J3.h.

**5.1.2.3395 STM8\_EEPROM\_SIZE** [12/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S207K8.h.

**5.1.2.3396 STM8\_EEPROM\_SIZE** [13/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8AF5268.h.

**5.1.2.3397 STM8\_EEPROM\_SIZE** [14/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S207R6.h.

**5.1.2.3398 STM8\_EEPROM\_SIZE** [15/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8AF6266.h.

**5.1.2.3399 STM8\_EEPROM\_SIZE** [16/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF6288.h.

**5.1.2.3400 STM8\_EEPROM\_SIZE** [17/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF5288.h.

**5.1.2.3401 STM8\_EEPROM\_SIZE** [18/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S207S6.h.

**5.1.2.3402 STM8\_EEPROM\_SIZE** [19/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF5289.h.

**5.1.2.3403 STM8\_EEPROM\_SIZE** [20/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8AF6268.h.

**5.1.2.3404 STM8\_EEPROM\_SIZE** [21/71]

```
#define STM8_EEPROM_SIZE 512
```

Definition at line 49 of file STM8AF6246.h.

**5.1.2.3405 STM8\_EEPROM\_SIZE** [22/71]

```
#define STM8_EEPROM_SIZE 1536
```

Definition at line 49 of file STM8S207SB.h.

**5.1.2.3406 STM8\_EEPROM\_SIZE** [23/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF528A.h.

**5.1.2.3407 STM8\_EEPROM\_SIZE** [24/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8AF6366.h.

**5.1.2.3408 STM8\_EEPROM\_SIZE** [25/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF6388.h.

**5.1.2.3409 STM8\_EEPROM\_SIZE** [26/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S208CB.h.

**5.1.2.3410 STM8\_EEPROM\_SIZE** [27/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S208RB.h.

**5.1.2.3411 STM8\_EEPROM\_SIZE** [28/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF52AA.h.

**5.1.2.3412 STM8\_EEPROM\_SIZE** [29/71]

```
#define STM8_EEPROM_SIZE 1536
```

Definition at line 49 of file STM8S208S8.h.

**5.1.2.3413 STM8\_EEPROM\_SIZE** [30/71]

```
#define STM8_EEPROM_SIZE 128
```

Definition at line 49 of file STM8S005K6.h.

**5.1.2.3414 STM8\_EEPROM\_SIZE** [31/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S207K6.h.

**5.1.2.3415 STM8\_EEPROM\_SIZE** [32/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S207M8.h.

**5.1.2.3416 STM8\_EEPROM\_SIZE** [33/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S207C6.h.

**5.1.2.3417 STM8\_EEPROM\_SIZE** [34/71]

```
#define STM8_EEPROM_SIZE 128
```

Definition at line 49 of file STM8S003K3.h.

**5.1.2.3418 STM8\_EEPROM\_SIZE** [35/71]

```
#define STM8_EEPROM_SIZE 128
```

Definition at line 49 of file STM8S005C6.h.

**5.1.2.3419 STM8\_EEPROM\_SIZE** [36/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8S903K3.h.

**5.1.2.3420 STM8\_EEPROM\_SIZE** [37/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF628A.h.

**5.1.2.3421 STM8\_EEPROM\_SIZE** [38/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF62A8.h.

**5.1.2.3422 STM8\_EEPROM\_SIZE** [39/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF6286.h.

**5.1.2.3423 STM8\_EEPROM\_SIZE** [40/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S208R6.h.

**5.1.2.3424 STM8\_EEPROM\_SIZE** [41/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S208MB.h.

**5.1.2.3425 STM8\_EEPROM\_SIZE** [42/71]

```
#define STM8_EEPROM_SIZE 1536
```

Definition at line 49 of file STM8S207C8.h.

**5.1.2.3426 STM8\_EEPROM\_SIZE** [43/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8S103F3.h.

**5.1.2.3427 STM8\_EEPROM\_SIZE** [44/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8S103K3.h.

**5.1.2.3428 STM8\_EEPROM\_SIZE** [45/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8AF5269.h.

**5.1.2.3429 STM8\_EEPROM\_SIZE** [46/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8AF6226.h.

**5.1.2.3430 STM8\_EEPROM\_SIZE** [47/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S105C6.h.

**5.1.2.3431 STM8\_EEPROM\_SIZE** [48/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF52A6.h.

**5.1.2.3432 STM8\_EEPROM\_SIZE** [49/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S207CB.h.

**5.1.2.3433 STM8\_EEPROM\_SIZE** [50/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S105S6.h.

**5.1.2.3434 STM8\_EEPROM\_SIZE** [51/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S208R8.h.

**5.1.2.3435 STM8\_EEPROM\_SIZE** [52/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8AF6223A.h.

**5.1.2.3436 STM8\_EEPROM\_SIZE** [53/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8AF6269.h.

**5.1.2.3437 STM8\_EEPROM\_SIZE** [54/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8AF6223.h.



**5.1.2.3438 STM8\_EEPROM\_SIZE** [55/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF5286.h.

**5.1.2.3439 STM8\_EEPROM\_SIZE** [56/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S208C6.h.

**5.1.2.3440 STM8\_EEPROM\_SIZE** [57/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S105K6.h.

**5.1.2.3441 STM8\_EEPROM\_SIZE** [58/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF52A8.h.

**5.1.2.3442 STM8\_EEPROM\_SIZE** [59/71]

```
#define STM8_EEPROM_SIZE 512
```

Definition at line 49 of file STM8AF6248.h.

**5.1.2.3443 STM8\_EEPROM\_SIZE** [60/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF62A6.h.

**5.1.2.3444 STM8\_EEPROM\_SIZE** [61/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8AF6213A.h.

**5.1.2.3445 STM8\_EEPROM\_SIZE** [62/71]

```
#define STM8_EEPROM_SIZE 128
```

Definition at line 49 of file STM8S003F3.h.

**5.1.2.3446 STM8\_EEPROM\_SIZE** [63/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF52A9.h.

**5.1.2.3447 STM8\_EEPROM\_SIZE** [64/71]

```
#define STM8_EEPROM_SIZE 640
```

Definition at line 49 of file STM8S103F2.h.

**5.1.2.3448 STM8\_EEPROM\_SIZE** [65/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF62AA.h.

**5.1.2.3449 STM8\_EEPROM\_SIZE** [66/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S105S4.h.

**5.1.2.3450 STM8\_EEPROM\_SIZE** [67/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S105K4.h.

**5.1.2.3451 STM8\_EEPROM\_SIZE** [68/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8S208C8.h.

**5.1.2.3452 STM8\_EEPROM\_SIZE** [69/71]

```
#define STM8_EEPROM_SIZE 2048
```

Definition at line 49 of file STM8AF62A9.h.

**5.1.2.3453 STM8\_EEPROM\_SIZE** [70/71]

```
#define STM8_EEPROM_SIZE 1024
```

Definition at line 49 of file STM8S105C4.h.

**5.1.2.3454 STM8\_EEPROM\_SIZE** [71/71]

```
#define STM8_EEPROM_SIZE 128
```

size of data EEPROM [B]

Definition at line 68 of file STM8AF\_STM8S.h.

**5.1.2.3455 STM8\_EEPROM\_START**

```
#define STM8_EEPROM_START 0x4000
```

first address in EEPROM

Definition at line 76 of file STM8AF\_STM8S.h.

**5.1.2.3456 STM8\_MEM\_POINTER\_T**

```
#define STM8_MEM_POINTER_T uint16_t
```

address variable type

Definition at line 82 of file STM8AF\_STM8S.h.

**5.1.2.3457 STM8\_PFLASH\_END**

```
#define STM8_PFLASH_END (STM8_PFLASH_START + STM8_PFLASH_SIZE - 1)
```

last address in program flash

Definition at line 73 of file STM8AF\_STM8S.h.

**5.1.2.3458 STM8\_PFLASH\_SIZE** [1/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S208SB.h.

**5.1.2.3459 STM8\_PFLASH\_SIZE** [2/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S207RB.h.

**5.1.2.3460 STM8\_PFLASH\_SIZE** [3/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S207S6.h.

**5.1.2.3461 STM8\_PFLASH\_SIZE** [4/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF6286.h.

**5.1.2.3462 STM8\_PFLASH\_SIZE** [5/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF628A.h.

**5.1.2.3463 STM8\_PFLASH\_SIZE** [6/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF62AA.h.

**5.1.2.3464 STM8\_PFLASH\_SIZE** [7/71]

```
#define STM8_PFLASH_SIZE 16*1024
```

Definition at line 47 of file STM8AF6248.h.

**5.1.2.3465 STM8\_PFLASH\_SIZE** [8/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S207MB.h.

**5.1.2.3466 STM8\_PFLASH\_SIZE** [9/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8S003F3.h.

**5.1.2.3467 STM8\_PFLASH\_SIZE** [10/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8S001J3.h.

**5.1.2.3468 STM8\_PFLASH\_SIZE** [11/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8AF6266.h.

**5.1.2.3469 STM8\_PFLASH\_SIZE** [12/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S208RB.h.

**5.1.2.3470 STM8\_PFLASH\_SIZE** [13/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF52A9.h.

**5.1.2.3471 STM8\_PFLASH\_SIZE** [14/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S207K6.h.

**5.1.2.3472 STM8\_PFLASH\_SIZE** [15/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF5289.h.

**5.1.2.3473 STM8\_PFLASH\_SIZE** [16/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF5286.h.

**5.1.2.3474 STM8\_PFLASH\_SIZE** [17/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S207S8.h.

**5.1.2.3475 STM8\_PFLASH\_SIZE** [18/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8AF6223.h.

**5.1.2.3476 STM8\_PFLASH\_SIZE** [19/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S208MB.h.

**5.1.2.3477 STM8\_PFLASH\_SIZE** [20/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF52A6.h.

**5.1.2.3478 STM8\_PFLASH\_SIZE** [21/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8AF5268.h.

**5.1.2.3479 STM8\_PFLASH\_SIZE** [22/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S207K8.h.

**5.1.2.3480 STM8\_PFLASH\_SIZE** [23/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8S903F3.h.

**5.1.2.3481 STM8\_PFLASH\_SIZE** [24/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S207R8.h.

**5.1.2.3482 STM8\_PFLASH\_SIZE** [25/71]

```
#define STM8_PFLASH_SIZE 4*1024
```

Definition at line 47 of file STM8AF6213.h.

**5.1.2.3483 STM8\_PFLASH\_SIZE** [26/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8S003K3.h.

**5.1.2.3484 STM8\_PFLASH\_SIZE** [27/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S007C8.h.

**5.1.2.3485 STM8\_PFLASH\_SIZE** [28/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S005C6.h.



**5.1.2.3486 STM8\_PFLASH\_SIZE** [29/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF6289.h.

**5.1.2.3487 STM8\_PFLASH\_SIZE** [30/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S005K6.h.

**5.1.2.3488 STM8\_PFLASH\_SIZE** [31/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF6388.h.

**5.1.2.3489 STM8\_PFLASH\_SIZE** [32/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S208C8.h.

**5.1.2.3490 STM8\_PFLASH\_SIZE** [33/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S208S8.h.

**5.1.2.3491 STM8\_PFLASH\_SIZE** [34/71]

```
#define STM8_PFLASH_SIZE 4*1024
```

Definition at line 47 of file STM8S103F2.h.

**5.1.2.3492 STM8\_PFLASH\_SIZE** [35/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF62A9.h.

**5.1.2.3493 STM8\_PFLASH\_SIZE** [36/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S207CB.h.

**5.1.2.3494 STM8\_PFLASH\_SIZE** [37/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF5288.h.

**5.1.2.3495 STM8\_PFLASH\_SIZE** [38/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8AF6268.h.

**5.1.2.3496 STM8\_PFLASH\_SIZE** [39/71]

```
#define STM8_PFLASH_SIZE 16*1024
```

Definition at line 47 of file STM8AF6246.h.

**5.1.2.3497 STM8\_PFLASH\_SIZE** [40/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8S103K3.h.

**5.1.2.3498 STM8\_PFLASH\_SIZE** [41/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S208R6.h.

**5.1.2.3499 STM8\_PFLASH\_SIZE** [42/71]

```
#define STM8_PFLASH_SIZE 16*1024
```

Definition at line 47 of file STM8S105K4.h.

**5.1.2.3500 STM8\_PFLASH\_SIZE** [43/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S208S6.h.

**5.1.2.3501 STM8\_PFLASH\_SIZE** [44/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S208CB.h.

**5.1.2.3502 STM8\_PFLASH\_SIZE** [45/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF52AA.h.

**5.1.2.3503 STM8\_PFLASH\_SIZE** [46/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S105K6.h.

**5.1.2.3504 STM8\_PFLASH\_SIZE** [47/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8AF6223A.h.

**5.1.2.3505 STM8\_PFLASH\_SIZE** [48/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8AF6269.h.

**5.1.2.3506 STM8\_PFLASH\_SIZE** [49/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S208R8.h.

**5.1.2.3507 STM8\_PFLASH\_SIZE** [50/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF528A.h.

**5.1.2.3508 STM8\_PFLASH\_SIZE** [51/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S207C6.h.

**5.1.2.3509 STM8\_PFLASH\_SIZE** [52/71]

```
#define STM8_PFLASH_SIZE 4*1024
```

Definition at line 47 of file STM8AF6213A.h.

**5.1.2.3510 STM8\_PFLASH\_SIZE** [53/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S105S6.h.

**5.1.2.3511 STM8\_PFLASH\_SIZE** [54/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S207C8.h.

**5.1.2.3512 STM8\_PFLASH\_SIZE** [55/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8AF5269.h.

**5.1.2.3513 STM8\_PFLASH\_SIZE** [56/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF62A8.h.

**5.1.2.3514 STM8\_PFLASH\_SIZE** [57/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8S903K3.h.

**5.1.2.3515 STM8\_PFLASH\_SIZE** [58/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S208C6.h.

**5.1.2.3516 STM8\_PFLASH\_SIZE** [59/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF62A6.h.

**5.1.2.3517 STM8\_PFLASH\_SIZE** [60/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8S103F3.h.

**5.1.2.3518 STM8\_PFLASH\_SIZE** [61/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S207R6.h.

**5.1.2.3519 STM8\_PFLASH\_SIZE** [62/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8AF6288.h.

**5.1.2.3520 STM8\_PFLASH\_SIZE** [63/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8S105C6.h.

**5.1.2.3521 STM8\_PFLASH\_SIZE** [64/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8S207SB.h.

**5.1.2.3522 STM8\_PFLASH\_SIZE** [65/71]

```
#define STM8_PFLASH_SIZE 32*1024
```

Definition at line 47 of file STM8AF6366.h.

**5.1.2.3523 STM8\_PFLASH\_SIZE** [66/71]

```
#define STM8_PFLASH_SIZE 16*1024
```

Definition at line 47 of file STM8S105S4.h.

**5.1.2.3524 STM8\_PFLASH\_SIZE** [67/71]

```
#define STM8_PFLASH_SIZE 8*1024
```

Definition at line 47 of file STM8AF6226.h.

**5.1.2.3525 STM8\_PFLASH\_SIZE** [68/71]

```
#define STM8_PFLASH_SIZE 64*1024
```

Definition at line 47 of file STM8S207M8.h.

**5.1.2.3526 STM8\_PFLASH\_SIZE** [69/71]

```
#define STM8_PFLASH_SIZE 128*1024
```

Definition at line 47 of file STM8AF52A8.h.

**5.1.2.3527 STM8\_PFLASH\_SIZE** [70/71]

```
#define STM8_PFLASH_SIZE 16*1024
```

Definition at line 47 of file STM8S105C4.h.

**5.1.2.3528 STM8\_PFLASH\_SIZE** [71/71]

```
#define STM8_PFLASH_SIZE 2*1024
```

size of program flash [B]

Definition at line 60 of file STM8AF\_STM8S.h.

**5.1.2.3529 STM8\_PFLASH\_START**

```
#define STM8_PFLASH_START 0x8000
```

first address in program flash

Definition at line 72 of file STM8AF\_STM8S.h.

**5.1.2.3530 STM8\_RAM\_END**

```
#define STM8_RAM_END (STM8_RAM_START + STM8_RAM_SIZE - 1)
```

last address in RAM

Definition at line 75 of file STM8AF\_STM8S.h.

**5.1.2.3531 STM8\_RAM\_SIZE** [1/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207SB.h.

**5.1.2.3532 STM8\_RAM\_SIZE** [2/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF628A.h.



**5.1.2.3533 STM8\_RAM\_SIZE** [3/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8AF6266.h.

**5.1.2.3534 STM8\_RAM\_SIZE** [4/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S903F3.h.

**5.1.2.3535 STM8\_RAM\_SIZE** [5/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S103F2.h.

**5.1.2.3536 STM8\_RAM\_SIZE** [6/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208SB.h.

**5.1.2.3537 STM8\_RAM\_SIZE** [7/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S007C8.h.

**5.1.2.3538 STM8\_RAM\_SIZE** [8/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF5288.h.

**5.1.2.3539 STM8\_RAM\_SIZE** [9/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8AF6213.h.

**5.1.2.3540 STM8\_RAM\_SIZE** [10/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207K8.h.

**5.1.2.3541 STM8\_RAM\_SIZE** [11/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S001J3.h.

**5.1.2.3542 STM8\_RAM\_SIZE** [12/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207M8.h.

**5.1.2.3543 STM8\_RAM\_SIZE** [13/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S003F3.h.

**5.1.2.3544 STM8\_RAM\_SIZE** [14/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF52A9.h.

**5.1.2.3545 STM8\_RAM\_SIZE** [15/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207K6.h.

**5.1.2.3546 STM8\_RAM\_SIZE** [16/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208R6.h.

**5.1.2.3547 STM8\_RAM\_SIZE** [17/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8AF6268.h.

**5.1.2.3548 STM8\_RAM\_SIZE** [18/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF5286.h.

**5.1.2.3549 STM8\_RAM\_SIZE** [19/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S103F3.h.

**5.1.2.3550 STM8\_RAM\_SIZE** [20/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S105S6.h.

**5.1.2.3551 STM8\_RAM\_SIZE** [21/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207R6.h.

**5.1.2.3552 STM8\_RAM\_SIZE** [22/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8AF6246.h.

**5.1.2.3553 STM8\_RAM\_SIZE** [23/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208S8.h.

**5.1.2.3554 STM8\_RAM\_SIZE** [24/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF62A8.h.

**5.1.2.3555 STM8\_RAM\_SIZE** [25/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S103K3.h.

**5.1.2.3556 STM8\_RAM\_SIZE** [26/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF5289.h.

**5.1.2.3557 STM8\_RAM\_SIZE** [27/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S005C6.h.

**5.1.2.3558 STM8\_RAM\_SIZE** [28/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8AF6223A.h.

**5.1.2.3559 STM8\_RAM\_SIZE** [29/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208MB.h.

**5.1.2.3560 STM8\_RAM\_SIZE** [30/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S105K6.h.

**5.1.2.3561 STM8\_RAM\_SIZE** [31/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208RB.h.

**5.1.2.3562 STM8\_RAM\_SIZE** [32/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF528A.h.

**5.1.2.3563 STM8\_RAM\_SIZE** [33/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S105C4.h.

**5.1.2.3564 STM8\_RAM\_SIZE** [34/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208C6.h.

**5.1.2.3565 STM8\_RAM\_SIZE** [35/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207RB.h.

**5.1.2.3566 STM8\_RAM\_SIZE** [36/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8AF6226.h.

**5.1.2.3567 STM8\_RAM\_SIZE** [37/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF5268.h.

**5.1.2.3568 STM8\_RAM\_SIZE** [38/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF6388.h.

**5.1.2.3569 STM8\_RAM\_SIZE** [39/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207S8.h.

**5.1.2.3570 STM8\_RAM\_SIZE** [40/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF52AA.h.

**5.1.2.3571 STM8\_RAM\_SIZE** [41/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207S6.h.

**5.1.2.3572 STM8\_RAM\_SIZE** [42/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8AF6248.h.

**5.1.2.3573 STM8\_RAM\_SIZE** [43/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208R8.h.

**5.1.2.3574 STM8\_RAM\_SIZE** [44/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF6289.h.

**5.1.2.3575 STM8\_RAM\_SIZE** [45/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207CB.h.

**5.1.2.3576 STM8\_RAM\_SIZE** [46/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF5269.h.

**5.1.2.3577 STM8\_RAM\_SIZE** [47/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208C8.h.

**5.1.2.3578 STM8\_RAM\_SIZE** [48/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF52A8.h.

**5.1.2.3579 STM8\_RAM\_SIZE** [49/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF6269.h.

**5.1.2.3580 STM8\_RAM\_SIZE** [50/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF52A6.h.



**5.1.2.3581 STM8\_RAM\_SIZE** [51/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8AF6366.h.

**5.1.2.3582 STM8\_RAM\_SIZE** [52/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S903K3.h.

**5.1.2.3583 STM8\_RAM\_SIZE** [53/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S005K6.h.

**5.1.2.3584 STM8\_RAM\_SIZE** [54/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208CB.h.

**5.1.2.3585 STM8\_RAM\_SIZE** [55/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF6288.h.

**5.1.2.3586 STM8\_RAM\_SIZE** [56/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF62AA.h.

**5.1.2.3587 STM8\_RAM\_SIZE** [57/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S105S4.h.

**5.1.2.3588 STM8\_RAM\_SIZE** [58/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207MB.h.

**5.1.2.3589 STM8\_RAM\_SIZE** [59/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8S003K3.h.

**5.1.2.3590 STM8\_RAM\_SIZE** [60/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S105C6.h.

**5.1.2.3591 STM8\_RAM\_SIZE** [61/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF62A6.h.

**5.1.2.3592 STM8\_RAM\_SIZE** [62/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF6286.h.

**5.1.2.3593 STM8\_RAM\_SIZE** [63/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8AF6213A.h.

**5.1.2.3594 STM8\_RAM\_SIZE** [64/71]

```
#define STM8_RAM_SIZE 2*1024
```

Definition at line 48 of file STM8S105K4.h.

**5.1.2.3595 STM8\_RAM\_SIZE** [65/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207C8.h.

**5.1.2.3596 STM8\_RAM\_SIZE** [66/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S208S6.h.

**5.1.2.3597 STM8\_RAM\_SIZE** [67/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207C6.h.

**5.1.2.3598 STM8\_RAM\_SIZE** [68/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8AF62A9.h.

**5.1.2.3599 STM8\_RAM\_SIZE** [69/71]

```
#define STM8_RAM_SIZE 1*1024
```

Definition at line 48 of file STM8AF6223.h.

**5.1.2.3600 STM8\_RAM\_SIZE** [70/71]

```
#define STM8_RAM_SIZE 6*1024
```

Definition at line 48 of file STM8S207R8.h.

**5.1.2.3601 STM8\_RAM\_SIZE** [71/71]

```
#define STM8_RAM_SIZE 1*1024
```

size of RAM [B]

Definition at line 64 of file STM8AF\_STM8S.h.

**5.1.2.3602 STM8\_RAM\_START**

```
#define STM8_RAM_START 0x0000
```

first address in RAM

Definition at line 74 of file STM8AF\_STM8S.h.

**5.1.2.3603 STM8AF5268**

```
#define STM8AF5268
```

Definition at line 40 of file STM8AF5268.h.

**5.1.2.3604 STM8AF5269**

```
#define STM8AF5269
```

Definition at line 40 of file STM8AF5269.h.

**5.1.2.3605 STM8AF526x** [1/2]

```
#define STM8AF526x
```

Definition at line 43 of file STM8AF5269.h.

**5.1.2.3606 STM8AF526x** [2/2]

```
#define STM8AF526x
```

Definition at line 43 of file STM8AF5268.h.

**5.1.2.3607 STM8AF5286**

```
#define STM8AF5286
```

Definition at line 40 of file STM8AF5286.h.

**5.1.2.3608 STM8AF5288**

```
#define STM8AF5288
```

Definition at line 40 of file STM8AF5288.h.

**5.1.2.3609 STM8AF5289**

```
#define STM8AF5289
```

Definition at line 40 of file STM8AF5289.h.

**5.1.2.3610 STM8AF528A**

```
#define STM8AF528A
```

Definition at line 40 of file STM8AF528A.h.

**5.1.2.3611 STM8AF528x** [1/4]

```
#define STM8AF528x
```

Definition at line 43 of file STM8AF528A.h.

**5.1.2.3612 STM8AF528x** [2/4]

```
#define STM8AF528x
```

Definition at line 43 of file STM8AF5286.h.

**5.1.2.3613 STM8AF528x** [3/4]

```
#define STM8AF528x
```

Definition at line 43 of file STM8AF5289.h.

**5.1.2.3614 STM8AF528x** [4/4]

```
#define STM8AF528x
```

Definition at line 43 of file STM8AF5288.h.

**5.1.2.3615 STM8AF52A6**

```
#define STM8AF52A6
```

Definition at line 40 of file STM8AF52A6.h.

**5.1.2.3616 STM8AF52A8**

```
#define STM8AF52A8
```

Definition at line 40 of file STM8AF52A8.h.

**5.1.2.3617 STM8AF52A9**

```
#define STM8AF52A9
```

Definition at line 40 of file STM8AF52A9.h.

**5.1.2.3618 STM8AF52AA**

```
#define STM8AF52AA
```

Definition at line 40 of file STM8AF52AA.h.

**5.1.2.3619 STM8AF52Ax** [1/4]

```
#define STM8AF52Ax
```

Definition at line 43 of file STM8AF52A9.h.

**5.1.2.3620 STM8AF52Ax** [2/4]

```
#define STM8AF52Ax
```

Definition at line 43 of file STM8AF52AA.h.

**5.1.2.3621 STM8AF52Ax** [3/4]

```
#define STM8AF52Ax
```

Definition at line 43 of file STM8AF52A8.h.

**5.1.2.3622 STM8AF52Ax** [4/4]

```
#define STM8AF52Ax
```

Definition at line 43 of file STM8AF52A6.h.

**5.1.2.3623 STM8AF6213**

```
#define STM8AF6213
```

Definition at line 40 of file STM8AF6213.h.

**5.1.2.3624 STM8AF6213A**

```
#define STM8AF6213A
```

Definition at line 40 of file STM8AF6213A.h.

**5.1.2.3625 STM8AF621x** [1/2]

```
#define STM8AF621x
```

Definition at line 43 of file STM8AF6213.h.

**5.1.2.3626 STM8AF621x** [2/2]

```
#define STM8AF621x
```

Definition at line 43 of file STM8AF6213A.h.

**5.1.2.3627 STM8AF6223**

```
#define STM8AF6223
```

Definition at line 40 of file STM8AF6223.h.

**5.1.2.3628 STM8AF6223A**

```
#define STM8AF6223A
```

Definition at line 40 of file STM8AF6223A.h.



**5.1.2.3629 STM8AF6226**

```
#define STM8AF6226
```

Definition at line 40 of file STM8AF6226.h.

**5.1.2.3630 STM8AF622x** [1/3]

```
#define STM8AF622x
```

Definition at line 43 of file STM8AF6223A.h.

**5.1.2.3631 STM8AF622x** [2/3]

```
#define STM8AF622x
```

Definition at line 43 of file STM8AF6223.h.

**5.1.2.3632 STM8AF622x** [3/3]

```
#define STM8AF622x
```

Definition at line 43 of file STM8AF6226.h.

**5.1.2.3633 STM8AF6246**

```
#define STM8AF6246
```

Definition at line 40 of file STM8AF6246.h.

**5.1.2.3634 STM8AF6248**

```
#define STM8AF6248
```

Definition at line 40 of file STM8AF6248.h.

**5.1.2.3635 STM8AF624x** [1/2]

```
#define STM8AF624x
```

Definition at line 43 of file STM8AF6246.h.

**5.1.2.3636 STM8AF624x** [2/2]

```
#define STM8AF624x
```

Definition at line 43 of file STM8AF6248.h.

**5.1.2.3637 STM8AF6266**

```
#define STM8AF6266
```

Definition at line 40 of file STM8AF6266.h.

**5.1.2.3638 STM8AF6268**

```
#define STM8AF6268
```

Definition at line 40 of file STM8AF6268.h.

**5.1.2.3639 STM8AF6269**

```
#define STM8AF6269
```

Definition at line 40 of file STM8AF6269.h.

**5.1.2.3640 STM8AF626x** [1/3]

```
#define STM8AF626x
```

Definition at line 43 of file STM8AF6266.h.

**5.1.2.3641 STM8AF626x** [2/3]

```
#define STM8AF626x
```

Definition at line 43 of file STM8AF6268.h.

**5.1.2.3642 STM8AF626x** [3/3]

```
#define STM8AF626x
```

Definition at line 43 of file STM8AF6269.h.

**5.1.2.3643 STM8AF6286**

```
#define STM8AF6286
```

Definition at line 40 of file STM8AF6286.h.

**5.1.2.3644 STM8AF6288**

```
#define STM8AF6288
```

Definition at line 40 of file STM8AF6288.h.

**5.1.2.3645 STM8AF6289**

```
#define STM8AF6289
```

Definition at line 40 of file STM8AF6289.h.

**5.1.2.3646 STM8AF628A**

```
#define STM8AF628A
```

Definition at line 40 of file STM8AF628A.h.

**5.1.2.3647 STM8AF628x** [1/4]

```
#define STM8AF628x
```

Definition at line 43 of file STM8AF6288.h.

**5.1.2.3648 STM8AF628x** [2/4]

```
#define STM8AF628x
```

Definition at line 43 of file STM8AF6289.h.

**5.1.2.3649 STM8AF628x** [3/4]

```
#define STM8AF628x
```

Definition at line 43 of file STM8AF6286.h.

**5.1.2.3650 STM8AF628x** [4/4]

```
#define STM8AF628x
```

Definition at line 43 of file STM8AF628A.h.

**5.1.2.3651 STM8AF62A6**

```
#define STM8AF62A6
```

Definition at line 40 of file STM8AF62A6.h.

**5.1.2.3652 STM8AF62A8**

```
#define STM8AF62A8
```

Definition at line 40 of file STM8AF62A8.h.

**5.1.2.3653 STM8AF62A9**

```
#define STM8AF62A9
```

Definition at line 40 of file STM8AF62A9.h.

**5.1.2.3654 STM8AF62AA**

```
#define STM8AF62AA
```

Definition at line 40 of file STM8AF62AA.h.

**5.1.2.3655 STM8AF62Ax** [1/4]

```
#define STM8AF62Ax
```

Definition at line 43 of file STM8AF62AA.h.

**5.1.2.3656 STM8AF62Ax** [2/4]

```
#define STM8AF62Ax
```

Definition at line 43 of file STM8AF62A9.h.

**5.1.2.3657 STM8AF62Ax** [3/4]

```
#define STM8AF62Ax
```

Definition at line 43 of file STM8AF62A6.h.

**5.1.2.3658 STM8AF62Ax** [4/4]

```
#define STM8AF62Ax
```

Definition at line 43 of file STM8AF62A8.h.

**5.1.2.3659 STM8AF6366**

```
#define STM8AF6366
```

Definition at line 40 of file STM8AF6366.h.

**5.1.2.3660 STM8AF636x**

```
#define STM8AF636x
```

Definition at line 43 of file STM8AF6366.h.

**5.1.2.3661 STM8AF6388**

```
#define STM8AF6388
```

Definition at line 40 of file STM8AF6388.h.

**5.1.2.3662 STM8AF638x**

```
#define STM8AF638x
```

Definition at line 43 of file STM8AF6388.h.

**5.1.2.3663 STM8S001**

```
#define STM8S001
```

Definition at line 43 of file STM8S001J3.h.

**5.1.2.3664 STM8S001J3**

```
#define STM8S001J3
```

Definition at line 40 of file STM8S001J3.h.

**5.1.2.3665 STM8S003** [1/2]

```
#define STM8S003
```

Definition at line 43 of file STM8S003F3.h.

**5.1.2.3666 STM8S003** [2/2]

```
#define STM8S003
```

Definition at line 43 of file STM8S003K3.h.

**5.1.2.3667 STM8S003F3**

```
#define STM8S003F3
```

Definition at line 40 of file STM8S003F3.h.

**5.1.2.3668 STM8S003K3**

```
#define STM8S003K3
```

Definition at line 40 of file STM8S003K3.h.

**5.1.2.3669 STM8S005** [1/2]

```
#define STM8S005
```

Definition at line 43 of file STM8S005K6.h.

**5.1.2.3670 STM8S005** [2/2]

```
#define STM8S005
```

Definition at line 43 of file STM8S005C6.h.

**5.1.2.3671 STM8S005C6**

```
#define STM8S005C6
```

Definition at line 40 of file STM8S005C6.h.

**5.1.2.3672 STM8S005K6**

```
#define STM8S005K6
```

Definition at line 40 of file STM8S005K6.h.

**5.1.2.3673 STM8S007**

```
#define STM8S007
```

Definition at line 43 of file STM8S007C8.h.

**5.1.2.3674 STM8S007C8**

```
#define STM8S007C8
```

Definition at line 40 of file STM8S007C8.h.

**5.1.2.3675 STM8S103** [1/3]

```
#define STM8S103
```

Definition at line 43 of file STM8S103F2.h.

**5.1.2.3676 STM8S103** [2/3]

```
#define STM8S103
```

Definition at line 43 of file STM8S103F3.h.



**5.1.2.3677 STM8S103** [3/3]

```
#define STM8S103
```

Definition at line 43 of file STM8S103K3.h.

**5.1.2.3678 STM8S103F2**

```
#define STM8S103F2
```

Definition at line 40 of file STM8S103F2.h.

**5.1.2.3679 STM8S103F3**

```
#define STM8S103F3
```

Definition at line 40 of file STM8S103F3.h.

**5.1.2.3680 STM8S103K3**

```
#define STM8S103K3
```

Definition at line 40 of file STM8S103K3.h.

**5.1.2.3681 STM8S105** [1/6]

```
#define STM8S105
```

Definition at line 43 of file STM8S105K4.h.

**5.1.2.3682 STM8S105** [2/6]

```
#define STM8S105
```

Definition at line 43 of file STM8S105C6.h.

**5.1.2.3683 STM8S105** [3/6]

```
#define STM8S105
```

Definition at line 43 of file STM8S105C4.h.

**5.1.2.3684 STM8S105** [4/6]

```
#define STM8S105
```

Definition at line 43 of file STM8S105S4.h.

**5.1.2.3685 STM8S105** [5/6]

```
#define STM8S105
```

Definition at line 43 of file STM8S105S6.h.

**5.1.2.3686 STM8S105** [6/6]

```
#define STM8S105
```

Definition at line 43 of file STM8S105K6.h.

**5.1.2.3687 STM8S105C4**

```
#define STM8S105C4
```

Definition at line 40 of file STM8S105C4.h.

**5.1.2.3688 STM8S105C6**

```
#define STM8S105C6
```

Definition at line 40 of file STM8S105C6.h.

**5.1.2.3689 STM8S105K4**

```
#define STM8S105K4
```

Definition at line 40 of file STM8S105K4.h.

**5.1.2.3690 STM8S105K6**

```
#define STM8S105K6
```

Definition at line 40 of file STM8S105K6.h.

**5.1.2.3691 STM8S105S4**

```
#define STM8S105S4
```

Definition at line 40 of file STM8S105S4.h.

**5.1.2.3692 STM8S105S6**

```
#define STM8S105S6
```

Definition at line 40 of file STM8S105S6.h.

**5.1.2.3693 STM8S207** [1/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207S6.h.

**5.1.2.3694 STM8S207** [2/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207CB.h.

**5.1.2.3695 STM8S207** [3/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207R6.h.

**5.1.2.3696 STM8S207** [4/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207M8.h.

**5.1.2.3697 STM8S207** [5/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207K6.h.

**5.1.2.3698 STM8S207** [6/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207R8.h.

**5.1.2.3699 STM8S207** [7/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207S8.h.

**5.1.2.3700 STM8S207** [8/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207RB.h.

**5.1.2.3701 STM8S207** [9/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207SB.h.

**5.1.2.3702 STM8S207** [10/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207MB.h.

**5.1.2.3703 STM8S207** [11/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207C8.h.

**5.1.2.3704 STM8S207** [12/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207C6.h.

**5.1.2.3705 STM8S207** [13/13]

```
#define STM8S207
```

Definition at line 43 of file STM8S207K8.h.

**5.1.2.3706 STM8S207C6**

```
#define STM8S207C6
```

Definition at line 40 of file STM8S207C6.h.

**5.1.2.3707 STM8S207C8**

```
#define STM8S207C8
```

Definition at line 40 of file STM8S207C8.h.

**5.1.2.3708 STM8S207CB**

```
#define STM8S207CB
```

Definition at line 40 of file STM8S207CB.h.

**5.1.2.3709 STM8S207K6**

```
#define STM8S207K6
```

Definition at line 40 of file STM8S207K6.h.

**5.1.2.3710 STM8S207K8**

```
#define STM8S207K8
```

Definition at line 40 of file STM8S207K8.h.

**5.1.2.3711 STM8S207M8**

```
#define STM8S207M8
```

Definition at line 40 of file STM8S207M8.h.

**5.1.2.3712 STM8S207MB**

```
#define STM8S207MB
```

Definition at line 40 of file STM8S207MB.h.

**5.1.2.3713 STM8S207R6**

```
#define STM8S207R6
```

Definition at line 40 of file STM8S207R6.h.

**5.1.2.3714 STM8S207R8**

```
#define STM8S207R8
```

Definition at line 40 of file STM8S207R8.h.

**5.1.2.3715 STM8S207RB**

```
#define STM8S207RB
```

Definition at line 40 of file STM8S207RB.h.

**5.1.2.3716 STM8S207S6**

```
#define STM8S207S6
```

Definition at line 40 of file STM8S207S6.h.

**5.1.2.3717 STM8S207S8**

```
#define STM8S207S8
```

Definition at line 40 of file STM8S207S8.h.

**5.1.2.3718 STM8S207SB**

```
#define STM8S207SB
```

Definition at line 40 of file STM8S207SB.h.

**5.1.2.3719 STM8S208** [1/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208R8.h.

**5.1.2.3720 STM8S208** [2/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208C8.h.

**5.1.2.3721 STM8S208** [3/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208CB.h.

**5.1.2.3722 STM8S208** [4/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208R6.h.

**5.1.2.3723 STM8S208** [5/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208C6.h.

**5.1.2.3724 STM8S208** [6/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208RB.h.



**5.1.2.3725 STM8S208** [7/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208S8.h.

**5.1.2.3726 STM8S208** [8/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208SB.h.

**5.1.2.3727 STM8S208** [9/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208MB.h.

**5.1.2.3728 STM8S208** [10/10]

```
#define STM8S208
```

Definition at line 43 of file STM8S208S6.h.

**5.1.2.3729 STM8S208C6**

```
#define STM8S208C6
```

Definition at line 40 of file STM8S208C6.h.

**5.1.2.3730 STM8S208C8**

```
#define STM8S208C8
```

Definition at line 40 of file STM8S208C8.h.

**5.1.2.3731 STM8S208CB**

```
#define STM8S208CB
```

Definition at line 40 of file STM8S208CB.h.

**5.1.2.3732 STM8S208MB**

```
#define STM8S208MB
```

Definition at line 40 of file STM8S208MB.h.

**5.1.2.3733 STM8S208R6**

```
#define STM8S208R6
```

Definition at line 40 of file STM8S208R6.h.

**5.1.2.3734 STM8S208R8**

```
#define STM8S208R8
```

Definition at line 40 of file STM8S208R8.h.

**5.1.2.3735 STM8S208RB**

```
#define STM8S208RB
```

Definition at line 40 of file STM8S208RB.h.

**5.1.2.3736 STM8S208S6**

```
#define STM8S208S6
```

Definition at line 40 of file STM8S208S6.h.

**5.1.2.3737 STM8S208S8**

```
#define STM8S208S8
```

Definition at line 40 of file STM8S208S8.h.

**5.1.2.3738 STM8S208SB**

```
#define STM8S208SB
```

Definition at line 40 of file STM8S208SB.h.

**5.1.2.3739 STM8S903** [1/2]

```
#define STM8S903
```

Definition at line 43 of file STM8S903K3.h.

**5.1.2.3740 STM8S903** [2/2]

```
#define STM8S903
```

Definition at line 43 of file STM8S903F3.h.

**5.1.2.3741 STM8S903F3**

```
#define STM8S903F3
```

Definition at line 40 of file STM8S903F3.h.

**5.1.2.3742 STM8S903K3**

```
#define STM8S903K3
```

Definition at line 40 of file STM8S903K3.h.

**5.1.2.3743 SW\_RESET**

```
#define SW_RESET( ) (_WWDG_CR=0xBF)
```

reset controller via WWGD module

Definition at line 174 of file STM8AF\_STM8S.h.

**5.1.2.3744 TIM1\_AddressBase** [1/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S103K3.h.

**5.1.2.3745 TIM1\_AddressBase** [2/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8AF6226.h.

**5.1.2.3746 TIM1\_AddressBase** [3/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S001J3.h.

**5.1.2.3747 TIM1\_AddressBase** [4/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S003K3.h.

**5.1.2.3748 TIM1\_AddressBase** [5/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8AF6223.h.

**5.1.2.3749 TIM1\_AddressBase** [6/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8AF6213.h.

**5.1.2.3750 TIM1\_AddressBase** [7/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S103F2.h.

**5.1.2.3751 TIM1\_AddressBase** [8/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S903F3.h.

**5.1.2.3752 TIM1\_AddressBase** [9/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8AF6366.h.

**5.1.2.3753 TIM1\_AddressBase** [10/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S003F3.h.

**5.1.2.3754 TIM1\_AddressBase** [11/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8AF6223A.h.

**5.1.2.3755** TIM1\_AddressBase [12/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S903K3.h.

**5.1.2.3756** TIM1\_AddressBase [13/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8AF6213A.h.

**5.1.2.3757** TIM1\_AddressBase [14/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 70 of file STM8S103F3.h.

**5.1.2.3758** TIM1\_AddressBase [15/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S105S6.h.

**5.1.2.3759** TIM1\_AddressBase [16/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S005K6.h.

**5.1.2.3760** TIM1\_AddressBase [17/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8AF6246.h.

**5.1.2.3761** TIM1\_AddressBase [18/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S105C6.h.

**5.1.2.3762** TIM1\_AddressBase [19/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8AF6269.h.

**5.1.2.3763** TIM1\_AddressBase [20/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S005C6.h.

**5.1.2.3764** TIM1\_AddressBase [21/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8AF6248.h.

**5.1.2.3765** TIM1\_AddressBase [22/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8AF6268.h.

**5.1.2.3766** TIM1\_AddressBase [23/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S105S4.h.

**5.1.2.3767 TIM1\_AddressBase** [24/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8AF6266.h.

**5.1.2.3768 TIM1\_AddressBase** [25/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S105K4.h.

**5.1.2.3769 TIM1\_AddressBase** [26/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S105K6.h.

**5.1.2.3770 TIM1\_AddressBase** [27/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 71 of file STM8S105C4.h.

**5.1.2.3771 TIM1\_AddressBase** [28/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF628A.h.

**5.1.2.3772 TIM1\_AddressBase** [29/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF6286.h.



**5.1.2.3773 TIM1\_AddressBase** [30/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF62A9.h.

**5.1.2.3774 TIM1\_AddressBase** [31/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207C6.h.

**5.1.2.3775 TIM1\_AddressBase** [32/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207MB.h.

**5.1.2.3776 TIM1\_AddressBase** [33/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208C8.h.

**5.1.2.3777 TIM1\_AddressBase** [34/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207R6.h.

**5.1.2.3778 TIM1\_AddressBase** [35/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208SB.h.

**5.1.2.3779 TIM1\_AddressBase** [36/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF6289.h.

**5.1.2.3780 TIM1\_AddressBase** [37/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207CB.h.

**5.1.2.3781 TIM1\_AddressBase** [38/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF52AA.h.

**5.1.2.3782 TIM1\_AddressBase** [39/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208C6.h.

**5.1.2.3783 TIM1\_AddressBase** [40/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207M8.h.

**5.1.2.3784 TIM1\_AddressBase** [41/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF5268.h.

**5.1.2.3785** TIM1\_AddressBase [42/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207RB.h.

**5.1.2.3786** TIM1\_AddressBase [43/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF62A8.h.

**5.1.2.3787** TIM1\_AddressBase [44/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF5289.h.

**5.1.2.3788** TIM1\_AddressBase [45/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208RB.h.

**5.1.2.3789** TIM1\_AddressBase [46/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF52A6.h.

**5.1.2.3790** TIM1\_AddressBase [47/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208MB.h.

**5.1.2.3791** TIM1\_AddressBase [48/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207R8.h.

**5.1.2.3792** TIM1\_AddressBase [49/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF5288.h.

**5.1.2.3793** TIM1\_AddressBase [50/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208R6.h.

**5.1.2.3794** TIM1\_AddressBase [51/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207C8.h.

**5.1.2.3795** TIM1\_AddressBase [52/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208CB.h.

**5.1.2.3796** TIM1\_AddressBase [53/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207K6.h.

**5.1.2.3797 TIM1\_AddressBase** [54/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207S6.h.

**5.1.2.3798 TIM1\_AddressBase** [55/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF62AA.h.

**5.1.2.3799 TIM1\_AddressBase** [56/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208S6.h.

**5.1.2.3800 TIM1\_AddressBase** [57/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF52A9.h.

**5.1.2.3801 TIM1\_AddressBase** [58/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208R8.h.

**5.1.2.3802 TIM1\_AddressBase** [59/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S007C8.h.

**5.1.2.3803 TIM1\_AddressBase** [60/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207S8.h.

**5.1.2.3804 TIM1\_AddressBase** [61/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S208S8.h.

**5.1.2.3805 TIM1\_AddressBase** [62/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF6288.h.

**5.1.2.3806 TIM1\_AddressBase** [63/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF5269.h.

**5.1.2.3807 TIM1\_AddressBase** [64/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207SB.h.

**5.1.2.3808 TIM1\_AddressBase** [65/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8S207K8.h.

**5.1.2.3809 TIM1\_AddressBase** [66/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF6388.h.

**5.1.2.3810 TIM1\_AddressBase** [67/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF528A.h.

**5.1.2.3811 TIM1\_AddressBase** [68/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF52A8.h.

**5.1.2.3812 TIM1\_AddressBase** [69/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF62A6.h.

**5.1.2.3813 TIM1\_AddressBase** [70/70]

```
#define TIM1_AddressBase 0x5250
```

Definition at line 74 of file STM8AF5286.h.

**5.1.2.3814 TIM2\_AddressBase** [1/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 71 of file STM8S001J3.h.

**5.1.2.3815 TIM2\_AddressBase** [2/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 71 of file STM8S003F3.h.

**5.1.2.3816 TIM2\_AddressBase** [3/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 71 of file STM8S103F3.h.

**5.1.2.3817 TIM2\_AddressBase** [4/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 71 of file STM8AF6366.h.

**5.1.2.3818 TIM2\_AddressBase** [5/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 71 of file STM8S103F2.h.

**5.1.2.3819 TIM2\_AddressBase** [6/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 71 of file STM8S003K3.h.

**5.1.2.3820 TIM2\_AddressBase** [7/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 71 of file STM8S103K3.h.



**5.1.2.3821** TIM2\_AddressBase [8/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S005C6.h.

**5.1.2.3822** TIM2\_AddressBase [9/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S005K6.h.

**5.1.2.3823** TIM2\_AddressBase [10/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S105K6.h.

**5.1.2.3824** TIM2\_AddressBase [11/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8AF6266.h.

**5.1.2.3825** TIM2\_AddressBase [12/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8AF6246.h.

**5.1.2.3826** TIM2\_AddressBase [13/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S105S4.h.

**5.1.2.3827 TIM2\_AddressBase** [14/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S105S6.h.

**5.1.2.3828 TIM2\_AddressBase** [15/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8AF6269.h.

**5.1.2.3829 TIM2\_AddressBase** [16/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S105C6.h.

**5.1.2.3830 TIM2\_AddressBase** [17/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8AF6268.h.

**5.1.2.3831 TIM2\_AddressBase** [18/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S105C4.h.

**5.1.2.3832 TIM2\_AddressBase** [19/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8AF6248.h.

**5.1.2.3833** TIM2\_AddressBase [20/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 72 of file STM8S105K4.h.

**5.1.2.3834** TIM2\_AddressBase [21/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208MB.h.

**5.1.2.3835** TIM2\_AddressBase [22/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF5289.h.

**5.1.2.3836** TIM2\_AddressBase [23/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF6288.h.

**5.1.2.3837** TIM2\_AddressBase [24/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207S6.h.

**5.1.2.3838** TIM2\_AddressBase [25/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207SB.h.

**5.1.2.3839 TIM2\_AddressBase** [26/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207K8.h.

**5.1.2.3840 TIM2\_AddressBase** [27/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF52A8.h.

**5.1.2.3841 TIM2\_AddressBase** [28/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208R8.h.

**5.1.2.3842 TIM2\_AddressBase** [29/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208CB.h.

**5.1.2.3843 TIM2\_AddressBase** [30/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208SB.h.

**5.1.2.3844 TIM2\_AddressBase** [31/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF5269.h.

**5.1.2.3845** TIM2\_AddressBase [32/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF628A.h.

**5.1.2.3846** TIM2\_AddressBase [33/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF52A9.h.

**5.1.2.3847** TIM2\_AddressBase [34/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF6289.h.

**5.1.2.3848** TIM2\_AddressBase [35/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF6286.h.

**5.1.2.3849** TIM2\_AddressBase [36/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF52AA.h.

**5.1.2.3850** TIM2\_AddressBase [37/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207M8.h.

**5.1.2.3851 TIM2\_AddressBase** [38/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207MB.h.

**5.1.2.3852 TIM2\_AddressBase** [39/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207R8.h.

**5.1.2.3853 TIM2\_AddressBase** [40/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF6388.h.

**5.1.2.3854 TIM2\_AddressBase** [41/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207K6.h.

**5.1.2.3855 TIM2\_AddressBase** [42/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF62AA.h.

**5.1.2.3856 TIM2\_AddressBase** [43/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207C6.h.

**5.1.2.3857** TIM2\_AddressBase [44/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207R6.h.

**5.1.2.3858** TIM2\_AddressBase [45/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF62A9.h.

**5.1.2.3859** TIM2\_AddressBase [46/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF5286.h.

**5.1.2.3860** TIM2\_AddressBase [47/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208C8.h.

**5.1.2.3861** TIM2\_AddressBase [48/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF528A.h.

**5.1.2.3862** TIM2\_AddressBase [49/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208R6.h.

**5.1.2.3863** **TIM2\_AddressBase** [50/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF5268.h.

**5.1.2.3864** **TIM2\_AddressBase** [51/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208RB.h.

**5.1.2.3865** **TIM2\_AddressBase** [52/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF52A6.h.

**5.1.2.3866** **TIM2\_AddressBase** [53/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208S6.h.

**5.1.2.3867** **TIM2\_AddressBase** [54/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF62A8.h.

**5.1.2.3868** **TIM2\_AddressBase** [55/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207RB.h.



**5.1.2.3869 TIM2\_AddressBase** [56/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S007C8.h.

**5.1.2.3870 TIM2\_AddressBase** [57/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208S8.h.

**5.1.2.3871 TIM2\_AddressBase** [58/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S208C6.h.

**5.1.2.3872 TIM2\_AddressBase** [59/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207C8.h.

**5.1.2.3873 TIM2\_AddressBase** [60/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF62A6.h.

**5.1.2.3874 TIM2\_AddressBase** [61/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207S8.h.

**5.1.2.3875 TIM2\_AddressBase** [62/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8AF5288.h.

**5.1.2.3876 TIM2\_AddressBase** [63/63]

```
#define TIM2_AddressBase 0x5300
```

Definition at line 75 of file STM8S207CB.h.

**5.1.2.3877 TIM3\_AddressBase** [1/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 72 of file STM8AF6366.h.

**5.1.2.3878 TIM3\_AddressBase** [2/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8AF6269.h.

**5.1.2.3879 TIM3\_AddressBase** [3/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S005K6.h.

**5.1.2.3880 TIM3\_AddressBase** [4/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S105C6.h.

**5.1.2.3881** TIM3\_AddressBase [5/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8AF6246.h.

**5.1.2.3882** TIM3\_AddressBase [6/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8AF6268.h.

**5.1.2.3883** TIM3\_AddressBase [7/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S105K4.h.

**5.1.2.3884** TIM3\_AddressBase [8/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S105S4.h.

**5.1.2.3885** TIM3\_AddressBase [9/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S105C4.h.

**5.1.2.3886** TIM3\_AddressBase [10/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S105S6.h.

**5.1.2.3887 TIM3\_AddressBase** [11/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8AF6248.h.

**5.1.2.3888 TIM3\_AddressBase** [12/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8AF6266.h.

**5.1.2.3889 TIM3\_AddressBase** [13/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S005C6.h.

**5.1.2.3890 TIM3\_AddressBase** [14/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 73 of file STM8S105K6.h.

**5.1.2.3891 TIM3\_AddressBase** [15/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF628A.h.

**5.1.2.3892 TIM3\_AddressBase** [16/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF6286.h.

**5.1.2.3893** TIM3\_AddressBase [17/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF62A8.h.

**5.1.2.3894** TIM3\_AddressBase [18/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S007C8.h.

**5.1.2.3895** TIM3\_AddressBase [19/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208C6.h.

**5.1.2.3896** TIM3\_AddressBase [20/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208C8.h.

**5.1.2.3897** TIM3\_AddressBase [21/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207MB.h.

**5.1.2.3898** TIM3\_AddressBase [22/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF5289.h.

**5.1.2.3899 TIM3\_AddressBase** [23/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207C8.h.

**5.1.2.3900 TIM3\_AddressBase** [24/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208MB.h.

**5.1.2.3901 TIM3\_AddressBase** [25/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF5286.h.

**5.1.2.3902 TIM3\_AddressBase** [26/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF5269.h.

**5.1.2.3903 TIM3\_AddressBase** [27/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207SB.h.

**5.1.2.3904 TIM3\_AddressBase** [28/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF52A6.h.

**5.1.2.3905 TIM3\_AddressBase** [29/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208S8.h.

**5.1.2.3906 TIM3\_AddressBase** [30/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207RB.h.

**5.1.2.3907 TIM3\_AddressBase** [31/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207R6.h.

**5.1.2.3908 TIM3\_AddressBase** [32/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF52A8.h.

**5.1.2.3909 TIM3\_AddressBase** [33/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207K6.h.

**5.1.2.3910 TIM3\_AddressBase** [34/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207M8.h.

**5.1.2.3911 TIM3\_AddressBase** [35/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208SB.h.

**5.1.2.3912 TIM3\_AddressBase** [36/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF5268.h.

**5.1.2.3913 TIM3\_AddressBase** [37/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208R8.h.

**5.1.2.3914 TIM3\_AddressBase** [38/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208S6.h.

**5.1.2.3915 TIM3\_AddressBase** [39/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207K8.h.

**5.1.2.3916 TIM3\_AddressBase** [40/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF5288.h.



**5.1.2.3917 TIM3\_AddressBase** [41/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207C6.h.

**5.1.2.3918 TIM3\_AddressBase** [42/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208CB.h.

**5.1.2.3919 TIM3\_AddressBase** [43/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208R6.h.

**5.1.2.3920 TIM3\_AddressBase** [44/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207CB.h.

**5.1.2.3921 TIM3\_AddressBase** [45/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207S8.h.

**5.1.2.3922 TIM3\_AddressBase** [46/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF62AA.h.

**5.1.2.3923** **TIM3\_AddressBase** [47/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF62A9.h.

**5.1.2.3924** **TIM3\_AddressBase** [48/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF6388.h.

**5.1.2.3925** **TIM3\_AddressBase** [49/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF6288.h.

**5.1.2.3926** **TIM3\_AddressBase** [50/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S208RB.h.

**5.1.2.3927** **TIM3\_AddressBase** [51/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207R8.h.

**5.1.2.3928** **TIM3\_AddressBase** [52/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8S207S6.h.

**5.1.2.3929 TIM3\_AddressBase** [53/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF52AA.h.

**5.1.2.3930 TIM3\_AddressBase** [54/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF528A.h.

**5.1.2.3931 TIM3\_AddressBase** [55/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF52A9.h.

**5.1.2.3932 TIM3\_AddressBase** [56/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF62A6.h.

**5.1.2.3933 TIM3\_AddressBase** [57/57]

```
#define TIM3_AddressBase 0x5320
```

Definition at line 76 of file STM8AF6289.h.

**5.1.2.3934 TIM4\_AddressBase** [1/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 72 of file STM8S003F3.h.

**5.1.2.3935 TIM4\_AddressBase** [2/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 72 of file STM8S103K3.h.

**5.1.2.3936 TIM4\_AddressBase** [3/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 72 of file STM8S003K3.h.

**5.1.2.3937 TIM4\_AddressBase** [4/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 72 of file STM8S103F2.h.

**5.1.2.3938 TIM4\_AddressBase** [5/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 72 of file STM8S001J3.h.

**5.1.2.3939 TIM4\_AddressBase** [6/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 72 of file STM8S103F3.h.

**5.1.2.3940 TIM4\_AddressBase** [7/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 73 of file STM8AF6366.h.

**5.1.2.3941** TIM4\_AddressBase [8/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S005C6.h.

**5.1.2.3942** TIM4\_AddressBase [9/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8AF6268.h.

**5.1.2.3943** TIM4\_AddressBase [10/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8AF6266.h.

**5.1.2.3944** TIM4\_AddressBase [11/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S105K4.h.

**5.1.2.3945** TIM4\_AddressBase [12/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8AF6269.h.

**5.1.2.3946** TIM4\_AddressBase [13/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S105S6.h.

**5.1.2.3947 TIM4\_AddressBase** [14/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S005K6.h.

**5.1.2.3948 TIM4\_AddressBase** [15/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S105K6.h.

**5.1.2.3949 TIM4\_AddressBase** [16/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S105C6.h.

**5.1.2.3950 TIM4\_AddressBase** [17/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8AF6248.h.

**5.1.2.3951 TIM4\_AddressBase** [18/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8AF6246.h.

**5.1.2.3952 TIM4\_AddressBase** [19/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S105C4.h.

**5.1.2.3953** TIM4\_AddressBase [20/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 74 of file STM8S105S4.h.

**5.1.2.3954** TIM4\_AddressBase [21/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207SB.h.

**5.1.2.3955** TIM4\_AddressBase [22/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208CB.h.

**5.1.2.3956** TIM4\_AddressBase [23/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF62AA.h.

**5.1.2.3957** TIM4\_AddressBase [24/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207S6.h.

**5.1.2.3958** TIM4\_AddressBase [25/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF62A9.h.

**5.1.2.3959 TIM4\_AddressBase** [26/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF5286.h.

**5.1.2.3960 TIM4\_AddressBase** [27/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207K6.h.

**5.1.2.3961 TIM4\_AddressBase** [28/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207R8.h.

**5.1.2.3962 TIM4\_AddressBase** [29/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208MB.h.

**5.1.2.3963 TIM4\_AddressBase** [30/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF528A.h.

**5.1.2.3964 TIM4\_AddressBase** [31/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207RB.h.



**5.1.2.3965** TIM4\_AddressBase [32/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF52A8.h.

**5.1.2.3966** TIM4\_AddressBase [33/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208C8.h.

**5.1.2.3967** TIM4\_AddressBase [34/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207CB.h.

**5.1.2.3968** TIM4\_AddressBase [35/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207R6.h.

**5.1.2.3969** TIM4\_AddressBase [36/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF5289.h.

**5.1.2.3970** TIM4\_AddressBase [37/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF6289.h.

**5.1.2.3971 TIM4\_AddressBase** [38/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208R6.h.

**5.1.2.3972 TIM4\_AddressBase** [39/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207MB.h.

**5.1.2.3973 TIM4\_AddressBase** [40/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF5269.h.

**5.1.2.3974 TIM4\_AddressBase** [41/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF52A6.h.

**5.1.2.3975 TIM4\_AddressBase** [42/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207S8.h.

**5.1.2.3976 TIM4\_AddressBase** [43/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF62A8.h.

**5.1.2.3977 TIM4\_AddressBase** [44/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S007C8.h.

**5.1.2.3978 TIM4\_AddressBase** [45/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF628A.h.

**5.1.2.3979 TIM4\_AddressBase** [46/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207C6.h.

**5.1.2.3980 TIM4\_AddressBase** [47/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208R8.h.

**5.1.2.3981 TIM4\_AddressBase** [48/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208RB.h.

**5.1.2.3982 TIM4\_AddressBase** [49/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF62A6.h.

**5.1.2.3983** TIM4\_AddressBase [50/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207M8.h.

**5.1.2.3984** TIM4\_AddressBase [51/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207K8.h.

**5.1.2.3985** TIM4\_AddressBase [52/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208S8.h.

**5.1.2.3986** TIM4\_AddressBase [53/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF52AA.h.

**5.1.2.3987** TIM4\_AddressBase [54/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF6286.h.

**5.1.2.3988** TIM4\_AddressBase [55/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208C6.h.

**5.1.2.3989 TIM4\_AddressBase** [56/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF6388.h.

**5.1.2.3990 TIM4\_AddressBase** [57/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208SB.h.

**5.1.2.3991 TIM4\_AddressBase** [58/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S208S6.h.

**5.1.2.3992 TIM4\_AddressBase** [59/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF5268.h.

**5.1.2.3993 TIM4\_AddressBase** [60/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8S207C8.h.

**5.1.2.3994 TIM4\_AddressBase** [61/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF52A9.h.

**5.1.2.3995** TIM4\_AddressBase [62/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF5288.h.

**5.1.2.3996** TIM4\_AddressBase [63/63]

```
#define TIM4_AddressBase 0x5340
```

Definition at line 77 of file STM8AF6288.h.

**5.1.2.3997** TIM5\_AddressBase [1/7]

```
#define TIM5_AddressBase 0x5300
```

Definition at line 71 of file STM8AF6213A.h.

**5.1.2.3998** TIM5\_AddressBase [2/7]

```
#define TIM5_AddressBase 0x5300
```

Definition at line 71 of file STM8AF6223.h.

**5.1.2.3999** TIM5\_AddressBase [3/7]

```
#define TIM5_AddressBase 0x5300
```

Definition at line 71 of file STM8S903F3.h.

**5.1.2.4000** TIM5\_AddressBase [4/7]

```
#define TIM5_AddressBase 0x5300
```

Definition at line 71 of file STM8AF6223A.h.

**5.1.2.4001** TIM5\_AddressBase [5/7]

```
#define TIM5_AddressBase 0x5300
```

Definition at line 71 of file STM8S903K3.h.

**5.1.2.4002** TIM5\_AddressBase [6/7]

```
#define TIM5_AddressBase 0x5300
```

Definition at line 71 of file STM8AF6226.h.

**5.1.2.4003** TIM5\_AddressBase [7/7]

```
#define TIM5_AddressBase 0x5300
```

Definition at line 71 of file STM8AF6213.h.

**5.1.2.4004** TIM6\_AddressBase [1/7]

```
#define TIM6_AddressBase 0x5340
```

Definition at line 72 of file STM8S903K3.h.

**5.1.2.4005** TIM6\_AddressBase [2/7]

```
#define TIM6_AddressBase 0x5340
```

Definition at line 72 of file STM8AF6213.h.

**5.1.2.4006** TIM6\_AddressBase [3/7]

```
#define TIM6_AddressBase 0x5340
```

Definition at line 72 of file STM8AF6226.h.

**5.1.2.4007 TIM6\_AddressBase** [4/7]

```
#define TIM6_AddressBase 0x5340
```

Definition at line 72 of file STM8AF6213A.h.

**5.1.2.4008 TIM6\_AddressBase** [5/7]

```
#define TIM6_AddressBase 0x5340
```

Definition at line 72 of file STM8S903F3.h.

**5.1.2.4009 TIM6\_AddressBase** [6/7]

```
#define TIM6_AddressBase 0x5340
```

Definition at line 72 of file STM8AF6223A.h.

**5.1.2.4010 TIM6\_AddressBase** [7/7]

```
#define TIM6_AddressBase 0x5340
```

Definition at line 72 of file STM8AF6223.h.

**5.1.2.4011 TRIGGER\_TRAP**

```
#define TRIGGER_TRAP __asm__("trap")
```

trigger a trap (=soft interrupt) e.g. for EMC robustness (see AN1015)

Definition at line 171 of file STM8AF\_STM8S.h.

**5.1.2.4012 UART1\_AddressBase** [1/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S903K3.h.



**5.1.2.4013** UART1\_AddressBase [2/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S103F3.h.

**5.1.2.4014** UART1\_AddressBase [3/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S103F2.h.

**5.1.2.4015** UART1\_AddressBase [4/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S903F3.h.

**5.1.2.4016** UART1\_AddressBase [5/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S003K3.h.

**5.1.2.4017** UART1\_AddressBase [6/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S001J3.h.

**5.1.2.4018** UART1\_AddressBase [7/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S003F3.h.

**5.1.2.4019** **UART1\_AddressBase** [8/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 69 of file STM8S103K3.h.

**5.1.2.4020** **UART1\_AddressBase** [9/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF52A9.h.

**5.1.2.4021** **UART1\_AddressBase** [10/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF5269.h.

**5.1.2.4022** **UART1\_AddressBase** [11/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207C8.h.

**5.1.2.4023** **UART1\_AddressBase** [12/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207C6.h.

**5.1.2.4024** **UART1\_AddressBase** [13/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF62A9.h.

**5.1.2.4025** UART1\_AddressBase [14/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208R6.h.

**5.1.2.4026** UART1\_AddressBase [15/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208SB.h.

**5.1.2.4027** UART1\_AddressBase [16/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF62A8.h.

**5.1.2.4028** UART1\_AddressBase [17/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208R8.h.

**5.1.2.4029** UART1\_AddressBase [18/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207MB.h.

**5.1.2.4030** UART1\_AddressBase [19/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208S6.h.

**5.1.2.4031** **UART1\_AddressBase** [20/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF5268.h.

**5.1.2.4032** **UART1\_AddressBase** [21/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF6289.h.

**5.1.2.4033** **UART1\_AddressBase** [22/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208CB.h.

**5.1.2.4034** **UART1\_AddressBase** [23/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF5289.h.

**5.1.2.4035** **UART1\_AddressBase** [24/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207S6.h.

**5.1.2.4036** **UART1\_AddressBase** [25/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207S8.h.

**5.1.2.4037** UART1\_AddressBase [26/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208C6.h.

**5.1.2.4038** UART1\_AddressBase [27/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207CB.h.

**5.1.2.4039** UART1\_AddressBase [28/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF52A6.h.

**5.1.2.4040** UART1\_AddressBase [29/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF6286.h.

**5.1.2.4041** UART1\_AddressBase [30/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208MB.h.

**5.1.2.4042** UART1\_AddressBase [31/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208RB.h.

**5.1.2.4043** **UART1\_AddressBase** [32/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207SB.h.

**5.1.2.4044** **UART1\_AddressBase** [33/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207RB.h.

**5.1.2.4045** **UART1\_AddressBase** [34/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207K8.h.

**5.1.2.4046** **UART1\_AddressBase** [35/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF62AA.h.

**5.1.2.4047** **UART1\_AddressBase** [36/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF528A.h.

**5.1.2.4048** **UART1\_AddressBase** [37/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208S8.h.

**5.1.2.4049** UART1\_AddressBase [38/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S007C8.h.

**5.1.2.4050** UART1\_AddressBase [39/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF52AA.h.

**5.1.2.4051** UART1\_AddressBase [40/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF62A6.h.

**5.1.2.4052** UART1\_AddressBase [41/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207R6.h.

**5.1.2.4053** UART1\_AddressBase [42/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S208C8.h.

**5.1.2.4054** UART1\_AddressBase [43/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF5286.h.

**5.1.2.4055** **UART1\_AddressBase** [44/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF52A8.h.

**5.1.2.4056** **UART1\_AddressBase** [45/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF628A.h.

**5.1.2.4057** **UART1\_AddressBase** [46/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF6388.h.

**5.1.2.4058** **UART1\_AddressBase** [47/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207R8.h.

**5.1.2.4059** **UART1\_AddressBase** [48/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207K6.h.

**5.1.2.4060** **UART1\_AddressBase** [49/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF5288.h.



**5.1.2.4061** UART1\_AddressBase [50/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8AF6288.h.

**5.1.2.4062** UART1\_AddressBase [51/51]

```
#define UART1_AddressBase 0x5230
```

Definition at line 72 of file STM8S207M8.h.

**5.1.2.4063** UART2\_AddressBase [1/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 69 of file STM8AF6366.h.

**5.1.2.4064** UART2\_AddressBase [2/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S105K4.h.

**5.1.2.4065** UART2\_AddressBase [3/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S105K6.h.

**5.1.2.4066** UART2\_AddressBase [4/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8AF6248.h.

**5.1.2.4067** **UART2\_AddressBase** [5/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8AF6268.h.

**5.1.2.4068** **UART2\_AddressBase** [6/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S105C6.h.

**5.1.2.4069** **UART2\_AddressBase** [7/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S005C6.h.

**5.1.2.4070** **UART2\_AddressBase** [8/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S105C4.h.

**5.1.2.4071** **UART2\_AddressBase** [9/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8AF6266.h.

**5.1.2.4072** **UART2\_AddressBase** [10/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S105S4.h.

**5.1.2.4073** **UART2\_AddressBase** [11/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8AF6269.h.

**5.1.2.4074** **UART2\_AddressBase** [12/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S105S6.h.

**5.1.2.4075** **UART2\_AddressBase** [13/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8AF6246.h.

**5.1.2.4076** **UART2\_AddressBase** [14/14]

```
#define UART2_AddressBase 0x5240
```

Definition at line 70 of file STM8S005K6.h.

**5.1.2.4077** **UART3\_AddressBase** [1/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207CB.h.

**5.1.2.4078** **UART3\_AddressBase** [2/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207S8.h.

**5.1.2.4079** **UART3\_AddressBase** [3/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF52A8.h.

**5.1.2.4080** **UART3\_AddressBase** [4/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF6289.h.

**5.1.2.4081** **UART3\_AddressBase** [5/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S007C8.h.

**5.1.2.4082** **UART3\_AddressBase** [6/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF62A9.h.

**5.1.2.4083** **UART3\_AddressBase** [7/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF62A6.h.

**5.1.2.4084** **UART3\_AddressBase** [8/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF5286.h.

**5.1.2.4085** UART3\_AddressBase [9/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207MB.h.

**5.1.2.4086** UART3\_AddressBase [10/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208S8.h.

**5.1.2.4087** UART3\_AddressBase [11/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF5288.h.

**5.1.2.4088** UART3\_AddressBase [12/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF6288.h.

**5.1.2.4089** UART3\_AddressBase [13/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF628A.h.

**5.1.2.4090** UART3\_AddressBase [14/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207K6.h.

**5.1.2.4091** **UART3\_AddressBase** [15/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF5289.h.

**5.1.2.4092** **UART3\_AddressBase** [16/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208CB.h.

**5.1.2.4093** **UART3\_AddressBase** [17/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207R8.h.

**5.1.2.4094** **UART3\_AddressBase** [18/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF6286.h.

**5.1.2.4095** **UART3\_AddressBase** [19/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208R8.h.

**5.1.2.4096** **UART3\_AddressBase** [20/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF5268.h.

**5.1.2.4097** UART3\_AddressBase [21/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208C6.h.

**5.1.2.4098** UART3\_AddressBase [22/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF52A9.h.

**5.1.2.4099** UART3\_AddressBase [23/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208SB.h.

**5.1.2.4100** UART3\_AddressBase [24/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF5269.h.

**5.1.2.4101** UART3\_AddressBase [25/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208R6.h.

**5.1.2.4102** UART3\_AddressBase [26/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208S6.h.

**5.1.2.4103** **UART3\_AddressBase** [27/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF62A8.h.

**5.1.2.4104** **UART3\_AddressBase** [28/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208MB.h.

**5.1.2.4105** **UART3\_AddressBase** [29/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207C8.h.

**5.1.2.4106** **UART3\_AddressBase** [30/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207M8.h.

**5.1.2.4107** **UART3\_AddressBase** [31/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207RB.h.

**5.1.2.4108** **UART3\_AddressBase** [32/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207S6.h.



**5.1.2.4109** UART3\_AddressBase [33/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208C8.h.

**5.1.2.4110** UART3\_AddressBase [34/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF52A6.h.

**5.1.2.4111** UART3\_AddressBase [35/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207C6.h.

**5.1.2.4112** UART3\_AddressBase [36/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207R6.h.

**5.1.2.4113** UART3\_AddressBase [37/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S208RB.h.

**5.1.2.4114** UART3\_AddressBase [38/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207K8.h.

**5.1.2.4115** **UART3\_AddressBase** [39/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF6388.h.

**5.1.2.4116** **UART3\_AddressBase** [40/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF528A.h.

**5.1.2.4117** **UART3\_AddressBase** [41/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF62AA.h.

**5.1.2.4118** **UART3\_AddressBase** [42/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8AF52AA.h.

**5.1.2.4119** **UART3\_AddressBase** [43/43]

```
#define UART3_AddressBase 0x5240
```

Definition at line 73 of file STM8S207SB.h.

**5.1.2.4120** **UART4\_AddressBase** [1/5]

```
#define UART4_AddressBase 0x5230
```

Definition at line 69 of file STM8AF6226.h.

**5.1.2.4121** UART4\_AddressBase [2/5]

```
#define UART4_AddressBase 0x5230
```

Definition at line 69 of file STM8AF6213.h.

**5.1.2.4122** UART4\_AddressBase [3/5]

```
#define UART4_AddressBase 0x5230
```

Definition at line 69 of file STM8AF6223.h.

**5.1.2.4123** UART4\_AddressBase [4/5]

```
#define UART4_AddressBase 0x5230
```

Definition at line 69 of file STM8AF6213A.h.

**5.1.2.4124** UART4\_AddressBase [5/5]

```
#define UART4_AddressBase 0x5230
```

Definition at line 69 of file STM8AF6223A.h.

**5.1.2.4125** UID\_AddressBase [1/34]

```
#define UID_AddressBase 0x4865
```

Definition at line 77 of file STM8S903F3.h.

**5.1.2.4126** UID\_AddressBase [2/34]

```
#define UID_AddressBase 0x4865
```

Definition at line 77 of file STM8S103K3.h.

**5.1.2.4127 UID\_AddressBase** [3/34]

```
#define UID_AddressBase 0x4865
```

Definition at line 77 of file STM8S103F2.h.

**5.1.2.4128 UID\_AddressBase** [4/34]

```
#define UID_AddressBase 0x4865
```

Definition at line 77 of file STM8S103F3.h.

**5.1.2.4129 UID\_AddressBase** [5/34]

```
#define UID_AddressBase 0x4865
```

Definition at line 77 of file STM8S903K3.h.

**5.1.2.4130 UID\_AddressBase** [6/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 79 of file STM8S105S4.h.

**5.1.2.4131 UID\_AddressBase** [7/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 79 of file STM8S105C6.h.

**5.1.2.4132 UID\_AddressBase** [8/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 79 of file STM8S105K6.h.

**5.1.2.4133**    **UID\_AddressBase** [9/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 79 of file STM8S105K4.h.

**5.1.2.4134**    **UID\_AddressBase** [10/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 79 of file STM8S105S6.h.

**5.1.2.4135**    **UID\_AddressBase** [11/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 79 of file STM8S105C4.h.

**5.1.2.4136**    **UID\_AddressBase** [12/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207R8.h.

**5.1.2.4137**    **UID\_AddressBase** [13/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207R6.h.

**5.1.2.4138**    **UID\_AddressBase** [14/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207C6.h.

**5.1.2.4139 UID\_AddressBase** [15/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207M8.h.

**5.1.2.4140 UID\_AddressBase** [16/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207SB.h.

**5.1.2.4141 UID\_AddressBase** [17/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207S8.h.

**5.1.2.4142 UID\_AddressBase** [18/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207K8.h.

**5.1.2.4143 UID\_AddressBase** [19/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207RB.h.

**5.1.2.4144 UID\_AddressBase** [20/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207CB.h.

**5.1.2.4145** UID\_AddressBase [21/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207K6.h.

**5.1.2.4146** UID\_AddressBase [22/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207C8.h.

**5.1.2.4147** UID\_AddressBase [23/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207MB.h.

**5.1.2.4148** UID\_AddressBase [24/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 82 of file STM8S207S6.h.

**5.1.2.4149** UID\_AddressBase [25/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208MB.h.

**5.1.2.4150** UID\_AddressBase [26/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208CB.h.

**5.1.2.4151 UID\_AddressBase** [27/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208C6.h.

**5.1.2.4152 UID\_AddressBase** [28/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208C8.h.

**5.1.2.4153 UID\_AddressBase** [29/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208S8.h.

**5.1.2.4154 UID\_AddressBase** [30/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208RB.h.

**5.1.2.4155 UID\_AddressBase** [31/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208R6.h.

**5.1.2.4156 UID\_AddressBase** [32/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208SB.h.



**5.1.2.4157 UID\_AddressBase** [33/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208S6.h.

**5.1.2.4158 UID\_AddressBase** [34/34]

```
#define UID_AddressBase 0x48CD
```

Definition at line 83 of file STM8S208R8.h.

**5.1.2.4159 WAIT\_FOR\_INTERRUPT**

```
#define WAIT_FOR_INTERRUPT( ) __asm__("wfi")
```

stop code execution and wait for interrupt

Definition at line 172 of file STM8AF\_STM8S.h.

**5.1.2.4160 WWDG\_AddressBase** [1/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8AF6213A.h.

**5.1.2.4161 WWDG\_AddressBase** [2/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8AF6226.h.

**5.1.2.4162 WWDG\_AddressBase** [3/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8AF6223A.h.

**5.1.2.4163 WWDG\_AddressBase** [4/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S003K3.h.

**5.1.2.4164 WWDG\_AddressBase** [5/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S001J3.h.

**5.1.2.4165 WWDG\_AddressBase** [6/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8AF6366.h.

**5.1.2.4166 WWDG\_AddressBase** [7/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S103F2.h.

**5.1.2.4167 WWDG\_AddressBase** [8/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S903F3.h.

**5.1.2.4168 WWDG\_AddressBase** [9/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8AF6223.h.

**5.1.2.4169 WWDG\_AddressBase** [10/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S903K3.h.

**5.1.2.4170 WWDG\_AddressBase** [11/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8AF6213.h.

**5.1.2.4171 WWDG\_AddressBase** [12/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S103F3.h.

**5.1.2.4172 WWDG\_AddressBase** [13/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S003F3.h.

**5.1.2.4173 WWDG\_AddressBase** [14/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 63 of file STM8S103K3.h.

**5.1.2.4174 WWDG\_AddressBase** [15/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S005C6.h.

**5.1.2.4175 WWDG\_AddressBase** [16/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S105C6.h.

**5.1.2.4176 WWDG\_AddressBase** [17/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8AF6246.h.

**5.1.2.4177 WWDG\_AddressBase** [18/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S005K6.h.

**5.1.2.4178 WWDG\_AddressBase** [19/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S105S6.h.

**5.1.2.4179 WWDG\_AddressBase** [20/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8AF6269.h.

**5.1.2.4180 WWDG\_AddressBase** [21/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S105K6.h.

**5.1.2.4181 WWDG\_AddressBase** [22/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S105S4.h.

**5.1.2.4182 WWDG\_AddressBase** [23/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S105C4.h.

**5.1.2.4183 WWDG\_AddressBase** [24/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8AF6268.h.

**5.1.2.4184 WWDG\_AddressBase** [25/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8AF6248.h.

**5.1.2.4185 WWDG\_AddressBase** [26/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8S105K4.h.

**5.1.2.4186 WWDG\_AddressBase** [27/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 64 of file STM8AF6266.h.

**5.1.2.4187 WWDG\_AddressBase** [28/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207C6.h.

**5.1.2.4188 WWDG\_AddressBase** [29/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF5268.h.

**5.1.2.4189 WWDG\_AddressBase** [30/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207S8.h.

**5.1.2.4190 WWDG\_AddressBase** [31/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208R6.h.

**5.1.2.4191 WWDG\_AddressBase** [32/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF6288.h.

**5.1.2.4192 WWDG\_AddressBase** [33/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207S6.h.

**5.1.2.4193 WWDG\_AddressBase** [34/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207MB.h.

**5.1.2.4194 WWDG\_AddressBase** [35/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207SB.h.

**5.1.2.4195 WWDG\_AddressBase** [36/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208C8.h.

**5.1.2.4196 WWDG\_AddressBase** [37/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF62A8.h.

**5.1.2.4197 WWDG\_AddressBase** [38/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207CB.h.

**5.1.2.4198 WWDG\_AddressBase** [39/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF5286.h.

**5.1.2.4199 WWDG\_AddressBase** [40/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF62AA.h.

**5.1.2.4200 WWDG\_AddressBase** [41/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S007C8.h.

**5.1.2.4201 WWDG\_AddressBase** [42/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF528A.h.

**5.1.2.4202 WWDG\_AddressBase** [43/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF52A8.h.

**5.1.2.4203 WWDG\_AddressBase** [44/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207R6.h.

**5.1.2.4204 WWDG\_AddressBase** [45/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208CB.h.



**5.1.2.4205 WWDG\_AddressBase** [46/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF6388.h.

**5.1.2.4206 WWDG\_AddressBase** [47/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207M8.h.

**5.1.2.4207 WWDG\_AddressBase** [48/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF6289.h.

**5.1.2.4208 WWDG\_AddressBase** [49/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208MB.h.

**5.1.2.4209 WWDG\_AddressBase** [50/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF62A9.h.

**5.1.2.4210 WWDG\_AddressBase** [51/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207RB.h.

**5.1.2.4211 WWDG\_AddressBase** [52/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208R8.h.

**5.1.2.4212 WWDG\_AddressBase** [53/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF5288.h.

**5.1.2.4213 WWDG\_AddressBase** [54/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208SB.h.

**5.1.2.4214 WWDG\_AddressBase** [55/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF628A.h.

**5.1.2.4215 WWDG\_AddressBase** [56/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207K6.h.

**5.1.2.4216 WWDG\_AddressBase** [57/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF52A6.h.

**5.1.2.4217 WWDG\_AddressBase** [58/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF5289.h.

**5.1.2.4218 WWDG\_AddressBase** [59/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208S6.h.

**5.1.2.4219 WWDG\_AddressBase** [60/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF5269.h.

**5.1.2.4220 WWDG\_AddressBase** [61/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207K8.h.

**5.1.2.4221 WWDG\_AddressBase** [62/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207R8.h.

**5.1.2.4222 WWDG\_AddressBase** [63/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF6286.h.

**5.1.2.4223 WWDG\_AddressBase** [64/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF52AA.h.

**5.1.2.4224 WWDG\_AddressBase** [65/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S207C8.h.

**5.1.2.4225 WWDG\_AddressBase** [66/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208S8.h.

**5.1.2.4226 WWDG\_AddressBase** [67/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF62A6.h.

**5.1.2.4227 WWDG\_AddressBase** [68/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208C6.h.

**5.1.2.4228 WWDG\_AddressBase** [69/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8S208RB.h.

**5.1.2.4229 WWDG\_AddressBase** [70/70]

```
#define WWDG_AddressBase 0x50D1
```

Definition at line 66 of file STM8AF52A9.h.

## Chapter 6

# Data Structure Documentation

### 6.1 ADC1\_t Struct Reference

struct containing Analog Digital Converter 1 (ADC1)

```
#include <STM8AF_STM8S.h>
```

#### Data Fields

- struct {
  - [\\_BITS DATA: 2](#)  
*Data buffer 0 value [9:8].*
  - [\\_BITS res: 6](#)  
*reserved*
- } [DB0RH](#)  
  
*ADC1 10-bit Data Buffer Register 0 (ADC1\_DB0RH)*
- struct {
  - [\\_BITS DATA: 8](#)  
*Data buffer 0 value (low)*
- } [DB0RL](#)  
  
*ADC1 10-bit Data Buffer Register 0 (ADC1\_DB0RL)*
- struct {
  - [\\_BITS DATA: 2](#)  
*Data buffer 1 value [9:8].*
  - [\\_BITS res: 6](#)  
*reserved*
- } [DB1RH](#)  
  
*ADC1 10-bit Data Buffer Register 1 (ADC1\_DB1RH)*
- struct {
  - [\\_BITS DATA: 8](#)  
*Data buffer 1 value (low)*
- } [DB1RL](#)  
  
*ADC1 10-bit Data Buffer Register 1 (ADC1\_DB1RL)*

```

• struct {
    _BITS DATA: 2
    Data buffer 2 value [9:8].
    _BITS res: 6
    reserved
} DB2RH

ADC1 10-bit Data Buffer Register 2 (ADC1_DB2RH)

• struct {
    _BITS DATA: 8
    Data buffer 2 value (low)
} DB2RL

ADC1 10-bit Data Buffer Register 2 (ADC1_DB2RL)

• struct {
    _BITS DATA: 2
    Data buffer 3 value [9:8].
    _BITS res: 6
    reserved
} DB3RH

ADC1 10-bit Data Buffer Register 3 (ADC1_DB3RH)

• struct {
    _BITS DATA: 8
    Data buffer 3 value (low)
} DB3RL

ADC1 10-bit Data Buffer Register 3 (ADC1_DB3RL)

• struct {
    _BITS DATA: 2
    Data buffer 4 value [9:8].
    _BITS res: 6
    reserved
} DB4RH

ADC1 10-bit Data Buffer Register 4 (ADC1_DB4RH)

• struct {
    _BITS DATA: 8
    Data buffer 4 value (low)
} DB4RL

ADC1 10-bit Data Buffer Register 4 (ADC1_DB4RL)

• struct {
    _BITS DATA: 2
    Data buffer 5 value [9:8].
    _BITS res: 6
    reserved
} DB5RH

ADC1 10-bit Data Buffer Register 5 (ADC1_DB5RH)

• struct {
    _BITS DATA: 8
    Data buffer 5 value (low)
} DB5RL

ADC1 10-bit Data Buffer Register 5 (ADC1_DB5RL)

```

- struct {
  - [\\_BITS DATA](#): 2  
Data buffer 6 value [9:8].
  - [\\_BITS res](#): 6  
reserved
- } [DB6RH](#)
  
- ADC1 10-bit Data Buffer Register 6 (ADC1\_DB6RH)
- struct {
  - [\\_BITS DATA](#): 8  
Data buffer 6 value (low)
- } [DB6RL](#)
  
- ADC1 10-bit Data Buffer Register 6 (ADC1\_DB6RL)
- struct {
  - [\\_BITS DATA](#): 2  
Data buffer 7 value [9:8].
  - [\\_BITS res](#): 6  
reserved
- } [DB7RH](#)
  
- ADC1 10-bit Data Buffer Register 7 (ADC1\_DB7RH)
- struct {
  - [\\_BITS DATA](#): 8  
Data buffer 7 value (low)
- } [DB7RL](#)
  
- ADC1 10-bit Data Buffer Register 7 (ADC1\_DB7RL)
- struct {
  - [\\_BITS DATA](#): 2  
Data buffer 8 value [9:8].
  - [\\_BITS res](#): 6  
reserved
- } [DB8RH](#)
  
- ADC1 10-bit Data Buffer Register 8 (ADC1\_DB8RH)
- struct {
  - [\\_BITS DATA](#): 8  
Data buffer 8 value (low)
- } [DB8RL](#)
  
- ADC1 10-bit Data Buffer Register 8 (ADC1\_DB8RL)
- struct {
  - [\\_BITS DATA](#): 2  
Data buffer 9 value [9:8].
  - [\\_BITS res](#): 6  
reserved
- } [DB9RH](#)
  
- ADC1 10-bit Data Buffer Register 9 (ADC1\_DB9RH)
- struct {
  - [\\_BITS DATA](#): 8  
Data buffer 9 value (low)
- } [DB9RL](#)
  
- ADC1 10-bit Data Buffer Register 9 (ADC1\_DB8RL)

- uint8\_t **res** [12]  
Reserved register (12B)
  - struct {
    - \_BITS CH**: 4  
Channel selection bits.
    - \_BITS AWDIE**: 1  
Analog watchdog interrupt enable.
    - \_BITS EOCIE**: 1  
Interrupt enable for EOC.
    - \_BITS AWD**: 1  
Analog Watchdog flag.
    - \_BITS EOC**: 1  
End of conversion.
- CSR**
- ADC1 control/status register (ADC1\_CSR)
- struct {
    - \_BITS ADON**: 1  
A/D Converter on/off.
    - \_BITS CONT**: 1  
Continuous conversion.
    - \_BITS res**: 2  
Reserved, always read as 0.
    - \_BITS SPSEL**: 3  
Clock prescaler selection.
    - \_BITS res2**: 1  
Reserved, always read as 0.
- CR1**
- ADC1 Configuration Register 1 (ADC1\_CR1)
- struct {
    - \_BITS res**: 1  
Reserved, must be kept cleared.
    - \_BITS SCAN**: 1  
Scan mode enable.
    - \_BITS res2**: 1  
Reserved, must be kept cleared.
    - \_BITS ALIGN**: 1  
Data alignment.
    - \_BITS EXTSEL**: 2  
External event selection.
    - \_BITS EXTTRIG**: 1  
External trigger enable.
    - \_BITS res3**: 1  
Reserved, must be kept cleared.
- CR2**
- ADC1 Configuration Register 2 (ADC1\_CR2)
- struct {
    - \_BITS res**: 6  
Reserved, must be kept cleared.
    - \_BITS OVR**: 1  
Overrun flag.
    - \_BITS DBUF**: 1  
Data buffer enable.
- CR3**



*ADC1 Configuration Register 3 (ADC1\_CR3)*

- struct {
  - [\\_BITS DATA](#): 2  
Data value [9:8].
  - [\\_BITS res](#): 6  
reserved
- } [DRH](#)

*ADC1 (unbuffered) 10-bit measurement result (ADC1\_DRH)*

- struct {
  - [\\_BITS DATA](#): 8  
Data value [7:0].
- } [DRL](#)

*ADC1 (unbuffered) 10-bit measurement result (ADC1\_DRL)*

- struct {
  - [\\_BITS TDH](#): 8  
Schmitt trigger disable [15:8].
- } [TDRH](#)

*ADC1 Schmitt trigger disable register (ADC1\_TDRH)*

- struct {
  - [\\_BITS TDL](#): 8  
Schmitt trigger disable [7:0].
- } [TDRL](#)

*ADC1 Schmitt trigger disable register (ADC1\_TDRL)*

- struct {
  - [\\_BITS HT](#): 8  
watchdog high threshold [9:2]
- } [HTRH](#)

*ADC1 watchdog high threshold register (ADC1\_HTRH)*

- struct {
  - [\\_BITS HT](#): 2  
watchdog high threshold [1:0]
  - [\\_BITS res](#): 6  
Reserved.
- } [HTRL](#)

*ADC1 watchdog high threshold register (ADC1\_HTRL)*

- struct {
  - [\\_BITS LT](#): 8  
watchdog low threshold [9:2]
- } [LTRH](#)

*ADC1 watchdog low threshold register (ADC1\_LTRH)*

- struct {
  - [\\_BITS LT](#): 2  
watchdog low threshold [1:0]
  - [\\_BITS res](#): 6  
Reserved.
- } [LTRL](#)

*ADC1 watchdog low threshold register (ADC1\_LTRL)*

- struct {
  - [\\_BITS AWS](#): 2  
watchdog status register [9:8]
  - [\\_BITS res](#): 6  
Reserved.
- } [AWSRH](#)

*ADC1 watchdog status register (ADC1\_AWSRH)*

- struct {
  - [\\_BITS AWS](#): 8  
watchdog status register [7:0]
- } [AWSRL](#)

*ADC1 watchdog status register (ADC1\_AWSRL)*

- struct {
  - [\\_BITS AWEN](#): 2  
watchdog control register [9:8]
  - [\\_BITS res](#): 6  
Reserved.
- } [AWCRH](#)

*ADC1 watchdog control register (ADC1\_AWCRH)*

- struct {
  - [\\_BITS AWEN](#): 8  
watchdog control register [7:0]
- } [AWCRL](#)

*ADC1 watchdog control register (ADC1\_AWCRL)*

### 6.1.1 Detailed Description

struct containing Analog Digital Converter 1 (ADC1)

Definition at line 4539 of file STM8AF\_STM8S.h.

### 6.1.2 Field Documentation

#### 6.1.2.1 ADON

[\\_BITS](#) ADON

A/D Converter on/off.

Definition at line 4686 of file STM8AF\_STM8S.h.

### 6.1.2.2 ALIGN

`_BITS` ALIGN

Data alignment.

Definition at line 4699 of file STM8AF\_STM8S.h.

### 6.1.2.3 AWRH

```
struct { ... } AWRH
```

ADC1 watchdog control register (ADC1\_AWRH)

### 6.1.2.4 AWCRL

```
struct { ... } AWCRL
```

ADC1 watchdog control register (ADC1\_AWCRL)

### 6.1.2.5 AWD

`_BITS` AWD

Analog Watchdog flag.

Definition at line 4679 of file STM8AF\_STM8S.h.

### 6.1.2.6 AWDIE

`_BITS` AWDIE

Analog watchdog interrupt enable.

Definition at line 4677 of file STM8AF\_STM8S.h.

#### 6.1.2.7 AWEN

`_BITS` AWEN

watchdog control register [9:8]

watchdog control register [7:0]

Definition at line 4781 of file STM8AF\_STM8S.h.

#### 6.1.2.8 AWS

`_BITS` AWS

watchdog status register [9:8]

watchdog status register [7:0]

Definition at line 4768 of file STM8AF\_STM8S.h.

#### 6.1.2.9 AWSRH

```
struct { ... } AWSRH
```

ADC1 watchdog status register (ADC1\_AWSRH)

#### 6.1.2.10 AWSRL

```
struct { ... } AWSRL
```

ADC1 watchdog status register (ADC1\_AWSRL)

#### 6.1.2.11 CH

`_BITS` CH

Channel selection bits.

Definition at line 4676 of file STM8AF\_STM8S.h.

#### 6.1.2.12 CONT

`_BITS` CONT

Continuous conversion.

Definition at line 4687 of file STM8AF\_STM8S.h.

#### 6.1.2.13 CR1

```
struct { ... } CR1
```

ADC1 Configuration Register 1 (ADC1\_CR1)

#### 6.1.2.14 CR2

```
struct { ... } CR2
```

ADC1 Configuration Register 2 (ADC1\_CR2)

#### 6.1.2.15 CR3

```
struct { ... } CR3
```

ADC1 Configuration Register 3 (ADC1\_CR3)

#### 6.1.2.16 CSR

```
struct { ... } CSR
```

ADC1 control/status register (ADC1\_CSR)

#### 6.1.2.17 DATA

`_BITS` DATA

Data buffer 0 value [9:8].

Data value [7:0].

Data value [9:8].

Data buffer 9 value (low)

Data buffer 9 value [9:8].

Data buffer 8 value (low)

Data buffer 8 value [9:8].

Data buffer 7 value (low)

Data buffer 7 value [9:8].

Data buffer 6 value (low)

Data buffer 6 value [9:8].

Data buffer 5 value (low)

Data buffer 5 value [9:8].

Data buffer 4 value (low)

Data buffer 4 value [9:8].

Data buffer 3 value (low)

Data buffer 3 value [9:8].

Data buffer 2 value (low)

Data buffer 2 value [9:8].

Data buffer 1 value (low)

Data buffer 1 value [9:8].

Data buffer 0 value (low)

Definition at line 4543 of file STM8AF\_STM8S.h.

#### 6.1.2.18 DB0RH

```
struct { ... } DB0RH
```

ADC1 10-bit Data Buffer Register 0 (ADC1\_DB0RH)

**6.1.2.19 DB0RL**

```
struct { ... } DB0RL
```

ADC1 10-bit Data Buffer Register 0 (ADC1\_DB0RL)

**6.1.2.20 DB1RH**

```
struct { ... } DB1RH
```

ADC1 10-bit Data Buffer Register 1 (ADC1\_DB1RH)

**6.1.2.21 DB1RL**

```
struct { ... } DB1RL
```

ADC1 10-bit Data Buffer Register 1 (ADC1\_DB1RL)

**6.1.2.22 DB2RH**

```
struct { ... } DB2RH
```

ADC1 10-bit Data Buffer Register 2 (ADC1\_DB2RH)

**6.1.2.23 DB2RL**

```
struct { ... } DB2RL
```

ADC1 10-bit Data Buffer Register 2 (ADC1\_DB2RL)

**6.1.2.24 DB3RH**

```
struct { ... } DB3RH
```

ADC1 10-bit Data Buffer Register 3 (ADC1\_DB3RH)

#### 6.1.2.25 DB3RL

```
struct { ... } DB3RL
```

ADC1 10-bit Data Buffer Register 3 (ADC1\_DB3RL)

#### 6.1.2.26 DB4RH

```
struct { ... } DB4RH
```

ADC1 10-bit Data Buffer Register 4 (ADC1\_DB4RH)

#### 6.1.2.27 DB4RL

```
struct { ... } DB4RL
```

ADC1 10-bit Data Buffer Register 4 (ADC1\_DB4RL)

#### 6.1.2.28 DB5RH

```
struct { ... } DB5RH
```

ADC1 10-bit Data Buffer Register 5 (ADC1\_DB5RH)

#### 6.1.2.29 DB5RL

```
struct { ... } DB5RL
```

ADC1 10-bit Data Buffer Register 5 (ADC1\_DB5RL)

#### 6.1.2.30 DB6RH

```
struct { ... } DB6RH
```

ADC1 10-bit Data Buffer Register 6 (ADC1\_DB6RH)



**6.1.2.31 DB6RL**

```
struct { ... } DB6RL
```

ADC1 10-bit Data Buffer Register 6 (ADC1\_DB6RL)

**6.1.2.32 DB7RH**

```
struct { ... } DB7RH
```

ADC1 10-bit Data Buffer Register 7 (ADC1\_DB7RH)

**6.1.2.33 DB7RL**

```
struct { ... } DB7RL
```

ADC1 10-bit Data Buffer Register 7 (ADC1\_DB7RL)

**6.1.2.34 DB8RH**

```
struct { ... } DB8RH
```

ADC1 10-bit Data Buffer Register 8 (ADC1\_DB8RH)

**6.1.2.35 DB8RL**

```
struct { ... } DB8RL
```

ADC1 10-bit Data Buffer Register 8 (ADC1\_DB8RL)

**6.1.2.36 DB9RH**

```
struct { ... } DB9RH
```

ADC1 10-bit Data Buffer Register 9 (ADC1\_DB9RH)

#### 6.1.2.37 DB9RL

```
struct { ... } DB9RL
```

ADC1 10-bit Data Buffer Register 9 (ADC1\_DB8RL)

#### 6.1.2.38 DBUF

```
_BITS DBUF
```

Data buffer enable.

Definition at line 4710 of file STM8AF\_STM8S.h.

#### 6.1.2.39 DRH

```
struct { ... } DRH
```

ADC1 (unbuffered) 10-bit measurement result (ADC1\_DRH)

#### 6.1.2.40 DRL

```
struct { ... } DRL
```

ADC1 (unbuffered) 10-bit measurement result (ADC1\_DRL)

#### 6.1.2.41 EOC

```
_BITS EOC
```

End of conversion.

Definition at line 4680 of file STM8AF\_STM8S.h.

#### 6.1.2.42 EOCIE

```
_BITS EOCIE
```

Interrupt enable for EOC.

Definition at line 4678 of file STM8AF\_STM8S.h.

#### 6.1.2.43 EXTSEL

`_BITS` EXTSEL

External event selection.

Definition at line 4700 of file STM8AF\_STM8S.h.

#### 6.1.2.44 EXTTRIG

`_BITS` EXTTRIG

External trigger enable.

Definition at line 4701 of file STM8AF\_STM8S.h.

#### 6.1.2.45 HT

`_BITS` HT

watchdog high threshold [9:2]

watchdog high threshold [1:0]

Definition at line 4741 of file STM8AF\_STM8S.h.

#### 6.1.2.46 HTRH

```
struct { ... } HTRH
```

ADC1 watchdog high threshold register (ADC1\_HTRH)

#### 6.1.2.47 HTRL

```
struct { ... } HTRL
```

ADC1 watchdog high threshold register (ADC1\_HTRL)

#### 6.1.2.48 LT

`_BITS` LT

watchdog low threshold [9:2]

watchdog low threshold [1:0]

Definition at line 4755 of file STM8AF\_STM8S.h.

#### 6.1.2.49 LTRH

```
struct { ... } LTRH
```

ADC1 watchdog low threshold register (ADC1\_LTRH)

#### 6.1.2.50 LTRL

```
struct { ... } LTRL
```

ADC1 watchdog low threshold register (ADC1\_LTRL)

#### 6.1.2.51 OVR

`_BITS` OVR

Overrun flag.

Definition at line 4709 of file STM8AF\_STM8S.h.

#### 6.1.2.52 res [1/2]

`_BITS` res

reserved

Reserved.

Reserved, must be kept cleared.

Reserved, always read as 0.

Definition at line 4544 of file STM8AF\_STM8S.h.

**6.1.2.53 res** [2/2]

```
uint8_t res[12]
```

Reserved register (12B)

Definition at line 4671 of file STM8AF\_STM8S.h.

**6.1.2.54 res2**

```
_BITS res2
```

Reserved, always read as 0.

Reserved, must be kept cleared.

Definition at line 4690 of file STM8AF\_STM8S.h.

**6.1.2.55 res3**

```
_BITS res3
```

Reserved, must be kept cleared.

Definition at line 4702 of file STM8AF\_STM8S.h.

**6.1.2.56 SCAN**

```
_BITS SCAN
```

Scan mode enable.

Definition at line 4697 of file STM8AF\_STM8S.h.

**6.1.2.57 SPSEL**

```
_BITS SPSEL
```

Clock prescaler selection.

Definition at line 4689 of file STM8AF\_STM8S.h.

#### 6.1.2.58 TDH

`_BITS` TDH

Schmitt trigger disable [15:8].

Definition at line 4729 of file STM8AF\_STM8S.h.

#### 6.1.2.59 TDL

`_BITS` TDL

Schmitt trigger disable [7:0].

Definition at line 4735 of file STM8AF\_STM8S.h.

#### 6.1.2.60 TDRH

```
struct { ... } TDRH
```

ADC1 Schmitt trigger disable register (ADC1\_TDRH)

#### 6.1.2.61 TDRL

```
struct { ... } TDRL
```

ADC1 Schmitt trigger disable register (ADC1\_TDRL)

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.2 ADC2\_t Struct Reference

struct containing Analog Digital Converter 2 (ADC2)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
  - [\\_BITS CH](#): 4  
*Channel selection bits.*
  - [\\_BITS res](#): 1  
*Reserved.*
  - [\\_BITS EOCIE](#): 1  
*Interrupt enable for EOC.*
  - [\\_BITS res2](#): 1  
*Reserved.*
  - [\\_BITS EOC](#): 1  
*End of conversion.*

} [CSR](#)

*ADC2 control/status register (ADC2\_CSR)*

- struct {
  - [\\_BITS ADON](#): 1  
*A/D Converter on/off.*
  - [\\_BITS CONT](#): 1  
*Continuous conversion.*
  - [\\_BITS res](#): 2  
*Reserved, always read as 0.*
  - [\\_BITS SPSEL](#): 3  
*Clock prescaler selection.*
  - [\\_BITS res2](#): 1  
*Reserved, always read as 0.*

} [CR1](#)

*ADC2 Configuration Register 1 (ADC2\_CR1)*

- struct {
  - [\\_BITS res](#): 3  
*Reserved, must be kept cleared.*
  - [\\_BITS ALIGN](#): 1  
*Data alignment.*
  - [\\_BITS EXTSEL](#): 2  
*External event selection.*
  - [\\_BITS EXTTRIG](#): 1  
*External trigger enable.*
  - [\\_BITS res3](#): 1  
*Reserved, must be kept cleared.*

} [CR2](#)

*ADC2 Configuration Register 2 (ADC2\_CR2)*

- [uint8\\_t res](#) [1]  
*Reserved register (1B)*
- struct {
  - [\\_BITS DATA](#): 2  
*Data value [9:8].*
  - [\\_BITS res](#): 6  
*Reserved.*

} [DRH](#)

*ADC2 (unbuffered) 10-bit measurement result (ADC2\_DRH)*

- struct {
  - [\\_BITS DATA](#): 8  
*Data value [7:0].*

```
} DRL
```

*ADC2 (unbuffered) 10-bit measurement result (ADC2\_DRL)*

- struct {  
     \_BITS TDH: 8  
     *Schmitt trigger disable [9:8].*  
 } TDRH

*ADC2 Schmitt trigger disable register (ADC2\_TDRH)*

- struct {  
     \_BITS TDL: 8  
     *Schmitt trigger disable [7:0].*  
 } TDRL

*ADC2 Schmitt trigger disable register (ADC2\_TDRL)*

### 6.2.1 Detailed Description

struct containing Analog Digital Converter 2 (ADC2)

Definition at line 4894 of file STM8AF\_STM8S.h.

### 6.2.2 Field Documentation

#### 6.2.2.1 ADON

\_BITS ADON

A/D Converter on/off.

Definition at line 4908 of file STM8AF\_STM8S.h.

#### 6.2.2.2 ALIGN

\_BITS ALIGN

Data alignment.

Definition at line 4919 of file STM8AF\_STM8S.h.



### 6.2.2.3 CH

`_BITS` CH

Channel selection bits.

Definition at line 4898 of file STM8AF\_STM8S.h.

### 6.2.2.4 CONT

`_BITS` CONT

Continuous conversion.

Definition at line 4909 of file STM8AF\_STM8S.h.

### 6.2.2.5 CR1

```
struct { ... } CR1
```

ADC2 Configuration Register 1 (ADC2\_CR1)

### 6.2.2.6 CR2

```
struct { ... } CR2
```

ADC2 Configuration Register 2 (ADC2\_CR2)

### 6.2.2.7 CSR

```
struct { ... } CSR
```

ADC2 control/status register (ADC2\_CSR)

#### 6.2.2.8 DATA

`_BITS` DATA

Data value [9:8].

Data value [7:0].

Definition at line 4932 of file STM8AF\_STM8S.h.

#### 6.2.2.9 DRH

```
struct { ... } DRH
```

ADC2 (unbuffered) 10-bit measurement result (ADC2\_DRH)

#### 6.2.2.10 DRL

```
struct { ... } DRL
```

ADC2 (unbuffered) 10-bit measurement result (ADC2\_DRL)

#### 6.2.2.11 EOC

`_BITS` EOC

End of conversion.

Definition at line 4902 of file STM8AF\_STM8S.h.

#### 6.2.2.12 EOCIE

`_BITS` EOCIE

Interrupt enable for EOC.

Definition at line 4900 of file STM8AF\_STM8S.h.

#### 6.2.2.13 EXTSEL

`_BITS EXTSEL`

External event selection.

Definition at line 4920 of file STM8AF\_STM8S.h.

#### 6.2.2.14 EXTTRIG

`_BITS EXTTRIG`

External trigger enable.

Definition at line 4921 of file STM8AF\_STM8S.h.

#### 6.2.2.15 res [1/2]

`_BITS res`

Reserved.

Reserved, must be kept cleared.

Reserved, always read as 0.

Definition at line 4899 of file STM8AF\_STM8S.h.

#### 6.2.2.16 res [2/2]

`uint8_t res[1]`

Reserved register (1B)

Definition at line 4927 of file STM8AF\_STM8S.h.

#### 6.2.2.17 res2

`_BITS res2`

Reserved.

Reserved, always read as 0.

Definition at line 4901 of file STM8AF\_STM8S.h.

#### 6.2.2.18 res3

`_BITS res3`

Reserved, must be kept cleared.

Definition at line 4922 of file STM8AF\_STM8S.h.

#### 6.2.2.19 SPSEL

`_BITS SPSEL`

Clock prescaler selection.

Definition at line 4911 of file STM8AF\_STM8S.h.

#### 6.2.2.20 TDH

`_BITS TDH`

Schmitt trigger disable [9:8].

Definition at line 4945 of file STM8AF\_STM8S.h.

#### 6.2.2.21 TDL

`_BITS TDL`

Schmitt trigger disable [7:0].

Definition at line 4951 of file STM8AF\_STM8S.h.

#### 6.2.2.22 TDRH

```
struct { ... } TDRH
```

ADC2 Schmitt trigger disable register (ADC2\_TDRH)

## 6.2.2.23 TDRL

```
struct { ... } TDRL
```

ADC2 Schmitt trigger disable register (ADC2\_TDRL)

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.3 AWU\_t Struct Reference

struct for cofiguring the Auto Wake-Up Module (AWU)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
    - [\\_BITS MSR](#): 1  
*LSI measurement enable.*
    - [\\_BITS res](#): 3  
*Reserved.*
    - [\\_BITS AWUEN](#): 1  
*Auto-wakeup enable.*
    - [\\_BITS AWUF](#): 1  
*Auto-wakeup flag.*
    - [\\_BITS res2](#): 2  
*Reserved.*
- } [CSR](#)

*AWU Control/status register (AWU\_CSR)*

- struct {
    - [\\_BITS APRE](#): 6  
*Asynchronous prescaler divider.*
    - [\\_BITS res](#): 2  
*Reserved.*
- } [APR](#)

*AWU Asynchronous prescaler register (AWU\_APR)*

- struct {
    - [\\_BITS AWUTB](#): 4  
*Auto-wakeup timebase selection.*
    - [\\_BITS res](#): 4  
*Reserved.*
- } [TBR](#)

*AWU Timebase selection register (AWU\_TBR)*

### 6.3.1 Detailed Description

struct for configuring the Auto Wake-Up Module (AWU)

Definition at line 1100 of file STM8AF\_STM8S.h.

### 6.3.2 Field Documentation

#### 6.3.2.1 APR

```
struct { ... } APR
```

AWU Asynchronous prescaler register (AWU\_APR)

#### 6.3.2.2 APRE

```
_BITS APRE
```

Asynchronous prescaler divider.

Definition at line 1114 of file STM8AF\_STM8S.h.

#### 6.3.2.3 AWUEN

```
_BITS AWUEN
```

Auto-wakeup enable.

Definition at line 1106 of file STM8AF\_STM8S.h.

#### 6.3.2.4 AWUF

```
_BITS AWUF
```

Auto-wakeup flag.

Definition at line 1107 of file STM8AF\_STM8S.h.

#### 6.3.2.5 AWUTB

`_BITS` AWUTB

Auto-wakeup timebase selection.

Definition at line 1121 of file STM8AF\_STM8S.h.

#### 6.3.2.6 CSR

```
struct { ... } CSR
```

AWU Control/status register (AWU\_CSR)

#### 6.3.2.7 MSR

`_BITS` MSR

LSI measurement enable.

Definition at line 1104 of file STM8AF\_STM8S.h.

#### 6.3.2.8 res

`_BITS` res

Reserved.

Definition at line 1105 of file STM8AF\_STM8S.h.

#### 6.3.2.9 res2

`_BITS` res2

Reserved.

Definition at line 1108 of file STM8AF\_STM8S.h.

### 6.3.2.10 TBR

```
struct { ... } TBR
```

AWU Timebase selection register (AWU\_TBR)

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.4 BEEP\_t Struct Reference

struct for beeper control (BEEP)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {
    - [\\_BITS BEEPDIV](#): 5  
*Beep clock prescaler divider.*
    - [\\_BITS BEEPEN](#): 1  
*Beep enable.*
    - [\\_BITS BEEPSEL](#): 2  
*Beeper frequency selectioint8\_tn.*
  - } [CSR](#)
- Beeper control/status register (BEEP\_CSR)*

### 6.4.1 Detailed Description

struct for beeper control (BEEP)

Definition at line 1173 of file STM8AF\_STM8S.h.

### 6.4.2 Field Documentation

#### 6.4.2.1 BEEPDIV

[\\_BITS BEEPDIV](#)

Beep clock prescaler divider.

Definition at line 1177 of file STM8AF\_STM8S.h.



#### 6.4.2.2 BEEPEN

`_BITS` BEEPEN

Beep enable.

Definition at line 1178 of file STM8AF\_STM8S.h.

#### 6.4.2.3 BEEPSEL

`_BITS` BEEPSEL

Beeper frequency selectioint8\_tn.

Definition at line 1179 of file STM8AF\_STM8S.h.

#### 6.4.2.4 CSR

```
struct { ... } CSR
```

Beeper control/status register (BEEP\_CSR)

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.5 CAN\_t Struct Reference

struct for controlling Controller Area Network Module (CAN)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
  - [\\_BITS INRQ](#): 1  
*Initialization Request.*
  - [\\_BITS SLEEP](#): 1  
*Sleep Mode Request.*
  - [\\_BITS TXFP](#): 1  
*Transmit FIFO Priority.*
  - [\\_BITS RFLM](#): 1  
*Receive FIFO Locked Mode.*
  - [\\_BITS NART](#): 1  
*No Automatic Retransmission.*
  - [\\_BITS AWUM](#): 1  
*Automatic Wakeup Mode.*
  - [\\_BITS ABOM](#): 1  
*Automatic Bus-Off Management.*
  - [\\_BITS TTCM](#): 1  
*Time Triggered Communication Mode.*

} [MCR](#)

*CAN master control register (CAN\_MCR)*

- struct {
  - [\\_BITS INAK](#): 1  
*Initialization Acknowledge.*
  - [\\_BITS SLAK](#): 1  
*Sleep Acknowledge.*
  - [\\_BITS ERRI](#): 1  
*Error Interrupt.*
  - [\\_BITS WKUI](#): 1  
*Wakeup Interrupt.*
  - [\\_BITS TX](#): 1  
*Transmit.*
  - [\\_BITS RX](#): 1  
*Receive.*
  - [\\_BITS res](#): 2  
*Reserved.*

} [MSR](#)

*CAN master status register (CAN\_MSR)*

- struct {
  - [\\_BITS RQCP0](#): 1  
*Request Completed for Mailbox 0.*
  - [\\_BITS RQCP1](#): 1  
*Request Completed for Mailbox 1.*
  - [\\_BITS RQCP2](#): 1  
*Request Completed for Mailbox 2.*
  - [\\_BITS res](#): 1  
*Reserved.*
  - [\\_BITS TXOK0](#): 1  
*Transmission ok for Mailbox 0.*
  - [\\_BITS TXOK1](#): 1  
*Transmission ok for Mailbox 1.*
  - [\\_BITS TXOK2](#): 1  
*Transmission ok for Mailbox 2.*
  - [\\_BITS res2](#): 1  
*Reserved.*

} [TSR](#)

*CAN transmit status register (CAN\_TSR)*

- struct {
  - [\\_BITS CODE](#): 2  
*Mailbox Code.*
  - [\\_BITS TME0](#): 1  
*Transmit Mailbox 0 Empty.*
  - [\\_BITS TME1](#): 1  
*Transmit Mailbox 1 Empty.*
  - [\\_BITS TME2](#): 1  
*Transmit Mailbox 2 Empty.*
  - [\\_BITS LOW0](#): 1  
*Lowest Priority Flag for Mailbox 0.*
  - [\\_BITS LOW1](#): 1  
*Lowest Priority Flag for Mailbox 1.*
  - [\\_BITS LOW2](#): 1  
*Lowest Priority Flag for Mailbox 2.*
- } [TPR](#)

*CAN transmit priority register (CAN\_TPR)*

- struct {
  - [\\_BITS FMP](#): 2  
*FIFO Message Pending.*
  - [\\_BITS res](#): 1  
*Reserved.*
  - [\\_BITS FULL](#): 1  
*FIFO Full.*
  - [\\_BITS FOVR](#): 1  
*FIFO Overrun.*
  - [\\_BITS RFOM](#): 1  
*Release FIFO Output Mailbox.*
  - [\\_BITS res2](#): 2  
*Reserved.*
- } [RFR](#)

*CAN receive FIFO register (CAN\_RFR)*

- struct {
  - [\\_BITS TMEIE](#): 1  
*Transmit Mailbox Empty Interrupt Enable.*
  - [\\_BITS FMPIE](#): 1  
*FIFO Message Pending Interrupt Enable.*
  - [\\_BITS FFIE](#): 1  
*FIFO Full Interrupt Enable.*
  - [\\_BITS FOVIE](#): 1  
*FIFO Overrun Interrupt Enable.*
  - [\\_BITS res](#): 3  
*Reserved.*
  - [\\_BITS WKUIE](#): 1  
*Wakeup Interrupt Enable.*
- } [IER](#)

*CAN interrupt enable register (CAN\_IER)*

- struct {
  - [\\_BITS LBKM](#): 1  
*Loop back mode.*
  - [\\_BITS SILM](#): 1  
*Silent mode.*

```

    _BITS SAMP: 1
        Last sample point.
    _BITS RX: 1
        CAN Rx Signal.
    _BITS TXM2E: 1
        TX Mailbox 2 enable.
    _BITS res: 3
        Reserved.
} DGR

```

CAN diagnosis register (CAN\_DGR)

```

• struct {
    _BITS PS: 3
        Page select.
    _BITS res: 5
        Reserved.
} PSR

```

CAN page selection register for paged registers (CAN\_PSR)

```

• union {
    struct {
        struct {
            _BITS TXRQ: 1
                Transmission mailbox request.
            _BITS ABRQ: 1
                Abort request for mailbox.
            _BITS RQCP: 1
                Request completed.
            _BITS TXOK: 1
                Transmission OK.
            _BITS ALST: 1
                Arbitration lost.
            _BITS TERR: 1
                Transmission error.
            _BITS res: 2
                Reserved.
        } MCSR
    }
}

```

CAN message control/status register (CAN\_MCSR)

```

struct {
    _BITS DLC: 4
        Data length code.
    _BITS res: 3
        Reserved.
    _BITS TGT: 1
        Transmit global time.
} MDLCR

```

} MDLCR

CAN mailbox data length control register (CAN\_MDLCR)

```

struct {
    _BITS ID: 5
        STID[10:6] or EXID[28:24].
    _BITS RTR: 1
        Remote transmission request.
    _BITS IDE: 1
        Extended identifier.
    _BITS res: 1
        Reserved.
} MIDR1

```

} MIDR1

CAN mailbox identifier register 1 (CAN\_MIDR1)

```

struct {
    _BITS EXID: 2
        EXID[17:16].
    _BITS ID: 6
        STID[5:0] or EXID[23:18].
} MIDR2
    CAN mailbox identifier register 2 (CAN_MIDR2)
struct {
    _BITS EXID: 8
        EXID[15:8].
} MIDR3
    CAN mailbox identifier register 3 (CAN_MIDR3)
struct {
    _BITS EXID: 8
        EXID[7:0].
} MIDR4
    CAN mailbox identifier register 4 (CAN_MIDR4)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR1
    CAN mailbox data register 1 (CAN_MDAR1)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR2
    CAN mailbox data register 2 (CAN_MDAR2)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR3
    CAN mailbox data register 3 (CAN_MDAR3)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR4
    CAN mailbox data register 4 (CAN_MDAR4)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR5
    CAN mailbox data register 5 (CAN_MDAR5)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR6
    CAN mailbox data register 6 (CAN_MDAR6)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR7
    CAN mailbox data register 7 (CAN_MDAR7)
struct {
    _BITS DATA: 8
        data[7:0]
} MDAR8
    CAN mailbox data register 8 (CAN_MDAR8)
struct {

```

```

    _BITS TIME: 8
        Message time stamp [7:0].
} MTSRL
    CAN mailbox time stamp register low byte (CAN_MTSRL)
struct {
    _BITS TIME: 8
        Message time stamp [15:8].
} MTSRH
    CAN mailbox time stamp register high byte (CAN_MTSRH)
} PAGE_0
    CAN page 0: Tx Mailbox 0 (CAN.PAGE_0)
struct {
    struct {
        _BITS TXRQ: 1
            Transmission mailbox request.
        _BITS ABRQ: 1
            Abort request for mailbox.
        _BITS RQCP: 1
            Request completed.
        _BITS TXOK: 1
            Transmission OK.
        _BITS ALST: 1
            Arbitration lost.
        _BITS TERR: 1
            Transmission error.
        _BITS res: 2
            Reserved.
    } MCSR
        CAN message control/status register (CAN_MCSR)
    struct {
        _BITS DLC: 4
            Data length code.
        _BITS res: 3
            Reserved.
        _BITS TGT: 1
            Transmit global time.
    } MDLCR
        CAN mailbox data length control register (CAN_MDLCR)
    struct {
        _BITS ID: 5
            STID[10:6] or EXID[28:24].
        _BITS RTR: 1
            Remote transmission request.
        _BITS IDE: 1
            Extended identifier.
        _BITS res: 1
            Reserved.
    } MIDR1
        CAN mailbox identifier register 1 (CAN_MIDR1)
    struct {
        _BITS EXID: 2
            EXID[17:16].
        _BITS ID: 6
            STID[5:0] or EXID[23:18].
    } MIDR2
        CAN mailbox identifier register 2 (CAN_MIDR2)
    struct {
        _BITS EXID: 8

```

```

    EXID[15:8].
} MIDR3
    CAN mailbox identifier register 3 (CAN_MIDR3)
struct {
    _BITS EXID: 8
    EXID[7:0].
} MIDR4
    CAN mailbox identifier register 4 (CAN_MIDR4)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR1
    CAN mailbox data register 1 (CAN_MDAR1)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR2
    CAN mailbox data register 2 (CAN_MDAR2)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR3
    CAN mailbox data register 3 (CAN_MDAR3)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR4
    CAN mailbox data register 4 (CAN_MDAR4)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR5
    CAN mailbox data register 5 (CAN_MDAR5)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR6
    CAN mailbox data register 6 (CAN_MDAR6)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR7
    CAN mailbox data register 7 (CAN_MDAR7)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR8
    CAN mailbox data register 8 (CAN_MDAR8)
struct {
    _BITS TIME: 8
    Message time stamp [7:0].
} MTSRL
    CAN mailbox time stamp register low byte (CAN_MTSRL)
struct {
    _BITS TIME: 8
    Message time stamp [15:8].
} MTSRH
    CAN mailbox time stamp register high byte (CAN_MTSRH)

```

```

} PAGE_1
    CAN page 1: Tx Mailbox 1 (CAN.PAGE_1)
struct {
    struct {
        _BITS DATA: 8
        CAN reception filter 0/1.
    } F0R1
        CAN reception filter 0/1 (CAN_F0R1)
    struct {
        _BITS DATA: 8
        CAN reception filter 0/2.
    } F0R2
        CAN reception filter 0/2 (CAN_F0R2)
    struct {
        _BITS DATA: 8
        CAN reception filter 0/3.
    } F0R3
        CAN reception filter 0/3 (CAN_F0R3)
    struct {
        _BITS DATA: 8
        CAN reception filter 0/4.
    } F0R4
        CAN reception filter 0/4 (CAN_F0R4)
    struct {
        _BITS DATA: 8
        CAN reception filter 0/5.
    } F0R5
        CAN reception filter 0/5 (CAN_F0R5)
    struct {
        _BITS DATA: 8
        CAN reception filter 0/6.
    } F0R6
        CAN reception filter 0/6 (CAN_F0R6)
    struct {
        _BITS DATA: 8
        CAN reception filter 0/7.
    } F0R7
        CAN reception filter 0/7 (CAN_F0R7)
    struct {
        _BITS DATA: 8
        CAN reception filter 0/8.
    } F0R8
        CAN reception filter 0/8 (CAN_F0R8)
    struct {
        _BITS DATA: 8
        CAN reception filter 1/1.
    } F1R1
        CAN reception filter 1/1 (CAN_F1R1)
    struct {
        _BITS DATA: 8
        CAN reception filter 1/2.
    } F1R2
        CAN reception filter 1/2 (CAN_F1R2)
    struct {
        _BITS DATA: 8
        CAN reception filter 1/3.
    } F1R3
        CAN reception filter 1/3 (CAN_F1R3)

```



```

struct {
    \_BITS DATA: 8
        CAN reception filter 1/4.
} F1R4
    CAN reception filter 1/4 (CAN_F1R4)
struct {
    \_BITS DATA: 8
        CAN reception filter 1/5.
} F1R5
    CAN reception filter 1/5 (CAN_F1R5)
struct {
    \_BITS DATA: 8
        CAN reception filter 1/6.
} F1R6
    CAN reception filter 1/6 (CAN_F1R6)
struct {
    \_BITS DATA: 8
        CAN reception filter 1/7.
} F1R7
    CAN reception filter 1/7 (CAN_F1R7)
struct {
    \_BITS DATA: 8
        CAN reception filter 1/8.
} F1R8
    CAN reception filter 1/8 (CAN_F1R8)
} PAGE\_2
    CAN page 2: Acceptance Filter 0:1 (CAN.PAGE_2)
struct {
    struct {
        \_BITS DATA: 8
            CAN reception filter 2/1.
    } F2R1
        CAN reception filter 2/1 (CAN_F2R1)
    struct {
        \_BITS DATA: 8
            CAN reception filter 2/2.
    } F2R2
        CAN reception filter 2/2 (CAN_F2R2)
    struct {
        \_BITS DATA: 8
            CAN reception filter 2/3.
    } F2R3
        CAN reception filter 2/3 (CAN_F2R3)
    struct {
        \_BITS DATA: 8
            CAN reception filter 2/4.
    } F2R4
        CAN reception filter 2/4 (CAN_F2R4)
    struct {
        \_BITS DATA: 8
            CAN reception filter 2/5.
    } F2R5
        CAN reception filter 2/5 (CAN_F2R5)
    struct {
        \_BITS DATA: 8
            CAN reception filter 2/6.
    } F2R6
        CAN reception filter 2/6 (CAN_F2R6)

```

```

struct {
    \_BITS DATA: 8
        CAN reception filter 2/7.
} F2R7
    CAN reception filter 2/7 (CAN_F2R7)
struct {
    \_BITS DATA: 8
        CAN reception filter 2/8.
} F2R8
    CAN reception filter 2/8 (CAN_F2R8)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/1.
} F3R1
    CAN reception filter 3/1 (CAN_F3R1)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/2.
} F3R2
    CAN reception filter 3/2 (CAN_F3R2)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/3.
} F3R3
    CAN reception filter 3/3 (CAN_F3R3)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/4.
} F3R4
    CAN reception filter 3/4 (CAN_F3R4)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/5.
} F3R5
    CAN reception filter 3/5 (CAN_F3R5)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/6.
} F3R6
    CAN reception filter 3/6 (CAN_F3R6)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/7.
} F3R7
    CAN reception filter 3/7 (CAN_F3R7)
struct {
    \_BITS DATA: 8
        CAN reception filter 3/8.
} F3R8
    CAN reception filter 3/8 (CAN_F3R8)
} PAGE\_3
    CAN page 3: Acceptance Filter 2:3 (CAN.PAGE_3)
struct {
    struct {
        \_BITS DATA: 8
            CAN reception filter 4/1.
    } F4R1
        CAN reception filter 4/1 (CAN_F4R1)

```

```
struct {  
    \_BITS DATA: 8  
    CAN reception filter 4/2.  
} F4R2  
    CAN reception filter 4/2 (CAN_F4R2)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 4/3.  
} F4R3  
    CAN reception filter 4/3 (CAN_F4R3)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 4/4.  
} F4R4  
    CAN reception filter 4/4 (CAN_F4R4)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 4/5.  
} F4R5  
    CAN reception filter 4/5 (CAN_F4R5)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 4/6.  
} F4R6  
    CAN reception filter 4/6 (CAN_F4R6)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 4/7.  
} F4R7  
    CAN reception filter 4/7 (CAN_F4R7)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 4/8.  
} F4R8  
    CAN reception filter 4/8 (CAN_F4R8)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 5/1.  
} F5R1  
    CAN reception filter 5/1 (CAN_F5R1)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 5/2.  
} F5R2  
    CAN reception filter 5/2 (CAN_F5R2)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 5/3.  
} F5R3  
    CAN reception filter 5/3 (CAN_F5R3)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 5/4.  
} F5R4  
    CAN reception filter 5/4 (CAN_F5R4)  
struct {  
    \_BITS DATA: 8  
    CAN reception filter 5/5.
```

```

} F5R5
    CAN reception filter 5/5 (CAN_F5R5)
struct {
    _BITS DATA: 8
        CAN reception filter 5/6.
} F5R6
    CAN reception filter 5/6 (CAN_F5R6)
struct {
    _BITS DATA: 8
        CAN reception filter 5/7.
} F5R7
    CAN reception filter 5/7 (CAN_F5R7)
struct {
    _BITS DATA: 8
        CAN reception filter 5/8.
} F5R8
    CAN reception filter 5/8 (CAN_F5R8)
} PAGE_4
    CAN page 4: Acceptance Filter 4:5 (CAN.PAGE_4)
struct {
    struct {
        _BITS TXRQ: 1
            Transmission mailbox request.
        _BITS ABRQ: 1
            Abort request for mailbox.
        _BITS RQCP: 1
            Request completed.
        _BITS TXOK: 1
            Transmission OK.
        _BITS ALST: 1
            Arbitration lost.
        _BITS TERR: 1
            Transmission error.
        _BITS res: 2
            Reserved.
    } MCSR
        CAN message control/status register (CAN_MCSR)
    struct {
        _BITS DLC: 4
            Data length code.
        _BITS res: 3
            Reserved.
        _BITS TGT: 1
            Transmit global time.
    } MDLCR
        CAN mailbox data length control register (CAN_MDLCR)
    struct {
        _BITS ID: 5
            STID[10:6] or EXID[28:24].
        _BITS RTR: 1
            Remote transmission request.
        _BITS IDE: 1
            Extended identifier.
        _BITS res: 1
            Reserved.
    } MIDR1
        CAN mailbox identifier register 1 (CAN_MIDR1)
    struct {

```

```

    _BITS EXID: 2
    EXID[17:16].
    _BITS ID: 6
    STID[5:0] or EXID[23:18].
} MIDR2
    CAN mailbox identifier register 2 (CAN_MIDR2)
struct {
    _BITS EXID: 8
    EXID[15:8].
} MIDR3
    CAN mailbox identifier register 3 (CAN_MIDR3)
struct {
    _BITS EXID: 8
    EXID[7:0].
} MIDR4
    CAN mailbox identifier register 4 (CAN_MIDR4)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR1
    CAN mailbox data register 1 (CAN_MDAR1)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR2
    CAN mailbox data register 2 (CAN_MDAR2)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR3
    CAN mailbox data register 3 (CAN_MDAR3)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR4
    CAN mailbox data register 4 (CAN_MDAR4)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR5
    CAN mailbox data register 5 (CAN_MDAR5)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR6
    CAN mailbox data register 6 (CAN_MDAR6)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR7
    CAN mailbox data register 7 (CAN_MDAR7)
struct {
    _BITS DATA: 8
    data[7:0]
} MDAR8
    CAN mailbox data register 8 (CAN_MDAR8)
struct {
    _BITS TIME: 8

```

```

    Message time stamp [7:0].
} MTSRL
    CAN mailbox time stamp register low byte (CAN_MTSRL)
struct {
    _BITS TIME: 8
        Message time stamp [15:8].
} MTSRH
    CAN mailbox time stamp register high byte (CAN_MTSRH)
} PAGE_5
    CAN page 5: Tx Mailbox 2 (CAN.PAGE_5)
struct {
    struct {
        _BITS EWGF: 1
            Error warning flag.
        _BITS EPVF: 1
            Error passive flag.
        _BITS BOFF: 1
            Bus off flag.
        _BITS res: 1
            Reserved.
        _BITS LEC: 3
            Last error code.
        _BITS res2: 1
            Reserved.
    } ESR
        CAN error status register (CAN_ESR)
    struct {
        _BITS EWGIE: 1
            Error warning interrupt enable.
        _BITS EPVIE: 1
            Error passive interrupt enable.
        _BITS BOFIE: 1
            Bus-Off interrupt enable.
        _BITS res: 1
            Reserved.
        _BITS LECIE: 1
            Last error code interrupt enable.
        _BITS res2: 2
            Reserved.
        _BITS ERRIE: 1
            Error interrupt enable.
    } EIER
        CAN error interrupt enable register (CAN_EIER)
    struct {
        _BITS TEC: 8
            Transmit error counter.
    } TECR
        CAN transmit error counter register (CAN_TECR)
    struct {
        _BITS REC: 8
            Receive error counter.
    } RECR
        CAN receive error counter register (CAN_RECR)
    struct {
        _BITS BRP: 6
            Baud rate prescaler.
        _BITS SJW: 2
            Resynchronization jump width.
    }

```

```

} BTR1
    CAN bit timing register 1 (CAN_BTR1)
struct {
    _BITS BS1: 4
        Bit segment 1.
    _BITS BS2: 3
        Bit segment 2.
    _BITS res: 1
        Reserved, must be kept cleared.
} BTR2
    CAN bit timing register 2 (CAN_BTR2)
uint8_t res [2]
    Reserved registers (2B)
struct {
    _BITS FML0: 1
        Filter 0 mode low.
    _BITS FMH0: 1
        Filter 0 mode high.
    _BITS FML1: 1
        Filter 1 mode low.
    _BITS FMH1: 1
        Filter 1 mode high.
    _BITS FML2: 1
        Filter 2 mode low.
    _BITS FMH2: 1
        Filter 2 mode high.
    _BITS FML3: 1
        Filter 3 mode low.
    _BITS FMH3: 1
        Filter 3 mode high.
} FMR1
    CAN filter mode register 1 (CAN_FMR1)
struct {
    _BITS FML4: 1
        Filter 4 mode low.
    _BITS FMH4: 1
        Filter 4 mode high.
    _BITS FML5: 1
        Filter 5 mode low.
    _BITS FMH5: 1
        Filter 5 mode high.
    _BITS res: 4
        Reserved.
} FMR2
    CAN filter mode register 2 (CAN_FMR2)
struct {
    _BITS FACT0: 1
        Filter 0 active.
    _BITS FSC0: 2
        Filter 0 scale configuration.
    _BITS res: 1
        Reserved.
    _BITS FACT1: 1
        Filter 1 active.
    _BITS FSC1: 2
        Filter 1 scale configuration.
    _BITS res2: 1
        Reserved.

```

```

} FCR1
    CAN filter configuration register 1 (CAN_FCR1)
struct {
    _BITS FACT2: 1
        Filter 2 active.
    _BITS FSC2: 2
        Filter 2 scale configuration.
    _BITS res: 1
        Reserved.
    _BITS FACT3: 1
        Filter 3 active.
    _BITS FSC3: 2
        Filter 3 scale configuration.
    _BITS res2: 1
        Reserve.
} FCR2
    CAN filter configuration register 2 (CAN_FCR2)
struct {
    _BITS FACT4: 1
        Filter 4 active.
    _BITS FSC4: 2
        Filter 4 scale configuration.
    _BITS res: 1
        Reserved.
    _BITS FACT5: 1
        Filter 5 active.
    _BITS FSC5: 2
        Filter 5 scale configuration.
    _BITS res2: 1
        Reserve.
} FCR3
    CAN filter configuration register 3 (CAN_FCR3)
    _BITS res2 [3]
        Reserved registers (3B)
} PAGE_6
    CAN page 6: Configuration/Diagnostics (CAN.PAGE_6)
struct {
    struct {
        _BITS FMI: 8
            Filter match index.
    } MF MIR
        CAN mailbox filter match index register (CAN_MFMIR)
    struct {
        _BITS DLC: 4
            Data length code.
        _BITS res: 3
            Reserved.
        _BITS TGT: 1
            Transmit global time.
    } MDLC R
        CAN mailbox data length control register (CAN_MDLCR)
    struct {
        _BITS ID: 5
            STID[10:6] or EXID[28:24].
        _BITS RTR: 1
            Remote transmission request.
        _BITS IDE: 1
            Extended identifier.

```



```

    _BITS res: 1
        Reserved.
} MIDR1
    CAN mailbox identifier register 1 (CAN_MIDR1)
struct {
    _BITS EXID: 2
        EXID[17:16].
    _BITS ID: 6
        STID[5:0] or EXID[23:18].
} MIDR2
    CAN mailbox identifier register 2 (CAN_MIDR2)
struct {
    _BITS EXID: 8
        EXID[15:8].
} MIDR3
    CAN mailbox identifier register 3 (CAN_MIDR3)
struct {
    _BITS EXID: 8
        EXID[7:0].
} MIDR4
    CAN mailbox identifier register 4 (CAN_MIDR4)
struct {
    _BITS DATA: 8
        CAN mailbox data register 1.
} MDAR1
    CAN mailbox data register 1 (CAN_MDAR1)
struct {
    _BITS DATA: 8
        CAN mailbox data register 2.
} MDAR2
    CAN mailbox data register 2 (CAN_MDAR2)
struct {
    _BITS DATA: 8
        CAN mailbox data register 3.
} MDAR3
    CAN mailbox data register 3 (CAN_MDAR3)
struct {
    _BITS DATA: 8
        CAN mailbox data register 4.
} MDAR4
    CAN mailbox data register 4 (CAN_MDAR4)
struct {
    _BITS DATA: 8
        CAN mailbox data register 5.
} MDAR5
    CAN mailbox data register 5 (CAN_MDAR5)
struct {
    _BITS DATA: 8
        CAN mailbox data register 6.
} MDAR6
    CAN mailbox data register 6 (CAN_MDAR6)
struct {
    _BITS DATA: 8
        CAN mailbox data register 7.
} MDAR7
    CAN mailbox data register 7 (CAN_MDAR7)
struct {
    _BITS DATA: 8

```

```

    CAN mailbox data register 8.
} MDAR8
    CAN mailbox data register 8 (CAN_MDAR8)
struct {
    _BITS TIME: 8
    Message time stamp [7:0].
} MTSRL
    CAN mailbox time stamp register low byte (CAN_MTSRL)
struct {
    _BITS TIME: 8
    Message time stamp [15:8].
} MTSRH
    CAN mailbox time stamp register high byte (CAN_MTSRH)
} PAGE_7
    CAN page 7: Receive FIFO (CAN.PAGE_7)
} Page

```

*paged CAN registers (selection via CAN\_PSR)*

### 6.5.1 Detailed Description

struct for controlling Controller Area Network Module (CAN)

Definition at line 5013 of file STM8AF\_STM8S.h.

### 6.5.2 Field Documentation

#### 6.5.2.1 ABOM

`_BITS ABOM`

Automatic Bus-Off Management.

Definition at line 5023 of file STM8AF\_STM8S.h.

#### 6.5.2.2 ABRQ

`_BITS ABRQ`

Abort request for mailbox.

Definition at line 5115 of file STM8AF\_STM8S.h.

#### 6.5.2.3 ALST

`_BITS` ALST

Arbitration lost.

Definition at line 5118 of file STM8AF\_STM8S.h.

#### 6.5.2.4 AWUM

`_BITS` AWUM

Automatic Wakeup Mode.

Definition at line 5022 of file STM8AF\_STM8S.h.

#### 6.5.2.5 BOFF

`_BITS` BOFF

Bus off flag.

Definition at line 5722 of file STM8AF\_STM8S.h.

#### 6.5.2.6 BOFIE

`_BITS` BOFIE

Bus-Off interrupt enable.

Definition at line 5733 of file STM8AF\_STM8S.h.

#### 6.5.2.7 BRP

`_BITS` BRP

Baud rate prescaler.

Definition at line 5755 of file STM8AF\_STM8S.h.

#### 6.5.2.8 BS1

`_BITS` BS1

Bit segment 1.

Definition at line 5762 of file STM8AF\_STM8S.h.

#### 6.5.2.9 BS2

`_BITS` BS2

Bit segment 2.

Definition at line 5763 of file STM8AF\_STM8S.h.

#### 6.5.2.10 BTR1

```
struct { ... } BTR1
```

CAN bit timing register 1 (CAN\_BTR1)

#### 6.5.2.11 BTR2

```
struct { ... } BTR2
```

CAN bit timing register 2 (CAN\_BTR2)

#### 6.5.2.12 CODE

`_BITS` CODE

Mailbox Code.

Definition at line 5055 of file STM8AF\_STM8S.h.

## 6.5.2.13 DATA

[\\_BITS](#) DATA

data[7:0]

CAN mailbox data register 8.

CAN mailbox data register 7.

CAN mailbox data register 6.

CAN mailbox data register 5.

CAN mailbox data register 4.

CAN mailbox data register 3.

CAN mailbox data register 2.

CAN mailbox data register 1.

CAN reception filter 5/8.

CAN reception filter 5/7.

CAN reception filter 5/6.

CAN reception filter 5/5.

CAN reception filter 5/4.

CAN reception filter 5/3.

CAN reception filter 5/2.

CAN reception filter 5/1.

CAN reception filter 4/8.

CAN reception filter 4/7.

CAN reception filter 4/6.

CAN reception filter 4/5.

CAN reception filter 4/4.

CAN reception filter 4/3.

CAN reception filter 4/2.

CAN reception filter 4/1.

CAN reception filter 3/8.

CAN reception filter 3/7.

CAN reception filter 3/6.

CAN reception filter 3/5.

CAN reception filter 3/4.

CAN reception filter 3/3.

CAN reception filter 3/2.

CAN reception filter 3/1.

CAN reception filter 2/8.

CAN reception filter 2/7.

CAN reception filter 2/6.

CAN reception filter 2/5.

CAN reception filter 2/4.

CAN reception filter 2/3.

CAN reception filter 2/2.

CAN reception filter 2/1.

CAN reception filter 1/8.

CAN reception filter 1/7.

CAN reception filter 1/6.

CAN reception filter 1/5.

CAN reception filter 1/4.

CAN reception filter 1/3.

CAN reception filter 1/2.

CAN reception filter 1/1.

CAN reception filter 0/8.

CAN reception filter 0/7.

CAN reception filter 0/6.

CAN reception filter 0/5.

CAN reception filter 0/4.

CAN reception filter 0/3.

CAN reception filter 0/2.

CAN reception filter 0/1.

Definition at line 5162 of file STM8AF\_STM8S.h.

#### 6.5.2.14 DGR

```
struct { ... } DGR
```

CAN diagnosis register (CAN\_DGR)

#### 6.5.2.15 DLC

[\\_BITS](#) DLC

Data length code.

Definition at line 5126 of file STM8AF\_STM8S.h.

#### 6.5.2.16 EIER

```
struct { ... } EIER
```

CAN error interrupt enable register (CAN\_EIER)

#### 6.5.2.17 EPVF

[\\_BITS](#) EPVF

Error passive flag.

Definition at line 5721 of file STM8AF\_STM8S.h.

#### 6.5.2.18 EPVIE

[\\_BITS](#) EPVIE

Error passive interrupt enable.

Definition at line 5732 of file STM8AF\_STM8S.h.

#### 6.5.2.19 ERRI

`_BITS` `ERRI`

Error Interrupt.

Definition at line 5032 of file STM8AF\_STM8S.h.

#### 6.5.2.20 ERRIE

`_BITS` `ERRIE`

Error interrupt enable.

Definition at line 5737 of file STM8AF\_STM8S.h.

#### 6.5.2.21 ESR

```
struct { ... } ESR
```

CAN error status register (CAN\_ESR)

#### 6.5.2.22 EWGF

`_BITS` `EWGF`

Error warning flag.

Definition at line 5720 of file STM8AF\_STM8S.h.

#### 6.5.2.23 EWGIE

`_BITS` `EWGIE`

Error warning interrupt enable.

Definition at line 5731 of file STM8AF\_STM8S.h.



#### 6.5.2.24 EXID

`_BITS` EXID

EXID[17:16].

EXID[7:0].

EXID[15:8].

Definition at line 5143 of file STM8AF\_STM8S.h.

#### 6.5.2.25 F0R1

```
struct { ... } F0R1
```

CAN reception filter 0/1 (CAN\_F0R1)

#### 6.5.2.26 F0R2

```
struct { ... } F0R2
```

CAN reception filter 0/2 (CAN\_F0R2)

#### 6.5.2.27 F0R3

```
struct { ... } F0R3
```

CAN reception filter 0/3 (CAN\_F0R3)

#### 6.5.2.28 F0R4

```
struct { ... } F0R4
```

CAN reception filter 0/4 (CAN\_F0R4)

**6.5.2.29 F0R5**

```
struct { ... } F0R5
```

CAN reception filter 0/5 (CAN\_F0R5)

**6.5.2.30 F0R6**

```
struct { ... } F0R6
```

CAN reception filter 0/6 (CAN\_F0R6)

**6.5.2.31 F0R7**

```
struct { ... } F0R7
```

CAN reception filter 0/7 (CAN\_F0R7)

**6.5.2.32 F0R8**

```
struct { ... } F0R8
```

CAN reception filter 0/8 (CAN\_F0R8)

**6.5.2.33 F1R1**

```
struct { ... } F1R1
```

CAN reception filter 1/1 (CAN\_F1R1)

**6.5.2.34 F1R2**

```
struct { ... } F1R2
```

CAN reception filter 1/2 (CAN\_F1R2)

**6.5.2.35 F1R3**

```
struct { ... } F1R3
```

CAN reception filter 1/3 (CAN\_F1R3)

**6.5.2.36 F1R4**

```
struct { ... } F1R4
```

CAN reception filter 1/4 (CAN\_F1R4)

**6.5.2.37 F1R5**

```
struct { ... } F1R5
```

CAN reception filter 1/5 (CAN\_F1R5)

**6.5.2.38 F1R6**

```
struct { ... } F1R6
```

CAN reception filter 1/6 (CAN\_F1R6)

**6.5.2.39 F1R7**

```
struct { ... } F1R7
```

CAN reception filter 1/7 (CAN\_F1R7)

**6.5.2.40 F1R8**

```
struct { ... } F1R8
```

CAN reception filter 1/8 (CAN\_F1R8)

**6.5.2.41 F2R1**

```
struct { ... } F2R1
```

CAN reception filter 2/1 (CAN\_F2R1)

**6.5.2.42 F2R2**

```
struct { ... } F2R2
```

CAN reception filter 2/2 (CAN\_F2R2)

**6.5.2.43 F2R3**

```
struct { ... } F2R3
```

CAN reception filter 2/3 (CAN\_F2R3)

**6.5.2.44 F2R4**

```
struct { ... } F2R4
```

CAN reception filter 2/4 (CAN\_F2R4)

**6.5.2.45 F2R5**

```
struct { ... } F2R5
```

CAN reception filter 2/5 (CAN\_F2R5)

**6.5.2.46 F2R6**

```
struct { ... } F2R6
```

CAN reception filter 2/6 (CAN\_F2R6)

**6.5.2.47 F2R7**

```
struct { ... } F2R7
```

CAN reception filter 2/7 (CAN\_F2R7)

**6.5.2.48 F2R8**

```
struct { ... } F2R8
```

CAN reception filter 2/8 (CAN\_F2R8)

**6.5.2.49 F3R1**

```
struct { ... } F3R1
```

CAN reception filter 3/1 (CAN\_F3R1)

**6.5.2.50 F3R2**

```
struct { ... } F3R2
```

CAN reception filter 3/2 (CAN\_F3R2)

**6.5.2.51 F3R3**

```
struct { ... } F3R3
```

CAN reception filter 3/3 (CAN\_F3R3)

**6.5.2.52 F3R4**

```
struct { ... } F3R4
```

CAN reception filter 3/4 (CAN\_F3R4)

**6.5.2.53 F3R5**

```
struct { ... } F3R5
```

CAN reception filter 3/5 (CAN\_F3R5)

**6.5.2.54 F3R6**

```
struct { ... } F3R6
```

CAN reception filter 3/6 (CAN\_F3R6)

**6.5.2.55 F3R7**

```
struct { ... } F3R7
```

CAN reception filter 3/7 (CAN\_F3R7)

**6.5.2.56 F3R8**

```
struct { ... } F3R8
```

CAN reception filter 3/8 (CAN\_F3R8)

**6.5.2.57 F4R1**

```
struct { ... } F4R1
```

CAN reception filter 4/1 (CAN\_F4R1)

**6.5.2.58 F4R2**

```
struct { ... } F4R2
```

CAN reception filter 4/2 (CAN\_F4R2)

**6.5.2.59 F4R3**

```
struct { ... } F4R3
```

CAN reception filter 4/3 (CAN\_F4R3)

**6.5.2.60 F4R4**

```
struct { ... } F4R4
```

CAN reception filter 4/4 (CAN\_F4R4)

**6.5.2.61 F4R5**

```
struct { ... } F4R5
```

CAN reception filter 4/5 (CAN\_F4R5)

**6.5.2.62 F4R6**

```
struct { ... } F4R6
```

CAN reception filter 4/6 (CAN\_F4R6)

**6.5.2.63 F4R7**

```
struct { ... } F4R7
```

CAN reception filter 4/7 (CAN\_F4R7)

**6.5.2.64 F4R8**

```
struct { ... } F4R8
```

CAN reception filter 4/8 (CAN\_F4R8)

**6.5.2.65 F5R1**

```
struct { ... } F5R1
```

CAN reception filter 5/1 (CAN\_F5R1)

**6.5.2.66 F5R2**

```
struct { ... } F5R2
```

CAN reception filter 5/2 (CAN\_F5R2)

**6.5.2.67 F5R3**

```
struct { ... } F5R3
```

CAN reception filter 5/3 (CAN\_F5R3)

**6.5.2.68 F5R4**

```
struct { ... } F5R4
```

CAN reception filter 5/4 (CAN\_F5R4)

**6.5.2.69 F5R5**

```
struct { ... } F5R5
```

CAN reception filter 5/5 (CAN\_F5R5)

**6.5.2.70 F5R6**

```
struct { ... } F5R6
```

CAN reception filter 5/6 (CAN\_F5R6)



**6.5.2.71 F5R7**

```
struct { ... } F5R7
```

CAN reception filter 5/7 (CAN\_F5R7)

**6.5.2.72 F5R8**

```
struct { ... } F5R8
```

CAN reception filter 5/8 (CAN\_F5R8)

**6.5.2.73 FACT0**

```
_BITS FACT0
```

Filter 0 active.

Definition at line 5797 of file STM8AF\_STM8S.h.

**6.5.2.74 FACT1**

```
_BITS FACT1
```

Filter 1 active.

Definition at line 5800 of file STM8AF\_STM8S.h.

**6.5.2.75 FACT2**

```
_BITS FACT2
```

Filter 2 active.

Definition at line 5808 of file STM8AF\_STM8S.h.

#### 6.5.2.76 FACT3

`_BITS` FACT3

Filter 3 active.

Definition at line 5811 of file STM8AF\_STM8S.h.

#### 6.5.2.77 FACT4

`_BITS` FACT4

Filter 4 active.

Definition at line 5819 of file STM8AF\_STM8S.h.

#### 6.5.2.78 FACT5

`_BITS` FACT5

Filter 5 active.

Definition at line 5822 of file STM8AF\_STM8S.h.

#### 6.5.2.79 FCR1

```
struct { ... } FCR1
```

CAN filter configuration register 1 (CAN\_FCR1)

#### 6.5.2.80 FCR2

```
struct { ... } FCR2
```

CAN filter configuration register 2 (CAN\_FCR2)

#### 6.5.2.81 FCR3

```
struct { ... } FCR3
```

CAN filter configuration register 3 (CAN\_FCR3)

#### 6.5.2.82 FFIE

```
_BITS FFIE
```

FIFO Full Interrupt Enable.

Definition at line 5080 of file STM8AF\_STM8S.h.

#### 6.5.2.83 FMH0

```
_BITS FMH0
```

Filter 0 mode high.

Definition at line 5775 of file STM8AF\_STM8S.h.

#### 6.5.2.84 FMH1

```
_BITS FMH1
```

Filter 1 mode high.

Definition at line 5777 of file STM8AF\_STM8S.h.

#### 6.5.2.85 FMH2

```
_BITS FMH2
```

Filter 2 mode high.

Definition at line 5779 of file STM8AF\_STM8S.h.

**6.5.2.86 FMH3**

`_BITS` FMH3

Filter 3 mode high.

Definition at line 5781 of file STM8AF\_STM8S.h.

**6.5.2.87 FMH4**

`_BITS` FMH4

Filter 4 mode high.

Definition at line 5788 of file STM8AF\_STM8S.h.

**6.5.2.88 FMH5**

`_BITS` FMH5

Filter 5 mode high.

Definition at line 5790 of file STM8AF\_STM8S.h.

**6.5.2.89 FMI**

`_BITS` FMI

Filter match index.

Definition at line 5840 of file STM8AF\_STM8S.h.

**6.5.2.90 FML0**

`_BITS` FML0

Filter 0 mode low.

Definition at line 5774 of file STM8AF\_STM8S.h.

#### 6.5.2.91 FML1

`_BITS FML1`

Filter 1 mode low.

Definition at line 5776 of file STM8AF\_STM8S.h.

#### 6.5.2.92 FML2

`_BITS FML2`

Filter 2 mode low.

Definition at line 5778 of file STM8AF\_STM8S.h.

#### 6.5.2.93 FML3

`_BITS FML3`

Filter 3 mode low.

Definition at line 5780 of file STM8AF\_STM8S.h.

#### 6.5.2.94 FML4

`_BITS FML4`

Filter 4 mode low.

Definition at line 5787 of file STM8AF\_STM8S.h.

#### 6.5.2.95 FML5

`_BITS FML5`

Filter 5 mode low.

Definition at line 5789 of file STM8AF\_STM8S.h.

#### 6.5.2.96 FMP

`_BITS` FMP

FIFO Message Pending.

Definition at line 5067 of file STM8AF\_STM8S.h.

#### 6.5.2.97 FMPIE

`_BITS` FMPIE

FIFO Message Pending Interrupt Enable.

Definition at line 5079 of file STM8AF\_STM8S.h.

#### 6.5.2.98 FMR1

```
struct { ... } FMR1
```

CAN filter mode register 1 (CAN\_FMR1)

#### 6.5.2.99 FMR2

```
struct { ... } FMR2
```

CAN filter mode register 2 (CAN\_FMR2)

#### 6.5.2.100 FOVIE

`_BITS` FOVIE

FIFO Overrun Interrupt Enable.

Definition at line 5081 of file STM8AF\_STM8S.h.

#### 6.5.2.101 FOVR

`_BITS` FOVR

FIFO Overrun.

Definition at line 5070 of file STM8AF\_STM8S.h.

#### 6.5.2.102 FSC0

`_BITS` FSC0

Filter 0 scale configuration.

Definition at line 5798 of file STM8AF\_STM8S.h.

#### 6.5.2.103 FSC1

`_BITS` FSC1

Filter 1 scale configuration.

Definition at line 5801 of file STM8AF\_STM8S.h.

#### 6.5.2.104 FSC2

`_BITS` FSC2

Filter 2 scale configuration.

Definition at line 5809 of file STM8AF\_STM8S.h.

#### 6.5.2.105 FSC3

`_BITS` FSC3

Filter 3 scale configuration.

Definition at line 5812 of file STM8AF\_STM8S.h.

**6.5.2.106 FSC4**

`_BITS FSC4`

Filter 4 scale configuration.

Definition at line 5820 of file STM8AF\_STM8S.h.

**6.5.2.107 FSC5**

`_BITS FSC5`

Filter 5 scale configuration.

Definition at line 5823 of file STM8AF\_STM8S.h.

**6.5.2.108 FULL**

`_BITS FULL`

FIFO Full.

Definition at line 5069 of file STM8AF\_STM8S.h.

**6.5.2.109 ID**

`_BITS ID`

STID[10:6] or EXID[28:24].

STID[5:0] or EXID[23:18].

Definition at line 5134 of file STM8AF\_STM8S.h.

**6.5.2.110 IDE**

`_BITS IDE`

Extended identifier.

Definition at line 5136 of file STM8AF\_STM8S.h.



#### 6.5.2.111 IER

```
struct { ... } IER
```

CAN interrupt enable register (CAN\_IER)

#### 6.5.2.112 INAK

```
_BITS INAK
```

Initialization Acknowledge.

Definition at line 5030 of file STM8AF\_STM8S.h.

#### 6.5.2.113 INRQ

```
_BITS INRQ
```

Initialization Request.

Definition at line 5017 of file STM8AF\_STM8S.h.

#### 6.5.2.114 LBKM

```
_BITS LBKM
```

Loop back mode.

Definition at line 5089 of file STM8AF\_STM8S.h.

#### 6.5.2.115 LEC

```
_BITS LEC
```

Last error code.

Definition at line 5724 of file STM8AF\_STM8S.h.

#### 6.5.2.116 LECIE

`_BITS` LECIE

Last error code interrupt enable.

Definition at line 5735 of file STM8AF\_STM8S.h.

#### 6.5.2.117 LOW0

`_BITS` LOW0

Lowest Priority Flag for Mailbox 0.

Definition at line 5059 of file STM8AF\_STM8S.h.

#### 6.5.2.118 LOW1

`_BITS` LOW1

Lowest Priority Flag for Mailbox 1.

Definition at line 5060 of file STM8AF\_STM8S.h.

#### 6.5.2.119 LOW2

`_BITS` LOW2

Lowest Priority Flag for Mailbox 2.

Definition at line 5061 of file STM8AF\_STM8S.h.

#### 6.5.2.120 MCR

```
struct { ... } MCR
```

CAN master control register (CAN\_MCR)

**6.5.2.121 MCSR** [1/3]

```
struct { ... } MCSR
```

CAN message control/status register (CAN\_MCSR)

**6.5.2.122 MCSR** [2/3]

```
struct { ... } MCSR
```

CAN message control/status register (CAN\_MCSR)

**6.5.2.123 MCSR** [3/3]

```
struct { ... } MCSR
```

CAN message control/status register (CAN\_MCSR)

**6.5.2.124 MDAR1** [1/4]

```
struct { ... } MDAR1
```

CAN mailbox data register 1 (CAN\_MDAR1)

**6.5.2.125 MDAR1** [2/4]

```
struct { ... } MDAR1
```

CAN mailbox data register 1 (CAN\_MDAR1)

**6.5.2.126 MDAR1** [3/4]

```
struct { ... } MDAR1
```

CAN mailbox data register 1 (CAN\_MDAR1)

**6.5.2.127 MDAR1** [4/4]

```
struct { ... } MDAR1
```

CAN mailbox data register 1 (CAN\_MDAR1)

**6.5.2.128 MDAR2** [1/4]

```
struct { ... } MDAR2
```

CAN mailbox data register 2 (CAN\_MDAR2)

**6.5.2.129 MDAR2** [2/4]

```
struct { ... } MDAR2
```

CAN mailbox data register 2 (CAN\_MDAR2)

**6.5.2.130 MDAR2** [3/4]

```
struct { ... } MDAR2
```

CAN mailbox data register 2 (CAN\_MDAR2)

**6.5.2.131 MDAR2** [4/4]

```
struct { ... } MDAR2
```

CAN mailbox data register 2 (CAN\_MDAR2)

**6.5.2.132 MDAR3** [1/4]

```
struct { ... } MDAR3
```

CAN mailbox data register 3 (CAN\_MDAR3)

**6.5.2.133 MDAR3** [2/4]

```
struct { ... } MDAR3
```

CAN mailbox data register 3 (CAN\_MDAR3)

**6.5.2.134 MDAR3** [3/4]

```
struct { ... } MDAR3
```

CAN mailbox data register 3 (CAN\_MDAR3)

**6.5.2.135 MDAR3** [4/4]

```
struct { ... } MDAR3
```

CAN mailbox data register 3 (CAN\_MDAR3)

**6.5.2.136 MDAR4** [1/4]

```
struct { ... } MDAR4
```

CAN mailbox data register 4 (CAN\_MDAR4)

**6.5.2.137 MDAR4** [2/4]

```
struct { ... } MDAR4
```

CAN mailbox data register 4 (CAN\_MDAR4)

**6.5.2.138 MDAR4** [3/4]

```
struct { ... } MDAR4
```

CAN mailbox data register 4 (CAN\_MDAR4)

**6.5.2.139 MDAR4** [4/4]

```
struct { ... } MDAR4
```

CAN mailbox data register 4 (CAN\_MDAR4)

**6.5.2.140 MDAR5** [1/4]

```
struct { ... } MDAR5
```

CAN mailbox data register 5 (CAN\_MDAR5)

**6.5.2.141 MDAR5** [2/4]

```
struct { ... } MDAR5
```

CAN mailbox data register 5 (CAN\_MDAR5)

**6.5.2.142 MDAR5** [3/4]

```
struct { ... } MDAR5
```

CAN mailbox data register 5 (CAN\_MDAR5)

**6.5.2.143 MDAR5** [4/4]

```
struct { ... } MDAR5
```

CAN mailbox data register 5 (CAN\_MDAR5)

**6.5.2.144 MDAR6** [1/4]

```
struct { ... } MDAR6
```

CAN mailbox data register 6 (CAN\_MDAR6)

**6.5.2.145 MDAR6** [2/4]

```
struct { ... } MDAR6
```

CAN mailbox data register 6 (CAN\_MDAR6)

**6.5.2.146 MDAR6** [3/4]

```
struct { ... } MDAR6
```

CAN mailbox data register 6 (CAN\_MDAR6)

**6.5.2.147 MDAR6** [4/4]

```
struct { ... } MDAR6
```

CAN mailbox data register 6 (CAN\_MDAR6)

**6.5.2.148 MDAR7** [1/4]

```
struct { ... } MDAR7
```

CAN mailbox data register 7 (CAN\_MDAR7)

**6.5.2.149 MDAR7** [2/4]

```
struct { ... } MDAR7
```

CAN mailbox data register 7 (CAN\_MDAR7)

**6.5.2.150 MDAR7** [3/4]

```
struct { ... } MDAR7
```

CAN mailbox data register 7 (CAN\_MDAR7)

**6.5.2.151 MDAR7** [4/4]

```
struct { ... } MDAR7
```

CAN mailbox data register 7 (CAN\_MDAR7)

**6.5.2.152 MDAR8** [1/4]

```
struct { ... } MDAR8
```

CAN mailbox data register 8 (CAN\_MDAR8)

**6.5.2.153 MDAR8** [2/4]

```
struct { ... } MDAR8
```

CAN mailbox data register 8 (CAN\_MDAR8)

**6.5.2.154 MDAR8** [3/4]

```
struct { ... } MDAR8
```

CAN mailbox data register 8 (CAN\_MDAR8)

**6.5.2.155 MDAR8** [4/4]

```
struct { ... } MDAR8
```

CAN mailbox data register 8 (CAN\_MDAR8)

**6.5.2.156 MDLCR** [1/4]

```
struct { ... } MDLCR
```

CAN mailbox data length control register (CAN\_MDLCR)



**6.5.2.157 MDLCR** [2/4]

```
struct { ... } MDLCR
```

CAN mailbox data length control register (CAN\_MDLCR)

**6.5.2.158 MDLCR** [3/4]

```
struct { ... } MDLCR
```

CAN mailbox data length control register (CAN\_MDLCR)

**6.5.2.159 MDLCR** [4/4]

```
struct { ... } MDLCR
```

CAN mailbox data length control register (CAN\_MDLCR)

**6.5.2.160 MFMIR**

```
struct { ... } MFMIR
```

CAN mailbox filter match index register (CAN\_MFMIR)

**6.5.2.161 MIDR1** [1/4]

```
struct { ... } MIDR1
```

CAN mailbox identifier register 1 (CAN\_MIDR1)

**6.5.2.162 MIDR1** [2/4]

```
struct { ... } MIDR1
```

CAN mailbox identifier register 1 (CAN\_MIDR1)

**6.5.2.163 MIDR1** [3/4]

```
struct { ... } MIDR1
```

CAN mailbox identifier register 1 (CAN\_MIDR1)

**6.5.2.164 MIDR1** [4/4]

```
struct { ... } MIDR1
```

CAN mailbox identifier register 1 (CAN\_MIDR1)

**6.5.2.165 MIDR2** [1/4]

```
struct { ... } MIDR2
```

CAN mailbox identifier register 2 (CAN\_MIDR2)

**6.5.2.166 MIDR2** [2/4]

```
struct { ... } MIDR2
```

CAN mailbox identifier register 2 (CAN\_MIDR2)

**6.5.2.167 MIDR2** [3/4]

```
struct { ... } MIDR2
```

CAN mailbox identifier register 2 (CAN\_MIDR2)

**6.5.2.168 MIDR2** [4/4]

```
struct { ... } MIDR2
```

CAN mailbox identifier register 2 (CAN\_MIDR2)

**6.5.2.169 MIDR3** [1/4]

```
struct { ... } MIDR3
```

CAN mailbox identifier register 3 (CAN\_MIDR3)

**6.5.2.170 MIDR3** [2/4]

```
struct { ... } MIDR3
```

CAN mailbox identifier register 3 (CAN\_MIDR3)

**6.5.2.171 MIDR3** [3/4]

```
struct { ... } MIDR3
```

CAN mailbox identifier register 3 (CAN\_MIDR3)

**6.5.2.172 MIDR3** [4/4]

```
struct { ... } MIDR3
```

CAN mailbox identifier register 3 (CAN\_MIDR3)

**6.5.2.173 MIDR4** [1/4]

```
struct { ... } MIDR4
```

CAN mailbox identifier register 4 (CAN\_MIDR4)

**6.5.2.174 MIDR4** [2/4]

```
struct { ... } MIDR4
```

CAN mailbox identifier register 4 (CAN\_MIDR4)

**6.5.2.175 MIDR4** [3/4]

```
struct { ... } MIDR4
```

CAN mailbox identifier register 4 (CAN\_MIDR4)

**6.5.2.176 MIDR4** [4/4]

```
struct { ... } MIDR4
```

CAN mailbox identifier register 4 (CAN\_MIDR4)

**6.5.2.177 MSR**

```
struct { ... } MSR
```

CAN master status register (CAN\_MSR)

**6.5.2.178 MTSRH** [1/4]

```
struct { ... } MTSRH
```

CAN mailbox time stamp register high byte (CAN\_MTSRH)

**6.5.2.179 MTSRH** [2/4]

```
struct { ... } MTSRH
```

CAN mailbox time stamp register high byte (CAN\_MTSRH)

**6.5.2.180 MTSRH** [3/4]

```
struct { ... } MTSRH
```

CAN mailbox time stamp register high byte (CAN\_MTSRH)

**6.5.2.181 MTSRH** [4/4]

```
struct { ... } MTSRH
```

CAN mailbox time stamp register high byte (CAN\_MTSRH)

**6.5.2.182 MTSRL** [1/4]

```
struct { ... } MTSRL
```

CAN mailbox time stamp register low byte (CAN\_MTSRL)

**6.5.2.183 MTSRL** [2/4]

```
struct { ... } MTSRL
```

CAN mailbox time stamp register low byte (CAN\_MTSRL)

**6.5.2.184 MTSRL** [3/4]

```
struct { ... } MTSRL
```

CAN mailbox time stamp register low byte (CAN\_MTSRL)

**6.5.2.185 MTSRL** [4/4]

```
struct { ... } MTSRL
```

CAN mailbox time stamp register low byte (CAN\_MTSRL)

**6.5.2.186 NART**

`_BITS` NART

No Automatic Retransmission.

Definition at line 5021 of file STM8AF\_STM8S.h.

**6.5.2.187 Page**

```
union { ... } Page
```

paged CAN registers (selection via CAN\_PSR)

**6.5.2.188 PAGE\_0**

```
struct { ... } PAGE_0
```

CAN page 0: Tx Mailbox 0 (CAN.PAGE\_0)

**6.5.2.189 PAGE\_1**

```
struct { ... } PAGE_1
```

CAN page 1: Tx Mailbox 1 (CAN.PAGE\_1)

**6.5.2.190 PAGE\_2**

```
struct { ... } PAGE_2
```

CAN page 2: Acceptance Filter 0:1 (CAN.PAGE\_2)

**6.5.2.191 PAGE\_3**

```
struct { ... } PAGE_3
```

CAN page 3: Acceptance Filter 2:3 (CAN.PAGE\_3)

**6.5.2.192 PAGE\_4**

```
struct { ... } PAGE_4
```

CAN page 4: Acceptance Filter 4:5 (CAN.PAGE\_4)

**6.5.2.193 PAGE\_5**

```
struct { ... } PAGE_5
```

CAN page 5: Tx Mailbox 2 (CAN.PAGE\_5)

**6.5.2.194 PAGE\_6**

```
struct { ... } PAGE_6
```

CAN page 6: Configuration/Diagnostics (CAN.PAGE\_6)

**6.5.2.195 PAGE\_7**

```
struct { ... } PAGE_7
```

CAN page 7: Receive FIFO (CAN.PAGE\_7)

**6.5.2.196 PS**

```
_BITS PS
```

Page select.

Definition at line 5100 of file STM8AF\_STM8S.h.

**6.5.2.197 PSR**

```
struct { ... } PSR
```

CAN page selection register for paged registers (CAN\_PSR)

**6.5.2.198 REC**

```
_BITS REC
```

Receive error counter.

Definition at line 5749 of file STM8AF\_STM8S.h.

**6.5.2.199 RECR**

```
struct { ... } RECR
```

CAN receive error counter register (CAN\_RECR)

**6.5.2.200 res [1/2]**

```
_BITS res
```

Reserved.

Reserved, must be kept cleared.

Definition at line 5036 of file STM8AF\_STM8S.h.

**6.5.2.201 res [2/2]**

```
uint8_t res[2]
```

Reserved registers (2B)

Definition at line 5769 of file STM8AF\_STM8S.h.

**6.5.2.202 res2**

```
_BITS res2[3]
```

Reserved.

Reserved registers (3B)

Reserve.

Definition at line 5049 of file STM8AF\_STM8S.h.

**6.5.2.203 RFLM**

```
_BITS RFLM
```

Receive FIFO Locked Mode.

Definition at line 5020 of file STM8AF\_STM8S.h.



#### 6.5.2.204 RFOM

`_BITS` RFOM

Release FIFO Output Mailbox.

Definition at line 5071 of file STM8AF\_STM8S.h.

#### 6.5.2.205 RFR

```
struct { ... } RFR
```

CAN receive FIFO register (CAN\_RFR)

#### 6.5.2.206 RQCP

`_BITS` RQCP

Request completed.

Definition at line 5116 of file STM8AF\_STM8S.h.

#### 6.5.2.207 RQCP0

`_BITS` RQCP0

Request Completed for Mailbox 0.

Definition at line 5042 of file STM8AF\_STM8S.h.

#### 6.5.2.208 RQCP1

`_BITS` RQCP1

Request Completed for Mailbox 1.

Definition at line 5043 of file STM8AF\_STM8S.h.

**6.5.2.209 RQCP2**

`_BITS` RQCP2

Request Completed for Mailbox 2.

Definition at line 5044 of file STM8AF\_STM8S.h.

**6.5.2.210 RTR**

`_BITS` RTR

Remote transmission request.

Definition at line 5135 of file STM8AF\_STM8S.h.

**6.5.2.211 RX**

`_BITS` RX

Receive.

CAN Rx Signal.

Definition at line 5035 of file STM8AF\_STM8S.h.

**6.5.2.212 SAMP**

`_BITS` SAMP

Last sample point.

Definition at line 5091 of file STM8AF\_STM8S.h.

**6.5.2.213 SILM**

`_BITS` SILM

Silent mode.

Definition at line 5090 of file STM8AF\_STM8S.h.

#### 6.5.2.214 SJW

`_BITS` SJW

Resynchronization jump width.

Definition at line 5756 of file STM8AF\_STM8S.h.

#### 6.5.2.215 SLAK

`_BITS` SLAK

Sleep Acknowledge.

Definition at line 5031 of file STM8AF\_STM8S.h.

#### 6.5.2.216 SLEEP

`_BITS` SLEEP

Sleep Mode Request.

Definition at line 5018 of file STM8AF\_STM8S.h.

#### 6.5.2.217 TEC

`_BITS` TEC

Transmit error counter.

Definition at line 5743 of file STM8AF\_STM8S.h.

#### 6.5.2.218 TECR

```
struct { ... } TECR
```

CAN transmit error counter register (CAN\_TECR)

**6.5.2.219 TERR**

`_BITS TERR`

Transmission error.

Definition at line 5119 of file STM8AF\_STM8S.h.

**6.5.2.220 TGT**

`_BITS TGT`

Transmit global time.

Definition at line 5128 of file STM8AF\_STM8S.h.

**6.5.2.221 TIME**

`_BITS TIME`

Message time stamp [7:0].

Message time stamp [15:8].

Definition at line 5210 of file STM8AF\_STM8S.h.

**6.5.2.222 TME0**

`_BITS TME0`

Transmit Mailbox 0 Empty.

Definition at line 5056 of file STM8AF\_STM8S.h.

**6.5.2.223 TME1**

`_BITS TME1`

Transmit Mailbox 1 Empty.

Definition at line 5057 of file STM8AF\_STM8S.h.

#### 6.5.2.224 TME2

`_BITS TME2`

Transmit Mailbox 2 Empty.

Definition at line 5058 of file STM8AF\_STM8S.h.

#### 6.5.2.225 TMEIE

`_BITS TMEIE`

Transmit Mailbox Empty Interrupt Enable.

Definition at line 5078 of file STM8AF\_STM8S.h.

#### 6.5.2.226 TPR

```
struct { ... } TPR
```

CAN transmit priority register (CAN\_TPR)

#### 6.5.2.227 TSR

```
struct { ... } TSR
```

CAN transmit status register (CAN\_TSR)

#### 6.5.2.228 TTCM

`_BITS TTCM`

Time Triggered Communication Mode.

Definition at line 5024 of file STM8AF\_STM8S.h.

**6.5.2.229 TX**

`_BITS TX`

Transmit.

Definition at line 5034 of file STM8AF\_STM8S.h.

**6.5.2.230 TXFP**

`_BITS TXFP`

Transmit FIFO Priority.

Definition at line 5019 of file STM8AF\_STM8S.h.

**6.5.2.231 TXM2E**

`_BITS TXM2E`

TX Mailbox 2 enable.

Definition at line 5093 of file STM8AF\_STM8S.h.

**6.5.2.232 TXOK**

`_BITS TXOK`

Transmission OK.

Definition at line 5117 of file STM8AF\_STM8S.h.

**6.5.2.233 TXOK0**

`_BITS TXOK0`

Transmission ok for Mailbox 0.

Definition at line 5046 of file STM8AF\_STM8S.h.

**6.5.2.234 TXOK1**

`_BITS TXOK1`

Transmission ok for Mailbox 1.

Definition at line 5047 of file STM8AF\_STM8S.h.

**6.5.2.235 TXOK2**

`_BITS TXOK2`

Transmission ok for Mailbox 2.

Definition at line 5048 of file STM8AF\_STM8S.h.

**6.5.2.236 TXRQ**

`_BITS TXRQ`

Transmission mailbox request.

Definition at line 5114 of file STM8AF\_STM8S.h.

**6.5.2.237 WKUI**

`_BITS WKUI`

Wakeup Interrupt.

Definition at line 5033 of file STM8AF\_STM8S.h.

**6.5.2.238 WKUIE**

`_BITS WKUIE`

Wakeup Interrupt Enable.

Definition at line 5083 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.6 CFG\_t Struct Reference

struct for Global Configuration registers (CFG)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {
    - [\\_BITS SWD](#): 1  
*SWIM disable.*
    - [\\_BITS AL](#): 1  
*Activation level.*
    - [\\_BITS res](#): 6  
*Reserved.*
- } [GCR](#)
- Global configuration register (CFG\_GCR)*

### 6.6.1 Detailed Description

struct for Global Configuration registers (CFG)

Definition at line 6293 of file STM8AF\_STM8S.h.

### 6.6.2 Field Documentation

#### 6.6.2.1 AL

[\\_BITS AL](#)

Activation level.

Definition at line 6298 of file STM8AF\_STM8S.h.

#### 6.6.2.2 GCR

```
struct { ... } GCR
```

Global configuration register (CFG\_GCR)



## 6.6.2.3 res

`_BITS res`

Reserved.

Definition at line 6299 of file STM8AF\_STM8S.h.

## 6.6.2.4 SWD

`_BITS SWD`

SWIM disable.

Definition at line 6297 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.7 CLK\_t Struct Reference

struct for configuring/monitoring clock module (CLK)

`#include <STM8AF_STM8S.h>`

### Data Fields

- struct {
  - `_BITS HSIEN`: 1  
*High speed internal RC oscillator enable.*
  - `_BITS HSIRDY`: 1  
*High speed internal oscillator ready flag.*
  - `_BITS FHWU`: 1  
*Fast wakeup from Halt/Active-halt modes enable.*
  - `_BITS LSIEN`: 1  
*Low speed internal RC oscillator enable.*
  - `_BITS LSIRDY`: 1  
*Low speed internal oscillator ready flag.*
  - `_BITS REGAH`: 1  
*Regulator power off in Active-halt mode enable.*
  - `_BITS res`: 2  
*Reserved, must be kept cleared.*

`} ICKR`*Internal clock register (CLK\_ICKR)*

- struct {
  - [\\_BITS HSEEN](#): 1  
*High speed external crystal oscillator enable.*
  - [\\_BITS HSERDY](#): 1  
*High speed external crystal oscillator ready.*
  - [\\_BITS res](#): 6  
*Reserved, must be kept cleared.*

} [ECKR](#)

*External clock register (CLK\_ECKR)*

- uint8\_t [res](#) [1]  
*Reserved register (1B)*
- struct {
  - [\\_BITS CKM](#): 8  
*Clock master status bits.*

} [CMSR](#)

*Clock master status register (CLK\_CMSR)*

- struct {
  - [\\_BITS SWI](#): 8  
*Clock master selection bits.*

} [SWR](#)

*Clock master switch register (CLK\_SWR)*

- struct {
  - [\\_BITS SWBSY](#): 1  
*Switch busy flag.*
  - [\\_BITS SWEN](#): 1  
*Switch start/stop enable.*
  - [\\_BITS SWIEN](#): 1  
*Clock switch interrupt enable.*
  - [\\_BITS SWIF](#): 1  
*Clock switch interrupt flag.*
  - [\\_BITS res](#): 4  
*Reserved.*

} [SWCR](#)

*Switch control register (CLK\_SWCR)*

- struct {
  - [\\_BITS CPUDIV](#): 3  
*CPU clock prescaler.*
  - [\\_BITS HSIDIV](#): 2  
*High speed internal clock prescaler.*
  - [\\_BITS res](#): 3  
*Reserved, must be kept cleared.*

} [CKDIVR](#)

*Clock divider register (CLK\_CKDIVR)*

- struct {
  - [\\_BITS PCKEN\\_I2C](#): 1  
*clock enable I2C*
  - [\\_BITS PCKEN\\_SPI](#): 1  
*clock enable SPI*
  - [\\_BITS PCKEN\\_UART1](#): 1  
*clock enable UART1*
  - [\\_BITS PCKEN\\_UART2](#): 1  
*clock enable UART2*

```

_BITS PCKEN_TIM4_TIM6: 1
    clock enable TIM4/TIM6
_BITS PCKEN_TIM2_TIM5: 1
    clock enable TIM4/TIM6
_BITS PCKEN_TIM3: 1
    clock enable TIM3
_BITS PCKEN_TIM1: 1
    clock enable TIM1
} PCKENR1

```

Peripheral clock gating register 1 (CLK\_PCKENR1)

- struct {
 

```

_BITS CSSEN: 1
    Clock security system enable.
_BITS AUX: 1
    Auxiliary oscillator connected to master clock.
_BITS CSSDIE: 1
    Clock security system detection interrupt enable.
_BITS CSSD: 1
    Clock security system detection.
_BITS res: 4
    Reserved, must be kept cleared.
} CSSR

```

Clock security system register (CLK\_CSSR)

- struct {
 

```

_BITS CCOEN: 1
    Configurable clock output enable.
_BITS CCOSEL: 4
    Configurable clock output selection.
_BITS CCORDY: 1
    Configurable clock output ready.
_BITS CCOBSY: 1
    Configurable clock output busy.
_BITS res: 1
    Reserved, must be kept cleared.
} CCOR

```

Configurable clock output register (CLK\_CCOR)

- struct {
 

```

_BITS res: 2
    Reserved.
_BITS PCKEN_AWU: 1
    clock enable AWU
_BITS PCKEN_ADC: 1
    clock enable ADC
_BITS res2: 3
    Reserved.
_BITS PCKEN_CAN: 1
    clock enable CAN
} PCKENR2

```

Peripheral clock gating register 2 (CLK\_PCKENR2)

- uint8\_t res2 [1]
 

Reserved register (1B). Was CAN clock control (obsolete as of STM8 UM rev 7)
- struct {
 

```

_BITS HSITRIM: 4

```

*HSI trimming value (some devices only support 3 bits, see DS!)*  
 \_BITS res: 4  
*Reserved, must be kept cleared.*  
 } HSITRIMR

*HSI clock calibration trimming register (CLK\_HSITRIMR)*

- struct {
  - \_BITS SWIMCLK: 1  
*SWIM clock divider.*
  - \_BITS res: 7  
*Reserved.*
 } SWIMCCR

*SWIM clock control register (CLK\_SWIMCCR)*

### 6.7.1 Detailed Description

struct for configuring/monitoring clock module (CLK)

Definition at line 746 of file STM8AF\_STM8S.h.

### 6.7.2 Field Documentation

#### 6.7.2.1 AUX

\_BITS AUX

Auxiliary oscillator connected to master clock.

Definition at line 818 of file STM8AF\_STM8S.h.

#### 6.7.2.2 CCOBSY

\_BITS CCOBSY

Configurable clock output busy.

Definition at line 830 of file STM8AF\_STM8S.h.

### 6.7.2.3 CCOEN

`_BITS CCOEN`

Configurable clock output enable.

Definition at line 827 of file STM8AF\_STM8S.h.

### 6.7.2.4 CCOR

```
struct { ... } CCOR
```

Configurable clock output register (CLK\_CCOR)

### 6.7.2.5 CCORDY

`_BITS CCORDY`

Configurable clock output ready.

Definition at line 829 of file STM8AF\_STM8S.h.

### 6.7.2.6 CCOSEL

`_BITS CCOSEL`

Configurable clock output selection.

Definition at line 828 of file STM8AF\_STM8S.h.

### 6.7.2.7 CKDIVR

```
struct { ... } CKDIVR
```

Clock divider register (CLK\_CKDIVR)

#### 6.7.2.8 CKM

`_BITS` CKM

Clock master status bits.

Definition at line 774 of file STM8AF\_STM8S.h.

#### 6.7.2.9 CMSR

```
struct { ... } CMSR
```

Clock master status register (CLK\_CMSR)

#### 6.7.2.10 CPUDIV

`_BITS` CPUDIV

CPU clock prescaler.

Definition at line 796 of file STM8AF\_STM8S.h.

#### 6.7.2.11 CSSD

`_BITS` CSSD

Clock security system detection.

Definition at line 820 of file STM8AF\_STM8S.h.

#### 6.7.2.12 CSSDIE

`_BITS` CSSDIE

Clock security system detection interrupt enable.

Definition at line 819 of file STM8AF\_STM8S.h.

#### 6.7.2.13 CSSEN

`_BITS CSSEN`

Clock security system enable.

Definition at line 817 of file STM8AF\_STM8S.h.

#### 6.7.2.14 CSSR

```
struct { ... } CSSR
```

Clock security system register (CLK\_CSSR)

#### 6.7.2.15 ECKR

```
struct { ... } ECKR
```

External clock register (CLK\_ECKR)

#### 6.7.2.16 FHWU

`_BITS FHWU`

Fast wakeup from Halt/Active-halt modes enable.

Definition at line 752 of file STM8AF\_STM8S.h.

#### 6.7.2.17 HSEEN

`_BITS HSEEN`

High speed external crystal oscillator enable.

Definition at line 762 of file STM8AF\_STM8S.h.

#### 6.7.2.18 HSERDY

`_BITS HSERDY`

High speed external crystal oscillator ready.

Definition at line 763 of file STM8AF\_STM8S.h.

#### 6.7.2.19 HSIDIV

`_BITS HSIDIV`

High speed internal clock prescaler.

Definition at line 797 of file STM8AF\_STM8S.h.

#### 6.7.2.20 HSIEN

`_BITS HSIEN`

High speed internal RC oscillator enable.

Definition at line 750 of file STM8AF\_STM8S.h.

#### 6.7.2.21 HSIRDY

`_BITS HSIRDY`

High speed internal oscillator ready flag.

Definition at line 751 of file STM8AF\_STM8S.h.

#### 6.7.2.22 HSITRIM

`_BITS HSITRIM`

HSI trimming value (some devices only support 3 bits, see DS!)

Definition at line 851 of file STM8AF\_STM8S.h.



#### 6.7.2.23 HSITRIMR

```
struct { ... } HSITRIMR
```

HSI clock calibration trimming register (CLK\_HSITRIMR)

#### 6.7.2.24 ICKR

```
struct { ... } ICKR
```

Internal clock register (CLK\_ICKR)

#### 6.7.2.25 LSIEN

```
_BITS LSIEN
```

Low speed internal RC oscillator enable.

Definition at line 753 of file STM8AF\_STM8S.h.

#### 6.7.2.26 LSIRDY

```
_BITS LSIRDY
```

Low speed internal oscillator ready flag.

Definition at line 754 of file STM8AF\_STM8S.h.

#### 6.7.2.27 PCKEN\_ADC

```
_BITS PCKEN_ADC
```

clock enable ADC

Definition at line 839 of file STM8AF\_STM8S.h.

#### 6.7.2.28 PCKEN\_AWU

`_BITS PCKEN_AWU`

clock enable AWU

Definition at line 838 of file STM8AF\_STM8S.h.

#### 6.7.2.29 PCKEN\_CAN

`_BITS PCKEN_CAN`

clock enable CAN

Definition at line 841 of file STM8AF\_STM8S.h.

#### 6.7.2.30 PCKEN\_I2C

`_BITS PCKEN_I2C`

clock enable I2C

Definition at line 804 of file STM8AF\_STM8S.h.

#### 6.7.2.31 PCKEN\_SPI

`_BITS PCKEN_SPI`

clock enable SPI

Definition at line 805 of file STM8AF\_STM8S.h.

#### 6.7.2.32 PCKEN\_TIM1

`_BITS PCKEN_TIM1`

clock enable TIM1

Definition at line 811 of file STM8AF\_STM8S.h.

#### 6.7.2.33 PCKEN\_TIM2\_TIM5

`_BITS PCKEN_TIM2_TIM5`

clock enable TIM4/TIM6

Definition at line 809 of file STM8AF\_STM8S.h.

#### 6.7.2.34 PCKEN\_TIM3

`_BITS PCKEN_TIM3`

clock enable TIM3

Definition at line 810 of file STM8AF\_STM8S.h.

#### 6.7.2.35 PCKEN\_TIM4\_TIM6

`_BITS PCKEN_TIM4_TIM6`

clock enable TIM4/TIM6

Definition at line 808 of file STM8AF\_STM8S.h.

#### 6.7.2.36 PCKEN\_UART1

`_BITS PCKEN_UART1`

clock enable UART1

Definition at line 806 of file STM8AF\_STM8S.h.

#### 6.7.2.37 PCKEN\_UART2

`_BITS PCKEN_UART2`

clock enable UART2

Definition at line 807 of file STM8AF\_STM8S.h.

#### 6.7.2.38 PCKENR1

```
struct { ... } PCKENR1
```

Peripheral clock gating register 1 (CLK\_PCKENR1)

#### 6.7.2.39 PCKENR2

```
struct { ... } PCKENR2
```

Peripheral clock gating register 2 (CLK\_PCKENR2)

#### 6.7.2.40 REGAH

```
_BITS REGAH
```

Regulator power off in Active-halt mode enable.

Definition at line 755 of file STM8AF\_STM8S.h.

#### 6.7.2.41 res [1/2]

```
_BITS res
```

Reserved, must be kept cleared.

Reserved.

Definition at line 756 of file STM8AF\_STM8S.h.

#### 6.7.2.42 res [2/2]

```
uint8_t res[1]
```

Reserved register (1B)

Definition at line 769 of file STM8AF\_STM8S.h.

**6.7.2.43 res2** [1/2]

`_BITS res2`

Reserved.

Definition at line 840 of file STM8AF\_STM8S.h.

**6.7.2.44 res2** [2/2]

`uint8_t res2[1]`

Reserved register (1B). Was CAN clock control (obsolete as of STM8 UM rev 7)

Definition at line 846 of file STM8AF\_STM8S.h.

**6.7.2.45 SWBSY**

`_BITS SWBSY`

Switch busy flag.

Definition at line 786 of file STM8AF\_STM8S.h.

**6.7.2.46 SWCR**

`struct { ... } SWCR`

Switch control register (CLK\_SWCR)

**6.7.2.47 SWEN**

`_BITS SWEN`

Switch start/stop enable.

Definition at line 787 of file STM8AF\_STM8S.h.

#### 6.7.2.48 SWI

`_BITS SWI`

Clock master selection bits.

Definition at line 780 of file STM8AF\_STM8S.h.

#### 6.7.2.49 SWIEN

`_BITS SWIEN`

Clock switch interrupt enable.

Definition at line 788 of file STM8AF\_STM8S.h.

#### 6.7.2.50 SWIF

`_BITS SWIF`

Clock switch interrupt flag.

Definition at line 789 of file STM8AF\_STM8S.h.

#### 6.7.2.51 SWIMCCR

```
struct { ... } SWIMCCR
```

SWIM clock control register (CLK\_SWIMCCR)

#### 6.7.2.52 SWIMCLK

`_BITS SWIMCLK`

SWIM clock divider.

Definition at line 858 of file STM8AF\_STM8S.h.

## 6.7.2.53 SWR

```
struct { ... } SWR
```

Clock master switch register (CLK\_SWR)

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.8 EXTI\_t Struct Reference

struct for configuring external port interrupts (EXTI)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
  - [\\_BITS PAIS](#): 2  
*Port A external interrupt sensitivity bits.*
  - [\\_BITS PBIS](#): 2  
*Port B external interrupt sensitivity bits.*
  - [\\_BITS PCIS](#): 2  
*Port C external interrupt sensitivity bits.*
  - [\\_BITS PDIS](#): 2  
*Port D external interrupt sensitivity bits.*
 } [CR1](#)

*External interrupt control register 1 (EXTI\_CR1)*

- struct {
  - [\\_BITS PEIS](#): 2  
*Port E external interrupt sensitivity bits.*
  - [\\_BITS TLIS](#): 1  
*Top level interrupt sensitivity.*
  - [\\_BITS res](#): 5  
*Reserved.*
 } [CR2](#)

*External interrupt control register 2 (EXTI\_CR2)*

## 6.8.1 Detailed Description

struct for configuring external port interrupts (EXTI)

Definition at line 650 of file STM8AF\_STM8S.h.

## 6.8.2 Field Documentation

#### 6.8.2.1 CR1

```
struct { ... } CR1
```

External interrupt control register 1 (EXTI\_CR1)

#### 6.8.2.2 CR2

```
struct { ... } CR2
```

External interrupt control register 2 (EXTI\_CR2)

#### 6.8.2.3 PAIS

```
_BITS PAIS
```

Port A external interrupt sensitivity bits.

Definition at line 654 of file STM8AF\_STM8S.h.

#### 6.8.2.4 PBIS

```
_BITS PBIS
```

Port B external interrupt sensitivity bits.

Definition at line 655 of file STM8AF\_STM8S.h.

#### 6.8.2.5 PCIS

```
_BITS PCIS
```

Port C external interrupt sensitivity bits.

Definition at line 656 of file STM8AF\_STM8S.h.



#### 6.8.2.6 PDIS

`_BITS PDIS`

Port D external interrupt sensitivity bits.

Definition at line 657 of file STM8AF\_STM8S.h.

#### 6.8.2.7 PEIS

`_BITS PEIS`

Port E external interrupt sensitivity bits.

Definition at line 663 of file STM8AF\_STM8S.h.

#### 6.8.2.8 res

`_BITS res`

Reserved.

Definition at line 665 of file STM8AF\_STM8S.h.

#### 6.8.2.9 TLIS

`_BITS TLIS`

Top level interrupt sensitivity.

Definition at line 664 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.9 FLASH\_t Struct Reference

struct to control write/erase of flash memory (FLASH)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
  - [\\_BITS FIX](#): 1  
*Fixed Byte programming time.*
  - [\\_BITS IE](#): 1  
*Flash Interrupt enable.*
  - [\\_BITS AHALT](#): 1  
*Power-down in Active-halt mode.*
  - [\\_BITS HALT](#): 1  
*Power-down in Halt mode.*
  - [\\_BITS res](#): 4  
*Reserved.*

} [CR1](#)

*Flash control register 1 (FLASH\_CR1)*

- struct {
  - [\\_BITS PRG](#): 1  
*Standard block programming.*
  - [\\_BITS res](#): 3  
*Reserved.*
  - [\\_BITS FPRG](#): 1  
*Fast block programming.*
  - [\\_BITS ERASE](#): 1  
*Block erasing.*
  - [\\_BITS WPRG](#): 1  
*Word programming.*
  - [\\_BITS OPT](#): 1  
*Write option bytes.*

} [CR2](#)

*Flash control register 2 (FLASH\_CR2)*

- struct {
  - [\\_BITS NPRG](#): 1  
*Standard block programming.*
  - [\\_BITS res](#): 3  
*Reserved.*
  - [\\_BITS NFPRG](#): 1  
*Fast block programming.*
  - [\\_BITS NERASE](#): 1  
*Block erasing.*
  - [\\_BITS NWPRG](#): 1  
*Word programming.*
  - [\\_BITS NOPT](#): 1  
*Write option bytes.*

} [NCR2](#)

*Complementary flash control register 2 (FLASH\_NCR2)*

- struct {
  - [\\_BITS WPB](#): 6  
*User boot code area protection bits.*
  - [\\_BITS res](#): 2  
*Reserved.*

} [FPR](#)

*Flash protection register (FLASH\_FPR)*

- struct {
  - [\\_BITS NWPB](#): 6  
*User boot code area protection bits.*
  - [\\_BITS res](#): 2  
*Reserved.*

} [NFPR](#)

*Complementary flash protection register (FLASH\_NFPR)*

- struct {
  - [\\_BITS WR\\_PG\\_DIS](#): 1  
*Write attempted to protected page flag.*
  - [\\_BITS PUL](#): 1  
*Flash Program memory unlocked flag.*
  - [\\_BITS EOP](#): 1  
*End of programming (write or erase operation) flag.*
  - [\\_BITS DUL](#): 1  
*Data EEPROM area unlocked flag.*
  - [\\_BITS res](#): 2  
*Reserved, forced by hardware to 0.*
  - [\\_BITS HVOFF](#): 1  
*End of high voltage flag.*
  - [\\_BITS res2](#): 1  
*Reserved.*

} [IAPSR](#)

*Flash status register (FLASH\_IAPSR)*

- [uint8\\_t res](#) [2]  
*Reserved registers (2B)*
- struct {
  - [\\_BITS PUK](#): 8  
*Program memory write unlock key.*

} [PUKR](#)

*Flash program memory unprotecting key register (FLASH\_PUKR)*

- [uint8\\_t res2](#) [1]  
*Reserved register (1B)*
- struct {
  - [\\_BITS DUK](#): 8  
*Data EEPROM write unlock key.*

} [DUKR](#)

*Data EEPROM unprotection key register (FLASH\_DUKR)*

### 6.9.1 Detailed Description

struct to control write/erase of flash memory (FLASH)

Definition at line 504 of file STM8AF\_STM8S.h.

### 6.9.2 Field Documentation

### 6.9.2.1 AHALT

`_BITS` AHALT

Power-down in Active-halt mode.

Definition at line 510 of file STM8AF\_STM8S.h.

### 6.9.2.2 CR1

```
struct { ... } CR1
```

Flash control register 1 (FLASH\_CR1)

### 6.9.2.3 CR2

```
struct { ... } CR2
```

Flash control register 2 (FLASH\_CR2)

### 6.9.2.4 DUK

`_BITS` DUK

Data EEPROM write unlock key.

Definition at line 580 of file STM8AF\_STM8S.h.

### 6.9.2.5 DUKR

```
struct { ... } DUKR
```

Data EEPROM unprotection key register (FLASH\_DUKR)

### 6.9.2.6 DUL

`_BITS` DUL

Data EEPROM area unlocked flag.

Definition at line 557 of file STM8AF\_STM8S.h.

#### 6.9.2.7 EOP

`_BITS` EOP

End of programming (write or erase operation) flag.

Definition at line 556 of file STM8AF\_STM8S.h.

#### 6.9.2.8 ERASE

`_BITS` ERASE

Block erasing.

Definition at line 521 of file STM8AF\_STM8S.h.

#### 6.9.2.9 FIX

`_BITS` FIX

Fixed Byte programming time.

Definition at line 508 of file STM8AF\_STM8S.h.

#### 6.9.2.10 FPR

```
struct { ... } FPR
```

Flash protection register (FLASH\_FPR)

#### 6.9.2.11 FPRG

`_BITS` FPRG

Fast block programming.

Definition at line 520 of file STM8AF\_STM8S.h.

#### 6.9.2.12 HALT

`_BITS` HALT

Power-down in Halt mode.

Definition at line 511 of file STM8AF\_STM8S.h.

#### 6.9.2.13 HVOFF

`_BITS` HVOFF

End of high voltage flag.

Definition at line 559 of file STM8AF\_STM8S.h.

#### 6.9.2.14 IAPSR

```
struct { ... } IAPSR
```

Flash status register (FLASH\_IAPSR)

#### 6.9.2.15 IE

`_BITS` IE

Flash Interrupt enable.

Definition at line 509 of file STM8AF\_STM8S.h.

#### 6.9.2.16 NCR2

```
struct { ... } NCR2
```

Complementary flash control register 2 (FLASH\_NCR2)

#### 6.9.2.17 NERASE

`_BITS` NERASE

Block erasing.

Definition at line 532 of file STM8AF\_STM8S.h.

#### 6.9.2.18 NFPR

```
struct { ... } NFPR
```

Complementary flash protection register (FLASH\_NFPR)

#### 6.9.2.19 NFPRG

`_BITS` NFPRG

Fast block programming.

Definition at line 531 of file STM8AF\_STM8S.h.

#### 6.9.2.20 NOPT

`_BITS` NOPT

Write option bytes.

Definition at line 534 of file STM8AF\_STM8S.h.

#### 6.9.2.21 NPRG

`_BITS` NPRG

Standard block programming.

Definition at line 529 of file STM8AF\_STM8S.h.

#### 6.9.2.22 NWPB

`_BITS` NWPB

User boot code area protection bits.

Definition at line 547 of file STM8AF\_STM8S.h.

#### 6.9.2.23 NWPRG

`_BITS` NWPRG

Word programming.

Definition at line 533 of file STM8AF\_STM8S.h.

#### 6.9.2.24 OPT

`_BITS` OPT

Write option bytes.

Definition at line 523 of file STM8AF\_STM8S.h.

#### 6.9.2.25 PRG

`_BITS` PRG

Standard block programming.

Definition at line 518 of file STM8AF\_STM8S.h.

#### 6.9.2.26 PUK

`_BITS` PUK

Program memory write unlock key.

Definition at line 570 of file STM8AF\_STM8S.h.



#### 6.9.2.27 PUKR

```
struct { ... } PUKR
```

Flash program memory unprotecting key register (FLASH\_PUKR)

#### 6.9.2.28 PUL

```
_BITS PUL
```

Flash Program memory unlocked flag.

Definition at line 555 of file STM8AF\_STM8S.h.

#### 6.9.2.29 res [1/2]

```
_BITS res
```

Reserved.

Reserved, forced by hardware to 0.

Definition at line 512 of file STM8AF\_STM8S.h.

#### 6.9.2.30 res [2/2]

```
uint8_t res[2]
```

Reserved registers (2B)

Definition at line 565 of file STM8AF\_STM8S.h.

#### 6.9.2.31 res2 [1/2]

```
_BITS res2
```

Reserved.

Definition at line 560 of file STM8AF\_STM8S.h.

#### 6.9.2.32 res2 [2/2]

```
uint8_t res2[1]
```

Reserved register (1B)

Definition at line 575 of file STM8AF\_STM8S.h.

#### 6.9.2.33 WPB

```
_BITS WPB
```

User boot code area protection bits.

Definition at line 540 of file STM8AF\_STM8S.h.

#### 6.9.2.34 WPRG

```
_BITS WPRG
```

Word programming.

Definition at line 522 of file STM8AF\_STM8S.h.

#### 6.9.2.35 WR\_PG\_DIS

```
_BITS WR_PG_DIS
```

Write attempted to protected page flag.

Definition at line 554 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.10 I2C\_t Struct Reference

struct for controlling I2C module (I2C)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
    - [\\_BITS PE](#): 1  
*Peripheral enable.*
    - [\\_BITS res](#): 5  
*Reserved.*
    - [\\_BITS ENGCG](#): 1  
*General call enable.*
    - [\\_BITS NOSTRETCH](#): 1  
*Clock stretching disable (Slave mode)*
- } [CR1](#)

*I2C Control register 1 (I2C\_CR1)*

- struct {
    - [\\_BITS START](#): 1  
*Start generation.*
    - [\\_BITS STOP](#): 1  
*Stop generation.*
    - [\\_BITS ACK](#): 1  
*Acknowledge enable.*
    - [\\_BITS POS](#): 1  
*Acknowledge position (for data reception)*
    - [\\_BITS res](#): 3  
*Reserved.*
    - [\\_BITS SWRST](#): 1  
*Software reset.*
- } [CR2](#)

*I2C Control register 2 (I2C\_CR2)*

- struct {
    - [\\_BITS FREQ](#): 6  
*Peripheral clock frequency.*
    - [\\_BITS res](#): 2  
*Reserved.*
- } [FREQR](#)

*I2C Frequency register (I2C\_FREQR)*

- struct {
    - [\\_BITS ADD0](#): 1  
*Interface address [0] (in 10-bit address mode)*
    - [\\_BITS ADD](#): 7  
*Interface address [7:1].*
- } [OARL](#)

*I2C own address register low byte (I2C\_OARL)*

- struct {
  - [\\_BITS res](#): 1  
*Reserved.*
  - [\\_BITS ADD](#): 2  
*Interface address [9:8] (in 10-bit address mode)*
  - [\\_BITS res2](#): 3  
*Reserved.*
  - [\\_BITS ADDCONF](#): 1  
*Address mode configuration (must always be written as '1')*
  - [\\_BITS ADDMODE](#): 1  
*7-/10-bit addressing mode (Slave mode)*

```
} OARH
```

*I2C own address register high byte (I2C\_OARH)*

- uint8\_t [res](#) [1]

*Reserved register (1B)*

- struct {  
     [\\_BITS DATA](#): 8  
     *I2C data.*  
 } [DR](#)

*I2C data register (I2C\_DR)*

- struct {  
     [\\_BITS SB](#): 1  
     *Start bit (Mastermode)*  
     [\\_BITS ADDR](#): 1  
     *Address sent (master mode) / matched (slave mode)*  
     [\\_BITS BTF](#): 1  
     *Byte transfer finished.*  
     [\\_BITS ADD10](#): 1  
     *10-bit header sent (Master mode)*  
     [\\_BITS STOPF](#): 1  
     *Stop detection (Slave mode)*  
     [\\_BITS res](#): 1  
     *Reserved.*  
     [\\_BITS RXNE](#): 1  
     *Data register not empty (receivers)*  
     [\\_BITS TXE](#): 1  
     *Data register empty (transmitters)*  
 } [SR1](#)

*I2C Status register 1 (I2C\_SR1)*

- struct {  
     [\\_BITS BERR](#): 1  
     *Bus error.*  
     [\\_BITS ARLO](#): 1  
     *Arbitration lost (master mode)*  
     [\\_BITS AF](#): 1  
     *Acknowledge failure.*  
     [\\_BITS OVR](#): 1  
     *Overrun/underrun.*  
     [\\_BITS res](#): 1  
     *Reserved.*  
     [\\_BITS WUFH](#): 1  
     *Wakeup from Halt.*  
     [\\_BITS res2](#): 2  
     *Reserved.*  
 } [SR2](#)

*I2C Status register 2 (I2C\_SR2)*

- struct {  
     [\\_BITS MSL](#): 1  
     *Master/Slave.*  
     [\\_BITS BUSY](#): 1  
     *Bus busy.*  
     [\\_BITS TRA](#): 1  
     *Transmitter/Receiver.*  
     [\\_BITS res](#): 1

```

    Reserved.
    _BITS GENCALL: 1
    General call header (Slavemode)
    _BITS res2: 3
    Reserved.
} SR3

```

```

    I2C Status register 3 (I2C_SR3)
• struct {
    _BITS ITERREN: 1
    Error interrupt enable.
    _BITS ITEVTEN: 1
    Event interrupt enable.
    _BITS ITBUFEN: 1
    Buffer interrupt enable.
    _BITS res: 5
    Reserved.
} ITR

```

```

    I2C Interrupt register (I2C_ITR)
• struct {
    _BITS CCR: 8
    Clock control register (Master mode)
} CCRL

```

```

    I2C Clock control register low byte (I2C_CCRL)
• struct {
    _BITS CCR: 4
    Clock control register in Fast/Standard mode (Master mode)
    _BITS res: 2
    Reserved.
    _BITS DUTY: 1
    Fast mode duty cycle.
    _BITS FS: 1
    I2C master mode selection.
} CCRH

```

```

    I2C Clock control register high byte (I2C_CCRH)
• struct {
    _BITS TRISE: 6
    Maximum rise time in Fast/Standard mode (Master mode)
    _BITS res: 2
    Reserved.
} TRISER

```

```

    I2C rise time register (I2C_TRISER)
• uint8_t res2 [1]
    Reserved register (1B). Was I2C packet error checking (undocumented in STM8 UM rev 9)

```

### 6.10.1 Detailed Description

struct for controlling I2C module (I2C)

Definition at line 1356 of file STM8AF\_STM8S.h.

## 6.10.2 Field Documentation

### 6.10.2.1 ACK

`_BITS ACK`

Acknowledge enable.

Definition at line 1371 of file STM8AF\_STM8S.h.

### 6.10.2.2 ADD

`_BITS ADD`

Interface address [7:1].

Interface address [9:8] (in 10-bit address mode)

Definition at line 1388 of file STM8AF\_STM8S.h.

### 6.10.2.3 ADD0

`_BITS ADD0`

Interface address [0] (in 10-bit address mode)

Definition at line 1387 of file STM8AF\_STM8S.h.

### 6.10.2.4 ADD10

`_BITS ADD10`

10-bit header sent (Master mode)

Definition at line 1417 of file STM8AF\_STM8S.h.

#### 6.10.2.5 ADDCONF

`_BITS ADDCONF`

Address mode configuration (must always be written as '1')

Definition at line 1397 of file STM8AF\_STM8S.h.

#### 6.10.2.6 ADDMODE

`_BITS ADDMODE`

7-/10-bit addressing mode (Slave mode)

Definition at line 1398 of file STM8AF\_STM8S.h.

#### 6.10.2.7 ADDR

`_BITS ADDR`

Address sent (master mode) / matched (slave mode)

Definition at line 1415 of file STM8AF\_STM8S.h.

#### 6.10.2.8 AF

`_BITS AF`

Acknowledge failure.

Definition at line 1429 of file STM8AF\_STM8S.h.

#### 6.10.2.9 ARLO

`_BITS ARLO`

Arbitration lost (master mode)

Definition at line 1428 of file STM8AF\_STM8S.h.

#### 6.10.2.10 BERR

`_BITS` BERR

Bus error.

Definition at line 1427 of file STM8AF\_STM8S.h.

#### 6.10.2.11 BTF

`_BITS` BTF

Byte transfer finished.

Definition at line 1416 of file STM8AF\_STM8S.h.

#### 6.10.2.12 BUSY

`_BITS` BUSY

Bus busy.

Definition at line 1440 of file STM8AF\_STM8S.h.

#### 6.10.2.13 CCR

`_BITS` CCR

Clock control register (Master mode)

Clock control register in Fast/Standard mode (Master mode)

Definition at line 1459 of file STM8AF\_STM8S.h.

#### 6.10.2.14 CCRH

```
struct { ... } CCRH
```

I2C Clock control register high byte (I2C\_CCRH)



#### 6.10.2.15 CCRL

```
struct { ... } CCRL
```

I2C Clock control register low byte (I2C\_CCRL)

#### 6.10.2.16 CR1

```
struct { ... } CR1
```

I2C Control register 1 (I2C\_CR1)

#### 6.10.2.17 CR2

```
struct { ... } CR2
```

I2C Control register 2 (I2C\_CR2)

#### 6.10.2.18 DATA

```
_BITS DATA
```

I2C data.

Definition at line 1408 of file STM8AF\_STM8S.h.

#### 6.10.2.19 DR

```
struct { ... } DR
```

I2C data register (I2C\_DR)

#### 6.10.2.20 DUTY

```
_BITS DUTY
```

Fast mode duty cycle.

Definition at line 1467 of file STM8AF\_STM8S.h.

**6.10.2.21 ENG**

`_BITS` ENG

General call enable.

Definition at line 1362 of file STM8AF\_STM8S.h.

**6.10.2.22 FREQ**

`_BITS` FREQ

Peripheral clock frequency.

Definition at line 1380 of file STM8AF\_STM8S.h.

**6.10.2.23 FREQR**

```
struct { ... } FREQR
```

I2C Frequency register (I2C\_FREQR)

**6.10.2.24 FS**

`_BITS` FS

I2C master mode selection.

Definition at line 1468 of file STM8AF\_STM8S.h.

**6.10.2.25 GENCALL**

`_BITS` GENCALL

General call header (Slavemode)

Definition at line 1443 of file STM8AF\_STM8S.h.

#### 6.10.2.26 ITBUFEN

`_BITS ITBUFEN`

Buffer interrupt enable.

Definition at line 1452 of file STM8AF\_STM8S.h.

#### 6.10.2.27 ITERREN

`_BITS ITERREN`

Error interrupt enable.

Definition at line 1450 of file STM8AF\_STM8S.h.

#### 6.10.2.28 ITEVTEN

`_BITS ITEVTEN`

Event interrupt enable.

Definition at line 1451 of file STM8AF\_STM8S.h.

#### 6.10.2.29 ITR

```
struct { ... } ITR
```

I2C Interrupt register (I2C\_ITR)

#### 6.10.2.30 MSL

`_BITS MSL`

Master/Slave.

Definition at line 1439 of file STM8AF\_STM8S.h.

#### 6.10.2.31 NOSTRETCH

`_BITS` NOSTRETCH

Clock stretching disable (Slave mode)

Definition at line 1363 of file STM8AF\_STM8S.h.

#### 6.10.2.32 OARH

```
struct { ... } OARH
```

I2C own address register high byte (I2C\_OARH)

#### 6.10.2.33 OARL

```
struct { ... } OARL
```

I2C own address register low byte (I2C\_OARL)

#### 6.10.2.34 OVR

`_BITS` OVR

Overrun/underrun.

Definition at line 1430 of file STM8AF\_STM8S.h.

#### 6.10.2.35 PE

`_BITS` PE

Peripheral enable.

Definition at line 1360 of file STM8AF\_STM8S.h.

**6.10.2.36 POS**

`_BITS` POS

Acknowledge position (for data reception)

Definition at line 1372 of file STM8AF\_STM8S.h.

**6.10.2.37 res** [1/2]

`_BITS` res

Reserved.

Definition at line 1361 of file STM8AF\_STM8S.h.

**6.10.2.38 res** [2/2]

`uint8_t` res[1]

Reserved register (1B)

Definition at line 1403 of file STM8AF\_STM8S.h.

**6.10.2.39 res2** [1/2]

`_BITS` res2

Reserved.

Definition at line 1396 of file STM8AF\_STM8S.h.

**6.10.2.40 res2** [2/2]

`uint8_t` res2[1]

Reserved register (1B). Was I2C packet error checking (undocumented in STM8 UM rev 9)

Definition at line 1480 of file STM8AF\_STM8S.h.

#### 6.10.2.41 RXNE

`_BITS` RXNE

Data register not empty (receivers)

Definition at line 1420 of file STM8AF\_STM8S.h.

#### 6.10.2.42 SB

`_BITS` SB

Start bit (Mastermode)

Definition at line 1414 of file STM8AF\_STM8S.h.

#### 6.10.2.43 SR1

```
struct { ... } SR1
```

I2C Status register 1 (I2C\_SR1)

#### 6.10.2.44 SR2

```
struct { ... } SR2
```

I2C Status register 2 (I2C\_SR2)

#### 6.10.2.45 SR3

```
struct { ... } SR3
```

I2C Status register 3 (I2C\_SR3)

#### 6.10.2.46 START

`_BITS` START

Start generation.

Definition at line 1369 of file STM8AF\_STM8S.h.

#### 6.10.2.47 STOP

`_BITS STOP`

Stop generation.

Definition at line 1370 of file STM8AF\_STM8S.h.

#### 6.10.2.48 STOPF

`_BITS STOPF`

Stop detection (Slave mode)

Definition at line 1418 of file STM8AF\_STM8S.h.

#### 6.10.2.49 SWRST

`_BITS SWRST`

Software reset.

Definition at line 1374 of file STM8AF\_STM8S.h.

#### 6.10.2.50 TRA

`_BITS TRA`

Transmitter/Receiver.

Definition at line 1441 of file STM8AF\_STM8S.h.

#### 6.10.2.51 TRISE

`_BITS TRISE`

Maximum rise time in Fast/Standard mode (Master mode)

Definition at line 1474 of file STM8AF\_STM8S.h.

#### 6.10.2.52 TRISER

```
struct { ... } TRISER
```

I2C rise time register (I2C\_TRISER)

#### 6.10.2.53 TXE

`_BITS` TXE

Data register empty (transmitters)

Definition at line 1421 of file STM8AF\_STM8S.h.

#### 6.10.2.54 WUFH

`_BITS` WUFH

Wakeup from Halt.

Definition at line 1432 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.11 ITC\_t Struct Reference

struct for setting interrupt Priority (ITC)

```
#include <STM8AF_STM8S.h>
```



## Data Fields

- struct {
  - [\\_BITS res](#): 2  
*Reserved (TLI always highest prio)*
  - [\\_BITS VECT1SPR](#): 2  
*interrupt priority vector 1*
  - [\\_BITS VECT2SPR](#): 2  
*interrupt priority vector 2*
  - [\\_BITS VECT3SPR](#): 2  
*interrupt priority vector 3*

} [SPR1](#)

*interrupt priority register 1 (ITC\_SPR1)*

- struct {
  - [\\_BITS VECT4SPR](#): 2  
*interrupt priority vector 4*
  - [\\_BITS VECT5SPR](#): 2  
*interrupt priority vector 5*
  - [\\_BITS VECT6SPR](#): 2  
*interrupt priority vector 6*
  - [\\_BITS VECT7SPR](#): 2  
*interrupt priority vector 7*

} [SPR2](#)

*interrupt priority register 2 (ITC\_SPR2)*

- struct {
  - [\\_BITS VECT8SPR](#): 2  
*interrupt priority vector 8*
  - [\\_BITS VECT9SPR](#): 2  
*interrupt priority vector 9*
  - [\\_BITS VECT10SPR](#): 2  
*interrupt priority vector 10*
  - [\\_BITS VECT11SPR](#): 2  
*interrupt priority vector 11*

} [SPR3](#)

*interrupt priority register 3 (ITC\_SPR3)*

- struct {
  - [\\_BITS VECT12SPR](#): 2  
*interrupt priority vector 12*
  - [\\_BITS VECT13SPR](#): 2  
*interrupt priority vector 13*
  - [\\_BITS VECT14SPR](#): 2  
*interrupt priority vector 14*
  - [\\_BITS VECT15SPR](#): 2  
*interrupt priority vector 15*

} [SPR4](#)

*interrupt priority register 4 (ITC\_SPR4)*

- struct {
  - [\\_BITS VECT16SPR](#): 2  
*interrupt priority vector 16*
  - [\\_BITS VECT17SPR](#): 2  
*interrupt priority vector 17*
  - [\\_BITS VECT18SPR](#): 2  
*interrupt priority vector 18*

```

    _BITS VECT19SPR: 2
        interrupt priority vector 19
} SPR5

```

*interrupt priority register 5 (ITC\_SPR5)*

```

• struct {
    _BITS VECT20SPR: 2
        interrupt priority vector 20
    _BITS VECT21SPR: 2
        interrupt priority vector 21
    _BITS VECT22SPR: 2
        interrupt priority vector 22
    _BITS VECT23SPR: 2
        interrupt priority vector 23
} SPR6

```

*interrupt priority register 6 (ITC\_SPR6)*

```

• struct {
    _BITS VECT24SPR: 2
        interrupt priority vector 24
    _BITS VECT25SPR: 2
        interrupt priority vector 25
    _BITS VECT26SPR: 2
        interrupt priority vector 26
    _BITS VECT27SPR: 2
        interrupt priority vector 27
} SPR7

```

*interrupt priority register 7 (ITC\_SPR7)*

```

• struct {
    _BITS VECT28SPR: 2
        interrupt priority vector 28
    _BITS VECT29SPR: 2
        interrupt priority vector 29
    _BITS res: 4
        Reserved.
} SPR8

```

*interrupt priority register 8 (ITC\_SPR8)*

### 6.11.1 Detailed Description

struct for setting interrupt Priority (ITC)

Definition at line 6326 of file STM8AF\_STM8S.h.

### 6.11.2 Field Documentation

### 6.11.2.1 res

`_BITS res`

Reserved (TLI always highest prio)

Reserved.

Definition at line 6330 of file STM8AF\_STM8S.h.

### 6.11.2.2 SPR1

```
struct { ... } SPR1
```

interrupt priority register 1 (ITC\_SPR1)

### 6.11.2.3 SPR2

```
struct { ... } SPR2
```

interrupt priority register 2 (ITC\_SPR2)

### 6.11.2.4 SPR3

```
struct { ... } SPR3
```

interrupt priority register 3 (ITC\_SPR3)

### 6.11.2.5 SPR4

```
struct { ... } SPR4
```

interrupt priority register 4 (ITC\_SPR4)

### 6.11.2.6 SPR5

```
struct { ... } SPR5
```

interrupt priority register 5 (ITC\_SPR5)

#### 6.11.2.7 SPR6

```
struct { ... } SPR6
```

interrupt priority register 6 (ITC\_SPR6)

#### 6.11.2.8 SPR7

```
struct { ... } SPR7
```

interrupt priority register 7 (ITC\_SPR7)

#### 6.11.2.9 SPR8

```
struct { ... } SPR8
```

interrupt priority register 8 (ITC\_SPR8)

#### 6.11.2.10 VECT10SPR

```
\_BITS VECT10SPR
```

interrupt priority vector 10

Definition at line 6350 of file STM8AF\_STM8S.h.

#### 6.11.2.11 VECT11SPR

```
\_BITS VECT11SPR
```

interrupt priority vector 11

Definition at line 6351 of file STM8AF\_STM8S.h.

#### 6.11.2.12 VECT12SPR

```
\_BITS VECT12SPR
```

interrupt priority vector 12

Definition at line 6357 of file STM8AF\_STM8S.h.

#### 6.11.2.13 VECT13SPR

`_BITS` VECT13SPR

interrupt priority vector 13

Definition at line 6358 of file STM8AF\_STM8S.h.

#### 6.11.2.14 VECT14SPR

`_BITS` VECT14SPR

interrupt priority vector 14

Definition at line 6359 of file STM8AF\_STM8S.h.

#### 6.11.2.15 VECT15SPR

`_BITS` VECT15SPR

interrupt priority vector 15

Definition at line 6360 of file STM8AF\_STM8S.h.

#### 6.11.2.16 VECT16SPR

`_BITS` VECT16SPR

interrupt priority vector 16

Definition at line 6366 of file STM8AF\_STM8S.h.

#### 6.11.2.17 VECT17SPR

`_BITS` VECT17SPR

interrupt priority vector 17

Definition at line 6367 of file STM8AF\_STM8S.h.

**6.11.2.18 VECT18SPR**

`_BITS VECT18SPR`

interrupt priority vector 18

Definition at line 6368 of file STM8AF\_STM8S.h.

**6.11.2.19 VECT19SPR**

`_BITS VECT19SPR`

interrupt priority vector 19

Definition at line 6369 of file STM8AF\_STM8S.h.

**6.11.2.20 VECT1SPR**

`_BITS VECT1SPR`

interrupt priority vector 1

Definition at line 6331 of file STM8AF\_STM8S.h.

**6.11.2.21 VECT20SPR**

`_BITS VECT20SPR`

interrupt priority vector 20

Definition at line 6375 of file STM8AF\_STM8S.h.

**6.11.2.22 VECT21SPR**

`_BITS VECT21SPR`

interrupt priority vector 21

Definition at line 6376 of file STM8AF\_STM8S.h.

#### 6.11.2.23 VECT22SPR

`_BITS` VECT22SPR

interrupt priority vector 22

Definition at line 6377 of file STM8AF\_STM8S.h.

#### 6.11.2.24 VECT23SPR

`_BITS` VECT23SPR

interrupt priority vector 23

Definition at line 6378 of file STM8AF\_STM8S.h.

#### 6.11.2.25 VECT24SPR

`_BITS` VECT24SPR

interrupt priority vector 24

Definition at line 6384 of file STM8AF\_STM8S.h.

#### 6.11.2.26 VECT25SPR

`_BITS` VECT25SPR

interrupt priority vector 25

Definition at line 6385 of file STM8AF\_STM8S.h.

#### 6.11.2.27 VECT26SPR

`_BITS` VECT26SPR

interrupt priority vector 26

Definition at line 6386 of file STM8AF\_STM8S.h.

**6.11.2.28 VECT27SPR**

`_BITS VECT27SPR`

interrupt priority vector 27

Definition at line 6387 of file STM8AF\_STM8S.h.

**6.11.2.29 VECT28SPR**

`_BITS VECT28SPR`

interrupt priority vector 28

Definition at line 6393 of file STM8AF\_STM8S.h.

**6.11.2.30 VECT29SPR**

`_BITS VECT29SPR`

interrupt priority vector 29

Definition at line 6394 of file STM8AF\_STM8S.h.

**6.11.2.31 VECT2SPR**

`_BITS VECT2SPR`

interrupt priority vector 2

Definition at line 6332 of file STM8AF\_STM8S.h.

**6.11.2.32 VECT3SPR**

`_BITS VECT3SPR`

interrupt priority vector 3

Definition at line 6333 of file STM8AF\_STM8S.h.



#### 6.11.2.33 VECT4SPR

`_BITS VECT4SPR`

interrupt priority vector 4

Definition at line 6339 of file STM8AF\_STM8S.h.

#### 6.11.2.34 VECT5SPR

`_BITS VECT5SPR`

interrupt priority vector 5

Definition at line 6340 of file STM8AF\_STM8S.h.

#### 6.11.2.35 VECT6SPR

`_BITS VECT6SPR`

interrupt priority vector 6

Definition at line 6341 of file STM8AF\_STM8S.h.

#### 6.11.2.36 VECT7SPR

`_BITS VECT7SPR`

interrupt priority vector 7

Definition at line 6342 of file STM8AF\_STM8S.h.

#### 6.11.2.37 VECT8SPR

`_BITS VECT8SPR`

interrupt priority vector 8

Definition at line 6348 of file STM8AF\_STM8S.h.

### 6.11.2.38 VECT9SPR

`_BITS` VECT9SPR

interrupt priority vector 9

Definition at line 6349 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.12 IWDG\_t Struct Reference

struct for access to Independent Timeout Watchdog registers (IWDG)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {  
    `_BITS` `KEY`: 8  
        IWDG Key.  
} `KR`  
  
    IWDG Key register (IWDG\_KR)
- struct {  
    `_BITS` `PRE`: 3  
        Prescaler divider.  
    `_BITS` `res`: 5  
        Reserved.  
} `PR`  
  
    IWDG Prescaler register (IWDG\_PR)
- struct {  
    `_BITS` `RL`: 8  
        IWDG Reload value.  
} `RLR`  
  
    IWDG Reload register (IWDG\_RLR)

### 6.12.1 Detailed Description

struct for access to Independent Timeout Watchdog registers (IWDG)

Definition at line 1046 of file STM8AF\_STM8S.h.

### 6.12.2 Field Documentation

#### 6.12.2.1 KEY

`_BITS` KEY

IWDG Key.

Definition at line 1050 of file STM8AF\_STM8S.h.

#### 6.12.2.2 KR

`struct { ... } KR`

IWDG Key register (IWDG\_KR)

#### 6.12.2.3 PR

`struct { ... } PR`

IWDG Prescaler register (IWDG\_PR)

#### 6.12.2.4 PRE

`_BITS` PRE

Prescaler divider.

Definition at line 1056 of file STM8AF\_STM8S.h.

#### 6.12.2.5 res

`_BITS` res

Reserved.

Definition at line 1057 of file STM8AF\_STM8S.h.

## 6.12.2.6 RL

`_BITS` RL

IWDG Reload value.

Definition at line 1063 of file STM8AF\_STM8S.h.

## 6.12.2.7 RLR

`struct { ... }` RLR

IWDG Reload register (IWDG\_RLR)

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.13 PORT\_t Struct Reference

structure for controlling pins in PORT mode (PORTx, x=A..I)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
    - `_BITS PIN0`: 1  
*pin 0 output control*
    - `_BITS PIN1`: 1  
*pin 1 output control*
    - `_BITS PIN2`: 1  
*pin 2 output control*
    - `_BITS PIN3`: 1  
*pin 3 output control*
    - `_BITS PIN4`: 1  
*pin 4 output control*
    - `_BITS PIN5`: 1  
*pin 5 output control*
    - `_BITS PIN6`: 1  
*pin 6 output control*
    - `_BITS PIN7`: 1  
*pin 7 output control*
- } `ODR`

*Port x output data register (Px\_ODR)*

- struct {
  - [\\_BITS PIN0](#): 1  
*pin 0 input control*
  - [\\_BITS PIN1](#): 1  
*pin 1 input control*
  - [\\_BITS PIN2](#): 1  
*pin 2 input control*
  - [\\_BITS PIN3](#): 1  
*pin 3 input control*
  - [\\_BITS PIN4](#): 1  
*pin 4 input control*
  - [\\_BITS PIN5](#): 1  
*pin 5 input control*
  - [\\_BITS PIN6](#): 1  
*pin 6 input control*
  - [\\_BITS PIN7](#): 1  
*pin 7 input control*

} [IDR](#)

*Port x input data register (Px\_IDR)*

- struct {
  - [\\_BITS PIN0](#): 1  
*pin 0 direction control*
  - [\\_BITS PIN1](#): 1  
*pin 1 direction control*
  - [\\_BITS PIN2](#): 1  
*pin 2 direction control*
  - [\\_BITS PIN3](#): 1  
*pin 3 direction control*
  - [\\_BITS PIN4](#): 1  
*pin 4 direction control*
  - [\\_BITS PIN5](#): 1  
*pin 5 direction control*
  - [\\_BITS PIN6](#): 1  
*pin 6 direction control*
  - [\\_BITS PIN7](#): 1  
*pin 7 direction control*

} [DDR](#)

*Port x data direction data register (Px\_DDR)*

- struct {
  - [\\_BITS PIN0](#): 1  
*pin 0 control register 1*
  - [\\_BITS PIN1](#): 1  
*pin 1 control register 1*
  - [\\_BITS PIN2](#): 1  
*pin 2 control register 1*
  - [\\_BITS PIN3](#): 1  
*pin 3 control register 1*
  - [\\_BITS PIN4](#): 1  
*pin 4 control register 1*
  - [\\_BITS PIN5](#): 1  
*pin 5 control register 1*
  - [\\_BITS PIN6](#): 1  
*pin 6 control register 1*
  - [\\_BITS PIN7](#): 1  
*pin 7 control register 1*

} [CR1](#)

*Port x control register 1 (Px\_CR1)*

- struct {
  - [\\_BITS PIN0](#): 1  
*pin 0 control register 2*
  - [\\_BITS PIN1](#): 1  
*pin 1 control register 2*
  - [\\_BITS PIN2](#): 1  
*pin 2 control register 2*
  - [\\_BITS PIN3](#): 1  
*pin 3 control register 2*
  - [\\_BITS PIN4](#): 1  
*pin 4 control register 2*
  - [\\_BITS PIN5](#): 1  
*pin 5 control register 2*
  - [\\_BITS PIN6](#): 1  
*pin 6 control register 2*
  - [\\_BITS PIN7](#): 1  
*pin 7 control register 2*

[} CR2](#)

*Port x control register 1 (Px\_CR2)*

### 6.13.1 Detailed Description

structure for controlling pins in PORT mode (PORTx, x=A..I)

Definition at line 324 of file STM8AF\_STM8S.h.

### 6.13.2 Field Documentation

#### 6.13.2.1 CR1

```
struct { ... } CR1
```

Port x control register 1 (Px\_CR1)

#### 6.13.2.2 CR2

```
struct { ... } CR2
```

Port x control register 1 (Px\_CR2)

#### 6.13.2.3 DDR

```
struct { ... } DDR
```

Port x data direction data register (Px\_DDR)

#### 6.13.2.4 IDR

```
struct { ... } IDR
```

Port x input data register (Px\_IDR)

#### 6.13.2.5 ODR

```
struct { ... } ODR
```

Port x output data register (Px\_ODR)

#### 6.13.2.6 PIN0

```
_BITS PIN0
```

pin 0 output control

pin 0 control register 2

pin 0 control register 1

pin 0 direction control

pin 0 input control

Definition at line 328 of file STM8AF\_STM8S.h.

#### 6.13.2.7 PIN1

```
_BITS PIN1
```

pin 1 output control

pin 1 control register 2

pin 1 control register 1

pin 1 direction control

pin 1 input control

Definition at line 329 of file STM8AF\_STM8S.h.

#### 6.13.2.8 PIN2

`_BITS` PIN2

pin 2 output control

pin 2 control register 2

pin 2 control register 1

pin 2 direction control

pin 2 input control

Definition at line 330 of file STM8AF\_STM8S.h.

#### 6.13.2.9 PIN3

`_BITS` PIN3

pin 3 output control

pin 3 control register 2

pin 3 control register 1

pin 3 direction control

pin 3 input control

Definition at line 331 of file STM8AF\_STM8S.h.

#### 6.13.2.10 PIN4

`_BITS` PIN4

pin 4 output control

pin 4 control register 2

pin 4 control register 1

pin 4 direction control

pin 4 input control

Definition at line 332 of file STM8AF\_STM8S.h.



#### 6.13.2.11 PIN5

[\\_BITS](#) PIN5

pin 5 output control

pin 5 control register 2

pin 5 control register 1

pin 5 direction control

pin 5 input control

Definition at line 333 of file STM8AF\_STM8S.h.

#### 6.13.2.12 PIN6

[\\_BITS](#) PIN6

pin 6 output control

pin 6 control register 2

pin 6 control register 1

pin 6 direction control

pin 6 input control

Definition at line 334 of file STM8AF\_STM8S.h.

#### 6.13.2.13 PIN7

[\\_BITS](#) PIN7

pin 7 output control

pin 7 control register 2

pin 7 control register 1

pin 7 direction control

pin 7 input control

Definition at line 335 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.14 RST\_t Struct Reference

struct for determining reset source (RST)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {
    - [\\_BITS WWDGF](#): 1  
*Window Watchdog reset flag.*
    - [\\_BITS IWDGF](#): 1  
*Independent Watchdog reset flag.*
    - [\\_BITS ILLOPF](#): 1  
*Illegal opcode reset flag.*
    - [\\_BITS SWIMF](#): 1  
*SWIM reset flag.*
    - [\\_BITS EMCF](#): 1  
*EMC reset flag.*
    - [\\_BITS res](#): 3  
*Reserved.*
- } [SR](#)

*Reset status register (RST\_SR)*

### 6.14.1 Detailed Description

struct for determining reset source (RST)

Definition at line 710 of file STM8AF\_STM8S.h.

### 6.14.2 Field Documentation

#### 6.14.2.1 EMCF

[\\_BITS EMCF](#)

EMC reset flag.

Definition at line 718 of file STM8AF\_STM8S.h.

#### 6.14.2.2 ILLOPF

`_BITS` ILLOPF

Illegal opcode reset flag.

Definition at line 716 of file STM8AF\_STM8S.h.

#### 6.14.2.3 IWDGF

`_BITS` IWDGF

Independent Watchdog reset flag.

Definition at line 715 of file STM8AF\_STM8S.h.

#### 6.14.2.4 res

`_BITS` res

Reserved.

Definition at line 719 of file STM8AF\_STM8S.h.

#### 6.14.2.5 SR

```
struct { ... } SR
```

Reset status register (RST\_SR)

#### 6.14.2.6 SWIMF

`_BITS` SWIMF

SWIM reset flag.

Definition at line 717 of file STM8AF\_STM8S.h.

### 6.14.2.7 WWDGF

`__BITS` WWDGF

Window Watchdog reset flag.

Definition at line 714 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.15 SPI\_t Struct Reference

struct for controlling SPI module (SPI)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {
  - `__BITS CPHA`: 1  
*Clock phase.*
  - `__BITS CPOL`: 1  
*Clock polarity.*
  - `__BITS MSTR`: 1  
*Master/slave selection.*
  - `__BITS BR`: 3  
*Baudrate control.*
  - `__BITS SPE`: 1  
*SPI enable.*
  - `__BITS LSBFIRST`: 1  
*Frame format.*
 } `CR1`

*SPI control register 1 (SPI\_CR1)*

- struct {
  - `__BITS SSI`: 1  
*Internal slave select.*
  - `__BITS SSM`: 1  
*Software slave management.*
  - `__BITS RXONLY`: 1  
*Receive only.*
  - `__BITS res`: 1  
*Reserved.*
  - `__BITS CRCNEXT`: 1  
*Transmit CRC next.*
  - `__BITS CRCEN`: 1  
*Hardware CRC calculation enable.*
  - `__BITS BDOE`: 1  
*Input/Output enable in bidirectional mode.*
  - `__BITS BDM`: 1  
*Bidirectional data mode enable.*
 } `CR2`

*SPI control register 2 (SPI\_CR2)*

- struct {
    - [\\_BITS res](#): 4  
*Reserved.*
    - [\\_BITS WKIE](#): 1  
*Wakeup interrupt enable.*
    - [\\_BITS ERRIE](#): 1  
*Error interrupt enable.*
    - [\\_BITS RXIE](#): 1  
*Rx buffer not empty interrupt enable.*
    - [\\_BITS TXIE](#): 1  
*Tx buffer empty interrupt enable.*
- } [ICR](#)

*SPI interrupt control register (SPI\_ICR)*

- struct {
    - [\\_BITS RXNE](#): 1  
*Receive buffer not empty.*
    - [\\_BITS TXE](#): 1  
*Transmit buffer empty.*
    - [\\_BITS res](#): 1  
*Reserved.*
    - [\\_BITS WKUP](#): 1  
*Wakeup flag.*
    - [\\_BITS CRCERR](#): 1  
*CRC error flag.*
    - [\\_BITS MODF](#): 1  
*Mode fault.*
    - [\\_BITS OVR](#): 1  
*Overrun flag.*
    - [\\_BITS BSY](#): 1  
*Busy flag.*
- } [SR](#)

*SPI status register (SPI\_SR)*

- struct {
    - [\\_BITS DATA](#): 8  
*SPI data.*
- } [DR](#)

*SPI data register (SPI\_DR)*

- struct {
    - [\\_BITS CRCPOLY](#): 8  
*CRC polynomial register.*
- } [CRCPR](#)

*SPI CRC polynomial register (SPI\_CRCPR)*

- struct {
    - [\\_BITS RxCRC](#): 8  
*Rx CRC Register.*
- } [RXCRCR](#)

*SPI Rx CRC register (SPI\_RXCR)*

- struct {
    - [\\_BITS TxCRC](#): 8  
*Tx CRC register.*
- } [TXCRCR](#)

*SPI Tx CRC register (SPI\_TXCRCR)*

### 6.15.1 Detailed Description

struct for controlling SPI module (SPI)

Definition at line 1213 of file STM8AF\_STM8S.h.

### 6.15.2 Field Documentation

#### 6.15.2.1 BDM

`_BITS` BDM

Bidirectional data mode enable.

Definition at line 1235 of file STM8AF\_STM8S.h.

#### 6.15.2.2 BDOE

`_BITS` BDOE

Input/Output enable in bidirectional mode.

Definition at line 1234 of file STM8AF\_STM8S.h.

#### 6.15.2.3 BR

`_BITS` BR

Baudrate control.

Definition at line 1220 of file STM8AF\_STM8S.h.

#### 6.15.2.4 BSY

`_BITS` BSY

Busy flag.

Definition at line 1258 of file STM8AF\_STM8S.h.

#### 6.15.2.5 CPHA

`_BITS` CPHA

Clock phase.

Definition at line 1217 of file STM8AF\_STM8S.h.

#### 6.15.2.6 CPOL

`_BITS` CPOL

Clock polarity.

Definition at line 1218 of file STM8AF\_STM8S.h.

#### 6.15.2.7 CR1

```
struct { ... } CR1
```

SPI control register 1 (SPI\_CR1)

#### 6.15.2.8 CR2

```
struct { ... } CR2
```

SPI control register 2 (SPI\_CR2)

#### 6.15.2.9 CRCEN

`_BITS CRCEN`

Hardware CRC calculation enable.

Definition at line 1233 of file STM8AF\_STM8S.h.

#### 6.15.2.10 CRCERR

`_BITS CRCERR`

CRC error flag.

Definition at line 1255 of file STM8AF\_STM8S.h.

#### 6.15.2.11 CRCNEXT

`_BITS CRCNEXT`

Transmit CRC next.

Definition at line 1232 of file STM8AF\_STM8S.h.

#### 6.15.2.12 CRCPOLY

`_BITS CRCPOLY`

CRC polynomial register.

Definition at line 1270 of file STM8AF\_STM8S.h.

#### 6.15.2.13 CRCPR

```
struct { ... } CRCPR
```

SPI CRC polynomial register (SPI\_CRCPR)



#### 6.15.2.14 DATA

`_BITS` DATA

SPI data.

Definition at line 1264 of file STM8AF\_STM8S.h.

#### 6.15.2.15 DR

```
struct { ... } DR
```

SPI data register (SPI\_DR)

#### 6.15.2.16 ERRIE

`_BITS` ERRIE

Error interrupt enable.

Definition at line 1243 of file STM8AF\_STM8S.h.

#### 6.15.2.17 ICR

```
struct { ... } ICR
```

SPI interrupt control register (SPI\_ICR)

#### 6.15.2.18 LSBFIRST

`_BITS` LSBFIRST

Frame format.

Definition at line 1222 of file STM8AF\_STM8S.h.

**6.15.2.19 MODF**

`_BITS` MODF

Mode fault.

Definition at line 1256 of file STM8AF\_STM8S.h.

**6.15.2.20 MSTR**

`_BITS` MSTR

Master/slave selection.

Definition at line 1219 of file STM8AF\_STM8S.h.

**6.15.2.21 OVR**

`_BITS` OVR

Overrun flag.

Definition at line 1257 of file STM8AF\_STM8S.h.

**6.15.2.22 res**

`_BITS` res

Reserved.

Definition at line 1231 of file STM8AF\_STM8S.h.

**6.15.2.23 RxCRC**

`_BITS` RxCRC

Rx CRC Register.

Definition at line 1276 of file STM8AF\_STM8S.h.

#### 6.15.2.24 RXCRCR

```
struct { ... } RXCRCR
```

SPI Rx CRC register (SPI\_RXCRCCR)

#### 6.15.2.25 RXIE

```
_BITS RXIE
```

Rx buffer not empty interrupt enable.

Definition at line 1244 of file STM8AF\_STM8S.h.

#### 6.15.2.26 RXNE

```
_BITS RXNE
```

Receive buffer not empty.

Definition at line 1251 of file STM8AF\_STM8S.h.

#### 6.15.2.27 RXONLY

```
_BITS RXONLY
```

Receive only.

Definition at line 1230 of file STM8AF\_STM8S.h.

#### 6.15.2.28 SPE

```
_BITS SPE
```

SPI enable.

Definition at line 1221 of file STM8AF\_STM8S.h.

**6.15.2.29 SR**

```
struct { ... } SR
```

SPI status register (SPI\_SR)

**6.15.2.30 SSI**

```
_BITS SSI
```

Internal slave select.

Definition at line 1228 of file STM8AF\_STM8S.h.

**6.15.2.31 SSM**

```
_BITS SSM
```

Software slave management.

Definition at line 1229 of file STM8AF\_STM8S.h.

**6.15.2.32 TxCRC**

```
_BITS TxCRC
```

Tx CRC register.

Definition at line 1282 of file STM8AF\_STM8S.h.

**6.15.2.33 TXCRCR**

```
struct { ... } TXCRCR
```

SPI Tx CRC register (SPI\_TXCRCR)

#### 6.15.2.34 TXE

`_BITS TXE`

Transmit buffer empty.

Definition at line 1252 of file STM8AF\_STM8S.h.

#### 6.15.2.35 TXIE

`_BITS TXIE`

Tx buffer empty interrupt enable.

Definition at line 1245 of file STM8AF\_STM8S.h.

#### 6.15.2.36 WKIE

`_BITS WKIE`

Wakeup interrupt enable.

Definition at line 1242 of file STM8AF\_STM8S.h.

#### 6.15.2.37 WKUP

`_BITS WKUP`

Wakeup flag.

Definition at line 1254 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h`

## 6.16 TIM1\_t Struct Reference

struct for controlling 16-Bit Timer 1 (TIM1)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
    - [\\_BITS CEN](#): 1  
*Counter enable.*
    - [\\_BITS UDIS](#): 1  
*Update disable.*
    - [\\_BITS URS](#): 1  
*Update request source.*
    - [\\_BITS OPM](#): 1  
*One-pulse mode.*
    - [\\_BITS DIR](#): 1  
*Direction.*
    - [\\_BITS CMS](#): 2  
*Center-aligned mode selection.*
    - [\\_BITS ARPE](#): 1  
*Auto-reload preload enable.*
- [} CR1](#)
- TIM1 Control register 1 (TIM1\_CR1)*
- struct {
    - [\\_BITS CCPC](#): 1  
*Capture/compare preloaded control.*
    - [\\_BITS res](#): 1  
*Reserved, forced by hardware to 0.*
    - [\\_BITS COMS](#): 1  
*Capture/compare control update selection.*
    - [\\_BITS res2](#): 1  
*Reserved, must be kept cleared.*
    - [\\_BITS MMS](#): 3  
*Master mode selection.*
    - [\\_BITS res3](#): 1  
*Reserved.*
- [} CR2](#)
- TIM1 Control register 2 (TIM1\_CR2)*
- struct {
    - [\\_BITS SMS](#): 3  
*Clock/trigger/slave mode selection.*
    - [\\_BITS res](#): 1  
*Reserved.*
    - [\\_BITS TS](#): 3  
*Trigger selection.*
    - [\\_BITS MSM](#): 1  
*Master/slave mode.*
- [} SMCR](#)
- Slave mode control register (TIM1\_SMCR)*
- struct {
    - [\\_BITS ETF](#): 4  
*External trigger filter.*
    - [\\_BITS ETPS](#): 2  
*External trigger prescaler.*
    - [\\_BITS ECE](#): 1  
*External clock enable.*
    - [\\_BITS ETP](#): 1  
*External trigger polarity.*
- [} ETR](#)

*TIM1 External trigger register (TIM1\_ETR)*

- struct {
    - [\\_BITS UIE](#): 1  
Update interrupt enable.
    - [\\_BITS CC1IE](#): 1  
Capture/compare 1 interrupt enable.
    - [\\_BITS CC2IE](#): 1  
Capture/compare 2 interrupt enable.
    - [\\_BITS CC3IE](#): 1  
Capture/compare 3 interrupt enable.
    - [\\_BITS CC4IE](#): 1  
Capture/compare 4 interrupt enable.
    - [\\_BITS COMIE](#): 1  
Commutation interrupt enable.
    - [\\_BITS TIE](#): 1  
Trigger interrupt enable.
    - [\\_BITS BIE](#): 1  
Break interrupt enable.
- } IER

*TIM1 Interrupt enable register (TIM1\_IER)*

- struct {
    - [\\_BITS UIF](#): 1  
Update interrupt flag.
    - [\\_BITS CC1IF](#): 1  
Capture/compare 1 interrupt flag.
    - [\\_BITS CC2IF](#): 1  
Capture/compare 2 interrupt flag.
    - [\\_BITS CC3IF](#): 1  
Capture/compare 3 interrupt flag.
    - [\\_BITS CC4IF](#): 1  
Capture/compare 4 interrupt flag.
    - [\\_BITS COMIF](#): 1  
Commutation interrupt flag.
    - [\\_BITS TIF](#): 1  
Trigger interrupt flag.
    - [\\_BITS BIF](#): 1  
Break interrupt flag.
- } SR1

*TIM1 Status register 1 (TIM1\_SR1)*

- struct {
    - [\\_BITS res](#): 1  
Reserved, must be kept cleared.
    - [\\_BITS CC1OF](#): 1  
Capture/compare 1 overcapture flag.
    - [\\_BITS CC2OF](#): 1  
Capture/compare 2 overcapture flag.
    - [\\_BITS CC3OF](#): 1  
Capture/compare 3 overcapture flag.
    - [\\_BITS CC4OF](#): 1  
Capture/compare 4 overcapture flag.
    - [\\_BITS res2](#): 3  
Reserved, must be kept cleared.
- } SR2

*TIM1 Status register 2 (TIM1\_SR2)*

- struct {
  - [\\_BITS UG](#): 1  
*Update generation.*
  - [\\_BITS CC1G](#): 1  
*Capture/compare 1 generation.*
  - [\\_BITS CC2G](#): 1  
*Capture/compare 2 generation.*
  - [\\_BITS CC3G](#): 1  
*Capture/compare 3 generation.*
  - [\\_BITS CC4G](#): 1  
*Capture/compare 4 generation.*
  - [\\_BITS COMG](#): 1  
*Capture/compare control update generation.*
  - [\\_BITS TG](#): 1  
*Trigger generation.*
  - [\\_BITS BG](#): 1  
*Break generation.*
- } [EGR](#)

*TIM1 Event generation register (TIM1\_EGR)*

- union {
  - struct {
    - [\\_BITS CC1S](#): 2  
*Compare 1 selection.*
    - [\\_BITS OC1FE](#): 1  
*Output compare 1 fast enable.*
    - [\\_BITS OC1PE](#): 1  
*Output compare 1 preload enable.*
    - [\\_BITS OC1M](#): 3  
*Output compare 1 mode.*
    - [\\_BITS OC1CE](#): 1  
*Output compare 1 clear enable.*
  - } [OUT](#)  
*bitwise access to register (output mode)*
  - struct {
    - [\\_BITS CC1S](#): 2  
*Capture 1 selection.*
    - [\\_BITS IC1PSC](#): 2  
*Input capture 1 prescaler.*
    - [\\_BITS IC1F](#): 4  
*Input capture 1 filter.*
  - } [IN](#)  
*bitwise access to register (input mode)*
- } [CCMR1](#)

*TIM1 Capture/compare mode register 1 (TIM1\_CCMR1)*

- union {
  - struct {
    - [\\_BITS CC2S](#): 2  
*Capture/compare 2 selection.*
    - [\\_BITS OC2FE](#): 1  
*Output compare 2 fast enable.*
    - [\\_BITS OC2PE](#): 1  
*Output compare 2 preload enable.*
    - [\\_BITS OC2M](#): 3  
*Output compare 2 mode.*
    - [\\_BITS OC2CE](#): 1



```

    Output compare 2 clear enable.
} OUT
    bitwise access to register (output mode)
struct {
    _BITS CC2S: 2
        Capture/compare 2 selection.
    _BITS IC2PSC: 2
        Input capture 2 prescaler.
    _BITS IC2F: 4
        Input capture 2 filter.
} IN
    bitwise access to register (input mode)
} CCMR2

```

TIM1 Capture/compare mode register 2 (TIM1\_CCMR2)

```

• union {
    struct {
        _BITS CC3S: 2
            Capture/compare 3 selection.
        _BITS OC3FE: 1
            Output compare 3 fast enable.
        _BITS OC3PE: 1
            Output compare 3 preload enable.
        _BITS OC3M: 3
            Output compare 3 mode.
        _BITS OC3CE: 1
            Output compare 3 clear enable.
    } OUT
        bitwise access to register (output mode)
    struct {
        _BITS CC3S: 2
            Capture/compare 3 selection.
        _BITS IC3PSC: 2
            Input capture 3 prescaler.
        _BITS IC3F: 4
            Input capture 3 filter.
    } IN
        bitwise access to register (input mode)
    } CCMR3

```

TIM1 Capture/compare mode register 3 (TIM1\_CCMR3)

```

• union {
    struct {
        _BITS CC4S: 2
            Capture/compare 4 selection.
        _BITS OC4FE: 1
            Output compare 4 fast enable.
        _BITS OC4PE: 1
            Output compare 4 preload enable.
        _BITS OC4M: 3
            Output compare 4 mode.
        _BITS OC4CE: 1
            Output compare 4 clear enable.
    } OUT
        bitwise access to register (output mode)
    struct {
        _BITS CC4S: 2
            Capture/compare 4 selection.
    } IN
        bitwise access to register (input mode)
    } CCMR4

```

```

    _BITS IC4PSC: 2
        Input capture 4 prescaler.
    _BITS IC4F: 4
        Input capture 4 filter.
} IN
    bitwise access to register (input mode)
} CCMR4

```

*TIM1 Capture/compare mode register 4 (TIM1\_CCMR4)*

```

• struct {
    _BITS CC1E: 1
        Capture/compare 1 output enable.
    _BITS CC1P: 1
        Capture/compare 1 output polarity.
    _BITS CC1NE: 1
        Capture/compare 1 complementary output enable.
    _BITS CC1NP: 1
        Capture/compare 1 complementary output polarity.
    _BITS CC2E: 1
        Capture/compare 2 output enable.
    _BITS CC2P: 1
        Capture/compare 2 output polarity.
    _BITS CC2NE: 1
        Capture/compare 2 complementary output enable.
    _BITS CC2NP: 1
        Capture/compare 2 complementary output polarity.
} CCER1

```

*TIM1 Capture/compare enable register 1 (TIM1\_CCER1)*

```

• struct {
    _BITS CC3E: 1
        Capture/compare 3 output enable.
    _BITS CC3P: 1
        Capture/compare 3 output polarity.
    _BITS CC3NE: 1
        Capture/compare 3 complementary output enable.
    _BITS CC3NP: 1
        Capture/compare 3 complementary output polarity.
    _BITS CC4E: 1
        Capture/compare 4 output enable.
    _BITS CC4P: 1
        Capture/compare 4 output polarity.
    _BITS res: 2
        Reserved.
} CCER2

```

*TIM1 Capture/compare enable register 2 (TIM1\_CCER2)*

```

• struct {
    _BITS CNT: 8
        16-bit counter [15:8]
} CNTRH

```

*TIM1 16-bit counter high byte (TIM1\_CNTRH)*

```

• struct {
    _BITS CNT: 8
        16-bit counter [7:0]
} CNTRL

```

*TIM1 16-bit counter low byte (TIM1\_CNTRL)*

- struct {  
     \_BITS PSC: 8  
     16-bit prescaler [15:8]  
  } PSCRH

*TIM1 16-bit prescaler high byte (TIM1\_PSCRH)*

- struct {  
     \_BITS PSC: 8  
     16-bit prescaler [7:0]  
  } PSCRL

*TIM1 16-bit prescaler low byte (TIM1\_PSCRL)*

- struct {  
     \_BITS ARR: 8  
     16-bit auto-reload value [15:8]  
  } ARRH

*TIM1 16-bit auto-reload value high byte (TIM1\_ARRH)*

- struct {  
     \_BITS ARR: 8  
     16-bit auto-reload value [7:0]  
  } ARRL

*TIM1 16-bit auto-reload value low byte (TIM1\_ARRL)*

- struct {  
     \_BITS REP: 8  
     Repetition counter value.  
  } RCR

*TIM1 Repetition counter (TIM1\_RCR)*

- struct {  
     \_BITS CCR1: 8  
     16-bit capture/compare value 1 [15:8]  
  } CCR1H

*TIM1 16-bit capture/compare value 1 high byte (TIM1\_CCR1H)*

- struct {  
     \_BITS CCR1: 8  
     16-bit capture/compare value 1 [7:0]  
  } CCR1L

*TIM1 16-bit capture/compare value 1 low byte (TIM1\_CCR1L)*

- struct {  
     \_BITS CCR2: 8  
     16-bit capture/compare value 2 [15:8]  
  } CCR2H

*TIM1 16-bit capture/compare value 2 high byte (TIM1\_CCR2H)*

- struct {  
     \_BITS CCR2: 8  
     16-bit capture/compare value 2 [7:0]  
  } CCR2L

*TIM1 16-bit capture/compare value 2 low byte (TIM1\_CCR2L)*

- struct {
  - [\\_BITS CCR3](#): 8  
16-bit capture/compare value 3 [15:8]

} [CCR3H](#)

*TIM1 16-bit capture/compare value 3 high byte (TIM1\_CCR3H)*

- struct {
  - [\\_BITS CCR3](#): 8  
16-bit capture/compare value 3 [7:0]

} [CCR3L](#)

*TIM1 16-bit capture/compare value 3 low byte (TIM1\_CCR3L)*

- struct {
  - [\\_BITS CCR4](#): 8  
16-bit capture/compare value 4 [15:8]

} [CCR4H](#)

*TIM1 16-bit capture/compare value 4 high byte (TIM1\_CCR4H)*

- struct {
  - [\\_BITS CCR4](#): 8  
16-bit capture/compare value 4 [7:0]

} [CCR4L](#)

*TIM1 16-bit capture/compare value 4 low byte (TIM1\_CCR4L)*

- struct {
  - [\\_BITS LOCK](#): 2  
Lock configuration.
  - [\\_BITS OSSI](#): 1  
Off state selection for idle mode.
  - [\\_BITS OSSR](#): 1  
Off state selection for Run mode.
  - [\\_BITS BKE](#): 1  
Break enable.
  - [\\_BITS BKP](#): 1  
Break polarity.
  - [\\_BITS AOE](#): 1  
Automatic output enable.
  - [\\_BITS MOE](#): 1  
Main output enable.

} [BKR](#)

*TIM1 Break register (TIM1\_BKR)*

- struct {
  - [\\_BITS DTG](#): 8  
Deadtime generator set-up.

} [DTR](#)

*TIM1 Dead-time register (TIM1\_DTR)*

- struct {
  - [\\_BITS OIS1](#): 1  
Output idle state 1 (OC1 output)
  - [\\_BITS OIS1N](#): 1  
Output idle state 1 (OC1N output)
  - [\\_BITS OIS2](#): 1  
Output idle state 2 (OC2 output)
  - [\\_BITS OIS2N](#): 1

```

    Output idle state 2 (OC2N output)
    _BITS OIS3: 1
    Output idle state 3 (OC3 output)
    _BITS OIS3N: 1
    Output idle state 3 (OC3N output)
    _BITS OIS4: 1
    Output idle state 4 (OC4 output)
    _BITS res: 1
    Reserved, forced by hardware to 0.
} OISR

TIM1 Output idle state register (TIM1_OISR)

```

### 6.16.1 Detailed Description

struct for controlling 16-Bit Timer 1 (TIM1)

Definition at line 2474 of file STM8AF\_STM8S.h.

### 6.16.2 Field Documentation

#### 6.16.2.1 AOE

```
_BITS AOE
```

Automatic output enable.

Definition at line 2777 of file STM8AF\_STM8S.h.

#### 6.16.2.2 ARPE

```
_BITS ARPE
```

Auto-reload preload enable.

Definition at line 2484 of file STM8AF\_STM8S.h.

#### 6.16.2.3 ARR

```
_BITS ARR
```

16-bit auto-reload value [15:8]

16-bit auto-reload value [7:0]

Definition at line 2706 of file STM8AF\_STM8S.h.

#### 6.16.2.4 **ARRH**

```
struct { ... } ARRH
```

TIM1 16-bit auto-reload value high byte (TIM1\_ARRH)

#### 6.16.2.5 **ARRL**

```
struct { ... } ARRL
```

TIM1 16-bit auto-reload value low byte (TIM1\_ARRL)

#### 6.16.2.6 **BG**

`_BITS` BG

Break generation.

Definition at line 2563 of file STM8AF\_STM8S.h.

#### 6.16.2.7 **BIE**

`_BITS` BIE

Break interrupt enable.

Definition at line 2526 of file STM8AF\_STM8S.h.

#### 6.16.2.8 **BIF**

`_BITS` BIF

Break interrupt flag.

Definition at line 2539 of file STM8AF\_STM8S.h.

#### 6.16.2.9 BKE

`_BITS` BKE

Break enable.

Definition at line 2775 of file STM8AF\_STM8S.h.

#### 6.16.2.10 BKP

`_BITS` BKP

Break polarity.

Definition at line 2776 of file STM8AF\_STM8S.h.

#### 6.16.2.11 BKR

```
struct { ... } BKR
```

TIM1 Break register (TIM1\_BKR)

#### 6.16.2.12 CC1E

`_BITS` CC1E

Capture/compare 1 output enable.

Definition at line 2657 of file STM8AF\_STM8S.h.

#### 6.16.2.13 CC1G

`_BITS` CC1G

Capture/compare 1 generation.

Definition at line 2557 of file STM8AF\_STM8S.h.

**6.16.2.14 CC1IE**

`_BITS CC1IE`

Capture/compare 1 interrupt enable.

Definition at line 2520 of file STM8AF\_STM8S.h.

**6.16.2.15 CC1IF**

`_BITS CC1IF`

Capture/compare 1 interrupt flag.

Definition at line 2533 of file STM8AF\_STM8S.h.

**6.16.2.16 CC1NE**

`_BITS CC1NE`

Capture/compare 1 complementary output enable.

Definition at line 2659 of file STM8AF\_STM8S.h.

**6.16.2.17 CC1NP**

`_BITS CC1NP`

Capture/compare 1 complementary output polarity.

Definition at line 2660 of file STM8AF\_STM8S.h.

**6.16.2.18 CC1OF**

`_BITS CC1OF`

Capture/compare 1 overcapture flag.

Definition at line 2546 of file STM8AF\_STM8S.h.



#### 6.16.2.19 CC1P

`_BITS` CC1P

Capture/compare 1 output polarity.

Definition at line 2658 of file STM8AF\_STM8S.h.

#### 6.16.2.20 CC1S

`_BITS` CC1S

Compare 1 selection.

Capture 1 selection.

Definition at line 2572 of file STM8AF\_STM8S.h.

#### 6.16.2.21 CC2E

`_BITS` CC2E

Capture/compare 2 output enable.

Definition at line 2661 of file STM8AF\_STM8S.h.

#### 6.16.2.22 CC2G

`_BITS` CC2G

Capture/compare 2 generation.

Definition at line 2558 of file STM8AF\_STM8S.h.

#### 6.16.2.23 CC2IE

`_BITS` CC2IE

Capture/compare 2 interrupt enable.

Definition at line 2521 of file STM8AF\_STM8S.h.

**6.16.2.24 CC2IF**

`_BITS CC2IF`

Capture/compare 2 interrupt flag.

Definition at line 2534 of file STM8AF\_STM8S.h.

**6.16.2.25 CC2NE**

`_BITS CC2NE`

Capture/compare 2 complementary output enable.

Definition at line 2663 of file STM8AF\_STM8S.h.

**6.16.2.26 CC2NP**

`_BITS CC2NP`

Capture/compare 2 complementary output polarity.

Definition at line 2664 of file STM8AF\_STM8S.h.

**6.16.2.27 CC2OF**

`_BITS CC2OF`

Capture/compare 2 overcapture flag.

Definition at line 2547 of file STM8AF\_STM8S.h.

**6.16.2.28 CC2P**

`_BITS CC2P`

Capture/compare 2 output polarity.

Definition at line 2662 of file STM8AF\_STM8S.h.

#### 6.16.2.29 CC2S

`_BITS` CC2S

Capture/compare 2 selection.

Definition at line 2594 of file STM8AF\_STM8S.h.

#### 6.16.2.30 CC3E

`_BITS` CC3E

Capture/compare 3 output enable.

Definition at line 2670 of file STM8AF\_STM8S.h.

#### 6.16.2.31 CC3G

`_BITS` CC3G

Capture/compare 3 generation.

Definition at line 2559 of file STM8AF\_STM8S.h.

#### 6.16.2.32 CC3IE

`_BITS` CC3IE

Capture/compare 3 interrupt enable.

Definition at line 2522 of file STM8AF\_STM8S.h.

#### 6.16.2.33 CC3IF

`_BITS` CC3IF

Capture/compare 3 interrupt flag.

Definition at line 2535 of file STM8AF\_STM8S.h.

**6.16.2.34 CC3NE**

`_BITS CC3NE`

Capture/compare 3 complementary output enable.

Definition at line 2672 of file STM8AF\_STM8S.h.

**6.16.2.35 CC3NP**

`_BITS CC3NP`

Capture/compare 3 complementary output polarity.

Definition at line 2673 of file STM8AF\_STM8S.h.

**6.16.2.36 CC3OF**

`_BITS CC3OF`

Capture/compare 3 overcapture flag.

Definition at line 2548 of file STM8AF\_STM8S.h.

**6.16.2.37 CC3P**

`_BITS CC3P`

Capture/compare 3 output polarity.

Definition at line 2671 of file STM8AF\_STM8S.h.

**6.16.2.38 CC3S**

`_BITS CC3S`

Capture/compare 3 selection.

Definition at line 2616 of file STM8AF\_STM8S.h.

#### 6.16.2.39 CC4E

`_BITS` CC4E

Capture/compare 4 output enable.

Definition at line 2674 of file STM8AF\_STM8S.h.

#### 6.16.2.40 CC4G

`_BITS` CC4G

Capture/compare 4 generation.

Definition at line 2560 of file STM8AF\_STM8S.h.

#### 6.16.2.41 CC4IE

`_BITS` CC4IE

Capture/compare 4 interrupt enable.

Definition at line 2523 of file STM8AF\_STM8S.h.

#### 6.16.2.42 CC4IF

`_BITS` CC4IF

Capture/compare 4 interrupt flag.

Definition at line 2536 of file STM8AF\_STM8S.h.

#### 6.16.2.43 CC4OF

`_BITS` CC4OF

Capture/compare 4 overcapture flag.

Definition at line 2549 of file STM8AF\_STM8S.h.

**6.16.2.44 CC4P**

`_BITS` CC4P

Capture/compare 4 output polarity.

Definition at line 2675 of file STM8AF\_STM8S.h.

**6.16.2.45 CC4S**

`_BITS` CC4S

Capture/compare 4 selection.

Definition at line 2638 of file STM8AF\_STM8S.h.

**6.16.2.46 CCER1**

```
struct { ... } CCER1
```

TIM1 Capture/compare enable register 1 (TIM1\_CCER1)

**6.16.2.47 CCER2**

```
struct { ... } CCER2
```

TIM1 Capture/compare enable register 2 (TIM1\_CCER2)

**6.16.2.48 CCMR1**

```
union { ... } CCMR1
```

TIM1 Capture/compare mode register 1 (TIM1\_CCMR1)

**6.16.2.49 CCMR2**

```
union { ... } CCMR2
```

TIM1 Capture/compare mode register 2 (TIM1\_CCMR2)

#### 6.16.2.50 CCMR3

```
union { ... } CCMR3
```

TIM1 Capture/compare mode register 3 (TIM1\_CCMR3)

#### 6.16.2.51 CCMR4

```
union { ... } CCMR4
```

TIM1 Capture/compare mode register 4 (TIM1\_CCMR4)

#### 6.16.2.52 CCPC

```
_BITS CCPC
```

Capture/compare preloaded control.

Definition at line 2490 of file STM8AF\_STM8S.h.

#### 6.16.2.53 CCR1

```
_BITS CCR1
```

16-bit capture/compare value 1 [15:8]

16-bit capture/compare value 1 [7:0]

Definition at line 2724 of file STM8AF\_STM8S.h.

#### 6.16.2.54 CCR1H

```
struct { ... } CCR1H
```

TIM1 16-bit capture/compare value 1 high byte (TIM1\_CCR1H)

#### 6.16.2.55 CCR1L

```
struct { ... } CCR1L
```

TIM1 16-bit capture/compare value 1 low byte (TIM1\_CCR1L)

#### 6.16.2.56 CCR2

[\\_BITS](#) CCR2

16-bit capture/compare value 2 [15:8]

16-bit capture/compare value 2 [7:0]

Definition at line 2736 of file STM8AF\_STM8S.h.

#### 6.16.2.57 CCR2H

```
struct { ... } CCR2H
```

TIM1 16-bit capture/compare value 2 high byte (TIM1\_CCR2H)

#### 6.16.2.58 CCR2L

```
struct { ... } CCR2L
```

TIM1 16-bit capture/compare value 2 low byte (TIM1\_CCR2L)

#### 6.16.2.59 CCR3

[\\_BITS](#) CCR3

16-bit capture/compare value 3 [15:8]

16-bit capture/compare value 3 [7:0]

Definition at line 2748 of file STM8AF\_STM8S.h.



#### 6.16.2.60 CCR3H

```
struct { ... } CCR3H
```

TIM1 16-bit capture/compare value 3 high byte (TIM1\_CCR3H)

#### 6.16.2.61 CCR3L

```
struct { ... } CCR3L
```

TIM1 16-bit capture/compare value 3 low byte (TIM1\_CCR3L)

#### 6.16.2.62 CCR4

```
_BITS CCR4
```

16-bit capture/compare value 4 [15:8]

16-bit capture/compare value 4 [7:0]

Definition at line 2760 of file STM8AF\_STM8S.h.

#### 6.16.2.63 CCR4H

```
struct { ... } CCR4H
```

TIM1 16-bit capture/compare value 4 high byte (TIM1\_CCR4H)

#### 6.16.2.64 CCR4L

```
struct { ... } CCR4L
```

TIM1 16-bit capture/compare value 4 low byte (TIM1\_CCR4L)

#### 6.16.2.65 CEN

```
_BITS CEN
```

Counter enable.

Definition at line 2478 of file STM8AF\_STM8S.h.

**6.16.2.66 CMS**

`_BITS` CMS

Center-aligned mode selection.

Definition at line 2483 of file STM8AF\_STM8S.h.

**6.16.2.67 CNT**

`_BITS` CNT

16-bit counter [15:8]

16-bit counter [7:0]

Definition at line 2682 of file STM8AF\_STM8S.h.

**6.16.2.68 CNTRH**

```
struct { ... } CNTRH
```

TIM1 16-bit counter high byte (TIM1\_CNTRH)

**6.16.2.69 CNTRL**

```
struct { ... } CNTRL
```

TIM1 16-bit counter low byte (TIM1\_CNTRL)

**6.16.2.70 COMG**

`_BITS` COMG

Capture/compare control update generation.

Definition at line 2561 of file STM8AF\_STM8S.h.

#### 6.16.2.71 COMIE

`_BITS` COMIE

Commutation interrupt enable.

Definition at line 2524 of file STM8AF\_STM8S.h.

#### 6.16.2.72 COMIF

`_BITS` COMIF

Commutation interrupt flag.

Definition at line 2537 of file STM8AF\_STM8S.h.

#### 6.16.2.73 COMS

`_BITS` COMS

Capture/compare control update selection.

Definition at line 2492 of file STM8AF\_STM8S.h.

#### 6.16.2.74 CR1

```
struct { ... } CR1
```

TIM1 Control register 1 (TIM1\_CR1)

#### 6.16.2.75 CR2

```
struct { ... } CR2
```

TIM1 Control register 2 (TIM1\_CR2)

**6.16.2.76 DIR**

`_BITS` DIR

Direction.

Definition at line 2482 of file STM8AF\_STM8S.h.

**6.16.2.77 DTG**

`_BITS` DTG

Deadtime generator set-up.

Definition at line 2784 of file STM8AF\_STM8S.h.

**6.16.2.78 DTR**

```
struct { ... } DTR
```

TIM1 Dead-time register (TIM1\_DTR)

**6.16.2.79 ECE**

`_BITS` ECE

External clock enable.

Definition at line 2512 of file STM8AF\_STM8S.h.

**6.16.2.80 EGR**

```
struct { ... } EGR
```

TIM1 Event generation register (TIM1\_EGR)

#### 6.16.2.81 ETF

`_BITS ETF`

External trigger filter.

Definition at line 2510 of file STM8AF\_STM8S.h.

#### 6.16.2.82 ETP

`_BITS ETP`

External trigger polarity.

Definition at line 2513 of file STM8AF\_STM8S.h.

#### 6.16.2.83 ETPS

`_BITS ETPS`

External trigger prescaler.

Definition at line 2511 of file STM8AF\_STM8S.h.

#### 6.16.2.84 ETR

```
struct { ... } ETR
```

TIM1 External trigger register (TIM1\_ETR)

#### 6.16.2.85 IC1F

`_BITS IC1F`

Input capture 1 filter.

Definition at line 2583 of file STM8AF\_STM8S.h.

**6.16.2.86 IC1PSC**

`_BITS IC1PSC`

Input capture 1 prescaler.

Definition at line 2582 of file STM8AF\_STM8S.h.

**6.16.2.87 IC2F**

`_BITS IC2F`

Input capture 2 filter.

Definition at line 2605 of file STM8AF\_STM8S.h.

**6.16.2.88 IC2PSC**

`_BITS IC2PSC`

Input capture 2 prescaler.

Definition at line 2604 of file STM8AF\_STM8S.h.

**6.16.2.89 IC3F**

`_BITS IC3F`

Input capture 3 filter.

Definition at line 2627 of file STM8AF\_STM8S.h.

**6.16.2.90 IC3PSC**

`_BITS IC3PSC`

Input capture 3 prescaler.

Definition at line 2626 of file STM8AF\_STM8S.h.

#### 6.16.2.91 IC4F

`_BITS` IC4F

Input capture 4 filter.

Definition at line 2649 of file STM8AF\_STM8S.h.

#### 6.16.2.92 IC4PSC

`_BITS` IC4PSC

Input capture 4 prescaler.

Definition at line 2648 of file STM8AF\_STM8S.h.

#### 6.16.2.93 IER

```
struct { ... } IER
```

TIM1 Interrupt enable register (TIM1\_IER)

#### 6.16.2.94 IN [1/4]

```
struct { ... } IN
```

bitwise access to register (input mode)

#### 6.16.2.95 IN [2/4]

```
struct { ... } IN
```

bitwise access to register (input mode)

#### 6.16.2.96 IN [3/4]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.16.2.97 IN** [4/4]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.16.2.98 LOCK**

```
_BITS LOCK
```

Lock configuration.

Definition at line 2772 of file STM8AF\_STM8S.h.

**6.16.2.99 MMS**

```
_BITS MMS
```

Master mode selection.

Definition at line 2494 of file STM8AF\_STM8S.h.

**6.16.2.100 MOE**

```
_BITS MOE
```

Main output enable.

Definition at line 2778 of file STM8AF\_STM8S.h.

**6.16.2.101 MSM**

```
_BITS MSM
```

Master/slave mode.

Definition at line 2504 of file STM8AF\_STM8S.h.



**6.16.2.102 OC1CE**

`_BITS OC1CE`

Output compare 1 clear enable.

Definition at line 2576 of file STM8AF\_STM8S.h.

**6.16.2.103 OC1FE**

`_BITS OC1FE`

Output compare 1 fast enable.

Definition at line 2573 of file STM8AF\_STM8S.h.

**6.16.2.104 OC1M**

`_BITS OC1M`

Output compare 1 mode.

Definition at line 2575 of file STM8AF\_STM8S.h.

**6.16.2.105 OC1PE**

`_BITS OC1PE`

Output compare 1 preload enable.

Definition at line 2574 of file STM8AF\_STM8S.h.

**6.16.2.106 OC2CE**

`_BITS OC2CE`

Output compare 2 clear enable.

Definition at line 2598 of file STM8AF\_STM8S.h.

**6.16.2.107 OC2FE**

`_BITS` OC2FE

Output compare 2 fast enable.

Definition at line 2595 of file STM8AF\_STM8S.h.

**6.16.2.108 OC2M**

`_BITS` OC2M

Output compare 2 mode.

Definition at line 2597 of file STM8AF\_STM8S.h.

**6.16.2.109 OC2PE**

`_BITS` OC2PE

Output compare 2 preload enable.

Definition at line 2596 of file STM8AF\_STM8S.h.

**6.16.2.110 OC3CE**

`_BITS` OC3CE

Output compare 3 clear enable.

Definition at line 2620 of file STM8AF\_STM8S.h.

**6.16.2.111 OC3FE**

`_BITS` OC3FE

Output compare 3 fast enable.

Definition at line 2617 of file STM8AF\_STM8S.h.

**6.16.2.112 OC3M**

`_BITS OC3M`

Output compare 3 mode.

Definition at line 2619 of file STM8AF\_STM8S.h.

**6.16.2.113 OC3PE**

`_BITS OC3PE`

Output compare 3 preload enable.

Definition at line 2618 of file STM8AF\_STM8S.h.

**6.16.2.114 OC4CE**

`_BITS OC4CE`

Output compare 4 clear enable.

Definition at line 2642 of file STM8AF\_STM8S.h.

**6.16.2.115 OC4FE**

`_BITS OC4FE`

Output compare 4 fast enable.

Definition at line 2639 of file STM8AF\_STM8S.h.

**6.16.2.116 OC4M**

`_BITS OC4M`

Output compare 4 mode.

Definition at line 2641 of file STM8AF\_STM8S.h.

**6.16.2.117 OC4PE**

`_BITS OC4PE`

Output compare 4 preload enable.

Definition at line 2640 of file STM8AF\_STM8S.h.

**6.16.2.118 OIS1**

`_BITS OIS1`

Output idle state 1 (OC1 output)

Definition at line 2790 of file STM8AF\_STM8S.h.

**6.16.2.119 OIS1N**

`_BITS OIS1N`

Output idle state 1 (OC1N output)

Definition at line 2791 of file STM8AF\_STM8S.h.

**6.16.2.120 OIS2**

`_BITS OIS2`

Output idle state 2 (OC2 output)

Definition at line 2792 of file STM8AF\_STM8S.h.

**6.16.2.121 OIS2N**

`_BITS OIS2N`

Output idle state 2 (OC2N output)

Definition at line 2793 of file STM8AF\_STM8S.h.

**6.16.2.122 OIS3**

`_BITS OIS3`

Output idle state 3 (OC3 output)

Definition at line 2794 of file STM8AF\_STM8S.h.

**6.16.2.123 OIS3N**

`_BITS OIS3N`

Output idle state 3 (OC3N output)

Definition at line 2795 of file STM8AF\_STM8S.h.

**6.16.2.124 OIS4**

`_BITS OIS4`

Output idle state 4 (OC4 output)

Definition at line 2796 of file STM8AF\_STM8S.h.

**6.16.2.125 OISR**

```
struct { ... } OISR
```

TIM1 Output idle state register (TIM1\_OISR)

**6.16.2.126 OPM**

`_BITS OPM`

One-pulse mode.

Definition at line 2481 of file STM8AF\_STM8S.h.

**6.16.2.127 OSSI**

`_BITS` OSSI

Off state selection for idle mode.

Definition at line 2773 of file STM8AF\_STM8S.h.

**6.16.2.128 OSSR**

`_BITS` OSSR

Off state selection for Run mode.

Definition at line 2774 of file STM8AF\_STM8S.h.

**6.16.2.129 OUT [1/4]**

```
struct { ... } OUT
```

bitwise access to register (output mode)

**6.16.2.130 OUT [2/4]**

```
struct { ... } OUT
```

bitwise access to register (output mode)

**6.16.2.131 OUT [3/4]**

```
struct { ... } OUT
```

bitwise access to register (output mode)

**6.16.2.132 OUT [4/4]**

```
struct { ... } OUT
```

bitwise access to register (output mode)

#### 6.16.2.133 PSC

`_BITS` PSC

16-bit prescaler [15:8]

16-bit prescaler [7:0]

Definition at line 2694 of file STM8AF\_STM8S.h.

#### 6.16.2.134 PSCRH

```
struct { ... } PSCRH
```

TIM1 16-bit prescaler high byte (TIM1\_PSCRH)

#### 6.16.2.135 PSCRL

```
struct { ... } PSCRL
```

TIM1 16-bit prescaler low byte (TIM1\_PSCRL)

#### 6.16.2.136 RCR

```
struct { ... } RCR
```

TIM1 Repetition counter (TIM1\_RCR)

#### 6.16.2.137 REP

`_BITS` REP

Repetition counter value.

Definition at line 2718 of file STM8AF\_STM8S.h.

**6.16.2.138 res**

`_BITS res`

Reserved, forced by hardware to 0.

Reserved, must be kept cleared.

Reserved.

Definition at line 2491 of file STM8AF\_STM8S.h.

**6.16.2.139 res2**

`_BITS res2`

Reserved, must be kept cleared.

Definition at line 2493 of file STM8AF\_STM8S.h.

**6.16.2.140 res3**

`_BITS res3`

Reserved.

Definition at line 2495 of file STM8AF\_STM8S.h.

**6.16.2.141 SMCR**

```
struct { ... } SMCR
```

Slave mode control register (TIM1\_SMCR)

**6.16.2.142 SMS**

`_BITS SMS`

Clock/trigger/slave mode selection.

Definition at line 2501 of file STM8AF\_STM8S.h.



**6.16.2.143 SR1**

```
struct { ... } SR1
```

TIM1 Status register 1 (TIM1\_SR1)

**6.16.2.144 SR2**

```
struct { ... } SR2
```

TIM1 Status register 2 (TIM1\_SR2)

**6.16.2.145 TG**

```
_BITS TG
```

Trigger generation.

Definition at line 2562 of file STM8AF\_STM8S.h.

**6.16.2.146 TIE**

```
_BITS TIE
```

Trigger interrupt enable.

Definition at line 2525 of file STM8AF\_STM8S.h.

**6.16.2.147 TIF**

```
_BITS TIF
```

Trigger interrupt flag.

Definition at line 2538 of file STM8AF\_STM8S.h.

**6.16.2.148 TS**

`_BITS TS`

Trigger selection.

Definition at line 2503 of file STM8AF\_STM8S.h.

**6.16.2.149 UDIS**

`_BITS UDIS`

Update disable.

Definition at line 2479 of file STM8AF\_STM8S.h.

**6.16.2.150 UG**

`_BITS UG`

Update generation.

Definition at line 2556 of file STM8AF\_STM8S.h.

**6.16.2.151 UIE**

`_BITS UIE`

Update interrupt enable.

Definition at line 2519 of file STM8AF\_STM8S.h.

**6.16.2.152 UIF**

`_BITS UIF`

Update interrupt flag.

Definition at line 2532 of file STM8AF\_STM8S.h.

## 6.16.2.153 URS

[\\_BITS](#) URS

Update request source.

Definition at line 2480 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.17 TIM2\_t Struct Reference

struct for controlling 16-Bit Timer 2 (TIM2)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {
    - [\\_BITS CEN](#): 1  
*Counter enable.*
    - [\\_BITS UDIS](#): 1  
*Update disable.*
    - [\\_BITS URS](#): 1  
*Update request source.*
    - [\\_BITS OPM](#): 1  
*One-pulse mode.*
    - [\\_BITS res](#): 3  
*Reserved.*
    - [\\_BITS ARPE](#): 1  
*Auto-reload preload enable.*
  - } [CR1](#)
- TIM2 Control register 1 (TIM2\_CR1)*
- struct {
    - [\\_BITS UIE](#): 1  
*Update interrupt enable.*
    - [\\_BITS CC1IE](#): 1  
*Capture/compare 1 interrupt enable.*
    - [\\_BITS CC2IE](#): 1  
*Capture/compare 2 interrupt enable.*
    - [\\_BITS CC3IE](#): 1  
*Capture/compare 3 interrupt enable.*
    - [\\_BITS res](#): 4  
*Reserved.*
  - } [IER](#)
- Reserved registers on selected devices (2B)*

- struct {
  - [\\_BITS UIF](#): 1  
*Update interrupt flag.*
  - [\\_BITS CC1IF](#): 1  
*Capture/compare 1 interrupt flag.*
  - [\\_BITS CC2IF](#): 1  
*Capture/compare 2 interrupt flag.*
  - [\\_BITS CC3IF](#): 1  
*Capture/compare 3 interrupt flag.*
  - [\\_BITS res](#): 4  
*Reserved.*

} [SR1](#)

*TIM2 Status register 1 (TIM2\_SR1)*

- struct {
  - [\\_BITS res](#): 1  
*Reserved, must be kept cleared.*
  - [\\_BITS CC1OF](#): 1  
*Capture/compare 1 overcapture flag.*
  - [\\_BITS CC2OF](#): 1  
*Capture/compare 2 overcapture flag.*
  - [\\_BITS CC3OF](#): 1  
*Capture/compare 3 overcapture flag.*
  - [\\_BITS res2](#): 4  
*Reserved, must be kept cleared.*

} [SR2](#)

*TIM2 Status register 2 (TIM2\_SR2)*

- struct {
  - [\\_BITS UG](#): 1  
*Update generation.*
  - [\\_BITS CC1G](#): 1  
*Capture/compare 1 generation.*
  - [\\_BITS CC2G](#): 1  
*Capture/compare 2 generation.*
  - [\\_BITS CC3G](#): 1  
*Capture/compare 3 generation.*
  - [\\_BITS res](#): 4  
*Reserved.*

} [EGR](#)

*TIM2 Event generation register (TIM2\_EGR)*

- union {
  - struct {
    - [\\_BITS CC1S](#): 2  
*Compare 1 selection.*
    - [\\_BITS res](#): 1  
*Reserved.*
    - [\\_BITS OC1PE](#): 1  
*Output compare 1 preload enable.*
    - [\\_BITS OC1M](#): 3  
*Output compare 1 mode.*
    - [\\_BITS res2](#): 1  
*Reserved.*

} [OUT](#)  
*bitwise access to register (output mode)*

- struct {
  - [\\_BITS CC1S](#): 2

```

    Capture 1 selection.
    _BITS IC1PSC: 2
    Input capture 1 prescaler.
    _BITS IC1F: 4
    Input capture 1 filter.
} IN
    bitwise access to register (input mode)
} CCMR1

```

TIM2 Capture/compare mode register 1 (TIM2\_CCMR1)

```

• union {
    struct {
        _BITS CC2S: 2
        Capture/compare 2 selection.
        _BITS res: 1
        Reserved.
        _BITS OC2PE: 1
        Output compare 2 preload enable.
        _BITS OC2M: 3
        Output compare 2 mode.
        _BITS res2: 1
        Reserved.
    } OUT
    bitwise access to register (output mode)
    struct {
        _BITS CC2S: 2
        Capture/compare 2 selection.
        _BITS IC2PSC: 2
        Input capture 2 prescaler.
        _BITS IC2F: 4
        Input capture 2 filter.
    } IN
    bitwise access to register (input mode)
} CCMR2

```

TIM2 Capture/compare mode register 2 (TIM2\_CCMR2)

```

• union {
    struct {
        _BITS CC3S: 2
        Capture/compare 3 selection.
        _BITS res: 1
        Reserved.
        _BITS OC3PE: 1
        Output compare 3 preload enable.
        _BITS OC3M: 3
        Output compare 3 mode.
        _BITS OC3CE: 1
        Output compare 3 clear enable.
    } OUT
    bitwise access to register (output mode)
    struct {
        _BITS CC3S: 2
        Capture/compare 3 selection.
        _BITS IC3PSC: 2
        Input capture 3 prescaler.
        _BITS IC3F: 4
        Input capture 3 filter.
    } IN

```

*bitwise access to register (input mode)*  
**CCMR3**

*TIM2 Capture/compare mode register 3 (TIM2\_CCMR3)*

- struct {
  - \_BITS CC1E**: 1  
Capture/compare 1 output enable.
  - \_BITS CC1P**: 1  
Capture/compare 1 output polarity.
  - \_BITS res**: 2  
Reserved.
  - \_BITS CC2E**: 1  
Capture/compare 2 output enable.
  - \_BITS CC2P**: 1  
Capture/compare 2 output polarity.
  - \_BITS res2**: 2  
Reserved.

**CCER1**

*TIM2 Capture/compare enable register 1 (TIM2\_CCER1)*

- struct {
  - \_BITS CC3E**: 1  
Capture/compare 3 output enable.
  - \_BITS CC3P**: 1  
Capture/compare 3 output polarity.
  - \_BITS res**: 6  
Reserved.

**CCER2**

*TIM2 Capture/compare enable register 2 (TIM2\_CCER2)*

- struct {
  - \_BITS CNT**: 8  
16-bit counter [15:8]

**CNTRH**

*TIM2 16-bit counter high byte (TIM2\_CNTRH)*

- struct {
  - \_BITS CNT**: 8  
16-bit counter [7:0]

**CNTRL**

*TIM2 16-bit counter low byte (TIM2\_CNTRL)*

- struct {
  - \_BITS PSC**: 4  
prescaler [3:0]
  - \_BITS res**: 4  
Reserved.

**PSCR**

*TIM2 16-bit prescaler high byte (TIM2\_PSCR)*

- struct {
  - \_BITS ARR**: 8  
16-bit auto-reload value [15:8]

**ARRH**

*TIM2 16-bit auto-reload value high byte (TIM2\_ARRH)*

- struct {
  - [\\_BITS ARR](#): 8
  - 16-bit auto-reload value [7:0]
- } [ARRL](#)
- TIM2 16-bit auto-reload value low byte (TIM2\_ARRL)
- struct {
  - [\\_BITS CCR1](#): 8
  - 16-bit capture/compare value 1 [15:8]
- } [CCR1H](#)
- TIM2 16-bit capture/compare value 1 high byte (TIM2\_CCR1H)
- struct {
  - [\\_BITS CCR1](#): 8
  - 16-bit capture/compare value 1 [7:0]
- } [CCR1L](#)
- TIM2 16-bit capture/compare value 1 low byte (TIM2\_CCR1L)
- struct {
  - [\\_BITS CCR2](#): 8
  - 16-bit capture/compare value 2 [15:8]
- } [CCR2H](#)
- TIM2 16-bit capture/compare value 2 high byte (TIM2\_CCR2H)
- struct {
  - [\\_BITS CCR2](#): 8
  - 16-bit capture/compare value 2 [7:0]
- } [CCR2L](#)
- TIM2 16-bit capture/compare value 2 low byte (TIM2\_CCR2L)
- struct {
  - [\\_BITS CCR3](#): 8
  - 16-bit capture/compare value 3 [15:8]
- } [CCR3H](#)
- TIM2 16-bit capture/compare value 3 high byte (TIM2\_CCR3H)
- struct {
  - [\\_BITS CCR3](#): 8
  - 16-bit capture/compare value 3 [7:0]
- } [CCR3L](#)
- TIM2 16-bit capture/compare value 3 low byte (TIM2\_CCR3L)

### 6.17.1 Detailed Description

struct for controlling 16-Bit Timer 2 (TIM2)

Definition at line 3097 of file STM8AF\_STM8S.h.

### 6.17.2 Field Documentation

#### 6.17.2.1 ARPE

`_BITS` ARPE

Auto-reload preload enable.

Definition at line 3106 of file STM8AF\_STM8S.h.

#### 6.17.2.2 ARR

`_BITS` ARR

16-bit auto-reload value [15:8]

16-bit auto-reload value [7:0]

Definition at line 3262 of file STM8AF\_STM8S.h.

#### 6.17.2.3 ARRH

```
struct { ... } ARRH
```

TIM2 16-bit auto-reload value high byte (TIM2\_ARRH)

#### 6.17.2.4 ARRL

```
struct { ... } ARRL
```

TIM2 16-bit auto-reload value low byte (TIM2\_ARRL)

#### 6.17.2.5 CC1E

`_BITS` CC1E

Capture/compare 1 output enable.

Definition at line 3224 of file STM8AF\_STM8S.h.



#### 6.17.2.6 CC1G

`_BITS CC1G`

Capture/compare 1 generation.

Definition at line 3149 of file STM8AF\_STM8S.h.

#### 6.17.2.7 CC1IE

`_BITS CC1IE`

Capture/compare 1 interrupt enable.

Definition at line 3119 of file STM8AF\_STM8S.h.

#### 6.17.2.8 CC1IF

`_BITS CC1IF`

Capture/compare 1 interrupt flag.

Definition at line 3129 of file STM8AF\_STM8S.h.

#### 6.17.2.9 CC1OF

`_BITS CC1OF`

Capture/compare 1 overcapture flag.

Definition at line 3139 of file STM8AF\_STM8S.h.

#### 6.17.2.10 CC1P

`_BITS CC1P`

Capture/compare 1 output polarity.

Definition at line 3225 of file STM8AF\_STM8S.h.

**6.17.2.11 CC1S**

`_BITS` CC1S

Compare 1 selection.

Capture 1 selection.

Definition at line 3161 of file STM8AF\_STM8S.h.

**6.17.2.12 CC2E**

`_BITS` CC2E

Capture/compare 2 output enable.

Definition at line 3227 of file STM8AF\_STM8S.h.

**6.17.2.13 CC2G**

`_BITS` CC2G

Capture/compare 2 generation.

Definition at line 3150 of file STM8AF\_STM8S.h.

**6.17.2.14 CC2IE**

`_BITS` CC2IE

Capture/compare 2 interrupt enable.

Definition at line 3120 of file STM8AF\_STM8S.h.

**6.17.2.15 CC2IF**

`_BITS` CC2IF

Capture/compare 2 interrupt flag.

Definition at line 3130 of file STM8AF\_STM8S.h.

#### 6.17.2.16 CC2OF

`_BITS` CC2OF

Capture/compare 2 overcapture flag.

Definition at line 3140 of file STM8AF\_STM8S.h.

#### 6.17.2.17 CC2P

`_BITS` CC2P

Capture/compare 2 output polarity.

Definition at line 3228 of file STM8AF\_STM8S.h.

#### 6.17.2.18 CC2S

`_BITS` CC2S

Capture/compare 2 selection.

Definition at line 3183 of file STM8AF\_STM8S.h.

#### 6.17.2.19 CC3E

`_BITS` CC3E

Capture/compare 3 output enable.

Definition at line 3235 of file STM8AF\_STM8S.h.

#### 6.17.2.20 CC3G

`_BITS` CC3G

Capture/compare 3 generation.

Definition at line 3151 of file STM8AF\_STM8S.h.

**6.17.2.21 CC3IE**

`_BITS CC3IE`

Capture/compare 3 interrupt enable.

Definition at line 3121 of file STM8AF\_STM8S.h.

**6.17.2.22 CC3IF**

`_BITS CC3IF`

Capture/compare 3 interrupt flag.

Definition at line 3131 of file STM8AF\_STM8S.h.

**6.17.2.23 CC3OF**

`_BITS CC3OF`

Capture/compare 3 overcapture flag.

Definition at line 3141 of file STM8AF\_STM8S.h.

**6.17.2.24 CC3P**

`_BITS CC3P`

Capture/compare 3 output polarity.

Definition at line 3236 of file STM8AF\_STM8S.h.

**6.17.2.25 CC3S**

`_BITS CC3S`

Capture/compare 3 selection.

Definition at line 3205 of file STM8AF\_STM8S.h.

#### 6.17.2.26 CCER1

```
struct { ... } CCER1
```

TIM2 Capture/compare enable register 1 (TIM2\_CCER1)

#### 6.17.2.27 CCER2

```
struct { ... } CCER2
```

TIM2 Capture/compare enable register 2 (TIM2\_CCER2)

#### 6.17.2.28 CCMR1

```
union { ... } CCMR1
```

TIM2 Capture/compare mode register 1 (TIM2\_CCMR1)

#### 6.17.2.29 CCMR2

```
union { ... } CCMR2
```

TIM2 Capture/compare mode register 2 (TIM2\_CCMR2)

#### 6.17.2.30 CCMR3

```
union { ... } CCMR3
```

TIM2 Capture/compare mode register 3 (TIM2\_CCMR3)

#### 6.17.2.31 CCR1

```
\_BITS CCR1
```

16-bit capture/compare value 1 [15:8]

16-bit capture/compare value 1 [7:0]

Definition at line 3274 of file STM8AF\_STM8S.h.

**6.17.2.32 CCR1H**

```
struct { ... } CCR1H
```

TIM2 16-bit capture/compare value 1 high byte (TIM2\_CCR1H)

**6.17.2.33 CCR1L**

```
struct { ... } CCR1L
```

TIM2 16-bit capture/compare value 1 low byte (TIM2\_CCR1L)

**6.17.2.34 CCR2**

```
\_BITS CCR2
```

16-bit capture/compare value 2 [15:8]

16-bit capture/compare value 2 [7:0]

Definition at line 3286 of file STM8AF\_STM8S.h.

**6.17.2.35 CCR2H**

```
struct { ... } CCR2H
```

TIM2 16-bit capture/compare value 2 high byte (TIM2\_CCR2H)

**6.17.2.36 CCR2L**

```
struct { ... } CCR2L
```

TIM2 16-bit capture/compare value 2 low byte (TIM2\_CCR2L)

#### 6.17.2.37 CCR3

`_BITS` CCR3

16-bit capture/compare value 3 [15:8]

16-bit capture/compare value 3 [7:0]

Definition at line 3298 of file STM8AF\_STM8S.h.

#### 6.17.2.38 CCR3H

```
struct { ... } CCR3H
```

TIM2 16-bit capture/compare value 3 high byte (TIM2\_CCR3H)

#### 6.17.2.39 CCR3L

```
struct { ... } CCR3L
```

TIM2 16-bit capture/compare value 3 low byte (TIM2\_CCR3L)

#### 6.17.2.40 CEN

`_BITS` CEN

Counter enable.

Definition at line 3101 of file STM8AF\_STM8S.h.

#### 6.17.2.41 CNT

`_BITS` CNT

16-bit counter [15:8]

16-bit counter [7:0]

Definition at line 3243 of file STM8AF\_STM8S.h.

#### 6.17.2.42 CNTRH

```
struct { ... } CNTRH
```

TIM2 16-bit counter high byte (TIM2\_CNTRH)

#### 6.17.2.43 CNTRL

```
struct { ... } CNTRL
```

TIM2 16-bit counter low byte (TIM2\_CNTRL)

#### 6.17.2.44 CR1

```
struct { ... } CR1
```

TIM2 Control register 1 (TIM2\_CR1)

#### 6.17.2.45 EGR

```
struct { ... } EGR
```

TIM2 Event generation register (TIM2\_EGR)

#### 6.17.2.46 IC1F

```
\_BITS IC1F
```

Input capture 1 filter.

Definition at line 3172 of file STM8AF\_STM8S.h.

#### 6.17.2.47 IC1PSC

```
\_BITS IC1PSC
```

Input capture 1 prescaler.

Definition at line 3171 of file STM8AF\_STM8S.h.



#### 6.17.2.48 IC2F

`_BITS IC2F`

Input capture 2 filter.

Definition at line 3194 of file STM8AF\_STM8S.h.

#### 6.17.2.49 IC2PSC

`_BITS IC2PSC`

Input capture 2 prescaler.

Definition at line 3193 of file STM8AF\_STM8S.h.

#### 6.17.2.50 IC3F

`_BITS IC3F`

Input capture 3 filter.

Definition at line 3216 of file STM8AF\_STM8S.h.

#### 6.17.2.51 IC3PSC

`_BITS IC3PSC`

Input capture 3 prescaler.

Definition at line 3215 of file STM8AF\_STM8S.h.

#### 6.17.2.52 IER

```
struct { ... } IER
```

Reserved registers on selected devices (2B)

TIM2 Interrupt enable register (TIM2\_IER)

**6.17.2.53 IN** [1/3]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.17.2.54 IN** [2/3]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.17.2.55 IN** [3/3]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.17.2.56 OC1M**

```
\_BITS OC1M
```

Output compare 1 mode.

Definition at line 3164 of file STM8AF\_STM8S.h.

**6.17.2.57 OC1PE**

```
\_BITS OC1PE
```

Output compare 1 preload enable.

Definition at line 3163 of file STM8AF\_STM8S.h.

**6.17.2.58 OC2M**

```
\_BITS OC2M
```

Output compare 2 mode.

Definition at line 3186 of file STM8AF\_STM8S.h.

#### 6.17.2.59 OC2PE

`_BITS` OC2PE

Output compare 2 preload enable.

Definition at line 3185 of file STM8AF\_STM8S.h.

#### 6.17.2.60 OC3CE

`_BITS` OC3CE

Output compare 3 clear enable.

Definition at line 3209 of file STM8AF\_STM8S.h.

#### 6.17.2.61 OC3M

`_BITS` OC3M

Output compare 3 mode.

Definition at line 3208 of file STM8AF\_STM8S.h.

#### 6.17.2.62 OC3PE

`_BITS` OC3PE

Output compare 3 preload enable.

Definition at line 3207 of file STM8AF\_STM8S.h.

#### 6.17.2.63 OPM

`_BITS` OPM

One-pulse mode.

Definition at line 3104 of file STM8AF\_STM8S.h.

**6.17.2.64 OUT** [1/3]

```
struct { ... } OUT
```

bitwise access to register (output mode)

**6.17.2.65 OUT** [2/3]

```
struct { ... } OUT
```

bitwise access to register (output mode)

**6.17.2.66 OUT** [3/3]

```
struct { ... } OUT
```

bitwise access to register (output mode)

**6.17.2.67 PSC**

```
_BITS PSC
```

prescaler [3:0]

Definition at line 3255 of file STM8AF\_STM8S.h.

**6.17.2.68 PSCR**

```
struct { ... } PSCR
```

TIM2 16-bit prescaler high byte (TIM2\_PSCR)

**6.17.2.69 res**

```
_BITS res
```

Reserved.

Reserved, must be kept cleared.

Definition at line 3105 of file STM8AF\_STM8S.h.

#### 6.17.2.70 res2

`_BITS res2`

Reserved, must be kept cleared.

Reserved.

Definition at line 3142 of file STM8AF\_STM8S.h.

#### 6.17.2.71 SR1

```
struct { ... } SR1
```

TIM2 Status register 1 (TIM2\_SR1)

#### 6.17.2.72 SR2

```
struct { ... } SR2
```

TIM2 Status register 2 (TIM2\_SR2)

#### 6.17.2.73 UDIS

`_BITS UDIS`

Update disable.

Definition at line 3102 of file STM8AF\_STM8S.h.

#### 6.17.2.74 UG

`_BITS UG`

Update generation.

Definition at line 3148 of file STM8AF\_STM8S.h.

#### 6.17.2.75 UIE

`_BITS` UIE

Update interrupt enable.

Definition at line 3118 of file STM8AF\_STM8S.h.

#### 6.17.2.76 UIF

`_BITS` UIF

Update interrupt flag.

Definition at line 3128 of file STM8AF\_STM8S.h.

#### 6.17.2.77 URS

`_BITS` URS

Update request source.

Definition at line 3103 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.18 TIM3\_t Struct Reference

struct for controlling 16-Bit Timer 3 (TIM3)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
  - [\\_BITS CEN](#): 1  
*Counter enable.*
  - [\\_BITS UDIS](#): 1  
*Update disable.*
  - [\\_BITS URS](#): 1  
*Update request source.*
  - [\\_BITS OPM](#): 1  
*One-pulse mode.*
  - [\\_BITS res](#): 3  
*Reserved.*
  - [\\_BITS ARPE](#): 1  
*Auto-reload preload enable.*

} [CR1](#)

*TIM3 Control register 1 (TIM3\_CR1)*

- struct {
  - [\\_BITS UIE](#): 1  
*Update interrupt enable.*
  - [\\_BITS CC1IE](#): 1  
*Capture/compare 1 interrupt enable.*
  - [\\_BITS CC2IE](#): 1  
*Capture/compare 2 interrupt enable.*
  - [\\_BITS res](#): 5  
*Reserved.*

} [IER](#)

*TIM3 Interrupt enable register (TIM3\_IER)*

- struct {
  - [\\_BITS UIF](#): 1  
*Update interrupt flag.*
  - [\\_BITS CC1IF](#): 1  
*Capture/compare 1 interrupt flag.*
  - [\\_BITS CC2IF](#): 1  
*Capture/compare 2 interrupt flag.*
  - [\\_BITS res](#): 5  
*Reserved.*

} [SR1](#)

*TIM3 Status register 1 (TIM3\_SR1)*

- struct {
  - [\\_BITS res](#): 1  
*Reserved, must be kept cleared.*
  - [\\_BITS CC1OF](#): 1  
*Capture/compare 1 overcapture flag.*
  - [\\_BITS CC2OF](#): 1  
*Capture/compare 2 overcapture flag.*
  - [\\_BITS res2](#): 5  
*Reserved, must be kept cleared.*

} [SR2](#)

*TIM3 Status register 2 (TIM3\_SR2)*

- struct {
  - [\\_BITS UG](#): 1  
*Update generation.*

```

    _BITS CC1G: 1
        Capture/compare 1 generation.
    _BITS CC2G: 1
        Capture/compare 2 generation.
    _BITS res: 5
        Reserved.
} EGR

```

*TIM3 Event generation register (TIM3\_EGR)*

```

• union {
    struct {
        _BITS CC1S: 2
            Compare 1 selection.
        _BITS res: 1
            Reserved.
        _BITS OC1PE: 1
            Output compare 1 preload enable.
        _BITS OC1M: 3
            Output compare 1 mode.
        _BITS res2: 1
            Reserved.
    } OUT
        bitwise access to register (output mode)
    struct {
        _BITS CC1S: 2
            Capture 1 selection.
        _BITS IC1PSC: 2
            Input capture 1 prescaler.
        _BITS IC1F: 4
            Input capture 1 filter.
    } IN
        bitwise access to register (input mode)
} CCMR1

```

*TIM3 Capture/compare mode register 1 (TIM3\_CCMR1)*

```

• union {
    struct {
        _BITS CC2S: 2
            Capture/compare 2 selection.
        _BITS res: 1
            Reserved.
        _BITS OC2PE: 1
            Output compare 2 preload enable.
        _BITS OC2M: 3
            Output compare 2 mode.
        _BITS res2: 1
            Reserved.
    } OUT
        bitwise access to register (output mode)
    struct {
        _BITS CC2S: 2
            Capture/compare 2 selection.
        _BITS IC2PSC: 2
            Input capture 2 prescaler.
        _BITS IC2F: 4
            Input capture 2 filter.
    } IN
        bitwise access to register (input mode)

```



} CCMR2

*TIM3 Capture/compare mode register 2 (TIM3\_CCMR2)*

- struct {
    - [\\_BITS CC1E](#): 1  
*Capture/compare 1 output enable.*
    - [\\_BITS CC1P](#): 1  
*Capture/compare 1 output polarity.*
    - [\\_BITS res](#): 2  
*Reserved.*
    - [\\_BITS CC2E](#): 1  
*Capture/compare 2 output enable.*
    - [\\_BITS CC2P](#): 1  
*Capture/compare 2 output polarity.*
    - [\\_BITS res2](#): 2  
*Reserved.*
- } CCER1

*TIM3 Capture/compare enable register 1 (TIM3\_CCER1)*

- struct {
    - [\\_BITS CNT](#): 8  
*16-bit counter [15:8]*
- } CNTRH

*TIM3 16-bit counter high byte (TIM3\_CNTRH)*

- struct {
    - [\\_BITS CNT](#): 8  
*16-bit counter [7:0]*
- } CNTRL

*TIM3 16-bit counter low byte (TIM3\_CNTRL)*

- struct {
    - [\\_BITS PSC](#): 4  
*prescaler [3:0]*
    - [\\_BITS res](#): 4  
*Reserved.*
- } PSCR

*TIM3 16-bit prescaler high byte (TIM3\_PSCR)*

- struct {
    - [\\_BITS ARR](#): 8  
*16-bit auto-reload value [15:8]*
- } ARRH

*TIM3 16-bit auto-reload value high byte (TIM3\_ARRH)*

- struct {
    - [\\_BITS ARR](#): 8  
*16-bit auto-reload value [7:0]*
- } ARRL

*TIM3 16-bit auto-reload value low byte (TIM3\_ARRL)*

- struct {
    - [\\_BITS CCR1](#): 8  
*16-bit capture/compare value 1 [15:8]*
- } CCR1H

*TIM3 16-bit capture/compare value 1 high byte (TIM3\_CCR1H)*

- struct {  
     \_BITS CCR1: 8  
     16-bit capture/compare value 1 [7:0]  
   } CCR1L

*TIM3 16-bit capture/compare value 1 low byte (TIM3\_CCR1L)*

- struct {  
     \_BITS CCR2: 8  
     16-bit capture/compare value 2 [15:8]  
   } CCR2H

*TIM3 16-bit capture/compare value 2 high byte (TIM3\_CCR2H)*

- struct {  
     \_BITS CCR2: 8  
     16-bit capture/compare value 2 [7:0]  
   } CCR2L

*TIM3 16-bit capture/compare value 2 low byte (TIM3\_CCR2L)*

### 6.18.1 Detailed Description

struct for controlling 16-Bit Timer 3 (TIM3)

Definition at line 3516 of file STM8AF\_STM8S.h.

### 6.18.2 Field Documentation

#### 6.18.2.1 ARPE

\_BITS ARPE

Auto-reload preload enable.

Definition at line 3525 of file STM8AF\_STM8S.h.

#### 6.18.2.2 ARR

\_BITS ARR

16-bit auto-reload value [15:8]

16-bit auto-reload value [7:0]

Definition at line 3641 of file STM8AF\_STM8S.h.

### 6.18.2.3 ARRH

```
struct { ... } ARRH
```

TIM3 16-bit auto-reload value high byte (TIM3\_ARRH)

### 6.18.2.4 ARRL

```
struct { ... } ARRL
```

TIM3 16-bit auto-reload value low byte (TIM3\_ARRL)

### 6.18.2.5 CC1E

```
_BITS CC1E
```

Capture/compare 1 output enable.

Definition at line 3611 of file STM8AF\_STM8S.h.

### 6.18.2.6 CC1G

```
_BITS CC1G
```

Capture/compare 1 generation.

Definition at line 3559 of file STM8AF\_STM8S.h.

### 6.18.2.7 CC1IE

```
_BITS CC1IE
```

Capture/compare 1 interrupt enable.

Definition at line 3532 of file STM8AF\_STM8S.h.

**6.18.2.8 CC1IF**

`_BITS CC1IF`

Capture/compare 1 interrupt flag.

Definition at line 3541 of file STM8AF\_STM8S.h.

**6.18.2.9 CC1OF**

`_BITS CC1OF`

Capture/compare 1 overcapture flag.

Definition at line 3550 of file STM8AF\_STM8S.h.

**6.18.2.10 CC1P**

`_BITS CC1P`

Capture/compare 1 output polarity.

Definition at line 3612 of file STM8AF\_STM8S.h.

**6.18.2.11 CC1S**

`_BITS CC1S`

Compare 1 selection.

Capture 1 selection.

Definition at line 3570 of file STM8AF\_STM8S.h.

**6.18.2.12 CC2E**

`_BITS CC2E`

Capture/compare 2 output enable.

Definition at line 3614 of file STM8AF\_STM8S.h.

#### 6.18.2.13 CC2G

`_BITS` CC2G

Capture/compare 2 generation.

Definition at line 3560 of file STM8AF\_STM8S.h.

#### 6.18.2.14 CC2IE

`_BITS` CC2IE

Capture/compare 2 interrupt enable.

Definition at line 3533 of file STM8AF\_STM8S.h.

#### 6.18.2.15 CC2IF

`_BITS` CC2IF

Capture/compare 2 interrupt flag.

Definition at line 3542 of file STM8AF\_STM8S.h.

#### 6.18.2.16 CC2OF

`_BITS` CC2OF

Capture/compare 2 overcapture flag.

Definition at line 3551 of file STM8AF\_STM8S.h.

#### 6.18.2.17 CC2P

`_BITS` CC2P

Capture/compare 2 output polarity.

Definition at line 3615 of file STM8AF\_STM8S.h.

#### 6.18.2.18 CC2S

`_BITS` CC2S

Capture/compare 2 selection.

Definition at line 3592 of file STM8AF\_STM8S.h.

#### 6.18.2.19 CCER1

```
struct { ... } CCER1
```

TIM3 Capture/compare enable register 1 (TIM3\_CCER1)

#### 6.18.2.20 CCMR1

```
union { ... } CCMR1
```

TIM3 Capture/compare mode register 1 (TIM3\_CCMR1)

#### 6.18.2.21 CCMR2

```
union { ... } CCMR2
```

TIM3 Capture/compare mode register 2 (TIM3\_CCMR2)

#### 6.18.2.22 CCR1

`_BITS` CCR1

16-bit capture/compare value 1 [15:8]

16-bit capture/compare value 1 [7:0]

Definition at line 3653 of file STM8AF\_STM8S.h.

#### 6.18.2.23 CCR1H

```
struct { ... } CCR1H
```

TIM3 16-bit capture/compare value 1 high byte (TIM3\_CCR1H)

#### 6.18.2.24 CCR1L

```
struct { ... } CCR1L
```

TIM3 16-bit capture/compare value 1 low byte (TIM3\_CCR1L)

#### 6.18.2.25 CCR2

```
_BITS CCR2
```

16-bit capture/compare value 2 [15:8]

16-bit capture/compare value 2 [7:0]

Definition at line 3665 of file STM8AF\_STM8S.h.

#### 6.18.2.26 CCR2H

```
struct { ... } CCR2H
```

TIM3 16-bit capture/compare value 2 high byte (TIM3\_CCR2H)

#### 6.18.2.27 CCR2L

```
struct { ... } CCR2L
```

TIM3 16-bit capture/compare value 2 low byte (TIM3\_CCR2L)

#### 6.18.2.28 CEN

```
_BITS CEN
```

Counter enable.

Definition at line 3520 of file STM8AF\_STM8S.h.

#### 6.18.2.29 CNT

`_BITS` CNT

16-bit counter [15:8]

16-bit counter [7:0]

Definition at line 3622 of file STM8AF\_STM8S.h.

#### 6.18.2.30 CNTRH

```
struct { ... } CNTRH
```

TIM3 16-bit counter high byte (TIM3\_CNTRH)

#### 6.18.2.31 CNTRL

```
struct { ... } CNTRL
```

TIM3 16-bit counter low byte (TIM3\_CNTRL)

#### 6.18.2.32 CR1

```
struct { ... } CR1
```

TIM3 Control register 1 (TIM3\_CR1)

#### 6.18.2.33 EGR

```
struct { ... } EGR
```

TIM3 Event generation register (TIM3\_EGR)

#### 6.18.2.34 IC1F

`_BITS` IC1F

Input capture 1 filter.

Definition at line 3581 of file STM8AF\_STM8S.h.



#### 6.18.2.35 IC1PSC

`_BITS IC1PSC`

Input capture 1 prescaler.

Definition at line 3580 of file STM8AF\_STM8S.h.

#### 6.18.2.36 IC2F

`_BITS IC2F`

Input capture 2 filter.

Definition at line 3603 of file STM8AF\_STM8S.h.

#### 6.18.2.37 IC2PSC

`_BITS IC2PSC`

Input capture 2 prescaler.

Definition at line 3602 of file STM8AF\_STM8S.h.

#### 6.18.2.38 IER

```
struct { ... } IER
```

TIM3 Interrupt enable register (TIM3\_IER)

#### 6.18.2.39 IN [1/2]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.18.2.40 IN** [2/2]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.18.2.41 OC1M**

```
_BITS OC1M
```

Output compare 1 mode.

Definition at line 3573 of file STM8AF\_STM8S.h.

**6.18.2.42 OC1PE**

```
_BITS OC1PE
```

Output compare 1 preload enable.

Definition at line 3572 of file STM8AF\_STM8S.h.

**6.18.2.43 OC2M**

```
_BITS OC2M
```

Output compare 2 mode.

Definition at line 3595 of file STM8AF\_STM8S.h.

**6.18.2.44 OC2PE**

```
_BITS OC2PE
```

Output compare 2 preload enable.

Definition at line 3594 of file STM8AF\_STM8S.h.

#### 6.18.2.45 OPM

`_BITS` OPM

One-pulse mode.

Definition at line 3523 of file STM8AF\_STM8S.h.

#### 6.18.2.46 OUT [1/2]

`struct { ... } OUT`

bitwise access to register (output mode)

#### 6.18.2.47 OUT [2/2]

`struct { ... } OUT`

bitwise access to register (output mode)

#### 6.18.2.48 PSC

`_BITS` PSC

prescaler [3:0]

Definition at line 3634 of file STM8AF\_STM8S.h.

#### 6.18.2.49 PSCR

`struct { ... } PSCR`

TIM3 16-bit prescaler high byte (TIM3\_PSCR)

**6.18.2.50 res**

`_BITS res`

Reserved.

Reserved, must be kept cleared.

Definition at line 3524 of file STM8AF\_STM8S.h.

**6.18.2.51 res2**

`_BITS res2`

Reserved, must be kept cleared.

Reserved.

Definition at line 3552 of file STM8AF\_STM8S.h.

**6.18.2.52 SR1**

```
struct { ... } SR1
```

TIM3 Status register 1 (TIM3\_SR1)

**6.18.2.53 SR2**

```
struct { ... } SR2
```

TIM3 Status register 2 (TIM3\_SR2)

**6.18.2.54 UDIS**

`_BITS UDIS`

Update disable.

Definition at line 3521 of file STM8AF\_STM8S.h.

#### 6.18.2.55 UG

`_BITS` UG

Update generation.

Definition at line 3558 of file STM8AF\_STM8S.h.

#### 6.18.2.56 UIE

`_BITS` UIE

Update interrupt enable.

Definition at line 3531 of file STM8AF\_STM8S.h.

#### 6.18.2.57 UIF

`_BITS` UIF

Update interrupt flag.

Definition at line 3540 of file STM8AF\_STM8S.h.

#### 6.18.2.58 URS

`_BITS` URS

Update request source.

Definition at line 3522 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.19 TIM4\_t Struct Reference

struct for controlling 8-Bit Timer 4 (TIM4)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
  - [\\_BITS CEN](#): 1  
Counter enable.
  - [\\_BITS UDIS](#): 1  
Update disable.
  - [\\_BITS URS](#): 1  
Update request source.
  - [\\_BITS OPM](#): 1  
One-pulse mode.
  - [\\_BITS res](#): 3  
Reserved.
  - [\\_BITS ARPE](#): 1  
Auto-reload preload enable.

} [CR](#)

*TIM4 Control register (TIM4\_CR)*

- struct {
  - [\\_BITS UIE](#): 1  
Update interrupt enable.
  - [\\_BITS res](#): 7  
Reserved.

} [IER](#)

*Reserved registers on selected devices (2B)*

- struct {
  - [\\_BITS UIF](#): 1  
Update interrupt flag.
  - [\\_BITS res](#): 7  
Reserved.

} [SR](#)

*TIM4 Status register (TIM4\_SR)*

- struct {
  - [\\_BITS UG](#): 1  
Update generation.
  - [\\_BITS res](#): 7  
Reserved.

} [EGR](#)

*TIM4 Event Generation (TIM4\_EGR)*

- struct {
  - [\\_BITS CNT](#): 8  
8-bit counter

} [CNTR](#)

*TIM4 8-bit counter register (TIM4\_CNTR)*

- struct {
  - [\\_BITS PSC](#): 3  
clock prescaler
  - [\\_BITS res](#): 5  
Reserved.

} [PSCR](#)

*TIM4 clock prescaler (TIM4\_PSCR)*

- struct {  
    \_BITS ARR: 8  
        *auto-reload value*  
} ARR

*TIM4 8-bit auto-reload register (TIM4\_ARR)*

### 6.19.1 Detailed Description

struct for controlling 8-Bit Timer 4 (TIM4)

Definition at line 3819 of file STM8AF\_STM8S.h.

### 6.19.2 Field Documentation

#### 6.19.2.1 ARPE

\_BITS ARPE

Auto-reload preload enable.

Definition at line 3828 of file STM8AF\_STM8S.h.

#### 6.19.2.2 ARR [1/2]

\_BITS ARR

auto-reload value

Definition at line 3873 of file STM8AF\_STM8S.h.

#### 6.19.2.3 ARR [2/2]

struct { ... } ARR

TIM4 8-bit auto-reload register (TIM4\_ARR)

#### 6.19.2.4 CEN

`_BITS` CEN

Counter enable.

Definition at line 3823 of file STM8AF\_STM8S.h.

#### 6.19.2.5 CNT

`_BITS` CNT

8-bit counter

Definition at line 3861 of file STM8AF\_STM8S.h.

#### 6.19.2.6 CNTR

```
struct { ... } CNTR
```

TIM4 8-bit counter register (TIM4\_CNTR)

#### 6.19.2.7 CR

```
struct { ... } CR
```

TIM4 Control register (TIM4\_CR)

#### 6.19.2.8 EGR

```
struct { ... } EGR
```

TIM4 Event Generation (TIM4\_EGR)

#### 6.19.2.9 IER

```
struct { ... } IER
```

Reserved registers on selected devices (2B)

TIM4 Interrupt enable (TIM4\_IER)



#### 6.19.2.10 OPM

`_BITS` OPM

One-pulse mode.

Definition at line 3826 of file STM8AF\_STM8S.h.

#### 6.19.2.11 PSC

`_BITS` PSC

clock prescaler

Definition at line 3866 of file STM8AF\_STM8S.h.

#### 6.19.2.12 PSCR

```
struct { ... } PSCR
```

TIM4 clock prescaler (TIM4\_PSCR)

#### 6.19.2.13 res

`_BITS` res

Reserved.

Definition at line 3827 of file STM8AF\_STM8S.h.

#### 6.19.2.14 SR

```
struct { ... } SR
```

TIM4 Status register (TIM4\_SR)

#### 6.19.2.15 UDIS

`_BITS` UDIS

Update disable.

Definition at line 3824 of file STM8AF\_STM8S.h.

#### 6.19.2.16 UG

`_BITS` UG

Update generation.

Definition at line 3854 of file STM8AF\_STM8S.h.

#### 6.19.2.17 UIE

`_BITS` UIE

Update interrupt enable.

Definition at line 3840 of file STM8AF\_STM8S.h.

#### 6.19.2.18 UIF

`_BITS` UIF

Update interrupt flag.

Definition at line 3847 of file STM8AF\_STM8S.h.

#### 6.19.2.19 URS

`_BITS` URS

Update request source.

Definition at line 3825 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.20 TIM5\_† Struct Reference

struct for controlling 16-Bit Timer 5 (TIM5)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {
    - [\\_BITS CEN](#): 1  
*Counter enable.*
    - [\\_BITS UDIS](#): 1  
*Update disable.*
    - [\\_BITS URS](#): 1  
*Update request source.*
    - [\\_BITS OPM](#): 1  
*One-pulse mode.*
    - [\\_BITS res](#): 3  
*Reserved.*
    - [\\_BITS ARPE](#): 1  
*Auto-reload preload enable.*
- [} CR1](#)

*TIM5 Control register 1 (TIM5\_CR1)*

- struct {
    - [\\_BITS CCPC](#): 1  
*Capture/compare preloaded control.*
    - [\\_BITS res](#): 1  
*Reserved, forced by hardware to 0.*
    - [\\_BITS COMS](#): 1  
*Capture/compare control update selection.*
    - [\\_BITS res2](#): 1  
*Reserved, must be kept cleared.*
    - [\\_BITS MMS](#): 3  
*Master mode selection.*
    - [\\_BITS res3](#): 1  
*Reserved.*
- [} CR2](#)

*TIM5 Control register 2 (TIM5\_CR2)*

- struct {
    - [\\_BITS SMS](#): 3  
*Clock/trigger/slave mode selection.*
    - [\\_BITS res](#): 1  
*Reserved.*
    - [\\_BITS TS](#): 3  
*Trigger selection.*
    - [\\_BITS MSM](#): 1  
*Master/slave mode.*
- [} SMCR](#)

*Slave mode control register (TIM5\_SMCR)*

- struct {
  - [\\_BITS\\_UIE](#): 1  
Update interrupt enable.
  - [\\_BITS\\_CC1IE](#): 1  
Capture/compare 1 interrupt enable.
  - [\\_BITS\\_CC2IE](#): 1  
Capture/compare 2 interrupt enable.
  - [\\_BITS\\_CC3IE](#): 1  
Capture/compare 3 interrupt enable.
  - [\\_BITS\\_res](#): 2  
Reserved.
  - [\\_BITS\\_TIE](#): 1  
Trigger interrupt enable.
  - [\\_BITS\\_res2](#): 1  
Reserved.

} [IER](#)

*TIM5 Interrupt enable register (TIM5\_IER)*

- struct {
  - [\\_BITS\\_UIF](#): 1  
Update interrupt flag.
  - [\\_BITS\\_CC1IF](#): 1  
Capture/compare 1 interrupt flag.
  - [\\_BITS\\_CC2IF](#): 1  
Capture/compare 2 interrupt flag.
  - [\\_BITS\\_CC3IF](#): 1  
Capture/compare 3 interrupt flag.
  - [\\_BITS\\_res](#): 2  
Reserved.
  - [\\_BITS\\_TIF](#): 1  
Trigger interrupt flag.
  - [\\_BITS\\_res2](#): 1  
Reserved.

} [SR1](#)

*TIM5 Status register 1 (TIM5\_SR1)*

- struct {
  - [\\_BITS\\_res](#): 1  
Reserved.
  - [\\_BITS\\_CC1OF](#): 1  
Capture/compare 1 overcapture flag.
  - [\\_BITS\\_CC2OF](#): 1  
Capture/compare 2 overcapture flag.
  - [\\_BITS\\_CC3OF](#): 1  
Capture/compare 3 overcapture flag.
  - [\\_BITS\\_res2](#): 4  
Reserved.

} [SR2](#)

*TIM5 Status register 2 (TIM5\_SR2)*

- struct {
  - [\\_BITS\\_UG](#): 1  
Update generation.
  - [\\_BITS\\_CC1G](#): 1  
Capture/compare 1 generation.
  - [\\_BITS\\_CC2G](#): 1  
Capture/compare 2 generation.
  - [\\_BITS\\_CC3G](#): 1

```

    Capture/compare 3 generation.
    _BITS res: 2
    Reserved.
    _BITS TG: 1
    Trigger generation.
    _BITS res2: 1
    Reserved.
} EGR

TIM5 Event generation register (TIM5_EGR)
• union {
    struct {
        _BITS CC1S: 2
        Compare 1 selection.
        _BITS res: 1
        Reserved.
        _BITS OC1PE: 1
        Output compare 1 preload enable.
        _BITS OC1M: 3
        Output compare 1 mode.
        _BITS res2: 1
        Reserved.
    } OUT
    bitwise access to register (output mode)
    struct {
        _BITS CC1S: 2
        Capture 1 selection.
        _BITS IC1PSC: 2
        Input capture 1 prescaler.
        _BITS IC1F: 4
        Input capture 1 filter.
    } IN
    bitwise access to register (input mode)
} CCMR1

```

```

TIM5 Capture/compare mode register 1 (TIM5_CCMR1)
• union {
    struct {
        _BITS CC2S: 2
        Capture/compare 2 selection.
        _BITS res: 1
        Reserved.
        _BITS OC2PE: 1
        Output compare 2 preload enable.
        _BITS OC2M: 3
        Output compare 2 mode.
        _BITS res2: 1
        Reserved.
    } OUT
    bitwise access to register (output mode)
    struct {
        _BITS CC2S: 2
        Capture/compare 2 selection.
        _BITS IC2PSC: 2
        Input capture 2 prescaler.
        _BITS IC2F: 4
        Input capture 2 filter.
    } IN
}

```

*bitwise access to register (input mode)*  
**} CCMR2**

*TIM5 Capture/compare mode register 2 (TIM5\_CCMR2)*

- union {
  - struct {
    - \_BITS CC3S:** 2  
*Capture/compare 3 selection.*
    - \_BITS res:** 1  
*Reserved.*
    - \_BITS OC3PE:** 1  
*Output compare 3 preload enable.*
    - \_BITS OC3M:** 3  
*Output compare 3 mode.*
    - \_BITS res2:** 1  
*Reserved.*
  - OUT**  
*bitwise access to register (output mode)*
  - struct {
    - \_BITS CC3S:** 2  
*Capture/compare 3 selection.*
    - \_BITS IC3PSC:** 2  
*Input capture 3 prescaler.*
    - \_BITS IC3F:** 4  
*Input capture 3 filter.*
  - IN**  
*bitwise access to register (input mode)*

**} CCMR3**

*TIM5 Capture/compare mode register 3 (TIM5\_CCMR3)*

- struct {
  - \_BITS CC1E:** 1  
*Capture/compare 1 output enable.*
  - \_BITS CC1P:** 1  
*Capture/compare 1 output polarity.*
  - \_BITS res:** 2  
*Reserved.*
  - \_BITS CC2E:** 1  
*Capture/compare 2 output enable.*
  - \_BITS CC2P:** 1  
*Capture/compare 2 output polarity.*
  - \_BITS res2:** 2  
*Reserved.*

**} CCER1**

*TIM5 Capture/compare enable register 1 (TIM5\_CCER1)*

- struct {
  - \_BITS CC3E:** 1  
*Capture/compare 3 output enable.*
  - \_BITS CC3P:** 1  
*Capture/compare 3 output polarity.*
  - \_BITS res:** 6  
*Reserved.*

**} CCER2**

*TIM5 Capture/compare enable register 2 (TIM5\_CCER2)*

- struct {

```

    _BITS CNT: 8
        16-bit counter [15:8]
} CNTRH

```

*TIM5 16-bit counter high byte (TIM5\_CNTRH)*

```

• struct {
    _BITS CNT: 8
        16-bit counter [7:0]
} CNTRL

```

*TIM5 16-bit counter low byte (TIM5\_CNTRL)*

```

• struct {
    _BITS PSC: 4
        clock prescaler
    _BITS res: 4
        Reserved.
} PSCR

```

*TIM5 prescaler (TIM5\_PSCR)*

```

• struct {
    _BITS ARR: 8
        16-bit auto-reload value [15:8]
} ARRH

```

*TIM5 16-bit auto-reload value high byte (TIM5\_ARRH)*

```

• struct {
    _BITS ARR: 8
        16-bit auto-reload value [7:0]
} ARRL

```

*TIM5 16-bit auto-reload value low byte (TIM5\_ARRL)*

```

• struct {
    _BITS CCR1: 8
        16-bit capture/compare value 1 [15:8]
} CCR1H

```

*TIM5 16-bit capture/compare value 1 high byte (TIM5\_CCR1H)*

```

• struct {
    _BITS CCR1: 8
        16-bit capture/compare value 1 [7:0]
} CCR1L

```

*TIM5 16-bit capture/compare value 1 low byte (TIM5\_CCR1L)*

```

• struct {
    _BITS CCR2: 8
        16-bit capture/compare value 2 [15:8]
} CCR2H

```

*TIM5 16-bit capture/compare value 2 high byte (TIM5\_CCR2H)*

```

• struct {
    _BITS CCR2: 8
        16-bit capture/compare value 2 [7:0]
} CCR2L

```

*TIM5 16-bit capture/compare value 2 low byte (TIM5\_CCR2L)*

- struct {
  - `_BITS CCR3: 8`  
16-bit capture/compare value 3 [15:8]
- } `CCR3H`

*TIM5 16-bit capture/compare value 3 high byte (TIM5\_CCR3H)*
- struct {
  - `_BITS CCR3: 8`  
16-bit capture/compare value 3 [7:0]
- } `CCR3L`

*TIM5 16-bit capture/compare value 3 low byte (TIM5\_CCR3L)*

### 6.20.1 Detailed Description

struct for controlling 16-Bit Timer 5 (TIM5)

Definition at line 3944 of file STM8AF\_STM8S.h.

### 6.20.2 Field Documentation

#### 6.20.2.1 ARPE

`_BITS ARPE`

Auto-reload preload enable.

Definition at line 3953 of file STM8AF\_STM8S.h.

#### 6.20.2.2 ARR

`_BITS ARR`

16-bit auto-reload value [15:8]

16-bit auto-reload value [7:0]

Definition at line 4129 of file STM8AF\_STM8S.h.

#### 6.20.2.3 ARRH

struct { ... } `ARRH`

TIM5 16-bit auto-reload value high byte (TIM5\_ARRH)



#### 6.20.2.4 ARRL

```
struct { ... } ARRL
```

TIM5 16-bit auto-reload value low byte (TIM5\_ARRL)

#### 6.20.2.5 CC1E

```
_BITS CC1E
```

Capture/compare 1 output enable.

Definition at line 4091 of file STM8AF\_STM8S.h.

#### 6.20.2.6 CC1G

```
_BITS CC1G
```

Capture/compare 1 generation.

Definition at line 4014 of file STM8AF\_STM8S.h.

#### 6.20.2.7 CC1IE

```
_BITS CC1IE
```

Capture/compare 1 interrupt enable.

Definition at line 3980 of file STM8AF\_STM8S.h.

#### 6.20.2.8 CC1IF

```
_BITS CC1IF
```

Capture/compare 1 interrupt flag.

Definition at line 3992 of file STM8AF\_STM8S.h.

**6.20.2.9 CC10F**

`_BITS CC10F`

Capture/compare 1 overcapture flag.

Definition at line 4004 of file STM8AF\_STM8S.h.

**6.20.2.10 CC1P**

`_BITS CC1P`

Capture/compare 1 output polarity.

Definition at line 4092 of file STM8AF\_STM8S.h.

**6.20.2.11 CC1S**

`_BITS CC1S`

Compare 1 selection.

Capture 1 selection.

Definition at line 4028 of file STM8AF\_STM8S.h.

**6.20.2.12 CC2E**

`_BITS CC2E`

Capture/compare 2 output enable.

Definition at line 4094 of file STM8AF\_STM8S.h.

**6.20.2.13 CC2G**

`_BITS CC2G`

Capture/compare 2 generation.

Definition at line 4015 of file STM8AF\_STM8S.h.

#### 6.20.2.14 CC2IE

`_BITS CC2IE`

Capture/compare 2 interrupt enable.

Definition at line 3981 of file STM8AF\_STM8S.h.

#### 6.20.2.15 CC2IF

`_BITS CC2IF`

Capture/compare 2 interrupt flag.

Definition at line 3993 of file STM8AF\_STM8S.h.

#### 6.20.2.16 CC2OF

`_BITS CC2OF`

Capture/compare 2 overcapture flag.

Definition at line 4005 of file STM8AF\_STM8S.h.

#### 6.20.2.17 CC2P

`_BITS CC2P`

Capture/compare 2 output polarity.

Definition at line 4095 of file STM8AF\_STM8S.h.

#### 6.20.2.18 CC2S

`_BITS CC2S`

Capture/compare 2 selection.

Definition at line 4050 of file STM8AF\_STM8S.h.

**6.20.2.19 CC3E**

`_BITS CC3E`

Capture/compare 3 output enable.

Definition at line 4102 of file STM8AF\_STM8S.h.

**6.20.2.20 CC3G**

`_BITS CC3G`

Capture/compare 3 generation.

Definition at line 4016 of file STM8AF\_STM8S.h.

**6.20.2.21 CC3IE**

`_BITS CC3IE`

Capture/compare 3 interrupt enable.

Definition at line 3982 of file STM8AF\_STM8S.h.

**6.20.2.22 CC3IF**

`_BITS CC3IF`

Capture/compare 3 interrupt flag.

Definition at line 3994 of file STM8AF\_STM8S.h.

**6.20.2.23 CC3OF**

`_BITS CC3OF`

Capture/compare 3 overcapture flag.

Definition at line 4006 of file STM8AF\_STM8S.h.

#### 6.20.2.24 CC3P

`_BITS` CC3P

Capture/compare 3 output polarity.

Definition at line 4103 of file STM8AF\_STM8S.h.

#### 6.20.2.25 CC3S

`_BITS` CC3S

Capture/compare 3 selection.

Definition at line 4072 of file STM8AF\_STM8S.h.

#### 6.20.2.26 CCER1

```
struct { ... } CCER1
```

TIM5 Capture/compare enable register 1 (TIM5\_CCER1)

#### 6.20.2.27 CCER2

```
struct { ... } CCER2
```

TIM5 Capture/compare enable register 2 (TIM5\_CCER2)

#### 6.20.2.28 CCMR1

```
union { ... } CCMR1
```

TIM5 Capture/compare mode register 1 (TIM5\_CCMR1)

#### 6.20.2.29 CCMR2

```
union { ... } CCMR2
```

TIM5 Capture/compare mode register 2 (TIM5\_CCMR2)

#### 6.20.2.30 CCMR3

```
union { ... } CCMR3
```

TIM5 Capture/compare mode register 3 (TIM5\_CCMR3)

#### 6.20.2.31 CCPC

```
_BITS CCPC
```

Capture/compare preloaded control.

Definition at line 3959 of file STM8AF\_STM8S.h.

#### 6.20.2.32 CCR1

```
_BITS CCR1
```

16-bit capture/compare value 1 [15:8]

16-bit capture/compare value 1 [7:0]

Definition at line 4141 of file STM8AF\_STM8S.h.

#### 6.20.2.33 CCR1H

```
struct { ... } CCR1H
```

TIM5 16-bit capture/compare value 1 high byte (TIM5\_CCR1H)

#### 6.20.2.34 CCR1L

```
struct { ... } CCR1L
```

TIM5 16-bit capture/compare value 1 low byte (TIM5\_CCR1L)

### 6.20.2.35 CCR2

`_BITS` CCR2

16-bit capture/compare value 2 [15:8]

16-bit capture/compare value 2 [7:0]

Definition at line 4153 of file STM8AF\_STM8S.h.

### 6.20.2.36 CCR2H

```
struct { ... } CCR2H
```

TIM5 16-bit capture/compare value 2 high byte (TIM5\_CCR2H)

### 6.20.2.37 CCR2L

```
struct { ... } CCR2L
```

TIM5 16-bit capture/compare value 2 low byte (TIM5\_CCR2L)

### 6.20.2.38 CCR3

`_BITS` CCR3

16-bit capture/compare value 3 [15:8]

16-bit capture/compare value 3 [7:0]

Definition at line 4165 of file STM8AF\_STM8S.h.

### 6.20.2.39 CCR3H

```
struct { ... } CCR3H
```

TIM5 16-bit capture/compare value 3 high byte (TIM5\_CCR3H)

#### 6.20.2.40 CCR3L

```
struct { ... } CCR3L
```

TIM5 16-bit capture/compare value 3 low byte (TIM5\_CCR3L)

#### 6.20.2.41 CEN

```
_BITS CEN
```

Counter enable.

Definition at line 3948 of file STM8AF\_STM8S.h.

#### 6.20.2.42 CNT

```
_BITS CNT
```

16-bit counter [15:8]

16-bit counter [7:0]

Definition at line 4110 of file STM8AF\_STM8S.h.

#### 6.20.2.43 CNTRH

```
struct { ... } CNTRH
```

TIM5 16-bit counter high byte (TIM5\_CNTRH)

#### 6.20.2.44 CNTRL

```
struct { ... } CNTRL
```

TIM5 16-bit counter low byte (TIM5\_CNTRL)



#### 6.20.2.45 COMS

`_BITS` COMS

Capture/compare control update selection.

Definition at line 3961 of file STM8AF\_STM8S.h.

#### 6.20.2.46 CR1

```
struct { ... } CR1
```

TIM5 Control register 1 (TIM5\_CR1)

#### 6.20.2.47 CR2

```
struct { ... } CR2
```

TIM5 Control register 2 (TIM5\_CR2)

#### 6.20.2.48 EGR

```
struct { ... } EGR
```

TIM5 Event generation register (TIM5\_EGR)

#### 6.20.2.49 IC1F

`_BITS` IC1F

Input capture 1 filter.

Definition at line 4039 of file STM8AF\_STM8S.h.

#### 6.20.2.50 IC1PSC

`_BITS` IC1PSC

Input capture 1 prescaler.

Definition at line 4038 of file STM8AF\_STM8S.h.

**6.20.2.51 IC2F**

`_BITS IC2F`

Input capture 2 filter.

Definition at line 4061 of file STM8AF\_STM8S.h.

**6.20.2.52 IC2PSC**

`_BITS IC2PSC`

Input capture 2 prescaler.

Definition at line 4060 of file STM8AF\_STM8S.h.

**6.20.2.53 IC3F**

`_BITS IC3F`

Input capture 3 filter.

Definition at line 4083 of file STM8AF\_STM8S.h.

**6.20.2.54 IC3PSC**

`_BITS IC3PSC`

Input capture 3 prescaler.

Definition at line 4082 of file STM8AF\_STM8S.h.

**6.20.2.55 IER**

```
struct { ... } IER
```

TIM5 Interrupt enable register (TIM5\_IER)

**6.20.2.56 IN** [1/3]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.20.2.57 IN** [2/3]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.20.2.58 IN** [3/3]

```
struct { ... } IN
```

bitwise access to register (input mode)

**6.20.2.59 MMS**

```
__BITS MMS
```

Master mode selection.

Definition at line 3963 of file STM8AF\_STM8S.h.

**6.20.2.60 MSM**

```
__BITS MSM
```

Master/slave mode.

Definition at line 3973 of file STM8AF\_STM8S.h.

**6.20.2.61 OC1M**

```
__BITS OC1M
```

Output compare 1 mode.

Definition at line 4031 of file STM8AF\_STM8S.h.

**6.20.2.62 OC1PE**

`_BITS OC1PE`

Output compare 1 preload enable.

Definition at line 4030 of file STM8AF\_STM8S.h.

**6.20.2.63 OC2M**

`_BITS OC2M`

Output compare 2 mode.

Definition at line 4053 of file STM8AF\_STM8S.h.

**6.20.2.64 OC2PE**

`_BITS OC2PE`

Output compare 2 preload enable.

Definition at line 4052 of file STM8AF\_STM8S.h.

**6.20.2.65 OC3M**

`_BITS OC3M`

Output compare 3 mode.

Definition at line 4075 of file STM8AF\_STM8S.h.

**6.20.2.66 OC3PE**

`_BITS OC3PE`

Output compare 3 preload enable.

Definition at line 4074 of file STM8AF\_STM8S.h.

#### 6.20.2.67 OPM

`_BITS` OPM

One-pulse mode.

Definition at line 3951 of file STM8AF\_STM8S.h.

#### 6.20.2.68 OUT [1/3]

```
struct { ... } OUT
```

bitwise access to register (output mode)

#### 6.20.2.69 OUT [2/3]

```
struct { ... } OUT
```

bitwise access to register (output mode)

#### 6.20.2.70 OUT [3/3]

```
struct { ... } OUT
```

bitwise access to register (output mode)

#### 6.20.2.71 PSC

`_BITS` PSC

clock prescaler

Definition at line 4122 of file STM8AF\_STM8S.h.

#### 6.20.2.72 PSCR

```
struct { ... } PSCR
```

TIM5 prescaler (TIM5\_PSCR)

**6.20.2.73 res**

`_BITS res`

Reserved.

Reserved, forced by hardware to 0.

Definition at line 3952 of file STM8AF\_STM8S.h.

**6.20.2.74 res2**

`_BITS res2`

Reserved, must be kept cleared.

Reserved.

Definition at line 3962 of file STM8AF\_STM8S.h.

**6.20.2.75 res3**

`_BITS res3`

Reserved.

Definition at line 3964 of file STM8AF\_STM8S.h.

**6.20.2.76 SMCR**

```
struct { ... } SMCR
```

Slave mode control register (TIM5\_SMCR)

**6.20.2.77 SMS**

`_BITS SMS`

Clock/trigger/slave mode selection.

Definition at line 3970 of file STM8AF\_STM8S.h.

**6.20.2.78 SR1**

```
struct { ... } SR1
```

TIM5 Status register 1 (TIM5\_SR1)

**6.20.2.79 SR2**

```
struct { ... } SR2
```

TIM5 Status register 2 (TIM5\_SR2)

**6.20.2.80 TG**

[\\_BITS](#) TG

Trigger generation.

Definition at line 4018 of file STM8AF\_STM8S.h.

**6.20.2.81 TIE**

[\\_BITS](#) TIE

Trigger interrupt enable.

Definition at line 3984 of file STM8AF\_STM8S.h.

**6.20.2.82 TIF**

[\\_BITS](#) TIF

Trigger interrupt flag.

Definition at line 3996 of file STM8AF\_STM8S.h.

**6.20.2.83 TS**

`_BITS` TS

Trigger selection.

Definition at line 3972 of file STM8AF\_STM8S.h.

**6.20.2.84 UDIS**

`_BITS` UDIS

Update disable.

Definition at line 3949 of file STM8AF\_STM8S.h.

**6.20.2.85 UG**

`_BITS` UG

Update generation.

Definition at line 4013 of file STM8AF\_STM8S.h.

**6.20.2.86 UIE**

`_BITS` UIE

Update interrupt enable.

Definition at line 3979 of file STM8AF\_STM8S.h.

**6.20.2.87 UIF**

`_BITS` UIF

Update interrupt flag.

Definition at line 3991 of file STM8AF\_STM8S.h.



## 6.20.2.88 URS

`_BITS URS`

Update request source.

Definition at line 3950 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.21 TIM6\_t Struct Reference

struct for controlling 8-Bit Timer 6 (TIM6)

`#include <STM8AF_STM8S.h>`

## Data Fields

- struct {
  - `_BITS CEN: 1`  
*Counter enable.*
  - `_BITS UDIS: 1`  
*Update disable.*
  - `_BITS URS: 1`  
*Update request source.*
  - `_BITS OPM: 1`  
*One-pulse mode.*
  - `_BITS res: 3`  
*Reserved.*
  - `_BITS ARPE: 1`  
*Auto-reload preload enable.*
 } `CR1`

*TIM6 Control register (TIM6\_CR1)*

- struct {
  - `_BITS res: 4`  
*Reserved.*
  - `_BITS MMS: 3`  
*Master mode selection.*
  - `_BITS res2: 1`  
*Reserved.*
 } `CR2`

*TIM6 Control register (TIM6\_CR2)*

- struct {
  - `_BITS SMS: 3`  
*Clock/trigger/slave mode selection.*
  - `_BITS res: 1`  
*Reserved.*
  - `_BITS TS: 3`  
*Trigger selection.*
  - `_BITS MSM: 1`  
*Master/slave mode*
 } `SMCR`

*Master/slave mode #define \_TIM5\_CR1\_RESET\_VALUE ((uint8\_t) 0x00) ///< TIM5 control register 1 reset value.*

```

    Slave mode control register (TIM6_SMCR)
    • struct {
        _BITS UIE: 1
        Update interrupt enable.
        _BITS res: 7
        Reserved.
    } IER

    TIM6 Interrupt enable (TIM6_IER)
    • struct {
        _BITS UIF: 1
        Update interrupt flag.
        _BITS res: 7
        Reserved.
    } SR

    TIM6 Status register (TIM6_SR)
    • struct {
        _BITS UG: 1
        Update generation.
        _BITS res: 7
        Reserved.
    } EGR

    TIM6 Event Generation (TIM6_EGR)
    • struct {
        _BITS CNT: 8
        8-bit counter
    } CNTR

    TIM6 8-bit counter register (TIM6_CNTR)
    • struct {
        _BITS PSC: 3
        clock prescaler
        _BITS res: 5
        Reserved.
    } PSCR

    TIM6 clock prescaler (TIM6_PSCR)
    • struct {
        _BITS ARR: 8
        auto-reload value
    } ARR

    TIM6 8-bit auto-reload register (TIM6_ARR)

```

### 6.21.1 Detailed Description

struct for controlling 8-Bit Timer 6 (TIM6)

Definition at line 4392 of file STM8AF\_STM8S.h.

### 6.21.2 Field Documentation

#### 6.21.2.1 ARPE

`_BITS` ARPE

Auto-reload preload enable.

Definition at line 4401 of file STM8AF\_STM8S.h.

#### 6.21.2.2 ARR [1/2]

`_BITS` ARR

auto-reload value

Definition at line 4458 of file STM8AF\_STM8S.h.

#### 6.21.2.3 ARR [2/2]

```
struct { ... } ARR
```

TIM6 8-bit auto-reload register (TIM6\_ARR)

#### 6.21.2.4 CEN

`_BITS` CEN

Counter enable.

Definition at line 4396 of file STM8AF\_STM8S.h.

#### 6.21.2.5 CNT

`_BITS` CNT

8-bit counter

Definition at line 4446 of file STM8AF\_STM8S.h.

#### 6.21.2.6 CNTR

```
struct { ... } CNTR
```

TIM6 8-bit counter register (TIM6\_CNTR)

#### 6.21.2.7 CR1

```
struct { ... } CR1
```

TIM6 Control register (TIM6\_CR1)

#### 6.21.2.8 CR2

```
struct { ... } CR2
```

TIM6 Control register (TIM6\_CR2)

#### 6.21.2.9 EGR

```
struct { ... } EGR
```

TIM6 Event Generation (TIM6\_EGR)

#### 6.21.2.10 IER

```
struct { ... } IER
```

TIM6 Interrupt enable (TIM6\_IER)

#### 6.21.2.11 MMS

`_BITS` MMS

Master mode selection.

Definition at line 4408 of file STM8AF\_STM8S.h.

#### 6.21.2.12 MSM

`_BITS` MSM

Master/slave mode #define \_TIM5\_CR1\_RESET\_VALUE ((uint8\_t) 0x00) ///< TIM5 control register 1 reset value.

Definition at line 4418 of file STM8AF\_STM8S.h.

#### 6.21.2.13 OPM

`_BITS` OPM

One-pulse mode.

Definition at line 4399 of file STM8AF\_STM8S.h.

#### 6.21.2.14 PSC

`_BITS` PSC

clock prescaler

Definition at line 4451 of file STM8AF\_STM8S.h.

#### 6.21.2.15 PSCR

```
struct { ... } PSCR
```

TIM6 clock prescaler (TIM6\_PSCR)

#### 6.21.2.16 res

`_BITS` res

Reserved.

Definition at line 4400 of file STM8AF\_STM8S.h.

**6.21.2.17 res2**

`_BITS res2`

Reserved.

Definition at line 4409 of file STM8AF\_STM8S.h.

**6.21.2.18 SMCR**

```
struct { ... } SMCR
```

Slave mode control register (TIM6\_SMCR)

**6.21.2.19 SMS**

`_BITS SMS`

Clock/trigger/slave mode selection.

Definition at line 4415 of file STM8AF\_STM8S.h.

**6.21.2.20 SR**

```
struct { ... } SR
```

TIM6 Status register (TIM6\_SR)

**6.21.2.21 TS**

`_BITS TS`

Trigger selection.

Definition at line 4417 of file STM8AF\_STM8S.h.

#### 6.21.2.22 UDIS

`_BITS` UDIS

Update disable.

Definition at line 4397 of file STM8AF\_STM8S.h.

#### 6.21.2.23 UG

`_BITS` UG

Update generation.

Definition at line 4439 of file STM8AF\_STM8S.h.

#### 6.21.2.24 UIE

`_BITS` UIE

Update interrupt enable.

Definition at line 4425 of file STM8AF\_STM8S.h.

#### 6.21.2.25 UIF

`_BITS` UIF

Update interrupt flag.

Definition at line 4432 of file STM8AF\_STM8S.h.

#### 6.21.2.26 URS

`_BITS` URS

Update request source.

Definition at line 4398 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.22 UART1\_t Struct Reference

struct for controlling Universal Asynchronous Receiver Transmitter 1 (UART1)

```
#include <STM8AF_STM8S.h>
```

### Data Fields

- struct {
    - [\\_BITS PE](#): 1  
*Parity error.*
    - [\\_BITS FE](#): 1  
*Framing error.*
    - [\\_BITS NF](#): 1  
*Noise flag.*
    - [\\_BITS OR](#): 1  
*LIN Header Error (LIN slave mode) / Overrun error.*
    - [\\_BITS IDLE](#): 1  
*IDLE line detected.*
    - [\\_BITS RXNE](#): 1  
*Read data register not empty.*
    - [\\_BITS TC](#): 1  
*Transmission complete.*
    - [\\_BITS TXE](#): 1  
*Transmit data register empty.*
- } [SR](#)

*UART1 Status register (UART1\_SR)*

- struct {
    - [\\_BITS DATA](#): 8  
*UART1 data.*
- } [DR](#)

*UART1 data register (UART1\_DR)*

- struct {
    - [\\_BITS DIV\\_4\\_11](#): 8  
*UART\_DIV bits [11:4].*
- } [BRR1](#)

*UART1 Baud rate register 1 (UART1\_BRR1)*

- struct {
    - [\\_BITS DIV\\_0\\_3](#): 4  
*UART\_DIV bits [3:0].*
    - [\\_BITS DIV\\_12\\_15](#): 4  
*UART\_DIV bits [15:12].*
- } [BRR2](#)

*UART1 Baud rate register 2 (UART1\_BRR2)*

- struct {
  - [\\_BITS PIEN](#): 1  
*Parity interrupt enable.*
  - [\\_BITS PS](#): 1  
*Parity selection.*
  - [\\_BITS PCEN](#): 1  
*Parity control enable.*



```

    _BITS WAKE: 1
        Wakeup method.
    _BITS M: 1
        word length
    _BITS UARTD: 1
        UART Disable (for low power consumption)
    _BITS T8: 1
        Transmit Data bit 8 (in 9-bit mode)
    _BITS R8: 1
        Receive Data bit 8 (in 9-bit mode)
} CR1

```

UART1 Control register 1 (UART1\_CR1)

```

• struct {
    _BITS SBK: 1
        Send break.
    _BITS RWU: 1
        Receiver wakeup.
    _BITS REN: 1
        Receiver enable.
    _BITS TEN: 1
        Transmitter enable.
    _BITS ILIEN: 1
        IDLE Line interrupt enable.
    _BITS RIEN: 1
        Receiver interrupt enable.
    _BITS TCIE: 1
        Transmission complete interrupt enable.
    _BITS TIEN: 1
        Transmitter interrupt enable.
} CR2

```

UART1 Control register 2 (UART1\_CR2)

```

• struct {
    _BITS LBCL: 1
        Last bit clock pulse.
    _BITS CPHA: 1
        Clock phase.
    _BITS CPOL: 1
        Clock polarity.
    _BITS CKEN: 1
        Clock enable.
    _BITS STOP: 2
        STOP bits.
    _BITS LINEN: 1
        LIN mode enable.
    _BITS res: 1
        Reserved, must be kept cleared.
} CR3

```

UART1 Control register 3 (UART1\_CR3)

```

• struct {
    _BITS ADD: 4
        Address of the UART node.
    _BITS LBDF: 1
        LIN Break Detection Flag.
    _BITS LBDL: 1
        LIN Break Detection Length.
}

```

```

    _BITS LBDIEN: 1
        LIN Break Detection Interrupt Enable.
    _BITS res: 1
        Reserved, must be kept cleared.
} CR4

```

UART1 Control register 4 (UART1\_CR4)

```

• struct {
    _BITS res: 1
        Reserved, must be kept cleared.
    _BITS IREN: 1
        IrDA mode Enable.
    _BITS IRLP: 1
        IrDA Low Power.
    _BITS HDSEL: 1
        Half-Duplex Selection.
    _BITS NACK: 1
        Smartcard NACK enable.
    _BITS SCEN: 1
        Smartcard mode enable.
    _BITS res2: 2
        Reserved, must be kept cleared.
} CR5

```

UART1 Control register 5 (UART1\_CR5)

```

• struct {
    _BITS GT: 8
        UART1 guard time.
} GTR

```

UART1 guard time register (UART1\_GTR)

```

• struct {
    _BITS PSC: 8
        UART1 prescaler.
} PSCR

```

UART1 prescaler register (UART1\_PSCR)

### 6.22.1 Detailed Description

struct for controlling Universal Asynchronous Receiver Transmitter 1 (UART1)

Definition at line 1623 of file STM8AF\_STM8S.h.

### 6.22.2 Field Documentation

#### 6.22.2.1 ADD

`_BITS ADD`

Address of the UART node.

Definition at line 1697 of file STM8AF\_STM8S.h.

#### 6.22.2.2 BRR1

```
struct { ... } BRR1
```

UART1 Baud rate register 1 (UART1\_BRR1)

#### 6.22.2.3 BRR2

```
struct { ... } BRR2
```

UART1 Baud rate register 2 (UART1\_BRR2)

#### 6.22.2.4 CKEN

```
_BITS CKEN
```

Clock enable.

Definition at line 1688 of file STM8AF\_STM8S.h.

#### 6.22.2.5 CPHA

```
_BITS CPHA
```

Clock phase.

Definition at line 1686 of file STM8AF\_STM8S.h.

#### 6.22.2.6 CPOL

```
_BITS CPOL
```

Clock polarity.

Definition at line 1687 of file STM8AF\_STM8S.h.

#### 6.22.2.7 CR1

```
struct { ... } CR1
```

UART1 Control register 1 (UART1\_CR1)

#### 6.22.2.8 CR2

```
struct { ... } CR2
```

UART1 Control register 2 (UART1\_CR2)

#### 6.22.2.9 CR3

```
struct { ... } CR3
```

UART1 Control register 3 (UART1\_CR3)

#### 6.22.2.10 CR4

```
struct { ... } CR4
```

UART1 Control register 4 (UART1\_CR4)

#### 6.22.2.11 CR5

```
struct { ... } CR5
```

UART1 Control register 5 (UART1\_CR5)

#### 6.22.2.12 DATA

```
\_BITS DATA
```

UART1 data.

Definition at line 1640 of file STM8AF\_STM8S.h.

#### 6.22.2.13 DIV\_0\_3

`_BITS` DIV\_0\_3

UART\_DIV bits [3:0].

Definition at line 1652 of file STM8AF\_STM8S.h.

#### 6.22.2.14 DIV\_12\_15

`_BITS` DIV\_12\_15

UART\_DIV bits [15:12].

Definition at line 1653 of file STM8AF\_STM8S.h.

#### 6.22.2.15 DIV\_4\_11

`_BITS` DIV\_4\_11

UART\_DIV bits [11:4].

Definition at line 1646 of file STM8AF\_STM8S.h.

#### 6.22.2.16 DR

```
struct { ... } DR
```

UART1 data register (UART1\_DR)

#### 6.22.2.17 FE

`_BITS` FE

Framing error.

Definition at line 1628 of file STM8AF\_STM8S.h.

**6.22.2.18 GT**

`_BITS GT`

UART1 guard time.

Definition at line 1719 of file STM8AF\_STM8S.h.

**6.22.2.19 GTR**

```
struct { ... } GTR
```

UART1 guard time register (UART1\_GTR)

**6.22.2.20 HDSEL**

`_BITS HDSEL`

Half-Duplex Selection.

Definition at line 1710 of file STM8AF\_STM8S.h.

**6.22.2.21 IDLE**

`_BITS IDLE`

IDLE line detected.

Definition at line 1631 of file STM8AF\_STM8S.h.

**6.22.2.22 ILIEN**

`_BITS ILIEN`

IDLE Line interrupt enable.

Definition at line 1676 of file STM8AF\_STM8S.h.

#### 6.22.2.23 IREN

`_BITS IREN`

IrDA mode Enable.

Definition at line 1708 of file STM8AF\_STM8S.h.

#### 6.22.2.24 IRLP

`_BITS IRLP`

IrDA Low Power.

Definition at line 1709 of file STM8AF\_STM8S.h.

#### 6.22.2.25 LBCL

`_BITS LBCL`

Last bit clock pulse.

Definition at line 1685 of file STM8AF\_STM8S.h.

#### 6.22.2.26 LBDF

`_BITS LBDF`

LIN Break Detection Flag.

Definition at line 1698 of file STM8AF\_STM8S.h.

#### 6.22.2.27 LBDIEN

`_BITS LBDIEN`

LIN Break Detection Interrupt Enable.

Definition at line 1700 of file STM8AF\_STM8S.h.

**6.22.2.28 LBDL**

`_BITS` LBDL

LIN Break Detection Length.

Definition at line 1699 of file STM8AF\_STM8S.h.

**6.22.2.29 LINEN**

`_BITS` LINEN

LIN mode enable.

Definition at line 1690 of file STM8AF\_STM8S.h.

**6.22.2.30 M**

`_BITS` M

word length

Definition at line 1663 of file STM8AF\_STM8S.h.

**6.22.2.31 NACK**

`_BITS` NACK

Smartcard NACK enable.

Definition at line 1711 of file STM8AF\_STM8S.h.

**6.22.2.32 NF**

`_BITS` NF

Noise flag.

Definition at line 1629 of file STM8AF\_STM8S.h.



#### 6.22.2.33 OR

`_BITS` OR

LIN Header Error (LIN slave mode) / Overrun error.

Definition at line 1630 of file STM8AF\_STM8S.h.

#### 6.22.2.34 PCEN

`_BITS` PCEN

Parity control enable.

Definition at line 1661 of file STM8AF\_STM8S.h.

#### 6.22.2.35 PE

`_BITS` PE

Parity error.

Definition at line 1627 of file STM8AF\_STM8S.h.

#### 6.22.2.36 PIEN

`_BITS` PIEN

Parity interrupt enable.

Definition at line 1659 of file STM8AF\_STM8S.h.

#### 6.22.2.37 PS

`_BITS` PS

Parity selection.

Definition at line 1660 of file STM8AF\_STM8S.h.

**6.22.2.38 PSC**

`_BITS` PSC

UART1 prescaler.

Definition at line 1725 of file STM8AF\_STM8S.h.

**6.22.2.39 PSCR**

```
struct { ... } PSCR
```

UART1 prescaler register (UART1\_PSCR)

**6.22.2.40 R8**

`_BITS` R8

Receive Data bit 8 (in 9-bit mode)

Definition at line 1666 of file STM8AF\_STM8S.h.

**6.22.2.41 REN**

`_BITS` REN

Receiver enable.

Definition at line 1674 of file STM8AF\_STM8S.h.

**6.22.2.42 res**

`_BITS` res

Reserved, must be kept cleared.

Definition at line 1691 of file STM8AF\_STM8S.h.

**6.22.2.43 res2**

`_BITS res2`

Reserved, must be kept cleared.

Definition at line 1713 of file STM8AF\_STM8S.h.

**6.22.2.44 RIEN**

`_BITS RIEN`

Receiver interrupt enable.

Definition at line 1677 of file STM8AF\_STM8S.h.

**6.22.2.45 RWU**

`_BITS RWU`

Receiver wakeup.

Definition at line 1673 of file STM8AF\_STM8S.h.

**6.22.2.46 RXNE**

`_BITS RXNE`

Read data register not empty.

Definition at line 1632 of file STM8AF\_STM8S.h.

**6.22.2.47 SBK**

`_BITS SBK`

Send break.

Definition at line 1672 of file STM8AF\_STM8S.h.

**6.22.2.48 SCEN**

`_BITS` SCEN

Smartcard mode enable.

Definition at line 1712 of file STM8AF\_STM8S.h.

**6.22.2.49 SR**

```
struct { ... } SR
```

UART1 Status register (UART1\_SR)

**6.22.2.50 STOP**

`_BITS` STOP

STOP bits.

Definition at line 1689 of file STM8AF\_STM8S.h.

**6.22.2.51 T8**

`_BITS` T8

Transmit Data bit 8 (in 9-bit mode)

Definition at line 1665 of file STM8AF\_STM8S.h.

**6.22.2.52 TC**

`_BITS` TC

Transmission complete.

Definition at line 1633 of file STM8AF\_STM8S.h.

#### 6.22.2.53 TCEN

`_BITS TCEN`

Transmission complete interrupt enable.

Definition at line 1678 of file STM8AF\_STM8S.h.

#### 6.22.2.54 TEN

`_BITS TEN`

Transmitter enable.

Definition at line 1675 of file STM8AF\_STM8S.h.

#### 6.22.2.55 TIEN

`_BITS TIEN`

Transmitter interrupt enable.

Definition at line 1679 of file STM8AF\_STM8S.h.

#### 6.22.2.56 TXE

`_BITS TXE`

Transmit data register empty.

Definition at line 1634 of file STM8AF\_STM8S.h.

#### 6.22.2.57 UARTD

`_BITS UARTD`

UART Disable (for low power consumption)

Definition at line 1664 of file STM8AF\_STM8S.h.

## 6.22.2.58 WAKE

`_BITS WAKE`

Wakeup method.

Definition at line 1662 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.23 UART2\_t Struct Reference

struct for controlling Universal Asynchronous Receiver Transmitter 2 (UART2)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
    - `_BITS PE`: 1  
*Parity error.*
    - `_BITS FE`: 1  
*Framing error.*
    - `_BITS NF`: 1  
*Noise flag.*
    - `_BITS OR`: 1  
*LIN Header Error (LIN slave mode) / Overrun error.*
    - `_BITS IDLE`: 1  
*IDLE line detected.*
    - `_BITS RXNE`: 1  
*Read data register not empty.*
    - `_BITS TC`: 1  
*Transmission complete.*
    - `_BITS TXE`: 1  
*Transmit data register empty.*
- } `SR`

*UART2 Status register (UART2\_SR)*

- struct {
    - `_BITS DATA`: 8  
*UART2 data.*
- } `DR`

*UART2 data register (UART2\_DR)*

- struct {
    - `_BITS DIV_4_11`: 8  
*UART2\_BRR bits [11:4].*
- } `BRR1`

*UART2 Baud rate register 1 (UART2\_BRR1)*

- struct {
  - [\\_BITS DIV\\_0\\_3](#): 4  
*UART2\_BRR bits [3:0].*
  - [\\_BITS DIV\\_12\\_15](#): 4  
*UART2\_BRR bits [15:12].*
- } [BRR2](#)
  
- UART2 Baud rate register 2 (UART2\_BRR2)*
- struct {
  - [\\_BITS PIEN](#): 1  
*Parity interrupt enable.*
  - [\\_BITS PS](#): 1  
*Parity selection.*
  - [\\_BITS PCEN](#): 1  
*Parity control enable.*
  - [\\_BITS WAKE](#): 1  
*Wakeup method.*
  - [\\_BITS M](#): 1  
*word length*
  - [\\_BITS UARTD](#): 1  
*UART Disable (for low power consumption)*
  - [\\_BITS T8](#): 1  
*Transmit Data bit 8 (in 9-bit mode)*
  - [\\_BITS R8](#): 1  
*Receive Data bit 8 (in 9-bit mode)*
- } [CR1](#)
  
- UART2 Control register 1 (UART2\_CR1)*
- struct {
  - [\\_BITS SBK](#): 1  
*Send break.*
  - [\\_BITS RWU](#): 1  
*Receiver wakeup.*
  - [\\_BITS REN](#): 1  
*Receiver enable.*
  - [\\_BITS TEN](#): 1  
*Transmitter enable.*
  - [\\_BITS ILIEN](#): 1  
*IDLE Line interrupt enable.*
  - [\\_BITS RIEN](#): 1  
*Receiver interrupt enable.*
  - [\\_BITS TCEN](#): 1  
*Transmission complete interrupt enable.*
  - [\\_BITS TIEN](#): 1  
*Transmitter interrupt enable.*
- } [CR2](#)
  
- UART2 Control register 2 (UART2\_CR2)*
- struct {
  - [\\_BITS LBCL](#): 1  
*Last bit clock pulse.*
  - [\\_BITS CPHA](#): 1  
*Clock phase.*
  - [\\_BITS CPOL](#): 1  
*Clock polarity.*
  - [\\_BITS CKEN](#): 1  
*Clock enable.*
  - [\\_BITS STOP](#): 2

```

    STOP bits.
    _BITS LINEN: 1
        LIN mode enable.
    _BITS res: 1
        Reserved, must be kept cleared.
} CR3

```

```

    UART2 Control register 3 (UART2_CR3)
• struct {
    _BITS ADD: 4
        Address of the UART node.
    _BITS LBDF: 1
        LIN Break Detection Flag.
    _BITS LBDL: 1
        LIN Break Detection Length.
    _BITS LBDIEN: 1
        LIN Break Detection Interrupt Enable.
    _BITS res: 1
        Reserved, must be kept cleared.
} CR4

```

```

    UART2 Control register 4 (UART2_CR4)
• struct {
    _BITS res: 1
        Reserved, must be kept cleared.
    _BITS IREN: 1
        IrDA mode Enable.
    _BITS IRLP: 1
        IrDA Low Power.
    _BITS res2: 1
        Reserved, must be kept cleared.
    _BITS NACK: 1
        Smartcard NACK enable.
    _BITS SCEN: 1
        Smartcard mode enable.
    _BITS res3: 2
        Reserved, must be kept cleared.
} CR5

```

```

    UART2 Control register 5 (UART2_CR5)
• struct {
    _BITS LSF: 1
        LIN Sync Field.
    _BITS LHDF: 1
        LIN Header Detection Flag.
    _BITS LHDIEN: 1
        LIN Header Detection Interrupt Enable.
    _BITS res: 1
        Reserved.
    _BITS LASE: 1
        LIN automatic resynchronisation enable.
    _BITS LSLV: 1
        LIN Slave Enable.
    _BITS res2: 1
        Reserved.
    _BITS LDUM: 1
        LIN Divider Update Method.
} CR6

```



- UART2 Control register 6 (UART2\_CR6)*
  - struct {
    - `_BITS GT: 8`  
*UART2 guard time.*
  - } `GTR`
- UART2 guard time register (UART2\_GTR)*
  - struct {
    - `_BITS PSC: 8`  
*UART1 prescaler.*
  - } `PSCR`
- UART2 prescaler register (UART2\_PSCR)*

### 6.23.1 Detailed Description

struct for controlling Universal Asynchronous Receiver Transmitter 2 (UART2)

Definition at line 1827 of file STM8AF\_STM8S.h.

### 6.23.2 Field Documentation

#### 6.23.2.1 ADD

`_BITS ADD`

Address of the UART node.

Definition at line 1901 of file STM8AF\_STM8S.h.

#### 6.23.2.2 BRR1

```
struct { ... } BRR1
```

UART2 Baud rate register 1 (UART2\_BRR1)

#### 6.23.2.3 BRR2

```
struct { ... } BRR2
```

UART2 Baud rate register 2 (UART2\_BRR2)

#### 6.23.2.4 CKEN

`_BITS` CKEN

Clock enable.

Definition at line 1892 of file STM8AF\_STM8S.h.

#### 6.23.2.5 CPHA

`_BITS` CPHA

Clock phase.

Definition at line 1890 of file STM8AF\_STM8S.h.

#### 6.23.2.6 CPOL

`_BITS` CPOL

Clock polarity.

Definition at line 1891 of file STM8AF\_STM8S.h.

#### 6.23.2.7 CR1

```
struct { ... } CR1
```

UART2 Control register 1 (UART2\_CR1)

#### 6.23.2.8 CR2

```
struct { ... } CR2
```

UART2 Control register 2 (UART2\_CR2)

#### 6.23.2.9 CR3

```
struct { ... } CR3
```

UART2 Control register 3 (UART2\_CR3)

#### 6.23.2.10 CR4

```
struct { ... } CR4
```

UART2 Control register 4 (UART2\_CR4)

#### 6.23.2.11 CR5

```
struct { ... } CR5
```

UART2 Control register 5 (UART2\_CR5)

#### 6.23.2.12 CR6

```
struct { ... } CR6
```

UART2 Control register 6 (UART2\_CR6)

#### 6.23.2.13 DATA

```
_BITS DATA
```

UART2 data.

Definition at line 1844 of file STM8AF\_STM8S.h.

#### 6.23.2.14 DIV\_0\_3

```
_BITS DIV_0_3
```

UART2\_BRR bits [3:0].

Definition at line 1856 of file STM8AF\_STM8S.h.

#### 6.23.2.15 DIV\_12\_15

`_BITS` DIV\_12\_15

UART2\_BRR bits [15:12].

Definition at line 1857 of file STM8AF\_STM8S.h.

#### 6.23.2.16 DIV\_4\_11

`_BITS` DIV\_4\_11

UART2\_BRR bits [11:4].

Definition at line 1850 of file STM8AF\_STM8S.h.

#### 6.23.2.17 DR

```
struct { ... } DR
```

UART2 data register (UART2\_DR)

#### 6.23.2.18 FE

`_BITS` FE

Framing error.

Definition at line 1832 of file STM8AF\_STM8S.h.

#### 6.23.2.19 GT

`_BITS` GT

UART2 guard time.

Definition at line 1936 of file STM8AF\_STM8S.h.

#### 6.23.2.20 GTR

```
struct { ... } GTR
```

UART2 guard time register (UART2\_GTR)

#### 6.23.2.21 IDLE

```
_BITS IDLE
```

IDLE line detected.

Definition at line 1835 of file STM8AF\_STM8S.h.

#### 6.23.2.22 ILIEN

```
_BITS ILIEN
```

IDLE Line interrupt enable.

Definition at line 1880 of file STM8AF\_STM8S.h.

#### 6.23.2.23 IREN

```
_BITS IREN
```

IrDA mode Enable.

Definition at line 1912 of file STM8AF\_STM8S.h.

#### 6.23.2.24 IRLP

```
_BITS IRLP
```

IrDA Low Power.

Definition at line 1913 of file STM8AF\_STM8S.h.

**6.23.2.25 LASE**

`_BITS` LASE

LIN automatic resynchronisation enable.

Definition at line 1927 of file STM8AF\_STM8S.h.

**6.23.2.26 LBCL**

`_BITS` LBCL

Last bit clock pulse.

Definition at line 1889 of file STM8AF\_STM8S.h.

**6.23.2.27 LBDF**

`_BITS` LBDF

LIN Break Detection Flag.

Definition at line 1902 of file STM8AF\_STM8S.h.

**6.23.2.28 LBDIEN**

`_BITS` LBDIEN

LIN Break Detection Interrupt Enable.

Definition at line 1904 of file STM8AF\_STM8S.h.

**6.23.2.29 LBDL**

`_BITS` LBDL

LIN Break Detection Length.

Definition at line 1903 of file STM8AF\_STM8S.h.

#### 6.23.2.30 LDUM

`_BITS` LDUM

LIN Divider Update Method.

Definition at line 1930 of file STM8AF\_STM8S.h.

#### 6.23.2.31 LHDF

`_BITS` LHDF

LIN Header Detection Flag.

Definition at line 1924 of file STM8AF\_STM8S.h.

#### 6.23.2.32 LHDIE

`_BITS` LHDIE

LIN Header Detection Interrupt Enable.

Definition at line 1925 of file STM8AF\_STM8S.h.

#### 6.23.2.33 LINEN

`_BITS` LINEN

LIN mode enable.

Definition at line 1894 of file STM8AF\_STM8S.h.

#### 6.23.2.34 LSF

`_BITS` LSF

LIN Sync Field.

Definition at line 1923 of file STM8AF\_STM8S.h.

**6.23.2.35 LSLV**

`_BITS` LSLV

LIN Slave Enable.

Definition at line 1928 of file STM8AF\_STM8S.h.

**6.23.2.36 M**

`_BITS` M

word length

Definition at line 1867 of file STM8AF\_STM8S.h.

**6.23.2.37 NACK**

`_BITS` NACK

Smartcard NACK enable.

Definition at line 1915 of file STM8AF\_STM8S.h.

**6.23.2.38 NF**

`_BITS` NF

Noise flag.

Definition at line 1833 of file STM8AF\_STM8S.h.

**6.23.2.39 OR**

`_BITS` OR

LIN Header Error (LIN slave mode) / Overrun error.

Definition at line 1834 of file STM8AF\_STM8S.h.



#### 6.23.2.40 PCEN

`_BITS` PCEN

Parity control enable.

Definition at line 1865 of file STM8AF\_STM8S.h.

#### 6.23.2.41 PE

`_BITS` PE

Parity error.

Definition at line 1831 of file STM8AF\_STM8S.h.

#### 6.23.2.42 PIEN

`_BITS` PIEN

Parity interrupt enable.

Definition at line 1863 of file STM8AF\_STM8S.h.

#### 6.23.2.43 PS

`_BITS` PS

Parity selection.

Definition at line 1864 of file STM8AF\_STM8S.h.

#### 6.23.2.44 PSC

`_BITS` PSC

UART1 prescaler.

Definition at line 1942 of file STM8AF\_STM8S.h.

#### 6.23.2.45 PSCR

```
struct { ... } PSCR
```

UART2 prescaler register (UART2\_PSCR)

#### 6.23.2.46 R8

`_BITS` R8

Receive Data bit 8 (in 9-bit mode)

Definition at line 1870 of file STM8AF\_STM8S.h.

#### 6.23.2.47 REN

`_BITS` REN

Receiver enable.

Definition at line 1878 of file STM8AF\_STM8S.h.

#### 6.23.2.48 res

`_BITS` res

Reserved, must be kept cleared.

Reserved.

Definition at line 1895 of file STM8AF\_STM8S.h.

#### 6.23.2.49 res2

`_BITS` res2

Reserved, must be kept cleared.

Reserved.

Definition at line 1914 of file STM8AF\_STM8S.h.

**6.23.2.50 res3**

`_BITS res3`

Reserved, must be kept cleared.

Definition at line 1917 of file STM8AF\_STM8S.h.

**6.23.2.51 RIEN**

`_BITS RIEN`

Receiver interrupt enable.

Definition at line 1881 of file STM8AF\_STM8S.h.

**6.23.2.52 RWU**

`_BITS RWU`

Receiver wakeup.

Definition at line 1877 of file STM8AF\_STM8S.h.

**6.23.2.53 RXNE**

`_BITS RXNE`

Read data register not empty.

Definition at line 1836 of file STM8AF\_STM8S.h.

**6.23.2.54 SBK**

`_BITS SBK`

Send break.

Definition at line 1876 of file STM8AF\_STM8S.h.

**6.23.2.55 SCEN**

`_BITS` SCEN

Smartcard mode enable.

Definition at line 1916 of file STM8AF\_STM8S.h.

**6.23.2.56 SR**

```
struct { ... } SR
```

UART2 Status register (UART2\_SR)

**6.23.2.57 STOP**

`_BITS` STOP

STOP bits.

Definition at line 1893 of file STM8AF\_STM8S.h.

**6.23.2.58 T8**

`_BITS` T8

Transmit Data bit 8 (in 9-bit mode)

Definition at line 1869 of file STM8AF\_STM8S.h.

**6.23.2.59 TC**

`_BITS` TC

Transmission complete.

Definition at line 1837 of file STM8AF\_STM8S.h.

#### 6.23.2.60 TCEN

`_BITS TCEN`

Transmission complete interrupt enable.

Definition at line 1882 of file STM8AF\_STM8S.h.

#### 6.23.2.61 TEN

`_BITS TEN`

Transmitter enable.

Definition at line 1879 of file STM8AF\_STM8S.h.

#### 6.23.2.62 TIEN

`_BITS TIEN`

Transmitter interrupt enable.

Definition at line 1883 of file STM8AF\_STM8S.h.

#### 6.23.2.63 TXE

`_BITS TXE`

Transmit data register empty.

Definition at line 1838 of file STM8AF\_STM8S.h.

#### 6.23.2.64 UARTD

`_BITS UARTD`

UART Disable (for low power consumption)

Definition at line 1868 of file STM8AF\_STM8S.h.

## 6.23.2.65 WAKE

`_BITS WAKE`

Wakeup method.

Definition at line 1866 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.24 UART3\_t Struct Reference

struct for controlling Universal Asynchronous Receiver Transmitter 3 (UART3)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
    - `_BITS PE`: 1  
*Parity error.*
    - `_BITS FE`: 1  
*Framing error.*
    - `_BITS NF`: 1  
*Noise flag.*
    - `_BITS OR`: 1  
*LIN Header Error (LIN slave mode) / Overrun error.*
    - `_BITS IDLE`: 1  
*IDLE line detected.*
    - `_BITS RXNE`: 1  
*Read data register not empty.*
    - `_BITS TC`: 1  
*Transmission complete.*
    - `_BITS TXE`: 1  
*Transmit data register empty.*
- } `SR`

*UART3 Status register (UART3\_SR)*

- struct {
    - `_BITS DATA`: 8  
*UART3 data.*
- } `DR`

*UART3 data register (UART3\_DR)*

- struct {
    - `_BITS DIV_4_11`: 8  
*UART\_DIV bits [11:4].*
- } `BRR1`

*UART3 Baud rate register 1 (UART3\_BRR1)*

- struct {
  - [\\_BITS DIV\\_0\\_3](#): 4  
*UART\_DIV bits [3:0].*
  - [\\_BITS DIV\\_12\\_15](#): 4  
*UART\_DIV bits [15:12].*
- } [BRR2](#)
  
- UART3 Baud rate register 2 (UART3\_BRR2)*
- struct {
  - [\\_BITS PIEN](#): 1  
*Parity interrupt enable.*
  - [\\_BITS PS](#): 1  
*Parity selection.*
  - [\\_BITS PCEN](#): 1  
*Parity control enable.*
  - [\\_BITS WAKE](#): 1  
*Wakeup method.*
  - [\\_BITS M](#): 1  
*word length*
  - [\\_BITS UARTD](#): 1  
*UART Disable (for low power consumption)*
  - [\\_BITS T8](#): 1  
*Transmit Data bit 8 (in 9-bit mode)*
  - [\\_BITS R8](#): 1  
*Receive Data bit 8 (in 9-bit mode)*
- } [CR1](#)
  
- UART3 Control register 1 (UART3\_CR1)*
- struct {
  - [\\_BITS SBK](#): 1  
*Send break.*
  - [\\_BITS RWU](#): 1  
*Receiver wakeup.*
  - [\\_BITS REN](#): 1  
*Receiver enable.*
  - [\\_BITS TEN](#): 1  
*Transmitter enable.*
  - [\\_BITS ILIEN](#): 1  
*IDLE Line interrupt enable.*
  - [\\_BITS RIEN](#): 1  
*Receiver interrupt enable.*
  - [\\_BITS TCEN](#): 1  
*Transmission complete interrupt enable.*
  - [\\_BITS TIEN](#): 1  
*Transmitter interrupt enable.*
- } [CR2](#)
  
- UART3 Control register 2 (UART3\_CR2)*
- struct {
  - [\\_BITS res](#): 4  
*Reserved, must be kept cleared.*
  - [\\_BITS STOP](#): 2  
*STOP bits.*
  - [\\_BITS LINEN](#): 1  
*LIN mode enable.*
  - [\\_BITS res2](#): 1  
*Reserved, must be kept cleared.*
- } [CR3](#)

UART3 Control register 3 (UART3\_CR3)

- struct {
    - [\\_BITS ADD](#): 4  
*Address of the UART node.*
    - [\\_BITS LBDF](#): 1  
*LIN Break Detection Flag.*
    - [\\_BITS LBDL](#): 1  
*LIN Break Detection Length.*
    - [\\_BITS LBDIEN](#): 1  
*LIN Break Detection Interrupt Enable.*
    - [\\_BITS res](#): 1  
*Reserved, must be kept cleared.*
- } [CR4](#)

UART3 Control register 4 (UART3\_CR4)

- uint8\_t [res](#) [1]  
*Reserved register (1B)*
  - struct {
    - [\\_BITS LSF](#): 1  
*LIN Sync Field.*
    - [\\_BITS LHDF](#): 1  
*LIN Header Detection Flag.*
    - [\\_BITS LHD IEN](#): 1  
*LIN Header Detection Interrupt Enable.*
    - [\\_BITS res](#): 1  
*Reserved.*
    - [\\_BITS LASE](#): 1  
*LIN automatic resynchronisation enable.*
    - [\\_BITS LSLV](#): 1  
*LIN Slave Enable.*
    - [\\_BITS res2](#): 1  
*Reserved.*
    - [\\_BITS LDUM](#): 1  
*LIN Divider Update Method.*
- } [CR6](#)

UART3 Control register 6 (UART3\_CR6)

### 6.24.1 Detailed Description

struct for controlling Universal Asynchronous Receiver Transmitter 3 (UART3)

Definition at line 2056 of file STM8AF\_STM8S.h.

### 6.24.2 Field Documentation



#### 6.24.2.1 ADD

`_BITS` ADD

Address of the UART node.

Definition at line 2127 of file STM8AF\_STM8S.h.

#### 6.24.2.2 BRR1

```
struct { ... } BRR1
```

UART3 Baud rate register 1 (UART3\_BRR1)

#### 6.24.2.3 BRR2

```
struct { ... } BRR2
```

UART3 Baud rate register 2 (UART3\_BRR2)

#### 6.24.2.4 CR1

```
struct { ... } CR1
```

UART3 Control register 1 (UART3\_CR1)

#### 6.24.2.5 CR2

```
struct { ... } CR2
```

UART3 Control register 2 (UART3\_CR2)

#### 6.24.2.6 CR3

```
struct { ... } CR3
```

UART3 Control register 3 (UART3\_CR3)

#### 6.24.2.7 CR4

```
struct { ... } CR4
```

UART3 Control register 4 (UART3\_CR4)

#### 6.24.2.8 CR6

```
struct { ... } CR6
```

UART3 Control register 6 (UART3\_CR6)

#### 6.24.2.9 DATA

```
_BITS DATA
```

UART3 data.

Definition at line 2073 of file STM8AF\_STM8S.h.

#### 6.24.2.10 DIV\_0\_3

```
_BITS DIV_0_3
```

UART\_DIV bits [3:0].

Definition at line 2085 of file STM8AF\_STM8S.h.

#### 6.24.2.11 DIV\_12\_15

```
_BITS DIV_12_15
```

UART\_DIV bits [15:12].

Definition at line 2086 of file STM8AF\_STM8S.h.

#### 6.24.2.12 DIV\_4\_11

`_BITS DIV_4_11`

UART\_DIV bits [11:4].

Definition at line 2079 of file STM8AF\_STM8S.h.

#### 6.24.2.13 DR

```
struct { ... } DR
```

UART3 data register (UART3\_DR)

#### 6.24.2.14 FE

`_BITS FE`

Framing error.

Definition at line 2061 of file STM8AF\_STM8S.h.

#### 6.24.2.15 IDLE

`_BITS IDLE`

IDLE line detected.

Definition at line 2064 of file STM8AF\_STM8S.h.

#### 6.24.2.16 ILIEN

`_BITS ILIEN`

IDLE Line interrupt enable.

Definition at line 2109 of file STM8AF\_STM8S.h.

#### 6.24.2.17 LASE

`_BITS` LASE

LIN automatic resynchronisation enable.

Definition at line 2145 of file STM8AF\_STM8S.h.

#### 6.24.2.18 LBDF

`_BITS` LBDF

LIN Break Detection Flag.

Definition at line 2128 of file STM8AF\_STM8S.h.

#### 6.24.2.19 LBDIEN

`_BITS` LBDIEN

LIN Break Detection Interrupt Enable.

Definition at line 2130 of file STM8AF\_STM8S.h.

#### 6.24.2.20 LBDL

`_BITS` LBDL

LIN Break Detection Length.

Definition at line 2129 of file STM8AF\_STM8S.h.

#### 6.24.2.21 LDUM

`_BITS` LDUM

LIN Divider Update Method.

Definition at line 2148 of file STM8AF\_STM8S.h.

#### 6.24.2.22 LHDF

`_BITS LHDF`

LIN Header Detection Flag.

Definition at line 2142 of file STM8AF\_STM8S.h.

#### 6.24.2.23 LHDIE

`_BITS LHDIE`

LIN Header Detection Interrupt Enable.

Definition at line 2143 of file STM8AF\_STM8S.h.

#### 6.24.2.24 LINEN

`_BITS LINEN`

LIN mode enable.

Definition at line 2120 of file STM8AF\_STM8S.h.

#### 6.24.2.25 LSF

`_BITS LSF`

LIN Sync Field.

Definition at line 2141 of file STM8AF\_STM8S.h.

#### 6.24.2.26 LSLV

`_BITS LSLV`

LIN Slave Enable.

Definition at line 2146 of file STM8AF\_STM8S.h.

**6.24.2.27 M**

`_BITS` M

word length

Definition at line 2096 of file STM8AF\_STM8S.h.

**6.24.2.28 NF**

`_BITS` NF

Noise flag.

Definition at line 2062 of file STM8AF\_STM8S.h.

**6.24.2.29 OR**

`_BITS` OR

LIN Header Error (LIN slave mode) / Overrun error.

Definition at line 2063 of file STM8AF\_STM8S.h.

**6.24.2.30 PCEN**

`_BITS` PCEN

Parity control enable.

Definition at line 2094 of file STM8AF\_STM8S.h.

**6.24.2.31 PE**

`_BITS` PE

Parity error.

Definition at line 2060 of file STM8AF\_STM8S.h.

#### 6.24.2.32 PIEN

`_BITS` PIEN

Parity interrupt enable.

Definition at line 2092 of file STM8AF\_STM8S.h.

#### 6.24.2.33 PS

`_BITS` PS

Parity selection.

Definition at line 2093 of file STM8AF\_STM8S.h.

#### 6.24.2.34 R8

`_BITS` R8

Receive Data bit 8 (in 9-bit mode)

Definition at line 2099 of file STM8AF\_STM8S.h.

#### 6.24.2.35 REN

`_BITS` REN

Receiver enable.

Definition at line 2107 of file STM8AF\_STM8S.h.

#### 6.24.2.36 res [1/2]

`_BITS` res

Reserved, must be kept cleared.

Reserved.

Definition at line 2118 of file STM8AF\_STM8S.h.

**6.24.2.37** **res** [2/2]

```
uint8_t res[1]
```

Reserved register (1B)

Definition at line 2136 of file STM8AF\_STM8S.h.

**6.24.2.38** **res2**

```
_BITS res2
```

Reserved, must be kept cleared.

Reserved.

Definition at line 2121 of file STM8AF\_STM8S.h.

**6.24.2.39** **RIEN**

```
_BITS RIEN
```

Receiver interrupt enable.

Definition at line 2110 of file STM8AF\_STM8S.h.

**6.24.2.40** **RWU**

```
_BITS RWU
```

Receiver wakeup.

Definition at line 2106 of file STM8AF\_STM8S.h.

**6.24.2.41** **RXNE**

```
_BITS RXNE
```

Read data register not empty.

Definition at line 2065 of file STM8AF\_STM8S.h.



#### 6.24.2.42 SBK

`_BITS` SBK

Send break.

Definition at line 2105 of file STM8AF\_STM8S.h.

#### 6.24.2.43 SR

```
struct { ... } SR
```

UART3 Status register (UART3\_SR)

#### 6.24.2.44 STOP

`_BITS` STOP

STOP bits.

Definition at line 2119 of file STM8AF\_STM8S.h.

#### 6.24.2.45 T8

`_BITS` T8

Transmit Data bit 8 (in 9-bit mode)

Definition at line 2098 of file STM8AF\_STM8S.h.

#### 6.24.2.46 TC

`_BITS` TC

Transmission complete.

Definition at line 2066 of file STM8AF\_STM8S.h.

**6.24.2.47 TCEN**

`_BITS TCEN`

Transmission complete interrupt enable.

Definition at line 2111 of file STM8AF\_STM8S.h.

**6.24.2.48 TEN**

`_BITS TEN`

Transmitter enable.

Definition at line 2108 of file STM8AF\_STM8S.h.

**6.24.2.49 TIEN**

`_BITS TIEN`

Transmitter interrupt enable.

Definition at line 2112 of file STM8AF\_STM8S.h.

**6.24.2.50 TXE**

`_BITS TXE`

Transmit data register empty.

Definition at line 2067 of file STM8AF\_STM8S.h.

**6.24.2.51 UARTD**

`_BITS UARTD`

UART Disable (for low power consumption)

Definition at line 2097 of file STM8AF\_STM8S.h.

## 6.24.2.52 WAKE

`_BITS WAKE`

Wakeup method.

Definition at line 2095 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.25 UART4\_t Struct Reference

struct for controlling Universal Asynchronous Receiver Transmitter 4 (UART4)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
    - `_BITS PE`: 1  
*Parity error.*
    - `_BITS FE`: 1  
*Framing error.*
    - `_BITS NF`: 1  
*Noise flag.*
    - `_BITS OR`: 1  
*LIN Header Error (LIN slave mode) / Overrun error.*
    - `_BITS IDLE`: 1  
*IDLE line detected.*
    - `_BITS RXNE`: 1  
*Read data register not empty.*
    - `_BITS TC`: 1  
*Transmission complete.*
    - `_BITS TXE`: 1  
*Transmit data register empty.*
- } `SR`

*UART4 Status register (UART4\_SR)*

- struct {
    - `_BITS DATA`: 8  
*UART4 data.*
- } `DR`

*UART4 data register (UART4\_DR)*

- struct {
    - `_BITS DIV_4_11`: 8  
*UART\_DIV bits [11:4].*
- } `BRR1`

*UART4 Baud rate register 1 (UART4\_BRR1)*

- struct {
  - [\\_BITS DIV\\_0\\_3](#): 4  
*UART\_DIV bits [3:0].*
  - [\\_BITS DIV\\_12\\_15](#): 4  
*UART\_DIV bits [15:12].*
- } [BRR2](#)
  
- UART4 Baud rate register 2 (UART4\_BRR2)*
- struct {
  - [\\_BITS PIEN](#): 1  
*Parity interrupt enable.*
  - [\\_BITS PS](#): 1  
*Parity selection.*
  - [\\_BITS PCEN](#): 1  
*Parity control enable.*
  - [\\_BITS WAKE](#): 1  
*Wakeup method.*
  - [\\_BITS M](#): 1  
*word length*
  - [\\_BITS UARTD](#): 1  
*UART Disable (for low power consumption)*
  - [\\_BITS T8](#): 1  
*Transmit Data bit 8 (in 9-bit mode)*
  - [\\_BITS R8](#): 1  
*Receive Data bit 8 (in 9-bit mode)*
- } [CR1](#)
  
- UART4 Control register 1 (UART4\_CR1)*
- struct {
  - [\\_BITS SBK](#): 1  
*Send break.*
  - [\\_BITS RWU](#): 1  
*Receiver wakeup.*
  - [\\_BITS REN](#): 1  
*Receiver enable.*
  - [\\_BITS TEN](#): 1  
*Transmitter enable.*
  - [\\_BITS ILIEN](#): 1  
*IDLE Line interrupt enable.*
  - [\\_BITS RIEN](#): 1  
*Receiver interrupt enable.*
  - [\\_BITS TCIEEN](#): 1  
*Transmission complete interrupt enable.*
  - [\\_BITS TIEN](#): 1  
*Transmitter interrupt enable.*
- } [CR2](#)
  
- UART4 Control register 2 (UART4\_CR2)*
- struct {
  - [\\_BITS LBCL](#): 1  
*Last bit clock pulse.*
  - [\\_BITS CPHA](#): 1  
*Clock phase.*
  - [\\_BITS CPOL](#): 1  
*Clock polarity.*
  - [\\_BITS CKEN](#): 1  
*Clock enable.*
  - [\\_BITS STOP](#): 2

*STOP bits.*  
 \_BITS LINEN: 1  
*LIN mode enable.*  
 \_BITS res: 1  
*Reserved, must be kept cleared.*  
 } CR3

UART4 Control register 3 (UART4\_CR3)

- struct {
 \_BITS ADD: 4  
*Address of the UART node.*  
 \_BITS LBDF: 1  
*LIN Break Detection Flag.*  
 \_BITS LBDL: 1  
*LIN Break Detection Length.*  
 \_BITS LBDIEN: 1  
*LIN Break Detection Interrupt Enable.*  
 \_BITS res: 1  
*Reserved, must be kept cleared.*  
 } CR4

UART4 Control register 4 (UART4\_CR4)

- struct {
 \_BITS res: 1  
*Reserved, must be kept cleared.*  
 \_BITS IREN: 1  
*IrDA mode Enable.*  
 \_BITS IRLP: 1  
*IrDA Low Power.*  
 \_BITS HDSEL: 1  
*Half-Duplex Selection.*  
 \_BITS NACK: 1  
*Smartcard NACK enable.*  
 \_BITS SCEN: 1  
*Smartcard mode enable.*  
 \_BITS res2: 2  
*Reserved, must be kept cleared.*  
 } CR5

UART4 Control register 5 (UART4\_CR5)

- struct {
 \_BITS LSF: 1  
*LIN Sync Field.*  
 \_BITS LHDF: 1  
*LIN Header Detection Flag.*  
 \_BITS LHDIEN: 1  
*LIN Header Detection Interrupt Enable.*  
 \_BITS res: 1  
*Reserved.*  
 \_BITS LASE: 1  
*LIN automatic resynchronisation enable.*  
 \_BITS LSLV: 1  
*LIN Slave Enable.*  
 \_BITS res2: 1  
*Reserved.*  
 \_BITS LDUM: 1  
*LIN Divider Update Method.*  
 } CR6

```

    UART4 Control register 6 (UART4_CR6)
    • struct {
        _BITS GT: 8
        UART4 guard time.
    } GTR

    UART4 guard time register (UART4_GTR)
    • struct {
        _BITS PSC: 8
        UART4 prescaler.
    } PSCR

    UART4 prescaler register (UART4_PSCR)

```

### 6.25.1 Detailed Description

struct for controlling Universal Asynchronous Receiver Transmitter 4 (UART4)

Definition at line 2245 of file STM8AF\_STM8S.h.

### 6.25.2 Field Documentation

#### 6.25.2.1 ADD

```
_BITS ADD
```

Address of the UART node.

Definition at line 2319 of file STM8AF\_STM8S.h.

#### 6.25.2.2 BRR1

```
struct { ... } BRR1
```

UART4 Baud rate register 1 (UART4\_BRR1)

#### 6.25.2.3 BRR2

```
struct { ... } BRR2
```

UART4 Baud rate register 2 (UART4\_BRR2)

#### 6.25.2.4 CKEN

`_BITS` CKEN

Clock enable.

Definition at line 2310 of file STM8AF\_STM8S.h.

#### 6.25.2.5 CPHA

`_BITS` CPHA

Clock phase.

Definition at line 2308 of file STM8AF\_STM8S.h.

#### 6.25.2.6 CPOL

`_BITS` CPOL

Clock polarity.

Definition at line 2309 of file STM8AF\_STM8S.h.

#### 6.25.2.7 CR1

```
struct { ... } CR1
```

UART4 Control register 1 (UART4\_CR1)

#### 6.25.2.8 CR2

```
struct { ... } CR2
```

UART4 Control register 2 (UART4\_CR2)

#### 6.25.2.9 CR3

```
struct { ... } CR3
```

UART4 Control register 3 (UART4\_CR3)

#### 6.25.2.10 CR4

```
struct { ... } CR4
```

UART4 Control register 4 (UART4\_CR4)

#### 6.25.2.11 CR5

```
struct { ... } CR5
```

UART4 Control register 5 (UART4\_CR5)

#### 6.25.2.12 CR6

```
struct { ... } CR6
```

UART4 Control register 6 (UART4\_CR6)

#### 6.25.2.13 DATA

```
_BITS DATA
```

UART4 data.

Definition at line 2262 of file STM8AF\_STM8S.h.

#### 6.25.2.14 DIV\_0\_3

```
_BITS DIV_0_3
```

UART\_DIV bits [3:0].

Definition at line 2274 of file STM8AF\_STM8S.h.



#### 6.25.2.15 DIV\_12\_15

`_BITS` DIV\_12\_15

UART\_DIV bits [15:12].

Definition at line 2275 of file STM8AF\_STM8S.h.

#### 6.25.2.16 DIV\_4\_11

`_BITS` DIV\_4\_11

UART\_DIV bits [11:4].

Definition at line 2268 of file STM8AF\_STM8S.h.

#### 6.25.2.17 DR

```
struct { ... } DR
```

UART4 data register (UART4\_DR)

#### 6.25.2.18 FE

`_BITS` FE

Framing error.

Definition at line 2250 of file STM8AF\_STM8S.h.

#### 6.25.2.19 GT

`_BITS` GT

UART4 guard time.

Definition at line 2354 of file STM8AF\_STM8S.h.

#### 6.25.2.20 GTR

```
struct { ... } GTR
```

UART4 guard time register (UART4\_GTR)

#### 6.25.2.21 HDSEL

```
_BITS HDSEL
```

Half-Duplex Selection.

Definition at line 2332 of file STM8AF\_STM8S.h.

#### 6.25.2.22 IDLE

```
_BITS IDLE
```

IDLE line detected.

Definition at line 2253 of file STM8AF\_STM8S.h.

#### 6.25.2.23 ILIEN

```
_BITS ILIEN
```

IDLE Line interrupt enable.

Definition at line 2298 of file STM8AF\_STM8S.h.

#### 6.25.2.24 IREN

```
_BITS IREN
```

IrDA mode Enable.

Definition at line 2330 of file STM8AF\_STM8S.h.

#### 6.25.2.25 IRLP

`_BITS` IRLP

IrDA Low Power.

Definition at line 2331 of file STM8AF\_STM8S.h.

#### 6.25.2.26 LASE

`_BITS` LASE

LIN automatic resynchronisation enable.

Definition at line 2345 of file STM8AF\_STM8S.h.

#### 6.25.2.27 LBCL

`_BITS` LBCL

Last bit clock pulse.

Definition at line 2307 of file STM8AF\_STM8S.h.

#### 6.25.2.28 LBDF

`_BITS` LBDF

LIN Break Detection Flag.

Definition at line 2320 of file STM8AF\_STM8S.h.

#### 6.25.2.29 LBDIEN

`_BITS` LBDIEN

LIN Break Detection Interrupt Enable.

Definition at line 2322 of file STM8AF\_STM8S.h.

**6.25.2.30 LBDL**

`_BITS` LBDL

LIN Break Detection Length.

Definition at line 2321 of file STM8AF\_STM8S.h.

**6.25.2.31 LDUM**

`_BITS` LDUM

LIN Divider Update Method.

Definition at line 2348 of file STM8AF\_STM8S.h.

**6.25.2.32 LHDF**

`_BITS` LHDF

LIN Header Detection Flag.

Definition at line 2342 of file STM8AF\_STM8S.h.

**6.25.2.33 LHDLEN**

`_BITS` LHDLEN

LIN Header Detection Interrupt Enable.

Definition at line 2343 of file STM8AF\_STM8S.h.

**6.25.2.34 LINEN**

`_BITS` LINEN

LIN mode enable.

Definition at line 2312 of file STM8AF\_STM8S.h.

#### 6.25.2.35 LSF

`_BITS` LSF

LIN Sync Field.

Definition at line 2341 of file STM8AF\_STM8S.h.

#### 6.25.2.36 LSLV

`_BITS` LSLV

LIN Slave Enable.

Definition at line 2346 of file STM8AF\_STM8S.h.

#### 6.25.2.37 M

`_BITS` M

word length

Definition at line 2285 of file STM8AF\_STM8S.h.

#### 6.25.2.38 NACK

`_BITS` NACK

Smartcard NACK enable.

Definition at line 2333 of file STM8AF\_STM8S.h.

#### 6.25.2.39 NF

`_BITS` NF

Noise flag.

Definition at line 2251 of file STM8AF\_STM8S.h.

**6.25.2.40 OR**

`_BITS` OR

LIN Header Error (LIN slave mode) / Overrun error.

Definition at line 2252 of file STM8AF\_STM8S.h.

**6.25.2.41 PCEN**

`_BITS` PCEN

Parity control enable.

Definition at line 2283 of file STM8AF\_STM8S.h.

**6.25.2.42 PE**

`_BITS` PE

Parity error.

Definition at line 2249 of file STM8AF\_STM8S.h.

**6.25.2.43 PIEN**

`_BITS` PIEN

Parity interrupt enable.

Definition at line 2281 of file STM8AF\_STM8S.h.

**6.25.2.44 PS**

`_BITS` PS

Parity selection.

Definition at line 2282 of file STM8AF\_STM8S.h.

#### 6.25.2.45 PSC

`_BITS` PSC

UART4 prescaler.

Definition at line 2360 of file STM8AF\_STM8S.h.

#### 6.25.2.46 PSCR

```
struct { ... } PSCR
```

UART4 prescaler register (UART4\_PSCR)

#### 6.25.2.47 R8

`_BITS` R8

Receive Data bit 8 (in 9-bit mode)

Definition at line 2288 of file STM8AF\_STM8S.h.

#### 6.25.2.48 REN

`_BITS` REN

Receiver enable.

Definition at line 2296 of file STM8AF\_STM8S.h.

#### 6.25.2.49 res

`_BITS` res

Reserved, must be kept cleared.

Reserved.

Definition at line 2313 of file STM8AF\_STM8S.h.

**6.25.2.50 res2**

`_BITS res2`

Reserved, must be kept cleared.

Reserved.

Definition at line 2335 of file STM8AF\_STM8S.h.

**6.25.2.51 RIEN**

`_BITS RIEN`

Receiver interrupt enable.

Definition at line 2299 of file STM8AF\_STM8S.h.

**6.25.2.52 RWU**

`_BITS RWU`

Receiver wakeup.

Definition at line 2295 of file STM8AF\_STM8S.h.

**6.25.2.53 RXNE**

`_BITS RXNE`

Read data register not empty.

Definition at line 2254 of file STM8AF\_STM8S.h.

**6.25.2.54 SBK**

`_BITS SBK`

Send break.

Definition at line 2294 of file STM8AF\_STM8S.h.



#### 6.25.2.55 SCEN

`_BITS` SCEN

Smartcard mode enable.

Definition at line 2334 of file STM8AF\_STM8S.h.

#### 6.25.2.56 SR

```
struct { ... } SR
```

UART4 Status register (UART4\_SR)

#### 6.25.2.57 STOP

`_BITS` STOP

STOP bits.

Definition at line 2311 of file STM8AF\_STM8S.h.

#### 6.25.2.58 T8

`_BITS` T8

Transmit Data bit 8 (in 9-bit mode)

Definition at line 2287 of file STM8AF\_STM8S.h.

#### 6.25.2.59 TC

`_BITS` TC

Transmission complete.

Definition at line 2255 of file STM8AF\_STM8S.h.

**6.25.2.60 TCEN**

`_BITS TCEN`

Transmission complete interrupt enable.

Definition at line 2300 of file STM8AF\_STM8S.h.

**6.25.2.61 TEN**

`_BITS TEN`

Transmitter enable.

Definition at line 2297 of file STM8AF\_STM8S.h.

**6.25.2.62 TIEN**

`_BITS TIEN`

Transmitter interrupt enable.

Definition at line 2301 of file STM8AF\_STM8S.h.

**6.25.2.63 TXE**

`_BITS TXE`

Transmit data register empty.

Definition at line 2256 of file STM8AF\_STM8S.h.

**6.25.2.64 UARTD**

`_BITS UARTD`

UART Disable (for low power consumption)

Definition at line 2286 of file STM8AF\_STM8S.h.

## 6.25.2.65 WAKE

`_BITS WAKE`

Wakeup method.

Definition at line 2284 of file STM8AF\_STM8S.h.

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## 6.26 WWDG\_t Struct Reference

struct for access to Window Watchdog registers (WWDG)

```
#include <STM8AF_STM8S.h>
```

## Data Fields

- struct {
  - `_BITS T: 7`  
*7-bit WWDG counter*
  - `_BITS WDGA: 1`  
*WWDG activation bit (n/a if WWDG enabled by option byte)*

`} CR`

*WWDG Control register (WWDG\_CR)*

- struct {
  - `_BITS W: 7`  
*7-bit window value*
  - `_BITS res: 1`  
*Reserved.*

`} WR`

*WWDG Window register (WWDG\_WR)*

## 6.26.1 Detailed Description

struct for access to Window Watchdog registers (WWDG)

Definition at line 988 of file STM8AF\_STM8S.h.

## 6.26.2 Field Documentation

#### 6.26.2.1 CR

```
struct { ... } CR
```

WWDG Control register (WWDG\_CR)

#### 6.26.2.2 res

```
_BITS res
```

Reserved.

Definition at line 1000 of file STM8AF\_STM8S.h.

#### 6.26.2.3 T

```
_BITS T
```

7-bit WWDG counter

Definition at line 992 of file STM8AF\_STM8S.h.

#### 6.26.2.4 W

```
_BITS W
```

7-bit window value

Definition at line 999 of file STM8AF\_STM8S.h.

#### 6.26.2.5 WDGA

```
_BITS WDGA
```

WWDG activation bit (n/a if WWDG enabled by option byte)

Definition at line 993 of file STM8AF\_STM8S.h.

#### 6.26.2.6 WR

```
struct { ... } WR
```

WWDG Window register (WWDG\_WR)

The documentation for this struct was generated from the following file:

- [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF\\_STM8S.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF_STM8S.h)

## Chapter 7

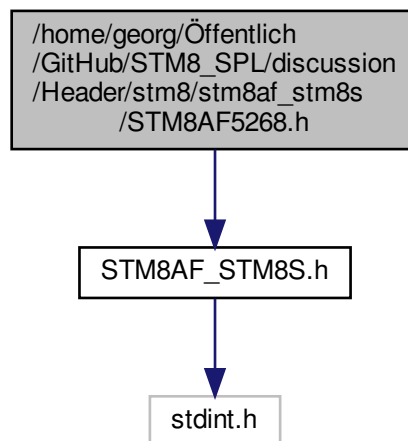
# File Documentation

### 7.1 pages/mainpage.txt File Reference

### 7.2 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF5268.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF5268.h:



### Macros

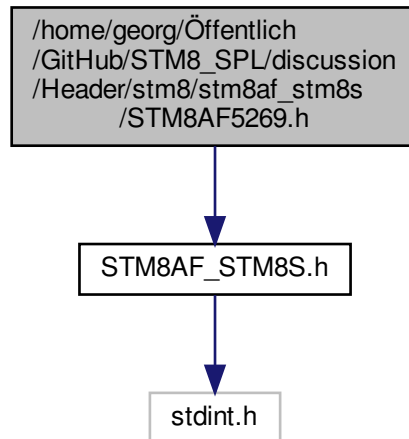
- #define STM8AF5268
- #define STM8AF526x
- #define STM8\_PFLASH\_SIZE 32\*1024

- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

### 7.3 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF5269.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5269.h) File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF5269.h:



## Macros

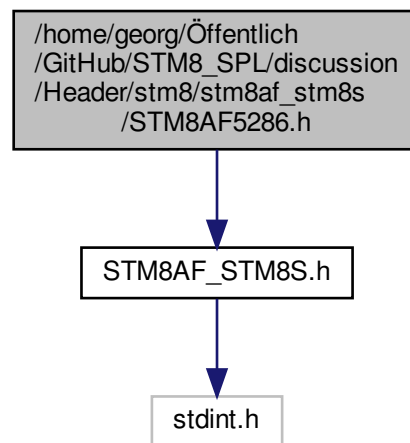
- #define STM8AF5269
- #define STM8AF526x
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1024
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.4 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF5286.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF5286.h:



### Macros

- `#define STM8AF5286`
- `#define STM8AF528x`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

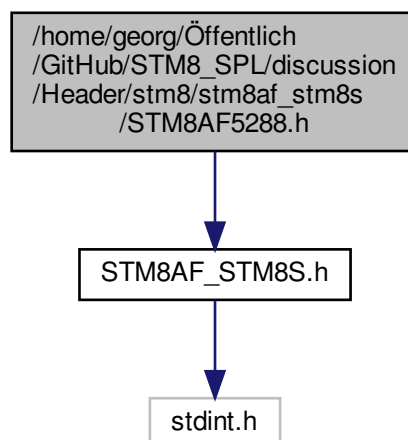


- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.5 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF5288.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF5288.h:



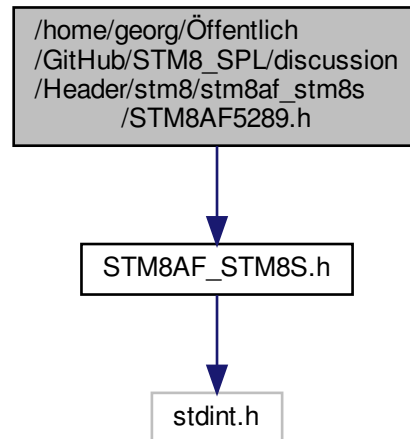
## Macros

- #define STM8AF5288
- #define STM8AF528x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

## 7.6 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF5289.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF5289.h) File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF5289.h:



## Macros

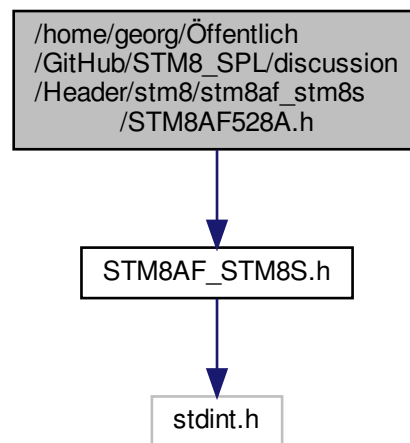
- `#define STM8AF5289`
- `#define STM8AF528x`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.7 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF528A.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF528A.h:



### Macros

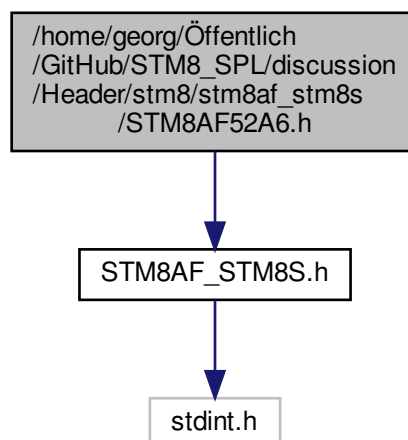
- `#define STM8AF528A`
- `#define STM8AF528x`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.8 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF52A6.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF52A6.h:



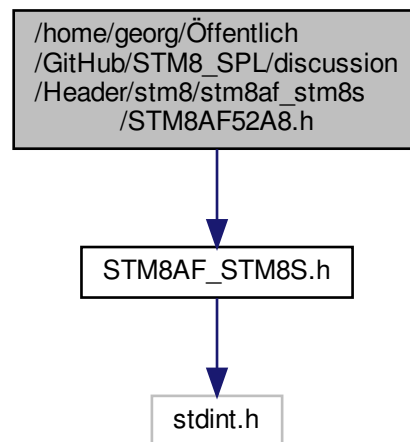
## Macros

- #define STM8AF52A6
- #define STM8AF52Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

## 7.9 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF52A8.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF52A8.h) File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF52A8.h:



## Macros

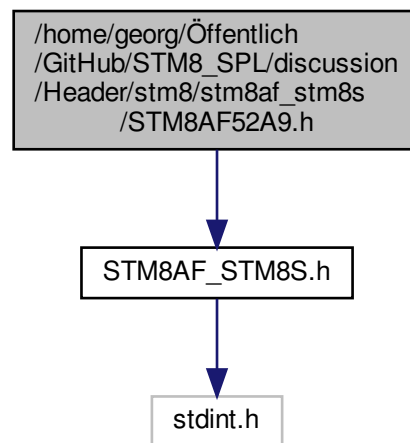
- #define [STM8AF52A8](#)
- #define [STM8AF52Ax](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 128\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.10 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF52A9.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF52A9.h:



### Macros

- `#define STM8AF52A9`
- `#define STM8AF52Ax`
- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

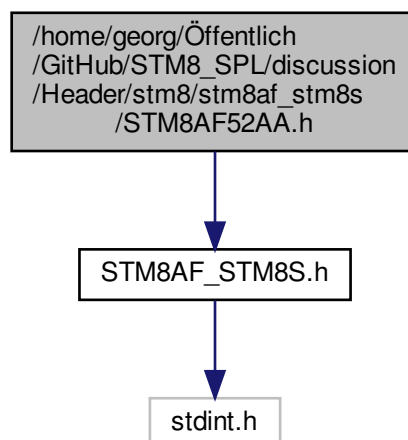


- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.11 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF52AA.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF52AA.h:



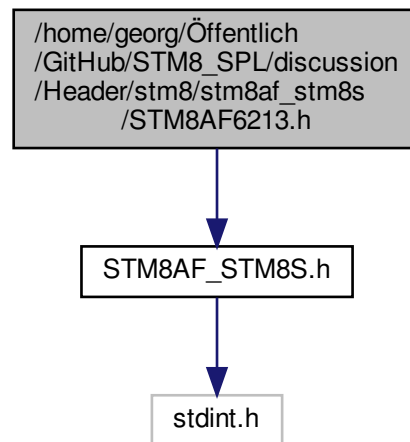
## Macros

- #define STM8AF52AA
- #define STM8AF52Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

## 7.12 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6213.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6213.h:



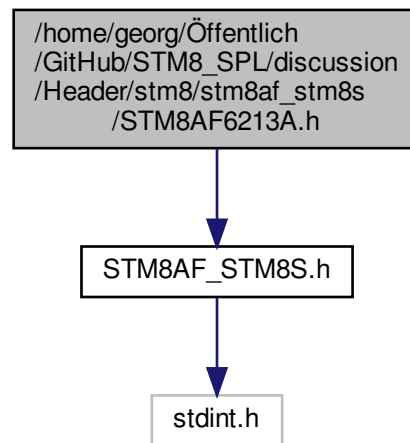
## Macros

- #define [STM8AF6213](#)
- #define [STM8AF621x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 4\*1024
- #define [STM8\\_RAM\\_SIZE](#) 1\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 640
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART4\\_AddressBase](#) 0x5230
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM5\\_AddressBase](#) 0x5300
- #define [TIM6\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.13 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6213A.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6213A.h:



### Macros

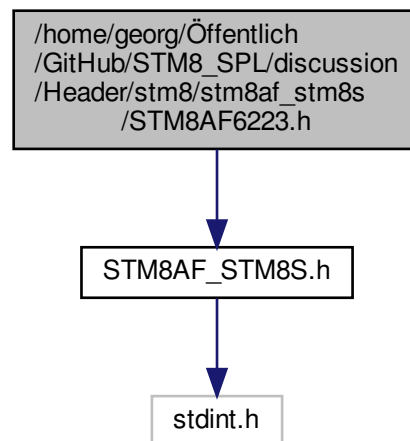
- `#define STM8AF6213A`
- `#define STM8AF621x`
- `#define STM8_PFLASH_SIZE 4*1024`
- `#define STM8_RAM_SIZE 1*1024`
- `#define STM8_EEPROM_SIZE 640`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART4_AddressBase 0x5230`

- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM5\\_AddressBase](#) 0x5300
- #define [TIM6\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.14 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6223.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6223.h:



### Macros

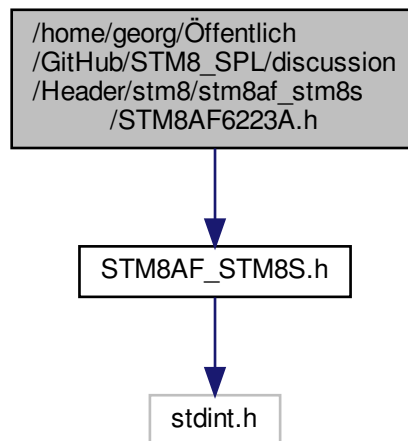
- #define [STM8AF6223](#)
- #define [STM8AF622x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 8\*1024
- #define [STM8\\_RAM\\_SIZE](#) 1\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 640
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0

- #define `RST_AddressBase` 0x50B3
- #define `CLK_AddressBase` 0x50C0
- #define `WWDG_AddressBase` 0x50D1
- #define `IWDG_AddressBase` 0x50E0
- #define `AWU_AddressBase` 0x50F0
- #define `BEEP_AddressBase` 0x50F3
- #define `SPI_AddressBase` 0x5200
- #define `I2C_AddressBase` 0x5210
- #define `UART4_AddressBase` 0x5230
- #define `TIM1_AddressBase` 0x5250
- #define `TIM5_AddressBase` 0x5300
- #define `TIM6_AddressBase` 0x5340
- #define `ADC1_AddressBase` 0x53E0
- #define `CFG_AddressBase` 0x7F60
- #define `ITC_AddressBase` 0x7F70
- #define `DM_AddressBase` 0x7F90

## 7.15 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6223A.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6223A.h:



### Macros

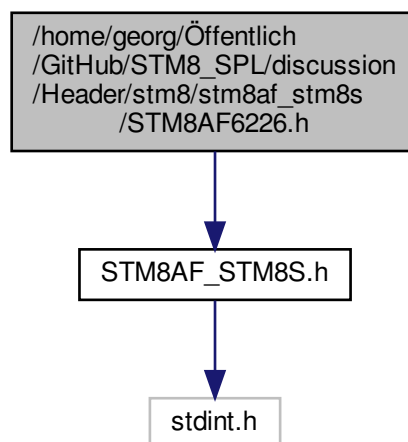
- #define `STM8AF6223A`
- #define `STM8AF622x`
- #define `STM8_PFLASH_SIZE` 8\*1024
- #define `STM8_RAM_SIZE` 1\*1024
- #define `STM8_EEPROM_SIZE` 640

- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART4\\_AddressBase](#) 0x5230
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM5\\_AddressBase](#) 0x5300
- #define [TIM6\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.16 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6226.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6226.h:



## Macros

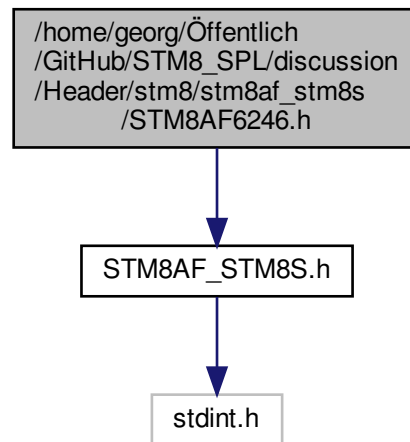
- #define [STM8AF6226](#)
- #define [STM8AF622x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 8\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 640
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART4\\_AddressBase](#) 0x5230
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM5\\_AddressBase](#) 0x5300
- #define [TIM6\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.17 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF6246.h](#) File Reference ↩

```
#include "STM8AF_STM8S.h"
```



Include dependency graph for STM8AF6246.h:



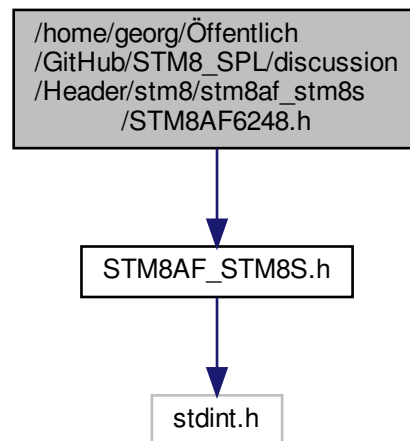
## Macros

- #define [STM8AF6246](#)
- #define [STM8AF624x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 16\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 512
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.18 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6248.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6248.h:



### Macros

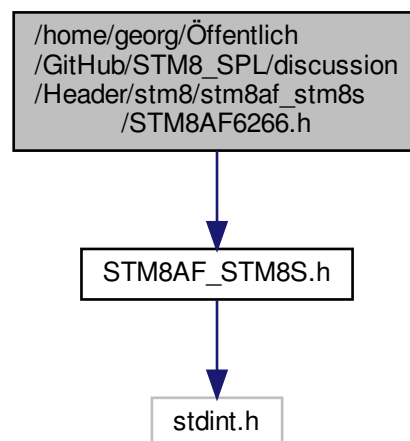
- #define STM8AF6248
- #define STM8AF624x
- #define STM8\_PFLASH\_SIZE 16\*1024
- #define STM8\_RAM\_SIZE 2\*1024
- #define STM8\_EEPROM\_SIZE 512
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210

- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.19 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6266.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6266.h:



### Macros

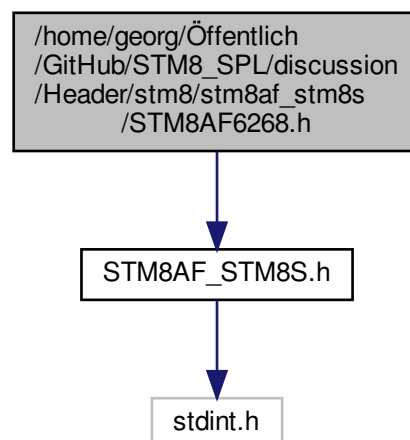
- #define [STM8AF6266](#)
- #define [STM8AF626x](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 32\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 1024
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019

- `#define PORTG_AddressBase 0x501E`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.20 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6268.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6268.h:



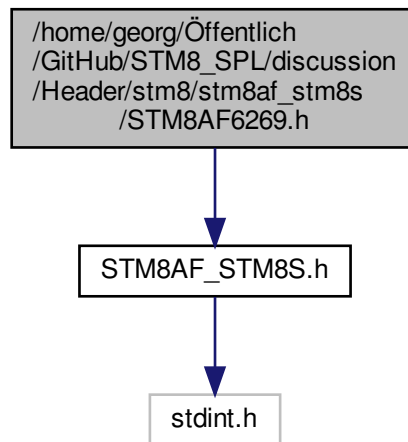
## Macros

- `#define STM8AF6268`
- `#define STM8AF626x`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 2*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.21 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6269.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6269.h:



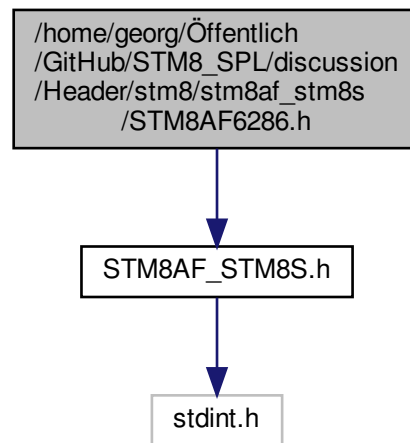
## Macros

- `#define STM8AF6269`
- `#define STM8AF626x`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.22 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6286.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6286.h:



### Macros

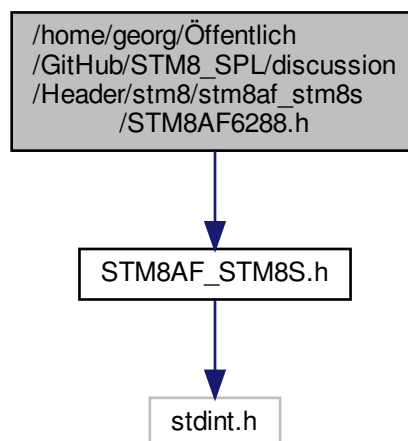
- #define STM8AF6286
- #define STM8AF628x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3

- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.23 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6288.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6288.h:



### Macros

- `#define STM8AF6288`
- `#define STM8AF628x`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`

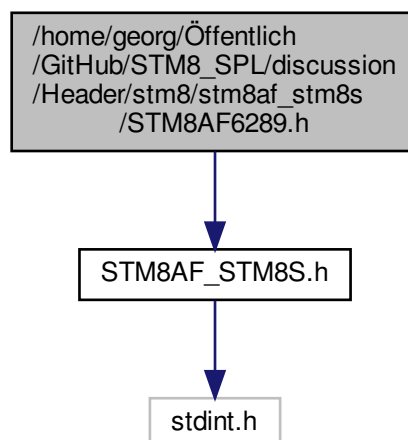


- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.24 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6289.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6289.h:



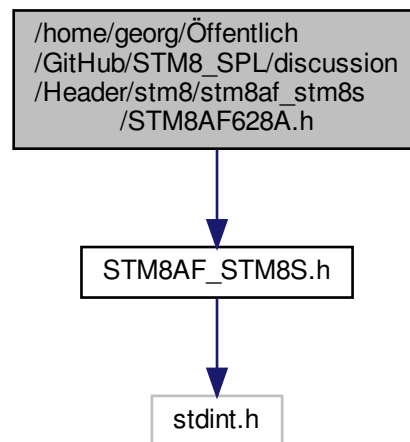
## Macros

- #define STM8AF6289
- #define STM8AF628x
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

## 7.25 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8AF628A.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF628A.h) File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF628A.h:



## Macros

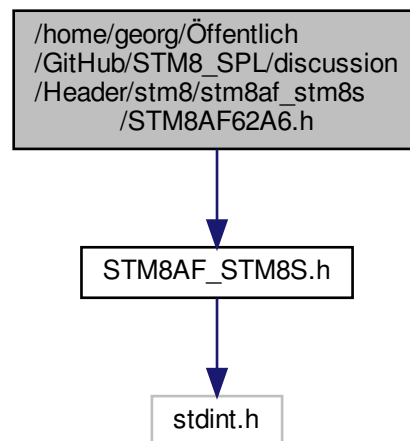
- `#define STM8AF628A`
- `#define STM8AF628x`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`

- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

## 7.26 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF62A6.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF62A6.h:



### Macros

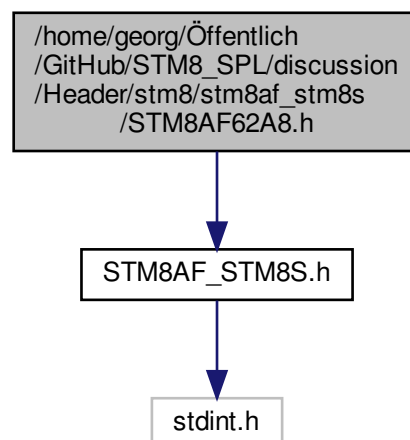
- #define STM8AF62A6
- #define STM8AF62Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023

- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.27 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF62A8.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF62A8.h:



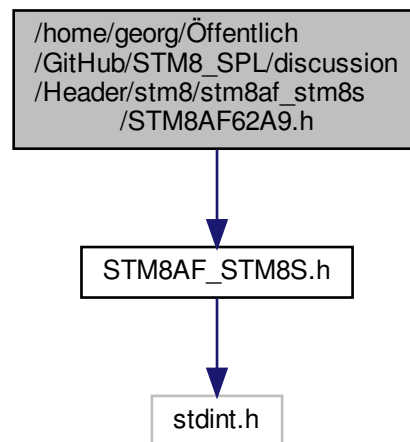
## Macros

- #define STM8AF62A8
- #define STM8AF62Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

## 7.28 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF62A9.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF62A9.h:



## Macros

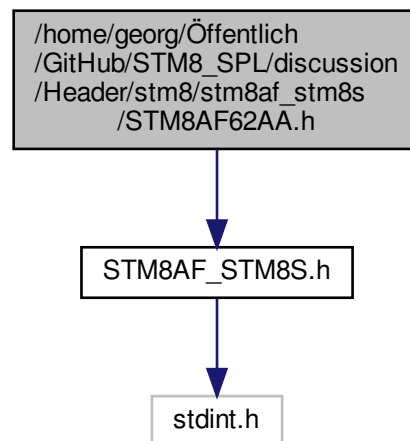
- #define [STM8AF62A9](#)
- #define [STM8AF62Ax](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 128\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250

- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90

## 7.29 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF62AA.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF62AA.h:



### Macros

- #define STM8AF62AA
- #define STM8AF62Ax
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023

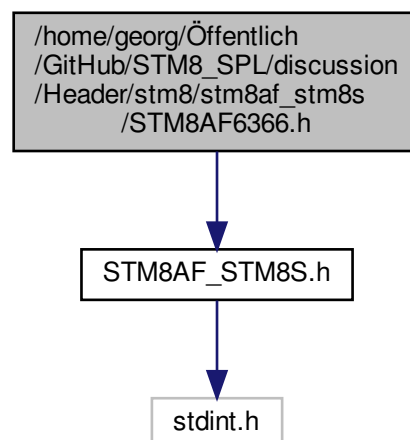


- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.30 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF6366.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6366.h:



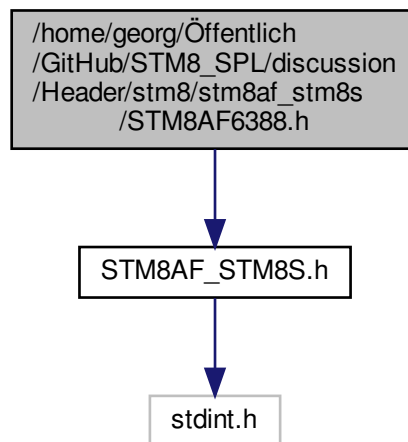
## Macros

- `#define STM8AF6366`
- `#define STM8AF636x`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 2*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

### 7.31 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8AF6388.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8AF6388.h:



## Macros

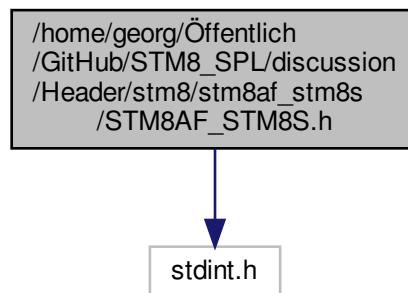
- `#define STM8AF6388`
- `#define STM8AF638x`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

## 7.32 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8AF\_STM8S.h File Reference

```
#include <stdint.h>
```

Include dependency graph for STM8AF\_STM8S.h:



### Data Structures

- struct [PORT\\_t](#)  
*structure for controlling pins in PORT mode (PORTx, x=A..I)*
- struct [FLASH\\_t](#)  
*struct to control write/erase of flash memory (FLASH)*
- struct [EXTI\\_t](#)  
*struct for configuring external port interrupts (EXTI)*
- struct [RST\\_t](#)  
*struct for determining reset source (RST)*
- struct [CLK\\_t](#)  
*struct for configuring/monitoring clock module (CLK)*
- struct [WWDG\\_t](#)  
*struct for access to Window Watchdog registers (WWDG)*
- struct [IWDG\\_t](#)  
*struct for access to Independent Timeout Watchdog registers (IWDG)*
- struct [AWU\\_t](#)  
*struct for configuring the Auto Wake-Up Module (AWU)*
- struct [BEEP\\_t](#)

- struct for beeper control (BEEP)*
- struct [SPI\\_t](#)
  - struct for controlling SPI module (SPI)*
- struct [I2C\\_t](#)
  - struct for controlling I2C module (I2C)*
- struct [UART1\\_t](#)
  - struct for controlling Universal Asynchronous Receiver Transmitter 1 (UART1)*
- struct [UART2\\_t](#)
  - struct for controlling Universal Asynchronous Receiver Transmitter 2 (UART2)*
- struct [UART3\\_t](#)
  - struct for controlling Universal Asynchronous Receiver Transmitter 3 (UART3)*
- struct [UART4\\_t](#)
  - struct for controlling Universal Asynchronous Receiver Transmitter 4 (UART4)*
- struct [TIM1\\_t](#)
  - struct for controlling 16-Bit Timer 1 (TIM1)*
- struct [TIM2\\_t](#)
  - struct for controlling 16-Bit Timer 2 (TIM2)*
- struct [TIM3\\_t](#)
  - struct for controlling 16-Bit Timer 3 (TIM3)*
- struct [TIM4\\_t](#)
  - struct for controlling 8-Bit Timer 4 (TIM4)*
- struct [TIM5\\_t](#)
  - struct for controlling 16-Bit Timer 5 (TIM5)*
- struct [TIM6\\_t](#)
  - struct for controlling 8-Bit Timer 6 (TIM6)*
- struct [ADC1\\_t](#)
  - struct containing Analog Digital Converter 1 (ADC1)*
- struct [ADC2\\_t](#)
  - struct containing Analog Digital Converter 2 (ADC2)*
- struct [CAN\\_t](#)
  - struct for controlling Controller Area Network Module (CAN)*
- struct [CFG\\_t](#)
  - struct for Global Configuration registers (CFG)*
- struct [ITC\\_t](#)
  - struct for setting interrupt Priority (ITC)*

## Macros

- #define [STM8\\_PFLASH\\_SIZE](#) 2\*1024
  - size of program flash [B]*
- #define [STM8\\_RAM\\_SIZE](#) 1\*1024
  - size of RAM [B]*
- #define [STM8\\_EEPROM\\_SIZE](#) 128
  - size of data EEPROM [B]*
- #define [STM8\\_PFLASH\\_START](#) 0x8000
  - first address in program flash*
- #define [STM8\\_PFLASH\\_END](#) ([STM8\\_PFLASH\\_START](#) + [STM8\\_PFLASH\\_SIZE](#) - 1)
  - last address in program flash*
- #define [STM8\\_RAM\\_START](#) 0x0000
  - first address in RAM*

- #define `STM8_RAM_END` (`STM8_RAM_START` + `STM8_RAM_SIZE` - 1)  
*last address in RAM*
- #define `STM8_EEPROM_START` 0x4000  
*first address in EEPROM*
- #define `STM8_EEPROM_END` (`STM8_EEPROM_START` + `STM8_EEPROM_SIZE` - 1)  
*last address in EEPROM*
- #define `STM8_ADDR_WIDTH` 16  
*width of address space*
- #define `STM8_MEM_POINTER_T` uint16\_t  
*address variable type*
- #define `ISR_HANDLER`(func, irq) void func(void) \_\_interrupt(irq)  
*handler for interrupt service routine*
- #define `ISR_HANDLER_TRAP`(func) void func() \_\_trap  
*handler for trap service routine*
- #define `NOP`() \_\_asm\_\_("nop")  
*perform a nop() operation (=minimum delay)*
- #define `DISABLE_INTERRUPTS`() \_\_asm\_\_("sim")  
*disable interrupt handling*
- #define `ENABLE_INTERRUPTS`() \_\_asm\_\_("rim")  
*enable interrupt handling*
- #define `TRIGGER_TRAP` \_\_asm\_\_("trap")  
*trigger a trap (=soft interrupt) e.g. for EMC robustness (see AN1015)*
- #define `WAIT_FOR_INTERRUPT`() \_\_asm\_\_("wfi")  
*stop code execution and wait for interrupt*
- #define `ENTER_HALT`() \_\_asm\_\_("halt")  
*put controller to HALT mode*
- #define `SW_RESET`() ( `_WWDG_CR`=0xBF)  
*reset controller via WWDG module*
- #define `_BITS` unsigned int  
*data type in bit structs (follow C90 standard)*
- #define `_SFR`(type, addr) (\*(volatile type\*) (addr))  
*peripheral register*
- #define `__TLI_VECTOR__` 0  
*irq0 - External Top Level interrupt (TLI) for pin PD7*
- #define `__AWU_VECTOR__` 1  
*irq1 - Auto Wake Up from Halt interrupt (AWU)*
- #define `__CLK_VECTOR__` 2
- #define `__PORTA_VECTOR__` 3
- #define `__PORTB_VECTOR__` 4
- #define `__PORTC_VECTOR__` 5
- #define `__PORTD_VECTOR__` 6
- #define `__PORTE_VECTOR__` 7
- #define `__CAN_RX_VECTOR__` 8  
*irq8 - CAN receive interrupt (shared with \_\_PORTF\_VECTOR\_\_)*
- #define `__PORTF_VECTOR__` 8  
*irq8 - External interrupt 5 (GPIOF, shared with \_\_CAN\_RX\_VECTOR\_\_)*
- #define `__CAN_TX_VECTOR__` 9  
*irq9 - CAN transmit interrupt*
- #define `__SPI_VECTOR__` 10
- #define `__TIM1_UPD_OVF_VECTOR__` 11
- #define `__TIM1_CAPCOM_VECTOR__` 12

- #define [\\_\\_TIM2\\_UPD\\_OVF\\_VECTOR\\_\\_](#) 13  
*irq13 - TIM2 Update/overflow interrupt (shared with [\\_\\_TIM5\\_UPD\\_OVF\\_VECTOR\\_\\_](#))*
- #define [\\_\\_TIM5\\_UPD\\_OVF\\_VECTOR\\_\\_](#) 13  
*irq13 - TIM5 Update/overflow interrupt (shared with [\\_\\_TIM2\\_UPD\\_OVF\\_VECTOR\\_\\_](#))*
- #define [\\_\\_TIM2\\_CAPCOM\\_VECTOR\\_\\_](#) 14  
*irq14 - TIM2 Capture/Compare interrupt (shared with [\\_\\_TIM5\\_CAPCOM\\_VECTOR\\_\\_](#))*
- #define [\\_\\_TIM3\\_UPD\\_OVF\\_VECTOR\\_\\_](#) 15  
*irq15 - TIM3 Update/overflow interrupt*
- #define [\\_\\_TIM3\\_CAPCOM\\_VECTOR\\_\\_](#) 16  
*irq16 - TIM3 Capture/Compare interrupt*
- #define [\\_\\_UART1\\_TXE\\_VECTOR\\_\\_](#) 17  
*irq17 - USART/UART1 send (TX empty) interrupt*
- #define [\\_\\_UART1\\_RXF\\_VECTOR\\_\\_](#) 18  
*irq18 - USART/UART1 receive (RX full) interrupt*
- #define [\\_\\_I2C\\_VECTOR\\_\\_](#) 19  
*irq19 - I2C interrupt*
- #define [\\_\\_UART2\\_TXE\\_VECTOR\\_\\_](#) 20  
*irq20 - UART2 send (TX empty) interrupt (shared with [\\_\\_UART3\\_TXE\\_VECTOR\\_\\_](#) and [\\_\\_UART4\\_TXE\\_VECTOR\\_\\_](#))*
- #define [\\_\\_UART2\\_RXF\\_VECTOR\\_\\_](#) 21  
*irq21 - UART2 receive (RX full) interrupt (shared with [\\_\\_UART3\\_RXF\\_VECTOR\\_\\_](#) and [\\_\\_UART4\\_RXF\\_VECTOR\\_\\_](#))*
- #define [\\_\\_ADC1\\_VECTOR\\_\\_](#) 22  
*irq22 - ADC1 end of conversion (shared with [\\_\\_ADC2\\_VECTOR\\_\\_](#))*
- #define [\\_\\_TIM4\\_UPD\\_OVF\\_VECTOR\\_\\_](#) 23  
*irq23 - TIM4 Update/Overflow interrupt (shared with [\\_\\_TIM6\\_UPD\\_OVF\\_VECTOR\\_\\_](#))*
- #define [\\_\\_FLASH\\_VECTOR\\_\\_](#) 24
- #define [\\_GPIOA\\_SFR\(PORT\\_t, PORTA\\_AddressBase\)](#)  
*port A struct/bit access*
- #define [\\_GPIOA\\_ODR\\_SFR\(uint8\\_t, PORTA\\_AddressBase+0x00\)](#)  
*port A output register*
- #define [\\_GPIOA\\_IDR\\_SFR\(uint8\\_t, PORTA\\_AddressBase+0x01\)](#)  
*port A input register*
- #define [\\_GPIOA\\_DDR\\_SFR\(uint8\\_t, PORTA\\_AddressBase+0x02\)](#)  
*port A direction register*
- #define [\\_GPIOA\\_CR1\\_SFR\(uint8\\_t, PORTA\\_AddressBase+0x03\)](#)  
*port A control register 1*
- #define [\\_GPIOA\\_CR2\\_SFR\(uint8\\_t, PORTA\\_AddressBase+0x04\)](#)  
*port A control register 2*
- #define [\\_GPIOB\\_SFR\(PORT\\_t, PORTB\\_AddressBase\)](#)  
*port B struct/bit access*
- #define [\\_GPIOB\\_ODR\\_SFR\(uint8\\_t, PORTB\\_AddressBase+0x00\)](#)  
*port B output register*
- #define [\\_GPIOB\\_IDR\\_SFR\(uint8\\_t, PORTB\\_AddressBase+0x01\)](#)  
*port B input register*
- #define [\\_GPIOB\\_DDR\\_SFR\(uint8\\_t, PORTB\\_AddressBase+0x02\)](#)  
*port B direction register*
- #define [\\_GPIOB\\_CR1\\_SFR\(uint8\\_t, PORTB\\_AddressBase+0x03\)](#)  
*port B control register 1*
- #define [\\_GPIOB\\_CR2\\_SFR\(uint8\\_t, PORTB\\_AddressBase+0x04\)](#)  
*port B control register 2*

- `#define _GPIOC_SFR(PORT_t, PORTC_AddressBase)`  
*port C struct/bit access*
- `#define _GPIOC_ODR_SFR(uint8_t, PORTC_AddressBase+0x00)`  
*port C output register*
- `#define _GPIOC_IDR_SFR(uint8_t, PORTC_AddressBase+0x01)`  
*port C input register*
- `#define _GPIOC_DDR_SFR(uint8_t, PORTC_AddressBase+0x02)`  
*port C direction register*
- `#define _GPIOC_CR1_SFR(uint8_t, PORTC_AddressBase+0x03)`  
*port C control register 1*
- `#define _GPIOC_CR2_SFR(uint8_t, PORTC_AddressBase+0x04)`  
*port C control register 2*
- `#define _GPIOD_SFR(PORT_t, PORTD_AddressBase)`  
*port D struct/bit access*
- `#define _GPIOD_ODR_SFR(uint8_t, PORTD_AddressBase+0x00)`  
*port D output register*
- `#define _GPIOD_IDR_SFR(uint8_t, PORTD_AddressBase+0x01)`  
*port D input register*
- `#define _GPIOD_DDR_SFR(uint8_t, PORTD_AddressBase+0x02)`  
*port D direction register*
- `#define _GPIOD_CR1_SFR(uint8_t, PORTD_AddressBase+0x03)`  
*port D control register 1*
- `#define _GPIOD_CR2_SFR(uint8_t, PORTD_AddressBase+0x04)`  
*port D control register 2*
- `#define _GPIOE_SFR(PORT_t, PORTE_AddressBase)`  
*port E struct/bit access*
- `#define _GPIOE_ODR_SFR(uint8_t, PORTE_AddressBase+0x00)`  
*port E output register*
- `#define _GPIOE_IDR_SFR(uint8_t, PORTE_AddressBase+0x01)`  
*port E input register*
- `#define _GPIOE_DDR_SFR(uint8_t, PORTE_AddressBase+0x02)`  
*port E direction register*
- `#define _GPIOE_CR1_SFR(uint8_t, PORTE_AddressBase+0x03)`  
*port E control register 1*
- `#define _GPIOE_CR2_SFR(uint8_t, PORTE_AddressBase+0x04)`  
*port E control register 2*
- `#define _GPIOF_SFR(PORT_t, PORTF_AddressBase)`  
*port F struct/bit access*
- `#define _GPIOF_ODR_SFR(uint8_t, PORTF_AddressBase+0x00)`  
*port F output register*
- `#define _GPIOF_IDR_SFR(uint8_t, PORTF_AddressBase+0x01)`  
*port F input register*
- `#define _GPIOF_DDR_SFR(uint8_t, PORTF_AddressBase+0x02)`  
*port F direction register*
- `#define _GPIOF_CR1_SFR(uint8_t, PORTF_AddressBase+0x03)`  
*port F control register 1*
- `#define _GPIOF_CR2_SFR(uint8_t, PORTF_AddressBase+0x04)`  
*port F control register 2*
- `#define _GPIOG_SFR(PORT_t, PORTG_AddressBase)`  
*port G struct/bit access*
- `#define _GPIOG_ODR_SFR(uint8_t, PORTG_AddressBase+0x00)`



- port G output register*
- #define [\\_GPIOG\\_IDR\\_SFR](#)(uint8\_t, [PORTG\\_AddressBase](#)+0x01)
- port G input register*
- #define [\\_GPIOG\\_DDR\\_SFR](#)(uint8\_t, [PORTG\\_AddressBase](#)+0x02)
- port G direction register*
- #define [\\_GPIOG\\_CR1\\_SFR](#)(uint8\_t, [PORTG\\_AddressBase](#)+0x03)
- port G control register 1*
- #define [\\_GPIOG\\_CR2\\_SFR](#)(uint8\_t, [PORTG\\_AddressBase](#)+0x04)
- port G control register 2*
- #define [\\_GPIOH\\_SFR](#)([PORT\\_t](#), [PORTH\\_AddressBase](#))
- port H struct/bit access*
- #define [\\_GPIOH\\_ODR\\_SFR](#)(uint8\_t, [PORTH\\_AddressBase](#)+0x00)
- port H output register*
- #define [\\_GPIOH\\_IDR\\_SFR](#)(uint8\_t, [PORTH\\_AddressBase](#)+0x01)
- port H input register*
- #define [\\_GPIOH\\_DDR\\_SFR](#)(uint8\_t, [PORTH\\_AddressBase](#)+0x02)
- port H direction register*
- #define [\\_GPIOH\\_CR1\\_SFR](#)(uint8\_t, [PORTH\\_AddressBase](#)+0x03)
- port H control register 1*
- #define [\\_GPIOH\\_CR2\\_SFR](#)(uint8\_t, [PORTH\\_AddressBase](#)+0x04)
- port H control register 2*
- #define [\\_GPIOI\\_SFR](#)([PORT\\_t](#), [PORTI\\_AddressBase](#))
- port I struct/bit access*
- #define [\\_GPIOI\\_ODR\\_SFR](#)(uint8\_t, [PORTI\\_AddressBase](#)+0x00)
- port I output register*
- #define [\\_GPIOI\\_IDR\\_SFR](#)(uint8\_t, [PORTI\\_AddressBase](#)+0x01)
- port I input register*
- #define [\\_GPIOI\\_DDR\\_SFR](#)(uint8\_t, [PORTI\\_AddressBase](#)+0x02)
- port I direction register*
- #define [\\_GPIOI\\_CR1\\_SFR](#)(uint8\_t, [PORTI\\_AddressBase](#)+0x03)
- port I control register 1*
- #define [\\_GPIOI\\_CR2\\_SFR](#)(uint8\_t, [PORTI\\_AddressBase](#)+0x04)
- port I control register 2*
- #define [\\_GPIO\\_ODR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- port output register reset value*
- #define [\\_GPIO\\_DDR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- port direction register reset value*
- #define [\\_GPIO\\_CR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- port control register 1 reset value*
- #define [\\_GPIO\\_CR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- port control register 2 reset value*
- #define [\\_GPIO\\_PIN0](#) ((uint8\_t) (0x01 << 0))
- port bit mask for pin 0 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)*
- #define [\\_GPIO\\_PIN1](#) ((uint8\_t) (0x01 << 1))
- port bit mask for pin 1 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)*
- #define [\\_GPIO\\_PIN2](#) ((uint8\_t) (0x01 << 2))
- port bit mask for pin 2 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)*
- #define [\\_GPIO\\_PIN3](#) ((uint8\_t) (0x01 << 3))
- port bit mask for pin 3 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)*
- #define [\\_GPIO\\_PIN4](#) ((uint8\_t) (0x01 << 4))
- port bit mask for pin 4 (in \_GPIOI\_ODR, \_GPIOI\_IDR, \_GPIOI\_DDR, \_GPIOI\_CR1, \_GPIOI\_CR2)*

- `#define _GPIO_PIN5 ((uint8_t) (0x01 << 5))`  
*port bit mask for pin 5 (in \_GPIO\_ODR, \_GPIO\_IDR, \_GPIO\_DDR, \_GPIO\_CR1, \_GPIO\_CR2)*
- `#define _GPIO_PIN6 ((uint8_t) (0x01 << 6))`  
*port bit mask for pin 6 (in \_GPIO\_ODR, \_GPIO\_IDR, \_GPIO\_DDR, \_GPIO\_CR1, \_GPIO\_CR2)*
- `#define _GPIO_PIN7 ((uint8_t) (0x01 << 7))`  
*port bit mask for pin 7 (in \_GPIO\_ODR, \_GPIO\_IDR, \_GPIO\_DDR, \_GPIO\_CR1, \_GPIO\_CR2)*
- `#define _FLASH_SFR(FLASH_t, FLASH_AddressBase)`  
*Flash struct/bit access.*
- `#define _FLASH_CR1_SFR(uint8_t, FLASH_AddressBase+0x00)`  
*Flash control register 1 (FLASH\_CR1)*
- `#define _FLASH_CR2_SFR(uint8_t, FLASH_AddressBase+0x01)`  
*Flash control register 2 (FLASH\_CR2)*
- `#define _FLASH_NCR2_SFR(uint8_t, FLASH_AddressBase+0x02)`  
*complementary Flash control register 2 (FLASH\_NCR2)*
- `#define _FLASH_FPR_SFR(uint8_t, FLASH_AddressBase+0x03)`  
*Flash protection register (FLASH\_FPR)*
- `#define _FLASH_NFPR_SFR(uint8_t, FLASH_AddressBase+0x04)`  
*complementary Flash protection register (FLASH\_NFPR)*
- `#define _FLASH_IAPSR_SFR(uint8_t, FLASH_AddressBase+0x05)`  
*Flash status register (FLASH\_IAPSR)*
- `#define _FLASH_PUKR_SFR(uint8_t, FLASH_AddressBase+0x08)`  
*Flash program memory unprotecting key register (FLASH\_PUKR)*
- `#define _FLASH_DUKR_SFR(uint8_t, FLASH_AddressBase+0x0A)`  
*Data EEPROM unprotection key register (FLASH\_DUKR)*
- `#define _FLASH_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*Flash control register 1 reset value.*
- `#define _FLASH_CR2_RESET_VALUE ((uint8_t) 0x00)`  
*Flash control register 2 reset value.*
- `#define _FLASH_NCR2_RESET_VALUE ((uint8_t) 0xFF)`  
*complementary Flash control register 2 reset value*
- `#define _FLASH_IAPSR_RESET_VALUE ((uint8_t) 0x40)`  
*Flash status register reset value.*
- `#define _FLASH_PUKR_RESET_VALUE ((uint8_t) 0x00)`  
*Flash program memory unprotecting key reset value.*
- `#define _FLASH_DUKR_RESET_VALUE ((uint8_t) 0x00)`  
*Data EEPROM unprotection key reset value.*
- `#define _FLASH_FIX ((uint8_t) (0x01 << 0))`  
*Fixed Byte programming time [0] (in \_FLASH\_CR1)*
- `#define _FLASH_IE ((uint8_t) (0x01 << 1))`  
*Flash Interrupt enable [0] (in \_FLASH\_CR1)*
- `#define _FLASH_AHALT ((uint8_t) (0x01 << 2))`  
*Power-down in Active-halt mode [0] (in \_FLASH\_CR1)*
- `#define _FLASH_HALT ((uint8_t) (0x01 << 3))`  
*Power-down in Halt mode [0] (in \_FLASH\_CR1)*
- `#define _FLASH_PRG ((uint8_t) (0x01 << 0))`  
*Standard block programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)*
- `#define _FLASH_FPRG ((uint8_t) (0x01 << 4))`  
*Fast block programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)*
- `#define _FLASH_ERASE ((uint8_t) (0x01 << 5))`  
*Block erasing [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)*
- `#define _FLASH_WPRG ((uint8_t) (0x01 << 6))`

- Word programming [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)*
  - #define `_FLASH_OPT` ((uint8\_t) (0x01 << 7))
- Write option bytes [0] (in \_FLASH\_CR2 and \_FLASH\_NCR2)*
  - #define `_FLASH_WPB` ((uint8\_t) (0x3F << 0))
- User boot code area protection bits [5:0] (in \_FLASH\_FPR and \_FLASH\_NFPR)*
  - #define `_FLASH_WPB0` ((uint8\_t) (0x01 << 0))
- User boot code area protection bit [0] (in \_FLASH\_FPR and \_FLASH\_NFPR)*
  - #define `_FLASH_WPB1` ((uint8\_t) (0x01 << 1))
- User boot code area protection bit [1] (in \_FLASH\_FPR and \_FLASH\_NFPR)*
  - #define `_FLASH_WPB2` ((uint8\_t) (0x01 << 2))
- User boot code area protection bit [2] (in \_FLASH\_FPR and \_FLASH\_NFPR)*
  - #define `_FLASH_WPB3` ((uint8\_t) (0x01 << 3))
- User boot code area protection bit [3] (in \_FLASH\_FPR and \_FLASH\_NFPR)*
  - #define `_FLASH_WPB4` ((uint8\_t) (0x01 << 4))
- User boot code area protection bit [4] (in \_FLASH\_FPR and \_FLASH\_NFPR)*
  - #define `_FLASH_WPB5` ((uint8\_t) (0x01 << 5))
- User boot code area protection bit [5] (in \_FLASH\_FPR and \_FLASH\_NFPR)*
  - #define `_FLASH_WR_PG_DIS` ((uint8\_t) (0x01 << 0))
- Write attempted to protected page flag [0] (in \_FLASH\_IAPSR)*
  - #define `_FLASH_PUL` ((uint8\_t) (0x01 << 1))
- Flash Program memory unlocked flag [0] (in \_FLASH\_IAPSR)*
  - #define `_FLASH_EOP` ((uint8\_t) (0x01 << 2))
- End of programming (write or erase operation) flag [0] (in \_FLASH\_IAPSR)*
  - #define `_FLASH_DUL` ((uint8\_t) (0x01 << 3))
- Data EEPROM area unlocked flag [0] (in \_FLASH\_IAPSR)*
  - #define `_FLASH_HVOFF` ((uint8\_t) (0x01 << 5))
- End of high voltage flag [0] (in \_FLASH\_IAPSR)*
  - #define `_EXTI_SFR(EXTI_t, EXTI_AddressBase)`
- External interrupt struct/bit access.*
  - #define `_EXTI_CR1_SFR`(uint8\_t, `EXTI_AddressBase`+0x00)
- External interrupt control register 1 (EXTI\_CR1)*
  - #define `_EXTI_CR2_SFR`(uint8\_t, `EXTI_AddressBase`+0x01)
- External interrupt control register 2 (EXTI\_CR2)*
  - #define `_EXTI_CR1_RESET_VALUE` ((uint8\_t) 0x00)
- External interrupt control register 1 reset value.*
  - #define `_EXTI_CR2_RESET_VALUE` ((uint8\_t) 0x00)
- External interrupt control register 2 reset value.*
  - #define `_EXTI_PAIS` ((uint8\_t) (0x03 << 0))
- External interrupt sensitivity for Port A [1:0] (in \_EXTI\_CR1)*
  - #define `_EXTI_PAIS0` ((uint8\_t) (0x01 << 0))
- External interrupt sensitivity for Port A [0] (in \_EXTI\_CR1)*
  - #define `_EXTI_PAIS1` ((uint8\_t) (0x01 << 1))
- External interrupt sensitivity for Port A [1] (in \_EXTI\_CR1)*
  - #define `_EXTI_PBIS` ((uint8\_t) (0x03 << 2))
- External interrupt sensitivity for Port B [1:0] (in \_EXTI\_CR1)*
  - #define `_EXTI_PBIS0` ((uint8\_t) (0x01 << 2))
- External interrupt sensitivity for Port B [0] (in \_EXTI\_CR1)*
  - #define `_EXTI_PBIS1` ((uint8\_t) (0x01 << 3))
- External interrupt sensitivity for Port B [1] (in \_EXTI\_CR1)*
  - #define `_EXTI_PCIS` ((uint8\_t) (0x03 << 4))
- External interrupt sensitivity for Port C [1:0] (in \_EXTI\_CR1)*

- `#define _EXTI_PCIS0 ((uint8_t) (0x01 << 4))`  
*External interrupt sensitivity for Port C [0] (in \_EXTI\_CR1)*
- `#define _EXTI_PCIS1 ((uint8_t) (0x01 << 5))`  
*External interrupt sensitivity for Port C [1] (in \_EXTI\_CR1)*
- `#define _EXTI_PDIS ((uint8_t) (0x03 << 6))`  
*External interrupt sensitivity for Port D [1:0] (in \_EXTI\_CR1)*
- `#define _EXTI_PDIS0 ((uint8_t) (0x01 << 6))`  
*External interrupt sensitivity for Port D [0] (in \_EXTI\_CR1)*
- `#define _EXTI_PDIS1 ((uint8_t) (0x01 << 7))`  
*External interrupt sensitivity for Port D [1] (in \_EXTI\_CR1)*
- `#define _EXTI_PEIS ((uint8_t) (0x03 << 0))`  
*Port E external interrupt sensitivity bits [1:0] (in \_EXTI\_CR2)*
- `#define _EXTI_PEIS0 ((uint8_t) (0x01 << 0))`  
*Port E external interrupt sensitivity bits [0] (in \_EXTI\_CR2)*
- `#define _EXTI_PEIS1 ((uint8_t) (0x01 << 1))`  
*Port E external interrupt sensitivity bits [1] (in \_EXTI\_CR2)*
- `#define _EXTI_TLIS ((uint8_t) (0x01 << 2))`  
*Top level interrupt sensitivity [0] (in \_EXTI\_CR2)*
- `#define _RST_SFR(RST_t, RST_AddressBase)`  
*Reset module struct/bit access.*
- `#define _RST_SR_SFR(uint8_t, RST_AddressBase+0x00)`  
*Reset module status register (RST\_SR)*
- `#define _RST_WWDGF ((uint8_t) (0x01 << 0))`  
*Window Watchdog reset flag [0] (in \_RST\_SR)*
- `#define _RST_IWDGF ((uint8_t) (0x01 << 1))`  
*Independent Watchdog reset flag [0] (in \_RST\_SR)*
- `#define _RST_ILLOPF ((uint8_t) (0x01 << 2))`  
*Illegal opcode reset flag [0] (in \_RST\_SR)*
- `#define _RST_SWIMF ((uint8_t) (0x01 << 3))`  
*SWIM reset flag [0] (in \_RST\_SR)*
- `#define _RST_EMCF ((uint8_t) (0x01 << 4))`  
*EMC reset flag [0] (in \_RST\_SR)*
- `#define _CLK_SFR(CLK_t, CLK_AddressBase)`  
*Clock module struct/bit access.*
- `#define _CLK_ICKR_SFR(uint8_t, CLK_AddressBase+0x00)`  
*Internal clock register.*
- `#define _CLK_ECKR_SFR(uint8_t, CLK_AddressBase+0x01)`  
*External clock register.*
- `#define _CLK_CMSR_SFR(uint8_t, CLK_AddressBase+0x03)`  
*Clock master status register.*
- `#define _CLK_SWR_SFR(uint8_t, CLK_AddressBase+0x04)`  
*Clock master switch register.*
- `#define _CLK_SWCR_SFR(uint8_t, CLK_AddressBase+0x05)`  
*Clock switch control register.*
- `#define _CLK_CKDIVR_SFR(uint8_t, CLK_AddressBase+0x06)`  
*Clock divider register.*
- `#define _CLK_PCKENR1_SFR(uint8_t, CLK_AddressBase+0x07)`  
*Peripheral clock gating register 1.*
- `#define _CLK_CSSR_SFR(uint8_t, CLK_AddressBase+0x08)`  
*Clock security system register.*
- `#define _CLK_CCOR_SFR(uint8_t, CLK_AddressBase+0x09)`

- *Configurable clock output register.*  
• #define `_CLK_PCKENR2_SFR`(uint8\_t, CLK\_AddressBase+0x0A)
- *Peripheral clock gating register 2.*  
• #define `_CLK_HSTRIMR_SFR`(uint8\_t, CLK\_AddressBase+0x0C)
- *HSI clock calibration trimming register.*  
• #define `_CLK_SWIMCCR_SFR`(uint8\_t, CLK\_AddressBase+0x0D)
- *SWIM clock control register.*  
• #define `_CLK_ICKR_RESET_VALUE` ((uint8\_t) 0x01)  
*Internal clock register reset value.*
- #define `_CLK_ECKR_RESET_VALUE` ((uint8\_t) 0x00)  
*External clock register reset value.*
- #define `_CLK_CMSR_RESET_VALUE` ((uint8\_t) 0xE1)  
*Clock master status reset value.*
- #define `_CLK_SWR_RESET_VALUE` ((uint8\_t) 0xE1)  
*Clock master switch reset value.*
- #define `_CLK_SWCR_RESET_VALUE` ((uint8\_t) 0x00)  
*Clock switch control reset value.*
- #define `_CLK_CKDIVR_RESET_VALUE` ((uint8\_t) 0x18)  
*Clock divider register reset value.*
- #define `_CLK_PCKENR1_RESET_VALUE` ((uint8\_t) 0xFF)  
*Peripheral clock gating register 1 reset value.*
- #define `_CLK_PCKENR2_RESET_VALUE` ((uint8\_t) 0xFF)  
*Peripheral clock gating register 2 reset value.*
- #define `_CLK_CSSR_RESET_VALUE` ((uint8\_t) 0x00)  
*Clock security system register reset value.*
- #define `_CLK_CCOR_RESET_VALUE` ((uint8\_t) 0x00)  
*Configurable clock output register reset value.*
- #define `_CLK_HSTRIMR_RESET_VALUE` ((uint8\_t) 0x00)  
*HSI clock calibration trimming register reset value.*
- #define `_CLK_SWIMCCR_RESET_VALUE` ((uint8\_t) 0x00)  
*SWIM clock control register reset value.*
- #define `_CLK_HSIEN` ((uint8\_t) (0x01 << 0))  
*High speed internal RC oscillator enable [0] (in \_CLK\_ICKR)*
- #define `_CLK_HSIRDY` ((uint8\_t) (0x01 << 1))  
*High speed internal oscillator ready [0] (in \_CLK\_ICKR)*
- #define `_CLK_FHWU` ((uint8\_t) (0x01 << 2))  
*Fast wakeup from Halt/Active-halt modes [0] (in \_CLK\_ICKR)*
- #define `_CLK_LSIEN` ((uint8\_t) (0x01 << 3))  
*Low speed internal RC oscillator enable [0] (in \_CLK\_ICKR)*
- #define `_CLK_LSIRDY` ((uint8\_t) (0x01 << 4))  
*Low speed internal oscillator ready [0] (in \_CLK\_ICKR)*
- #define `_CLK_REGAH` ((uint8\_t) (0x01 << 5))  
*Regulator power off in Active-halt mode [0] (in \_CLK\_ICKR)*
- #define `_CLK_HSEEN` ((uint8\_t) (0x01 << 0))  
*High speed external crystal oscillator enable [0] (in \_CLK\_ECKR)*
- #define `_CLK_ECKR_HSERDY` ((uint8\_t) (0x01 << 1))  
*High speed external crystal oscillator ready [0] (in \_CLK\_ECKR)*
- #define `_CLK_SWI_HSI` ((uint8\_t) 0xE1)  
*write to CLK\_SWR for HSI clock (in \_CLK\_SWR)*
- #define `_CLK_SWI_LSI` ((uint8\_t) 0xD2)  
*write to CLK\_SWR for LSI clock (in \_CLK\_SWR)*

- `#define _CLK_SWI_HSE` ((uint8\_t) 0xB4)  
*write to CLK\_SWR for HSE clock (in \_CLK\_SWR)*
- `#define _CLK_SWBSY` ((uint8\_t) (0x01 << 0))  
*Switch busy flag [0] (in \_CLK\_SWCR)*
- `#define _CLK_SWEN` ((uint8\_t) (0x01 << 1))  
*Switch start/stop enable [0] (in \_CLK\_SWCR)*
- `#define _CLK_SWIEN` ((uint8\_t) (0x01 << 2))  
*Clock switch interrupt enable [0] (in \_CLK\_SWCR)*
- `#define _CLK_SWIF` ((uint8\_t) (0x01 << 3))  
*Clock switch interrupt flag [0] (in \_CLK\_SWCR)*
- `#define _CLK_CPUDIV` ((uint8\_t) (0x07 << 0))  
*CPU clock prescaler [2:0] (in \_CLK\_CKDIVR)*
- `#define _CLK_CPUDIV0` ((uint8\_t) (0x01 << 0))  
*CPU clock prescaler [0] (in \_CLK\_CKDIVR)*
- `#define _CLK_CPUDIV1` ((uint8\_t) (0x01 << 1))  
*CPU clock prescaler [1] (in \_CLK\_CKDIVR)*
- `#define _CLK_CPUDIV2` ((uint8\_t) (0x01 << 2))  
*CPU clock prescaler [2] (in \_CLK\_CKDIVR)*
- `#define _CLK_HSIDIV` ((uint8\_t) (0x03 << 3))  
*High speed internal clock prescaler [1:0] (in \_CLK\_CKDIVR)*
- `#define _CLK_HSIDIV0` ((uint8\_t) (0x01 << 3))  
*High speed internal clock prescaler [0] (in \_CLK\_CKDIVR)*
- `#define _CLK_HSIDIV1` ((uint8\_t) (0x01 << 4))  
*High speed internal clock prescaler [1] (in \_CLK\_CKDIVR)*
- `#define _CLK_I2C` ((uint8\_t) (0x01 << 0))  
*clock enable I2C [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_SPI` ((uint8\_t) (0x01 << 1))  
*clock enable SPI [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_UART1` ((uint8\_t) (0x01 << 2))  
*clock enable UART1 [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_UART2` ((uint8\_t) (0x01 << 3))  
*clock enable UART2 [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_TIM4_TIM6` ((uint8\_t) (0x01 << 4))  
*clock enable TIM4/TIM6 [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_TIM2_TIM5` ((uint8\_t) (0x01 << 5))  
*clock enable TIM2/TIM5 [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_TIM3` ((uint8\_t) (0x01 << 6))  
*clock enable TIM3 [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_TIM1` ((uint8\_t) (0x01 << 7))  
*clock enable TIM1 [0] (in \_CLK\_PCKENR1)*
- `#define _CLK_CSSEN` ((uint8\_t) (0x01 << 0))  
*Clock security system enable [0] (in \_CLK\_CSSR)*
- `#define _CLK_AUX` ((uint8\_t) (0x01 << 1))  
*Auxiliary oscillator connected to master clock [0] (in \_CLK\_CSSR)*
- `#define _CLK_CSSDIE` ((uint8\_t) (0x01 << 2))  
*Clock security system detection interrupt enable [0] (in \_CLK\_CSSR)*
- `#define _CLK_CSSD` ((uint8\_t) (0x01 << 3))  
*Clock security system detection [0] (in \_CLK\_CSSR)*
- `#define _CLK_CCOEN` ((uint8\_t) (0x01 << 0))  
*Configurable clock output enable [0] (in \_CLK\_CCOR)*
- `#define _CLK_CCOSEL` ((uint8\_t) (0x0F << 1))

- Configurable clock output selection [3:0] (in \_CLK\_CCOR)*
- #define `_CLK_CCOSEL0` ((uint8\_t) (0x01 << 1))
- Configurable clock output selection [0] (in \_CLK\_CCOR)*
- #define `_CLK_CCOSEL1` ((uint8\_t) (0x01 << 2))
- Configurable clock output selection [1] (in \_CLK\_CCOR)*
- #define `_CLK_CCOSEL2` ((uint8\_t) (0x01 << 3))
- Configurable clock output selection [2] (in \_CLK\_CCOR)*
- #define `_CLK_CCOSEL3` ((uint8\_t) (0x01 << 4))
- Configurable clock output selection [3] (in \_CLK\_CCOR)*
- #define `_CLK_CCORDY` ((uint8\_t) (0x01 << 5))
- Configurable clock output ready [0] (in \_CLK\_CCOR)*
- #define `_CLK_CCOBSY` ((uint8\_t) (0x01 << 6))
- Configurable clock output busy [0] (in \_CLK\_CCOR)*
- #define `_CLK_AWU` ((uint8\_t) (0x01 << 2))
- clock enable AWU [0] (in \_CLK\_PCKENR2)*
- #define `_CLK_ADC` ((uint8\_t) (0x01 << 3))
- clock enable ADC [0] (in \_CLK\_PCKENR2)*
- #define `_CLK_CAN` ((uint8\_t) (0x01 << 7))
- clock enable CAN [0] (in \_CLK\_PCKENR2)*
- #define `_CLK_HSITRIM` ((uint8\_t) (0x0F << 0))
- HSI trimming value (some devices only support 3 bits, see DS!) [3:0] (in \_CLK\_HSITRIMR)*
- #define `_CLK_HSITRIM0` ((uint8\_t) (0x01 << 0))
- HSI trimming value [0] (in \_CLK\_HSITRIMR)*
- #define `_CLK_HSITRIM1` ((uint8\_t) (0x01 << 1))
- HSI trimming value [1] (in \_CLK\_HSITRIMR)*
- #define `_CLK_HSITRIM2` ((uint8\_t) (0x01 << 2))
- HSI trimming value [2] (in \_CLK\_HSITRIMR)*
- #define `_CLK_HSITRIM3` ((uint8\_t) (0x01 << 3))
- HSI trimming value [3] (in \_CLK\_HSITRIMR)*
- #define `_CLK_SWIMCLK` ((uint8\_t) (0x01 << 0))
- SWIM clock divider [0] (in \_CLK\_SWIMCCR)*
- #define `_WWDG_SFR`(WWDG\_t, WWDG\_AddressBase)
- Window Watchdog struct/bit access.*
- #define `_WWDG_CR_SFR`(uint8\_t, WWDG\_AddressBase+0x00)
- Window Watchdog Control register (WWDG\_CR)*
- #define `_WWDG_WR_SFR`(uint8\_t, WWDG\_AddressBase+0x01)
- Window Watchdog Window register (WWDG\_WR)*
- #define `_WWDG_CR_RESET_VALUE` ((uint8\_t) 0x7F)
- Window Watchdog Control register reset value.*
- #define `_WWDG_WR_RESET_VALUE` ((uint8\_t) 0x7F)
- Window Watchdog Window register reset value.*
- #define `_WWDG_T` ((uint8\_t) (0x7F << 0))
- Window Watchdog 7-bit counter [6:0] (in \_WWDG\_CR)*
- #define `_WWDG_T0` ((uint8\_t) (0x01 << 0))
- Window Watchdog 7-bit counter [0] (in \_WWDG\_CR)*
- #define `_WWDG_T1` ((uint8\_t) (0x01 << 1))
- Window Watchdog 7-bit counter [1] (in \_WWDG\_CR)*
- #define `_WWDG_T2` ((uint8\_t) (0x01 << 2))
- Window Watchdog 7-bit counter [2] (in \_WWDG\_CR)*
- #define `_WWDG_T3` ((uint8\_t) (0x01 << 3))
- Window Watchdog 7-bit counter [3] (in \_WWDG\_CR)*



- `#define _WWDG_T4` ((uint8\_t) (0x01 << 4))  
*Window Watchdog 7-bit counter [4] (in \_WWDG\_CR)*
- `#define _WWDG_T5` ((uint8\_t) (0x01 << 5))  
*Window Watchdog 7-bit counter [5] (in \_WWDG\_CR)*
- `#define _WWDG_T6` ((uint8\_t) (0x01 << 6))  
*Window Watchdog 7-bit counter [6] (in \_WWDG\_CR)*
- `#define _WWDG_WDGA` ((uint8\_t) (0x01 << 7))  
*Window Watchdog activation bit (n/a if WWDG enabled by option byte) [0] (in \_WWDG\_CR)*
- `#define _WWDG_W` ((uint8\_t) (0x7F << 0))  
*Window Watchdog 7-bit window value [6:0] (in \_WWDG\_WR)*
- `#define _WWDG_W0` ((uint8\_t) (0x01 << 0))  
*Window Watchdog 7-bit window value [0] (in \_WWDG\_WR)*
- `#define _WWDG_W1` ((uint8\_t) (0x01 << 1))  
*Window Watchdog 7-bit window value [1] (in \_WWDG\_WR)*
- `#define _WWDG_W2` ((uint8\_t) (0x01 << 2))  
*Window Watchdog 7-bit window value [2] (in \_WWDG\_WR)*
- `#define _WWDG_W3` ((uint8\_t) (0x01 << 3))  
*Window Watchdog 7-bit window value [3] (in \_WWDG\_WR)*
- `#define _WWDG_W4` ((uint8\_t) (0x01 << 4))  
*Window Watchdog 7-bit window value [4] (in \_WWDG\_WR)*
- `#define _WWDG_W5` ((uint8\_t) (0x01 << 5))  
*Window Watchdog 7-bit window value [5] (in \_WWDG\_WR)*
- `#define _WWDG_W6` ((uint8\_t) (0x01 << 6))  
*Window Watchdog 7-bit window value [6] (in \_WWDG\_WR)*
- `#define _IWDG_SFR(IWDG_t, IWDG_AddressBase)`  
*Independent Timeout Watchdog struct/bit access.*
- `#define _IWDG_KR_SFR`(uint8\_t, IWDG\_AddressBase+0x00)  
*Independent Timeout Watchdog Key register (IWDG\_KR)*
- `#define _IWDG_PR_SFR`(uint8\_t, IWDG\_AddressBase+0x01)  
*Independent Timeout Watchdog Prescaler register (IWDG\_PR)*
- `#define _IWDG_RLR_SFR`(uint8\_t, IWDG\_AddressBase+0x02)  
*Independent Timeout Watchdog Reload register (IWDG\_RLR)*
- `#define _IWDG_PR_RESET_VALUE` ((uint8\_t) 0x00)  
*Independent Timeout Watchdog Prescaler register reset value.*
- `#define _IWDG_RLR_RESET_VALUE` ((uint8\_t) 0xFF)  
*Independent Timeout Watchdog Reload register reset value.*
- `#define _IWDG_KEY_ENABLE` ((uint8\_t) 0xCC)  
*Independent Timeout Watchdog enable (in \_IWDG\_KR)*
- `#define _IWDG_KEY_REFRESH` ((uint8\_t) 0xAA)  
*Independent Timeout Watchdog refresh (in \_IWDG\_KR)*
- `#define _IWDG_KEY_ACCESS` ((uint8\_t) 0x55)  
*Independent Timeout Watchdog unlock write to IWDG\_PR and IWDG\_RLR (in \_IWDG\_KR)*
- `#define _IWDG_PRE` ((uint8\_t) (0x07 << 0))  
*Independent Timeout Watchdog Prescaler divider [2:0] (in \_IWDG\_PR)*
- `#define _IWDG_PRE0` ((uint8\_t) (0x01 << 0))  
*Independent Timeout Watchdog Prescaler divider [0] (in \_IWDG\_PR)*
- `#define _IWDG_PRE1` ((uint8\_t) (0x01 << 1))  
*Independent Timeout Watchdog Prescaler divider [1] (in \_IWDG\_PR)*
- `#define _IWDG_PRE2` ((uint8\_t) (0x01 << 2))  
*Independent Timeout Watchdog Prescaler divider [2] (in \_IWDG\_PR)*
- `#define _AWU_SFR(AWU_t, AWU_AddressBase)`



- Auto Wake-Up struct/bit access.*
- #define `_AWU_CSR_SFR`(uint8\_t, `AWU_AddressBase`+0x00)  
*Auto Wake-Up Control/status register (AWU\_CSR)*
- #define `_AWU_APR_SFR`(uint8\_t, `AWU_AddressBase`+0x01)  
*Auto Wake-Up Asynchronous prescaler register (AWU\_APR)*
- #define `_AWU_TBR_SFR`(uint8\_t, `AWU_AddressBase`+0x02)  
*Auto Wake-Up Timebase selection register (AWU\_TBR)*
- #define `_AWU_CSR_RESET_VALUE` ((uint8\_t) 0x00)  
*Auto Wake-Up Control/status register reset value.*
- #define `_AWU_APR_RESET_VALUE` ((uint8\_t) 0x3F)  
*Auto Wake-Up Asynchronous prescaler register reset value.*
- #define `_AWU_TBR_RESET_VALUE` ((uint8\_t) 0x00)  
*Auto Wake-Up Timebase selection register reset value.*
- #define `_AWU_MSR` ((uint8\_t) (0x01 << 0))  
*Auto Wake-Up LSI measurement enable [0] (in \_AWU\_CSR)*
- #define `_AWU_AWUEN` ((uint8\_t) (0x01 << 4))  
*Auto-wakeup enable [0] (in \_AWU\_CSR)*
- #define `_AWU_AWUF` ((uint8\_t) (0x01 << 5))  
*Auto-wakeup status flag [0] (in \_AWU\_CSR)*
- #define `_AWU_APRE` ((uint8\_t) (0x3F << 0))  
*Auto-wakeup asynchronous prescaler divider [5:0] (in \_AWU\_APR)*
- #define `_AWU_APRE0` ((uint8\_t) (0x01 << 0))  
*Auto-wakeup asynchronous prescaler divider [0] (in \_AWU\_APR)*
- #define `_AWU_APRE1` ((uint8\_t) (0x01 << 1))  
*Auto-wakeup asynchronous prescaler divider [1] (in \_AWU\_APR)*
- #define `_AWU_APRE2` ((uint8\_t) (0x01 << 2))  
*Auto-wakeup asynchronous prescaler divider [2] (in \_AWU\_APR)*
- #define `_AWU_APRE3` ((uint8\_t) (0x01 << 3))  
*Auto-wakeup asynchronous prescaler divider [3] (in \_AWU\_APR)*
- #define `_AWU_APRE4` ((uint8\_t) (0x01 << 4))  
*Auto-wakeup asynchronous prescaler divider [4] (in \_AWU\_APR)*
- #define `_AWU_APRE5` ((uint8\_t) (0x01 << 5))  
*Auto-wakeup asynchronous prescaler divider [5] (in \_AWU\_APR)*
- #define `_AWU_AWUTB` ((uint8\_t) (0x0F << 0))  
*Auto-wakeup timebase selection [3:0] (in \_AWU\_APR)*
- #define `_AWU_AWUTB0` ((uint8\_t) (0x01 << 0))  
*Auto-wakeup timebase selection [0] (in \_AWU\_APR)*
- #define `_AWU_AWUTB1` ((uint8\_t) (0x01 << 1))  
*Auto-wakeup timebase selection [1] (in \_AWU\_APR)*
- #define `_AWU_AWUTB2` ((uint8\_t) (0x01 << 2))  
*Auto-wakeup timebase selection [2] (in \_AWU\_APR)*
- #define `_AWU_AWUTB3` ((uint8\_t) (0x01 << 3))  
*Auto-wakeup timebase selection [3] (in \_AWU\_APR)*
- #define `_BEEP_SFR`(BEEP\_t, `BEEP_AddressBase`)  
*Beeper struct/bit access.*
- #define `_BEEP_CSR_SFR`(uint8\_t, `BEEP_AddressBase`+0x00)  
*Beeper control/status register (BEEP\_CSR)*
- #define `_BEEP_CSR_RESET_VALUE` ((uint8\_t) 0x1F)  
*Beeper control/status register reset value.*
- #define `_BEEP_BEEP_DIV` ((uint8\_t) (0x1F << 0))  
*Beeper clock prescaler divider [4:0] (in \_BEEP\_CSR)*

- `#define _BEEP_BEEDIV0 ((uint8_t) (0x01 << 0))`  
*Beeper clock prescaler divider [0] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEDIV1 ((uint8_t) (0x01 << 1))`  
*Beeper clock prescaler divider [1] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEDIV2 ((uint8_t) (0x01 << 2))`  
*Beeper clock prescaler divider [2] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEDIV3 ((uint8_t) (0x01 << 3))`  
*Beeper clock prescaler divider [3] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEDIV4 ((uint8_t) (0x01 << 4))`  
*Beeper clock prescaler divider [4] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEPEN ((uint8_t) (0x01 << 5))`  
*Beeper enable [0] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEPSEL ((uint8_t) (0x03 << 6))`  
*Beeper frequency selection [1:0] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEPSEL0 ((uint8_t) (0x01 << 6))`  
*Beeper frequency selection [0] (in \_BEEP\_CSR)*
- `#define _BEEP_BEEPSEL1 ((uint8_t) (0x01 << 7))`  
*Beeper frequency selection [1] (in \_BEEP\_CSR)*
- `#define _SPI_SFR(SPI_t, SPI_AddressBase)`  
*register for SPI control*
- `#define _SPI_CR1_SFR(uint8_t, SPI_AddressBase+0x00)`  
*SPI control register 1.*
- `#define _SPI_CR2_SFR(uint8_t, SPI_AddressBase+0x01)`  
*SPI control register 2.*
- `#define _SPI_ICR_SFR(uint8_t, SPI_AddressBase+0x02)`  
*SPI interrupt control register.*
- `#define _SPI_SR_SFR(uint8_t, SPI_AddressBase+0x03)`  
*SPI status register.*
- `#define _SPI_DR_SFR(uint8_t, SPI_AddressBase+0x04)`  
*SPI data register.*
- `#define _SPI_CRCPR_SFR(uint8_t, SPI_AddressBase+0x05)`  
*SPI CRC polynomial register.*
- `#define _SPI_RXCR_SFR(uint8_t, SPI_AddressBase+0x06)`  
*SPI Rx CRC register.*
- `#define _SPI_TXCR_SFR(uint8_t, SPI_AddressBase+0x07)`  
*SPI Tx CRC register.*
- `#define _SPI_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*SPI Control Register 1 reset value.*
- `#define _SPI_CR2_RESET_VALUE ((uint8_t) 0x00)`  
*SPI Control Register 2 reset value.*
- `#define _SPI_ICR_RESET_VALUE ((uint8_t) 0x00)`  
*SPI Interrupt Control Register reset value.*
- `#define _SPI_SR_RESET_VALUE ((uint8_t) 0x02)`  
*SPI Status Register reset value.*
- `#define _SPI_DR_RESET_VALUE ((uint8_t) 0x00)`  
*SPI Data Register reset value.*
- `#define _SPI_CRCPR_RESET_VALUE ((uint8_t) 0x07)`  
*SPI Polynomial Register reset value.*
- `#define _SPI_RXCR_RESET_VALUE ((uint8_t) 0x00)`  
*SPI RX CRC Register reset value.*
- `#define _SPI_TXCR_RESET_VALUE ((uint8_t) 0x00)`

- SPI TX CRC Register reset value.*

  - #define `_SPI_CPHA` ((uint8\_t) (0x01 << 0))

*SPI Clock phase [0] (in \_SPI\_CR1)*
- #define `_SPI_CPOL` ((uint8\_t) (0x01 << 1))

*SPI Clock polarity [0] (in \_SPI\_CR1)*
- #define `_SPI_MSTR` ((uint8\_t) (0x01 << 2))

*SPI Master/slave selection [0] (in \_SPI\_CR1)*
- #define `_SPI_BR` ((uint8\_t) (0x07 << 3))

*SPI Baudrate control [2:0] (in \_SPI\_CR1)*
- #define `_SPI_BR0` ((uint8\_t) (0x01 << 3))

*SPI Baudrate control [0] (in \_SPI\_CR1)*
- #define `_SPI_BR1` ((uint8\_t) (0x01 << 4))

*SPI Baudrate control [1] (in \_SPI\_CR1)*
- #define `_SPI_BR2` ((uint8\_t) (0x01 << 5))

*SPI Baudrate control [2] (in \_SPI\_CR1)*
- #define `_SPI_SPE` ((uint8\_t) (0x01 << 6))

*SPI enable [0] (in \_SPI\_CR1)*
- #define `_SPI_LSBFIRST` ((uint8\_t) (0x01 << 7))

*SPI Frame format [0] (in \_SPI\_CR1)*
- #define `_SPI_SSI` ((uint8\_t) (0x01 << 0))

*SPI Internal slave select [0] (in \_SPI\_CR2)*
- #define `_SPI_SSM` ((uint8\_t) (0x01 << 1))

*SPI Software slave management [0] (in \_SPI\_CR2)*
- #define `_SPI_RXONLY` ((uint8\_t) (0x01 << 2))

*SPI Receive only [0] (in \_SPI\_CR2)*
- #define `_SPI_CRCNEXT` ((uint8\_t) (0x01 << 4))

*SPI Transmit CRC next [0] (in \_SPI\_CR2)*
- #define `_SPI_CRCEN` ((uint8\_t) (0x01 << 5))

*SPI Hardware CRC calculation enable [0] (in \_SPI\_CR2)*
- #define `_SPI_BDOE` ((uint8\_t) (0x01 << 6))

*SPI Input/Output enable in bidirectional mode [0] (in \_SPI\_CR2)*
- #define `_SPI_BDM` ((uint8\_t) (0x01 << 7))

*SPI Bidirectional data mode enable [0] (in \_SPI\_CR2)*
- #define `_SPI_WKIE` ((uint8\_t) (0x01 << 4))

*SPI Wakeup interrupt enable [0] (in \_SPI\_ICR)*
- #define `_SPI_ERRIE` ((uint8\_t) (0x01 << 5))

*SPI Error interrupt enable [0] (in \_SPI\_ICR)*
- #define `_SPI_RXIE` ((uint8\_t) (0x01 << 6))

*SPI Rx buffer not empty interrupt enable [0] (in \_SPI\_ICR)*
- #define `_SPI_TXIE` ((uint8\_t) (0x01 << 7))

*SPI Tx buffer empty interrupt enable [0] (in \_SPI\_ICR)*
- #define `_SPI_RXNE` ((uint8\_t) (0x01 << 0))

*SPI Receive buffer not empty [0] (in \_SPI\_SR)*
- #define `_SPI_TXE` ((uint8\_t) (0x01 << 1))

*SPI Transmit buffer empty [0] (in \_SPI\_SR)*
- #define `_SPI_WKUP` ((uint8\_t) (0x01 << 3))

*SPI Wakeup flag [0] (in \_SPI\_SR)*
- #define `_SPI_CRCERR` ((uint8\_t) (0x01 << 4))

*SPI CRC error flag [0] (in \_SPI\_SR)*
- #define `_SPI_MODF` ((uint8\_t) (0x01 << 5))

*SPI Mode fault [0] (in \_SPI\_SR)*

- `#define _SPI_OVR ((uint8_t) (0x01 << 6))`  
*SPI Overrun flag [0] (in \_SPI\_SR)*
- `#define _SPI_BSY ((uint8_t) (0x01 << 7))`  
*SPI Busy flag [0] (in \_SPI\_SR)*
- `#define _I2C_SFR(I2C_t, I2C_AddressBase)`  
*register for SPI control*
- `#define _I2C_CR1_SFR(uint8_t, I2C_AddressBase+0x00)`  
*I2C Control register 1.*
- `#define _I2C_CR2_SFR(uint8_t, I2C_AddressBase+0x01)`  
*I2C Control register 2.*
- `#define _I2C_FREQR_SFR(uint8_t, I2C_AddressBase+0x02)`  
*I2C Frequency register.*
- `#define _I2C_OARL_SFR(uint8_t, I2C_AddressBase+0x03)`  
*I2C own address register low byte.*
- `#define _I2C_OARH_SFR(uint8_t, I2C_AddressBase+0x04)`  
*I2C own address register high byte.*
- `#define _I2C_DR_SFR(uint8_t, I2C_AddressBase+0x06)`  
*I2C data register.*
- `#define _I2C_SR1_SFR(uint8_t, I2C_AddressBase+0x07)`  
*I2C Status register 1.*
- `#define _I2C_SR2_SFR(uint8_t, I2C_AddressBase+0x08)`  
*I2C Status register 2.*
- `#define _I2C_SR3_SFR(uint8_t, I2C_AddressBase+0x09)`  
*I2C Status register 3.*
- `#define _I2C_ITR_SFR(uint8_t, I2C_AddressBase+0x0A)`  
*I2C Interrupt register.*
- `#define _I2C_CCRL_SFR(uint8_t, I2C_AddressBase+0x0B)`  
*I2C Clock control register low byte.*
- `#define _I2C_CCRH_SFR(uint8_t, I2C_AddressBase+0x0C)`  
*I2C Clock control register high byte.*
- `#define _I2C_TRISER_SFR(uint8_t, I2C_AddressBase+0x0D)`  
*I2C rise time register.*
- `#define _I2C_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Control register 1 reset value.*
- `#define _I2C_CR2_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Control register 2 reset value.*
- `#define _I2C_FREQR_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Frequency register reset value.*
- `#define _I2C_OARL_RESET_VALUE ((uint8_t) 0x00)`  
*I2C own address register low byte reset value.*
- `#define _I2C_OARH_RESET_VALUE ((uint8_t) 0x00)`  
*I2C own address register high byte reset value.*
- `#define _I2C_DR_RESET_VALUE ((uint8_t) 0x00)`  
*I2C data register reset value.*
- `#define _I2C_SR1_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Status register 1 reset value.*
- `#define _I2C_SR2_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Status register 2 reset value.*
- `#define _I2C_SR3_RESET_VALUE ((uint8_t) 0x00)`  
*I2C Status register 3 reset value.*
- `#define _I2C_ITR_RESET_VALUE ((uint8_t) 0x00)`

- I2C Interrupt register reset value.*
  - #define `_I2C_CCRL_RESET_VALUE` ((uint8\_t) 0x00)
- I2C Clock control register low byte reset value.*
  - #define `_I2C_CCRH_RESET_VALUE` ((uint8\_t) 0x00)
- I2C Clock control register high byte reset value.*
  - #define `_I2C_TRISER_RESET_VALUE` ((uint8\_t) 0x02)
- I2C rise time register reset value.*
  - #define `_I2C_PE` ((uint8\_t) (0x01 << 0))
- I2C Peripheral enable [0] (in \_I2C\_CR1)*
  - #define `_I2C_ENGC` ((uint8\_t) (0x01 << 6))
- I2C General call enable [0] (in \_I2C\_CR1)*
  - #define `_I2C_NOSTRETCH` ((uint8\_t) (0x01 << 7))
- I2C Clock stretching disable (Slave mode) [0] (in \_I2C\_CR1)*
  - #define `_I2C_START` ((uint8\_t) (0x01 << 0))
- I2C Start generation [0] (in \_I2C\_CR2)*
  - #define `_I2C_STOP` ((uint8\_t) (0x01 << 1))
- I2C Stop generation [0] (in \_I2C\_CR2)*
  - #define `_I2C_ACK` ((uint8\_t) (0x01 << 2))
- I2C Acknowledge enable [0] (in \_I2C\_CR2)*
  - #define `_I2C_POS` ((uint8\_t) (0x01 << 3))
- I2C Acknowledge position (for data reception) [0] (in \_I2C\_CR2)*
  - #define `_I2C_SWRST` ((uint8\_t) (0x01 << 7))
- I2C Software reset [0] (in \_I2C\_CR2)*
  - #define `_I2C_FREQ` ((uint8\_t) (0x3F << 0))
- I2C Peripheral clock frequency [5:0] (in \_I2C\_FREQR)*
  - #define `_I2C_FREQ0` ((uint8\_t) (0x01 << 0))
- I2C Peripheral clock frequency [0] (in \_I2C\_FREQR)*
  - #define `_I2C_FREQ1` ((uint8\_t) (0x01 << 1))
- I2C Peripheral clock frequency [1] (in \_I2C\_FREQR)*
  - #define `_I2C_FREQ2` ((uint8\_t) (0x01 << 2))
- I2C Peripheral clock frequency [2] (in \_I2C\_FREQR)*
  - #define `_I2C_FREQ3` ((uint8\_t) (0x01 << 3))
- I2C Peripheral clock frequency [3] (in \_I2C\_FREQR)*
  - #define `_I2C_FREQ4` ((uint8\_t) (0x01 << 4))
- I2C Peripheral clock frequency [4] (in \_I2C\_FREQR)*
  - #define `_I2C_FREQ5` ((uint8\_t) (0x01 << 5))
- I2C Peripheral clock frequency [5] (in \_I2C\_FREQR)*
  - #define `_I2C_ADD0` ((uint8\_t) (0x01 << 0))
- I2C Interface address [0] (in 10-bit address mode) (in \_I2C\_OARL)*
  - #define `_I2C_ADD1` ((uint8\_t) (0x01 << 1))
- I2C Interface address [1] (in \_I2C\_OARL)*
  - #define `_I2C_ADD2` ((uint8\_t) (0x01 << 2))
- I2C Interface address [2] (in \_I2C\_OARL)*
  - #define `_I2C_ADD3` ((uint8\_t) (0x01 << 3))
- I2C Interface address [3] (in \_I2C\_OARL)*
  - #define `_I2C_ADD4` ((uint8\_t) (0x01 << 4))
- I2C Interface address [4] (in \_I2C\_OARL)*
  - #define `_I2C_ADD5` ((uint8\_t) (0x01 << 5))
- I2C Interface address [5] (in \_I2C\_OARL)*
  - #define `_I2C_ADD6` ((uint8\_t) (0x01 << 6))
- I2C Interface address [6] (in \_I2C\_OARL)*

- `#define _I2C_ADD7 ((uint8_t) (0x01 << 7))`  
*I2C Interface address [7] (in \_I2C\_OARL)*
- `#define _I2C_ADD_8_9 ((uint8_t) (0x03 << 1))`  
*I2C Interface address [9:8] (in 10-bit address mode) (in \_I2C\_OARH)*
- `#define _I2C_ADD8 ((uint8_t) (0x01 << 1))`  
*I2C Interface address [8] (in \_I2C\_OARH)*
- `#define _I2C_ADD9 ((uint8_t) (0x01 << 2))`  
*I2C Interface address [9] (in \_I2C\_OARH)*
- `#define _I2C_ADDCONF ((uint8_t) (0x01 << 6))`  
*I2C Address mode configuration [0] (in \_I2C\_OARH)*
- `#define _I2C_ADDMODE ((uint8_t) (0x01 << 7))`  
*I2C 7-/10-bit addressing mode (Slave mode) [0] (in \_I2C\_OARH)*
- `#define _I2C_SB ((uint8_t) (0x01 << 0))`  
*I2C Start bit (Mastermode) [0] (in \_I2C\_SR1)*
- `#define _I2C_ADDR ((uint8_t) (0x01 << 1))`  
*I2C Address sent (master mode) / matched (slave mode) [0] (in \_I2C\_SR1)*
- `#define _I2C_BTFF ((uint8_t) (0x01 << 2))`  
*I2C Byte transfer finished [0] (in \_I2C\_SR1)*
- `#define _I2C_ADD10 ((uint8_t) (0x01 << 3))`  
*I2C 10-bit header sent (Master mode) [0] (in \_I2C\_SR1)*
- `#define _I2C_STOPF ((uint8_t) (0x01 << 4))`  
*I2C Stop detection (Slave mode) [0] (in \_I2C\_SR1)*
- `#define _I2C_RXNE ((uint8_t) (0x01 << 6))`  
*I2C Data register not empty (receivers) [0] (in \_I2C\_SR1)*
- `#define _I2C_TXE ((uint8_t) (0x01 << 7))`  
*I2C Data register empty (transmitters) [0] (in \_I2C\_SR1)*
- `#define _I2C_BERR ((uint8_t) (0x01 << 0))`  
*I2C Bus error [0] (in \_I2C\_SR2)*
- `#define _I2C_ARLO ((uint8_t) (0x01 << 1))`  
*I2C Arbitration lost (master mode) [0] (in \_I2C\_SR2)*
- `#define _I2C_AF ((uint8_t) (0x01 << 2))`  
*I2C Acknowledge failure [0] (in \_I2C\_SR2)*
- `#define _I2C_OVR ((uint8_t) (0x01 << 3))`  
*I2C Overrun/underrun [0] (in \_I2C\_SR2)*
- `#define _I2C_WUFH ((uint8_t) (0x01 << 5))`  
*I2C Wakeup from Halt [0] (in \_I2C\_SR2)*
- `#define _I2C_MSL ((uint8_t) (0x01 << 0))`  
*I2C Master/Slave [0] (in \_I2C\_SR3)*
- `#define _I2C_BUSY ((uint8_t) (0x01 << 1))`  
*I2C Bus busy [0] (in \_I2C\_SR3)*
- `#define _I2C_TRA ((uint8_t) (0x01 << 2))`  
*I2C Transmitter/Receiver [0] (in \_I2C\_SR3)*
- `#define _I2C_GENCALL ((uint8_t) (0x01 << 4))`  
*I2C General call header (Slavemode) [0] (in \_I2C\_SR3)*
- `#define _I2C_ITERREN ((uint8_t) (0x01 << 0))`  
*I2C Error interrupt enable [0] (in \_I2C\_ITR)*
- `#define _I2C_ITEVTEN ((uint8_t) (0x01 << 1))`  
*I2C Event interrupt enable [0] (in \_I2C\_ITR)*
- `#define _I2C_ITBUFEN ((uint8_t) (0x01 << 2))`  
*I2C Buffer interrupt enable [0] (in \_I2C\_ITR)*
- `#define _I2C_CCR ((uint8_t) (0x0F << 0))`

- I2C Clock control register (Master mode) [3:0] (in \_I2C\_CCRH)*
- #define `_I2C_CCR0` ((uint8\_t) (0x01 << 0))
- I2C Clock control register (Master mode) [0] (in \_I2C\_CCRH)*
- #define `_I2C_CCR1` ((uint8\_t) (0x01 << 1))
- I2C Clock control register (Master mode) [1] (in \_I2C\_CCRH)*
- #define `_I2C_CCR2` ((uint8\_t) (0x01 << 2))
- I2C Clock control register (Master mode) [2] (in \_I2C\_CCRH)*
- #define `_I2C_CCR3` ((uint8\_t) (0x01 << 3))
- I2C Clock control register (Master mode) [3] (in \_I2C\_CCRH)*
- #define `_I2C_DUTY` ((uint8\_t) (0x01 << 6))
- I2C Fast mode duty cycle [0] (in \_I2C\_CCRH)*
- #define `_I2C_FS` ((uint8\_t) (0x01 << 7))
- I2C master mode selection [0] (in \_I2C\_CCRH)*
- #define `_I2C_TRISE` ((uint8\_t) (0x3F << 0))
- I2C Maximum rise time (Master mode) [5:0] (in \_I2C\_TRISE)*
- #define `_I2C_TRISE0` ((uint8\_t) (0x01 << 0))
- I2C Maximum rise time (Master mode) [0] (in \_I2C\_TRISE)*
- #define `_I2C_TRISE1` ((uint8\_t) (0x01 << 1))
- I2C Maximum rise time (Master mode) [1] (in \_I2C\_TRISE)*
- #define `_I2C_TRISE2` ((uint8\_t) (0x01 << 2))
- I2C Maximum rise time (Master mode) [2] (in \_I2C\_TRISE)*
- #define `_I2C_TRISE3` ((uint8\_t) (0x01 << 3))
- I2C Maximum rise time (Master mode) [3] (in \_I2C\_TRISE)*
- #define `_I2C_TRISE4` ((uint8\_t) (0x01 << 4))
- I2C Maximum rise time (Master mode) [4] (in \_I2C\_TRISE)*
- #define `_I2C_TRISE5` ((uint8\_t) (0x01 << 5))
- I2C Maximum rise time (Master mode) [5] (in \_I2C\_TRISE)*
- #define `_UART1_SFR(UART1_t, UART1_AddressBase)`
- UART1 struct/bit access.*
- #define `_UART1_SR_SFR`(uint8\_t, `UART1_AddressBase`+0x00)
- UART1 Status register.*
- #define `_UART1_DR_SFR`(uint8\_t, `UART1_AddressBase`+0x01)
- UART1 data register.*
- #define `_UART1_BRR1_SFR`(uint8\_t, `UART1_AddressBase`+0x02)
- UART1 Baud rate register 1.*
- #define `_UART1_BRR2_SFR`(uint8\_t, `UART1_AddressBase`+0x03)
- UART1 Baud rate register 2.*
- #define `_UART1_CR1_SFR`(uint8\_t, `UART1_AddressBase`+0x04)
- UART1 Control register 1.*
- #define `_UART1_CR2_SFR`(uint8\_t, `UART1_AddressBase`+0x05)
- UART1 Control register 2.*
- #define `_UART1_CR3_SFR`(uint8\_t, `UART1_AddressBase`+0x06)
- UART1 Control register 3.*
- #define `_UART1_CR4_SFR`(uint8\_t, `UART1_AddressBase`+0x07)
- UART1 Control register 4.*
- #define `_UART1_CR5_SFR`(uint8\_t, `UART1_AddressBase`+0x08)
- UART1 Control register 5.*
- #define `_UART1_GTR_SFR`(uint8\_t, `UART1_AddressBase`+0x09)
- UART1 guard time register.*
- #define `_UART1_PSCR_SFR`(uint8\_t, `UART1_AddressBase`+0x0A)
- UART1 prescaler register.*



- `#define _UART1_SR_RESET_VALUE` ((uint8\_t) 0xC0)  
*UART1 Status register reset value.*
- `#define _UART1_BRR1_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 Baud rate register 1 reset value.*
- `#define _UART1_BRR2_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 Baud rate register 2 reset value.*
- `#define _UART1_CR1_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 Control register 1 reset value.*
- `#define _UART1_CR2_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 Control register 2 reset value.*
- `#define _UART1_CR3_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 Control register 3 reset value.*
- `#define _UART1_CR4_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 Control register 4 reset value.*
- `#define _UART1_CR5_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 Control register 5 reset value.*
- `#define _UART1_GTR_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 guard time register reset value.*
- `#define _UART1_PSCR_RESET_VALUE` ((uint8\_t) 0x00)  
*UART1 prescaler register reset value.*
- `#define _UART1_PE` ((uint8\_t) (0x01 << 0))  
*UART1 Parity error [0] (in \_UART1\_SR)*
- `#define _UART1_FE` ((uint8\_t) (0x01 << 1))  
*UART1 Framing error [0] (in \_UART1\_SR)*
- `#define _UART1_NF` ((uint8\_t) (0x01 << 2))  
*UART1 Noise flag [0] (in \_UART1\_SR)*
- `#define _UART1_OR_LHE` ((uint8\_t) (0x01 << 3))  
*UART1 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART1\_SR)*
- `#define _UART1_IDLE` ((uint8\_t) (0x01 << 4))  
*UART1 IDLE line detected [0] (in \_UART1\_SR)*
- `#define _UART1_RXNE` ((uint8\_t) (0x01 << 5))  
*UART1 Read data register not empty [0] (in \_UART1\_SR)*
- `#define _UART1_TC` ((uint8\_t) (0x01 << 6))  
*UART1 Transmission complete [0] (in \_UART1\_SR)*
- `#define _UART1_TXE` ((uint8\_t) (0x01 << 7))  
*UART1 Transmit data register empty [0] (in \_UART1\_SR)*
- `#define _UART1_PIEN` ((uint8\_t) (0x01 << 0))  
*UART1 Parity interrupt enable [0] (in \_UART1\_CR1)*
- `#define _UART1_PS` ((uint8\_t) (0x01 << 1))  
*UART1 Parity selection [0] (in \_UART1\_CR1)*
- `#define _UART1_PCEN` ((uint8\_t) (0x01 << 2))  
*UART1 Parity control enable [0] (in \_UART1\_CR1)*
- `#define _UART1_WAKE` ((uint8\_t) (0x01 << 3))  
*UART1 Wakeup method [0] (in \_UART1\_CR1)*
- `#define _UART1_M` ((uint8\_t) (0x01 << 4))  
*UART1 word length [0] (in \_UART1\_CR1)*
- `#define _UART1_UARTD` ((uint8\_t) (0x01 << 5))  
*UART1 Disable (for low power consumption) [0] (in \_UART1\_CR1)*
- `#define _UART1_T8` ((uint8\_t) (0x01 << 6))  
*UART1 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART1\_CR1)*
- `#define _UART1_R8` ((uint8\_t) (0x01 << 7))



- UART1 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART1\_CR1)*
  - #define `_UART1_SBK` ((uint8\_t) (0x01 << 0))
- UART1 Send break [0] (in \_UART1\_CR2)*
  - #define `_UART1_RWU` ((uint8\_t) (0x01 << 1))
- UART1 Receiver wakeup [0] (in \_UART1\_CR2)*
  - #define `_UART1_REN` ((uint8\_t) (0x01 << 2))
- UART1 Receiver enable [0] (in \_UART1\_CR2)*
  - #define `_UART1_TEN` ((uint8\_t) (0x01 << 3))
- UART1 Transmitter enable [0] (in \_UART1\_CR2)*
  - #define `_UART1_ILIEN` ((uint8\_t) (0x01 << 4))
- UART1 IDLE Line interrupt enable [0] (in \_UART1\_CR2)*
  - #define `_UART1_RIEN` ((uint8\_t) (0x01 << 5))
- UART1 Receiver interrupt enable [0] (in \_UART1\_CR2)*
  - #define `_UART1_TCIEN` ((uint8\_t) (0x01 << 6))
- UART1 Transmission complete interrupt enable [0] (in \_UART1\_CR2)*
  - #define `_UART1_TIEN` ((uint8\_t) (0x01 << 7))
- UART1 Transmitter interrupt enable [0] (in \_UART1\_CR2)*
  - #define `_UART1_LBCL` ((uint8\_t) (0x01 << 0))
- UART1 Last bit clock pulse [0] (in \_UART1\_CR3)*
  - #define `_UART1_CPHA` ((uint8\_t) (0x01 << 1))
- UART1 Clock phase [0] (in \_UART1\_CR3)*
  - #define `_UART1_CPOL` ((uint8\_t) (0x01 << 2))
- UART1 Clock polarity [0] (in \_UART1\_CR3)*
  - #define `_UART1_CKEN` ((uint8\_t) (0x01 << 3))
- UART1 Clock enable [0] (in \_UART1\_CR3)*
  - #define `_UART1_STOP` ((uint8\_t) (0x03 << 4))
- UART1 STOP bits [1:0] (in \_UART1\_CR3)*
  - #define `_UART1_STOP0` ((uint8\_t) (0x01 << 4))
- UART1 STOP bits [0] (in \_UART1\_CR3)*
  - #define `_UART1_STOP1` ((uint8\_t) (0x01 << 5))
- UART1 STOP bits [1] (in \_UART1\_CR3)*
  - #define `_UART1_LINEN` ((uint8\_t) (0x01 << 6))
- UART1 LIN mode enable [0] (in \_UART1\_CR3)*
  - #define `_UART1_ADD` ((uint8\_t) (0x0F << 0))
- UART1 Address of the UART node [3:0] (in \_UART1\_CR4)*
  - #define `_UART1_ADD0` ((uint8\_t) (0x01 << 0))
- UART1 Address of the UART node [0] (in \_UART1\_CR4)*
  - #define `_UART1_ADD1` ((uint8\_t) (0x01 << 1))
- UART1 Address of the UART node [1] (in \_UART1\_CR4)*
  - #define `_UART1_ADD2` ((uint8\_t) (0x01 << 2))
- UART1 Address of the UART node [2] (in \_UART1\_CR4)*
  - #define `_UART1_ADD3` ((uint8\_t) (0x01 << 3))
- UART1 Address of the UART node [3] (in \_UART1\_CR4)*
  - #define `_UART1_LBDF` ((uint8\_t) (0x01 << 4))
- UART1 LIN Break Detection Flag [0] (in \_UART1\_CR4)*
  - #define `_UART1_LBDL` ((uint8\_t) (0x01 << 5))
- UART1 LIN Break Detection Length [0] (in \_UART1\_CR4)*
  - #define `_UART1_LBDIEN` ((uint8\_t) (0x01 << 6))
- UART1 LIN Break Detection Interrupt Enable [0] (in \_UART1\_CR4)*
  - #define `_UART1_IREN` ((uint8\_t) (0x01 << 1))
- UART1 IrDA mode Enable [0] (in \_UART1\_CR5)*

- #define `_UART1_IRLP` ((uint8\_t) (0x01 << 2))  
*UART1 IrDA Low Power [0] (in \_UART1\_CR5)*
- #define `_UART1_HDSEL` ((uint8\_t) (0x01 << 3))  
*UART1 Half-Duplex Selection [0] (in \_UART1\_CR5)*
- #define `_UART1_NACK` ((uint8\_t) (0x01 << 4))  
*UART1 Smartcard NACK enable [0] (in \_UART1\_CR5)*
- #define `_UART1_SCEN` ((uint8\_t) (0x01 << 5))  
*UART1 Smartcard mode enable [0] (in \_UART1\_CR5)*
- #define `_UART2_SFR`(UART2\_t, UART2\_AddressBase)  
*UART2 struct/bit access.*
- #define `_UART2_SR_SFR`(uint8\_t, UART2\_AddressBase+0x00)  
*UART2 Status register.*
- #define `_UART2_DR_SFR`(uint8\_t, UART2\_AddressBase+0x01)  
*UART2 data register.*
- #define `_UART2_BRR1_SFR`(uint8\_t, UART2\_AddressBase+0x02)  
*UART2 Baud rate register 1.*
- #define `_UART2_BRR2_SFR`(uint8\_t, UART2\_AddressBase+0x03)  
*UART2 Baud rate register 2.*
- #define `_UART2_CR1_SFR`(uint8\_t, UART2\_AddressBase+0x04)  
*UART2 Control register 1.*
- #define `_UART2_CR2_SFR`(uint8\_t, UART2\_AddressBase+0x05)  
*UART2 Control register 2.*
- #define `_UART2_CR3_SFR`(uint8\_t, UART2\_AddressBase+0x06)  
*UART2 Control register 3.*
- #define `_UART2_CR4_SFR`(uint8\_t, UART2\_AddressBase+0x07)  
*UART2 Control register 4.*
- #define `_UART2_CR5_SFR`(uint8\_t, UART2\_AddressBase+0x08)  
*UART2 Control register 5.*
- #define `_UART2_CR6_SFR`(uint8\_t, UART2\_AddressBase+0x09)  
*UART2 Control register 6.*
- #define `_UART2_GTR_SFR`(uint8\_t, UART2\_AddressBase+0x0A)  
*UART2 guard time register.*
- #define `_UART2_PSCR_SFR`(uint8\_t, UART2\_AddressBase+0x0B)  
*UART2 prescaler register.*
- #define `_UART2_SR_RESET_VALUE` ((uint8\_t) 0xC0)  
*UART2 Status register reset value.*
- #define `_UART2_BRR1_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Baud rate register 1 reset value.*
- #define `_UART2_BRR2_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Baud rate register 2 reset value.*
- #define `_UART2_CR1_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 1 reset value.*
- #define `_UART2_CR2_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 2 reset value.*
- #define `_UART2_CR3_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 3 reset value.*
- #define `_UART2_CR4_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 4 reset value.*
- #define `_UART2_CR5_RESET_VALUE` ((uint8\_t) 0x00)  
*UART2 Control register 5 reset value.*
- #define `_UART2_CR6_RESET_VALUE` ((uint8\_t) 0x00)

- UART2 Control register 6 reset value.*
- #define `_UART2_GTR_RESET_VALUE` ((uint8\_t) 0x00)
- UART2 guard time register reset value.*
- #define `_UART2_PSCR_RESET_VALUE` ((uint8\_t) 0x00)
- UART2 prescaler register reset value.*
- #define `_UART2_PE` ((uint8\_t) (0x01 << 0))
- UART2 Parity error [0] (in \_UART2\_SR)*
- #define `_UART2_FE` ((uint8\_t) (0x01 << 1))
- UART2 Framing error [0] (in \_UART2\_SR)*
- #define `_UART2_NF` ((uint8\_t) (0x01 << 2))
- UART2 Noise flag [0] (in \_UART2\_SR)*
- #define `_UART2_OR_LHE` ((uint8\_t) (0x01 << 3))
- UART2 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART2\_SR)*
- #define `_UART2_IDLE` ((uint8\_t) (0x01 << 4))
- UART2 IDLE line detected [0] (in \_UART2\_SR)*
- #define `_UART2_RXNE` ((uint8\_t) (0x01 << 5))
- UART2 Read data register not empty [0] (in \_UART2\_SR)*
- #define `_UART2_TC` ((uint8\_t) (0x01 << 6))
- UART2 Transmission complete [0] (in \_UART2\_SR)*
- #define `_UART2_TXE` ((uint8\_t) (0x01 << 7))
- UART2 Transmit data register empty [0] (in \_UART2\_SR)*
- #define `_UART2_PIEN` ((uint8\_t) (0x01 << 0))
- UART2 Parity interrupt enable [0] (in \_UART2\_CR1)*
- #define `_UART2_PS` ((uint8\_t) (0x01 << 1))
- UART2 Parity selection [0] (in \_UART2\_CR1)*
- #define `_UART2_PCEN` ((uint8\_t) (0x01 << 2))
- UART2 Parity control enable [0] (in \_UART2\_CR1)*
- #define `_UART2_WAKE` ((uint8\_t) (0x01 << 3))
- UART2 Wakeup method [0] (in \_UART2\_CR1)*
- #define `_UART2_M` ((uint8\_t) (0x01 << 4))
- UART2 word length [0] (in \_UART2\_CR1)*
- #define `_UART2_UARTD` ((uint8\_t) (0x01 << 5))
- UART2 Disable (for low power consumption) [0] (in \_UART2\_CR1)*
- #define `_UART2_T8` ((uint8\_t) (0x01 << 6))
- UART2 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART2\_CR1)*
- #define `_UART2_R8` ((uint8\_t) (0x01 << 7))
- UART2 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART2\_CR1)*
- #define `_UART2_SBK` ((uint8\_t) (0x01 << 0))
- UART2 Send break [0] (in \_UART2\_CR2)*
- #define `_UART2_RWU` ((uint8\_t) (0x01 << 1))
- UART2 Receiver wakeup [0] (in \_UART2\_CR2)*
- #define `_UART2_REN` ((uint8\_t) (0x01 << 2))
- UART2 Receiver enable [0] (in \_UART2\_CR2)*
- #define `_UART2_TEN` ((uint8\_t) (0x01 << 3))
- UART2 Transmitter enable [0] (in \_UART2\_CR2)*
- #define `_UART2_ILIEN` ((uint8\_t) (0x01 << 4))
- UART2 IDLE Line interrupt enable [0] (in \_UART2\_CR2)*
- #define `_UART2_RIEN` ((uint8\_t) (0x01 << 5))
- UART2 Receiver interrupt enable [0] (in \_UART2\_CR2)*
- #define `_UART2_TCIEN` ((uint8\_t) (0x01 << 6))
- UART2 Transmission complete interrupt enable [0] (in \_UART2\_CR2)*

- `#define _UART2_TIEN` ((uint8\_t) (0x01 << 7))  
*UART2 Transmitter interrupt enable [0] (in \_UART2\_CR2)*
- `#define _UART2_LBCL` ((uint8\_t) (0x01 << 0))  
*UART2 Last bit clock pulse [0] (in \_UART2\_CR3)*
- `#define _UART2_CPHA` ((uint8\_t) (0x01 << 1))  
*UART2 Clock phase [0] (in \_UART2\_CR3)*
- `#define _UART2_CPOL` ((uint8\_t) (0x01 << 2))  
*UART2 Clock polarity [0] (in \_UART2\_CR3)*
- `#define _UART2_CKEN` ((uint8\_t) (0x01 << 3))  
*UART2 Clock enable [0] (in \_UART2\_CR3)*
- `#define _UART2_STOP` ((uint8\_t) (0x03 << 4))  
*UART2 STOP bits [1:0] (in \_UART2\_CR3)*
- `#define _UART2_STOP0` ((uint8\_t) (0x01 << 4))  
*UART2 STOP bits [0] (in \_UART2\_CR3)*
- `#define _UART2_STOP1` ((uint8\_t) (0x01 << 5))  
*UART2 STOP bits [1] (in \_UART2\_CR3)*
- `#define _UART2_LINEN` ((uint8\_t) (0x01 << 6))  
*UART2 LIN mode enable [0] (in \_UART2\_CR3)*
- `#define _UART2_ADD` ((uint8\_t) (0x0F << 0))  
*UART2 Address of the UART node [3:0] (in \_UART2\_CR4)*
- `#define _UART2_ADD0` ((uint8\_t) (0x01 << 0))  
*UART2 Address of the UART node [0] (in \_UART2\_CR4)*
- `#define _UART2_ADD1` ((uint8\_t) (0x01 << 1))  
*UART2 Address of the UART node [1] (in \_UART2\_CR4)*
- `#define _UART2_ADD2` ((uint8\_t) (0x01 << 2))  
*UART2 Address of the UART node [2] (in \_UART2\_CR4)*
- `#define _UART2_ADD3` ((uint8\_t) (0x01 << 3))  
*UART2 Address of the UART node [3] (in \_UART2\_CR4)*
- `#define _UART2_LBDF` ((uint8\_t) (0x01 << 4))  
*UART2 LIN Break Detection Flag [0] (in \_UART2\_CR4)*
- `#define _UART2_LBDL` ((uint8\_t) (0x01 << 5))  
*UART2 LIN Break Detection Length [0] (in \_UART2\_CR4)*
- `#define _UART2_LBDIEN` ((uint8\_t) (0x01 << 6))  
*UART2 LIN Break Detection Interrupt Enable [0] (in \_UART2\_CR4)*
- `#define _UART2_IREN` ((uint8\_t) (0x01 << 1))  
*UART2 IrDA mode Enable [0] (in \_UART2\_CR5)*
- `#define _UART2_IRLP` ((uint8\_t) (0x01 << 2))  
*UART2 IrDA Low Power [0] (in \_UART2\_CR5)*
- `#define _UART2_NACK` ((uint8\_t) (0x01 << 4))  
*UART2 Smartcard NACK enable [0] (in \_UART2\_CR5)*
- `#define _UART2_SCEN` ((uint8\_t) (0x01 << 5))  
*UART2 Smartcard mode enable [0] (in \_UART2\_CR5)*
- `#define _UART2_LSF` ((uint8\_t) (0x01 << 0))  
*UART2 LIN Sync Field [0] (in \_UART2\_CR6)*
- `#define _UART2_LHDF` ((uint8\_t) (0x01 << 1))  
*UART2 LIN Header Detection Flag [0] (in \_UART2\_CR6)*
- `#define _UART2_LHDIEN` ((uint8\_t) (0x01 << 2))  
*UART2 LIN Header Detection Interrupt Enable [0] (in \_UART2\_CR6)*
- `#define _UART2_LASE` ((uint8\_t) (0x01 << 4))  
*UART2 LIN automatic resynchronisation enable [0] (in \_UART2\_CR6)*
- `#define _UART2_LSLV` ((uint8\_t) (0x01 << 5))

- UART2 LIN Slave Enable [0] (in \_UART2\_CR6)*
- #define [\\_UART2\\_LDUM](#) ((uint8\_t) (0x01 << 7))
- UART2 LIN Divider Update Method [0] (in \_UART2\_CR6)*
- #define [\\_UART3\\_SFR](#)(UART3\_t, UART3\_AddressBase)
- UART3 struct/bit access.*
- #define [\\_UART3\\_SR\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x00)
- UART3 Status register.*
- #define [\\_UART3\\_DR\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x01)
- UART3 data register.*
- #define [\\_UART3\\_BRR1\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x02)
- UART3 Baud rate register 1.*
- #define [\\_UART3\\_BRR2\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x03)
- UART3 Baud rate register 2.*
- #define [\\_UART3\\_CR1\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x04)
- UART3 Control register 1.*
- #define [\\_UART3\\_CR2\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x05)
- UART3 Control register 2.*
- #define [\\_UART3\\_CR3\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x06)
- UART3 Control register 3.*
- #define [\\_UART3\\_CR4\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x07)
- UART3 Control register 4.*
- #define [\\_UART3\\_CR6\\_SFR](#)(uint8\_t, UART3\_AddressBase+0x09)
- UART3 Control register 6.*
- #define [\\_UART3\\_SR\\_RESET\\_VALUE](#) ((uint8\_t) 0xC0)
- UART3 Status register reset value.*
- #define [\\_UART3\\_BRR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- UART3 Baud rate register 1 reset value.*
- #define [\\_UART3\\_BRR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- UART3 Baud rate register 2 reset value.*
- #define [\\_UART3\\_CR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- UART3 Control register 1 reset value.*
- #define [\\_UART3\\_CR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- UART3 Control register 2 reset value.*
- #define [\\_UART3\\_CR3\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- UART3 Control register 3 reset value.*
- #define [\\_UART3\\_CR4\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- UART3 Control register 4 reset value.*
- #define [\\_UART3\\_CR6\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- UART3 Control register 6 reset value.*
- #define [\\_UART3\\_PE](#) ((uint8\_t) (0x01 << 0))
- UART3 Parity error [0] (in \_UART3\_SR)*
- #define [\\_UART3\\_FE](#) ((uint8\_t) (0x01 << 1))
- UART3 Framing error [0] (in \_UART3\_SR)*
- #define [\\_UART3\\_NF](#) ((uint8\_t) (0x01 << 2))
- UART3 Noise flag [0] (in \_UART3\_SR)*
- #define [\\_UART3\\_OR\\_LHE](#) ((uint8\_t) (0x01 << 3))
- UART3 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART3\_SR)*
- #define [\\_UART3\\_IDLE](#) ((uint8\_t) (0x01 << 4))
- UART3 IDLE line detected [0] (in \_UART3\_SR)*
- #define [\\_UART3\\_RXNE](#) ((uint8\_t) (0x01 << 5))
- UART3 Read data register not empty [0] (in \_UART3\_SR)*

- #define `_UART3_TC` ((uint8\_t) (0x01 << 6))  
*UART3 Transmission complete [0] (in \_UART3\_SR)*
- #define `_UART3_TXE` ((uint8\_t) (0x01 << 7))  
*UART3 Transmit data register empty [0] (in \_UART3\_SR)*
- #define `_UART3_PIEN` ((uint8\_t) (0x01 << 0))  
*UART3 Parity interrupt enable [0] (in \_UART3\_CR1)*
- #define `_UART3_PS` ((uint8\_t) (0x01 << 1))  
*UART3 Parity selection [0] (in \_UART3\_CR1)*
- #define `_UART3_PCEN` ((uint8\_t) (0x01 << 2))  
*UART3 Parity control enable [0] (in \_UART3\_CR1)*
- #define `_UART3_WAKE` ((uint8\_t) (0x01 << 3))  
*UART3 Wakeup method [0] (in \_UART3\_CR1)*
- #define `_UART3_M` ((uint8\_t) (0x01 << 4))  
*UART3 word length [0] (in \_UART3\_CR1)*
- #define `_UART3_UARTD` ((uint8\_t) (0x01 << 5))  
*UART3 Disable (for low power consumption) [0] (in \_UART3\_CR1)*
- #define `_UART3_T8` ((uint8\_t) (0x01 << 6))  
*UART3 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART3\_CR1)*
- #define `_UART3_R8` ((uint8\_t) (0x01 << 7))  
*UART3 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART3\_CR1)*
- #define `_UART3_SBK` ((uint8\_t) (0x01 << 0))  
*UART3 Send break [0] (in \_UART3\_CR2)*
- #define `_UART3_RWU` ((uint8\_t) (0x01 << 1))  
*UART3 Receiver wakeup [0] (in \_UART3\_CR2)*
- #define `_UART3_REN` ((uint8\_t) (0x01 << 2))  
*UART3 Receiver enable [0] (in \_UART3\_CR2)*
- #define `_UART3_TEN` ((uint8\_t) (0x01 << 3))  
*UART3 Transmitter enable [0] (in \_UART3\_CR2)*
- #define `_UART3_ILIEN` ((uint8\_t) (0x01 << 4))  
*UART3 IDLE Line interrupt enable [0] (in \_UART3\_CR2)*
- #define `_UART3_RIEN` ((uint8\_t) (0x01 << 5))  
*UART3 Receiver interrupt enable [0] (in \_UART3\_CR2)*
- #define `_UART3_TCIEN` ((uint8\_t) (0x01 << 6))  
*UART3 Transmission complete interrupt enable [0] (in \_UART3\_CR2)*
- #define `_UART3_TIEN` ((uint8\_t) (0x01 << 7))  
*UART3 Transmitter interrupt enable [0] (in \_UART3\_CR2)*
- #define `_UART3_STOP` ((uint8\_t) (0x03 << 4))  
*UART3 STOP bits [1:0] (in \_UART3\_CR3)*
- #define `_UART3_STOP0` ((uint8\_t) (0x01 << 4))  
*UART3 STOP bits [0] (in \_UART3\_CR3)*
- #define `_UART3_STOP1` ((uint8\_t) (0x01 << 5))  
*UART3 STOP bits [1] (in \_UART3\_CR3)*
- #define `_UART3_LINEN` ((uint8\_t) (0x01 << 6))  
*UART3 LIN mode enable [0] (in \_UART3\_CR3)*
- #define `_UART3_ADD` ((uint8\_t) (0x0F << 0))  
*UART3 Address of the UART node [3:0] (in \_UART3\_CR4)*
- #define `_UART3_ADD0` ((uint8\_t) (0x01 << 0))  
*UART3 Address of the UART node [0] (in \_UART3\_CR4)*
- #define `_UART3_ADD1` ((uint8\_t) (0x01 << 1))  
*UART3 Address of the UART node [1] (in \_UART3\_CR4)*
- #define `_UART3_ADD2` ((uint8\_t) (0x01 << 2))

- UART3 Address of the UART node [2] (in \_UART3\_CR4)*
- #define `_UART3_ADD3` ((uint8\_t) (0x01 << 3))
- UART3 Address of the UART node [3] (in \_UART3\_CR4)*
- #define `_UART3_LBDF` ((uint8\_t) (0x01 << 4))
- UART3 LIN Break Detection Flag [0] (in \_UART3\_CR4)*
- #define `_UART3_LBDL` ((uint8\_t) (0x01 << 5))
- UART3 LIN Break Detection Length [0] (in \_UART3\_CR4)*
- #define `_UART3_LBDIEN` ((uint8\_t) (0x01 << 6))
- UART3 LIN Break Detection Interrupt Enable [0] (in \_UART3\_CR4)*
- #define `_UART3_LSF` ((uint8\_t) (0x01 << 0))
- UART3 LIN Sync Field [0] (in \_UART3\_CR6)*
- #define `_UART3_LHDF` ((uint8\_t) (0x01 << 1))
- UART3 LIN Header Detection Flag [0] (in \_UART3\_CR6)*
- #define `_UART3_LHDIEN` ((uint8\_t) (0x01 << 2))
- UART3 LIN Header Detection Interrupt Enable [0] (in \_UART3\_CR6)*
- #define `_UART3_LASE` ((uint8\_t) (0x01 << 4))
- UART3 LIN automatic resynchronisation enable [0] (in \_UART3\_CR6)*
- #define `_UART3_LSLV` ((uint8\_t) (0x01 << 5))
- UART3 LIN Slave Enable [0] (in \_UART3\_CR6)*
- #define `_UART3_LDUM` ((uint8\_t) (0x01 << 7))
- UART3 LIN Divider Update Method [0] (in \_UART3\_CR6)*
- #define `_UART4_SFR`(UART4\_t, UART4\_AddressBase)
- UART4 struct/bit access.*
- #define `_UART4_SR_SFR`(uint8\_t, UART4\_AddressBase+0x00)
- UART4 Status register.*
- #define `_UART4_DR_SFR`(uint8\_t, UART4\_AddressBase+0x01)
- UART4 data register.*
- #define `_UART4_BRR1_SFR`(uint8\_t, UART4\_AddressBase+0x02)
- UART4 Baud rate register 1.*
- #define `_UART4_BRR2_SFR`(uint8\_t, UART4\_AddressBase+0x03)
- UART4 Baud rate register 2.*
- #define `_UART4_CR1_SFR`(uint8\_t, UART4\_AddressBase+0x04)
- UART4 Control register 1.*
- #define `_UART4_CR2_SFR`(uint8\_t, UART4\_AddressBase+0x05)
- UART4 Control register 2.*
- #define `_UART4_CR3_SFR`(uint8\_t, UART4\_AddressBase+0x06)
- UART4 Control register 3.*
- #define `_UART4_CR4_SFR`(uint8\_t, UART4\_AddressBase+0x07)
- UART4 Control register 4.*
- #define `_UART4_CR5_SFR`(uint8\_t, UART4\_AddressBase+0x08)
- UART4 Control register 5.*
- #define `_UART4_CR6_SFR`(uint8\_t, UART4\_AddressBase+0x09)
- UART4 Control register 6.*
- #define `_UART4_GTR_SFR`(uint8\_t, UART4\_AddressBase+0x0A)
- UART4 guard time register.*
- #define `_UART4_PSCR_SFR`(uint8\_t, UART4\_AddressBase+0x0B)
- UART4 prescaler register.*
- #define `_UART4_SR_RESET_VALUE` ((uint8\_t) 0xC0)
- UART4 Status register reset value.*
- #define `_UART4_BRR1_RESET_VALUE` ((uint8\_t) 0x00)
- UART4 Baud rate register 1 reset value.*



- `#define _UART4_BRR2_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 Baud rate register 2 reset value.*
- `#define _UART4_CR1_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 Control register 1 reset value.*
- `#define _UART4_CR2_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 Control register 2 reset value.*
- `#define _UART4_CR3_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 Control register 3 reset value.*
- `#define _UART4_CR4_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 Control register 4 reset value.*
- `#define _UART4_CR5_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 Control register 5 reset value.*
- `#define _UART4_CR6_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 Control register 6 reset value.*
- `#define _UART4_GTR_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 guard time register reset value.*
- `#define _UART4_PSCR_RESET_VALUE` ((uint8\_t) 0x00)  
*UART4 prescaler register reset value.*
- `#define _UART4_PE` ((uint8\_t) (0x01 << 0))  
*UART4 Parity error [0] (in \_UART4\_SR)*
- `#define _UART4_FE` ((uint8\_t) (0x01 << 1))  
*UART4 Framing error [0] (in \_UART4\_SR)*
- `#define _UART4_NF` ((uint8\_t) (0x01 << 2))  
*UART4 Noise flag [0] (in \_UART4\_SR)*
- `#define _UART4_OR_LHE` ((uint8\_t) (0x01 << 3))  
*UART4 LIN Header Error (LIN slave mode) / Overrun error [0] (in \_UART4\_SR)*
- `#define _UART4_IDLE` ((uint8\_t) (0x01 << 4))  
*UART4 IDLE line detected [0] (in \_UART4\_SR)*
- `#define _UART4_RXNE` ((uint8\_t) (0x01 << 5))  
*UART4 Read data register not empty [0] (in \_UART4\_SR)*
- `#define _UART4_TC` ((uint8\_t) (0x01 << 6))  
*UART4 Transmission complete [0] (in \_UART4\_SR)*
- `#define _UART4_TXE` ((uint8\_t) (0x01 << 7))  
*UART4 Transmit data register empty [0] (in \_UART4\_SR)*
- `#define _UART4_PIEN` ((uint8\_t) (0x01 << 0))  
*UART4 Parity interrupt enable [0] (in \_UART4\_CR1)*
- `#define _UART4_PS` ((uint8\_t) (0x01 << 1))  
*UART4 Parity selection [0] (in \_UART4\_CR1)*
- `#define _UART4_PCEN` ((uint8\_t) (0x01 << 2))  
*UART4 Parity control enable [0] (in \_UART4\_CR1)*
- `#define _UART4_WAKE` ((uint8\_t) (0x01 << 3))  
*UART4 Wakeup method [0] (in \_UART4\_CR1)*
- `#define _UART4_M` ((uint8\_t) (0x01 << 4))  
*UART4 word length [0] (in \_UART4\_CR1)*
- `#define _UART4_UARTD` ((uint8\_t) (0x01 << 5))  
*UART4 Disable (for low power consumption) [0] (in \_UART4\_CR1)*
- `#define _UART4_T8` ((uint8\_t) (0x01 << 6))  
*UART4 Transmit Data bit 8 (in 9-bit mode) [0] (in \_UART4\_CR1)*
- `#define _UART4_R8` ((uint8\_t) (0x01 << 7))  
*UART4 Receive Data bit 8 (in 9-bit mode) [0] (in \_UART4\_CR1)*
- `#define _UART4_SBK` ((uint8\_t) (0x01 << 0))



- UART4 Send break [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_RWU](#) ((uint8\_t) (0x01 << 1))
- UART4 Receiver wakeup [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_REN](#) ((uint8\_t) (0x01 << 2))
- UART4 Receiver enable [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_TEN](#) ((uint8\_t) (0x01 << 3))
- UART4 Transmitter enable [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_ILIEN](#) ((uint8\_t) (0x01 << 4))
- UART4 IDLE Line interrupt enable [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_RIEN](#) ((uint8\_t) (0x01 << 5))
- UART4 Receiver interrupt enable [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_TCIEN](#) ((uint8\_t) (0x01 << 6))
- UART4 Transmission complete interrupt enable [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_TIEN](#) ((uint8\_t) (0x01 << 7))
- UART4 Transmitter interrupt enable [0] (in \_UART4\_CR2)*
- #define [\\_UART4\\_LBCL](#) ((uint8\_t) (0x01 << 0))
- UART4 Last bit clock pulse [0] (in \_UART4\_CR3)*
- #define [\\_UART4\\_CPHA](#) ((uint8\_t) (0x01 << 1))
- UART4 Clock phase [0] (in \_UART4\_CR3)*
- #define [\\_UART4\\_CPOL](#) ((uint8\_t) (0x01 << 2))
- UART4 Clock polarity [0] (in \_UART4\_CR3)*
- #define [\\_UART4\\_CKEN](#) ((uint8\_t) (0x01 << 3))
- UART4 Clock enable [0] (in \_UART4\_CR3)*
- #define [\\_UART4\\_STOP](#) ((uint8\_t) (0x03 << 4))
- UART4 STOP bits [1:0] (in \_UART4\_CR3)*
- #define [\\_UART4\\_STOP0](#) ((uint8\_t) (0x01 << 4))
- UART4 STOP bits [0] (in \_UART4\_CR3)*
- #define [\\_UART4\\_STOP1](#) ((uint8\_t) (0x01 << 5))
- UART4 STOP bits [1] (in \_UART4\_CR3)*
- #define [\\_UART4\\_LINEN](#) ((uint8\_t) (0x01 << 6))
- UART4 LIN mode enable [0] (in \_UART4\_CR3)*
- #define [\\_UART4\\_ADD](#) ((uint8\_t) (0x0F << 0))
- UART4 Address of the UART node [3:0] (in \_UART4\_CR4)*
- #define [\\_UART4\\_ADD0](#) ((uint8\_t) (0x01 << 0))
- UART4 Address of the UART node [0] (in \_UART4\_CR4)*
- #define [\\_UART4\\_ADD1](#) ((uint8\_t) (0x01 << 1))
- UART4 Address of the UART node [1] (in \_UART4\_CR4)*
- #define [\\_UART4\\_ADD2](#) ((uint8\_t) (0x01 << 2))
- UART4 Address of the UART node [2] (in \_UART4\_CR4)*
- #define [\\_UART4\\_ADD3](#) ((uint8\_t) (0x01 << 3))
- UART4 Address of the UART node [3] (in \_UART4\_CR4)*
- #define [\\_UART4\\_LBDF](#) ((uint8\_t) (0x01 << 4))
- UART4 LIN Break Detection Flag [0] (in \_UART4\_CR4)*
- #define [\\_UART4\\_LBDL](#) ((uint8\_t) (0x01 << 5))
- UART4 LIN Break Detection Length [0] (in \_UART4\_CR4)*
- #define [\\_UART4\\_LBDIEN](#) ((uint8\_t) (0x01 << 6))
- UART4 LIN Break Detection Interrupt Enable [0] (in \_UART4\_CR4)*
- #define [\\_UART4\\_IREN](#) ((uint8\_t) (0x01 << 1))
- UART4 IrDA mode Enable [0] (in \_UART4\_CR5)*
- #define [\\_UART4\\_IRLP](#) ((uint8\_t) (0x01 << 2))
- UART4 IrDA Low Power [0] (in \_UART4\_CR5)*

- #define `_UART4_HDSEL` ((uint8\_t) (0x01 << 3))  
*UART4 Half-Duplex Selection [0] (in \_UART4\_CR5)*
- #define `_UART4_NACK` ((uint8\_t) (0x01 << 4))  
*UART4 Smartcard NACK enable [0] (in \_UART4\_CR5)*
- #define `_UART4_SCEN` ((uint8\_t) (0x01 << 5))  
*UART4 Smartcard mode enable [0] (in \_UART4\_CR5)*
- #define `_UART4_LSF` ((uint8\_t) (0x01 << 0))  
*UART4 LIN Sync Field [0] (in \_UART4\_CR6)*
- #define `_UART4_LHDF` ((uint8\_t) (0x01 << 1))  
*UART4 LIN Header Detection Flag [0] (in \_UART4\_CR6)*
- #define `_UART4_LHDIEN` ((uint8\_t) (0x01 << 2))  
*UART4 LIN Header Detection Interrupt Enable [0] (in \_UART4\_CR6)*
- #define `_UART4_LASE` ((uint8\_t) (0x01 << 4))  
*UART4 LIN automatic resynchronisation enable [0] (in \_UART4\_CR6)*
- #define `_UART4_LSLV` ((uint8\_t) (0x01 << 5))  
*UART4 LIN Slave Enable [0] (in \_UART4\_CR6)*
- #define `_UART4_LDUM` ((uint8\_t) (0x01 << 7))  
*UART4 LIN Divider Update Method [0] (in \_UART4\_CR6)*
- #define `_TIM1_SFR`(TIM1\_t, TIM1\_AddressBase)  
*TIM1 struct/bit access.*
- #define `_TIM1_CR1_SFR`(uint8\_t, TIM1\_AddressBase+0x00)  
*TIM1 control register 1.*
- #define `_TIM1_CR2_SFR`(uint8\_t, TIM1\_AddressBase+0x01)  
*TIM1 control register 2.*
- #define `_TIM1_SMCR_SFR`(uint8\_t, TIM1\_AddressBase+0x02)  
*TIM1 Slave mode control register.*
- #define `_TIM1_ETR_SFR`(uint8\_t, TIM1\_AddressBase+0x03)  
*TIM1 External trigger register.*
- #define `_TIM1_IER_SFR`(uint8\_t, TIM1\_AddressBase+0x04)  
*TIM1 interrupt enable register.*
- #define `_TIM1_SR1_SFR`(uint8\_t, TIM1\_AddressBase+0x05)  
*TIM1 status register 1.*
- #define `_TIM1_SR2_SFR`(uint8\_t, TIM1\_AddressBase+0x06)  
*TIM1 status register 2.*
- #define `_TIM1_EGR_SFR`(uint8\_t, TIM1\_AddressBase+0x07)  
*TIM1 Event generation register.*
- #define `_TIM1_CCMR1_SFR`(uint8\_t, TIM1\_AddressBase+0x08)  
*TIM1 Capture/compare mode register 1.*
- #define `_TIM1_CCMR2_SFR`(uint8\_t, TIM1\_AddressBase+0x09)  
*TIM1 Capture/compare mode register 2.*
- #define `_TIM1_CCMR3_SFR`(uint8\_t, TIM1\_AddressBase+0x0A)  
*TIM1 Capture/compare mode register 3.*
- #define `_TIM1_CCMR4_SFR`(uint8\_t, TIM1\_AddressBase+0x0B)  
*TIM1 Capture/compare mode register 4.*
- #define `_TIM1_CCER1_SFR`(uint8\_t, TIM1\_AddressBase+0x0C)  
*TIM1 Capture/compare enable register 1.*
- #define `_TIM1_CCER2_SFR`(uint8\_t, TIM1\_AddressBase+0x0D)  
*TIM1 Capture/compare enable register 2.*
- #define `_TIM1_CNTRH_SFR`(uint8\_t, TIM1\_AddressBase+0x0E)  
*TIM1 counter register high byte.*
- #define `_TIM1_CNTRL_SFR`(uint8\_t, TIM1\_AddressBase+0x0F)

- TIM1 counter register low byte.*
- #define [\\_TIM1\\_PSCRH\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x10)
- TIM1 clock prescaler register high byte.*
- #define [\\_TIM1\\_PSCRL\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x11)
- TIM1 clock prescaler register low byte.*
- #define [\\_TIM1\\_ARRH\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x12)
- TIM1 auto-reload register high byte.*
- #define [\\_TIM1\\_ARRL\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x13)
- TIM1 auto-reload register low byte.*
- #define [\\_TIM1\\_RCR\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x14)
- TIM1 Repetition counter.*
- #define [\\_TIM1\\_CCR1H\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x15)
- TIM1 16-bit capture/compare value 1 high byte.*
- #define [\\_TIM1\\_CCR1L\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x16)
- TIM1 16-bit capture/compare value 1 low byte.*
- #define [\\_TIM1\\_CCR2H\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x17)
- TIM1 16-bit capture/compare value 2 high byte.*
- #define [\\_TIM1\\_CCR2L\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x18)
- TIM1 16-bit capture/compare value 2 low byte.*
- #define [\\_TIM1\\_CCR3H\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x19)
- TIM1 16-bit capture/compare value 3 high byte.*
- #define [\\_TIM1\\_CCR3L\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x1A)
- TIM1 16-bit capture/compare value 3 low byte.*
- #define [\\_TIM1\\_CCR4H\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x1B)
- TIM1 16-bit capture/compare value 4 high byte.*
- #define [\\_TIM1\\_CCR4L\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x1C)
- TIM1 16-bit capture/compare value 4 low byte.*
- #define [\\_TIM1\\_BKR\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x1D)
- TIM1 Break register.*
- #define [\\_TIM1\\_DTR\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x1E)
- TIM1 Dead-time register.*
- #define [\\_TIM1\\_OISR\\_SFR](#)(uint8\_t, [TIM1\\_AddressBase](#)+0x1F)
- TIM1 Output idle state register.*
- #define [\\_TIM1\\_CR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 control register 1 reset value.*
- #define [\\_TIM1\\_CR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 control register 2 reset value.*
- #define [\\_TIM1\\_SMCR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 Slave mode control register reset value.*
- #define [\\_TIM1\\_ETR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 External trigger register reset value.*
- #define [\\_TIM1\\_IER\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 interrupt enable register reset value.*
- #define [\\_TIM1\\_SR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 status register 1 reset value.*
- #define [\\_TIM1\\_SR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 status register 2 reset value.*
- #define [\\_TIM1\\_EGR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 Event generation register reset value.*
- #define [\\_TIM1\\_CCMR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM1 Capture/compare mode register 1 reset value.*

- `#define _TIM1_CCMR2_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Capture/compare mode register 2 reset value.*
- `#define _TIM1_CCMR3_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Capture/compare mode register 3 reset value.*
- `#define _TIM1_CCMR4_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Capture/compare mode register 4 reset value.*
- `#define _TIM1_CCER1_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Capture/compare enable register 1 reset value.*
- `#define _TIM1_CCER2_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Capture/compare enable register 2 reset value.*
- `#define _TIM1_CNTRH_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 counter register high byte reset value.*
- `#define _TIM1_CNTRL_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 counter register low byte reset value.*
- `#define _TIM1_PSCRH_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 clock prescaler register high byte reset value.*
- `#define _TIM1_PSCRL_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 clock prescaler register low byte reset value.*
- `#define _TIM1_ARRH_RESET_VALUE` ((uint8\_t) 0xFF)  
*TIM1 auto-reload register high byte reset value.*
- `#define _TIM1_ARRL_RESET_VALUE` ((uint8\_t) 0xFF)  
*TIM1 auto-reload register low byte reset value.*
- `#define _TIM1_RCR_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Repetition counter reset value.*
- `#define _TIM1_CCR1H_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 1 high byte reset value.*
- `#define _TIM1_CCR1L_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 1 low byte reset value.*
- `#define _TIM1_CCR2H_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 2 high byte reset value.*
- `#define _TIM1_CCR2L_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 2 low byte reset value.*
- `#define _TIM1_CCR3H_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 3 high byte reset value.*
- `#define _TIM1_CCR3L_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 3 low byte reset value.*
- `#define _TIM1_CCR4H_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 4 high byte reset value.*
- `#define _TIM1_CCR4L_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 16-bit capture/compare value 4 low byte reset value.*
- `#define _TIM1_BKR_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Break register reset value.*
- `#define _TIM1_DTR_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Dead-time register reset value.*
- `#define _TIM1_OISR_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM1 Output idle state register reset value.*
- `#define _TIM1_CEN` ((uint8\_t) (0x01 << 0))  
*TIM1 Counter enable [0] (in \_TIM1\_CR1)*
- `#define _TIM1_UDIS` ((uint8\_t) (0x01 << 1))  
*TIM1 Update disable [0] (in \_TIM1\_CR1)*
- `#define _TIM1_URS` ((uint8\_t) (0x01 << 2))

- TIM1 Update request source [0] (in \_TIM1\_CR1)*
- #define `_TIM1_OPM` ((uint8\_t) (0x01 << 3))
- TIM1 One-pulse mode [0] (in \_TIM1\_CR1)*
- #define `_TIM1_DIR` ((uint8\_t) (0x01 << 4))
- TIM1 Direction [0] (in \_TIM1\_CR1)*
- #define `_TIM1_CMS` ((uint8\_t) (0x03 << 5))
- TIM1 Center-aligned mode selection [1:0] (in \_TIM1\_CR1)*
- #define `_TIM1_CMS0` ((uint8\_t) (0x01 << 5))
- TIM1 Center-aligned mode selection [0] (in \_TIM1\_CR1)*
- #define `_TIM1_CMS1` ((uint8\_t) (0x01 << 6))
- TIM1 Center-aligned mode selection [1] (in \_TIM1\_CR1)*
- #define `_TIM1_ARPE` ((uint8\_t) (0x01 << 7))
- TIM1 Auto-reload preload enable [0] (in \_TIM1\_CR1)*
- #define `_TIM1_CCPC` ((uint8\_t) (0x01 << 0))
- TIM1 Capture/compare preloaded control [0] (in \_TIM1\_CR2)*
- #define `_TIM1_COMS` ((uint8\_t) (0x01 << 2))
- TIM1 Capture/compare control update selection [0] (in \_TIM1\_CR2)*
- #define `_TIM1_MMS` ((uint8\_t) (0x07 << 4))
- TIM1 Master mode selection [2:0] (in \_TIM1\_CR2)*
- #define `_TIM1_MMS0` ((uint8\_t) (0x01 << 4))
- TIM1 Master mode selection [0] (in \_TIM1\_CR2)*
- #define `_TIM1_MMS1` ((uint8\_t) (0x01 << 5))
- TIM1 Master mode selection [1] (in \_TIM1\_CR2)*
- #define `_TIM1_MMS2` ((uint8\_t) (0x01 << 6))
- TIM1 Master mode selection [2] (in \_TIM1\_CR2)*
- #define `_TIM1_SMS` ((uint8\_t) (0x07 << 0))
- TIM1 Clock/trigger/slave mode selection [2:0] (in \_TIM1\_SMCR)*
- #define `_TIM1_SMS0` ((uint8\_t) (0x01 << 0))
- TIM1 Clock/trigger/slave mode selection [0] (in \_TIM1\_SMCR)*
- #define `_TIM1_SMS1` ((uint8\_t) (0x01 << 1))
- TIM1 Clock/trigger/slave mode selection [1] (in \_TIM1\_SMCR)*
- #define `_TIM1_SMS2` ((uint8\_t) (0x01 << 2))
- TIM1 Clock/trigger/slave mode selection [2] (in \_TIM1\_SMCR)*
- #define `_TIM1_TS` ((uint8\_t) (0x07 << 4))
- TIM1 Trigger selection [2:0] (in \_TIM1\_SMCR)*
- #define `_TIM1_TS0` ((uint8\_t) (0x01 << 4))
- TIM1 Trigger selection [0] (in \_TIM1\_SMCR)*
- #define `_TIM1_TS1` ((uint8\_t) (0x01 << 5))
- TIM1 Trigger selection [1] (in \_TIM1\_SMCR)*
- #define `_TIM1_TS2` ((uint8\_t) (0x01 << 6))
- TIM1 Trigger selection [2] (in \_TIM1\_SMCR)*
- #define `_TIM1_MSM` ((uint8\_t) (0x01 << 7))
- TIM1 Master/slave mode [0] (in \_TIM1\_SMCR)*
- #define `_TIM1_ETF` ((uint8\_t) (0x0F << 0))
- TIM1 External trigger filter [3:0] (in \_TIM1\_ETR)*
- #define `_TIM1_ETF0` ((uint8\_t) (0x01 << 0))
- TIM1 External trigger filter [0] (in \_TIM1\_ETR)*
- #define `_TIM1_ETF1` ((uint8\_t) (0x01 << 1))
- TIM1 External trigger filter [1] (in \_TIM1\_ETR)*
- #define `_TIM1_ETF2` ((uint8\_t) (0x01 << 2))
- TIM1 External trigger filter [2] (in \_TIM1\_ETR)*

- `#define _TIM1_ETF3 ((uint8_t) (0x01 << 3))`  
*TIM1 External trigger filter [3] (in \_TIM1\_ETR)*
- `#define _TIM1_ETPS ((uint8_t) (0x03 << 4))`  
*TIM1 External trigger prescaler [1:0] (in \_TIM1\_ETR)*
- `#define _TIM1_ETPS0 ((uint8_t) (0x01 << 4))`  
*TIM1 External trigger prescaler [0] (in \_TIM1\_ETR)*
- `#define _TIM1_ETPS1 ((uint8_t) (0x01 << 5))`  
*TIM1 External trigger prescaler [1] (in \_TIM1\_ETR)*
- `#define _TIM1_ECE ((uint8_t) (0x01 << 6))`  
*TIM1 External clock enable [0] (in \_TIM1\_ETR)*
- `#define _TIM1_ETP ((uint8_t) (0x01 << 7))`  
*TIM1 External trigger polarity [0] (in \_TIM1\_ETR)*
- `#define _TIM1_UIE ((uint8_t) (0x01 << 0))`  
*TIM1 Update interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_CC1IE ((uint8_t) (0x01 << 1))`  
*TIM1 Capture/compare 1 interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_CC2IE ((uint8_t) (0x01 << 2))`  
*TIM1 Capture/compare 2 interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_CC3IE ((uint8_t) (0x01 << 3))`  
*TIM1 Capture/compare 3 interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_CC4IE ((uint8_t) (0x01 << 4))`  
*TIM1 Capture/compare 4 interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_COMIE ((uint8_t) (0x01 << 5))`  
*TIM1 Commutation interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_TIE ((uint8_t) (0x01 << 6))`  
*TIM1 Trigger interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_BIE ((uint8_t) (0x01 << 7))`  
*TIM1 Break interrupt enable [0] (in \_TIM1\_IER)*
- `#define _TIM1_UIF ((uint8_t) (0x01 << 0))`  
*TIM1 Update interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_CC1IF ((uint8_t) (0x01 << 1))`  
*TIM1 Capture/compare 1 interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_CC2IF ((uint8_t) (0x01 << 2))`  
*TIM1 Capture/compare 2 interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_CC3IF ((uint8_t) (0x01 << 3))`  
*TIM1 Capture/compare 3 interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_CC4IF ((uint8_t) (0x01 << 4))`  
*TIM1 Capture/compare 4 interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_COMIF ((uint8_t) (0x01 << 5))`  
*TIM1 Commutation interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_TIF ((uint8_t) (0x01 << 6))`  
*TIM1 Trigger interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_BIF ((uint8_t) (0x01 << 7))`  
*TIM1 Break interrupt flag [0] (in \_TIM1\_SR1)*
- `#define _TIM1_CC1OF ((uint8_t) (0x01 << 1))`  
*TIM1 Capture/compare 1 overcapture flag [0] (in \_TIM1\_SR2)*
- `#define _TIM1_CC2OF ((uint8_t) (0x01 << 2))`  
*TIM1 Capture/compare 2 overcapture flag [0] (in \_TIM1\_SR2)*
- `#define _TIM1_CC3OF ((uint8_t) (0x01 << 3))`  
*TIM1 Capture/compare 3 overcapture flag [0] (in \_TIM1\_SR2)*
- `#define _TIM1_CC4OF ((uint8_t) (0x01 << 4))`

- TIM1 Capture/compare 4 overcapture flag [0] (in \_TIM1\_SR2)*
  - #define `_TIM1_UG` ((uint8\_t) (0x01 << 0))
- TIM1 Update generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_CC1G` ((uint8\_t) (0x01 << 1))
- TIM1 Capture/compare 1 generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_CC2G` ((uint8\_t) (0x01 << 2))
- TIM1 Capture/compare 2 generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_CC3G` ((uint8\_t) (0x01 << 3))
- TIM1 Capture/compare 3 generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_CC4G` ((uint8\_t) (0x01 << 4))
- TIM1 Capture/compare 4 generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_COMG` ((uint8\_t) (0x01 << 5))
- TIM1 Capture/compare control update generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_TG` ((uint8\_t) (0x01 << 6))
- TIM1 Trigger generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_BG` ((uint8\_t) (0x01 << 7))
- TIM1 Break generation [0] (in \_TIM1\_EGR)*
  - #define `_TIM1_CC1S` ((uint8\_t) (0x03 << 0))
- TIM1 Compare 1 selection [1:0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_CC1S0` ((uint8\_t) (0x01 << 0))
- TIM1 Compare 1 selection [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_CC1S1` ((uint8\_t) (0x01 << 1))
- TIM1 Compare 1 selection [1] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_OC1FE` ((uint8\_t) (0x01 << 2))
- TIM1 Output compare 1 fast enable [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_OC1PE` ((uint8\_t) (0x01 << 3))
- TIM1 Output compare 1 preload enable [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_OC1M` ((uint8\_t) (0x07 << 4))
- TIM1 Output compare 1 mode [2:0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_OC1M0` ((uint8\_t) (0x01 << 4))
- TIM1 Output compare 1 mode [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_OC1M1` ((uint8\_t) (0x01 << 5))
- TIM1 Output compare 1 mode [1] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_OC1M2` ((uint8\_t) (0x01 << 6))
- TIM1 Output compare 1 mode [2] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_OC1CE` ((uint8\_t) (0x01 << 7))
- TIM1 Output compare 1 clear enable [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1PSC` ((uint8\_t) (0x03 << 2))
- TIM1 Input capture 1 prescaler [1:0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1PSC0` ((uint8\_t) (0x01 << 2))
- TIM1 Input capture 1 prescaler [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1PSC1` ((uint8\_t) (0x01 << 3))
- TIM1 Input capture 1 prescaler [1] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F` ((uint8\_t) (0x0F << 4))
- TIM1 Output compare 1 mode [3:0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F0` ((uint8\_t) (0x01 << 4))
- TIM1 Input capture 1 filter [0] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F1` ((uint8\_t) (0x01 << 5))
- TIM1 Input capture 1 filter [1] (in \_TIM1\_CCMR1)*
  - #define `_TIM1_IC1F2` ((uint8\_t) (0x01 << 6))
- TIM1 Input capture 1 filter [2] (in \_TIM1\_CCMR1)*



- #define `_TIM1_IC1F3` ((uint8\_t) (0x01 << 7))  
*TIM1 Input capture 1 filter [3] (in \_TIM1\_CCMR1)*
- #define `_TIM1_CC2S` ((uint8\_t) (0x03 << 0))  
*TIM1 Compare 2 selection [1:0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_CC2S0` ((uint8\_t) (0x01 << 0))  
*TIM1 Compare 2 selection [0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_CC2S1` ((uint8\_t) (0x01 << 1))  
*TIM1 Compare 2 selection [1] (in \_TIM1\_CCMR2)*
- #define `_TIM1_OC2FE` ((uint8\_t) (0x01 << 2))  
*TIM1 Output compare 2 fast enable [0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_OC2PE` ((uint8\_t) (0x01 << 3))  
*TIM1 Output compare 2 preload enable [0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_OC2M` ((uint8\_t) (0x07 << 4))  
*TIM1 Output compare 2 mode [2:0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_OC2M0` ((uint8\_t) (0x01 << 4))  
*TIM1 Output compare 2 mode [0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_OC2M1` ((uint8\_t) (0x01 << 5))  
*TIM1 Output compare 2 mode [1] (in \_TIM1\_CCMR2)*
- #define `_TIM1_OC2M2` ((uint8\_t) (0x01 << 6))  
*TIM1 Output compare 2 mode [2] (in \_TIM1\_CCMR2)*
- #define `_TIM1_OC2CE` ((uint8\_t) (0x01 << 7))  
*TIM1 Output compare 2 clear enable [0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2PSC` ((uint8\_t) (0x03 << 2))  
*TIM1 Input capture 2 prescaler [1:0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2PSC0` ((uint8\_t) (0x01 << 2))  
*TIM1 Input capture 2 prescaler [0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2PSC1` ((uint8\_t) (0x01 << 3))  
*TIM1 Input capture 2 prescaler [1] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2F` ((uint8\_t) (0x0F << 4))  
*TIM1 Output compare 2 mode [3:0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2F0` ((uint8\_t) (0x01 << 4))  
*TIM1 Input capture 2 filter [0] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2F1` ((uint8\_t) (0x01 << 5))  
*TIM1 Input capture 2 filter [1] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2F2` ((uint8\_t) (0x01 << 6))  
*TIM1 Input capture 2 filter [2] (in \_TIM1\_CCMR2)*
- #define `_TIM1_IC2F3` ((uint8\_t) (0x01 << 7))  
*TIM1 Input capture 2 filter [3] (in \_TIM1\_CCMR2)*
- #define `_TIM1_CC3S` ((uint8\_t) (0x03 << 0))  
*TIM1 Compare 3 selection [1:0] (in \_TIM1\_CCMR3)*
- #define `_TIM1_CC3S0` ((uint8\_t) (0x01 << 0))  
*TIM1 Compare 3 selection [0] (in \_TIM1\_CCMR3)*
- #define `_TIM1_CC3S1` ((uint8\_t) (0x01 << 1))  
*TIM1 Compare 3 selection [1] (in \_TIM1\_CCMR3)*
- #define `_TIM1_OC3FE` ((uint8\_t) (0x01 << 2))  
*TIM1 Output compare 3 fast enable [0] (in \_TIM1\_CCMR3)*
- #define `_TIM1_OC3PE` ((uint8\_t) (0x01 << 3))  
*TIM1 Output compare 3 preload enable [0] (in \_TIM1\_CCMR3)*
- #define `_TIM1_OC3M` ((uint8\_t) (0x07 << 4))  
*TIM1 Output compare 3 mode [2:0] (in \_TIM1\_CCMR3)*
- #define `_TIM1_OC3M0` ((uint8\_t) (0x01 << 4))



- TIM1 Output compare 3 mode [0] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_OC3M1](#) ((uint8\_t) (0x01 << 5))
- TIM1 Output compare 3 mode [1] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_OC3M2](#) ((uint8\_t) (0x01 << 6))
- TIM1 Output compare 3 mode [2] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_OC3CE](#) ((uint8\_t) (0x01 << 7))
- TIM1 Output compare 3 clear enable [0] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3PSC](#) ((uint8\_t) (0x03 << 2))
- TIM1 Input capture 3 prescaler [1:0] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3PSC0](#) ((uint8\_t) (0x01 << 2))
- TIM1 Input capture 3 prescaler [0] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3PSC1](#) ((uint8\_t) (0x01 << 3))
- TIM1 Input capture 3 prescaler [1] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3F](#) ((uint8\_t) (0x0F << 4))
- TIM1 Output compare 3 mode [3:0] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3F0](#) ((uint8\_t) (0x01 << 4))
- TIM1 Input capture 3 filter [0] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3F1](#) ((uint8\_t) (0x01 << 5))
- TIM1 Input capture 3 filter [1] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3F2](#) ((uint8\_t) (0x01 << 6))
- TIM1 Input capture 3 filter [2] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_IC3F3](#) ((uint8\_t) (0x01 << 7))
- TIM1 Input capture 3 filter [3] (in \_TIM1\_CCMR3)*
- #define [\\_TIM1\\_CC4S](#) ((uint8\_t) (0x03 << 0))
- TIM1 Compare 4 selection [1:0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_CC4S0](#) ((uint8\_t) (0x01 << 0))
- TIM1 Compare 4 selection [0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_CC4S1](#) ((uint8\_t) (0x01 << 1))
- TIM1 Compare 4 selection [1] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_OC4FE](#) ((uint8\_t) (0x01 << 2))
- TIM1 Output compare 4 fast enable [0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_OC4PE](#) ((uint8\_t) (0x01 << 3))
- TIM1 Output compare 4 preload enable [0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_OC4M](#) ((uint8\_t) (0x07 << 4))
- TIM1 Output compare 4 mode [2:0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_OC4M0](#) ((uint8\_t) (0x01 << 4))
- TIM1 Output compare 4 mode [0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_OC4M1](#) ((uint8\_t) (0x01 << 5))
- TIM1 Output compare 4 mode [1] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_OC4M2](#) ((uint8\_t) (0x01 << 6))
- TIM1 Output compare 4 mode [2] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_OC4CE](#) ((uint8\_t) (0x01 << 7))
- TIM1 Output compare 4 clear enable [0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_IC4PSC](#) ((uint8\_t) (0x03 << 2))
- TIM1 Input capture 4 prescaler [1:0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_IC4PSC0](#) ((uint8\_t) (0x01 << 2))
- TIM1 Input capture 4 prescaler [0] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_IC4PSC1](#) ((uint8\_t) (0x01 << 3))
- TIM1 Input capture 4 prescaler [1] (in \_TIM1\_CCMR4)*
- #define [\\_TIM1\\_IC4F](#) ((uint8\_t) (0x0F << 4))
- TIM1 Output compare 4 mode [3:0] (in \_TIM1\_CCMR4)*

- `#define _TIM1_IC4F0` ((uint8\_t) (0x01 << 4))  
*TIM1 Input capture 4 filter [0] (in \_TIM1\_CCMR4)*
- `#define _TIM1_IC4F1` ((uint8\_t) (0x01 << 5))  
*TIM1 Input capture 4 filter [1] (in \_TIM1\_CCMR4)*
- `#define _TIM1_IC4F2` ((uint8\_t) (0x01 << 6))  
*TIM1 Input capture 4 filter [2] (in \_TIM1\_CCMR4)*
- `#define _TIM1_IC4F3` ((uint8\_t) (0x01 << 7))  
*TIM1 Input capture 4 filter [3] (in \_TIM1\_CCMR4)*
- `#define _TIM1_CC1E` ((uint8\_t) (0x01 << 0))  
*TIM1 Capture/compare 1 output enable [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC1P` ((uint8\_t) (0x01 << 1))  
*TIM1 Capture/compare 1 output polarity [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC1NE` ((uint8\_t) (0x01 << 2))  
*TIM1 Capture/compare 1 complementary output enable [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC1NP` ((uint8\_t) (0x01 << 3))  
*TIM1 Capture/compare 1 complementary output polarity [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC2E` ((uint8\_t) (0x01 << 4))  
*TIM1 Capture/compare 2 output enable [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC2P` ((uint8\_t) (0x01 << 5))  
*TIM1 Capture/compare 2 output polarity [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC2NE` ((uint8\_t) (0x01 << 6))  
*TIM1 Capture/compare 2 complementary output enable [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC2NP` ((uint8\_t) (0x01 << 7))  
*TIM1 Capture/compare 2 complementary output polarity [0] (in \_TIM1\_CCER1)*
- `#define _TIM1_CC3E` ((uint8\_t) (0x01 << 0))  
*TIM1 Capture/compare 3 output enable [0] (in \_TIM1\_CCER2)*
- `#define _TIM1_CC3P` ((uint8\_t) (0x01 << 1))  
*TIM1 Capture/compare 3 output polarity [0] (in \_TIM1\_CCER2)*
- `#define _TIM1_CC3NE` ((uint8\_t) (0x01 << 2))  
*TIM1 Capture/compare 3 complementary output enable [0] (in \_TIM1\_CCER2)*
- `#define _TIM1_CC3NP` ((uint8\_t) (0x01 << 3))  
*TIM1 Capture/compare 3 complementary output polarity [0] (in \_TIM1\_CCER2)*
- `#define _TIM1_CC4E` ((uint8\_t) (0x01 << 4))  
*TIM1 Capture/compare 4 output enable [0] (in \_TIM1\_CCER2)*
- `#define _TIM1_CC4P` ((uint8\_t) (0x01 << 5))  
*TIM1 Capture/compare 4 output polarity [0] (in \_TIM1\_CCER2)*
- `#define _TIM1_LOCK` ((int8\_t) (0x03 << 0))  
*TIM1 Lock configuration [1:0] (in \_TIM1\_BKR)*
- `#define _TIM1_LOCK0` ((uint8\_t) (0x01 << 0))  
*TIM1 Lock configuration [0] (in \_TIM1\_BKR)*
- `#define _TIM1_LOCK1` ((uint8\_t) (0x01 << 1))  
*TIM1 Lock configuration [1] (in \_TIM1\_BKR)*
- `#define _TIM1_OSSI` ((uint8\_t) (0x01 << 2))  
*TIM1 Off state selection for idle mode [0] (in \_TIM1\_BKR)*
- `#define _TIM1_OSSR` ((uint8\_t) (0x01 << 3))  
*TIM1 Off state selection for Run mode [0] (in \_TIM1\_BKR)*
- `#define _TIM1_BKE` ((uint8\_t) (0x01 << 4))  
*TIM1 Break enable [0] (in \_TIM1\_BKR)*
- `#define _TIM1_BKP` ((uint8\_t) (0x01 << 5))  
*TIM1 Break polarity [0] (in \_TIM1\_BKR)*
- `#define _TIM1_AOE` ((uint8\_t) (0x01 << 6))

- TIM1 Automatic output enable [0] (in \_TIM1\_BKR)*
- #define [\\_TIM1\\_MOE](#) ((uint8\_t) (0x01 << 7))
- TIM1 Main output enable [0] (in \_TIM1\_BKR)*
- #define [\\_TIM1\\_OIS1](#) ((uint8\_t) (0x01 << 0))
- TIM1 Output idle state 1 (OC1 output) [0] (in \_TIM1\_OISR)*
- #define [\\_TIM1\\_OIS1N](#) ((uint8\_t) (0x01 << 1))
- TIM1 Output idle state 1 (OC1N output) [0] (in \_TIM1\_OISR)*
- #define [\\_TIM1\\_OIS2](#) ((uint8\_t) (0x01 << 2))
- TIM1 Output idle state 2 (OC2 output) [0] (in \_TIM1\_OISR)*
- #define [\\_TIM1\\_OIS2N](#) ((uint8\_t) (0x01 << 3))
- TIM1 Output idle state 2 (OC2N output) [0] (in \_TIM1\_OISR)*
- #define [\\_TIM1\\_OIS3](#) ((uint8\_t) (0x01 << 4))
- TIM1 Output idle state 3 (OC3 output) [0] (in \_TIM1\_OISR)*
- #define [\\_TIM1\\_OIS3N](#) ((uint8\_t) (0x01 << 5))
- TIM1 Output idle state 3 (OC3N output) [0] (in \_TIM1\_OISR)*
- #define [\\_TIM1\\_OIS4](#) ((uint8\_t) (0x01 << 6))
- TIM1 Output idle state 4 (OC4 output) [0] (in \_TIM1\_OISR)*
- #define [\\_TIM2\\_SFR](#)(TIM2\_t, TIM2\_AddressBase)
- TIM2 struct/bit access.*
- #define [\\_TIM2\\_CR1\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x00)
- TIM2 control register 1.*
- #define [\\_TIM2\\_IER\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x01)
- TIM2 interrupt enable register.*
- #define [\\_TIM2\\_SR1\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x02)
- TIM2 status register 1.*
- #define [\\_TIM2\\_SR2\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x03)
- TIM2 status register 2.*
- #define [\\_TIM2\\_EGR\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x04)
- TIM2 Event generation register.*
- #define [\\_TIM2\\_CCMR1\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x05)
- TIM2 Capture/compare mode register 1.*
- #define [\\_TIM2\\_CCMR2\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x06)
- TIM2 Capture/compare mode register 2.*
- #define [\\_TIM2\\_CCMR3\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x07)
- TIM2 Capture/compare mode register 3.*
- #define [\\_TIM2\\_CCER1\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x08)
- TIM2 Capture/compare enable register 1.*
- #define [\\_TIM2\\_CCER2\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x09)
- TIM2 Capture/compare enable register 2.*
- #define [\\_TIM2\\_CNTRH\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x0A)
- TIM2 counter register high byte.*
- #define [\\_TIM2\\_CNTRL\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x0B)
- TIM2 counter register low byte.*
- #define [\\_TIM2\\_PSCR\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x0C)
- TIM2 clock prescaler register.*
- #define [\\_TIM2\\_ARRH\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x0D)
- TIM2 auto-reload register high byte.*
- #define [\\_TIM2\\_ARRL\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x0E)
- TIM2 auto-reload register low byte.*
- #define [\\_TIM2\\_CCR1H\\_SFR](#)(uint8\_t, TIM2\_AddressBase+0x0F)
- TIM2 16-bit capture/compare value 1 high byte.*

- `#define _TIM2_CCR1L_SFR(uint8_t, TIM2_AddressBase+0x10)`  
*TIM2 16-bit capture/compare value 1 low byte.*
- `#define _TIM2_CCR2H_SFR(uint8_t, TIM2_AddressBase+0x11)`  
*TIM2 16-bit capture/compare value 2 high byte.*
- `#define _TIM2_CCR2L_SFR(uint8_t, TIM2_AddressBase+0x12)`  
*TIM2 16-bit capture/compare value 2 low byte.*
- `#define _TIM2_CCR3H_SFR(uint8_t, TIM2_AddressBase+0x13)`  
*TIM2 16-bit capture/compare value 3 high byte.*
- `#define _TIM2_CCR3L_SFR(uint8_t, TIM2_AddressBase+0x14)`  
*TIM2 16-bit capture/compare value 3 low byte.*
- `#define _TIM2_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 control register 1 reset value.*
- `#define _TIM2_IER_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 interrupt enable register reset value.*
- `#define _TIM2_SR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 status register 1 reset value.*
- `#define _TIM2_SR2_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 status register 2 reset value.*
- `#define _TIM2_EGR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 Event generation register reset value.*
- `#define _TIM2_CCMR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 Capture/compare mode register 1 reset value.*
- `#define _TIM2_CCMR2_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 Capture/compare mode register 2 reset value.*
- `#define _TIM2_CCMR3_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 Capture/compare mode register 3 reset value.*
- `#define _TIM2_CCER1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 Capture/compare enable register 1 reset value.*
- `#define _TIM2_CCER2_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 Capture/compare enable register 2 reset value.*
- `#define _TIM2_CNTRH_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 counter register high byte reset value.*
- `#define _TIM2_CNTRL_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 counter register low byte reset value.*
- `#define _TIM2_PSCR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 clock prescaler register reset value.*
- `#define _TIM2_ARRH_RESET_VALUE ((uint8_t) 0xFF)`  
*TIM2 auto-reload register high byte reset value.*
- `#define _TIM2_ARRL_RESET_VALUE ((uint8_t) 0xFF)`  
*TIM2 auto-reload register low byte reset value.*
- `#define _TIM2_CCR1H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 1 high byte reset value.*
- `#define _TIM2_CCR1L_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 1 low byte reset value.*
- `#define _TIM2_CCR2H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 2 high byte reset value.*
- `#define _TIM2_CCR2L_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 2 low byte reset value.*
- `#define _TIM2_CCR3H_RESET_VALUE ((uint8_t) 0x00)`  
*TIM2 16-bit capture/compare value 3 high byte reset value.*
- `#define _TIM2_CCR3L_RESET_VALUE ((uint8_t) 0x00)`

- TIM2 16-bit capture/compare value 3 low byte reset value.*
- #define `_TIM2_CEN` ((uint8\_t) (0x01 << 0))  
*TIM2 Counter enable [0] (in \_TIM2\_CR1)*
- #define `_TIM2_UDIS` ((uint8\_t) (0x01 << 1))  
*TIM2 Update disable [0] (in \_TIM2\_CR1)*
- #define `_TIM2_URS` ((uint8\_t) (0x01 << 2))  
*TIM2 Update request source [0] (in \_TIM2\_CR1)*
- #define `_TIM2_OPM` ((uint8\_t) (0x01 << 3))  
*TIM2 One-pulse mode [0] (in \_TIM2\_CR1)*
- #define `_TIM2_ARPE` ((uint8\_t) (0x01 << 7))  
*TIM2 Auto-reload preload enable [0] (in \_TIM2\_CR1)*
- #define `_TIM2_UIE` ((uint8\_t) (0x01 << 0))  
*TIM2 Update interrupt enable [0] (in \_TIM2\_IER)*
- #define `_TIM2_CC1IE` ((uint8\_t) (0x01 << 1))  
*TIM2 Capture/compare 1 interrupt enable [0] (in \_TIM2\_IER)*
- #define `_TIM2_CC2IE` ((uint8\_t) (0x01 << 2))  
*TIM2 Capture/compare 2 interrupt enable [0] (in \_TIM2\_IER)*
- #define `_TIM2_CC3IE` ((uint8\_t) (0x01 << 3))  
*TIM2 Capture/compare 3 interrupt enable [0] (in \_TIM2\_IER)*
- #define `_TIM2_UIF` ((uint8\_t) (0x01 << 0))  
*TIM2 Update interrupt flag [0] (in \_TIM2\_SR1)*
- #define `_TIM2_CC1IF` ((uint8\_t) (0x01 << 1))  
*TIM2 Capture/compare 1 interrupt flag [0] (in \_TIM2\_SR1)*
- #define `_TIM2_CC2IF` ((uint8\_t) (0x01 << 2))  
*TIM2 Capture/compare 2 interrupt flag [0] (in \_TIM2\_SR1)*
- #define `_TIM2_CC3IF` ((uint8\_t) (0x01 << 3))  
*TIM2 Capture/compare 3 interrupt flag [0] (in \_TIM2\_SR1)*
- #define `_TIM2_CC1OF` ((uint8\_t) (0x01 << 1))  
*TIM2 Capture/compare 1 overcapture flag [0] (in \_TIM2\_SR2)*
- #define `_TIM2_CC2OF` ((uint8\_t) (0x01 << 2))  
*TIM2 Capture/compare 2 overcapture flag [0] (in \_TIM2\_SR2)*
- #define `_TIM2_CC3OF` ((uint8\_t) (0x01 << 3))  
*TIM2 Capture/compare 3 overcapture flag [0] (in \_TIM2\_SR2)*
- #define `_TIM2_UG` ((uint8\_t) (0x01 << 0))  
*TIM2 Update generation [0] (in \_TIM2\_EGR)*
- #define `_TIM2_CC1G` ((uint8\_t) (0x01 << 1))  
*TIM2 Capture/compare 1 generation [0] (in \_TIM2\_EGR)*
- #define `_TIM2_CC2G` ((uint8\_t) (0x01 << 2))  
*TIM2 Capture/compare 2 generation [0] (in \_TIM2\_EGR)*
- #define `_TIM2_CC3G` ((uint8\_t) (0x01 << 3))  
*TIM2 Capture/compare 3 generation [0] (in \_TIM2\_EGR)*
- #define `_TIM2_CC1S` ((uint8\_t) (0x03 << 0))  
*TIM2 Compare 1 selection [1:0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_CC1S0` ((uint8\_t) (0x01 << 0))  
*TIM2 Compare 1 selection [0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_CC1S1` ((uint8\_t) (0x01 << 1))  
*TIM2 Compare 1 selection [1] (in \_TIM2\_CCMR1)*
- #define `_TIM2_OC1PE` ((uint8\_t) (0x01 << 3))  
*TIM2 Output compare 1 preload enable [0] (in \_TIM2\_CCMR1)*
- #define `_TIM2_OC1M` ((uint8\_t) (0x07 << 4))  
*TIM2 Output compare 1 mode [2:0] (in \_TIM2\_CCMR1)*

- `#define _TIM2_OC1M0` ((uint8\_t) (0x01 << 4))  
*TIM2 Output compare 1 mode [0] (in \_TIM2\_CCMR1)*
- `#define _TIM2_OC1M1` ((uint8\_t) (0x01 << 5))  
*TIM2 Output compare 1 mode [1] (in \_TIM2\_CCMR1)*
- `#define _TIM2_OC1M2` ((uint8\_t) (0x01 << 6))  
*TIM2 Output compare 1 mode [2] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1PSC` ((uint8\_t) (0x03 << 2))  
*TIM2 Input capture 1 prescaler [1:0] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1PSC0` ((uint8\_t) (0x01 << 2))  
*TIM2 Input capture 1 prescaler [0] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1PSC1` ((uint8\_t) (0x01 << 3))  
*TIM2 Input capture 1 prescaler [1] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1F` ((uint8\_t) (0x0F << 4))  
*TIM2 Output compare 1 mode [3:0] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1F0` ((uint8\_t) (0x01 << 4))  
*TIM2 Output compare 1 mode [0] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1F1` ((uint8\_t) (0x01 << 5))  
*TIM2 Output compare 1 mode [1] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1F2` ((uint8\_t) (0x01 << 6))  
*TIM2 Output compare 1 mode [2] (in \_TIM2\_CCMR1)*
- `#define _TIM2_IC1F3` ((uint8\_t) (0x01 << 7))  
*TIM2 Output compare 1 mode [3] (in \_TIM2\_CCMR1)*
- `#define _TIM2_CC2S` ((uint8\_t) (0x03 << 0))  
*TIM2 Compare 2 selection [1:0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_CC2S0` ((uint8\_t) (0x01 << 0))  
*TIM2 Compare 2 selection [0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_CC2S1` ((uint8\_t) (0x01 << 1))  
*TIM2 Compare 2 selection [1] (in \_TIM2\_CCMR2)*
- `#define _TIM2_OC2PE` ((uint8\_t) (0x01 << 3))  
*TIM2 Output compare 2 preload enable [0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_OC2M` ((uint8\_t) (0x07 << 4))  
*TIM2 Output compare 2 mode [2:0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_OC2M0` ((uint8\_t) (0x01 << 4))  
*TIM2 Output compare 2 mode [0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_OC2M1` ((uint8\_t) (0x01 << 5))  
*TIM2 Output compare 2 mode [1] (in \_TIM2\_CCMR2)*
- `#define _TIM2_OC2M2` ((uint8\_t) (0x01 << 6))  
*TIM2 Output compare 2 mode [2] (in \_TIM2\_CCMR2)*
- `#define _TIM2_IC2PSC` ((uint8\_t) (0x03 << 2))  
*TIM2 Input capture 2 prescaler [1:0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_IC2PSC0` ((uint8\_t) (0x01 << 2))  
*TIM2 Input capture 2 prescaler [0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_IC2PSC1` ((uint8\_t) (0x01 << 3))  
*TIM2 Input capture 2 prescaler [1] (in \_TIM2\_CCMR2)*
- `#define _TIM2_IC2F` ((uint8\_t) (0x0F << 4))  
*TIM2 Output compare 2 mode [3:0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_IC2F0` ((uint8\_t) (0x01 << 4))  
*TIM2 Output compare 2 mode [0] (in \_TIM2\_CCMR2)*
- `#define _TIM2_IC2F1` ((uint8\_t) (0x01 << 5))  
*TIM2 Output compare 2 mode [1] (in \_TIM2\_CCMR2)*
- `#define _TIM2_IC2F2` ((uint8\_t) (0x01 << 6))  
*TIM2 Output compare 2 mode [2] (in \_TIM2\_CCMR2)*

- TIM2 Output compare 2 mode [2] (in \_TIM2\_CCMR2)*
- #define [\\_TIM2\\_IC2F3](#) ((uint8\_t) (0x01 << 7))
- TIM2 Output compare 2 mode [3] (in \_TIM2\_CCMR2)*
- #define [\\_TIM2\\_CC3S](#) ((uint8\_t) (0x03 << 0))
- TIM2 Compare 3 selection [1:0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_CC3S0](#) ((uint8\_t) (0x01 << 0))
- TIM2 Compare 3 selection [0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_CC3S1](#) ((uint8\_t) (0x01 << 1))
- TIM2 Compare 3 selection [1] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_OC3PE](#) ((uint8\_t) (0x01 << 3))
- TIM2 Output compare 3 preload enable [0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_OC3M](#) ((uint8\_t) (0x07 << 4))
- TIM2 Output compare 3 mode [2:0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_OC3M0](#) ((uint8\_t) (0x01 << 4))
- TIM2 Output compare 3 mode [0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_OC3M1](#) ((uint8\_t) (0x01 << 5))
- TIM2 Output compare 3 mode [1] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_OC3M2](#) ((uint8\_t) (0x01 << 6))
- TIM2 Output compare 3 mode [2] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3PSC](#) ((uint8\_t) (0x03 << 2))
- TIM2 Input capture 3 prescaler [1:0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3PSC0](#) ((uint8\_t) (0x01 << 2))
- TIM2 Input capture 3 prescaler [0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3PSC1](#) ((uint8\_t) (0x01 << 3))
- TIM2 Input capture 3 prescaler [1] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3F](#) ((uint8\_t) (0x0F << 4))
- TIM2 Output compare 3 mode [3:0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3F0](#) ((uint8\_t) (0x01 << 4))
- TIM2 Output compare 3 mode [0] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3F1](#) ((uint8\_t) (0x01 << 5))
- TIM2 Output compare 3 mode [1] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3F2](#) ((uint8\_t) (0x01 << 6))
- TIM2 Output compare 3 mode [2] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_IC3F3](#) ((uint8\_t) (0x01 << 7))
- TIM2 Output compare 3 mode [3] (in \_TIM2\_CCMR3)*
- #define [\\_TIM2\\_CC1E](#) ((uint8\_t) (0x01 << 0))
- TIM2 Capture/compare 1 output enable [0] (in \_TIM2\_CCER1)*
- #define [\\_TIM2\\_CC1P](#) ((uint8\_t) (0x01 << 1))
- TIM2 Capture/compare 1 output polarity [0] (in \_TIM2\_CCER1)*
- #define [\\_TIM2\\_CC2E](#) ((uint8\_t) (0x01 << 4))
- TIM2 Capture/compare 2 output enable [0] (in \_TIM2\_CCER1)*
- #define [\\_TIM2\\_CC2P](#) ((uint8\_t) (0x01 << 5))
- TIM2 Capture/compare 2 output polarity [0] (in \_TIM2\_CCER1)*
- #define [\\_TIM2\\_CC3E](#) ((uint8\_t) (0x01 << 0))
- TIM2 Capture/compare 3 output enable [0] (in \_TIM2\_CCER2)*
- #define [\\_TIM2\\_CC3P](#) ((uint8\_t) (0x01 << 1))
- TIM2 Capture/compare 3 output polarity [0] (in \_TIM2\_CCER2)*
- #define [\\_TIM2\\_PSC](#) ((uint8\_t) (0x0F << 0))
- TIM2 prescaler [3:0] (in \_TIM2\_PSCR)*
- #define [\\_TIM2\\_PSC0](#) ((uint8\_t) (0x01 << 0))
- TIM2 prescaler [0] (in \_TIM2\_PSCR)*



- `#define _TIM2_PSC1 ((uint8_t) (0x01 << 1))`  
*TIM2 prescaler [1] (in \_TIM2\_PSCR)*
- `#define _TIM2_PSC2 ((uint8_t) (0x01 << 2))`  
*TIM2 prescaler [2] (in \_TIM2\_PSCR)*
- `#define _TIM2_PSC3 ((uint8_t) (0x01 << 3))`  
*TIM2 prescaler [3] (in \_TIM2\_PSCR)*
- `#define _TIM3_SFR(TIM3_t, TIM3_AddressBase)`  
*TIM3 struct/bit access.*
- `#define _TIM3_CR1_SFR(uint8_t, TIM3_AddressBase+0x00)`  
*TIM3 control register 1.*
- `#define _TIM3_IER_SFR(uint8_t, TIM3_AddressBase+0x01)`  
*TIM3 interrupt enable register.*
- `#define _TIM3_SR1_SFR(uint8_t, TIM3_AddressBase+0x02)`  
*TIM3 status register 1.*
- `#define _TIM3_SR2_SFR(uint8_t, TIM3_AddressBase+0x03)`  
*TIM3 status register 2.*
- `#define _TIM3_EGR_SFR(uint8_t, TIM3_AddressBase+0x04)`  
*TIM3 Event generation register.*
- `#define _TIM3_CCMR1_SFR(uint8_t, TIM3_AddressBase+0x05)`  
*TIM3 Capture/compare mode register 1.*
- `#define _TIM3_CCMR2_SFR(uint8_t, TIM3_AddressBase+0x06)`  
*TIM3 Capture/compare mode register 2.*
- `#define _TIM3_CCER1_SFR(uint8_t, TIM3_AddressBase+0x08)`  
*TIM3 Capture/compare enable register 1.*
- `#define _TIM3_CNTRH_SFR(uint8_t, TIM3_AddressBase+0x0A)`  
*TIM3 counter register high byte.*
- `#define _TIM3_CNTRL_SFR(uint8_t, TIM3_AddressBase+0x0B)`  
*TIM3 counter register low byte.*
- `#define _TIM3_PSCR_SFR(uint8_t, TIM3_AddressBase+0x0C)`  
*TIM3 clock prescaler register.*
- `#define _TIM3_ARRH_SFR(uint8_t, TIM3_AddressBase+0x0D)`  
*TIM3 auto-reload register high byte.*
- `#define _TIM3_ARRL_SFR(uint8_t, TIM3_AddressBase+0x0E)`  
*TIM3 auto-reload register low byte.*
- `#define _TIM3_CCR1H_SFR(uint8_t, TIM3_AddressBase+0x0F)`  
*TIM3 16-bit capture/compare value 1 high byte.*
- `#define _TIM3_CCR1L_SFR(uint8_t, TIM3_AddressBase+0x10)`  
*TIM3 16-bit capture/compare value 1 low byte.*
- `#define _TIM3_CCR2H_SFR(uint8_t, TIM3_AddressBase+0x11)`  
*TIM3 16-bit capture/compare value 2 high byte.*
- `#define _TIM3_CCR2L_SFR(uint8_t, TIM3_AddressBase+0x12)`  
*TIM3 16-bit capture/compare value 2 low byte.*
- `#define _TIM3_CR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 control register 1 reset value.*
- `#define _TIM3_IER_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 interrupt enable register reset value.*
- `#define _TIM3_SR1_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 status register 1 reset value.*
- `#define _TIM3_SR2_RESET_VALUE ((uint8_t) 0x00)`  
*TIM3 status register 2 reset value.*
- `#define _TIM3_EGR_RESET_VALUE ((uint8_t) 0x00)`



- TIM3 Event generation register reset value.*
- #define [\\_TIM3\\_CCMR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 Capture/compare mode register 1 reset value.*
- #define [\\_TIM3\\_CCMR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 Capture/compare mode register 2 reset value.*
- #define [\\_TIM3\\_CCER1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 Capture/compare enable register 1 reset value.*
- #define [\\_TIM3\\_CNTRH\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 counter register high byte reset value.*
- #define [\\_TIM3\\_CNTRL\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 counter register low byte reset value.*
- #define [\\_TIM3\\_PSCR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 clock prescaler register reset value.*
- #define [\\_TIM3\\_ARRH\\_RESET\\_VALUE](#) ((uint8\_t) 0xFF)
- TIM3 auto-reload register high byte reset value.*
- #define [\\_TIM3\\_ARRL\\_RESET\\_VALUE](#) ((uint8\_t) 0xFF)
- TIM3 auto-reload register low byte reset value.*
- #define [\\_TIM3\\_CCR1H\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 16-bit capture/compare value 1 high byte reset value.*
- #define [\\_TIM3\\_CCR1L\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 16-bit capture/compare value 1 low byte reset value.*
- #define [\\_TIM3\\_CCR2H\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 16-bit capture/compare value 2 high byte reset value.*
- #define [\\_TIM3\\_CCR2L\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM3 16-bit capture/compare value 2 low byte reset value.*
- #define [\\_TIM3\\_CEN](#) ((uint8\_t) (0x01 << 0))
- TIM3 Counter enable [0] (in \_TIM3\_CR1)*
- #define [\\_TIM3\\_UDIS](#) ((uint8\_t) (0x01 << 1))
- TIM3 Update disable [0] (in \_TIM3\_CR1)*
- #define [\\_TIM3\\_URS](#) ((uint8\_t) (0x01 << 2))
- TIM3 Update request source [0] (in \_TIM3\_CR1)*
- #define [\\_TIM3\\_OPM](#) ((uint8\_t) (0x01 << 3))
- TIM3 One-pulse mode [0] (in \_TIM3\_CR1)*
- #define [\\_TIM3\\_ARPE](#) ((uint8\_t) (0x01 << 7))
- TIM3 Auto-reload preload enable [0] (in \_TIM3\_CR1)*
- #define [\\_TIM3\\_UIE](#) ((uint8\_t) (0x01 << 0))
- TIM3 Update interrupt enable [0] (in \_TIM3\_IER)*
- #define [\\_TIM3\\_CC1IE](#) ((uint8\_t) (0x01 << 1))
- TIM3 Capture/compare 1 interrupt enable [0] (in \_TIM3\_IER)*
- #define [\\_TIM3\\_CC2IE](#) ((uint8\_t) (0x01 << 2))
- TIM3 Capture/compare 2 interrupt enable [0] (in \_TIM3\_IER)*
- #define [\\_TIM3\\_UIF](#) ((uint8\_t) (0x01 << 0))
- TIM3 Update interrupt flag [0] (in \_TIM3\_SR1)*
- #define [\\_TIM3\\_CC1IF](#) ((uint8\_t) (0x01 << 1))
- TIM3 Capture/compare 1 interrupt flag [0] (in \_TIM3\_SR1)*
- #define [\\_TIM3\\_CC2IF](#) ((uint8\_t) (0x01 << 2))
- TIM3 Capture/compare 2 interrupt flag [0] (in \_TIM3\_SR1)*
- #define [\\_TIM3\\_CC1OF](#) ((uint8\_t) (0x01 << 1))
- TIM3 Capture/compare 1 overcapture flag [0] (in \_TIM3\_SR2)*
- #define [\\_TIM3\\_CC2OF](#) ((uint8\_t) (0x01 << 2))
- TIM3 Capture/compare 2 overcapture flag [0] (in \_TIM3\_SR2)*

- `#define _TIM3_UG` ((uint8\_t) (0x01 << 0))  
*TIM3 Update generation [0] (in \_TIM3\_EGR)*
- `#define _TIM3_CC1G` ((uint8\_t) (0x01 << 1))  
*TIM3 Capture/compare 1 generation [0] (in \_TIM3\_EGR)*
- `#define _TIM3_CC2G` ((uint8\_t) (0x01 << 2))  
*TIM3 Capture/compare 2 generation [0] (in \_TIM3\_EGR)*
- `#define _TIM3_CC1S` ((uint8\_t) (0x03 << 0))  
*TIM3 Compare 1 selection [1:0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_CC1S0` ((uint8\_t) (0x01 << 0))  
*TIM3 Compare 1 selection [0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_CC1S1` ((uint8\_t) (0x01 << 1))  
*TIM3 Compare 1 selection [1] (in \_TIM3\_CCMR1)*
- `#define _TIM3_OC1PE` ((uint8\_t) (0x01 << 3))  
*TIM3 Output compare 1 preload enable [0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_OC1M` ((uint8\_t) (0x07 << 4))  
*TIM3 Output compare 1 mode [2:0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_OC1M0` ((uint8\_t) (0x01 << 4))  
*TIM3 Output compare 1 mode [0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_OC1M1` ((uint8\_t) (0x01 << 5))  
*TIM3 Output compare 1 mode [1] (in \_TIM3\_CCMR1)*
- `#define _TIM3_OC1M2` ((uint8\_t) (0x01 << 6))  
*TIM3 Output compare 1 mode [2] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1PSC` ((uint8\_t) (0x03 << 2))  
*TIM3 Input capture 1 prescaler [1:0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1PSC0` ((uint8\_t) (0x01 << 2))  
*TIM3 Input capture 1 prescaler [0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1PSC1` ((uint8\_t) (0x01 << 3))  
*TIM3 Input capture 1 prescaler [1] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1F` ((uint8\_t) (0x0F << 4))  
*TIM3 Output compare 1 mode [3:0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1F0` ((uint8\_t) (0x01 << 4))  
*TIM3 Output compare 1 mode [0] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1F1` ((uint8\_t) (0x01 << 5))  
*TIM3 Output compare 1 mode [1] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1F2` ((uint8\_t) (0x01 << 6))  
*TIM3 Output compare 1 mode [2] (in \_TIM3\_CCMR1)*
- `#define _TIM3_IC1F3` ((uint8\_t) (0x01 << 7))  
*TIM3 Output compare 1 mode [3] (in \_TIM3\_CCMR1)*
- `#define _TIM3_CC2S` ((uint8\_t) (0x03 << 0))  
*TIM3 Compare 2 selection [1:0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_CC2S0` ((uint8\_t) (0x01 << 0))  
*TIM3 Compare 2 selection [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_CC2S1` ((uint8\_t) (0x01 << 1))  
*TIM3 Compare 2 selection [1] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2PE` ((uint8\_t) (0x01 << 3))  
*TIM3 Output compare 2 preload enable [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2M` ((uint8\_t) (0x07 << 4))  
*TIM3 Output compare 2 mode [2:0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2M0` ((uint8\_t) (0x01 << 4))  
*TIM3 Output compare 2 mode [0] (in \_TIM3\_CCMR2)*
- `#define _TIM3_OC2M1` ((uint8\_t) (0x01 << 5))

- TIM3 Output compare 2 mode [1] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_OC2M2](#) ((uint8\_t) (0x01 << 6))
- TIM3 Output compare 2 mode [2] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2PSC](#) ((uint8\_t) (0x03 << 2))
- TIM3 Input capture 2 prescaler [1:0] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2PSC0](#) ((uint8\_t) (0x01 << 2))
- TIM3 Input capture 2 prescaler [0] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2PSC1](#) ((uint8\_t) (0x01 << 3))
- TIM3 Input capture 2 prescaler [1] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2F](#) ((uint8\_t) (0x0F << 4))
- TIM3 Output compare 2 mode [3:0] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2F0](#) ((uint8\_t) (0x01 << 4))
- TIM3 Output compare 2 mode [0] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2F1](#) ((uint8\_t) (0x01 << 5))
- TIM3 Output compare 2 mode [1] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2F2](#) ((uint8\_t) (0x01 << 6))
- TIM3 Output compare 2 mode [2] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_IC2F3](#) ((uint8\_t) (0x01 << 7))
- TIM3 Output compare 2 mode [3] (in \_TIM3\_CCMR2)*
  - #define [\\_TIM3\\_CC1E](#) ((uint8\_t) (0x01 << 0))
- TIM3 Capture/compare 1 output enable [0] (in \_TIM3\_CCER1)*
  - #define [\\_TIM3\\_CC1P](#) ((uint8\_t) (0x01 << 1))
- TIM3 Capture/compare 1 output polarity [0] (in \_TIM3\_CCER1)*
  - #define [\\_TIM3\\_CC2E](#) ((uint8\_t) (0x01 << 4))
- TIM3 Capture/compare 2 output enable [0] (in \_TIM3\_CCER1)*
  - #define [\\_TIM3\\_CC2P](#) ((uint8\_t) (0x01 << 5))
- TIM3 Capture/compare 2 output polarity [0] (in \_TIM3\_CCER1)*
  - #define [\\_TIM3\\_PSC](#) ((uint8\_t) (0x0F << 0))
- TIM3 clock prescaler [3:0] (in \_TIM3\_PSCR)*
  - #define [\\_TIM3\\_PSC0](#) ((uint8\_t) (0x01 << 0))
- TIM3 clock prescaler [0] (in \_TIM3\_PSCR)*
  - #define [\\_TIM3\\_PSC1](#) ((uint8\_t) (0x01 << 1))
- TIM3 clock prescaler [1] (in \_TIM3\_PSCR)*
  - #define [\\_TIM3\\_PSC2](#) ((uint8\_t) (0x01 << 2))
- TIM3 clock prescaler [2] (in \_TIM3\_PSCR)*
  - #define [\\_TIM3\\_PSC3](#) ((uint8\_t) (0x01 << 3))
- TIM3 clock prescaler [3] (in \_TIM3\_PSCR)*
  - #define [\\_TIM4\\_SFR](#)(TIM4\_t, [TIM4\\_AddressBase](#))
- TIM4 struct/bit access.*
  - #define [\\_TIM4\\_CR\\_SFR](#)(uint8\_t, [TIM4\\_AddressBase](#)+0x00)
- TIM4 control register.*
  - #define [\\_TIM4\\_IER\\_SFR](#)(uint8\_t, [TIM4\\_AddressBase](#)+0x01)
- TIM4 interrupt enable register.*
  - #define [\\_TIM4\\_SR\\_SFR](#)(uint8\_t, [TIM4\\_AddressBase](#)+0x02)
- TIM4 status register.*
  - #define [\\_TIM4\\_EGR\\_SFR](#)(uint8\_t, [TIM4\\_AddressBase](#)+0x03)
- TIM4 event generation register.*
  - #define [\\_TIM4\\_CNTR\\_SFR](#)(uint8\_t, [TIM4\\_AddressBase](#)+0x04)
- TIM4 counter register.*
  - #define [\\_TIM4\\_PSCR\\_SFR](#)(uint8\_t, [TIM4\\_AddressBase](#)+0x05)
- TIM4 clock prescaler register.*

- `#define _TIM4_ARR_SFR(uint8_t, TIM4_AddressBase+0x06)`  
*TIM4 auto-reload register.*
- `#define _TIM4_CR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM4 control register reset value.*
- `#define _TIM4_IER_RESET_VALUE ((uint8_t) 0x00)`  
*TIM4 interrupt enable register reset value.*
- `#define _TIM4_SR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM4 status register reset value.*
- `#define _TIM4_EGR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM4 event generation register reset value.*
- `#define _TIM4_CNTR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM4 counter register reset value.*
- `#define _TIM4_PSCR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM4 clock prescaler register reset value.*
- `#define _TIM4_ARR_RESET_VALUE ((uint8_t) 0xFF)`  
*TIM4 auto-reload register reset value.*
- `#define _TIM4_CEN ((uint8_t) (0x01 << 0))`  
*TIM4 Counter enable [0] (in \_TIM4\_CR)*
- `#define _TIM4_UDIS ((uint8_t) (0x01 << 1))`  
*TIM4 Update disable [0] (in \_TIM4\_CR)*
- `#define _TIM4_URS ((uint8_t) (0x01 << 2))`  
*TIM4 Update request source [0] (in \_TIM4\_CR)*
- `#define _TIM4_OPM ((uint8_t) (0x01 << 3))`  
*TIM4 One-pulse mode [0] (in \_TIM4\_CR)*
- `#define _TIM4_ARPE ((uint8_t) (0x01 << 7))`  
*TIM4 Auto-reload preload enable [0] (in \_TIM4\_CR)*
- `#define _TIM4_UIE ((uint8_t) (0x01 << 0))`  
*TIM4 Update interrupt enable [0] (in \_TIM4\_IER)*
- `#define _TIM4_UIF ((uint8_t) (0x01 << 0))`  
*TIM4 Update interrupt flag [0] (in \_TIM4\_SR)*
- `#define _TIM4_UG ((uint8_t) (0x01 << 0))`  
*TIM4 Update generation [0] (in \_TIM4\_EGR)*
- `#define _TIM4_PSC ((uint8_t) (0x07 << 0))`  
*TIM4 clock prescaler [2:0] (in \_TIM4\_PSCR)*
- `#define _TIM4_PSC0 ((uint8_t) (0x01 << 0))`  
*TIM4 clock prescaler [0] (in \_TIM4\_PSCR)*
- `#define _TIM4_PSC1 ((uint8_t) (0x01 << 1))`  
*TIM4 clock prescaler [1] (in \_TIM4\_PSCR)*
- `#define _TIM4_PSC2 ((uint8_t) (0x01 << 2))`  
*TIM4 clock prescaler [2] (in \_TIM4\_PSCR)*
- `#define _TIM5_SFR(TIM5_t, TIM5_AddressBase)`  
*TIM5 struct/bit access.*
- `#define _TIM5_CR1_SFR(uint8_t, TIM5_AddressBase+0x00)`  
*TIM5 control register 1.*
- `#define _TIM5_CR2_SFR(uint8_t, TIM5_AddressBase+0x01)`  
*TIM5 control register 2.*
- `#define _TIM5_SMCR_SFR(uint8_t, TIM5_AddressBase+0x02)`  
*TIM5 Slave mode control register.*
- `#define _TIM5_IER_SFR(uint8_t, TIM5_AddressBase+0x03)`  
*TIM5 interrupt enable register.*
- `#define _TIM5_SR1_SFR(uint8_t, TIM5_AddressBase+0x04)`

- TIM5 status register 1.*
- #define [\\_TIM5\\_SR2\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x05)
- TIM5 status register 2.*
- #define [\\_TIM5\\_EGR\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x06)
- TIM5 Event generation register.*
- #define [\\_TIM5\\_CCMR1\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x07)
- TIM5 Capture/compare mode register 1.*
- #define [\\_TIM5\\_CCMR2\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x08)
- TIM5 Capture/compare mode register 2.*
- #define [\\_TIM5\\_CCMR3\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x09)
- TIM5 Capture/compare mode register 3.*
- #define [\\_TIM5\\_CCER1\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x0A)
- TIM5 Capture/compare enable register 1.*
- #define [\\_TIM5\\_CCER2\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x0B)
- TIM5 Capture/compare enable register 2.*
- #define [\\_TIM5\\_CNTRH\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x0C)
- TIM5 counter register high byte.*
- #define [\\_TIM5\\_CNTRL\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x0D)
- TIM5 counter register low byte.*
- #define [\\_TIM5\\_PSCR\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x0E)
- TIM5 clock prescaler register.*
- #define [\\_TIM5\\_ARRH\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x0F)
- TIM5 auto-reload register high byte.*
- #define [\\_TIM5\\_ARRL\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x10)
- TIM5 auto-reload register low byte.*
- #define [\\_TIM5\\_CCR1H\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x11)
- TIM5 16-bit capture/compare value 1 high byte.*
- #define [\\_TIM5\\_CCR1L\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x12)
- TIM5 16-bit capture/compare value 1 low byte.*
- #define [\\_TIM5\\_CCR2H\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x13)
- TIM5 16-bit capture/compare value 2 high byte.*
- #define [\\_TIM5\\_CCR2L\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x14)
- TIM5 16-bit capture/compare value 2 low byte.*
- #define [\\_TIM5\\_CCR3H\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x15)
- TIM5 16-bit capture/compare value 3 high byte.*
- #define [\\_TIM5\\_CCR3L\\_SFR](#)(uint8\_t, [TIM5\\_AddressBase](#)+0x16)
- TIM5 16-bit capture/compare value 3 low byte.*
- #define [\\_TIM5\\_CR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM5 control register 1 reset value.*
- #define [\\_TIM5\\_CR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM5 control register 2 reset value.*
- #define [\\_TIM5\\_SMCR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM5 Slave mode control register reset value.*
- #define [\\_TIM5\\_IER\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM5 interrupt enable register reset value.*
- #define [\\_TIM5\\_SR1\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM5 status register 1 reset value.*
- #define [\\_TIM5\\_SR2\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM5 status register 2 reset value.*
- #define [\\_TIM5\\_EGR\\_RESET\\_VALUE](#) ((uint8\_t) 0x00)
- TIM5 Event generation register reset value.*

- `#define _TIM5_CCMR1_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 Capture/compare mode register 1 reset value.*
- `#define _TIM5_CCMR2_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 Capture/compare mode register 2 reset value.*
- `#define _TIM5_CCMR3_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 Capture/compare mode register 3 reset value.*
- `#define _TIM5_CCER1_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 Capture/compare enable register 1 reset value.*
- `#define _TIM5_CCER2_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 Capture/compare enable register 2 reset value.*
- `#define _TIM5_CNTRH_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 counter register high byte reset value.*
- `#define _TIM5_CNTRL_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 counter register low byte reset value.*
- `#define _TIM5_PSCR_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 clock prescaler register reset value.*
- `#define _TIM5_ARRH_RESET_VALUE` ((uint8\_t) 0xFF)  
*TIM5 auto-reload register high byte reset value.*
- `#define _TIM5_ARRL_RESET_VALUE` ((uint8\_t) 0xFF)  
*TIM5 auto-reload register low byte reset value.*
- `#define _TIM5_CCR1H_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 16-bit capture/compare value 1 high byte reset value.*
- `#define _TIM5_CCR1L_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 16-bit capture/compare value 1 low byte reset value.*
- `#define _TIM5_CCR2H_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 16-bit capture/compare value 2 high byte reset value.*
- `#define _TIM5_CCR2L_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 16-bit capture/compare value 2 low byte reset value.*
- `#define _TIM5_CCR3H_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 16-bit capture/compare value 3 high byte reset value.*
- `#define _TIM5_CCR3L_RESET_VALUE` ((uint8\_t) 0x00)  
*TIM5 16-bit capture/compare value 3 low byte reset value.*
- `#define _TIM5_CEN` ((uint8\_t) (0x01 << 0))  
*TIM5 Counter enable [0] (in \_TIM5\_CR1)*
- `#define _TIM5_UDIS` ((uint8\_t) (0x01 << 1))  
*TIM5 Update disable [0] (in \_TIM5\_CR1)*
- `#define _TIM5_URS` ((uint8\_t) (0x01 << 2))  
*TIM5 Update request source [0] (in \_TIM5\_CR1)*
- `#define _TIM5_OPM` ((uint8\_t) (0x01 << 3))  
*TIM5 One-pulse mode [0] (in \_TIM5\_CR1)*
- `#define _TIM5_ARPE` ((uint8\_t) (0x01 << 7))  
*TIM5 Auto-reload preload enable [0] (in \_TIM5\_CR1)*
- `#define _TIM5_CCPC` ((uint8\_t) (0x01 << 0))  
*TIM5 Capture/compare preloaded control [0] (in \_TIM5\_CR2)*
- `#define _TIM5_COMS` ((uint8\_t) (0x01 << 2))  
*TIM5 Capture/compare control update selection [0] (in \_TIM5\_CR2)*
- `#define _TIM5_MMS` ((uint8\_t) (0x07 << 4))  
*TIM5 Master mode selection [2:0] (in \_TIM5\_CR2)*
- `#define _TIM5_MMS0` ((uint8\_t) (0x01 << 4))  
*TIM5 Master mode selection [0] (in \_TIM5\_CR2)*
- `#define _TIM5_MMS1` ((uint8\_t) (0x01 << 5))

- TIM5 Master mode selection [1] (in \_TIM5\_CR2)*
  - #define [\\_TIM5\\_MMS2](#) ((uint8\_t) (0x01 << 6))
- TIM5 Master mode selection [2] (in \_TIM5\_CR2)*
  - #define [\\_TIM5\\_SMS](#) ((uint8\_t) (0x07 << 0))
- TIM5 Clock/trigger/slave mode selection [2:0] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_SMS0](#) ((uint8\_t) (0x01 << 0))
- TIM5 Clock/trigger/slave mode selection [0] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_SMS1](#) ((uint8\_t) (0x01 << 1))
- TIM5 Clock/trigger/slave mode selection [1] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_SMS2](#) ((uint8\_t) (0x01 << 2))
- TIM5 Clock/trigger/slave mode selection [2] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_TS](#) ((uint8\_t) (0x07 << 4))
- TIM5 Trigger selection [2:0] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_TS0](#) ((uint8\_t) (0x01 << 4))
- TIM5 Trigger selection [0] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_TS1](#) ((uint8\_t) (0x01 << 5))
- TIM5 Trigger selection [1] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_TS2](#) ((uint8\_t) (0x01 << 6))
- TIM5 Trigger selection [2] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_MSM](#) ((uint8\_t) (0x01 << 7))
- TIM5 Master/slave mode [0] (in \_TIM5\_SMCR)*
  - #define [\\_TIM5\\_UIE](#) ((uint8\_t) (0x01 << 0))
- TIM5 Update interrupt enable [0] (in \_TIM5\_IER)*
  - #define [\\_TIM5\\_CC1IE](#) ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 1 interrupt enable [0] (in \_TIM5\_IER)*
  - #define [\\_TIM5\\_CC2IE](#) ((uint8\_t) (0x01 << 2))
- TIM5 Capture/compare 2 interrupt enable [0] (in \_TIM5\_IER)*
  - #define [\\_TIM5\\_CC3IE](#) ((uint8\_t) (0x01 << 3))
- TIM5 Capture/compare 3 interrupt enable [0] (in \_TIM5\_IER)*
  - #define [\\_TIM5\\_TIE](#) ((uint8\_t) (0x01 << 6))
- TIM5 Trigger interrupt enable [0] (in \_TIM5\_IER)*
  - #define [\\_TIM5\\_UIF](#) ((uint8\_t) (0x01 << 0))
- TIM5 Update interrupt flag [0] (in \_TIM5\_SR1)*
  - #define [\\_TIM5\\_CC1IF](#) ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 1 interrupt flag [0] (in \_TIM5\_SR1)*
  - #define [\\_TIM5\\_CC2IF](#) ((uint8\_t) (0x01 << 2))
- TIM5 Capture/compare 2 interrupt flag [0] (in \_TIM5\_SR1)*
  - #define [\\_TIM5\\_CC3IF](#) ((uint8\_t) (0x01 << 3))
- TIM5 Capture/compare 3 interrupt flag [0] (in \_TIM5\_SR1)*
  - #define [\\_TIM5\\_TIF](#) ((uint8\_t) (0x01 << 6))
- TIM5 Trigger interrupt flag [0] (in \_TIM5\_SR1)*
  - #define [\\_TIM5\\_CC1OF](#) ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 1 overcapture flag [0] (in \_TIM5\_SR2)*
  - #define [\\_TIM5\\_CC2OF](#) ((uint8\_t) (0x01 << 2))
- TIM5 Capture/compare 2 overcapture flag [0] (in \_TIM5\_SR2)*
  - #define [\\_TIM5\\_CC3OF](#) ((uint8\_t) (0x01 << 3))
- TIM5 Capture/compare 3 overcapture flag [0] (in \_TIM5\_SR2)*
  - #define [\\_TIM5\\_UG](#) ((uint8\_t) (0x01 << 0))
- TIM5 Update generation [0] (in \_TIM5\_EGR)*
  - #define [\\_TIM5\\_CC1G](#) ((uint8\_t) (0x01 << 1))
- TIM5 Capture/compare 1 generation [0] (in \_TIM5\_EGR)*



- `#define _TIM5_CC2G ((uint8_t) (0x01 << 2))`  
*TIM5 Capture/compare 2 generation [0] (in \_TIM5\_EGR)*
- `#define _TIM5_CC3G ((uint8_t) (0x01 << 3))`  
*TIM5 Capture/compare 3 generation [0] (in \_TIM5\_EGR)*
- `#define _TIM5_TG ((uint8_t) (0x01 << 6))`  
*TIM5 Trigger generation [0] (in \_TIM5\_EGR)*
- `#define _TIM5_CC1S ((uint8_t) (0x03 << 0))`  
*TIM5 Compare 1 selection [1:0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_CC1S0 ((uint8_t) (0x01 << 0))`  
*TIM5 Compare 1 selection [0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_CC1S1 ((uint8_t) (0x01 << 1))`  
*TIM5 Compare 1 selection [1] (in \_TIM5\_CCMR1)*
- `#define _TIM5_OC1PE ((uint8_t) (0x01 << 3))`  
*TIM5 Output compare 1 preload enable [0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_OC1M ((uint8_t) (0x07 << 4))`  
*TIM5 Output compare 1 mode [2:0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_OC1M0 ((uint8_t) (0x01 << 4))`  
*TIM5 Output compare 1 mode [0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_OC1M1 ((uint8_t) (0x01 << 5))`  
*TIM5 Output compare 1 mode [1] (in \_TIM5\_CCMR1)*
- `#define _TIM5_OC1M2 ((uint8_t) (0x01 << 6))`  
*TIM5 Output compare 1 mode [2] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1PSC ((uint8_t) (0x03 << 2))`  
*TIM5 Input capture 1 prescaler [1:0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1PSC0 ((uint8_t) (0x01 << 2))`  
*TIM5 Input capture 1 prescaler [0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1PSC1 ((uint8_t) (0x01 << 3))`  
*TIM5 Input capture 1 prescaler [1] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1F ((uint8_t) (0x0F << 4))`  
*TIM5 Output compare 1 mode [3:0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1F0 ((uint8_t) (0x01 << 4))`  
*TIM5 Input capture 1 filter [0] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1F1 ((uint8_t) (0x01 << 5))`  
*TIM5 Input capture 1 filter [1] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1F2 ((uint8_t) (0x01 << 6))`  
*TIM5 Input capture 1 filter [2] (in \_TIM5\_CCMR1)*
- `#define _TIM5_IC1F3 ((uint8_t) (0x01 << 7))`  
*TIM5 Input capture 1 filter [3] (in \_TIM5\_CCMR1)*
- `#define _TIM5_CC2S ((uint8_t) (0x03 << 0))`  
*TIM5 Compare 2 selection [1:0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_CC2S0 ((uint8_t) (0x01 << 0))`  
*TIM5 Compare 2 selection [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_CC2S1 ((uint8_t) (0x01 << 1))`  
*TIM5 Compare 2 selection [1] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2PE ((uint8_t) (0x01 << 3))`  
*TIM5 Output compare 2 preload enable [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2M ((uint8_t) (0x07 << 4))`  
*TIM5 Output compare 2 mode [2:0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2M0 ((uint8_t) (0x01 << 4))`  
*TIM5 Output compare 2 mode [0] (in \_TIM5\_CCMR2)*
- `#define _TIM5_OC2M1 ((uint8_t) (0x01 << 5))`



- TIM5 Output compare 2 mode [1] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_OC2M2](#) ((uint8\_t) (0x01 << 6))
- TIM5 Output compare 2 mode [2] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2PSC](#) ((uint8\_t) (0x03 << 2))
- TIM5 Input capture 2 prescaler [1:0] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2PSC0](#) ((uint8\_t) (0x01 << 2))
- TIM5 Input capture 2 prescaler [0] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2PSC1](#) ((uint8\_t) (0x01 << 3))
- TIM5 Input capture 2 prescaler [1] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2F](#) ((uint8\_t) (0x0F << 4))
- TIM5 Output compare 2 mode [3:0] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2F0](#) ((uint8\_t) (0x01 << 4))
- TIM5 Input capture 2 filter [0] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2F1](#) ((uint8\_t) (0x01 << 5))
- TIM5 Input capture 2 filter [1] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2F2](#) ((uint8\_t) (0x01 << 6))
- TIM5 Input capture 2 filter [2] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_IC2F3](#) ((uint8\_t) (0x01 << 7))
- TIM5 Input capture 2 filter [3] (in \_TIM5\_CCMR2)*
- #define [\\_TIM5\\_CC3S](#) ((uint8\_t) (0x03 << 0))
- TIM5 Compare 3 selection [1:0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_CC3S0](#) ((uint8\_t) (0x01 << 0))
- TIM5 Compare 3 selection [0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_CC3S1](#) ((uint8\_t) (0x01 << 1))
- TIM5 Compare 3 selection [1] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_OC3PE](#) ((uint8\_t) (0x01 << 3))
- TIM5 Output compare 3 preload enable [0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_OC3M](#) ((uint8\_t) (0x07 << 4))
- TIM5 Output compare 3 mode [2:0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_OC3M0](#) ((uint8\_t) (0x01 << 4))
- TIM5 Output compare 3 mode [0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_OC3M1](#) ((uint8\_t) (0x01 << 5))
- TIM5 Output compare 3 mode [1] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_OC3M2](#) ((uint8\_t) (0x01 << 6))
- TIM5 Output compare 3 mode [2] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3PSC](#) ((uint8\_t) (0x03 << 2))
- TIM5 Input capture 3 prescaler [1:0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3PSC0](#) ((uint8\_t) (0x01 << 2))
- TIM5 Input capture 3 prescaler [0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3PSC1](#) ((uint8\_t) (0x01 << 3))
- TIM5 Input capture 3 prescaler [1] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3F](#) ((uint8\_t) (0x0F << 4))
- TIM5 Output compare 3 mode [3:0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3F0](#) ((uint8\_t) (0x01 << 4))
- TIM5 Input capture 3 filter [0] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3F1](#) ((uint8\_t) (0x01 << 5))
- TIM5 Input capture 3 filter [1] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3F2](#) ((uint8\_t) (0x01 << 6))
- TIM5 Input capture 3 filter [2] (in \_TIM5\_CCMR3)*
- #define [\\_TIM5\\_IC3F3](#) ((uint8\_t) (0x01 << 7))
- TIM5 Input capture 3 filter [3] (in \_TIM5\_CCMR3)*

- `#define _TIM5_CC1E ((uint8_t) (0x01 << 0))`  
*TIM5 Capture/compare 1 output enable [0] (in \_TIM5\_CCER1)*
- `#define _TIM5_CC1P ((uint8_t) (0x01 << 1))`  
*TIM5 Capture/compare 1 output polarity [0] (in \_TIM5\_CCER1)*
- `#define _TIM5_CC2E ((uint8_t) (0x01 << 4))`  
*TIM5 Capture/compare 2 output enable [0] (in \_TIM5\_CCER1)*
- `#define _TIM5_CC2P ((uint8_t) (0x01 << 5))`  
*TIM5 Capture/compare 2 output polarity [0] (in \_TIM5\_CCER1)*
- `#define _TIM5_CC3E ((uint8_t) (0x01 << 0))`  
*TIM5 Capture/compare 3 output enable [0] (in \_TIM5\_CCER2)*
- `#define _TIM5_CC3P ((uint8_t) (0x01 << 1))`  
*TIM5 Capture/compare 3 output polarity [0] (in \_TIM5\_CCER2)*
- `#define _TIM5_PSC ((uint8_t) (0x0F << 0))`  
*TIM5 clock prescaler [3:0] (in \_TIM5\_PSCR)*
- `#define _TIM5_PSC0 ((uint8_t) (0x01 << 0))`  
*TIM5 clock prescaler [0] (in \_TIM5\_PSCR)*
- `#define _TIM5_PSC1 ((uint8_t) (0x01 << 1))`  
*TIM5 clock prescaler [1] (in \_TIM5\_PSCR)*
- `#define _TIM5_PSC2 ((uint8_t) (0x01 << 2))`  
*TIM5 clock prescaler [2] (in \_TIM5\_PSCR)*
- `#define _TIM5_PSC3 ((uint8_t) (0x01 << 3))`  
*TIM5 clock prescaler [3] (in \_TIM5\_PSCR)*
- `#define _TIM6_SFR(TIM6_t, TIM6_AddressBase)`  
*TIM6 struct/bit access.*
- `#define _TIM6_CR_SFR(uint8_t, TIM6_AddressBase+0x00)`  
*TIM6 control register.*
- `#define _TIM6_IER_SFR(uint8_t, TIM6_AddressBase+0x01)`  
*TIM6 interrupt enable register.*
- `#define _TIM6_SR_SFR(uint8_t, TIM6_AddressBase+0x02)`  
*TIM6 status register.*
- `#define _TIM6_EGR_SFR(uint8_t, TIM6_AddressBase+0x03)`  
*TIM6 event generation register.*
- `#define _TIM6_CNTR_SFR(uint8_t, TIM6_AddressBase+0x04)`  
*TIM6 counter register.*
- `#define _TIM6_PSCR_SFR(uint8_t, TIM6_AddressBase+0x05)`  
*TIM6 clock prescaler register.*
- `#define _TIM6_ARR_SFR(uint8_t, TIM6_AddressBase+0x06)`  
*TIM6 auto-reload register.*
- `#define _TIM6_CR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 control register reset value.*
- `#define _TIM6_IER_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 interrupt enable register reset value.*
- `#define _TIM6_SR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 status register reset value.*
- `#define _TIM6_EGR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 event generation register reset value.*
- `#define _TIM6_CNTR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 counter register reset value.*
- `#define _TIM6_PSCR_RESET_VALUE ((uint8_t) 0x00)`  
*TIM6 clock prescaler register reset value.*
- `#define _TIM6_ARR_RESET_VALUE ((uint8_t) 0xFF)`

- TIM6 auto-reload register reset value.*
- #define `_TIM6_CEN` ((uint8\_t) (0x01 << 0))  
*TIM6 Counter enable [0] (in \_TIM6\_CR1)*
  - #define `_TIM6_UDIS` ((uint8\_t) (0x01 << 1))  
*TIM6 Update disable [0] (in \_TIM6\_CR1)*
  - #define `_TIM6_URS` ((uint8\_t) (0x01 << 2))  
*TIM6 Update request source [0] (in \_TIM6\_CR1)*
  - #define `_TIM6_OPM` ((uint8\_t) (0x01 << 3))  
*TIM6 One-pulse mode [0] (in \_TIM6\_CR1)*
  - #define `_TIM6_ARPE` ((uint8\_t) (0x01 << 7))  
*TIM6 Auto-reload preload enable [0] (in \_TIM6\_CR1)*
  - #define `_TIM6_MMS` ((uint8\_t) (0x07 << 4))  
*TIM6 Master mode selection [2:0] (in \_TIM6\_CR2)*
  - #define `_TIM6_MMS0` ((uint8\_t) (0x01 << 4))  
*TIM6 Master mode selection [0] (in \_TIM6\_CR2)*
  - #define `_TIM6_MMS1` ((uint8\_t) (0x01 << 5))  
*TIM6 Master mode selection [1] (in \_TIM6\_CR2)*
  - #define `_TIM6_MMS2` ((uint8\_t) (0x01 << 6))  
*TIM6 Master mode selection [2] (in \_TIM6\_CR2)*
  - #define `_TIM6_SMS` ((uint8\_t) (0x07 << 0))  
*TIM6 Clock/trigger/slave mode selection [2:0] (in \_TIM6\_SMCR)*
  - #define `_TIM6_SMS0` ((uint8\_t) (0x01 << 0))  
*TIM6 Clock/trigger/slave mode selection [0] (in \_TIM6\_SMCR)*
  - #define `_TIM6_SMS1` ((uint8\_t) (0x01 << 1))  
*TIM6 Clock/trigger/slave mode selection [1] (in \_TIM6\_SMCR)*
  - #define `_TIM6_SMS2` ((uint8\_t) (0x01 << 2))  
*TIM6 Clock/trigger/slave mode selection [2] (in \_TIM6\_SMCR)*
  - #define `_TIM6_TS` ((uint8\_t) (0x07 << 4))  
*TIM6 Trigger selection [2:0] (in \_TIM6\_SMCR)*
  - #define `_TIM6_TS0` ((uint8\_t) (0x01 << 4))  
*TIM6 Trigger selection [0] (in \_TIM6\_SMCR)*
  - #define `_TIM6_TS1` ((uint8\_t) (0x01 << 5))  
*TIM6 Trigger selection [1] (in \_TIM6\_SMCR)*
  - #define `_TIM6_TS2` ((uint8\_t) (0x01 << 6))  
*TIM6 Trigger selection [2] (in \_TIM6\_SMCR)*
  - #define `_TIM6_UIE` ((uint8\_t) (0x01 << 0))  
*TIM6 Update interrupt enable [0] (in \_TIM6\_IER)*
  - #define `_TIM6_UIF` ((uint8\_t) (0x01 << 0))  
*TIM6 Update interrupt flag [0] (in \_TIM6\_SR)*
  - #define `_TIM6_UG` ((uint8\_t) (0x01 << 0))  
*TIM6 Update generation [0] (in \_TIM6\_EGR)*
  - #define `_TIM6_PSC` ((uint8\_t) (0x07 << 0))  
*TIM6 clock prescaler [2:0] (in \_TIM6\_PSCR)*
  - #define `_TIM6_PSC0` ((uint8\_t) (0x01 << 0))  
*TIM6 clock prescaler [0] (in \_TIM6\_PSCR)*
  - #define `_TIM6_PSC1` ((uint8\_t) (0x01 << 1))  
*TIM6 clock prescaler [1] (in \_TIM6\_PSCR)*
  - #define `_TIM6_PSC2` ((uint8\_t) (0x01 << 2))  
*TIM6 clock prescaler [2] (in \_TIM6\_PSCR)*
  - #define `_ADC1_SFR(ADC1_t, ADC1_AddressBase)`  
*ADC1 struct/bit access.*

- `#define _ADC1_DB0RH_SFR(uint8_t, ADC1_AddressBase+0x00)`  
*ADC1 10-bit Data Buffer Register 0.*
- `#define _ADC1_DB0RL_SFR(uint8_t, ADC1_AddressBase+0x01)`  
*ADC1 10-bit Data Buffer Register 0.*
- `#define _ADC1_DB1RH_SFR(uint8_t, ADC1_AddressBase+0x02)`  
*ADC1 10-bit Data Buffer Register 1.*
- `#define _ADC1_DB1RL_SFR(uint8_t, ADC1_AddressBase+0x03)`  
*ADC1 10-bit Data Buffer Register 1.*
- `#define _ADC1_DB2RH_SFR(uint8_t, ADC1_AddressBase+0x04)`  
*ADC1 10-bit Data Buffer Register 2.*
- `#define _ADC1_DB2RL_SFR(uint8_t, ADC1_AddressBase+0x05)`  
*ADC1 10-bit Data Buffer Register 2.*
- `#define _ADC1_DB3RH_SFR(uint8_t, ADC1_AddressBase+0x06)`  
*ADC1 10-bit Data Buffer Register 3.*
- `#define _ADC1_DB3RL_SFR(uint8_t, ADC1_AddressBase+0x07)`  
*ADC1 10-bit Data Buffer Register 3.*
- `#define _ADC1_DB4RH_SFR(uint8_t, ADC1_AddressBase+0x08)`  
*ADC1 10-bit Data Buffer Register 4.*
- `#define _ADC1_DB4RL_SFR(uint8_t, ADC1_AddressBase+0x09)`  
*ADC1 10-bit Data Buffer Register 4.*
- `#define _ADC1_DB5RH_SFR(uint8_t, ADC1_AddressBase+0x0A)`  
*ADC1 10-bit Data Buffer Register 5.*
- `#define _ADC1_DB5RL_SFR(uint8_t, ADC1_AddressBase+0x0B)`  
*ADC1 10-bit Data Buffer Register 5.*
- `#define _ADC1_DB6RH_SFR(uint8_t, ADC1_AddressBase+0x0C)`  
*ADC1 10-bit Data Buffer Register 6.*
- `#define _ADC1_DB6RL_SFR(uint8_t, ADC1_AddressBase+0x0D)`  
*ADC1 10-bit Data Buffer Register 6.*
- `#define _ADC1_DB7RH_SFR(uint8_t, ADC1_AddressBase+0x0E)`  
*ADC1 10-bit Data Buffer Register 7.*
- `#define _ADC1_DB7RL_SFR(uint8_t, ADC1_AddressBase+0x0F)`  
*ADC1 10-bit Data Buffer Register 7.*
- `#define _ADC1_DB8RH_SFR(uint8_t, ADC1_AddressBase+0x10)`  
*ADC1 10-bit Data Buffer Register 8.*
- `#define _ADC1_DB8RL_SFR(uint8_t, ADC1_AddressBase+0x11)`  
*ADC1 10-bit Data Buffer Register 8.*
- `#define _ADC1_DB9RH_SFR(uint8_t, ADC1_AddressBase+0x12)`  
*ADC1 10-bit Data Buffer Register 9.*
- `#define _ADC1_DB9RL_SFR(uint8_t, ADC1_AddressBase+0x13)`  
*ADC1 10-bit Data Buffer Register 9.*
- `#define _ADC1_CSR_SFR(uint8_t, ADC1_AddressBase+0x20)`  
*ADC1 control/status register.*
- `#define _ADC1_CR1_SFR(uint8_t, ADC1_AddressBase+0x21)`  
*ADC1 Configuration Register 1.*
- `#define _ADC1_CR2_SFR(uint8_t, ADC1_AddressBase+0x22)`  
*ADC1 Configuration Register 2.*
- `#define _ADC1_CR3_SFR(uint8_t, ADC1_AddressBase+0x23)`  
*ADC1 Configuration Register 3.*
- `#define _ADC1_DRH_SFR(uint8_t, ADC1_AddressBase+0x24)`  
*ADC1 (unbuffered) 10-bit measurement result.*
- `#define _ADC1_DRL_SFR(uint8_t, ADC1_AddressBase+0x25)`

- ADC1 (unbuffered) 10-bit measurement result.*
- #define `_ADC1_TDRH_SFR`(uint8\_t, `ADC1_AddressBase`+0x26)  
*ADC1 Schmitt trigger disable register.*
- #define `_ADC1_TDRL_SFR`(uint8\_t, `ADC1_AddressBase`+0x27)  
*ADC1 Schmitt trigger disable register.*
- #define `_ADC1_HTRH_SFR`(uint8\_t, `ADC1_AddressBase`+0x28)  
*ADC1 watchdog high threshold register.*
- #define `_ADC1_HTRL_SFR`(uint8\_t, `ADC1_AddressBase`+0x29)  
*ADC1 watchdog high threshold register.*
- #define `_ADC1_LTRH_SFR`(uint8\_t, `ADC1_AddressBase`+0x2A)  
*ADC1 watchdog low threshold register.*
- #define `_ADC1_LTRL_SFR`(uint8\_t, `ADC1_AddressBase`+0x2B)  
*ADC1 watchdog low threshold register.*
- #define `_ADC1_AWSRH_SFR`(uint8\_t, `ADC1_AddressBase`+0x2C)  
*ADC1 watchdog status register.*
- #define `_ADC1_AWSRL_SFR`(uint8\_t, `ADC1_AddressBase`+0x2D)  
*ADC1 watchdog status register.*
- #define `_ADC1_AWCRH_SFR`(uint8\_t, `ADC1_AddressBase`+0x2E)  
*ADC1 watchdog control register.*
- #define `_ADC1_AWCRL_SFR`(uint8\_t, `ADC1_AddressBase`+0x2F)  
*ADC1 watchdog control register.*
- #define `_ADC1_CSR_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 control/status register reset value.*
- #define `_ADC1_CR1_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 Configuration Register 1 reset value.*
- #define `_ADC1_CR2_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 Configuration Register 2 reset value.*
- #define `_ADC1_CR3_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 Configuration Register 3 reset value.*
- #define `_ADC1_TDRH_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 Schmitt trigger disable register reset value.*
- #define `_ADC1_TDRL_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 Schmitt trigger disable register reset value.*
- #define `_ADC1_HTRH_RESET_VALUE` ((uint8\_t) 0xFF)  
*ADC1 watchdog high threshold register reset value.*
- #define `_ADC1_HTRL_RESET_VALUE` ((uint8\_t) 0x03)  
*ADC1 watchdog high threshold register reset value.*
- #define `_ADC1_LTRH_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 watchdog low threshold register reset value.*
- #define `_ADC1_LTRL_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 watchdog low threshold register reset value.*
- #define `_ADC1_AWCRH_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 watchdog control register reset value.*
- #define `_ADC1_AWCRL_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC1 watchdog control register reset value.*
- #define `_ADC1_CH` ((uint8\_t) (0x0F << 0))  
*ADC1 Channel selection bits [3:0] (in \_ADC1\_CSR)*
- #define `_ADC1_CH0` ((uint8\_t) (0x01 << 0))  
*ADC1 Channel selection bits [0] (in \_ADC1\_CSR)*
- #define `_ADC1_CH1` ((uint8\_t) (0x01 << 1))  
*ADC1 Channel selection bits [1] (in \_ADC1\_CSR)*

- `#define _ADC1_CH2 ((uint8_t) (0x01 << 2))`  
*ADC1 Channel selection bits [2] (in \_ADC1\_CSR)*
- `#define _ADC1_CH3 ((uint8_t) (0x01 << 3))`  
*ADC1 Channel selection bits [3] (in \_ADC1\_CSR)*
- `#define _ADC1_AWDIE ((uint8_t) (0x01 << 4))`  
*ADC1 Analog watchdog interrupt enable [0] (in \_ADC1\_CSR)*
- `#define _ADC1_EOCIE ((uint8_t) (0x01 << 5))`  
*ADC1 Interrupt enable for EOC [0] (in \_ADC1\_CSR)*
- `#define _ADC1_AWD ((uint8_t) (0x01 << 6))`  
*ADC1 Analog Watchdog flag [0] (in \_ADC1\_CSR)*
- `#define _ADC1_EOC ((uint8_t) (0x01 << 7))`  
*ADC1 End of conversion [0] (in \_ADC1\_CSR)*
- `#define _ADC1_ADON ((uint8_t) (0x01 << 0))`  
*ADC1 Conversion on/off [0] (in \_ADC1\_CR1)*
- `#define _ADC1_CONT ((uint8_t) (0x01 << 1))`  
*ADC1 Continuous conversion [0] (in \_ADC1\_CR1)*
- `#define _ADC1_SPSEL ((uint8_t) (0x07 << 4))`  
*ADC1 clock prescaler selection [2:0] (in \_ADC1\_CR1)*
- `#define _ADC1_SPSEL0 ((uint8_t) (0x01 << 4))`  
*ADC1 clock prescaler selection [0] (in \_ADC1\_CR1)*
- `#define _ADC1_SPSEL1 ((uint8_t) (0x01 << 5))`  
*ADC1 clock prescaler selection [1] (in \_ADC1\_CR1)*
- `#define _ADC1_SPSEL2 ((uint8_t) (0x01 << 6))`  
*ADC1 clock prescaler selection [2] (in \_ADC1\_CR1)*
- `#define _ADC1_SCAN ((uint8_t) (0x01 << 1))`  
*ADC1 Scan mode enable [0] (in \_ADC1\_CR2)*
- `#define _ADC1_ALIGN ((uint8_t) (0x01 << 3))`  
*ADC1 Data alignment [0] (in \_ADC1\_CR2)*
- `#define _ADC1_EXTSEL ((uint8_t) (0x03 << 4))`  
*ADC1 External event selection [1:0] (in \_ADC1\_CR2)*
- `#define _ADC1_EXTSEL0 ((uint8_t) (0x01 << 4))`  
*ADC1 External event selection [0] (in \_ADC1\_CR2)*
- `#define _ADC1_EXTSEL1 ((uint8_t) (0x01 << 5))`  
*ADC1 External event selection [1] (in \_ADC1\_CR2)*
- `#define _ADC1_EXTTRIG ((uint8_t) (0x01 << 6))`  
*ADC1 External trigger enable [0] (in \_ADC1\_CR2)*
- `#define _ADC1_OVR ((uint8_t) (0x01 << 6))`  
*ADC1 Overrun flag [0] (in \_ADC1\_CR3)*
- `#define _ADC1_DBUF ((uint8_t) (0x01 << 7))`  
*ADC1 Data buffer enable [0] (in \_ADC1\_CR3)*
- `#define _ADC2_SFR(ADC2_t, ADC2_AddressBase)`  
*ADC2 struct/bit access.*
- `#define _ADC2_CSR_SFR(uint8_t, ADC2_AddressBase+0x00)`  
*ADC2 control/status register.*
- `#define _ADC2_CR1_SFR(uint8_t, ADC2_AddressBase+0x01)`  
*ADC2 Configuration Register 1.*
- `#define _ADC2_CR2_SFR(uint8_t, ADC2_AddressBase+0x02)`  
*ADC2 Configuration Register 2.*
- `#define _ADC2_DRH_SFR(uint8_t, ADC2_AddressBase+0x04)`  
*ADC2 (unbuffered) 10-bit measurement result.*
- `#define _ADC2_DRL_SFR(uint8_t, ADC2_AddressBase+0x05)`

- *ADC2 (unbuffered) 10-bit measurement result.*  
• #define `_ADC2_TDRH_SFR`(uint8\_t, `ADC2_AddressBase`+0x06)  
*ADC2 Schmitt trigger disable register.*
- #define `_ADC2_TDRL_SFR`(uint8\_t, `ADC2_AddressBase`+0x07)  
*ADC2 Schmitt trigger disable register.*
- #define `_ADC2_CSR_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC2 control/status register reset value.*
- #define `_ADC2_CR1_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC2 Configuration Register 1 reset value.*
- #define `_ADC2_CR2_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC2 Configuration Register 2 reset value.*
- #define `_ADC2_TDRL_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC2 Schmitt trigger disable register reset value.*
- #define `_ADC2_TDRH_RESET_VALUE` ((uint8\_t) 0x00)  
*ADC2 Schmitt trigger disable register reset value.*
- #define `_ADC2_CH` ((uint8\_t) (0x0F << 0))  
*ADC2 Channel selection bits [3:0] (in \_ADC2\_CSR)*
- #define `_ADC2_CH0` ((uint8\_t) (0x01 << 0))  
*ADC2 Channel selection bits [0] (in \_ADC2\_CSR)*
- #define `_ADC2_CH1` ((uint8\_t) (0x01 << 1))  
*ADC2 Channel selection bits [1] (in \_ADC2\_CSR)*
- #define `_ADC2_CH2` ((uint8\_t) (0x01 << 2))  
*ADC2 Channel selection bits [2] (in \_ADC2\_CSR)*
- #define `_ADC2_CH3` ((uint8\_t) (0x01 << 3))  
*ADC2 Channel selection bits [3] (in \_ADC2\_CSR)*
- #define `_ADC2_EOCIE` ((uint8\_t) (0x01 << 5))  
*ADC2 Interrupt enable for EOC [0] (in \_ADC2\_CSR)*
- #define `_ADC2_EOC` ((uint8\_t) (0x01 << 7))  
*ADC2 End of conversion [0] (in \_ADC2\_CSR)*
- #define `_ADC2_ADON` ((uint8\_t) (0x01 << 0))  
*ADC2 Conversion on/off [0] (in \_ADC2\_CR1)*
- #define `_ADC2_CONT` ((uint8\_t) (0x01 << 1))  
*ADC2 Continuous conversion [0] (in \_ADC2\_CR1)*
- #define `_ADC2_SPSEL` ((uint8\_t) (0x07 << 4))  
*ADC2 clock prescaler selection [2:0] (in \_ADC2\_CR1)*
- #define `_ADC2_SPSEL0` ((uint8\_t) (0x01 << 4))  
*ADC2 clock prescaler selection [0] (in \_ADC2\_CR1)*
- #define `_ADC2_SPSEL1` ((uint8\_t) (0x01 << 5))  
*ADC2 clock prescaler selection [1] (in \_ADC2\_CR1)*
- #define `_ADC2_SPSEL2` ((uint8\_t) (0x01 << 6))  
*ADC2 clock prescaler selection [2] (in \_ADC2\_CR1)*
- #define `_ADC2_ALIGN` ((uint8\_t) (0x01 << 3))  
*ADC2 Data alignment [0] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTSEL` ((uint8\_t) (0x03 << 4))  
*ADC2 External event selection [1:0] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTSEL0` ((uint8\_t) (0x01 << 4))  
*ADC2 External event selection [0] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTSEL1` ((uint8\_t) (0x01 << 5))  
*ADC2 External event selection [1] (in \_ADC2\_CR2)*
- #define `_ADC2_EXTTRIG` ((uint8\_t) (0x01 << 6))  
*ADC2 External trigger enable [0] (in \_ADC2\_CR2)*



- `#define _CAN_SFR(CAN_t, CAN_AddressBase)`  
*CAN struct/bit access.*
- `#define _CAN_MCR_SFR(uint8_t, CAN_AddressBase+0x00)`  
*CAN master control register.*
- `#define _CAN_MSR_SFR(uint8_t, CAN_AddressBase+0x01)`  
*CAN master status register.*
- `#define _CAN_TSR_SFR(uint8_t, CAN_AddressBase+0x02)`  
*CAN transmit status register.*
- `#define _CAN_TPR_SFR(uint8_t, CAN_AddressBase+0x03)`  
*CAN transmit priority register.*
- `#define _CAN_RFR_SFR(uint8_t, CAN_AddressBase+0x04)`  
*CAN receive FIFO register.*
- `#define _CAN_IER_SFR(uint8_t, CAN_AddressBase+0x05)`  
*CAN interrupt enable register.*
- `#define _CAN_DGR_SFR(uint8_t, CAN_AddressBase+0x06)`  
*CAN diagnosis register.*
- `#define _CAN_PSR_SFR(uint8_t, CAN_AddressBase+0x07)`  
*CAN page selection for below paged registers.*
- `#define _CAN_MCSR_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`  
*CAN message control/status register (page 0,1,5)*
- `#define _CAN_MDLCR_SFR(uint8_t, CAN_AddressBase+0x08+0x01)`  
*CAN mailbox data length control register (page 0,1,5,7)*
- `#define _CAN_MIDR1_SFR(uint8_t, CAN_AddressBase+0x08+0x02)`  
*CAN mailbox identifier register 1 (page 0,1,5,7)*
- `#define _CAN_MIDR2_SFR(uint8_t, CAN_AddressBase+0x08+0x03)`  
*CAN mailbox identifier register 2 (page 0,1,5,7)*
- `#define _CAN_MIDR3_SFR(uint8_t, CAN_AddressBase+0x08+0x04)`  
*CAN mailbox identifier register 3 (page 0,1,5,7)*
- `#define _CAN_MIDR4_SFR(uint8_t, CAN_AddressBase+0x08+0x05)`  
*CAN mailbox identifier register 4 (page 0,1,5,7)*
- `#define _CAN_MDAR1_SFR(uint8_t, CAN_AddressBase+0x08+0x06)`  
*CAN mailbox data register 1 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR2_SFR(uint8_t, CAN_AddressBase+0x08+0x07)`  
*CAN mailbox data register 2 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR3_SFR(uint8_t, CAN_AddressBase+0x08+0x08)`  
*CAN mailbox data register 3 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR4_SFR(uint8_t, CAN_AddressBase+0x08+0x09)`  
*CAN mailbox data register 4 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR5_SFR(uint8_t, CAN_AddressBase+0x08+0x0A)`  
*CAN mailbox data register 5 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR6_SFR(uint8_t, CAN_AddressBase+0x08+0x0B)`  
*CAN mailbox data register 6 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR7_SFR(uint8_t, CAN_AddressBase+0x08+0x0C)`  
*CAN mailbox data register 7 (page 0,1,5,7) \*/.*
- `#define _CAN_MDAR8_SFR(uint8_t, CAN_AddressBase+0x08+0x0D)`  
*CAN mailbox data register 8 (page 0,1,5,7) \*/.*
- `#define _CAN_MTSRL_SFR(uint8_t, CAN_AddressBase+0x08+0x0E)`  
*CAN mailbox time stamp register low byte (page 0,1,5,7) \*/.*
- `#define _CAN_MTSRH_SFR(uint8_t, CAN_AddressBase+0x08+0x0F)`  
*CAN mailbox time stamp register high byte (page 0,1,5,7) \*/.*
- `#define _CAN_F0R1_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`



- CAN acceptance filter 0/1 (page 2)*
- #define [\\_CAN\\_F0R2\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x01](#))
- CAN acceptance filter 0/2 (page 2)*
- #define [\\_CAN\\_F0R3\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x02](#))
- CAN acceptance filter 0/3 (page 2)*
- #define [\\_CAN\\_F0R4\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x03](#))
- CAN acceptance filter 0/4 (page 2)*
- #define [\\_CAN\\_F0R5\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x04](#))
- CAN acceptance filter 0/5 (page 2)*
- #define [\\_CAN\\_F0R6\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x05](#))
- CAN acceptance filter 0/6 (page 2)*
- #define [\\_CAN\\_F0R7\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x06](#))
- CAN acceptance filter 0/7 (page 2)*
- #define [\\_CAN\\_F0R8\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x07](#))
- CAN acceptance filter 0/8 (page 2)*
- #define [\\_CAN\\_F1R1\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x08](#))
- CAN acceptance filter 1/1 (page 2)*
- #define [\\_CAN\\_F1R2\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x09](#))
- CAN acceptance filter 1/2 (page 2)*
- #define [\\_CAN\\_F1R3\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x0A](#))
- CAN acceptance filter 1/3 (page 2)*
- #define [\\_CAN\\_F1R4\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x0B](#))
- CAN acceptance filter 1/4 (page 2)*
- #define [\\_CAN\\_F1R5\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x0C](#))
- CAN acceptance filter 1/5 (page 2)*
- #define [\\_CAN\\_F1R6\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x0D](#))
- CAN acceptance filter 1/6 (page 2)*
- #define [\\_CAN\\_F1R7\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x0E](#))
- CAN acceptance filter 1/7 (page 2)*
- #define [\\_CAN\\_F1R8\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x0F](#))
- CAN acceptance filter 1/8 (page 2)*
- #define [\\_CAN\\_F2R1\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x00](#))
- CAN acceptance filter 2/1 (page 3)*
- #define [\\_CAN\\_F2R2\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x01](#))
- CAN acceptance filter 2/2 (page 3)*
- #define [\\_CAN\\_F2R3\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x02](#))
- CAN acceptance filter 2/3 (page 3)*
- #define [\\_CAN\\_F2R4\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x03](#))
- CAN acceptance filter 2/4 (page 3)*
- #define [\\_CAN\\_F2R5\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x04](#))
- CAN acceptance filter 2/5 (page 3)*
- #define [\\_CAN\\_F2R6\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x05](#))
- CAN acceptance filter 2/6 (page 3)*
- #define [\\_CAN\\_F2R7\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x06](#))
- CAN acceptance filter 2/7 (page 3)*
- #define [\\_CAN\\_F2R8\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x07](#))
- CAN acceptance filter 2/8 (page 3)*
- #define [\\_CAN\\_F3R1\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x08](#))
- CAN acceptance filter 3/1 (page 3)*
- #define [\\_CAN\\_F3R2\\_SFR](#)(uint8\_t, [CAN\\_AddressBase+0x08+0x09](#))
- CAN acceptance filter 3/2 (page 3)*

- `#define _CAN_F3R3_SFR(uint8_t, CAN_AddressBase+0x08+0x0A)`  
*CAN acceptance filter 3/3 (page 3)*
- `#define _CAN_F3R4_SFR(uint8_t, CAN_AddressBase+0x08+0x0B)`  
*CAN acceptance filter 3/4 (page 3)*
- `#define _CAN_F3R5_SFR(uint8_t, CAN_AddressBase+0x08+0x0C)`  
*CAN acceptance filter 3/5 (page 3)*
- `#define _CAN_F3R6_SFR(uint8_t, CAN_AddressBase+0x08+0x0D)`  
*CAN acceptance filter 3/6 (page 3)*
- `#define _CAN_F3R7_SFR(uint8_t, CAN_AddressBase+0x08+0x0E)`  
*CAN acceptance filter 3/7 (page 3)*
- `#define _CAN_F3R8_SFR(uint8_t, CAN_AddressBase+0x08+0x0F)`  
*CAN acceptance filter 3/8 (page 3)*
- `#define _CAN_F4R1_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`  
*CAN acceptance filter 4/1 (page 4)*
- `#define _CAN_F4R2_SFR(uint8_t, CAN_AddressBase+0x08+0x01)`  
*CAN acceptance filter 4/2 (page 4)*
- `#define _CAN_F4R3_SFR(uint8_t, CAN_AddressBase+0x08+0x02)`  
*CAN acceptance filter 4/3 (page 4)*
- `#define _CAN_F4R4_SFR(uint8_t, CAN_AddressBase+0x08+0x03)`  
*CAN acceptance filter 4/4 (page 4)*
- `#define _CAN_F4R5_SFR(uint8_t, CAN_AddressBase+0x08+0x04)`  
*CAN acceptance filter 4/5 (page 4)*
- `#define _CAN_F4R6_SFR(uint8_t, CAN_AddressBase+0x08+0x05)`  
*CAN acceptance filter 4/6 (page 4)*
- `#define _CAN_F4R7_SFR(uint8_t, CAN_AddressBase+0x08+0x06)`  
*CAN acceptance filter 4/7 (page 4)*
- `#define _CAN_F4R8_SFR(uint8_t, CAN_AddressBase+0x08+0x07)`  
*CAN acceptance filter 4/8 (page 4)*
- `#define _CAN_F5R1_SFR(uint8_t, CAN_AddressBase+0x08+0x08)`  
*CAN acceptance filter 5/1 (page 4)*
- `#define _CAN_F5R2_SFR(uint8_t, CAN_AddressBase+0x08+0x09)`  
*CAN acceptance filter 5/2 (page 4)*
- `#define _CAN_F5R3_SFR(uint8_t, CAN_AddressBase+0x08+0x0A)`  
*CAN acceptance filter 5/3 (page 4)*
- `#define _CAN_F5R4_SFR(uint8_t, CAN_AddressBase+0x08+0x0B)`  
*CAN acceptance filter 5/4 (page 4)*
- `#define _CAN_F5R5_SFR(uint8_t, CAN_AddressBase+0x08+0x0C)`  
*CAN acceptance filter 5/5 (page 4)*
- `#define _CAN_F5R6_SFR(uint8_t, CAN_AddressBase+0x08+0x0D)`  
*CAN acceptance filter 5/6 (page 4)*
- `#define _CAN_F5R7_SFR(uint8_t, CAN_AddressBase+0x08+0x0E)`  
*CAN acceptance filter 5/7 (page 4)*
- `#define _CAN_F5R8_SFR(uint8_t, CAN_AddressBase+0x08+0x0F)`  
*CAN acceptance filter 5/8 (page 4)*
- `#define _CAN_ESR_SFR(uint8_t, CAN_AddressBase+0x08+0x00)`  
*CAN error status register (page 6)*
- `#define _CAN_EIER_SFR(uint8_t, CAN_AddressBase+0x08+0x01)`  
*CAN error interrupt enable register (page 6)*
- `#define _CAN_TECR_SFR(uint8_t, CAN_AddressBase+0x08+0x02)`  
*CAN transmit error counter register (page 6)*
- `#define _CAN_RECR_SFR(uint8_t, CAN_AddressBase+0x08+0x03)`

- CAN receive error counter register (page 6)*
- #define `_CAN_BTR1_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x04)
- CAN bit timing register 1 (page 6)*
- #define `_CAN_BTR2_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x05)
- CAN bit timing register 2 (page 6)*
- #define `_CAN_FMR1_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x08)
- CAN filter mode register 1 (page 6)*
- #define `_CAN_FMR2_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x09)
- CAN filter mode register 2 (page 6)*
- #define `_CAN_FCR1_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x0A)
- CAN filter configuration register 1 (page 6)*
- #define `_CAN_FCR2_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x0B)
- CAN filter configuration register 2 (page 6)*
- #define `_CAN_FCR3_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x0C)
- CAN filter configuration register 3 (page 6)*
- #define `_CAN_MFMIR_SFR`(uint8\_t, `CAN_AddressBase`+0x08+0x00)
- CAN mailbox filter match index register (page 7)*
- #define `_CAN_MCR_RESET_VALUE` ((uint8\_t) 0x02)
- CAN master control register reset value.*
- #define `_CAN_MSR_RESET_VALUE` ((uint8\_t) 0x02)
- CAN master status register reset value.*
- #define `_CAN_TSR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN transmit status register reset value.*
- #define `_CAN_TPR_RESET_VALUE` ((uint8\_t) 0x0C)
- CAN transmit priority register reset value.*
- #define `_CAN_RFR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN receive FIFO register reset value.*
- #define `_CAN_IER_RESET_VALUE` ((uint8\_t) 0x00)
- CAN interrupt enable register reset value.*
- #define `_CAN_DGR_RESET_VALUE` ((uint8\_t) 0x0C)
- CAN diagnosis register reset value.*
- #define `_CAN_PSR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN page selection reset value.*
- #define `_CAN_MCSR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN message control/status register (page 0,1,5) reset value.*
- #define `_CAN_MDLCR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN mailbox data length control register (page 0,1,5,7) reset value.*
- #define `_CAN_ESR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN error status register (page 6) reset value.*
- #define `_CAN_EIER_RESET_VALUE` ((uint8\_t) 0x00)
- CAN error interrupt enable register (page 6) reset value.*
- #define `_CAN_TECR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN transmit error counter register (page 6) reset value.*
- #define `_CAN_RECR_RESET_VALUE` ((uint8\_t) 0x00)
- CAN receive error counter register (page 6) reset value.*
- #define `_CAN_BTR1_RESET_VALUE` ((uint8\_t) 0x40)
- CAN bit timing register 1 (page 6) reset value.*
- #define `_CAN_BTR2_RESET_VALUE` ((uint8\_t) 0x23)
- CAN bit timing register 2 (page 6) reset value.*
- #define `_CAN_FMR1_RESET_VALUE` ((uint8\_t) 0x00)
- CAN filter mode register 1 (page 6) reset value.*

- `#define _CAN_FMR2_RESET_VALUE` ((uint8\_t) 0x00)  
*CAN filter mode register 2 (page 6) reset value.*
- `#define _CAN_FCR_RESET_VALUE` ((uint8\_t) 0x00)  
*CAN filter configuration register reset value.*
- `#define _CAN_MFMIR_RESET_VALUE` ((uint8\_t) 0x00)  
*CAN mailbox filter match index register reset value.*
- `#define _CAN_INRQ` ((uint8\_t) (0x01 << 0))  
*CAN Channel Initialization Request [0] (in \_CAN\_MCR)*
- `#define _CAN_SLEEP` ((uint8\_t) (0x01 << 1))  
*CAN Channel Sleep Mode Request [0] (in \_CAN\_MCR)*
- `#define _CAN_TXFP` ((uint8\_t) (0x01 << 2))  
*CAN Channel Transmit FIFO Priority [0] (in \_CAN\_MCR)*
- `#define _CAN_RFLM` ((uint8\_t) (0x01 << 3))  
*CAN Channel Receive FIFO Locked Mode [0] (in \_CAN\_MCR)*
- `#define _CAN_NART` ((uint8\_t) (0x01 << 4))  
*CAN Channel No Automatic Retransmission [0] (in \_CAN\_MCR)*
- `#define _CAN_AWUM` ((uint8\_t) (0x01 << 5))  
*CAN Channel Automatic Wakeup Mode [0] (in \_CAN\_MCR)*
- `#define _CAN_ABOM` ((uint8\_t) (0x01 << 6))  
*CAN Channel Automatic Bus-Off Management [0] (in \_CAN\_MCR)*
- `#define _CAN_TTCM` ((uint8\_t) (0x01 << 7))  
*CAN Channel Time Triggered Communication Mode [0] (in \_CAN\_MCR)*
- `#define _CAN_INAK` ((uint8\_t) (0x01 << 0))  
*CAN Initialization Acknowledge [0] (in \_CAN\_MSR)*
- `#define _CAN_SLAK` ((uint8\_t) (0x01 << 1))  
*CAN Sleep Acknowledge [0] (in \_CAN\_MSR)*
- `#define _CAN_ERRI` ((uint8\_t) (0x01 << 2))  
*CAN Error Interrupt [0] (in \_CAN\_MSR)*
- `#define _CAN_WKUI` ((uint8\_t) (0x01 << 3))  
*CAN Wakeup Interrupt [0] (in \_CAN\_MSR)*
- `#define _CAN_TX` ((uint8\_t) (0x01 << 4))  
*CAN Transmit [0] (in \_CAN\_MSR)*
- `#define _CAN_RX` ((uint8\_t) (0x01 << 5))  
*CAN Receive [0] (in \_CAN\_MSR)*
- `#define _CAN_RQCP0` ((uint8\_t) (0x01 << 0))  
*CAN Request Completed for Mailbox 0 [0] (in \_CAN\_TSR)*
- `#define _CAN_RQCP1` ((uint8\_t) (0x01 << 1))  
*CAN Request Completed for Mailbox 1 [0] (in \_CAN\_TSR)*
- `#define _CAN_RQCP2` ((uint8\_t) (0x01 << 2))  
*CAN Request Completed for Mailbox 2 [0] (in \_CAN\_TSR)*
- `#define _CAN_TXOK0` ((uint8\_t) (0x01 << 4))  
*CAN Transmission ok for Mailbox 0 [0] (in \_CAN\_TSR)*
- `#define _CAN_TXOK1` ((uint8\_t) (0x01 << 5))  
*CAN Transmission ok for Mailbox 1 [0] (in \_CAN\_TSR)*
- `#define _CAN_TXOK2` ((uint8\_t) (0x01 << 6))  
*CAN Transmission ok for Mailbox 2 [0] (in \_CAN\_TSR)*
- `#define _CAN_CODE` ((uint8\_t) (0x03 << 0))  
*CAN Mailbox Code [1:0] (in \_CAN\_TPR)*
- `#define _CAN_CODE0` ((uint8\_t) (0x01 << 0))  
*CAN Mailbox Code [0] (in \_CAN\_TPR)*
- `#define _CAN_CODE1` ((uint8\_t) (0x01 << 1))

- CAN Mailbox Code [1] (in \_CAN\_TPR)*
- #define `_CAN_TME0` ((uint8\_t) (0x01 << 2))
- CAN Transmit Mailbox 0 Empty [0] (in \_CAN\_TPR)*
- #define `_CAN_TME1` ((uint8\_t) (0x01 << 3))
- CAN Transmit Mailbox 1 Empty [0] (in \_CAN\_TPR)*
- #define `_CAN_TME2` ((uint8\_t) (0x01 << 4))
- CAN Transmit Mailbox 2 Empty [0] (in \_CAN\_TPR)*
- #define `_CAN_LOW0` ((uint8\_t) (0x01 << 5))
- CAN Lowest Priority Flag for Mailbox 0 [0] (in \_CAN\_TPR)*
- #define `_CAN_LOW1` ((uint8\_t) (0x01 << 6))
- CAN Lowest Priority Flag for Mailbox 1 [0] (in \_CAN\_TPR)*
- #define `_CAN_LOW2` ((uint8\_t) (0x01 << 7))
- CAN Lowest Priority Flag for Mailbox 2 [0] (in \_CAN\_TPR)*
- #define `_CAN_FMP` ((uint8\_t) (0x03 << 0))
- CAN FIFO Message Pending [1:0] (in \_CAN\_RFR)*
- #define `_CAN_FMP0` ((uint8\_t) (0x01 << 0))
- CAN FIFO Message Pending [0] (in \_CAN\_RFR)*
- #define `_CAN_FMP1` ((uint8\_t) (0x01 << 1))
- CAN FIFO Message Pending [1] (in \_CAN\_RFR)*
- #define `_CAN_FULL` ((uint8\_t) (0x01 << 3))
- CAN FIFO Full [0] (in \_CAN\_RFR)*
- #define `_CAN_FOVR` ((uint8\_t) (0x01 << 4))
- CAN FIFO Overrun [0] (in \_CAN\_RFR)*
- #define `_CAN_RFOM` ((uint8\_t) (0x01 << 5))
- CAN Release FIFO Output Mailbox [0] (in \_CAN\_RFR)*
- #define `_CAN_TMEIE` ((uint8\_t) (0x01 << 0))
- CAN Transmit Mailbox Empty Interrupt Enable [0] (in \_CAN\_IER)*
- #define `_CAN_FMPIE` ((uint8\_t) (0x01 << 1))
- CAN FIFO Message Pending Interrupt Enable [0] (in \_CAN\_IER)*
- #define `_CAN_FFIE` ((uint8\_t) (0x01 << 2))
- CAN FIFO Full Interrupt Enable [0] (in \_CAN\_IER)*
- #define `_CAN_FOVIE` ((uint8\_t) (0x01 << 3))
- CAN FIFO Overrun Interrupt Enable [0] (in \_CAN\_IER)*
- #define `_CAN_WKUIE` ((uint8\_t) (0x01 << 7))
- CAN Wakeup Interrupt Enable [0] (in \_CAN\_IER)*
- #define `_CAN_LBKM` ((uint8\_t) (0x01 << 0))
- CAN Loop back mode [0] (in \_CAN\_DGR)*
- #define `_CAN_SILM` ((uint8\_t) (0x01 << 1))
- CAN Silent mode [0] (in \_CAN\_DGR)*
- #define `_CAN_SAMP` ((uint8\_t) (0x01 << 2))
- CAN Last sample point [0] (in \_CAN\_DGR)*
- #define `_CAN_RXS` ((uint8\_t) (0x01 << 3))
- CAN Rx Signal (=pin status) [0] (in \_CAN\_DGR)*
- #define `_CAN_TXM2E` ((uint8\_t) (0x01 << 4))
- CAN TX Mailbox 2 enable [0] (in \_CAN\_DGR)*
- #define `_CAN_PS` ((uint8\_t) (0x07 << 0))
- CAN Page select [2:0] (in \_CAN\_PSR)*
- #define `_CAN_PS0` ((uint8\_t) (0x01 << 0))
- CAN Page select [0] (in \_CAN\_PSR)*
- #define `_CAN_PS1` ((uint8\_t) (0x01 << 1))
- CAN Page select [1] (in \_CAN\_PSR)*

- `#define _CAN_PS2 ((uint8_t) (0x01 << 2))`  
CAN Page select [2] (in \_CAN\_PSR)
- `#define _CAN_TXRQ ((uint8_t) (0x01 << 0))`  
CAN Transmission mailbox request [0] (in \_CAN\_MCSR, page 0,1,5)
- `#define _CAN_ABRQ ((uint8_t) (0x01 << 1))`  
CAN Abort request for mailbox [0] (in \_CAN\_MCSR, page 0,1,5)
- `#define _CAN_RQCP ((uint8_t) (0x01 << 2))`  
CAN Request completed [0] (in \_CAN\_MCSR, page 0,1,5)
- `#define _CAN_TXOK ((uint8_t) (0x01 << 3))`  
CAN Transmission OK [0] (in \_CAN\_MCSR, page 0,1,5)
- `#define _CAN_ALST ((uint8_t) (0x01 << 4))`  
CAN Arbitration lost [0] (in \_CAN\_MCSR, page 0,1,5)
- `#define _CAN_TERR ((uint8_t) (0x01 << 5))`  
CAN Transmission error [0] (in \_CAN\_MCSR, page 0,1,5)
- `#define _CAN_DLC ((uint8_t) (0x0F << 0))`  
CAN Data length code [3:0] (in \_CAN\_MDLCR, page 0,1,5,7)
- `#define _CAN_DLC0 ((uint8_t) (0x01 << 0))`  
CAN Data length code [0] (in \_CAN\_MDLCR, page 0,1,5,7)
- `#define _CAN_DLC1 ((uint8_t) (0x01 << 1))`  
CAN Data length code [1] (in \_CAN\_MDLCR, page 0,1,5,7)
- `#define _CAN_DLC2 ((uint8_t) (0x01 << 2))`  
CAN Data length code [2] (in \_CAN\_MDLCR, page 0,1,5,7)
- `#define _CAN_DLC3 ((uint8_t) (0x01 << 3))`  
CAN Data length code [3] (in \_CAN\_MDLCR, page 0,1,5,7)
- `#define _CAN_TGT ((uint8_t) (0x01 << 7))`  
CAN Transmit global time [0] (in \_CAN\_MDLCR, page 0,1,5,7)
- `#define _CAN_RTR ((uint8_t) (0x01 << 5))`  
CAN Remote transmission request [0] (in \_CAN\_MIDR1, page 0,1,5)
- `#define _CAN_IDE ((uint8_t) (0x01 << 6))`  
CAN Extended identifier [0] (in \_CAN\_MIDR1, page 0,1,5)
- `#define _CAN_EWGF ((uint8_t) (0x01 << 0))`  
CAN Error warning flag [0] (in \_CAN\_ESR, page 6)
- `#define _CAN_EPVF ((uint8_t) (0x01 << 1))`  
CAN Error passive flag [0] (in \_CAN\_ESR, page 6)
- `#define _CAN_BOFF ((uint8_t) (0x01 << 2))`  
CAN Bus off flag [0] (in \_CAN\_ESR, page 6)
- `#define _CAN_LEC ((uint8_t) (0x07 << 4))`  
CAN Last error code [2:0] (in \_CAN\_ESR, page 6)
- `#define _CAN_LEC0 ((uint8_t) (0x01 << 4))`  
CAN Last error code [0] (in \_CAN\_ESR, page 6)
- `#define _CAN_LEC1 ((uint8_t) (0x01 << 5))`  
CAN Last error code [1] (in \_CAN\_ESR, page 6)
- `#define _CAN_LEC2 ((uint8_t) (0x01 << 6))`  
CAN Last error code [3] (in \_CAN\_ESR, page 6)
- `#define _CAN_EWGIE ((uint8_t) (0x01 << 0))`  
CAN Error warning interrupt enable [0] (in \_CAN\_EIER, page 6)
- `#define _CAN_EPVIE ((uint8_t) (0x01 << 1))`  
CAN Error passive interrupt enable [0] (in \_CAN\_EIER, page 6)
- `#define _CAN_BOFIE ((uint8_t) (0x01 << 2))`  
CAN Bus-Off interrupt enable [0] (in \_CAN\_EIER, page 6)
- `#define _CAN_LECIE ((uint8_t) (0x01 << 4))`



- CAN Last error code interrupt enable [0] (in \_CAN\_EIER, page 6)*
  - #define `_CAN_ERRIE` ((uint8\_t) (0x01 << 6))
- CAN Error interrupt enable [0] (in \_CAN\_EIER, page 6)*
  - #define `_CAN_BRP` ((uint8\_t) (0x3F << 0))
- CAN Baud rate prescaler [5:0] (in \_CAN\_BTR1, page 6)*
  - #define `_CAN_BRP0` ((uint8\_t) (0x01 << 0))
- CAN Baud rate prescaler [0] (in \_CAN\_BTR1, page 6)*
  - #define `_CAN_BRP1` ((uint8\_t) (0x01 << 1))
- CAN Baud rate prescaler [1] (in \_CAN\_BTR1, page 6)*
  - #define `_CAN_BRP2` ((uint8\_t) (0x01 << 2))
- CAN Baud rate prescaler [2] (in \_CAN\_BTR1, page 6)*
  - #define `_CAN_BRP3` ((uint8\_t) (0x01 << 3))
- CAN Baud rate prescaler [3] (in \_CAN\_BTR1, page 6)*
  - #define `_CAN_BRP4` ((uint8\_t) (0x01 << 4))
- CAN Baud rate prescaler [4] (in \_CAN\_BTR1, page 6)*
  - #define `_CAN_BRP5` ((uint8\_t) (0x01 << 5))
- CAN Baud rate prescaler [5] (in \_CAN\_BTR1, page 6)*
  - #define `_CAN_SJW` ((uint8\_t) (0x03 << 6))
- CAN Resynchronization jump width [1:0] (in \_CAN\_EIER, page 6)*
  - #define `_CAN_SJW0` ((uint8\_t) (0x01 << 6))
- CAN Resynchronization jump width [0] (in \_CAN\_EIER, page 6)*
  - #define `_CAN_SJW1` ((uint8\_t) (0x01 << 7))
- CAN Resynchronization jump width [1] (in \_CAN\_EIER, page 6)*
  - #define `_CAN_BS1` ((uint8\_t) (0x0F << 0))
- CAN Bit segment 1 [3:0] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS10` ((uint8\_t) (0x01 << 0))
- CAN Bit segment 1 [0] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS11` ((uint8\_t) (0x01 << 1))
- CAN Bit segment 1 [1] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS12` ((uint8\_t) (0x01 << 2))
- CAN Bit segment 1 [2] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS13` ((uint8\_t) (0x01 << 3))
- CAN Bit segment 1 [3] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS2` ((uint8\_t) (0x07 << 4))
- CAN Bit segment 2 [2:0] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS20` ((uint8\_t) (0x01 << 4))
- CAN Bit segment 2 [0] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS21` ((uint8\_t) (0x01 << 5))
- CAN Bit segment 2 [1] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_BS22` ((uint8\_t) (0x01 << 6))
- CAN Bit segment 2 [2] (in \_CAN\_BTR2, page 6)*
  - #define `_CAN_FML0` ((uint8\_t) (0x01 << 0))
- CAN Filter 0 mode low [0] (in \_CAN\_FMR1, page 6)*
  - #define `_CAN_FMH0` ((uint8\_t) (0x01 << 1))
- CAN Filter 0 mode high [0] (in \_CAN\_FMR1, page 6)*
  - #define `_CAN_FML1` ((uint8\_t) (0x01 << 2))
- CAN Filter 1 mode low [0] (in \_CAN\_FMR1, page 6)*
  - #define `_CAN_FMH1` ((uint8\_t) (0x01 << 3))
- CAN Filter 1 mode high [0] (in \_CAN\_FMR1, page 6)*
  - #define `_CAN_FML2` ((uint8\_t) (0x01 << 4))
- CAN Filter 2 mode low [0] (in \_CAN\_FMR1, page 6)*

- `#define _CAN_FMH2 ((uint8_t) (0x01 << 5))`  
*CAN Filter 2 mode high [0] (in \_CAN\_FMR1, page 6)*
- `#define _CAN_FML3 ((uint8_t) (0x01 << 6))`  
*CAN Filter 3 mode low [0] (in \_CAN\_FMR1, page 6)*
- `#define _CAN_FMH3 ((uint8_t) (0x01 << 7))`  
*CAN Filter 3 mode high [0] (in \_CAN\_FMR1, page 6)*
- `#define _CAN_FML4 ((uint8_t) (0x01 << 0))`  
*CAN Filter 4 mode low [0] (in \_CAN\_FMR2, page 6)*
- `#define _CAN_FMH4 ((uint8_t) (0x01 << 1))`  
*CAN Filter 4 mode high [0] (in \_CAN\_FMR2, page 6)*
- `#define _CAN_FML5 ((uint8_t) (0x01 << 2))`  
*CAN Filter 5 mode low [0] (in \_CAN\_FMR2, page 6)*
- `#define _CAN_FMH5 ((uint8_t) (0x01 << 3))`  
*CAN Filter 5 mode high [0] (in \_CAN\_FMR2, page 6)*
- `#define _CAN_FACT0 ((uint8_t) (0x01 << 0))`  
*CAN Filter 0 active [0] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FSC0 ((uint8_t) (0x03 << 1))`  
*CAN Filter 0 scale configuration [1:0] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FSC00 ((uint8_t) (0x01 << 1))`  
*CAN Filter 0 scale configuration [0] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FSC01 ((uint8_t) (0x01 << 2))`  
*CAN Filter 0 scale configuration [1] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FACT1 ((uint8_t) (0x01 << 4))`  
*CAN Filter 1 active [0] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FSC1 ((uint8_t) (0x03 << 5))`  
*CAN Filter 1 scale configuration [1:0] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FSC10 ((uint8_t) (0x01 << 5))`  
*CAN Filter 1 scale configuration [0] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FSC11 ((uint8_t) (0x01 << 6))`  
*CAN Filter 1 scale configuration [1] (in \_CAN\_FCR1, page 6)*
- `#define _CAN_FACT2 ((uint8_t) (0x01 << 0))`  
*CAN Filter 2 active [0] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FSC2 ((uint8_t) (0x03 << 1))`  
*CAN Filter 2 scale configuration [1:0] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FSC20 ((uint8_t) (0x01 << 1))`  
*CAN Filter 2 scale configuration [0] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FSC21 ((uint8_t) (0x01 << 2))`  
*CAN Filter 2 scale configuration [1] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FACT3 ((uint8_t) (0x01 << 4))`  
*CAN Filter 3 active [0] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FSC3 ((uint8_t) (0x03 << 5))`  
*CAN Filter 3 scale configuration [1:0] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FSC30 ((uint8_t) (0x01 << 5))`  
*CAN Filter 3 scale configuration [0] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FSC31 ((uint8_t) (0x01 << 6))`  
*CAN Filter 3 scale configuration [1] (in \_CAN\_FCR2, page 6)*
- `#define _CAN_FACT4 ((uint8_t) (0x01 << 0))`  
*CAN Filter 4 active [0] (in \_CAN\_FCR3, page 6)*
- `#define _CAN_FSC4 ((uint8_t) (0x03 << 1))`  
*CAN Filter 4 scale configuration [1:0] (in \_CAN\_FCR3, page 6)*
- `#define _CAN_FSC40 ((uint8_t) (0x01 << 1))`



- CAN Filter 4 scale configuration [0] (in \_CAN\_FCR3, page 6)*
- #define `_CAN_FSC41` ((uint8\_t) (0x01 << 2))
- CAN Filter 4 scale configuration [1] (in \_CAN\_FCR3, page 6)*
- #define `_CAN_FACT5` ((uint8\_t) (0x01 << 4))
- CAN Filter 5 active [0] (in \_CAN\_FCR2, page 6)*
- #define `_CAN_FSC5` ((uint8\_t) (0x03 << 5))
- CAN Filter 5 scale configuration [1:0] (in \_CAN\_FCR3, page 6)*
- #define `_CAN_FSC50` ((uint8\_t) (0x01 << 5))
- CAN Filter 5 scale configuration [0] (in \_CAN\_FCR3, page 6)*
- #define `_CAN_FSC51` ((uint8\_t) (0x01 << 6))
- CAN Filter 5 scale configuration [1] (in \_CAN\_FCR3, page 6)*
- #define `_CFG_SFR`(CFG\_t, CFG\_AddressBase)
- CFG struct/bit access.*
- #define `_CFG_GCR_SFR`(uint8\_t, CFG\_AddressBase+0x00)
- Global configuration register (CFG\_GCR)*
- #define `_CFG_GCR_RESET_VALUE` ((uint8\_t)0x00)
- #define `_CFG_SWD` ((uint8\_t) (0x01 << 0))
- SWIM disable [0].*
- #define `_CFG_AL` ((uint8\_t) (0x01 << 1))
- Activation level [0].*
- #define `_ITC_SFR`(ITC\_t, ITC\_AddressBase)
- ITC struct/bit access.*
- #define `_ITC_SPR1_SFR`(uint8\_t, ITC\_AddressBase+0x00)
- Interrupt priority register 1/8.*
- #define `_ITC_SPR2_SFR`(uint8\_t, ITC\_AddressBase+0x01)
- Interrupt priority register 2/8.*
- #define `_ITC_SPR3_SFR`(uint8\_t, ITC\_AddressBase+0x02)
- Interrupt priority register 3/8.*
- #define `_ITC_SPR4_SFR`(uint8\_t, ITC\_AddressBase+0x03)
- Interrupt priority register 4/8.*
- #define `_ITC_SPR5_SFR`(uint8\_t, ITC\_AddressBase+0x04)
- Interrupt priority register 5/8.*
- #define `_ITC_SPR6_SFR`(uint8\_t, ITC\_AddressBase+0x05)
- Interrupt priority register 6/8.*
- #define `_ITC_SPR7_SFR`(uint8\_t, ITC\_AddressBase+0x06)
- Interrupt priority register 7/8.*
- #define `_ITC_SPR8_SFR`(uint8\_t, ITC\_AddressBase+0x07)
- Interrupt priority register 8/8.*
- #define `_ITC_SPR1_RESET_VALUE` ((uint8\_t) 0xFF)
- Interrupt priority register 1/8 reset value.*
- #define `_ITC_SPR2_RESET_VALUE` ((uint8\_t) 0xFF)
- Interrupt priority register 2/8 reset value.*
- #define `_ITC_SPR3_RESET_VALUE` ((uint8\_t) 0xFF)
- Interrupt priority register 3/8 reset value.*
- #define `_ITC_SPR4_RESET_VALUE` ((uint8\_t) 0xFF)
- Interrupt priority register 4/8 reset value.*
- #define `_ITC_SPR5_RESET_VALUE` ((uint8\_t) 0xFF)
- Interrupt priority register 5/8 reset value.*
- #define `_ITC_SPR6_RESET_VALUE` ((uint8\_t) 0xFF)
- Interrupt priority register 6/8 reset value.*
- #define `_ITC_SPR7_RESET_VALUE` ((uint8\_t) 0xFF)

- Interrupt priority register 7/8 reset value.*
- `#define _ITC_SPR8_RESET_VALUE ((uint8_t) 0x0F)`
- Interrupt priority register 8/8 reset value.*
- `#define _ITC_VECT1SPR ((uint8_t) (0x03 << 2))`
- ITC interrupt priority vector 1 [1:0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT1SPR0 ((uint8_t) (0x01 << 2))`
- ITC interrupt priority vector 1 [0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT1SPR1 ((uint8_t) (0x01 << 3))`
- ITC interrupt priority vector 1 [1] (in \_ITC\_SPR1)*
- `#define _ITC_VECT2SPR ((uint8_t) (0x03 << 4))`
- ITC interrupt priority vector 2 [1:0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT2SPR0 ((uint8_t) (0x01 << 4))`
- ITC interrupt priority vector 2 [0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT2SPR1 ((uint8_t) (0x01 << 5))`
- ITC interrupt priority vector 2 [1] (in \_ITC\_SPR1)*
- `#define _ITC_VECT3SPR ((uint8_t) (0x03 << 6))`
- ITC interrupt priority vector 3 [1:0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT3SPR0 ((uint8_t) (0x01 << 6))`
- ITC interrupt priority vector 3 [0] (in \_ITC\_SPR1)*
- `#define _ITC_VECT3SPR1 ((uint8_t) (0x01 << 7))`
- ITC interrupt priority vector 3 [1] (in \_ITC\_SPR1)*
- `#define _ITC_VECT4SPR ((uint8_t) (0x03 << 0))`
- ITC interrupt priority vector 4 [1:0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT4SPR0 ((uint8_t) (0x01 << 0))`
- ITC interrupt priority vector 4 [0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT4SPR1 ((uint8_t) (0x01 << 1))`
- ITC interrupt priority vector 4 [1] (in \_ITC\_SPR2)*
- `#define _ITC_VECT5SPR ((uint8_t) (0x03 << 2))`
- ITC interrupt priority vector 5 [1:0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT5SPR0 ((uint8_t) (0x01 << 2))`
- ITC interrupt priority vector 5 [0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT5SPR1 ((uint8_t) (0x01 << 3))`
- ITC interrupt priority vector 5 [1] (in \_ITC\_SPR2)*
- `#define _ITC_VECT6SPR ((uint8_t) (0x03 << 4))`
- ITC interrupt priority vector 6 [1:0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT6SPR0 ((uint8_t) (0x01 << 4))`
- ITC interrupt priority vector 6 [0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT6SPR1 ((uint8_t) (0x01 << 5))`
- ITC interrupt priority vector 6 [1] (in \_ITC\_SPR2)*
- `#define _ITC_VECT7SPR ((uint8_t) (0x03 << 6))`
- ITC interrupt priority vector 7 [1:0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT7SPR0 ((uint8_t) (0x01 << 6))`
- ITC interrupt priority vector 7 [0] (in \_ITC\_SPR2)*
- `#define _ITC_VECT7SPR1 ((uint8_t) (0x01 << 7))`
- ITC interrupt priority vector 7 [1] (in \_ITC\_SPR2)*
- `#define _ITC_VECT8SPR ((uint8_t) (0x03 << 0))`
- ITC interrupt priority vector 8 [1:0] (in \_ITC\_SPR3)*
- `#define _ITC_VECT8SPR0 ((uint8_t) (0x01 << 0))`
- ITC interrupt priority vector 8 [0] (in \_ITC\_SPR3)*
- `#define _ITC_VECT8SPR1 ((uint8_t) (0x01 << 1))`
- ITC interrupt priority vector 8 [1] (in \_ITC\_SPR3)*

- #define `_ITC_VECT9SPR` ((uint8\_t) (0x03 << 2))  
*ITC interrupt priority vector 9 [1:0] (in \_ITC\_SPR3)*
- #define `_ITC_VECT9SPR0` ((uint8\_t) (0x01 << 2))  
*ITC interrupt priority vector 9 [0] (in \_ITC\_SPR3)*
- #define `_ITC_VECT9SPR1` ((uint8\_t) (0x01 << 3))  
*ITC interrupt priority vector 9 [1] (in \_ITC\_SPR3)*
- #define `_ITC_VECT10SPR` ((uint8\_t) (0x03 << 4))  
*ITC interrupt priority vector 10 [1:0] (in \_ITC\_SPR3)*
- #define `_ITC_VECT10SPR0` ((uint8\_t) (0x01 << 4))  
*ITC interrupt priority vector 10 [0] (in \_ITC\_SPR3)*
- #define `_ITC_VECT10SPR1` ((uint8\_t) (0x01 << 5))  
*ITC interrupt priority vector 10 [1] (in \_ITC\_SPR3)*
- #define `_ITC_VECT11SPR` ((uint8\_t) (0x03 << 6))  
*ITC interrupt priority vector 11 [1:0] (in \_ITC\_SPR3)*
- #define `_ITC_VECT11SPR0` ((uint8\_t) (0x01 << 6))  
*ITC interrupt priority vector 11 [0] (in \_ITC\_SPR3)*
- #define `_ITC_VECT11SPR1` ((uint8\_t) (0x01 << 7))  
*ITC interrupt priority vector 11 [1] (in \_ITC\_SPR3)*
- #define `_ITC_VECT12SPR` ((uint8\_t) (0x03 << 0))  
*ITC interrupt priority vector 12 [1:0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT12SPR0` ((uint8\_t) (0x01 << 0))  
*ITC interrupt priority vector 12 [0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT12SPR1` ((uint8\_t) (0x01 << 1))  
*ITC interrupt priority vector 12 [1] (in \_ITC\_SPR4)*
- #define `_ITC_VECT13SPR` ((uint8\_t) (0x03 << 2))  
*ITC interrupt priority vector 13 [1:0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT13SPR0` ((uint8\_t) (0x01 << 2))  
*ITC interrupt priority vector 13 [0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT13SPR1` ((uint8\_t) (0x01 << 3))  
*ITC interrupt priority vector 13 [1] (in \_ITC\_SPR4)*
- #define `_ITC_VECT14SPR` ((uint8\_t) (0x03 << 4))  
*ITC interrupt priority vector 14 [1:0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT14SPR0` ((uint8\_t) (0x01 << 4))  
*ITC interrupt priority vector 14 [0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT14SPR1` ((uint8\_t) (0x01 << 5))  
*ITC interrupt priority vector 14 [1] (in \_ITC\_SPR4)*
- #define `_ITC_VECT15SPR` ((uint8\_t) (0x03 << 6))  
*ITC interrupt priority vector 15 [1:0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT15SPR0` ((uint8\_t) (0x01 << 6))  
*ITC interrupt priority vector 15 [0] (in \_ITC\_SPR4)*
- #define `_ITC_VECT15SPR1` ((uint8\_t) (0x01 << 7))  
*ITC interrupt priority vector 15 [1] (in \_ITC\_SPR4)*
- #define `_ITC_VECT16SPR` ((uint8\_t) (0x03 << 0))  
*ITC interrupt priority vector 16 [1:0] (in \_ITC\_SPR5)*
- #define `_ITC_VECT16SPR0` ((uint8\_t) (0x01 << 0))  
*ITC interrupt priority vector 16 [0] (in \_ITC\_SPR5)*
- #define `_ITC_VECT16SPR1` ((uint8\_t) (0x01 << 1))  
*ITC interrupt priority vector 16 [1] (in \_ITC\_SPR5)*
- #define `_ITC_VECT17SPR` ((uint8\_t) (0x03 << 2))  
*ITC interrupt priority vector 17 [1:0] (in \_ITC\_SPR5)*
- #define `_ITC_VECT17SPR0` ((uint8\_t) (0x01 << 2))

```

    ITC interrupt priority vector 17 [0] (in _ITC_SPR5)
    • #define _ITC_VECT17SPR1 ((uint8_t) (0x01 << 3))
    ITC interrupt priority vector 17 [1] (in _ITC_SPR5)
    • #define _ITC_VECT18SPR ((uint8_t) (0x03 << 4))
    ITC interrupt priority vector 18 [1:0] (in _ITC_SPR5)
    • #define _ITC_VECT18SPR0 ((uint8_t) (0x01 << 4))
    ITC interrupt priority vector 18 [0] (in _ITC_SPR5)
    • #define _ITC_VECT18SPR1 ((uint8_t) (0x01 << 5))
    ITC interrupt priority vector 18 [1] (in _ITC_SPR5)
    • #define _ITC_VECT19SPR ((uint8_t) (0x03 << 6))
    ITC interrupt priority vector 19 [1:0] (in _ITC_SPR5)
    • #define _ITC_VECT19SPR0 ((uint8_t) (0x01 << 6))
    ITC interrupt priority vector 19 [0] (in _ITC_SPR5)
    • #define _ITC_VECT19SPR1 ((uint8_t) (0x01 << 7))
    ITC interrupt priority vector 19 [1] (in _ITC_SPR5)
    • #define _ITC_VECT20SPR ((uint8_t) (0x03 << 0))
    ITC interrupt priority vector 20 [1:0] (in _ITC_SPR6)
    • #define _ITC_VECT20SPR0 ((uint8_t) (0x01 << 0))
    ITC interrupt priority vector 20 [0] (in _ITC_SPR6)
    • #define _ITC_VECT20SPR1 ((uint8_t) (0x01 << 1))
    ITC interrupt priority vector 20 [1] (in _ITC_SPR6)
    • #define _ITC_VECT21SPR ((uint8_t) (0x03 << 2))
    ITC interrupt priority vector 21 [1:0] (in _ITC_SPR6)
    • #define _ITC_VECT21SPR0 ((uint8_t) (0x01 << 2))
    ITC interrupt priority vector 21 [0] (in _ITC_SPR6)
    • #define _ITC_VECT21SPR1 ((uint8_t) (0x01 << 3))
    ITC interrupt priority vector 21 [1] (in _ITC_SPR6)
    • #define _ITC_VECT22SPR ((uint8_t) (0x03 << 4))
    ITC interrupt priority vector 22 [1:0] (in _ITC_SPR6)
    • #define _ITC_VECT22SPR0 ((uint8_t) (0x01 << 4))
    ITC interrupt priority vector 22 [0] (in _ITC_SPR6)
    • #define _ITC_VECT22SPR1 ((uint8_t) (0x01 << 5))
    ITC interrupt priority vector 22 [1] (in _ITC_SPR6)
    • #define _ITC_VECT23SPR ((uint8_t) (0x03 << 6))
    ITC interrupt priority vector 23 [1:0] (in _ITC_SPR6)
    • #define _ITC_VECT23SPR0 ((uint8_t) (0x01 << 6))
    ITC interrupt priority vector 23 [0] (in _ITC_SPR6)
    • #define _ITC_VECT23SPR1 ((uint8_t) (0x01 << 7))
    ITC interrupt priority vector 23 [1] (in _ITC_SPR6)
    • #define _ITC_VECT24SPR ((uint8_t) (0x03 << 0))
    ITC interrupt priority vector 24 [1:0] (in _ITC_SPR7)
    • #define _ITC_VECT24SPR0 ((uint8_t) (0x01 << 0))
    ITC interrupt priority vector 24 [0] (in _ITC_SPR7)
    • #define _ITC_VECT24SPR1 ((uint8_t) (0x01 << 1))
    ITC interrupt priority vector 24 [1] (in _ITC_SPR7)
    • #define _ITC_VECT25SPR ((uint8_t) (0x03 << 2))
    ITC interrupt priority vector 25 [1:0] (in _ITC_SPR7)
    • #define _ITC_VECT25SPR0 ((uint8_t) (0x01 << 2))
    ITC interrupt priority vector 25 [0] (in _ITC_SPR7)
    • #define _ITC_VECT25SPR1 ((uint8_t) (0x01 << 3))
    ITC interrupt priority vector 25 [1] (in _ITC_SPR7)

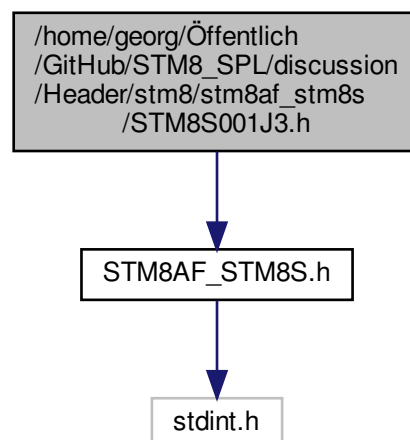
```

- `#define _ITC_VECT26SPR ((uint8_t) (0x03 << 4))`  
*ITC interrupt priority vector 26 [1:0] (in \_ITC\_SPR7)*
- `#define _ITC_VECT26SPR0 ((uint8_t) (0x01 << 4))`  
*ITC interrupt priority vector 26 [0] (in \_ITC\_SPR7)*
- `#define _ITC_VECT26SPR1 ((uint8_t) (0x01 << 5))`  
*ITC interrupt priority vector 26 [1] (in \_ITC\_SPR7)*
- `#define _ITC_VECT27SPR ((uint8_t) (0x03 << 6))`  
*ITC interrupt priority vector 27 [1:0] (in \_ITC\_SPR7)*
- `#define _ITC_VECT27SPR0 ((uint8_t) (0x01 << 6))`  
*ITC interrupt priority vector 27 [0] (in \_ITC\_SPR7)*
- `#define _ITC_VECT27SPR1 ((uint8_t) (0x01 << 7))`  
*ITC interrupt priority vector 27 [1] (in \_ITC\_SPR7)*
- `#define _ITC_VECT28SPR ((uint8_t) (0x03 << 0))`  
*ITC interrupt priority vector 28 [1:0] (in \_ITC\_SPR8)*
- `#define _ITC_VECT28SPR0 ((uint8_t) (0x01 << 0))`  
*ITC interrupt priority vector 28 [0] (in \_ITC\_SPR8)*
- `#define _ITC_VECT28SPR1 ((uint8_t) (0x01 << 1))`  
*ITC interrupt priority vector 28 [1] (in \_ITC\_SPR8)*
- `#define _ITC_VECT29SPR ((uint8_t) (0x03 << 2))`  
*ITC interrupt priority vector 29 [1:0] (in \_ITC\_SPR8)*
- `#define _ITC_VECT29SPR0 ((uint8_t) (0x01 << 2))`  
*ITC interrupt priority vector 29 [0] (in \_ITC\_SPR8)*
- `#define _ITC_VECT29SPR1 ((uint8_t) (0x01 << 3))`  
*ITC interrupt priority vector 29 [1] (in \_ITC\_SPR8)*

### 7.33 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S001J3.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S001J3.h:



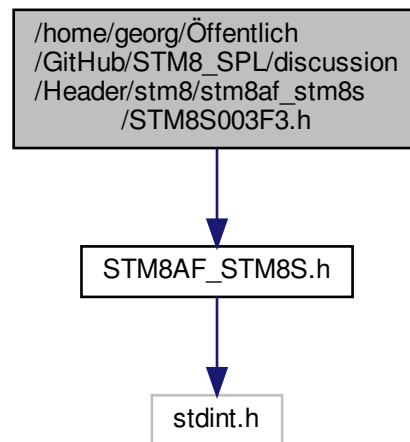
## Macros

- #define [STM8S001J3](#)
- #define [STM8S001](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 8\*1024
- #define [STM8\\_RAM\\_SIZE](#) 1\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 128
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.34 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8S003F3.h](#) File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S003F3.h:



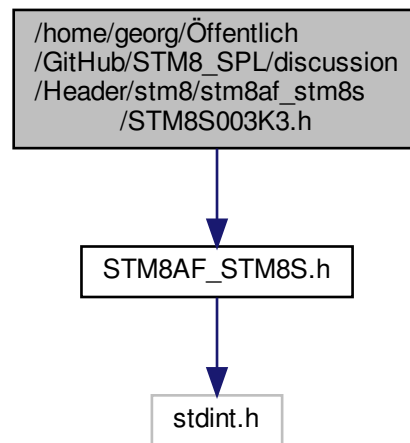
## Macros

- #define [STM8S003F3](#)
- #define [STM8S003](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 8\*1024
- #define [STM8\\_RAM\\_SIZE](#) 1\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 128
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.35 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S003K3.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S003K3.h:



### Macros

- `#define STM8S003K3`
- `#define STM8S003`
- `#define STM8_PFLASH_SIZE 8*1024`
- `#define STM8_RAM_SIZE 1*1024`
- `#define STM8_EEPROM_SIZE 128`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`

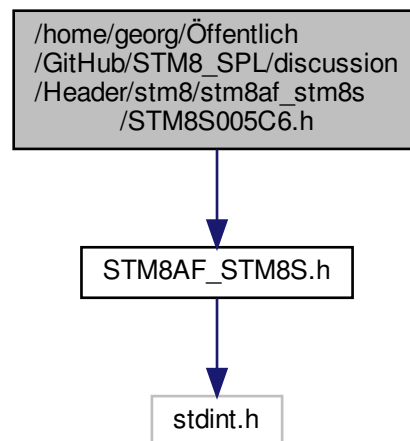


- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.36 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S005C6.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S005C6.h:



### Macros

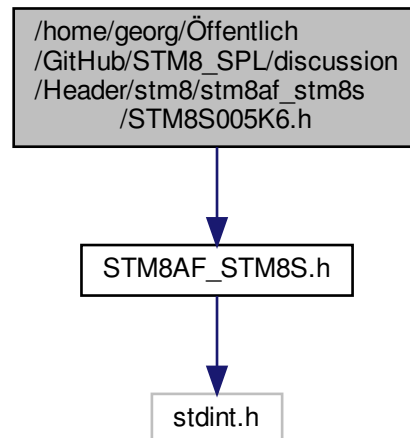
- #define [STM8S005C6](#)
- #define [STM8S005](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 32\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 128
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A

- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`

### 7.37 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S005K6.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S005K6.h:



#### Macros

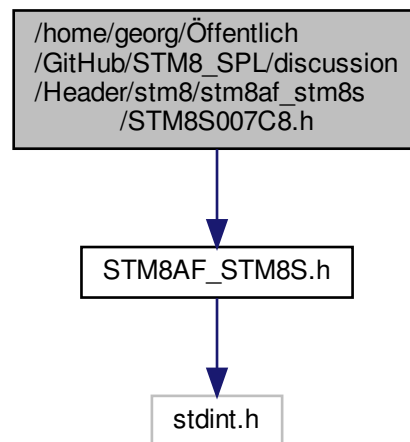
- `#define STM8S005K6`
- `#define STM8S005`
- `#define STM8_PFLASH_SIZE 32*1024`

- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 128
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.38 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S007C8.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S007C8.h:



## Macros

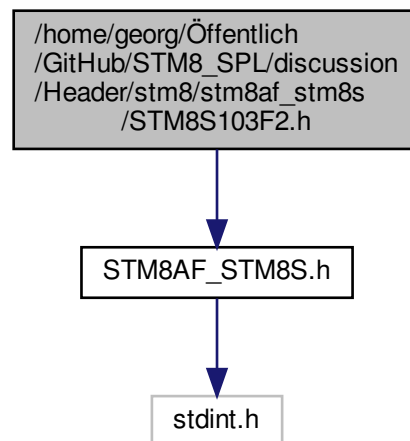
- `#define STM8S007C8`
- `#define STM8S007`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 128`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`

- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90

## 7.39 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S103F2.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S103F2.h:



### Macros

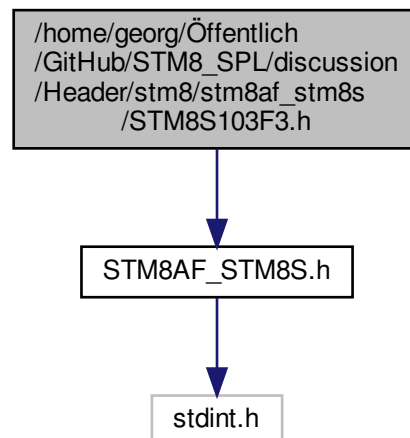
- #define [STM8S103F2](#)
- #define [STM8S103](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 4\*1024
- #define [STM8\\_RAM\\_SIZE](#) 1\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 640
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0

- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x4865`

## 7.40 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S103F3.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S103F3.h:



### Macros

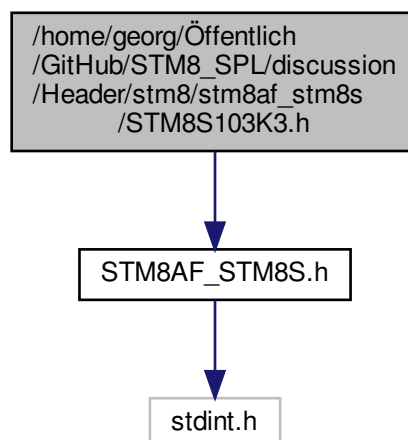
- `#define STM8S103F3`
- `#define STM8S103`
- `#define STM8_PFLASH_SIZE 8*1024`
- `#define STM8_RAM_SIZE 1*1024`

- #define [STM8\\_EEPROM\\_SIZE](#) 640
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x4865

## 7.41 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S103K3.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S103K3.h:



## Macros

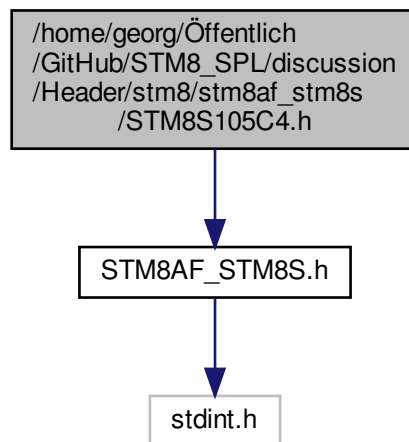
- `#define STM8S103K3`
- `#define STM8S103`
- `#define STM8_PFLASH_SIZE 8*1024`
- `#define STM8_RAM_SIZE 1*1024`
- `#define STM8_EEPROM_SIZE 640`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC1_AddressBase 0x53E0`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x4865`

## 7.42 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105C4.h` File Reference

```
#include "STM8AF_STM8S.h"
```



Include dependency graph for STM8S105C4.h:



## Macros

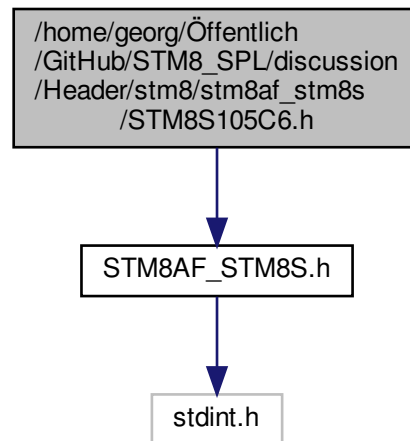
- `#define STM8S105C4`
- `#define STM8S105`
- `#define STM8_PFLASH_SIZE 16*1024`
- `#define STM8_RAM_SIZE 2*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`

- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.43 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/](#)STM8S105C6.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S105C6.h:



### Macros

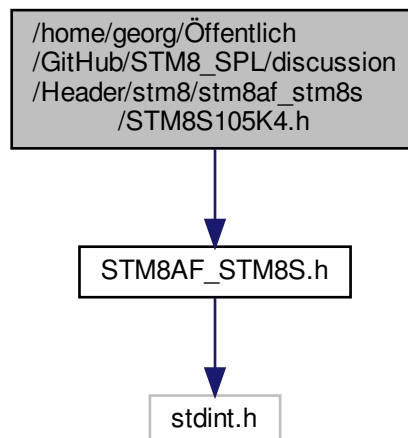
- #define [STM8S105C6](#)
- #define [STM8S105](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 32\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 1024
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3

- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.44 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S105K4.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S105K4.h:



### Macros

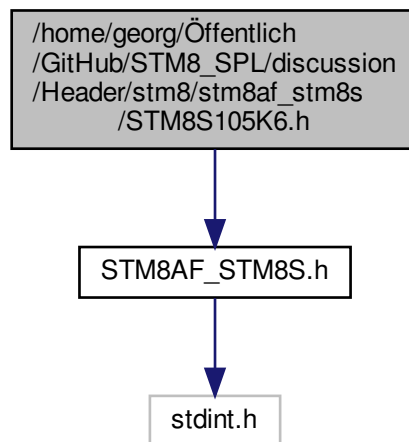
- #define [STM8S105K4](#)
- #define [STM8S105](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 16\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024

- #define `STM8_EEPROM_SIZE` 1024
- #define `OPT_AddressBase` 0x4800
- #define `PORTA_AddressBase` 0x5000
- #define `PORTB_AddressBase` 0x5005
- #define `PORTC_AddressBase` 0x500A
- #define `PORTD_AddressBase` 0x500F
- #define `PORTE_AddressBase` 0x5014
- #define `PORTF_AddressBase` 0x5019
- #define `PORTG_AddressBase` 0x501E
- #define `FLASH_AddressBase` 0x505A
- #define `EXTI_AddressBase` 0x50A0
- #define `RST_AddressBase` 0x50B3
- #define `CLK_AddressBase` 0x50C0
- #define `WWDG_AddressBase` 0x50D1
- #define `IWDG_AddressBase` 0x50E0
- #define `AWU_AddressBase` 0x50F0
- #define `BEEP_AddressBase` 0x50F3
- #define `SPI_AddressBase` 0x5200
- #define `I2C_AddressBase` 0x5210
- #define `UART2_AddressBase` 0x5240
- #define `TIM1_AddressBase` 0x5250
- #define `TIM2_AddressBase` 0x5300
- #define `TIM3_AddressBase` 0x5320
- #define `TIM4_AddressBase` 0x5340
- #define `ADC1_AddressBase` 0x53E0
- #define `CFG_AddressBase` 0x7F60
- #define `ITC_AddressBase` 0x7F70
- #define `DM_AddressBase` 0x7F90
- #define `UID_AddressBase` 0x48CD

## 7.45 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S105K6.h` File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S105K6.h:



## Macros

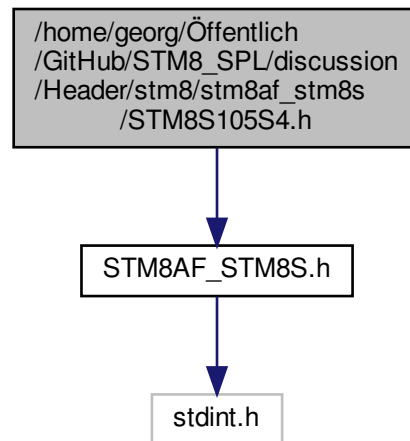
- `#define STM8S105K6`
- `#define STM8S105`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 2*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART2_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`
- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`

- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.46 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/](#)STM8S105S4.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S105S4.h:



### Macros

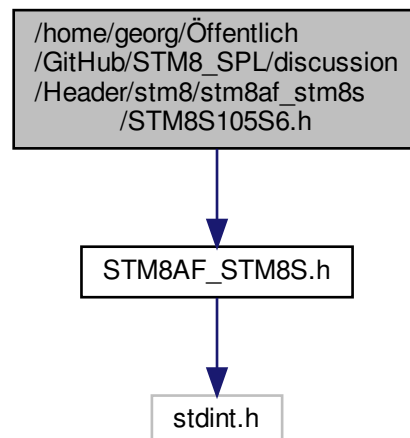
- #define [STM8S105S4](#)
- #define [STM8S105](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 16\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 1024
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3

- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART2\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC1\\_AddressBase](#) 0x53E0
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.47 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S105S6.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S105S6.h:



### Macros

- #define [STM8S105S6](#)
- #define [STM8S105](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 32\*1024
- #define [STM8\\_RAM\\_SIZE](#) 2\*1024

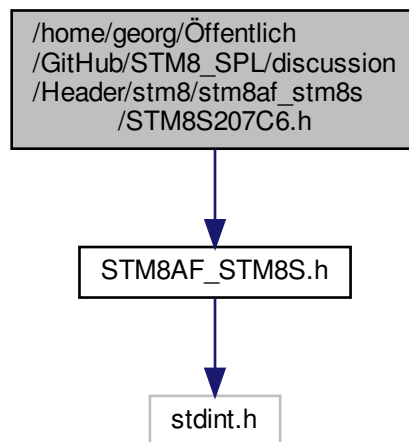
- #define `STM8_EEPROM_SIZE` 1024
- #define `OPT_AddressBase` 0x4800
- #define `PORTA_AddressBase` 0x5000
- #define `PORTB_AddressBase` 0x5005
- #define `PORTC_AddressBase` 0x500A
- #define `PORTD_AddressBase` 0x500F
- #define `PORTE_AddressBase` 0x5014
- #define `PORTF_AddressBase` 0x5019
- #define `PORTG_AddressBase` 0x501E
- #define `FLASH_AddressBase` 0x505A
- #define `EXTI_AddressBase` 0x50A0
- #define `RST_AddressBase` 0x50B3
- #define `CLK_AddressBase` 0x50C0
- #define `WWDG_AddressBase` 0x50D1
- #define `IWDG_AddressBase` 0x50E0
- #define `AWU_AddressBase` 0x50F0
- #define `BEEP_AddressBase` 0x50F3
- #define `SPI_AddressBase` 0x5200
- #define `I2C_AddressBase` 0x5210
- #define `UART2_AddressBase` 0x5240
- #define `TIM1_AddressBase` 0x5250
- #define `TIM2_AddressBase` 0x5300
- #define `TIM3_AddressBase` 0x5320
- #define `TIM4_AddressBase` 0x5340
- #define `ADC1_AddressBase` 0x53E0
- #define `CFG_AddressBase` 0x7F60
- #define `ITC_AddressBase` 0x7F70
- #define `DM_AddressBase` 0x7F90
- #define `UID_AddressBase` 0x48CD

## 7.48 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207C6.h` File Reference

```
#include "STM8AF_STM8S.h"
```



Include dependency graph for STM8S207C6.h:



## Macros

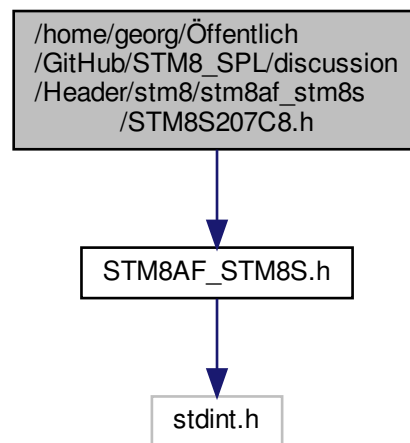
- `#define STM8S207C6`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.49 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207C8.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207C8.h:



### Macros

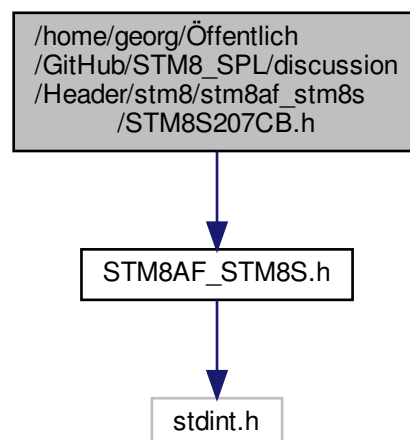
- `#define STM8S207C8`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1536`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.50 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207CB.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207CB.h:



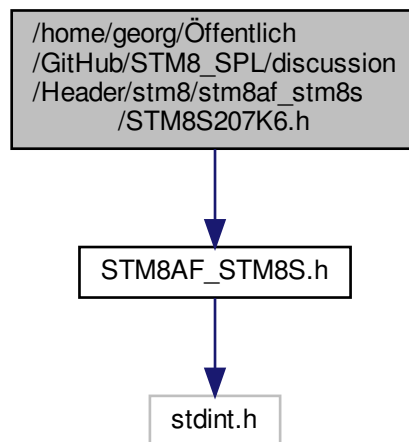
## Macros

- #define STM8S207CB
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

## 7.51 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8S207K6.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207K6.h) File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207K6.h:



## Macros

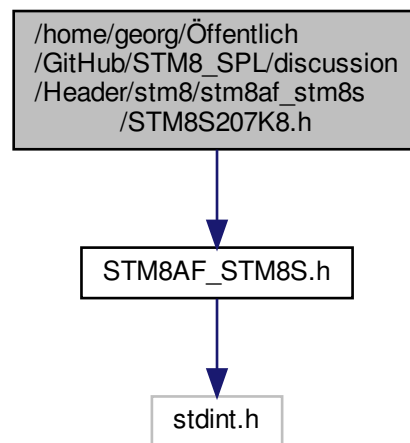
- `#define STM8S207K6`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`
- `#define PORTH_AddressBase 0x5023`
- `#define PORTI_AddressBase 0x5028`
- `#define FLASH_AddressBase 0x505A`
- `#define EXTI_AddressBase 0x50A0`
- `#define RST_AddressBase 0x50B3`
- `#define CLK_AddressBase 0x50C0`
- `#define WWDG_AddressBase 0x50D1`
- `#define IWDG_AddressBase 0x50E0`
- `#define AWU_AddressBase 0x50F0`
- `#define BEEP_AddressBase 0x50F3`
- `#define SPI_AddressBase 0x5200`
- `#define I2C_AddressBase 0x5210`
- `#define UART1_AddressBase 0x5230`
- `#define UART3_AddressBase 0x5240`
- `#define TIM1_AddressBase 0x5250`

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.52 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207K8.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207K8.h:



### Macros

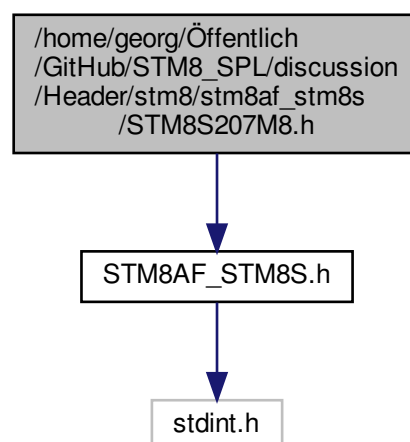
- `#define STM8S207K8`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 64*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.53 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207M8.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207M8.h:



## Macros

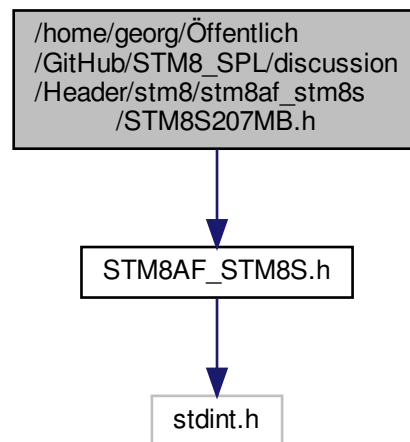
- #define STM8S207M8
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

## 7.54 [/home/georg/Öffentlich/GitHub/STM8\\_SPL/discussion/Header/stm8/stm8af\\_stm8s/STM8S207MB.h](/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S207MB.h) File Reference

```
#include "STM8AF_STM8S.h"
```



Include dependency graph for STM8S207MB.h:



## Macros

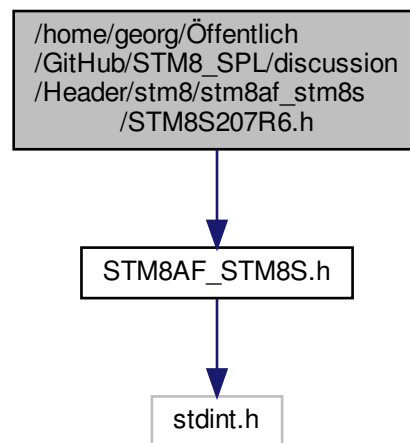
- #define STM8S207MB
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.55 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207R6.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207R6.h:



### Macros

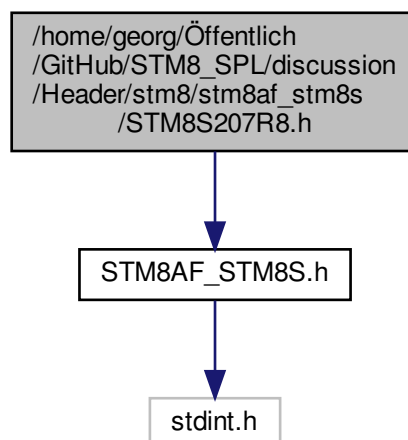
- `#define STM8S207R6`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.56 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207R8.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207R8.h:



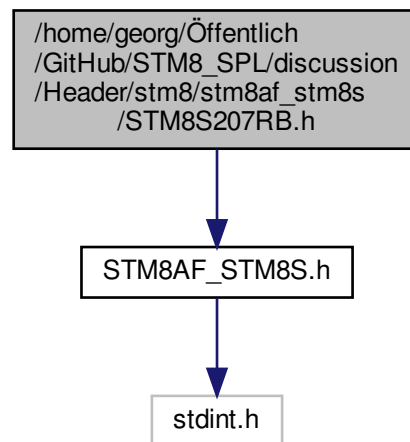
## Macros

- #define STM8S207R8
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

## 7.57 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207RB.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207RB.h:



## Macros

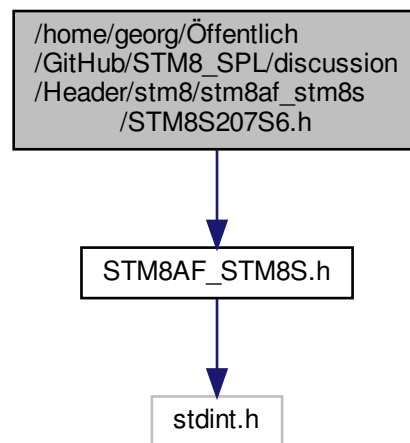
- #define [STM8S207RB](#)
- #define [STM8S207](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 128\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.58 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207S6.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207S6.h:



### Macros

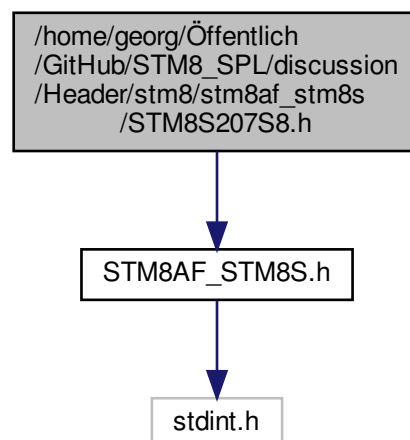
- `#define STM8S207S6`
- `#define STM8S207`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1024`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.59 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207S8.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S207S8.h:



## Macros

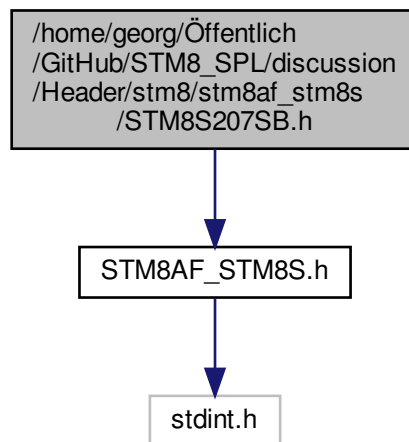
- #define STM8S207S8
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

## 7.60 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S207SB.h File Reference

```
#include "STM8AF_STM8S.h"
```



Include dependency graph for STM8S207SB.h:



## Macros

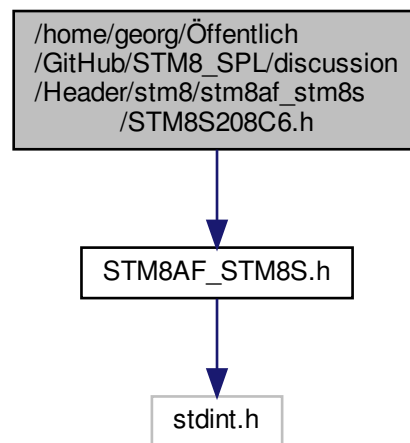
- #define STM8S207SB
- #define STM8S207
- #define STM8\_PFLASH\_SIZE 128\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.61 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208C6.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208C6.h:



### Macros

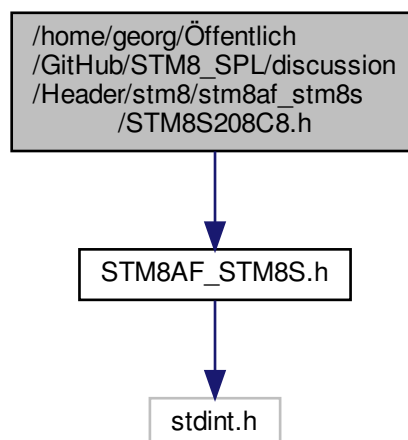
- `#define STM8S208C6`
- `#define STM8S208`
- `#define STM8_PFLASH_SIZE 32*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`
- `#define PORTG_AddressBase 0x501E`

- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.62 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208C8.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208C8.h:



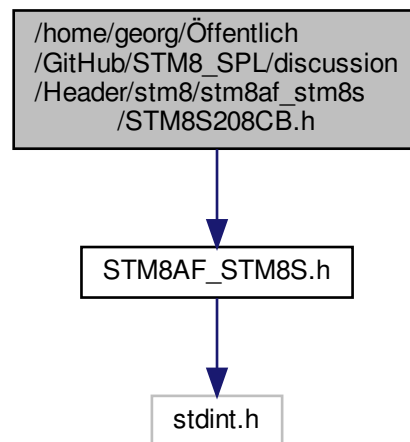
## Macros

- #define STM8S208C8
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

## 7.63 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208CB.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208CB.h:



## Macros

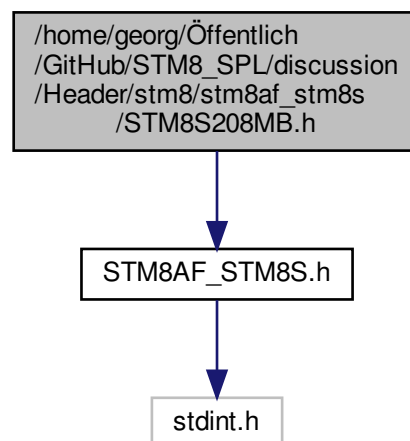
- #define [STM8S208CB](#)
- #define [STM8S208](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 128\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.64 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208MB.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208MB.h:



### Macros

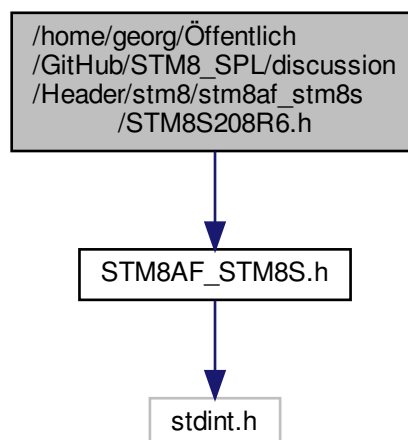
- `#define STM8S208MB`
- `#define STM8S208`
- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`

- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.65 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208R6.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208R6.h:



## Macros

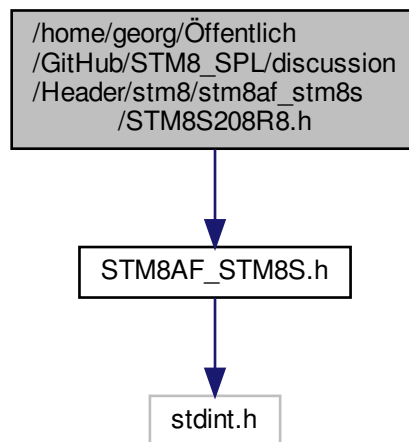
- #define STM8S208R6
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 2048
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

## 7.66 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208R8.h File Reference

```
#include "STM8AF_STM8S.h"
```



Include dependency graph for STM8S208R8.h:



## Macros

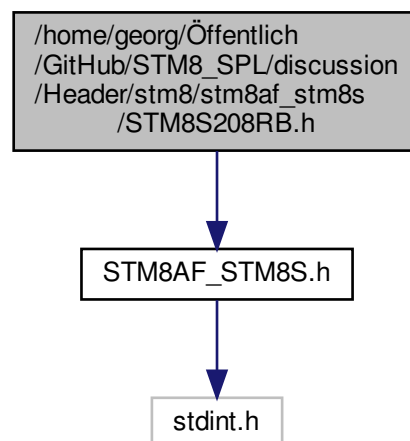
- #define [STM8S208R8](#)
- #define [STM8S208](#)
- #define [STM8\\_PFLASH\\_SIZE](#) 64\*1024
- #define [STM8\\_RAM\\_SIZE](#) 6\*1024
- #define [STM8\\_EEPROM\\_SIZE](#) 2048
- #define [OPT\\_AddressBase](#) 0x4800
- #define [PORTA\\_AddressBase](#) 0x5000
- #define [PORTB\\_AddressBase](#) 0x5005
- #define [PORTC\\_AddressBase](#) 0x500A
- #define [PORTD\\_AddressBase](#) 0x500F
- #define [PORTE\\_AddressBase](#) 0x5014
- #define [PORTF\\_AddressBase](#) 0x5019
- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.67 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208RB.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208RB.h:



### Macros

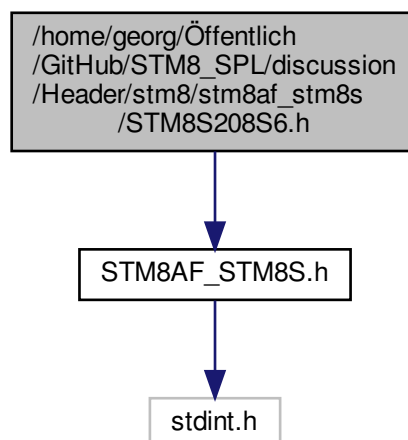
- `#define STM8S208RB`
- `#define STM8S208`
- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 2048`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`

- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.68 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208S6.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208S6.h:



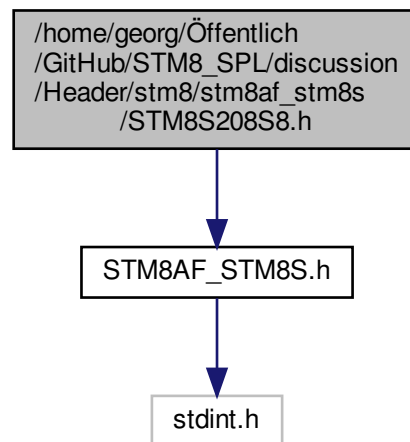
## Macros

- #define STM8S208S6
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 32\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250
- #define TIM2\_AddressBase 0x5300
- #define TIM3\_AddressBase 0x5320
- #define TIM4\_AddressBase 0x5340
- #define ADC2\_AddressBase 0x5400
- #define CAN\_AddressBase 0x5420
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x48CD

## 7.69 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S208S8.h File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208S8.h:



## Macros

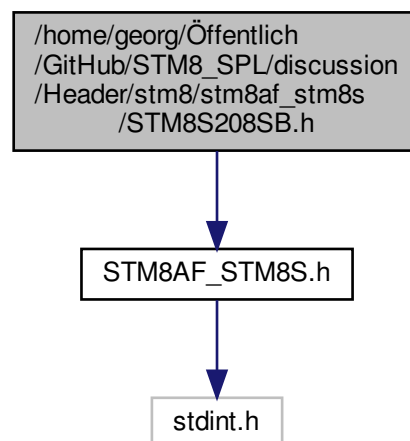
- #define STM8S208S8
- #define STM8S208
- #define STM8\_PFLASH\_SIZE 64\*1024
- #define STM8\_RAM\_SIZE 6\*1024
- #define STM8\_EEPROM\_SIZE 1536
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define PORTG\_AddressBase 0x501E
- #define PORTH\_AddressBase 0x5023
- #define PORTI\_AddressBase 0x5028
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define UART3\_AddressBase 0x5240
- #define TIM1\_AddressBase 0x5250

- `#define TIM2_AddressBase 0x5300`
- `#define TIM3_AddressBase 0x5320`
- `#define TIM4_AddressBase 0x5340`
- `#define ADC2_AddressBase 0x5400`
- `#define CAN_AddressBase 0x5420`
- `#define CFG_AddressBase 0x7F60`
- `#define ITC_AddressBase 0x7F70`
- `#define DM_AddressBase 0x7F90`
- `#define UID_AddressBase 0x48CD`

## 7.70 `/home/georg/Öffentlich/GitHub/STM8_SPL/discussion/Header/stm8/stm8af_stm8s/STM8S208SB.h` File Reference

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S208SB.h:



### Macros

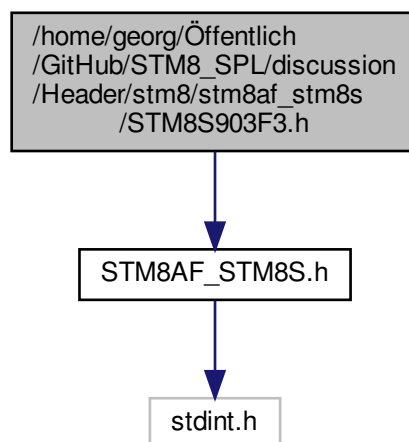
- `#define STM8S208SB`
- `#define STM8S208`
- `#define STM8_PFLASH_SIZE 128*1024`
- `#define STM8_RAM_SIZE 6*1024`
- `#define STM8_EEPROM_SIZE 1536`
- `#define OPT_AddressBase 0x4800`
- `#define PORTA_AddressBase 0x5000`
- `#define PORTB_AddressBase 0x5005`
- `#define PORTC_AddressBase 0x500A`
- `#define PORTD_AddressBase 0x500F`
- `#define PORTE_AddressBase 0x5014`
- `#define PORTF_AddressBase 0x5019`

- #define [PORTG\\_AddressBase](#) 0x501E
- #define [PORTH\\_AddressBase](#) 0x5023
- #define [PORTI\\_AddressBase](#) 0x5028
- #define [FLASH\\_AddressBase](#) 0x505A
- #define [EXTI\\_AddressBase](#) 0x50A0
- #define [RST\\_AddressBase](#) 0x50B3
- #define [CLK\\_AddressBase](#) 0x50C0
- #define [WWDG\\_AddressBase](#) 0x50D1
- #define [IWDG\\_AddressBase](#) 0x50E0
- #define [AWU\\_AddressBase](#) 0x50F0
- #define [BEEP\\_AddressBase](#) 0x50F3
- #define [SPI\\_AddressBase](#) 0x5200
- #define [I2C\\_AddressBase](#) 0x5210
- #define [UART1\\_AddressBase](#) 0x5230
- #define [UART3\\_AddressBase](#) 0x5240
- #define [TIM1\\_AddressBase](#) 0x5250
- #define [TIM2\\_AddressBase](#) 0x5300
- #define [TIM3\\_AddressBase](#) 0x5320
- #define [TIM4\\_AddressBase](#) 0x5340
- #define [ADC2\\_AddressBase](#) 0x5400
- #define [CAN\\_AddressBase](#) 0x5420
- #define [CFG\\_AddressBase](#) 0x7F60
- #define [ITC\\_AddressBase](#) 0x7F70
- #define [DM\\_AddressBase](#) 0x7F90
- #define [UID\\_AddressBase](#) 0x48CD

## 7.71 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S903F3.h File Reference ↩

```
#include "STM8AF_STM8S.h"
```

Include dependency graph for STM8S903F3.h:



## Macros

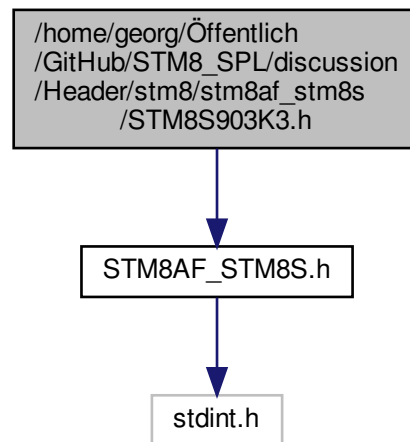
- #define STM8S903F3
- #define STM8S903
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM5\_AddressBase 0x5300
- #define TIM6\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x4865

## 7.72 /home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/Header/stm8/stm8af\_stm8s/STM8S903K3.h File Reference

```
#include "STM8AF_STM8S.h"
```



Include dependency graph for STM8S903K3.h:



## Macros

- #define STM8S903K3
- #define STM8S903
- #define STM8\_PFLASH\_SIZE 8\*1024
- #define STM8\_RAM\_SIZE 1\*1024
- #define STM8\_EEPROM\_SIZE 640
- #define OPT\_AddressBase 0x4800
- #define PORTA\_AddressBase 0x5000
- #define PORTB\_AddressBase 0x5005
- #define PORTC\_AddressBase 0x500A
- #define PORTD\_AddressBase 0x500F
- #define PORTE\_AddressBase 0x5014
- #define PORTF\_AddressBase 0x5019
- #define FLASH\_AddressBase 0x505A
- #define EXTI\_AddressBase 0x50A0
- #define RST\_AddressBase 0x50B3
- #define CLK\_AddressBase 0x50C0
- #define WWDG\_AddressBase 0x50D1
- #define IWDG\_AddressBase 0x50E0
- #define AWU\_AddressBase 0x50F0
- #define BEEP\_AddressBase 0x50F3
- #define SPI\_AddressBase 0x5200
- #define I2C\_AddressBase 0x5210
- #define UART1\_AddressBase 0x5230
- #define TIM1\_AddressBase 0x5250
- #define TIM5\_AddressBase 0x5300
- #define TIM6\_AddressBase 0x5340
- #define ADC1\_AddressBase 0x53E0
- #define CFG\_AddressBase 0x7F60
- #define ITC\_AddressBase 0x7F70
- #define DM\_AddressBase 0x7F90
- #define UID\_AddressBase 0x4865



# Index

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF5268.h,  
[1219](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF5269.h,  
[1220](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF5286.h,  
[1222](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF5288.h,  
[1223](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF5289.h,  
[1224](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF528A.h,  
[1226](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF52A6.h,  
[1227](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF52A8.h,  
[1228](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF52A9.h,  
[1230](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF52AA.h,  
[1231](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6213.h,  
[1232](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6213↔  
A.h, [1234](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6223.h,  
[1235](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6223↔  
A.h, [1236](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6226.h,  
[1237](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6246.h,  
[1238](#)

Header/stm8/stm8af\_stm8s/STM8AF6248.h,  
[1240](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6266.h,  
[1241](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6268.h,  
[1242](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6269.h,  
[1243](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6286.h,  
[1245](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6288.h,  
[1246](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6289.h,  
[1247](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF628A.h,  
[1248](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF62A6.h,  
[1250](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF62A8.h,  
[1251](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF62A9.h,  
[1252](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF62AA.h,  
[1254](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6366.h,  
[1255](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF6388.h,  
[1256](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8AF\_ST↔  
M8S.h, [1258](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8S001J3.h,  
[1331](#)

/home/georg/Öffentlich/GitHub/STM8\_SPL/discussion/↔  
Header/stm8/stm8af\_stm8s/STM8S003F3.h,



- STM8AF\_STM8S, [130](#)
- \_ADC1\_ADON
  - STM8AF\_STM8S, [131](#)
- \_ADC1\_ALIGN
  - STM8AF\_STM8S, [131](#)
- \_ADC1\_AWD
  - STM8AF\_STM8S, [132](#)
- \_ADC1\_AWCRH
  - STM8AF\_STM8S, [131](#)
- \_ADC1\_AWCRL
  - STM8AF\_STM8S, [131](#)
- \_ADC1\_AWCRH\_RESET\_VALUE
  - STM8AF\_STM8S, [131](#)
- \_ADC1\_AWCRL\_RESET\_VALUE
  - STM8AF\_STM8S, [132](#)
- \_ADC1\_AWDIE
  - STM8AF\_STM8S, [132](#)
- \_ADC1\_AWSRH
  - STM8AF\_STM8S, [132](#)
- \_ADC1\_AWSRL
  - STM8AF\_STM8S, [132](#)
- \_ADC1\_CH
  - STM8AF\_STM8S, [133](#)
- \_ADC1\_CH0
  - STM8AF\_STM8S, [133](#)
- \_ADC1\_CH1
  - STM8AF\_STM8S, [133](#)
- \_ADC1\_CH2
  - STM8AF\_STM8S, [133](#)
- \_ADC1\_CH3
  - STM8AF\_STM8S, [133](#)
- \_ADC1\_CONT
  - STM8AF\_STM8S, [134](#)
- \_ADC1\_CR1
  - STM8AF\_STM8S, [134](#)
- \_ADC1\_CR1\_RESET\_VALUE
  - STM8AF\_STM8S, [134](#)
- \_ADC1\_CR2
  - STM8AF\_STM8S, [134](#)
- \_ADC1\_CR2\_RESET\_VALUE
  - STM8AF\_STM8S, [134](#)
- \_ADC1\_CR3
  - STM8AF\_STM8S, [135](#)
- \_ADC1\_CR3\_RESET\_VALUE
  - STM8AF\_STM8S, [135](#)
- \_ADC1\_CSR
  - STM8AF\_STM8S, [135](#)
- \_ADC1\_CSR\_RESET\_VALUE
  - STM8AF\_STM8S, [135](#)
- \_ADC1\_DB0RH
  - STM8AF\_STM8S, [135](#)
- \_ADC1\_DB0RL
  - STM8AF\_STM8S, [136](#)
- \_ADC1\_DB1RH
  - STM8AF\_STM8S, [136](#)
- \_ADC1\_DB1RL
  - STM8AF\_STM8S, [136](#)
- \_ADC1\_DB2RH
  - STM8AF\_STM8S, [136](#)
- \_ADC1\_DB2RL
  - STM8AF\_STM8S, [136](#)
- \_ADC1\_DB3RH
  - STM8AF\_STM8S, [137](#)
- \_ADC1\_DB3RL
  - STM8AF\_STM8S, [137](#)
- \_ADC1\_DB4RH
  - STM8AF\_STM8S, [137](#)
- \_ADC1\_DB4RL
  - STM8AF\_STM8S, [137](#)
- \_ADC1\_DB5RH
  - STM8AF\_STM8S, [137](#)
- \_ADC1\_DB5RL
  - STM8AF\_STM8S, [138](#)
- \_ADC1\_DB6RH
  - STM8AF\_STM8S, [138](#)
- \_ADC1\_DB6RL
  - STM8AF\_STM8S, [138](#)
- \_ADC1\_DB7RH
  - STM8AF\_STM8S, [138](#)
- \_ADC1\_DB7RL
  - STM8AF\_STM8S, [138](#)
- \_ADC1\_DB8RH
  - STM8AF\_STM8S, [139](#)
- \_ADC1\_DB8RL
  - STM8AF\_STM8S, [139](#)
- \_ADC1\_DB9RH
  - STM8AF\_STM8S, [139](#)
- \_ADC1\_DB9RL
  - STM8AF\_STM8S, [139](#)
- \_ADC1\_DBUF
  - STM8AF\_STM8S, [139](#)
- \_ADC1\_DRH
  - STM8AF\_STM8S, [140](#)
- \_ADC1\_DRL
  - STM8AF\_STM8S, [140](#)
- \_ADC1\_EOC
  - STM8AF\_STM8S, [140](#)
- \_ADC1\_EOCIE
  - STM8AF\_STM8S, [140](#)
- \_ADC1\_EXTSEL
  - STM8AF\_STM8S, [140](#)
- \_ADC1\_EXTSEL0
  - STM8AF\_STM8S, [141](#)
- \_ADC1\_EXTSEL1
  - STM8AF\_STM8S, [141](#)
- \_ADC1\_EXTTRIG
  - STM8AF\_STM8S, [141](#)
- \_ADC1\_HTRH
  - STM8AF\_STM8S, [141](#)
- \_ADC1\_HTRL
  - STM8AF\_STM8S, [142](#)
- \_ADC1\_HTRH\_RESET\_VALUE
  - STM8AF\_STM8S, [141](#)
- \_ADC1\_HTRL\_RESET\_VALUE
  - STM8AF\_STM8S, [142](#)
- \_ADC1\_LTRH

STM8AF\_STM8S, [142](#)  
 \_ADC1\_LTRL  
     STM8AF\_STM8S, [142](#)  
 \_ADC1\_LTRH\_RESET\_VALUE  
     STM8AF\_STM8S, [142](#)  
 \_ADC1\_LTRL\_RESET\_VALUE  
     STM8AF\_STM8S, [143](#)  
 \_ADC1\_OVR  
     STM8AF\_STM8S, [143](#)  
 \_ADC1\_SCAN  
     STM8AF\_STM8S, [143](#)  
 \_ADC1\_SPSSEL  
     STM8AF\_STM8S, [143](#)  
 \_ADC1\_SPSSEL0  
     STM8AF\_STM8S, [143](#)  
 \_ADC1\_SPSSEL1  
     STM8AF\_STM8S, [144](#)  
 \_ADC1\_SPSSEL2  
     STM8AF\_STM8S, [144](#)  
 \_ADC1\_TDRH  
     STM8AF\_STM8S, [144](#)  
 \_ADC1\_TDRL  
     STM8AF\_STM8S, [144](#)  
 \_ADC1\_TDRH\_RESET\_VALUE  
     STM8AF\_STM8S, [144](#)  
 \_ADC1\_TDRL\_RESET\_VALUE  
     STM8AF\_STM8S, [145](#)  
 \_ADC2  
     STM8AF\_STM8S, [145](#)  
 \_ADC2\_ADON  
     STM8AF\_STM8S, [145](#)  
 \_ADC2\_ALIGN  
     STM8AF\_STM8S, [145](#)  
 \_ADC2\_CH  
     STM8AF\_STM8S, [145](#)  
 \_ADC2\_CH0  
     STM8AF\_STM8S, [146](#)  
 \_ADC2\_CH1  
     STM8AF\_STM8S, [146](#)  
 \_ADC2\_CH2  
     STM8AF\_STM8S, [146](#)  
 \_ADC2\_CH3  
     STM8AF\_STM8S, [146](#)  
 \_ADC2\_CONT  
     STM8AF\_STM8S, [146](#)  
 \_ADC2\_CR1  
     STM8AF\_STM8S, [147](#)  
 \_ADC2\_CR1\_RESET\_VALUE  
     STM8AF\_STM8S, [147](#)  
 \_ADC2\_CR2  
     STM8AF\_STM8S, [147](#)  
 \_ADC2\_CR2\_RESET\_VALUE  
     STM8AF\_STM8S, [147](#)  
 \_ADC2\_CSR  
     STM8AF\_STM8S, [147](#)  
 \_ADC2\_CSR\_RESET\_VALUE  
     STM8AF\_STM8S, [148](#)  
 \_ADC2\_DRH  
     STM8AF\_STM8S, [148](#)  
 \_ADC2\_DRL  
     STM8AF\_STM8S, [148](#)  
 \_ADC2\_EOC  
     STM8AF\_STM8S, [148](#)  
 \_ADC2\_EOCIE  
     STM8AF\_STM8S, [148](#)  
 \_ADC2\_EXTSEL  
     STM8AF\_STM8S, [149](#)  
 \_ADC2\_EXTSEL0  
     STM8AF\_STM8S, [149](#)  
 \_ADC2\_EXTSEL1  
     STM8AF\_STM8S, [149](#)  
 \_ADC2\_EXTTRIG  
     STM8AF\_STM8S, [149](#)  
 \_ADC2\_SPSSEL  
     STM8AF\_STM8S, [149](#)  
 \_ADC2\_SPSSEL0  
     STM8AF\_STM8S, [150](#)  
 \_ADC2\_SPSSEL1  
     STM8AF\_STM8S, [150](#)  
 \_ADC2\_SPSSEL2  
     STM8AF\_STM8S, [150](#)  
 \_ADC2\_TDRH  
     STM8AF\_STM8S, [150](#)  
 \_ADC2\_TDRL  
     STM8AF\_STM8S, [151](#)  
 \_ADC2\_TDRH\_RESET\_VALUE  
     STM8AF\_STM8S, [150](#)  
 \_ADC2\_TDRL\_RESET\_VALUE  
     STM8AF\_STM8S, [151](#)  
 \_AWU  
     STM8AF\_STM8S, [151](#)  
 \_AWU\_APR  
     STM8AF\_STM8S, [151](#)  
 \_AWU\_APRE  
     STM8AF\_STM8S, [152](#)  
 \_AWU\_APRE0  
     STM8AF\_STM8S, [152](#)  
 \_AWU\_APRE1  
     STM8AF\_STM8S, [152](#)  
 \_AWU\_APRE2  
     STM8AF\_STM8S, [152](#)  
 \_AWU\_APRE3  
     STM8AF\_STM8S, [152](#)  
 \_AWU\_APRE4  
     STM8AF\_STM8S, [153](#)  
 \_AWU\_APRE5  
     STM8AF\_STM8S, [153](#)  
 \_AWU\_APR\_RESET\_VALUE  
     STM8AF\_STM8S, [151](#)  
 \_AWU\_AWUF  
     STM8AF\_STM8S, [153](#)  
 \_AWU\_AWUEN  
     STM8AF\_STM8S, [153](#)  
 \_AWU\_AWUTB  
     STM8AF\_STM8S, [153](#)  
 \_AWU\_AWUTB0

- STM8AF\_STM8S, [154](#)
- \_AWU\_AWUTB1
  - STM8AF\_STM8S, [154](#)
- \_AWU\_AWUTB2
  - STM8AF\_STM8S, [154](#)
- \_AWU\_AWUTB3
  - STM8AF\_STM8S, [154](#)
- \_AWU\_CSR
  - STM8AF\_STM8S, [154](#)
- \_AWU\_CSR\_RESET\_VALUE
  - STM8AF\_STM8S, [155](#)
- \_AWU\_MSR
  - STM8AF\_STM8S, [155](#)
- \_AWU\_TBR
  - STM8AF\_STM8S, [155](#)
- \_AWU\_TBR\_RESET\_VALUE
  - STM8AF\_STM8S, [155](#)
- \_BEEP
  - STM8AF\_STM8S, [155](#)
- \_BEEP\_BEEPDI0
  - STM8AF\_STM8S, [156](#)
- \_BEEP\_BEEPDI1
  - STM8AF\_STM8S, [156](#)
- \_BEEP\_BEEPDI2
  - STM8AF\_STM8S, [156](#)
- \_BEEP\_BEEPDI3
  - STM8AF\_STM8S, [156](#)
- \_BEEP\_BEEPDI4
  - STM8AF\_STM8S, [157](#)
- \_BEEP\_BEEPEN
  - STM8AF\_STM8S, [157](#)
- \_BEEP\_BEEPSEL
  - STM8AF\_STM8S, [157](#)
- \_BEEP\_BEEPSEL0
  - STM8AF\_STM8S, [157](#)
- \_BEEP\_BEEPSEL1
  - STM8AF\_STM8S, [157](#)
- \_BEEP\_CSR
  - STM8AF\_STM8S, [158](#)
- \_BEEP\_CSR\_RESET\_VALUE
  - STM8AF\_STM8S, [158](#)
- \_BITS
  - STM8AF\_STM8S, [158](#)
- \_CAN
  - STM8AF\_STM8S, [158](#)
- \_CAN\_ABOM
  - STM8AF\_STM8S, [158](#)
- \_CAN\_ABRQ
  - STM8AF\_STM8S, [159](#)
- \_CAN\_ALST
  - STM8AF\_STM8S, [159](#)
- \_CAN\_AWUM
  - STM8AF\_STM8S, [159](#)
- \_CAN\_BOFF
  - STM8AF\_STM8S, [159](#)
- \_CAN\_BOFIE
  - STM8AF\_STM8S, [159](#)
- \_CAN\_BRP
  - STM8AF\_STM8S, [160](#)
- \_CAN\_BRP0
  - STM8AF\_STM8S, [160](#)
- \_CAN\_BRP1
  - STM8AF\_STM8S, [160](#)
- \_CAN\_BRP2
  - STM8AF\_STM8S, [160](#)
- \_CAN\_BRP3
  - STM8AF\_STM8S, [160](#)
- \_CAN\_BRP4
  - STM8AF\_STM8S, [161](#)
- \_CAN\_BRP5
  - STM8AF\_STM8S, [161](#)
- \_CAN\_BS1
  - STM8AF\_STM8S, [161](#)
- \_CAN\_BS10
  - STM8AF\_STM8S, [161](#)
- \_CAN\_BS11
  - STM8AF\_STM8S, [161](#)
- \_CAN\_BS12
  - STM8AF\_STM8S, [162](#)
- \_CAN\_BS13
  - STM8AF\_STM8S, [162](#)
- \_CAN\_BS2
  - STM8AF\_STM8S, [162](#)
- \_CAN\_BS20
  - STM8AF\_STM8S, [162](#)
- \_CAN\_BS21
  - STM8AF\_STM8S, [162](#)
- \_CAN\_BS22
  - STM8AF\_STM8S, [163](#)
- \_CAN\_BTR1
  - STM8AF\_STM8S, [163](#)
- \_CAN\_BTR1\_RESET\_VALUE
  - STM8AF\_STM8S, [163](#)
- \_CAN\_BTR2
  - STM8AF\_STM8S, [163](#)
- \_CAN\_BTR2\_RESET\_VALUE
  - STM8AF\_STM8S, [163](#)
- \_CAN\_CODE
  - STM8AF\_STM8S, [164](#)
- \_CAN\_CODE0
  - STM8AF\_STM8S, [164](#)
- \_CAN\_CODE1
  - STM8AF\_STM8S, [164](#)
- \_CAN\_DGR
  - STM8AF\_STM8S, [164](#)
- \_CAN\_DGR\_RESET\_VALUE
  - STM8AF\_STM8S, [164](#)
- \_CAN\_DLC
  - STM8AF\_STM8S, [165](#)
- \_CAN\_DLC0
  - STM8AF\_STM8S, [165](#)
- \_CAN\_DLC1
  - STM8AF\_STM8S, [165](#)
- \_CAN\_DLC2

STM8AF\_STM8S, [165](#)  
\_CAN\_DLC3  
STM8AF\_STM8S, [165](#)  
\_CAN\_EIER  
STM8AF\_STM8S, [166](#)  
\_CAN\_EIER\_RESET\_VALUE  
STM8AF\_STM8S, [166](#)  
\_CAN\_EPVF  
STM8AF\_STM8S, [166](#)  
\_CAN\_EPVIE  
STM8AF\_STM8S, [166](#)  
\_CAN\_ERRI  
STM8AF\_STM8S, [166](#)  
\_CAN\_ERRIE  
STM8AF\_STM8S, [167](#)  
\_CAN\_ESR  
STM8AF\_STM8S, [167](#)  
\_CAN\_ESR\_RESET\_VALUE  
STM8AF\_STM8S, [167](#)  
\_CAN\_EWGF  
STM8AF\_STM8S, [167](#)  
\_CAN\_EWGIE  
STM8AF\_STM8S, [167](#)  
\_CAN\_F0R1  
STM8AF\_STM8S, [168](#)  
\_CAN\_F0R2  
STM8AF\_STM8S, [168](#)  
\_CAN\_F0R3  
STM8AF\_STM8S, [168](#)  
\_CAN\_F0R4  
STM8AF\_STM8S, [168](#)  
\_CAN\_F0R5  
STM8AF\_STM8S, [168](#)  
\_CAN\_F0R6  
STM8AF\_STM8S, [169](#)  
\_CAN\_F0R7  
STM8AF\_STM8S, [169](#)  
\_CAN\_F0R8  
STM8AF\_STM8S, [169](#)  
\_CAN\_F1R1  
STM8AF\_STM8S, [169](#)  
\_CAN\_F1R2  
STM8AF\_STM8S, [169](#)  
\_CAN\_F1R3  
STM8AF\_STM8S, [170](#)  
\_CAN\_F1R4  
STM8AF\_STM8S, [170](#)  
\_CAN\_F1R5  
STM8AF\_STM8S, [170](#)  
\_CAN\_F1R6  
STM8AF\_STM8S, [170](#)  
\_CAN\_F1R7  
STM8AF\_STM8S, [170](#)  
\_CAN\_F1R8  
STM8AF\_STM8S, [171](#)  
\_CAN\_F2R1  
STM8AF\_STM8S, [171](#)  
\_CAN\_F2R2

STM8AF\_STM8S, [171](#)  
\_CAN\_F2R3  
STM8AF\_STM8S, [171](#)  
\_CAN\_F2R4  
STM8AF\_STM8S, [171](#)  
\_CAN\_F2R5  
STM8AF\_STM8S, [172](#)  
\_CAN\_F2R6  
STM8AF\_STM8S, [172](#)  
\_CAN\_F2R7  
STM8AF\_STM8S, [172](#)  
\_CAN\_F2R8  
STM8AF\_STM8S, [172](#)  
\_CAN\_F3R1  
STM8AF\_STM8S, [172](#)  
\_CAN\_F3R2  
STM8AF\_STM8S, [173](#)  
\_CAN\_F3R3  
STM8AF\_STM8S, [173](#)  
\_CAN\_F3R4  
STM8AF\_STM8S, [173](#)  
\_CAN\_F3R5  
STM8AF\_STM8S, [173](#)  
\_CAN\_F3R6  
STM8AF\_STM8S, [173](#)  
\_CAN\_F3R7  
STM8AF\_STM8S, [174](#)  
\_CAN\_F3R8  
STM8AF\_STM8S, [174](#)  
\_CAN\_F4R1  
STM8AF\_STM8S, [174](#)  
\_CAN\_F4R2  
STM8AF\_STM8S, [174](#)  
\_CAN\_F4R3  
STM8AF\_STM8S, [174](#)  
\_CAN\_F4R4  
STM8AF\_STM8S, [175](#)  
\_CAN\_F4R5  
STM8AF\_STM8S, [175](#)  
\_CAN\_F4R6  
STM8AF\_STM8S, [175](#)  
\_CAN\_F4R7  
STM8AF\_STM8S, [175](#)  
\_CAN\_F4R8  
STM8AF\_STM8S, [175](#)  
\_CAN\_F5R1  
STM8AF\_STM8S, [176](#)  
\_CAN\_F5R2  
STM8AF\_STM8S, [176](#)  
\_CAN\_F5R3  
STM8AF\_STM8S, [176](#)  
\_CAN\_F5R4  
STM8AF\_STM8S, [176](#)  
\_CAN\_F5R5  
STM8AF\_STM8S, [176](#)  
\_CAN\_F5R6  
STM8AF\_STM8S, [177](#)  
\_CAN\_F5R7



- STM8AF\_STM8S, [177](#)
- \_CAN\_F5R8
  - STM8AF\_STM8S, [177](#)
- \_CAN\_FACT0
  - STM8AF\_STM8S, [177](#)
- \_CAN\_FACT1
  - STM8AF\_STM8S, [177](#)
- \_CAN\_FACT2
  - STM8AF\_STM8S, [178](#)
- \_CAN\_FACT3
  - STM8AF\_STM8S, [178](#)
- \_CAN\_FACT4
  - STM8AF\_STM8S, [178](#)
- \_CAN\_FACT5
  - STM8AF\_STM8S, [178](#)
- \_CAN\_FCR1
  - STM8AF\_STM8S, [178](#)
- \_CAN\_FCR2
  - STM8AF\_STM8S, [179](#)
- \_CAN\_FCR3
  - STM8AF\_STM8S, [179](#)
- \_CAN\_FCR\_RESET\_VALUE
  - STM8AF\_STM8S, [179](#)
- \_CAN\_FFIE
  - STM8AF\_STM8S, [179](#)
- \_CAN\_FMP
  - STM8AF\_STM8S, [182](#)
- \_CAN\_FMH0
  - STM8AF\_STM8S, [179](#)
- \_CAN\_FMH1
  - STM8AF\_STM8S, [180](#)
- \_CAN\_FMH2
  - STM8AF\_STM8S, [180](#)
- \_CAN\_FMH3
  - STM8AF\_STM8S, [180](#)
- \_CAN\_FMH4
  - STM8AF\_STM8S, [180](#)
- \_CAN\_FMH5
  - STM8AF\_STM8S, [180](#)
- \_CAN\_FML0
  - STM8AF\_STM8S, [181](#)
- \_CAN\_FML1
  - STM8AF\_STM8S, [181](#)
- \_CAN\_FML2
  - STM8AF\_STM8S, [181](#)
- \_CAN\_FML3
  - STM8AF\_STM8S, [181](#)
- \_CAN\_FML4
  - STM8AF\_STM8S, [181](#)
- \_CAN\_FML5
  - STM8AF\_STM8S, [182](#)
- \_CAN\_FMP0
  - STM8AF\_STM8S, [182](#)
- \_CAN\_FMP1
  - STM8AF\_STM8S, [182](#)
- \_CAN\_FMPIE
  - STM8AF\_STM8S, [182](#)
- \_CAN\_FMR1
  - STM8AF\_STM8S, [183](#)
- \_CAN\_FMR1\_RESET\_VALUE
  - STM8AF\_STM8S, [183](#)
- \_CAN\_FMR2
  - STM8AF\_STM8S, [183](#)
- \_CAN\_FMR2\_RESET\_VALUE
  - STM8AF\_STM8S, [183](#)
- \_CAN\_FOVR
  - STM8AF\_STM8S, [184](#)
- \_CAN\_FOVIE
  - STM8AF\_STM8S, [183](#)
- \_CAN\_FSC0
  - STM8AF\_STM8S, [184](#)
- \_CAN\_FSC00
  - STM8AF\_STM8S, [184](#)
- \_CAN\_FSC01
  - STM8AF\_STM8S, [184](#)
- \_CAN\_FSC1
  - STM8AF\_STM8S, [184](#)
- \_CAN\_FSC10
  - STM8AF\_STM8S, [185](#)
- \_CAN\_FSC11
  - STM8AF\_STM8S, [185](#)
- \_CAN\_FSC2
  - STM8AF\_STM8S, [185](#)
- \_CAN\_FSC20
  - STM8AF\_STM8S, [185](#)
- \_CAN\_FSC21
  - STM8AF\_STM8S, [185](#)
- \_CAN\_FSC3
  - STM8AF\_STM8S, [186](#)
- \_CAN\_FSC30
  - STM8AF\_STM8S, [186](#)
- \_CAN\_FSC31
  - STM8AF\_STM8S, [186](#)
- \_CAN\_FSC4
  - STM8AF\_STM8S, [186](#)
- \_CAN\_FSC40
  - STM8AF\_STM8S, [186](#)
- \_CAN\_FSC41
  - STM8AF\_STM8S, [187](#)
- \_CAN\_FSC5
  - STM8AF\_STM8S, [187](#)
- \_CAN\_FSC50
  - STM8AF\_STM8S, [187](#)
- \_CAN\_FSC51
  - STM8AF\_STM8S, [187](#)
- \_CAN\_FULL
  - STM8AF\_STM8S, [187](#)
- \_CAN\_IDE
  - STM8AF\_STM8S, [188](#)
- \_CAN\_IER
  - STM8AF\_STM8S, [188](#)
- \_CAN\_IER\_RESET\_VALUE
  - STM8AF\_STM8S, [188](#)
- \_CAN\_INAK
  - STM8AF\_STM8S, [188](#)
- \_CAN\_INRQ

STM8AF\_STM8S, [188](#)  
\_CAN\_LBKM  
STM8AF\_STM8S, [189](#)  
\_CAN\_LEC  
STM8AF\_STM8S, [189](#)  
\_CAN\_LEC0  
STM8AF\_STM8S, [189](#)  
\_CAN\_LEC1  
STM8AF\_STM8S, [189](#)  
\_CAN\_LEC2  
STM8AF\_STM8S, [189](#)  
\_CAN\_LECIE  
STM8AF\_STM8S, [190](#)  
\_CAN\_LOW0  
STM8AF\_STM8S, [190](#)  
\_CAN\_LOW1  
STM8AF\_STM8S, [190](#)  
\_CAN\_LOW2  
STM8AF\_STM8S, [190](#)  
\_CAN\_MCR  
STM8AF\_STM8S, [190](#)  
\_CAN\_MCR\_RESET\_VALUE  
STM8AF\_STM8S, [191](#)  
\_CAN\_MCSR  
STM8AF\_STM8S, [191](#)  
\_CAN\_MCSR\_RESET\_VALUE  
STM8AF\_STM8S, [191](#)  
\_CAN\_MDAR1  
STM8AF\_STM8S, [191](#)  
\_CAN\_MDAR2  
STM8AF\_STM8S, [191](#)  
\_CAN\_MDAR3  
STM8AF\_STM8S, [192](#)  
\_CAN\_MDAR4  
STM8AF\_STM8S, [192](#)  
\_CAN\_MDAR5  
STM8AF\_STM8S, [192](#)  
\_CAN\_MDAR6  
STM8AF\_STM8S, [192](#)  
\_CAN\_MDAR7  
STM8AF\_STM8S, [192](#)  
\_CAN\_MDAR8  
STM8AF\_STM8S, [193](#)  
\_CAN\_MDLCR  
STM8AF\_STM8S, [193](#)  
\_CAN\_MDLCR\_RESET\_VALUE  
STM8AF\_STM8S, [193](#)  
\_CAN\_MFMIR  
STM8AF\_STM8S, [193](#)  
\_CAN\_MFMIR\_RESET\_VALUE  
STM8AF\_STM8S, [193](#)  
\_CAN\_MIDR1  
STM8AF\_STM8S, [194](#)  
\_CAN\_MIDR2  
STM8AF\_STM8S, [194](#)  
\_CAN\_MIDR3  
STM8AF\_STM8S, [194](#)  
\_CAN\_MIDR4  
STM8AF\_STM8S, [194](#)  
\_CAN\_MSR  
STM8AF\_STM8S, [194](#)  
\_CAN\_MSR\_RESET\_VALUE  
STM8AF\_STM8S, [195](#)  
\_CAN\_MTSRH  
STM8AF\_STM8S, [195](#)  
\_CAN\_MTSRL  
STM8AF\_STM8S, [195](#)  
\_CAN\_NART  
STM8AF\_STM8S, [195](#)  
\_CAN\_PS  
STM8AF\_STM8S, [195](#)  
\_CAN\_PS0  
STM8AF\_STM8S, [196](#)  
\_CAN\_PS1  
STM8AF\_STM8S, [196](#)  
\_CAN\_PS2  
STM8AF\_STM8S, [196](#)  
\_CAN\_PSR  
STM8AF\_STM8S, [196](#)  
\_CAN\_PSR\_RESET\_VALUE  
STM8AF\_STM8S, [196](#)  
\_CAN\_RX  
STM8AF\_STM8S, [199](#)  
\_CAN\_RECR  
STM8AF\_STM8S, [197](#)  
\_CAN\_RECR\_RESET\_VALUE  
STM8AF\_STM8S, [197](#)  
\_CAN\_RFR  
STM8AF\_STM8S, [197](#)  
\_CAN\_RFLM  
STM8AF\_STM8S, [197](#)  
\_CAN\_RFOM  
STM8AF\_STM8S, [197](#)  
\_CAN\_RFR\_RESET\_VALUE  
STM8AF\_STM8S, [198](#)  
\_CAN\_RQCP  
STM8AF\_STM8S, [198](#)  
\_CAN\_RQCP0  
STM8AF\_STM8S, [198](#)  
\_CAN\_RQCP1  
STM8AF\_STM8S, [198](#)  
\_CAN\_RQCP2  
STM8AF\_STM8S, [198](#)  
\_CAN\_RTR  
STM8AF\_STM8S, [199](#)  
\_CAN\_RXS  
STM8AF\_STM8S, [199](#)  
\_CAN\_SAMP  
STM8AF\_STM8S, [199](#)  
\_CAN\_SILM  
STM8AF\_STM8S, [199](#)  
\_CAN\_SJW  
STM8AF\_STM8S, [200](#)  
\_CAN\_SJW0  
STM8AF\_STM8S, [200](#)  
\_CAN\_SJW1

STM8AF\_STM8S, [200](#)  
\_CAN\_SLAK  
STM8AF\_STM8S, [200](#)  
\_CAN\_SLEEP  
STM8AF\_STM8S, [200](#)  
\_CAN\_TX  
STM8AF\_STM8S, [203](#)  
\_CAN\_TECR  
STM8AF\_STM8S, [201](#)  
\_CAN\_TECR\_RESET\_VALUE  
STM8AF\_STM8S, [201](#)  
\_CAN\_TERR  
STM8AF\_STM8S, [201](#)  
\_CAN\_TGT  
STM8AF\_STM8S, [201](#)  
\_CAN\_TME0  
STM8AF\_STM8S, [201](#)  
\_CAN\_TME1  
STM8AF\_STM8S, [202](#)  
\_CAN\_TME2  
STM8AF\_STM8S, [202](#)  
\_CAN\_TMEIE  
STM8AF\_STM8S, [202](#)  
\_CAN\_TPR  
STM8AF\_STM8S, [202](#)  
\_CAN\_TPR\_RESET\_VALUE  
STM8AF\_STM8S, [202](#)  
\_CAN\_TSR  
STM8AF\_STM8S, [203](#)  
\_CAN\_TSR\_RESET\_VALUE  
STM8AF\_STM8S, [203](#)  
\_CAN\_TTCM  
STM8AF\_STM8S, [203](#)  
\_CAN\_TXFP  
STM8AF\_STM8S, [203](#)  
\_CAN\_TXM2E  
STM8AF\_STM8S, [204](#)  
\_CAN\_TXOK  
STM8AF\_STM8S, [204](#)  
\_CAN\_TXOK0  
STM8AF\_STM8S, [204](#)  
\_CAN\_TXOK1  
STM8AF\_STM8S, [204](#)  
\_CAN\_TXOK2  
STM8AF\_STM8S, [204](#)  
\_CAN\_TXRQ  
STM8AF\_STM8S, [205](#)  
\_CAN\_WKUI  
STM8AF\_STM8S, [205](#)  
\_CAN\_WKUIE  
STM8AF\_STM8S, [205](#)  
\_CFG  
STM8AF\_STM8S, [205](#)  
\_CFG\_AL  
STM8AF\_STM8S, [205](#)  
\_CFG\_GCR  
STM8AF\_STM8S, [206](#)  
\_CFG\_GCR\_RESET\_VALUE  
STM8AF\_STM8S, [206](#)  
\_CFG\_SWD  
STM8AF\_STM8S, [206](#)  
\_CLK  
STM8AF\_STM8S, [206](#)  
\_CLK\_ADC  
STM8AF\_STM8S, [206](#)  
\_CLK\_AUX  
STM8AF\_STM8S, [207](#)  
\_CLK\_AWU  
STM8AF\_STM8S, [207](#)  
\_CLK\_CAN  
STM8AF\_STM8S, [207](#)  
\_CLK\_CCOR  
STM8AF\_STM8S, [208](#)  
\_CLK\_CCOBSY  
STM8AF\_STM8S, [207](#)  
\_CLK\_CCOEN  
STM8AF\_STM8S, [207](#)  
\_CLK\_CCORDY  
STM8AF\_STM8S, [208](#)  
\_CLK\_CCOR\_RESET\_VALUE  
STM8AF\_STM8S, [208](#)  
\_CLK\_CCOSEL  
STM8AF\_STM8S, [208](#)  
\_CLK\_CCOSEL0  
STM8AF\_STM8S, [208](#)  
\_CLK\_CCOSEL1  
STM8AF\_STM8S, [209](#)  
\_CLK\_CCOSEL2  
STM8AF\_STM8S, [209](#)  
\_CLK\_CCOSEL3  
STM8AF\_STM8S, [209](#)  
\_CLK\_CKDIVR  
STM8AF\_STM8S, [209](#)  
\_CLK\_CKDIVR\_RESET\_VALUE  
STM8AF\_STM8S, [209](#)  
\_CLK\_CMSR  
STM8AF\_STM8S, [210](#)  
\_CLK\_CMSR\_RESET\_VALUE  
STM8AF\_STM8S, [210](#)  
\_CLK\_CPUDIV  
STM8AF\_STM8S, [210](#)  
\_CLK\_CPUDIV0  
STM8AF\_STM8S, [210](#)  
\_CLK\_CPUDIV1  
STM8AF\_STM8S, [210](#)  
\_CLK\_CPUDIV2  
STM8AF\_STM8S, [211](#)  
\_CLK\_CSSD  
STM8AF\_STM8S, [211](#)  
\_CLK\_CSSR  
STM8AF\_STM8S, [211](#)  
\_CLK\_CSSDIE  
STM8AF\_STM8S, [211](#)  
\_CLK\_CSEN  
STM8AF\_STM8S, [211](#)  
\_CLK\_CSSR\_RESET\_VALUE

STM8AF\_STM8S, [212](#)  
 \_CLK\_ECKR  
   STM8AF\_STM8S, [212](#)  
 \_CLK\_ECKR\_HSERDY  
   STM8AF\_STM8S, [212](#)  
 \_CLK\_ECKR\_RESET\_VALUE  
   STM8AF\_STM8S, [212](#)  
 \_CLK\_FHWU  
   STM8AF\_STM8S, [212](#)  
 \_CLK\_HSEEN  
   STM8AF\_STM8S, [213](#)  
 \_CLK\_HSIDIV  
   STM8AF\_STM8S, [213](#)  
 \_CLK\_HSIDIV0  
   STM8AF\_STM8S, [213](#)  
 \_CLK\_HSIDIV1  
   STM8AF\_STM8S, [213](#)  
 \_CLK\_HSIEN  
   STM8AF\_STM8S, [213](#)  
 \_CLK\_HSIRDY  
   STM8AF\_STM8S, [214](#)  
 \_CLK\_HSITRIM  
   STM8AF\_STM8S, [214](#)  
 \_CLK\_HSITRIM0  
   STM8AF\_STM8S, [214](#)  
 \_CLK\_HSITRIM1  
   STM8AF\_STM8S, [214](#)  
 \_CLK\_HSITRIM2  
   STM8AF\_STM8S, [214](#)  
 \_CLK\_HSITRIM3  
   STM8AF\_STM8S, [215](#)  
 \_CLK\_HSITRIMR  
   STM8AF\_STM8S, [215](#)  
 \_CLK\_HSITRIMR\_RESET\_VALUE  
   STM8AF\_STM8S, [215](#)  
 \_CLK\_I2C  
   STM8AF\_STM8S, [215](#)  
 \_CLK\_ICKR  
   STM8AF\_STM8S, [215](#)  
 \_CLK\_ICKR\_RESET\_VALUE  
   STM8AF\_STM8S, [216](#)  
 \_CLK\_LSIEN  
   STM8AF\_STM8S, [216](#)  
 \_CLK\_LSIRDY  
   STM8AF\_STM8S, [216](#)  
 \_CLK\_PCKENR1  
   STM8AF\_STM8S, [216](#)  
 \_CLK\_PCKENR1\_RESET\_VALUE  
   STM8AF\_STM8S, [216](#)  
 \_CLK\_PCKENR2  
   STM8AF\_STM8S, [217](#)  
 \_CLK\_PCKENR2\_RESET\_VALUE  
   STM8AF\_STM8S, [217](#)  
 \_CLK\_REGAH  
   STM8AF\_STM8S, [217](#)  
 \_CLK\_SPI  
   STM8AF\_STM8S, [217](#)  
 \_CLK\_SWR  
   STM8AF\_STM8S, [220](#)  
 \_CLK\_SWBSY  
   STM8AF\_STM8S, [217](#)  
 \_CLK\_SWCR  
   STM8AF\_STM8S, [218](#)  
 \_CLK\_SWCR\_RESET\_VALUE  
   STM8AF\_STM8S, [218](#)  
 \_CLK\_SWEN  
   STM8AF\_STM8S, [218](#)  
 \_CLK\_SWIF  
   STM8AF\_STM8S, [219](#)  
 \_CLK\_SWIEN  
   STM8AF\_STM8S, [219](#)  
 \_CLK\_SWIMCCR  
   STM8AF\_STM8S, [219](#)  
 \_CLK\_SWIMCCR\_RESET\_VALUE  
   STM8AF\_STM8S, [219](#)  
 \_CLK\_SWIMCLK  
   STM8AF\_STM8S, [220](#)  
 \_CLK\_SWI\_HSE  
   STM8AF\_STM8S, [218](#)  
 \_CLK\_SWI\_HSI  
   STM8AF\_STM8S, [218](#)  
 \_CLK\_SWI\_LSI  
   STM8AF\_STM8S, [219](#)  
 \_CLK\_SWR\_RESET\_VALUE  
   STM8AF\_STM8S, [220](#)  
 \_CLK\_TIM1  
   STM8AF\_STM8S, [220](#)  
 \_CLK\_TIM2\_TIM5  
   STM8AF\_STM8S, [220](#)  
 \_CLK\_TIM3  
   STM8AF\_STM8S, [221](#)  
 \_CLK\_TIM4\_TIM6  
   STM8AF\_STM8S, [221](#)  
 \_CLK\_UART1  
   STM8AF\_STM8S, [221](#)  
 \_CLK\_UART2  
   STM8AF\_STM8S, [221](#)  
 \_EXTI  
   STM8AF\_STM8S, [221](#)  
 \_EXTI\_CR1  
   STM8AF\_STM8S, [222](#)  
 \_EXTI\_CR1\_RESET\_VALUE  
   STM8AF\_STM8S, [222](#)  
 \_EXTI\_CR2  
   STM8AF\_STM8S, [222](#)  
 \_EXTI\_CR2\_RESET\_VALUE  
   STM8AF\_STM8S, [222](#)  
 \_EXTI\_PAIS  
   STM8AF\_STM8S, [222](#)  
 \_EXTI\_PAIS0  
   STM8AF\_STM8S, [223](#)  
 \_EXTI\_PAIS1  
   STM8AF\_STM8S, [223](#)  
 \_EXTI\_PBS  
   STM8AF\_STM8S, [223](#)  
 \_EXTI\_PBS0

- STM8AF\_STM8S, [223](#)
- \_EXTI\_PBIS1
  - STM8AF\_STM8S, [223](#)
- \_EXTI\_PCIS
  - STM8AF\_STM8S, [224](#)
- \_EXTI\_PCIS0
  - STM8AF\_STM8S, [224](#)
- \_EXTI\_PCIS1
  - STM8AF\_STM8S, [224](#)
- \_EXTI\_PDIS
  - STM8AF\_STM8S, [224](#)
- \_EXTI\_PDIS0
  - STM8AF\_STM8S, [224](#)
- \_EXTI\_PDIS1
  - STM8AF\_STM8S, [225](#)
- \_EXTI\_PEIS
  - STM8AF\_STM8S, [225](#)
- \_EXTI\_PEIS0
  - STM8AF\_STM8S, [225](#)
- \_EXTI\_PEIS1
  - STM8AF\_STM8S, [225](#)
- \_EXTI\_TLIS
  - STM8AF\_STM8S, [225](#)
- \_FLASH
  - STM8AF\_STM8S, [226](#)
- \_FLASH\_AHALT
  - STM8AF\_STM8S, [226](#)
- \_FLASH\_CR1
  - STM8AF\_STM8S, [226](#)
- \_FLASH\_CR1\_RESET\_VALUE
  - STM8AF\_STM8S, [226](#)
- \_FLASH\_CR2
  - STM8AF\_STM8S, [226](#)
- \_FLASH\_CR2\_RESET\_VALUE
  - STM8AF\_STM8S, [227](#)
- \_FLASH\_DUL
  - STM8AF\_STM8S, [227](#)
- \_FLASH\_DUKR
  - STM8AF\_STM8S, [227](#)
- \_FLASH\_DUKR\_RESET\_VALUE
  - STM8AF\_STM8S, [227](#)
- \_FLASH\_EOP
  - STM8AF\_STM8S, [227](#)
- \_FLASH\_ERASE
  - STM8AF\_STM8S, [228](#)
- \_FLASH\_FIX
  - STM8AF\_STM8S, [228](#)
- \_FLASH\_FPR
  - STM8AF\_STM8S, [228](#)
- \_FLASH\_FPRG
  - STM8AF\_STM8S, [228](#)
- \_FLASH\_HALT
  - STM8AF\_STM8S, [228](#)
- \_FLASH\_HVOFF
  - STM8AF\_STM8S, [229](#)
- \_FLASH\_IE
  - STM8AF\_STM8S, [229](#)
- \_FLASH\_IAPSR
  - STM8AF\_STM8S, [229](#)
- \_FLASH\_IAPSR\_RESET\_VALUE
  - STM8AF\_STM8S, [229](#)
- \_FLASH\_NCR2
  - STM8AF\_STM8S, [229](#)
- \_FLASH\_NCR2\_RESET\_VALUE
  - STM8AF\_STM8S, [230](#)
- \_FLASH\_NFPR
  - STM8AF\_STM8S, [230](#)
- \_FLASH\_OPT
  - STM8AF\_STM8S, [230](#)
- \_FLASH\_PRG
  - STM8AF\_STM8S, [230](#)
- \_FLASH\_PUL
  - STM8AF\_STM8S, [231](#)
- \_FLASH\_PUKR
  - STM8AF\_STM8S, [230](#)
- \_FLASH\_PUKR\_RESET\_VALUE
  - STM8AF\_STM8S, [231](#)
- \_FLASH\_WPB
  - STM8AF\_STM8S, [231](#)
- \_FLASH\_WPB0
  - STM8AF\_STM8S, [231](#)
- \_FLASH\_WPB1
  - STM8AF\_STM8S, [231](#)
- \_FLASH\_WPB2
  - STM8AF\_STM8S, [232](#)
- \_FLASH\_WPB3
  - STM8AF\_STM8S, [232](#)
- \_FLASH\_WPB4
  - STM8AF\_STM8S, [232](#)
- \_FLASH\_WPB5
  - STM8AF\_STM8S, [232](#)
- \_FLASH\_WPRG
  - STM8AF\_STM8S, [232](#)
- \_FLASH\_WR\_PG\_DIS
  - STM8AF\_STM8S, [233](#)
- \_GPIOA
  - STM8AF\_STM8S, [235](#)
- \_GPIOB
  - STM8AF\_STM8S, [236](#)
- \_GPIOC
  - STM8AF\_STM8S, [238](#)
- \_GPIOD
  - STM8AF\_STM8S, [239](#)
- \_GPIOE
  - STM8AF\_STM8S, [240](#)
- \_GPIOF
  - STM8AF\_STM8S, [241](#)
- \_GPIOG
  - STM8AF\_STM8S, [242](#)
- \_GPIOH
  - STM8AF\_STM8S, [244](#)
- \_GPIOI
  - STM8AF\_STM8S, [245](#)
- \_GPIOA\_CR1
  - STM8AF\_STM8S, [235](#)
- \_GPIOA\_CR2

STM8AF\_STM8S, [236](#)  
\_GPIOA\_DDR  
STM8AF\_STM8S, [236](#)  
\_GPIOA\_IDR  
STM8AF\_STM8S, [236](#)  
\_GPIOA\_ODR  
STM8AF\_STM8S, [236](#)  
\_GPIOB\_CR1  
STM8AF\_STM8S, [237](#)  
\_GPIOB\_CR2  
STM8AF\_STM8S, [237](#)  
\_GPIOB\_DDR  
STM8AF\_STM8S, [237](#)  
\_GPIOB\_IDR  
STM8AF\_STM8S, [237](#)  
\_GPIOB\_ODR  
STM8AF\_STM8S, [237](#)  
\_GPIOC\_CR1  
STM8AF\_STM8S, [238](#)  
\_GPIOC\_CR2  
STM8AF\_STM8S, [238](#)  
\_GPIOC\_DDR  
STM8AF\_STM8S, [238](#)  
\_GPIOC\_IDR  
STM8AF\_STM8S, [238](#)  
\_GPIOC\_ODR  
STM8AF\_STM8S, [239](#)  
\_GPIOD\_CR1  
STM8AF\_STM8S, [239](#)  
\_GPIOD\_CR2  
STM8AF\_STM8S, [239](#)  
\_GPIOD\_DDR  
STM8AF\_STM8S, [239](#)  
\_GPIOD\_IDR  
STM8AF\_STM8S, [240](#)  
\_GPIOD\_ODR  
STM8AF\_STM8S, [240](#)  
\_GPIOE\_CR1  
STM8AF\_STM8S, [240](#)  
\_GPIOE\_CR2  
STM8AF\_STM8S, [240](#)  
\_GPIOE\_DDR  
STM8AF\_STM8S, [241](#)  
\_GPIOE\_IDR  
STM8AF\_STM8S, [241](#)  
\_GPIOE\_ODR  
STM8AF\_STM8S, [241](#)  
\_GPIOF\_CR1  
STM8AF\_STM8S, [241](#)  
\_GPIOF\_CR2  
STM8AF\_STM8S, [242](#)  
\_GPIOF\_DDR  
STM8AF\_STM8S, [242](#)  
\_GPIOF\_IDR  
STM8AF\_STM8S, [242](#)  
\_GPIOF\_ODR  
STM8AF\_STM8S, [242](#)  
\_GPIOG\_CR1  
STM8AF\_STM8S, [243](#)  
\_GPIOG\_CR2  
STM8AF\_STM8S, [243](#)  
\_GPIOG\_DDR  
STM8AF\_STM8S, [243](#)  
\_GPIOG\_IDR  
STM8AF\_STM8S, [243](#)  
\_GPIOG\_ODR  
STM8AF\_STM8S, [243](#)  
\_GPIOH\_CR1  
STM8AF\_STM8S, [244](#)  
\_GPIOH\_CR2  
STM8AF\_STM8S, [244](#)  
\_GPIOH\_DDR  
STM8AF\_STM8S, [244](#)  
\_GPIOH\_IDR  
STM8AF\_STM8S, [244](#)  
\_GPIOH\_ODR  
STM8AF\_STM8S, [245](#)  
\_GPIOI\_CR1  
STM8AF\_STM8S, [245](#)  
\_GPIOI\_CR2  
STM8AF\_STM8S, [245](#)  
\_GPIOI\_DDR  
STM8AF\_STM8S, [245](#)  
\_GPIOI\_IDR  
STM8AF\_STM8S, [246](#)  
\_GPIOI\_ODR  
STM8AF\_STM8S, [246](#)  
\_GPIO\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [233](#)  
\_GPIO\_CR2\_RESET\_VALUE  
STM8AF\_STM8S, [233](#)  
\_GPIO\_DDR\_RESET\_VALUE  
STM8AF\_STM8S, [233](#)  
\_GPIO\_ODR\_RESET\_VALUE  
STM8AF\_STM8S, [233](#)  
\_GPIO\_PIN0  
STM8AF\_STM8S, [234](#)  
\_GPIO\_PIN1  
STM8AF\_STM8S, [234](#)  
\_GPIO\_PIN2  
STM8AF\_STM8S, [234](#)  
\_GPIO\_PIN3  
STM8AF\_STM8S, [234](#)  
\_GPIO\_PIN4  
STM8AF\_STM8S, [234](#)  
\_GPIO\_PIN5  
STM8AF\_STM8S, [235](#)  
\_GPIO\_PIN6  
STM8AF\_STM8S, [235](#)  
\_GPIO\_PIN7  
STM8AF\_STM8S, [235](#)  
\_I2C  
STM8AF\_STM8S, [246](#)  
\_I2C\_AF  
STM8AF\_STM8S, [249](#)  
\_I2C\_ACK

- STM8AF\_STM8S, [246](#)
- \_I2C\_ADD0
  - STM8AF\_STM8S, [246](#)
- \_I2C\_ADD1
  - STM8AF\_STM8S, [247](#)
- \_I2C\_ADD10
  - STM8AF\_STM8S, [247](#)
- \_I2C\_ADD2
  - STM8AF\_STM8S, [247](#)
- \_I2C\_ADD3
  - STM8AF\_STM8S, [247](#)
- \_I2C\_ADD4
  - STM8AF\_STM8S, [247](#)
- \_I2C\_ADD5
  - STM8AF\_STM8S, [248](#)
- \_I2C\_ADD6
  - STM8AF\_STM8S, [248](#)
- \_I2C\_ADD7
  - STM8AF\_STM8S, [248](#)
- \_I2C\_ADD8
  - STM8AF\_STM8S, [248](#)
- \_I2C\_ADD9
  - STM8AF\_STM8S, [248](#)
- \_I2C\_ADDR
  - STM8AF\_STM8S, [249](#)
- \_I2C\_ADDCONF
  - STM8AF\_STM8S, [249](#)
- \_I2C\_ADDMODE
  - STM8AF\_STM8S, [249](#)
- \_I2C\_ADD\_8\_9
  - STM8AF\_STM8S, [249](#)
- \_I2C\_ARLO
  - STM8AF\_STM8S, [250](#)
- \_I2C\_BERR
  - STM8AF\_STM8S, [250](#)
- \_I2C\_BTF
  - STM8AF\_STM8S, [250](#)
- \_I2C\_BUSY
  - STM8AF\_STM8S, [250](#)
- \_I2C\_CCR
  - STM8AF\_STM8S, [250](#)
- \_I2C\_CCR0
  - STM8AF\_STM8S, [251](#)
- \_I2C\_CCR1
  - STM8AF\_STM8S, [251](#)
- \_I2C\_CCR2
  - STM8AF\_STM8S, [251](#)
- \_I2C\_CCR3
  - STM8AF\_STM8S, [251](#)
- \_I2C\_CCRH
  - STM8AF\_STM8S, [251](#)
- \_I2C\_CCRL
  - STM8AF\_STM8S, [252](#)
- \_I2C\_CCRH\_RESET\_VALUE
  - STM8AF\_STM8S, [252](#)
- \_I2C\_CCRL\_RESET\_VALUE
  - STM8AF\_STM8S, [252](#)
- \_I2C\_CR1
  - STM8AF\_STM8S, [252](#)
- \_I2C\_CR1\_RESET\_VALUE
  - STM8AF\_STM8S, [252](#)
- \_I2C\_CR2
  - STM8AF\_STM8S, [253](#)
- \_I2C\_CR2\_RESET\_VALUE
  - STM8AF\_STM8S, [253](#)
- \_I2C\_DR
  - STM8AF\_STM8S, [253](#)
- \_I2C\_DR\_RESET\_VALUE
  - STM8AF\_STM8S, [253](#)
- \_I2C\_DUTY
  - STM8AF\_STM8S, [253](#)
- \_I2C\_ENGC
  - STM8AF\_STM8S, [254](#)
- \_I2C\_FS
  - STM8AF\_STM8S, [256](#)
- \_I2C\_FREQ
  - STM8AF\_STM8S, [254](#)
- \_I2C\_FREQ0
  - STM8AF\_STM8S, [254](#)
- \_I2C\_FREQ1
  - STM8AF\_STM8S, [254](#)
- \_I2C\_FREQ2
  - STM8AF\_STM8S, [254](#)
- \_I2C\_FREQ3
  - STM8AF\_STM8S, [255](#)
- \_I2C\_FREQ4
  - STM8AF\_STM8S, [255](#)
- \_I2C\_FREQ5
  - STM8AF\_STM8S, [255](#)
- \_I2C\_FREQR
  - STM8AF\_STM8S, [255](#)
- \_I2C\_FREQR\_RESET\_VALUE
  - STM8AF\_STM8S, [255](#)
- \_I2C\_GENCALL
  - STM8AF\_STM8S, [256](#)
- \_I2C\_ITR
  - STM8AF\_STM8S, [257](#)
- \_I2C\_ITBUFEN
  - STM8AF\_STM8S, [256](#)
- \_I2C\_ITERREN
  - STM8AF\_STM8S, [256](#)
- \_I2C\_ITEVTEN
  - STM8AF\_STM8S, [256](#)
- \_I2C\_ITR\_RESET\_VALUE
  - STM8AF\_STM8S, [257](#)
- \_I2C\_MSL
  - STM8AF\_STM8S, [257](#)
- \_I2C\_NOSTRETCH
  - STM8AF\_STM8S, [257](#)
- \_I2C\_OARH
  - STM8AF\_STM8S, [257](#)
- \_I2C\_OARL
  - STM8AF\_STM8S, [258](#)
- \_I2C\_OARH\_RESET\_VALUE
  - STM8AF\_STM8S, [258](#)
- \_I2C\_OARL\_RESET\_VALUE

STM8AF\_STM8S, [258](#)  
\_I2C\_OVR  
STM8AF\_STM8S, [258](#)  
\_I2C\_PE  
STM8AF\_STM8S, [258](#)  
\_I2C\_POS  
STM8AF\_STM8S, [259](#)  
\_I2C\_RXNE  
STM8AF\_STM8S, [259](#)  
\_I2C\_SB  
STM8AF\_STM8S, [259](#)  
\_I2C\_SR1  
STM8AF\_STM8S, [259](#)  
\_I2C\_SR1\_RESET\_VALUE  
STM8AF\_STM8S, [259](#)  
\_I2C\_SR2  
STM8AF\_STM8S, [260](#)  
\_I2C\_SR2\_RESET\_VALUE  
STM8AF\_STM8S, [260](#)  
\_I2C\_SR3  
STM8AF\_STM8S, [260](#)  
\_I2C\_SR3\_RESET\_VALUE  
STM8AF\_STM8S, [260](#)  
\_I2C\_START  
STM8AF\_STM8S, [260](#)  
\_I2C\_STOP  
STM8AF\_STM8S, [261](#)  
\_I2C\_STOPF  
STM8AF\_STM8S, [261](#)  
\_I2C\_SWRST  
STM8AF\_STM8S, [261](#)  
\_I2C\_TRA  
STM8AF\_STM8S, [261](#)  
\_I2C\_TRISE  
STM8AF\_STM8S, [261](#)  
\_I2C\_TRISE0  
STM8AF\_STM8S, [262](#)  
\_I2C\_TRISE1  
STM8AF\_STM8S, [262](#)  
\_I2C\_TRISE2  
STM8AF\_STM8S, [262](#)  
\_I2C\_TRISE3  
STM8AF\_STM8S, [262](#)  
\_I2C\_TRISE4  
STM8AF\_STM8S, [262](#)  
\_I2C\_TRISE5  
STM8AF\_STM8S, [263](#)  
\_I2C\_TRISER  
STM8AF\_STM8S, [263](#)  
\_I2C\_TRISER\_RESET\_VALUE  
STM8AF\_STM8S, [263](#)  
\_I2C\_TXE  
STM8AF\_STM8S, [263](#)  
\_I2C\_WUFH  
STM8AF\_STM8S, [263](#)  
\_ITC  
STM8AF\_STM8S, [264](#)  
\_ITC\_SPR1  
STM8AF\_STM8S, [264](#)  
\_ITC\_SPR1\_RESET\_VALUE  
STM8AF\_STM8S, [264](#)  
\_ITC\_SPR2  
STM8AF\_STM8S, [264](#)  
\_ITC\_SPR2\_RESET\_VALUE  
STM8AF\_STM8S, [264](#)  
\_ITC\_SPR3  
STM8AF\_STM8S, [265](#)  
\_ITC\_SPR3\_RESET\_VALUE  
STM8AF\_STM8S, [265](#)  
\_ITC\_SPR4  
STM8AF\_STM8S, [265](#)  
\_ITC\_SPR4\_RESET\_VALUE  
STM8AF\_STM8S, [265](#)  
\_ITC\_SPR5  
STM8AF\_STM8S, [265](#)  
\_ITC\_SPR5\_RESET\_VALUE  
STM8AF\_STM8S, [266](#)  
\_ITC\_SPR6  
STM8AF\_STM8S, [266](#)  
\_ITC\_SPR6\_RESET\_VALUE  
STM8AF\_STM8S, [266](#)  
\_ITC\_SPR7  
STM8AF\_STM8S, [266](#)  
\_ITC\_SPR7\_RESET\_VALUE  
STM8AF\_STM8S, [266](#)  
\_ITC\_SPR8  
STM8AF\_STM8S, [267](#)  
\_ITC\_SPR8\_RESET\_VALUE  
STM8AF\_STM8S, [267](#)  
\_ITC\_VECT10SPR  
STM8AF\_STM8S, [267](#)  
\_ITC\_VECT10SPR0  
STM8AF\_STM8S, [267](#)  
\_ITC\_VECT10SPR1  
STM8AF\_STM8S, [267](#)  
\_ITC\_VECT11SPR  
STM8AF\_STM8S, [268](#)  
\_ITC\_VECT11SPR0  
STM8AF\_STM8S, [268](#)  
\_ITC\_VECT11SPR1  
STM8AF\_STM8S, [268](#)  
\_ITC\_VECT12SPR  
STM8AF\_STM8S, [268](#)  
\_ITC\_VECT12SPR0  
STM8AF\_STM8S, [268](#)  
\_ITC\_VECT12SPR1  
STM8AF\_STM8S, [269](#)  
\_ITC\_VECT13SPR  
STM8AF\_STM8S, [269](#)  
\_ITC\_VECT13SPR0  
STM8AF\_STM8S, [269](#)  
\_ITC\_VECT13SPR1  
STM8AF\_STM8S, [269](#)  
\_ITC\_VECT14SPR  
STM8AF\_STM8S, [269](#)  
\_ITC\_VECT14SPR0



STM8AF\_STM8S, [270](#)  
\_ITC\_VECT14SPR1  
STM8AF\_STM8S, [270](#)  
\_ITC\_VECT15SPR  
STM8AF\_STM8S, [270](#)  
\_ITC\_VECT15SPR0  
STM8AF\_STM8S, [270](#)  
\_ITC\_VECT15SPR1  
STM8AF\_STM8S, [270](#)  
\_ITC\_VECT16SPR  
STM8AF\_STM8S, [271](#)  
\_ITC\_VECT16SPR0  
STM8AF\_STM8S, [271](#)  
\_ITC\_VECT16SPR1  
STM8AF\_STM8S, [271](#)  
\_ITC\_VECT17SPR  
STM8AF\_STM8S, [271](#)  
\_ITC\_VECT17SPR0  
STM8AF\_STM8S, [271](#)  
\_ITC\_VECT17SPR1  
STM8AF\_STM8S, [272](#)  
\_ITC\_VECT18SPR  
STM8AF\_STM8S, [272](#)  
\_ITC\_VECT18SPR0  
STM8AF\_STM8S, [272](#)  
\_ITC\_VECT18SPR1  
STM8AF\_STM8S, [272](#)  
\_ITC\_VECT19SPR  
STM8AF\_STM8S, [272](#)  
\_ITC\_VECT19SPR0  
STM8AF\_STM8S, [273](#)  
\_ITC\_VECT19SPR1  
STM8AF\_STM8S, [273](#)  
\_ITC\_VECT1SPR  
STM8AF\_STM8S, [273](#)  
\_ITC\_VECT1SPR0  
STM8AF\_STM8S, [273](#)  
\_ITC\_VECT1SPR1  
STM8AF\_STM8S, [273](#)  
\_ITC\_VECT20SPR  
STM8AF\_STM8S, [274](#)  
\_ITC\_VECT20SPR0  
STM8AF\_STM8S, [274](#)  
\_ITC\_VECT20SPR1  
STM8AF\_STM8S, [274](#)  
\_ITC\_VECT21SPR  
STM8AF\_STM8S, [274](#)  
\_ITC\_VECT21SPR0  
STM8AF\_STM8S, [274](#)  
\_ITC\_VECT21SPR1  
STM8AF\_STM8S, [275](#)  
\_ITC\_VECT22SPR  
STM8AF\_STM8S, [275](#)  
\_ITC\_VECT22SPR0  
STM8AF\_STM8S, [275](#)  
\_ITC\_VECT22SPR1  
STM8AF\_STM8S, [275](#)  
\_ITC\_VECT23SPR  
STM8AF\_STM8S, [275](#)  
\_ITC\_VECT23SPR0  
STM8AF\_STM8S, [276](#)  
\_ITC\_VECT23SPR1  
STM8AF\_STM8S, [276](#)  
\_ITC\_VECT24SPR  
STM8AF\_STM8S, [276](#)  
\_ITC\_VECT24SPR0  
STM8AF\_STM8S, [276](#)  
\_ITC\_VECT24SPR1  
STM8AF\_STM8S, [276](#)  
\_ITC\_VECT25SPR  
STM8AF\_STM8S, [277](#)  
\_ITC\_VECT25SPR0  
STM8AF\_STM8S, [277](#)  
\_ITC\_VECT25SPR1  
STM8AF\_STM8S, [277](#)  
\_ITC\_VECT26SPR  
STM8AF\_STM8S, [277](#)  
\_ITC\_VECT26SPR0  
STM8AF\_STM8S, [277](#)  
\_ITC\_VECT26SPR1  
STM8AF\_STM8S, [278](#)  
\_ITC\_VECT27SPR  
STM8AF\_STM8S, [278](#)  
\_ITC\_VECT27SPR0  
STM8AF\_STM8S, [278](#)  
\_ITC\_VECT27SPR1  
STM8AF\_STM8S, [278](#)  
\_ITC\_VECT28SPR  
STM8AF\_STM8S, [278](#)  
\_ITC\_VECT28SPR0  
STM8AF\_STM8S, [279](#)  
\_ITC\_VECT28SPR1  
STM8AF\_STM8S, [279](#)  
\_ITC\_VECT29SPR  
STM8AF\_STM8S, [279](#)  
\_ITC\_VECT29SPR0  
STM8AF\_STM8S, [279](#)  
\_ITC\_VECT29SPR1  
STM8AF\_STM8S, [279](#)  
\_ITC\_VECT2SPR  
STM8AF\_STM8S, [280](#)  
\_ITC\_VECT2SPR0  
STM8AF\_STM8S, [280](#)  
\_ITC\_VECT2SPR1  
STM8AF\_STM8S, [280](#)  
\_ITC\_VECT3SPR  
STM8AF\_STM8S, [280](#)  
\_ITC\_VECT3SPR0  
STM8AF\_STM8S, [280](#)  
\_ITC\_VECT3SPR1  
STM8AF\_STM8S, [281](#)  
\_ITC\_VECT4SPR  
STM8AF\_STM8S, [281](#)  
\_ITC\_VECT4SPR0  
STM8AF\_STM8S, [281](#)  
\_ITC\_VECT4SPR1

STM8AF\_STM8S, [281](#)  
\_ITC\_VECT5SPR  
STM8AF\_STM8S, [281](#)  
\_ITC\_VECT5SPR0  
STM8AF\_STM8S, [282](#)  
\_ITC\_VECT5SPR1  
STM8AF\_STM8S, [282](#)  
\_ITC\_VECT6SPR  
STM8AF\_STM8S, [282](#)  
\_ITC\_VECT6SPR0  
STM8AF\_STM8S, [282](#)  
\_ITC\_VECT6SPR1  
STM8AF\_STM8S, [282](#)  
\_ITC\_VECT7SPR  
STM8AF\_STM8S, [283](#)  
\_ITC\_VECT7SPR0  
STM8AF\_STM8S, [283](#)  
\_ITC\_VECT7SPR1  
STM8AF\_STM8S, [283](#)  
\_ITC\_VECT8SPR  
STM8AF\_STM8S, [283](#)  
\_ITC\_VECT8SPR0  
STM8AF\_STM8S, [283](#)  
\_ITC\_VECT8SPR1  
STM8AF\_STM8S, [284](#)  
\_ITC\_VECT9SPR  
STM8AF\_STM8S, [284](#)  
\_ITC\_VECT9SPR0  
STM8AF\_STM8S, [284](#)  
\_ITC\_VECT9SPR1  
STM8AF\_STM8S, [284](#)  
\_IWDG  
STM8AF\_STM8S, [284](#)  
\_IWDG\_KR  
STM8AF\_STM8S, [285](#)  
\_IWDG\_KEY\_ACCESS  
STM8AF\_STM8S, [285](#)  
\_IWDG\_KEY\_ENABLE  
STM8AF\_STM8S, [285](#)  
\_IWDG\_KEY\_REFRESH  
STM8AF\_STM8S, [285](#)  
\_IWDG\_PR  
STM8AF\_STM8S, [285](#)  
\_IWDG\_PRE  
STM8AF\_STM8S, [286](#)  
\_IWDG\_PRE0  
STM8AF\_STM8S, [286](#)  
\_IWDG\_PRE1  
STM8AF\_STM8S, [286](#)  
\_IWDG\_PRE2  
STM8AF\_STM8S, [286](#)  
\_IWDG\_PR\_RESET\_VALUE  
STM8AF\_STM8S, [286](#)  
\_IWDG\_RLR  
STM8AF\_STM8S, [287](#)  
\_IWDG\_RLR\_RESET\_VALUE  
STM8AF\_STM8S, [287](#)  
\_RST  
STM8AF\_STM8S, [287](#)  
\_RST\_EMCF  
STM8AF\_STM8S, [287](#)  
\_RST\_ILLOPF  
STM8AF\_STM8S, [287](#)  
\_RST\_IWDGF  
STM8AF\_STM8S, [288](#)  
\_RST\_SR  
STM8AF\_STM8S, [288](#)  
\_RST\_SWIMF  
STM8AF\_STM8S, [288](#)  
\_RST\_WWDGF  
STM8AF\_STM8S, [288](#)  
\_SFR  
STM8AF\_STM8S, [288](#)  
\_SPI  
STM8AF\_STM8S, [289](#)  
\_SPI\_BR  
STM8AF\_STM8S, [289](#)  
\_SPI\_BDM  
STM8AF\_STM8S, [289](#)  
\_SPI\_BDOE  
STM8AF\_STM8S, [289](#)  
\_SPI\_BR0  
STM8AF\_STM8S, [289](#)  
\_SPI\_BR1  
STM8AF\_STM8S, [290](#)  
\_SPI\_BR2  
STM8AF\_STM8S, [290](#)  
\_SPI\_BSY  
STM8AF\_STM8S, [290](#)  
\_SPI\_CPHA  
STM8AF\_STM8S, [290](#)  
\_SPI\_CPOL  
STM8AF\_STM8S, [290](#)  
\_SPI\_CR1  
STM8AF\_STM8S, [291](#)  
\_SPI\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [291](#)  
\_SPI\_CR2  
STM8AF\_STM8S, [291](#)  
\_SPI\_CR2\_RESET\_VALUE  
STM8AF\_STM8S, [291](#)  
\_SPI\_CRCEN  
STM8AF\_STM8S, [291](#)  
\_SPI\_CRCERR  
STM8AF\_STM8S, [292](#)  
\_SPI\_CRCNEXT  
STM8AF\_STM8S, [292](#)  
\_SPI\_CRCPR  
STM8AF\_STM8S, [292](#)  
\_SPI\_CRCPR\_RESET\_VALUE  
STM8AF\_STM8S, [292](#)  
\_SPI\_DR  
STM8AF\_STM8S, [292](#)  
\_SPI\_DR\_RESET\_VALUE  
STM8AF\_STM8S, [293](#)  
\_SPI\_ERRIE

- STM8AF\_STM8S, [293](#)
- \_SPI\_ICR
  - STM8AF\_STM8S, [293](#)
- \_SPI\_ICR\_RESET\_VALUE
  - STM8AF\_STM8S, [293](#)
- \_SPI\_LSBFIRST
  - STM8AF\_STM8S, [293](#)
- \_SPI\_MODF
  - STM8AF\_STM8S, [294](#)
- \_SPI\_MSTR
  - STM8AF\_STM8S, [294](#)
- \_SPI\_OVR
  - STM8AF\_STM8S, [294](#)
- \_SPI\_RXCRCR
  - STM8AF\_STM8S, [294](#)
- \_SPI\_RXCRCR\_RESET\_VALUE
  - STM8AF\_STM8S, [294](#)
- \_SPI\_RXIE
  - STM8AF\_STM8S, [295](#)
- \_SPI\_RXNE
  - STM8AF\_STM8S, [295](#)
- \_SPI\_RXONLY
  - STM8AF\_STM8S, [295](#)
- \_SPI\_SR
  - STM8AF\_STM8S, [295](#)
- \_SPI\_SPE
  - STM8AF\_STM8S, [295](#)
- \_SPI\_SR\_RESET\_VALUE
  - STM8AF\_STM8S, [296](#)
- \_SPI\_SSI
  - STM8AF\_STM8S, [296](#)
- \_SPI\_SSM
  - STM8AF\_STM8S, [296](#)
- \_SPI\_TXE
  - STM8AF\_STM8S, [297](#)
- \_SPI\_TXCRCR
  - STM8AF\_STM8S, [296](#)
- \_SPI\_TXCRCR\_RESET\_VALUE
  - STM8AF\_STM8S, [296](#)
- \_SPI\_TXIE
  - STM8AF\_STM8S, [297](#)
- \_SPI\_WKIE
  - STM8AF\_STM8S, [297](#)
- \_SPI\_WKUP
  - STM8AF\_STM8S, [297](#)
- \_TIM1
  - STM8AF\_STM8S, [297](#)
- \_TIM1\_AOE
  - STM8AF\_STM8S, [298](#)
- \_TIM1\_ARPE
  - STM8AF\_STM8S, [298](#)
- \_TIM1\_ARRH
  - STM8AF\_STM8S, [298](#)
- \_TIM1\_ARRL
  - STM8AF\_STM8S, [298](#)
- \_TIM1\_ARRH\_RESET\_VALUE
  - STM8AF\_STM8S, [298](#)
- \_TIM1\_ARRL\_RESET\_VALUE
  - STM8AF\_STM8S, [299](#)
- \_TIM1\_BG
  - STM8AF\_STM8S, [299](#)
- \_TIM1\_BIE
  - STM8AF\_STM8S, [299](#)
- \_TIM1\_BIF
  - STM8AF\_STM8S, [299](#)
- \_TIM1\_BKE
  - STM8AF\_STM8S, [299](#)
- \_TIM1\_BKP
  - STM8AF\_STM8S, [300](#)
- \_TIM1\_BKR
  - STM8AF\_STM8S, [300](#)
- \_TIM1\_BKR\_RESET\_VALUE
  - STM8AF\_STM8S, [300](#)
- \_TIM1\_CC1E
  - STM8AF\_STM8S, [300](#)
- \_TIM1\_CC1G
  - STM8AF\_STM8S, [300](#)
- \_TIM1\_CC1P
  - STM8AF\_STM8S, [302](#)
- \_TIM1\_CC1S
  - STM8AF\_STM8S, [302](#)
- \_TIM1\_CC1IE
  - STM8AF\_STM8S, [301](#)
- \_TIM1\_CC1IF
  - STM8AF\_STM8S, [301](#)
- \_TIM1\_CC1NE
  - STM8AF\_STM8S, [301](#)
- \_TIM1\_CC1NP
  - STM8AF\_STM8S, [301](#)
- \_TIM1\_CC1OF
  - STM8AF\_STM8S, [301](#)
- \_TIM1\_CC1S0
  - STM8AF\_STM8S, [302](#)
- \_TIM1\_CC1S1
  - STM8AF\_STM8S, [302](#)
- \_TIM1\_CC2E
  - STM8AF\_STM8S, [302](#)
- \_TIM1\_CC2G
  - STM8AF\_STM8S, [303](#)
- \_TIM1\_CC2P
  - STM8AF\_STM8S, [304](#)
- \_TIM1\_CC2S
  - STM8AF\_STM8S, [304](#)
- \_TIM1\_CC2IE
  - STM8AF\_STM8S, [303](#)
- \_TIM1\_CC2IF
  - STM8AF\_STM8S, [303](#)
- \_TIM1\_CC2NE
  - STM8AF\_STM8S, [303](#)
- \_TIM1\_CC2NP
  - STM8AF\_STM8S, [303](#)
- \_TIM1\_CC2OF
  - STM8AF\_STM8S, [304](#)
- \_TIM1\_CC2S0
  - STM8AF\_STM8S, [304](#)
- \_TIM1\_CC2S1

STM8AF\_STM8S, [304](#)  
 \_TIM1\_CC3E  
   STM8AF\_STM8S, [305](#)  
 \_TIM1\_CC3G  
   STM8AF\_STM8S, [305](#)  
 \_TIM1\_CC3P  
   STM8AF\_STM8S, [306](#)  
 \_TIM1\_CC3S  
   STM8AF\_STM8S, [306](#)  
 \_TIM1\_CC3IE  
   STM8AF\_STM8S, [305](#)  
 \_TIM1\_CC3IF  
   STM8AF\_STM8S, [305](#)  
 \_TIM1\_CC3NE  
   STM8AF\_STM8S, [305](#)  
 \_TIM1\_CC3NP  
   STM8AF\_STM8S, [306](#)  
 \_TIM1\_CC3OF  
   STM8AF\_STM8S, [306](#)  
 \_TIM1\_CC3S0  
   STM8AF\_STM8S, [306](#)  
 \_TIM1\_CC3S1  
   STM8AF\_STM8S, [307](#)  
 \_TIM1\_CC4E  
   STM8AF\_STM8S, [307](#)  
 \_TIM1\_CC4G  
   STM8AF\_STM8S, [307](#)  
 \_TIM1\_CC4P  
   STM8AF\_STM8S, [308](#)  
 \_TIM1\_CC4S  
   STM8AF\_STM8S, [308](#)  
 \_TIM1\_CC4IE  
   STM8AF\_STM8S, [307](#)  
 \_TIM1\_CC4IF  
   STM8AF\_STM8S, [307](#)  
 \_TIM1\_CC4OF  
   STM8AF\_STM8S, [308](#)  
 \_TIM1\_CC4S0  
   STM8AF\_STM8S, [308](#)  
 \_TIM1\_CC4S1  
   STM8AF\_STM8S, [308](#)  
 \_TIM1\_CCER1  
   STM8AF\_STM8S, [309](#)  
 \_TIM1\_CCER1\_RESET\_VALUE  
   STM8AF\_STM8S, [309](#)  
 \_TIM1\_CCER2  
   STM8AF\_STM8S, [309](#)  
 \_TIM1\_CCER2\_RESET\_VALUE  
   STM8AF\_STM8S, [309](#)  
 \_TIM1\_CCMR1  
   STM8AF\_STM8S, [309](#)  
 \_TIM1\_CCMR1\_RESET\_VALUE  
   STM8AF\_STM8S, [310](#)  
 \_TIM1\_CCMR2  
   STM8AF\_STM8S, [310](#)  
 \_TIM1\_CCMR2\_RESET\_VALUE  
   STM8AF\_STM8S, [310](#)  
 \_TIM1\_CCMR3  
   STM8AF\_STM8S, [310](#)  
 \_TIM1\_CCMR3\_RESET\_VALUE  
   STM8AF\_STM8S, [310](#)  
 \_TIM1\_CCMR4  
   STM8AF\_STM8S, [311](#)  
 \_TIM1\_CCMR4\_RESET\_VALUE  
   STM8AF\_STM8S, [311](#)  
 \_TIM1\_CCPC  
   STM8AF\_STM8S, [311](#)  
 \_TIM1\_CCR1H  
   STM8AF\_STM8S, [311](#)  
 \_TIM1\_CCR1L  
   STM8AF\_STM8S, [312](#)  
 \_TIM1\_CCR1H\_RESET\_VALUE  
   STM8AF\_STM8S, [311](#)  
 \_TIM1\_CCR1L\_RESET\_VALUE  
   STM8AF\_STM8S, [312](#)  
 \_TIM1\_CCR2H  
   STM8AF\_STM8S, [312](#)  
 \_TIM1\_CCR2L  
   STM8AF\_STM8S, [312](#)  
 \_TIM1\_CCR2H\_RESET\_VALUE  
   STM8AF\_STM8S, [312](#)  
 \_TIM1\_CCR2L\_RESET\_VALUE  
   STM8AF\_STM8S, [313](#)  
 \_TIM1\_CCR3H  
   STM8AF\_STM8S, [313](#)  
 \_TIM1\_CCR3L  
   STM8AF\_STM8S, [313](#)  
 \_TIM1\_CCR3H\_RESET\_VALUE  
   STM8AF\_STM8S, [313](#)  
 \_TIM1\_CCR3L\_RESET\_VALUE  
   STM8AF\_STM8S, [313](#)  
 \_TIM1\_CCR4H  
   STM8AF\_STM8S, [314](#)  
 \_TIM1\_CCR4L  
   STM8AF\_STM8S, [314](#)  
 \_TIM1\_CCR4H\_RESET\_VALUE  
   STM8AF\_STM8S, [314](#)  
 \_TIM1\_CCR4L\_RESET\_VALUE  
   STM8AF\_STM8S, [314](#)  
 \_TIM1\_CEN  
   STM8AF\_STM8S, [314](#)  
 \_TIM1\_CMS  
   STM8AF\_STM8S, [315](#)  
 \_TIM1\_CMS0  
   STM8AF\_STM8S, [315](#)  
 \_TIM1\_CMS1  
   STM8AF\_STM8S, [315](#)  
 \_TIM1\_CNTRH  
   STM8AF\_STM8S, [315](#)  
 \_TIM1\_CNTRL  
   STM8AF\_STM8S, [316](#)  
 \_TIM1\_CNTRH\_RESET\_VALUE  
   STM8AF\_STM8S, [315](#)  
 \_TIM1\_CNTRL\_RESET\_VALUE  
   STM8AF\_STM8S, [316](#)  
 \_TIM1\_COMG

STM8AF\_STM8S, [316](#)  
\_TIM1\_COMS  
STM8AF\_STM8S, [317](#)  
\_TIM1\_COMIE  
STM8AF\_STM8S, [316](#)  
\_TIM1\_COMIF  
STM8AF\_STM8S, [316](#)  
\_TIM1\_CR1  
STM8AF\_STM8S, [317](#)  
\_TIM1\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [317](#)  
\_TIM1\_CR2  
STM8AF\_STM8S, [317](#)  
\_TIM1\_CR2\_RESET\_VALUE  
STM8AF\_STM8S, [317](#)  
\_TIM1\_DIR  
STM8AF\_STM8S, [318](#)  
\_TIM1\_DTR  
STM8AF\_STM8S, [318](#)  
\_TIM1\_DTR\_RESET\_VALUE  
STM8AF\_STM8S, [318](#)  
\_TIM1\_ECE  
STM8AF\_STM8S, [318](#)  
\_TIM1\_EGR  
STM8AF\_STM8S, [318](#)  
\_TIM1\_EGR\_RESET\_VALUE  
STM8AF\_STM8S, [319](#)  
\_TIM1 ETF  
STM8AF\_STM8S, [319](#)  
\_TIM1\_ETP  
STM8AF\_STM8S, [320](#)  
\_TIM1\_ETR  
STM8AF\_STM8S, [321](#)  
\_TIM1 ETF0  
STM8AF\_STM8S, [319](#)  
\_TIM1 ETF1  
STM8AF\_STM8S, [319](#)  
\_TIM1 ETF2  
STM8AF\_STM8S, [319](#)  
\_TIM1 ETF3  
STM8AF\_STM8S, [320](#)  
\_TIM1\_ETPS  
STM8AF\_STM8S, [320](#)  
\_TIM1\_ETPS0  
STM8AF\_STM8S, [320](#)  
\_TIM1\_ETPS1  
STM8AF\_STM8S, [320](#)  
\_TIM1\_ETR\_RESET\_VALUE  
STM8AF\_STM8S, [321](#)  
\_TIM1\_IC1F  
STM8AF\_STM8S, [321](#)  
\_TIM1\_IC1F0  
STM8AF\_STM8S, [321](#)  
\_TIM1\_IC1F1  
STM8AF\_STM8S, [321](#)  
\_TIM1\_IC1F2  
STM8AF\_STM8S, [322](#)  
\_TIM1\_IC1F3  
STM8AF\_STM8S, [322](#)  
\_TIM1\_IC1PSC  
STM8AF\_STM8S, [322](#)  
\_TIM1\_IC1PSC0  
STM8AF\_STM8S, [322](#)  
\_TIM1\_IC1PSC1  
STM8AF\_STM8S, [322](#)  
\_TIM1\_IC2F  
STM8AF\_STM8S, [323](#)  
\_TIM1\_IC2F0  
STM8AF\_STM8S, [323](#)  
\_TIM1\_IC2F1  
STM8AF\_STM8S, [323](#)  
\_TIM1\_IC2F2  
STM8AF\_STM8S, [323](#)  
\_TIM1\_IC2F3  
STM8AF\_STM8S, [323](#)  
\_TIM1\_IC2PSC  
STM8AF\_STM8S, [324](#)  
\_TIM1\_IC2PSC0  
STM8AF\_STM8S, [324](#)  
\_TIM1\_IC2PSC1  
STM8AF\_STM8S, [324](#)  
\_TIM1\_IC3F  
STM8AF\_STM8S, [324](#)  
\_TIM1\_IC3F0  
STM8AF\_STM8S, [324](#)  
\_TIM1\_IC3F1  
STM8AF\_STM8S, [325](#)  
\_TIM1\_IC3F2  
STM8AF\_STM8S, [325](#)  
\_TIM1\_IC3F3  
STM8AF\_STM8S, [325](#)  
\_TIM1\_IC3PSC  
STM8AF\_STM8S, [325](#)  
\_TIM1\_IC3PSC0  
STM8AF\_STM8S, [325](#)  
\_TIM1\_IC3PSC1  
STM8AF\_STM8S, [326](#)  
\_TIM1\_IC4F  
STM8AF\_STM8S, [326](#)  
\_TIM1\_IC4F0  
STM8AF\_STM8S, [326](#)  
\_TIM1\_IC4F1  
STM8AF\_STM8S, [326](#)  
\_TIM1\_IC4F2  
STM8AF\_STM8S, [326](#)  
\_TIM1\_IC4F3  
STM8AF\_STM8S, [327](#)  
\_TIM1\_IC4PSC  
STM8AF\_STM8S, [327](#)  
\_TIM1\_IC4PSC0  
STM8AF\_STM8S, [327](#)  
\_TIM1\_IC4PSC1  
STM8AF\_STM8S, [327](#)  
\_TIM1\_IER  
STM8AF\_STM8S, [327](#)  
\_TIM1\_IER\_RESET\_VALUE

STM8AF\_STM8S, [328](#)  
\_TIM1\_LOCK  
STM8AF\_STM8S, [328](#)  
\_TIM1\_LOCK0  
STM8AF\_STM8S, [328](#)  
\_TIM1\_LOCK1  
STM8AF\_STM8S, [328](#)  
\_TIM1\_MMS  
STM8AF\_STM8S, [328](#)  
\_TIM1\_MMS0  
STM8AF\_STM8S, [329](#)  
\_TIM1\_MMS1  
STM8AF\_STM8S, [329](#)  
\_TIM1\_MMS2  
STM8AF\_STM8S, [329](#)  
\_TIM1\_MOE  
STM8AF\_STM8S, [329](#)  
\_TIM1\_MSM  
STM8AF\_STM8S, [329](#)  
\_TIM1\_OC1M  
STM8AF\_STM8S, [330](#)  
\_TIM1\_OC1CE  
STM8AF\_STM8S, [330](#)  
\_TIM1\_OC1FE  
STM8AF\_STM8S, [330](#)  
\_TIM1\_OC1M0  
STM8AF\_STM8S, [330](#)  
\_TIM1\_OC1M1  
STM8AF\_STM8S, [330](#)  
\_TIM1\_OC1M2  
STM8AF\_STM8S, [331](#)  
\_TIM1\_OC1PE  
STM8AF\_STM8S, [331](#)  
\_TIM1\_OC2M  
STM8AF\_STM8S, [331](#)  
\_TIM1\_OC2CE  
STM8AF\_STM8S, [331](#)  
\_TIM1\_OC2FE  
STM8AF\_STM8S, [331](#)  
\_TIM1\_OC2M0  
STM8AF\_STM8S, [332](#)  
\_TIM1\_OC2M1  
STM8AF\_STM8S, [332](#)  
\_TIM1\_OC2M2  
STM8AF\_STM8S, [332](#)  
\_TIM1\_OC2PE  
STM8AF\_STM8S, [332](#)  
\_TIM1\_OC3M  
STM8AF\_STM8S, [333](#)  
\_TIM1\_OC3CE  
STM8AF\_STM8S, [332](#)  
\_TIM1\_OC3FE  
STM8AF\_STM8S, [333](#)  
\_TIM1\_OC3M0  
STM8AF\_STM8S, [333](#)  
\_TIM1\_OC3M1  
STM8AF\_STM8S, [333](#)  
\_TIM1\_OC3M2  
STM8AF\_STM8S, [333](#)  
\_TIM1\_OC3PE  
STM8AF\_STM8S, [334](#)  
\_TIM1\_OC4M  
STM8AF\_STM8S, [334](#)  
\_TIM1\_OC4CE  
STM8AF\_STM8S, [334](#)  
\_TIM1\_OC4FE  
STM8AF\_STM8S, [334](#)  
\_TIM1\_OC4M0  
STM8AF\_STM8S, [334](#)  
\_TIM1\_OC4M1  
STM8AF\_STM8S, [335](#)  
\_TIM1\_OC4M2  
STM8AF\_STM8S, [335](#)  
\_TIM1\_OC4PE  
STM8AF\_STM8S, [335](#)  
\_TIM1\_OIS1  
STM8AF\_STM8S, [335](#)  
\_TIM1\_OIS1N  
STM8AF\_STM8S, [335](#)  
\_TIM1\_OIS2  
STM8AF\_STM8S, [336](#)  
\_TIM1\_OIS2N  
STM8AF\_STM8S, [336](#)  
\_TIM1\_OIS3  
STM8AF\_STM8S, [336](#)  
\_TIM1\_OIS3N  
STM8AF\_STM8S, [336](#)  
\_TIM1\_OIS4  
STM8AF\_STM8S, [336](#)  
\_TIM1\_OISR  
STM8AF\_STM8S, [337](#)  
\_TIM1\_OISR\_RESET\_VALUE  
STM8AF\_STM8S, [337](#)  
\_TIM1\_OPM  
STM8AF\_STM8S, [337](#)  
\_TIM1\_OSSI  
STM8AF\_STM8S, [337](#)  
\_TIM1\_OSSR  
STM8AF\_STM8S, [337](#)  
\_TIM1\_PSCRH  
STM8AF\_STM8S, [338](#)  
\_TIM1\_PSCRL  
STM8AF\_STM8S, [338](#)  
\_TIM1\_PSCRH\_RESET\_VALUE  
STM8AF\_STM8S, [338](#)  
\_TIM1\_PSCRL\_RESET\_VALUE  
STM8AF\_STM8S, [338](#)  
\_TIM1\_RCR  
STM8AF\_STM8S, [338](#)  
\_TIM1\_RCR\_RESET\_VALUE  
STM8AF\_STM8S, [339](#)  
\_TIM1\_SMS  
STM8AF\_STM8S, [339](#)  
\_TIM1\_SMCR  
STM8AF\_STM8S, [339](#)  
\_TIM1\_SMCR\_RESET\_VALUE

- STM8AF\_STM8S, [339](#)
- \_TIM1\_SMS0
  - STM8AF\_STM8S, [339](#)
- \_TIM1\_SMS1
  - STM8AF\_STM8S, [340](#)
- \_TIM1\_SMS2
  - STM8AF\_STM8S, [340](#)
- \_TIM1\_SR1
  - STM8AF\_STM8S, [340](#)
- \_TIM1\_SR1\_RESET\_VALUE
  - STM8AF\_STM8S, [340](#)
- \_TIM1\_SR2
  - STM8AF\_STM8S, [340](#)
- \_TIM1\_SR2\_RESET\_VALUE
  - STM8AF\_STM8S, [341](#)
- \_TIM1\_TG
  - STM8AF\_STM8S, [341](#)
- \_TIM1\_TS
  - STM8AF\_STM8S, [341](#)
- \_TIM1\_TIE
  - STM8AF\_STM8S, [341](#)
- \_TIM1\_TIF
  - STM8AF\_STM8S, [341](#)
- \_TIM1\_TS0
  - STM8AF\_STM8S, [342](#)
- \_TIM1\_TS1
  - STM8AF\_STM8S, [342](#)
- \_TIM1\_TS2
  - STM8AF\_STM8S, [342](#)
- \_TIM1\_UG
  - STM8AF\_STM8S, [342](#)
- \_TIM1\_UDIS
  - STM8AF\_STM8S, [342](#)
- \_TIM1\_UIE
  - STM8AF\_STM8S, [343](#)
- \_TIM1\_UIF
  - STM8AF\_STM8S, [343](#)
- \_TIM1\_URS
  - STM8AF\_STM8S, [343](#)
- \_TIM2
  - STM8AF\_STM8S, [343](#)
- \_TIM2\_ARPE
  - STM8AF\_STM8S, [343](#)
- \_TIM2\_ARRH
  - STM8AF\_STM8S, [344](#)
- \_TIM2\_ARRL
  - STM8AF\_STM8S, [344](#)
- \_TIM2\_ARRH\_RESET\_VALUE
  - STM8AF\_STM8S, [344](#)
- \_TIM2\_ARRL\_RESET\_VALUE
  - STM8AF\_STM8S, [344](#)
- \_TIM2\_CC1E
  - STM8AF\_STM8S, [344](#)
- \_TIM2\_CC1G
  - STM8AF\_STM8S, [345](#)
- \_TIM2\_CC1P
  - STM8AF\_STM8S, [345](#)
- \_TIM2\_CC1S
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC1IE
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC1IF
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC1OF
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC1S0
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC1S1
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC2E
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC2G
  - STM8AF\_STM8S, [346](#)
- \_TIM2\_CC2P
  - STM8AF\_STM8S, [347](#)
- \_TIM2\_CC2S
  - STM8AF\_STM8S, [347](#)
- \_TIM2\_CC2IE
  - STM8AF\_STM8S, [347](#)
- \_TIM2\_CC2IF
  - STM8AF\_STM8S, [347](#)
- \_TIM2\_CC2OF
  - STM8AF\_STM8S, [347](#)
- \_TIM2\_CC2S0
  - STM8AF\_STM8S, [348](#)
- \_TIM2\_CC2S1
  - STM8AF\_STM8S, [348](#)
- \_TIM2\_CC3E
  - STM8AF\_STM8S, [348](#)
- \_TIM2\_CC3G
  - STM8AF\_STM8S, [348](#)
- \_TIM2\_CC3P
  - STM8AF\_STM8S, [349](#)
- \_TIM2\_CC3S
  - STM8AF\_STM8S, [349](#)
- \_TIM2\_CC3IE
  - STM8AF\_STM8S, [348](#)
- \_TIM2\_CC3IF
  - STM8AF\_STM8S, [349](#)
- \_TIM2\_CC3OF
  - STM8AF\_STM8S, [349](#)
- \_TIM2\_CC3S0
  - STM8AF\_STM8S, [349](#)
- \_TIM2\_CC3S1
  - STM8AF\_STM8S, [350](#)
- \_TIM2\_CCER1
  - STM8AF\_STM8S, [350](#)
- \_TIM2\_CCER1\_RESET\_VALUE
  - STM8AF\_STM8S, [350](#)
- \_TIM2\_CCER2
  - STM8AF\_STM8S, [350](#)
- \_TIM2\_CCER2\_RESET\_VALUE
  - STM8AF\_STM8S, [350](#)
- \_TIM2\_CCMR1
  - STM8AF\_STM8S, [351](#)
- \_TIM2\_CCMR1\_RESET\_VALUE

STM8AF\_STM8S, [351](#)  
\_TIM2\_CCMR2  
STM8AF\_STM8S, [351](#)  
\_TIM2\_CCMR2\_RESET\_VALUE  
STM8AF\_STM8S, [351](#)  
\_TIM2\_CCMR3  
STM8AF\_STM8S, [351](#)  
\_TIM2\_CCMR3\_RESET\_VALUE  
STM8AF\_STM8S, [352](#)  
\_TIM2\_CCR1H  
STM8AF\_STM8S, [352](#)  
\_TIM2\_CCR1L  
STM8AF\_STM8S, [352](#)  
\_TIM2\_CCR1H\_RESET\_VALUE  
STM8AF\_STM8S, [352](#)  
\_TIM2\_CCR1L\_RESET\_VALUE  
STM8AF\_STM8S, [352](#)  
\_TIM2\_CCR2H  
STM8AF\_STM8S, [353](#)  
\_TIM2\_CCR2L  
STM8AF\_STM8S, [353](#)  
\_TIM2\_CCR2H\_RESET\_VALUE  
STM8AF\_STM8S, [353](#)  
\_TIM2\_CCR2L\_RESET\_VALUE  
STM8AF\_STM8S, [353](#)  
\_TIM2\_CCR3H  
STM8AF\_STM8S, [353](#)  
\_TIM2\_CCR3L  
STM8AF\_STM8S, [354](#)  
\_TIM2\_CCR3H\_RESET\_VALUE  
STM8AF\_STM8S, [354](#)  
\_TIM2\_CCR3L\_RESET\_VALUE  
STM8AF\_STM8S, [354](#)  
\_TIM2\_CEN  
STM8AF\_STM8S, [354](#)  
\_TIM2\_CNTRH  
STM8AF\_STM8S, [354](#)  
\_TIM2\_CNTRL  
STM8AF\_STM8S, [355](#)  
\_TIM2\_CNTRH\_RESET\_VALUE  
STM8AF\_STM8S, [355](#)  
\_TIM2\_CNTRL\_RESET\_VALUE  
STM8AF\_STM8S, [355](#)  
\_TIM2\_CR1  
STM8AF\_STM8S, [355](#)  
\_TIM2\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [355](#)  
\_TIM2\_EGR  
STM8AF\_STM8S, [356](#)  
\_TIM2\_EGR\_RESET\_VALUE  
STM8AF\_STM8S, [356](#)  
\_TIM2\_IC1F  
STM8AF\_STM8S, [356](#)  
\_TIM2\_IC1F0  
STM8AF\_STM8S, [356](#)  
\_TIM2\_IC1F1  
STM8AF\_STM8S, [356](#)  
\_TIM2\_IC1F2  
STM8AF\_STM8S, [357](#)  
\_TIM2\_IC1F3  
STM8AF\_STM8S, [357](#)  
\_TIM2\_IC1PSC  
STM8AF\_STM8S, [357](#)  
\_TIM2\_IC1PSC0  
STM8AF\_STM8S, [357](#)  
\_TIM2\_IC1PSC1  
STM8AF\_STM8S, [357](#)  
\_TIM2\_IC2F  
STM8AF\_STM8S, [358](#)  
\_TIM2\_IC2F0  
STM8AF\_STM8S, [358](#)  
\_TIM2\_IC2F1  
STM8AF\_STM8S, [358](#)  
\_TIM2\_IC2F2  
STM8AF\_STM8S, [358](#)  
\_TIM2\_IC2F3  
STM8AF\_STM8S, [358](#)  
\_TIM2\_IC2PSC  
STM8AF\_STM8S, [359](#)  
\_TIM2\_IC2PSC0  
STM8AF\_STM8S, [359](#)  
\_TIM2\_IC2PSC1  
STM8AF\_STM8S, [359](#)  
\_TIM2\_IC3F  
STM8AF\_STM8S, [359](#)  
\_TIM2\_IC3F0  
STM8AF\_STM8S, [359](#)  
\_TIM2\_IC3F1  
STM8AF\_STM8S, [360](#)  
\_TIM2\_IC3F2  
STM8AF\_STM8S, [360](#)  
\_TIM2\_IC3F3  
STM8AF\_STM8S, [360](#)  
\_TIM2\_IC3PSC  
STM8AF\_STM8S, [360](#)  
\_TIM2\_IC3PSC0  
STM8AF\_STM8S, [360](#)  
\_TIM2\_IC3PSC1  
STM8AF\_STM8S, [361](#)  
\_TIM2\_IER  
STM8AF\_STM8S, [361](#)  
\_TIM2\_IER\_RESET\_VALUE  
STM8AF\_STM8S, [361](#)  
\_TIM2\_OC1M  
STM8AF\_STM8S, [361](#)  
\_TIM2\_OC1M0  
STM8AF\_STM8S, [361](#)  
\_TIM2\_OC1M1  
STM8AF\_STM8S, [362](#)  
\_TIM2\_OC1M2  
STM8AF\_STM8S, [362](#)  
\_TIM2\_OC1PE  
STM8AF\_STM8S, [362](#)  
\_TIM2\_OC2M  
STM8AF\_STM8S, [362](#)  
\_TIM2\_OC2M0



STM8AF\_STM8S, [362](#)  
\_TIM2\_OC2M1  
STM8AF\_STM8S, [363](#)  
\_TIM2\_OC2M2  
STM8AF\_STM8S, [363](#)  
\_TIM2\_OC2PE  
STM8AF\_STM8S, [363](#)  
\_TIM2\_OC3M  
STM8AF\_STM8S, [363](#)  
\_TIM2\_OC3M0  
STM8AF\_STM8S, [363](#)  
\_TIM2\_OC3M1  
STM8AF\_STM8S, [364](#)  
\_TIM2\_OC3M2  
STM8AF\_STM8S, [364](#)  
\_TIM2\_OC3PE  
STM8AF\_STM8S, [364](#)  
\_TIM2\_OPM  
STM8AF\_STM8S, [364](#)  
\_TIM2\_PSC  
STM8AF\_STM8S, [364](#)  
\_TIM2\_PSC0  
STM8AF\_STM8S, [365](#)  
\_TIM2\_PSC1  
STM8AF\_STM8S, [365](#)  
\_TIM2\_PSC2  
STM8AF\_STM8S, [365](#)  
\_TIM2\_PSC3  
STM8AF\_STM8S, [365](#)  
\_TIM2\_PSCR  
STM8AF\_STM8S, [365](#)  
\_TIM2\_PSCR\_RESET\_VALUE  
STM8AF\_STM8S, [366](#)  
\_TIM2\_SR1  
STM8AF\_STM8S, [366](#)  
\_TIM2\_SR1\_RESET\_VALUE  
STM8AF\_STM8S, [366](#)  
\_TIM2\_SR2  
STM8AF\_STM8S, [366](#)  
\_TIM2\_SR2\_RESET\_VALUE  
STM8AF\_STM8S, [366](#)  
\_TIM2\_UG  
STM8AF\_STM8S, [367](#)  
\_TIM2\_UDIS  
STM8AF\_STM8S, [367](#)  
\_TIM2\_UIE  
STM8AF\_STM8S, [367](#)  
\_TIM2\_UIF  
STM8AF\_STM8S, [367](#)  
\_TIM2\_URS  
STM8AF\_STM8S, [367](#)  
\_TIM3  
STM8AF\_STM8S, [368](#)  
\_TIM3\_ARPE  
STM8AF\_STM8S, [368](#)  
\_TIM3\_ARRH  
STM8AF\_STM8S, [368](#)  
\_TIM3\_ARRL  
STM8AF\_STM8S, [368](#)  
\_TIM3\_ARRH\_RESET\_VALUE  
STM8AF\_STM8S, [368](#)  
\_TIM3\_ARRL\_RESET\_VALUE  
STM8AF\_STM8S, [369](#)  
\_TIM3\_CC1E  
STM8AF\_STM8S, [369](#)  
\_TIM3\_CC1G  
STM8AF\_STM8S, [369](#)  
\_TIM3\_CC1P  
STM8AF\_STM8S, [370](#)  
\_TIM3\_CC1S  
STM8AF\_STM8S, [370](#)  
\_TIM3\_CC1IE  
STM8AF\_STM8S, [369](#)  
\_TIM3\_CC1IF  
STM8AF\_STM8S, [369](#)  
\_TIM3\_CC1OF  
STM8AF\_STM8S, [370](#)  
\_TIM3\_CC1S0  
STM8AF\_STM8S, [370](#)  
\_TIM3\_CC1S1  
STM8AF\_STM8S, [370](#)  
\_TIM3\_CC2E  
STM8AF\_STM8S, [371](#)  
\_TIM3\_CC2G  
STM8AF\_STM8S, [371](#)  
\_TIM3\_CC2P  
STM8AF\_STM8S, [372](#)  
\_TIM3\_CC2S  
STM8AF\_STM8S, [372](#)  
\_TIM3\_CC2IE  
STM8AF\_STM8S, [371](#)  
\_TIM3\_CC2IF  
STM8AF\_STM8S, [371](#)  
\_TIM3\_CC2OF  
STM8AF\_STM8S, [371](#)  
\_TIM3\_CC2S0  
STM8AF\_STM8S, [372](#)  
\_TIM3\_CC2S1  
STM8AF\_STM8S, [372](#)  
\_TIM3\_CCER1  
STM8AF\_STM8S, [372](#)  
\_TIM3\_CCER1\_RESET\_VALUE  
STM8AF\_STM8S, [373](#)  
\_TIM3\_CCMR1  
STM8AF\_STM8S, [373](#)  
\_TIM3\_CCMR1\_RESET\_VALUE  
STM8AF\_STM8S, [373](#)  
\_TIM3\_CCMR2  
STM8AF\_STM8S, [373](#)  
\_TIM3\_CCMR2\_RESET\_VALUE  
STM8AF\_STM8S, [373](#)  
\_TIM3\_CCR1H  
STM8AF\_STM8S, [374](#)  
\_TIM3\_CCR1L  
STM8AF\_STM8S, [374](#)  
\_TIM3\_CCR1H\_RESET\_VALUE

STM8AF\_STM8S, [374](#)  
\_TIM3\_CCR1L\_RESET\_VALUE  
STM8AF\_STM8S, [374](#)  
\_TIM3\_CCR2H  
STM8AF\_STM8S, [374](#)  
\_TIM3\_CCR2L  
STM8AF\_STM8S, [375](#)  
\_TIM3\_CCR2H\_RESET\_VALUE  
STM8AF\_STM8S, [375](#)  
\_TIM3\_CCR2L\_RESET\_VALUE  
STM8AF\_STM8S, [375](#)  
\_TIM3\_CEN  
STM8AF\_STM8S, [375](#)  
\_TIM3\_CNTRH  
STM8AF\_STM8S, [375](#)  
\_TIM3\_CNTRL  
STM8AF\_STM8S, [376](#)  
\_TIM3\_CNTRH\_RESET\_VALUE  
STM8AF\_STM8S, [376](#)  
\_TIM3\_CNTRL\_RESET\_VALUE  
STM8AF\_STM8S, [376](#)  
\_TIM3\_CR1  
STM8AF\_STM8S, [376](#)  
\_TIM3\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [376](#)  
\_TIM3\_EGR  
STM8AF\_STM8S, [377](#)  
\_TIM3\_EGR\_RESET\_VALUE  
STM8AF\_STM8S, [377](#)  
\_TIM3\_IC1F  
STM8AF\_STM8S, [377](#)  
\_TIM3\_IC1F0  
STM8AF\_STM8S, [377](#)  
\_TIM3\_IC1F1  
STM8AF\_STM8S, [377](#)  
\_TIM3\_IC1F2  
STM8AF\_STM8S, [378](#)  
\_TIM3\_IC1F3  
STM8AF\_STM8S, [378](#)  
\_TIM3\_IC1PSC  
STM8AF\_STM8S, [378](#)  
\_TIM3\_IC1PSC0  
STM8AF\_STM8S, [378](#)  
\_TIM3\_IC1PSC1  
STM8AF\_STM8S, [378](#)  
\_TIM3\_IC2F  
STM8AF\_STM8S, [379](#)  
\_TIM3\_IC2F0  
STM8AF\_STM8S, [379](#)  
\_TIM3\_IC2F1  
STM8AF\_STM8S, [379](#)  
\_TIM3\_IC2F2  
STM8AF\_STM8S, [379](#)  
\_TIM3\_IC2F3  
STM8AF\_STM8S, [379](#)  
\_TIM3\_IC2PSC  
STM8AF\_STM8S, [380](#)  
\_TIM3\_IC2PSC0  
STM8AF\_STM8S, [380](#)  
\_TIM3\_IC2PSC1  
STM8AF\_STM8S, [380](#)  
\_TIM3\_IER  
STM8AF\_STM8S, [380](#)  
\_TIM3\_IER\_RESET\_VALUE  
STM8AF\_STM8S, [380](#)  
\_TIM3\_OC1M  
STM8AF\_STM8S, [381](#)  
\_TIM3\_OC1M0  
STM8AF\_STM8S, [381](#)  
\_TIM3\_OC1M1  
STM8AF\_STM8S, [381](#)  
\_TIM3\_OC1M2  
STM8AF\_STM8S, [381](#)  
\_TIM3\_OC1PE  
STM8AF\_STM8S, [381](#)  
\_TIM3\_OC2M  
STM8AF\_STM8S, [382](#)  
\_TIM3\_OC2M0  
STM8AF\_STM8S, [382](#)  
\_TIM3\_OC2M1  
STM8AF\_STM8S, [382](#)  
\_TIM3\_OC2M2  
STM8AF\_STM8S, [382](#)  
\_TIM3\_OC2PE  
STM8AF\_STM8S, [382](#)  
\_TIM3\_OPM  
STM8AF\_STM8S, [383](#)  
\_TIM3\_PSC  
STM8AF\_STM8S, [383](#)  
\_TIM3\_PSC0  
STM8AF\_STM8S, [383](#)  
\_TIM3\_PSC1  
STM8AF\_STM8S, [383](#)  
\_TIM3\_PSC2  
STM8AF\_STM8S, [383](#)  
\_TIM3\_PSC3  
STM8AF\_STM8S, [384](#)  
\_TIM3\_PSCR  
STM8AF\_STM8S, [384](#)  
\_TIM3\_PSCR\_RESET\_VALUE  
STM8AF\_STM8S, [384](#)  
\_TIM3\_SR1  
STM8AF\_STM8S, [384](#)  
\_TIM3\_SR1\_RESET\_VALUE  
STM8AF\_STM8S, [384](#)  
\_TIM3\_SR2  
STM8AF\_STM8S, [385](#)  
\_TIM3\_SR2\_RESET\_VALUE  
STM8AF\_STM8S, [385](#)  
\_TIM3\_UG  
STM8AF\_STM8S, [385](#)  
\_TIM3\_UDIS  
STM8AF\_STM8S, [385](#)  
\_TIM3\_UIE  
STM8AF\_STM8S, [385](#)  
\_TIM3\_UIF

- STM8AF\_STM8S, [386](#)
- \_TIM3\_URS
  - STM8AF\_STM8S, [386](#)
- \_TIM4
  - STM8AF\_STM8S, [386](#)
  - \_TIM4\_ARR
    - STM8AF\_STM8S, [386](#)
  - \_TIM4\_ARPE
    - STM8AF\_STM8S, [386](#)
  - \_TIM4\_ARR\_RESET\_VALUE
    - STM8AF\_STM8S, [387](#)
  - \_TIM4\_CR
    - STM8AF\_STM8S, [387](#)
  - \_TIM4\_CEN
    - STM8AF\_STM8S, [387](#)
  - \_TIM4\_CNTR
    - STM8AF\_STM8S, [387](#)
  - \_TIM4\_CNTR\_RESET\_VALUE
    - STM8AF\_STM8S, [387](#)
  - \_TIM4\_CR\_RESET\_VALUE
    - STM8AF\_STM8S, [388](#)
  - \_TIM4\_EGR
    - STM8AF\_STM8S, [388](#)
  - \_TIM4\_EGR\_RESET\_VALUE
    - STM8AF\_STM8S, [388](#)
  - \_TIM4\_IER
    - STM8AF\_STM8S, [388](#)
  - \_TIM4\_IER\_RESET\_VALUE
    - STM8AF\_STM8S, [388](#)
  - \_TIM4\_OPM
    - STM8AF\_STM8S, [389](#)
  - \_TIM4\_PSC
    - STM8AF\_STM8S, [389](#)
  - \_TIM4\_PSC0
    - STM8AF\_STM8S, [389](#)
  - \_TIM4\_PSC1
    - STM8AF\_STM8S, [389](#)
  - \_TIM4\_PSC2
    - STM8AF\_STM8S, [389](#)
  - \_TIM4\_PSCR
    - STM8AF\_STM8S, [390](#)
  - \_TIM4\_PSCR\_RESET\_VALUE
    - STM8AF\_STM8S, [390](#)
  - \_TIM4\_SR
    - STM8AF\_STM8S, [390](#)
  - \_TIM4\_SR\_RESET\_VALUE
    - STM8AF\_STM8S, [390](#)
  - \_TIM4\_UG
    - STM8AF\_STM8S, [391](#)
  - \_TIM4\_UDIS
    - STM8AF\_STM8S, [390](#)
  - \_TIM4\_UIE
    - STM8AF\_STM8S, [391](#)
  - \_TIM4\_UIF
    - STM8AF\_STM8S, [391](#)
  - \_TIM4\_URS
    - STM8AF\_STM8S, [391](#)
  - \_TIM5
    - STM8AF\_STM8S, [391](#)
    - \_TIM5\_ARPE
      - STM8AF\_STM8S, [392](#)
    - \_TIM5\_ARRH
      - STM8AF\_STM8S, [392](#)
    - \_TIM5\_ARRL
      - STM8AF\_STM8S, [392](#)
    - \_TIM5\_ARRH\_RESET\_VALUE
      - STM8AF\_STM8S, [392](#)
    - \_TIM5\_ARRL\_RESET\_VALUE
      - STM8AF\_STM8S, [392](#)
    - \_TIM5\_CC1E
      - STM8AF\_STM8S, [393](#)
    - \_TIM5\_CC1G
      - STM8AF\_STM8S, [393](#)
    - \_TIM5\_CC1P
      - STM8AF\_STM8S, [394](#)
    - \_TIM5\_CC1S
      - STM8AF\_STM8S, [394](#)
    - \_TIM5\_CC1IE
      - STM8AF\_STM8S, [393](#)
    - \_TIM5\_CC1IF
      - STM8AF\_STM8S, [393](#)
    - \_TIM5\_CC1OF
      - STM8AF\_STM8S, [393](#)
    - \_TIM5\_CC1S0
      - STM8AF\_STM8S, [394](#)
    - \_TIM5\_CC1S1
      - STM8AF\_STM8S, [394](#)
    - \_TIM5\_CC2E
      - STM8AF\_STM8S, [394](#)
    - \_TIM5\_CC2G
      - STM8AF\_STM8S, [395](#)
    - \_TIM5\_CC2P
      - STM8AF\_STM8S, [395](#)
    - \_TIM5\_CC2S
      - STM8AF\_STM8S, [396](#)
    - \_TIM5\_CC2IE
      - STM8AF\_STM8S, [395](#)
    - \_TIM5\_CC2IF
      - STM8AF\_STM8S, [395](#)
    - \_TIM5\_CC2OF
      - STM8AF\_STM8S, [395](#)
    - \_TIM5\_CC2S0
      - STM8AF\_STM8S, [396](#)
    - \_TIM5\_CC2S1
      - STM8AF\_STM8S, [396](#)
    - \_TIM5\_CC3E
      - STM8AF\_STM8S, [396](#)
    - \_TIM5\_CC3G
      - STM8AF\_STM8S, [396](#)
    - \_TIM5\_CC3P
      - STM8AF\_STM8S, [397](#)
    - \_TIM5\_CC3S
      - STM8AF\_STM8S, [397](#)
    - \_TIM5\_CC3IE
      - STM8AF\_STM8S, [397](#)
    - \_TIM5\_CC3IF

STM8AF\_STM8S, [397](#)  
\_TIM5\_CC3OF  
STM8AF\_STM8S, [397](#)  
\_TIM5\_CC3S0  
STM8AF\_STM8S, [398](#)  
\_TIM5\_CC3S1  
STM8AF\_STM8S, [398](#)  
\_TIM5\_CCER1  
STM8AF\_STM8S, [398](#)  
\_TIM5\_CCER1\_RESET\_VALUE  
STM8AF\_STM8S, [398](#)  
\_TIM5\_CCER2  
STM8AF\_STM8S, [398](#)  
\_TIM5\_CCER2\_RESET\_VALUE  
STM8AF\_STM8S, [399](#)  
\_TIM5\_CCMR1  
STM8AF\_STM8S, [399](#)  
\_TIM5\_CCMR1\_RESET\_VALUE  
STM8AF\_STM8S, [399](#)  
\_TIM5\_CCMR2  
STM8AF\_STM8S, [399](#)  
\_TIM5\_CCMR2\_RESET\_VALUE  
STM8AF\_STM8S, [399](#)  
\_TIM5\_CCMR3  
STM8AF\_STM8S, [400](#)  
\_TIM5\_CCMR3\_RESET\_VALUE  
STM8AF\_STM8S, [400](#)  
\_TIM5\_CCPC  
STM8AF\_STM8S, [400](#)  
\_TIM5\_CCR1H  
STM8AF\_STM8S, [400](#)  
\_TIM5\_CCR1L  
STM8AF\_STM8S, [401](#)  
\_TIM5\_CCR1H\_RESET\_VALUE  
STM8AF\_STM8S, [400](#)  
\_TIM5\_CCR1L\_RESET\_VALUE  
STM8AF\_STM8S, [401](#)  
\_TIM5\_CCR2H  
STM8AF\_STM8S, [401](#)  
\_TIM5\_CCR2L  
STM8AF\_STM8S, [401](#)  
\_TIM5\_CCR2H\_RESET\_VALUE  
STM8AF\_STM8S, [401](#)  
\_TIM5\_CCR2L\_RESET\_VALUE  
STM8AF\_STM8S, [402](#)  
\_TIM5\_CCR3H  
STM8AF\_STM8S, [402](#)  
\_TIM5\_CCR3L  
STM8AF\_STM8S, [402](#)  
\_TIM5\_CCR3H\_RESET\_VALUE  
STM8AF\_STM8S, [402](#)  
\_TIM5\_CCR3L\_RESET\_VALUE  
STM8AF\_STM8S, [402](#)  
\_TIM5\_CEN  
STM8AF\_STM8S, [403](#)  
\_TIM5\_CNTRH  
STM8AF\_STM8S, [403](#)  
\_TIM5\_CNTRL  
STM8AF\_STM8S, [403](#)  
\_TIM5\_CNTRH\_RESET\_VALUE  
STM8AF\_STM8S, [403](#)  
\_TIM5\_CNTRL\_RESET\_VALUE  
STM8AF\_STM8S, [403](#)  
\_TIM5\_COMS  
STM8AF\_STM8S, [404](#)  
\_TIM5\_CR1  
STM8AF\_STM8S, [404](#)  
\_TIM5\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [404](#)  
\_TIM5\_CR2  
STM8AF\_STM8S, [404](#)  
\_TIM5\_CR2\_RESET\_VALUE  
STM8AF\_STM8S, [404](#)  
\_TIM5\_EGR  
STM8AF\_STM8S, [405](#)  
\_TIM5\_EGR\_RESET\_VALUE  
STM8AF\_STM8S, [405](#)  
\_TIM5\_IC1F  
STM8AF\_STM8S, [405](#)  
\_TIM5\_IC1F0  
STM8AF\_STM8S, [405](#)  
\_TIM5\_IC1F1  
STM8AF\_STM8S, [405](#)  
\_TIM5\_IC1F2  
STM8AF\_STM8S, [406](#)  
\_TIM5\_IC1F3  
STM8AF\_STM8S, [406](#)  
\_TIM5\_IC1PSC  
STM8AF\_STM8S, [406](#)  
\_TIM5\_IC1PSC0  
STM8AF\_STM8S, [406](#)  
\_TIM5\_IC1PSC1  
STM8AF\_STM8S, [406](#)  
\_TIM5\_IC2F  
STM8AF\_STM8S, [407](#)  
\_TIM5\_IC2F0  
STM8AF\_STM8S, [407](#)  
\_TIM5\_IC2F1  
STM8AF\_STM8S, [407](#)  
\_TIM5\_IC2F2  
STM8AF\_STM8S, [407](#)  
\_TIM5\_IC2F3  
STM8AF\_STM8S, [407](#)  
\_TIM5\_IC2PSC  
STM8AF\_STM8S, [408](#)  
\_TIM5\_IC2PSC0  
STM8AF\_STM8S, [408](#)  
\_TIM5\_IC2PSC1  
STM8AF\_STM8S, [408](#)  
\_TIM5\_IC3F  
STM8AF\_STM8S, [408](#)  
\_TIM5\_IC3F0  
STM8AF\_STM8S, [408](#)  
\_TIM5\_IC3F1  
STM8AF\_STM8S, [409](#)  
\_TIM5\_IC3F2

STM8AF\_STM8S, [409](#)  
\_TIM5\_IC3F3  
STM8AF\_STM8S, [409](#)  
\_TIM5\_IC3PSC  
STM8AF\_STM8S, [409](#)  
\_TIM5\_IC3PSC0  
STM8AF\_STM8S, [409](#)  
\_TIM5\_IC3PSC1  
STM8AF\_STM8S, [410](#)  
\_TIM5\_IER  
STM8AF\_STM8S, [410](#)  
\_TIM5\_IER\_RESET\_VALUE  
STM8AF\_STM8S, [410](#)  
\_TIM5\_MMS  
STM8AF\_STM8S, [410](#)  
\_TIM5\_MMS0  
STM8AF\_STM8S, [410](#)  
\_TIM5\_MMS1  
STM8AF\_STM8S, [411](#)  
\_TIM5\_MMS2  
STM8AF\_STM8S, [411](#)  
\_TIM5\_MSM  
STM8AF\_STM8S, [411](#)  
\_TIM5\_OC1M  
STM8AF\_STM8S, [411](#)  
\_TIM5\_OC1M0  
STM8AF\_STM8S, [411](#)  
\_TIM5\_OC1M1  
STM8AF\_STM8S, [412](#)  
\_TIM5\_OC1M2  
STM8AF\_STM8S, [412](#)  
\_TIM5\_OC1PE  
STM8AF\_STM8S, [412](#)  
\_TIM5\_OC2M  
STM8AF\_STM8S, [412](#)  
\_TIM5\_OC2M0  
STM8AF\_STM8S, [412](#)  
\_TIM5\_OC2M1  
STM8AF\_STM8S, [413](#)  
\_TIM5\_OC2M2  
STM8AF\_STM8S, [413](#)  
\_TIM5\_OC2PE  
STM8AF\_STM8S, [413](#)  
\_TIM5\_OC3M  
STM8AF\_STM8S, [413](#)  
\_TIM5\_OC3M0  
STM8AF\_STM8S, [413](#)  
\_TIM5\_OC3M1  
STM8AF\_STM8S, [414](#)  
\_TIM5\_OC3M2  
STM8AF\_STM8S, [414](#)  
\_TIM5\_OC3PE  
STM8AF\_STM8S, [414](#)  
\_TIM5\_OPM  
STM8AF\_STM8S, [414](#)  
\_TIM5\_PSC  
STM8AF\_STM8S, [414](#)  
\_TIM5\_PSC0  
STM8AF\_STM8S, [415](#)  
\_TIM5\_PSC1  
STM8AF\_STM8S, [415](#)  
\_TIM5\_PSC2  
STM8AF\_STM8S, [415](#)  
\_TIM5\_PSC3  
STM8AF\_STM8S, [415](#)  
\_TIM5\_PSCR  
STM8AF\_STM8S, [415](#)  
\_TIM5\_PSCR\_RESET\_VALUE  
STM8AF\_STM8S, [416](#)  
\_TIM5\_SMS  
STM8AF\_STM8S, [416](#)  
\_TIM5\_SMCR  
STM8AF\_STM8S, [416](#)  
\_TIM5\_SMCR\_RESET\_VALUE  
STM8AF\_STM8S, [416](#)  
\_TIM5\_SMS0  
STM8AF\_STM8S, [416](#)  
\_TIM5\_SMS1  
STM8AF\_STM8S, [417](#)  
\_TIM5\_SMS2  
STM8AF\_STM8S, [417](#)  
\_TIM5\_SR1  
STM8AF\_STM8S, [417](#)  
\_TIM5\_SR1\_RESET\_VALUE  
STM8AF\_STM8S, [417](#)  
\_TIM5\_SR2  
STM8AF\_STM8S, [417](#)  
\_TIM5\_SR2\_RESET\_VALUE  
STM8AF\_STM8S, [418](#)  
\_TIM5\_TG  
STM8AF\_STM8S, [418](#)  
\_TIM5\_TS  
STM8AF\_STM8S, [418](#)  
\_TIM5\_TIE  
STM8AF\_STM8S, [418](#)  
\_TIM5\_TIF  
STM8AF\_STM8S, [418](#)  
\_TIM5\_TS0  
STM8AF\_STM8S, [419](#)  
\_TIM5\_TS1  
STM8AF\_STM8S, [419](#)  
\_TIM5\_TS2  
STM8AF\_STM8S, [419](#)  
\_TIM5\_UG  
STM8AF\_STM8S, [419](#)  
\_TIM5\_UDIS  
STM8AF\_STM8S, [419](#)  
\_TIM5\_UIE  
STM8AF\_STM8S, [420](#)  
\_TIM5\_UIF  
STM8AF\_STM8S, [420](#)  
\_TIM5\_URS  
STM8AF\_STM8S, [420](#)  
\_TIM6  
STM8AF\_STM8S, [420](#)  
\_TIM6\_ARR

STM8AF\_STM8S, [421](#)  
 \_TIM6\_ARPE  
   STM8AF\_STM8S, [420](#)  
 \_TIM6\_ARR\_RESET\_VALUE  
   STM8AF\_STM8S, [421](#)  
 \_TIM6\_CR  
   STM8AF\_STM8S, [422](#)  
 \_TIM6\_CEN  
   STM8AF\_STM8S, [421](#)  
 \_TIM6\_CNTR  
   STM8AF\_STM8S, [421](#)  
 \_TIM6\_CNTR\_RESET\_VALUE  
   STM8AF\_STM8S, [421](#)  
 \_TIM6\_CR\_RESET\_VALUE  
   STM8AF\_STM8S, [422](#)  
 \_TIM6\_EGR  
   STM8AF\_STM8S, [422](#)  
 \_TIM6\_EGR\_RESET\_VALUE  
   STM8AF\_STM8S, [422](#)  
 \_TIM6\_IER  
   STM8AF\_STM8S, [422](#)  
 \_TIM6\_IER\_RESET\_VALUE  
   STM8AF\_STM8S, [423](#)  
 \_TIM6\_MMS  
   STM8AF\_STM8S, [423](#)  
 \_TIM6\_MMS0  
   STM8AF\_STM8S, [423](#)  
 \_TIM6\_MMS1  
   STM8AF\_STM8S, [423](#)  
 \_TIM6\_MMS2  
   STM8AF\_STM8S, [423](#)  
 \_TIM6\_OPM  
   STM8AF\_STM8S, [424](#)  
 \_TIM6\_PSC  
   STM8AF\_STM8S, [424](#)  
 \_TIM6\_PSC0  
   STM8AF\_STM8S, [424](#)  
 \_TIM6\_PSC1  
   STM8AF\_STM8S, [424](#)  
 \_TIM6\_PSC2  
   STM8AF\_STM8S, [424](#)  
 \_TIM6\_PSCR  
   STM8AF\_STM8S, [425](#)  
 \_TIM6\_PSCR\_RESET\_VALUE  
   STM8AF\_STM8S, [425](#)  
 \_TIM6\_SR  
   STM8AF\_STM8S, [426](#)  
 \_TIM6\_SMS  
   STM8AF\_STM8S, [425](#)  
 \_TIM6\_SMS0  
   STM8AF\_STM8S, [425](#)  
 \_TIM6\_SMS1  
   STM8AF\_STM8S, [425](#)  
 \_TIM6\_SMS2  
   STM8AF\_STM8S, [426](#)  
 \_TIM6\_SR\_RESET\_VALUE  
   STM8AF\_STM8S, [426](#)  
 \_TIM6\_TS  
   STM8AF\_STM8S, [426](#)  
 \_TIM6\_TS0  
   STM8AF\_STM8S, [426](#)  
 \_TIM6\_TS1  
   STM8AF\_STM8S, [427](#)  
 \_TIM6\_TS2  
   STM8AF\_STM8S, [427](#)  
 \_TIM6\_UG  
   STM8AF\_STM8S, [427](#)  
 \_TIM6\_UDIS  
   STM8AF\_STM8S, [427](#)  
 \_TIM6\_UIE  
   STM8AF\_STM8S, [427](#)  
 \_TIM6\_UIF  
   STM8AF\_STM8S, [428](#)  
 \_TIM6\_URS  
   STM8AF\_STM8S, [428](#)  
 \_UART1  
   STM8AF\_STM8S, [428](#)  
 \_UART1\_M  
   STM8AF\_STM8S, [435](#)  
 \_UART1\_ADD  
   STM8AF\_STM8S, [428](#)  
 \_UART1\_ADD0  
   STM8AF\_STM8S, [428](#)  
 \_UART1\_ADD1  
   STM8AF\_STM8S, [429](#)  
 \_UART1\_ADD2  
   STM8AF\_STM8S, [429](#)  
 \_UART1\_ADD3  
   STM8AF\_STM8S, [429](#)  
 \_UART1\_BRR1  
   STM8AF\_STM8S, [429](#)  
 \_UART1\_BRR1\_RESET\_VALUE  
   STM8AF\_STM8S, [429](#)  
 \_UART1\_BRR2  
   STM8AF\_STM8S, [430](#)  
 \_UART1\_BRR2\_RESET\_VALUE  
   STM8AF\_STM8S, [430](#)  
 \_UART1\_CKEN  
   STM8AF\_STM8S, [430](#)  
 \_UART1\_CPHA  
   STM8AF\_STM8S, [430](#)  
 \_UART1\_CPOL  
   STM8AF\_STM8S, [430](#)  
 \_UART1\_CR1  
   STM8AF\_STM8S, [431](#)  
 \_UART1\_CR1\_RESET\_VALUE  
   STM8AF\_STM8S, [431](#)  
 \_UART1\_CR2  
   STM8AF\_STM8S, [431](#)  
 \_UART1\_CR2\_RESET\_VALUE  
   STM8AF\_STM8S, [431](#)  
 \_UART1\_CR3  
   STM8AF\_STM8S, [431](#)  
 \_UART1\_CR3\_RESET\_VALUE  
   STM8AF\_STM8S, [432](#)  
 \_UART1\_CR4

STM8AF\_STM8S, [432](#)  
\_UART1\_CR4\_RESET\_VALUE  
STM8AF\_STM8S, [432](#)  
\_UART1\_CR5  
STM8AF\_STM8S, [432](#)  
\_UART1\_CR5\_RESET\_VALUE  
STM8AF\_STM8S, [432](#)  
\_UART1\_DR  
STM8AF\_STM8S, [433](#)  
\_UART1\_FE  
STM8AF\_STM8S, [433](#)  
\_UART1\_GTR  
STM8AF\_STM8S, [433](#)  
\_UART1\_GTR\_RESET\_VALUE  
STM8AF\_STM8S, [433](#)  
\_UART1\_HDSEL  
STM8AF\_STM8S, [433](#)  
\_UART1\_IDLE  
STM8AF\_STM8S, [434](#)  
\_UART1\_ILIEN  
STM8AF\_STM8S, [434](#)  
\_UART1\_IREN  
STM8AF\_STM8S, [434](#)  
\_UART1\_IRLP  
STM8AF\_STM8S, [434](#)  
\_UART1\_LBCL  
STM8AF\_STM8S, [434](#)  
\_UART1\_LBDF  
STM8AF\_STM8S, [435](#)  
\_UART1\_LBDL  
STM8AF\_STM8S, [435](#)  
\_UART1\_LBDIEN  
STM8AF\_STM8S, [435](#)  
\_UART1\_LINEN  
STM8AF\_STM8S, [435](#)  
\_UART1\_NF  
STM8AF\_STM8S, [436](#)  
\_UART1\_NACK  
STM8AF\_STM8S, [436](#)  
\_UART1\_OR\_LHE  
STM8AF\_STM8S, [436](#)  
\_UART1\_PE  
STM8AF\_STM8S, [436](#)  
\_UART1\_PS  
STM8AF\_STM8S, [437](#)  
\_UART1\_PCEN  
STM8AF\_STM8S, [436](#)  
\_UART1\_PIEN  
STM8AF\_STM8S, [437](#)  
\_UART1\_PSCR  
STM8AF\_STM8S, [437](#)  
\_UART1\_PSCR\_RESET\_VALUE  
STM8AF\_STM8S, [437](#)  
\_UART1\_R8  
STM8AF\_STM8S, [437](#)  
\_UART1\_REN  
STM8AF\_STM8S, [438](#)  
\_UART1\_RIEN  
STM8AF\_STM8S, [438](#)  
\_UART1\_RWU  
STM8AF\_STM8S, [438](#)  
\_UART1\_RXNE  
STM8AF\_STM8S, [438](#)  
\_UART1\_SR  
STM8AF\_STM8S, [439](#)  
\_UART1\_SBK  
STM8AF\_STM8S, [438](#)  
\_UART1\_SCEN  
STM8AF\_STM8S, [439](#)  
\_UART1\_SR\_RESET\_VALUE  
STM8AF\_STM8S, [439](#)  
\_UART1\_STOP  
STM8AF\_STM8S, [439](#)  
\_UART1\_STOP0  
STM8AF\_STM8S, [439](#)  
\_UART1\_STOP1  
STM8AF\_STM8S, [440](#)  
\_UART1\_T8  
STM8AF\_STM8S, [440](#)  
\_UART1\_TC  
STM8AF\_STM8S, [440](#)  
\_UART1\_TCIEN  
STM8AF\_STM8S, [440](#)  
\_UART1\_TEN  
STM8AF\_STM8S, [440](#)  
\_UART1\_TIEN  
STM8AF\_STM8S, [441](#)  
\_UART1\_TXE  
STM8AF\_STM8S, [441](#)  
\_UART1\_UARTD  
STM8AF\_STM8S, [441](#)  
\_UART1\_WAKE  
STM8AF\_STM8S, [441](#)  
\_UART2  
STM8AF\_STM8S, [441](#)  
\_UART2\_M  
STM8AF\_STM8S, [450](#)  
\_UART2\_ADD  
STM8AF\_STM8S, [442](#)  
\_UART2\_ADD0  
STM8AF\_STM8S, [442](#)  
\_UART2\_ADD1  
STM8AF\_STM8S, [442](#)  
\_UART2\_ADD2  
STM8AF\_STM8S, [442](#)  
\_UART2\_ADD3  
STM8AF\_STM8S, [442](#)  
\_UART2\_BRR1  
STM8AF\_STM8S, [443](#)  
\_UART2\_BRR1\_RESET\_VALUE  
STM8AF\_STM8S, [443](#)  
\_UART2\_BRR2  
STM8AF\_STM8S, [443](#)  
\_UART2\_BRR2\_RESET\_VALUE  
STM8AF\_STM8S, [443](#)  
\_UART2\_CKEN

STM8AF\_STM8S, [443](#)  
\_UART2\_CPHA  
STM8AF\_STM8S, [444](#)  
\_UART2\_CPOL  
STM8AF\_STM8S, [444](#)  
\_UART2\_CR1  
STM8AF\_STM8S, [444](#)  
\_UART2\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [444](#)  
\_UART2\_CR2  
STM8AF\_STM8S, [444](#)  
\_UART2\_CR2\_RESET\_VALUE  
STM8AF\_STM8S, [445](#)  
\_UART2\_CR3  
STM8AF\_STM8S, [445](#)  
\_UART2\_CR3\_RESET\_VALUE  
STM8AF\_STM8S, [445](#)  
\_UART2\_CR4  
STM8AF\_STM8S, [445](#)  
\_UART2\_CR4\_RESET\_VALUE  
STM8AF\_STM8S, [445](#)  
\_UART2\_CR5  
STM8AF\_STM8S, [446](#)  
\_UART2\_CR5\_RESET\_VALUE  
STM8AF\_STM8S, [446](#)  
\_UART2\_CR6  
STM8AF\_STM8S, [446](#)  
\_UART2\_CR6\_RESET\_VALUE  
STM8AF\_STM8S, [446](#)  
\_UART2\_DR  
STM8AF\_STM8S, [446](#)  
\_UART2\_FE  
STM8AF\_STM8S, [447](#)  
\_UART2\_GTR  
STM8AF\_STM8S, [447](#)  
\_UART2\_GTR\_RESET\_VALUE  
STM8AF\_STM8S, [447](#)  
\_UART2\_IDLE  
STM8AF\_STM8S, [447](#)  
\_UART2\_ILIEN  
STM8AF\_STM8S, [447](#)  
\_UART2\_IREN  
STM8AF\_STM8S, [448](#)  
\_UART2\_IRLP  
STM8AF\_STM8S, [448](#)  
\_UART2\_LASE  
STM8AF\_STM8S, [448](#)  
\_UART2\_LBCL  
STM8AF\_STM8S, [448](#)  
\_UART2\_LBDF  
STM8AF\_STM8S, [448](#)  
\_UART2\_LBDL  
STM8AF\_STM8S, [449](#)  
\_UART2\_LBDIEN  
STM8AF\_STM8S, [449](#)  
\_UART2\_LDUM  
STM8AF\_STM8S, [449](#)  
\_UART2\_LHDF  
STM8AF\_STM8S, [449](#)  
\_UART2\_LHDIEN  
STM8AF\_STM8S, [449](#)  
\_UART2\_LINEN  
STM8AF\_STM8S, [450](#)  
\_UART2\_LSF  
STM8AF\_STM8S, [450](#)  
\_UART2\_LSLV  
STM8AF\_STM8S, [450](#)  
\_UART2\_NF  
STM8AF\_STM8S, [451](#)  
\_UART2\_NACK  
STM8AF\_STM8S, [450](#)  
\_UART2\_OR\_LHE  
STM8AF\_STM8S, [451](#)  
\_UART2\_PE  
STM8AF\_STM8S, [451](#)  
\_UART2\_PS  
STM8AF\_STM8S, [452](#)  
\_UART2\_PCEN  
STM8AF\_STM8S, [451](#)  
\_UART2\_PIEN  
STM8AF\_STM8S, [451](#)  
\_UART2\_PSCR  
STM8AF\_STM8S, [452](#)  
\_UART2\_PSCR\_RESET\_VALUE  
STM8AF\_STM8S, [452](#)  
\_UART2\_R8  
STM8AF\_STM8S, [452](#)  
\_UART2\_REN  
STM8AF\_STM8S, [452](#)  
\_UART2\_RIEN  
STM8AF\_STM8S, [453](#)  
\_UART2\_RWU  
STM8AF\_STM8S, [453](#)  
\_UART2\_RXNE  
STM8AF\_STM8S, [453](#)  
\_UART2\_SR  
STM8AF\_STM8S, [454](#)  
\_UART2\_SBK  
STM8AF\_STM8S, [453](#)  
\_UART2\_SCEN  
STM8AF\_STM8S, [453](#)  
\_UART2\_SR\_RESET\_VALUE  
STM8AF\_STM8S, [454](#)  
\_UART2\_STOP  
STM8AF\_STM8S, [454](#)  
\_UART2\_STOP0  
STM8AF\_STM8S, [454](#)  
\_UART2\_STOP1  
STM8AF\_STM8S, [454](#)  
\_UART2\_T8  
STM8AF\_STM8S, [455](#)  
\_UART2\_TC  
STM8AF\_STM8S, [455](#)  
\_UART2\_TCIEN  
STM8AF\_STM8S, [455](#)  
\_UART2\_TEN



- STM8AF\_STM8S, [455](#)
- \_UART2\_TIEN
  - STM8AF\_STM8S, [455](#)
- \_UART2\_TXE
  - STM8AF\_STM8S, [456](#)
- \_UART2\_UARTD
  - STM8AF\_STM8S, [456](#)
- \_UART2\_WAKE
  - STM8AF\_STM8S, [456](#)
- \_UART3
  - STM8AF\_STM8S, [456](#)
- \_UART3\_M
  - STM8AF\_STM8S, [463](#)
- \_UART3\_ADD
  - STM8AF\_STM8S, [456](#)
- \_UART3\_ADD0
  - STM8AF\_STM8S, [457](#)
- \_UART3\_ADD1
  - STM8AF\_STM8S, [457](#)
- \_UART3\_ADD2
  - STM8AF\_STM8S, [457](#)
- \_UART3\_ADD3
  - STM8AF\_STM8S, [457](#)
- \_UART3\_BRR1
  - STM8AF\_STM8S, [457](#)
- \_UART3\_BRR1\_RESET\_VALUE
  - STM8AF\_STM8S, [458](#)
- \_UART3\_BRR2
  - STM8AF\_STM8S, [458](#)
- \_UART3\_BRR2\_RESET\_VALUE
  - STM8AF\_STM8S, [458](#)
- \_UART3\_CR1
  - STM8AF\_STM8S, [458](#)
- \_UART3\_CR1\_RESET\_VALUE
  - STM8AF\_STM8S, [458](#)
- \_UART3\_CR2
  - STM8AF\_STM8S, [459](#)
- \_UART3\_CR2\_RESET\_VALUE
  - STM8AF\_STM8S, [459](#)
- \_UART3\_CR3
  - STM8AF\_STM8S, [459](#)
- \_UART3\_CR3\_RESET\_VALUE
  - STM8AF\_STM8S, [459](#)
- \_UART3\_CR4
  - STM8AF\_STM8S, [459](#)
- \_UART3\_CR4\_RESET\_VALUE
  - STM8AF\_STM8S, [460](#)
- \_UART3\_CR6
  - STM8AF\_STM8S, [460](#)
- \_UART3\_CR6\_RESET\_VALUE
  - STM8AF\_STM8S, [460](#)
- \_UART3\_DR
  - STM8AF\_STM8S, [460](#)
- \_UART3\_FE
  - STM8AF\_STM8S, [460](#)
- \_UART3\_IDLE
  - STM8AF\_STM8S, [461](#)
- \_UART3\_ILIEN
  - STM8AF\_STM8S, [461](#)
- \_UART3\_LASE
  - STM8AF\_STM8S, [461](#)
- \_UART3\_LBDF
  - STM8AF\_STM8S, [461](#)
- \_UART3\_LBDL
  - STM8AF\_STM8S, [462](#)
- \_UART3\_LBDIEN
  - STM8AF\_STM8S, [461](#)
- \_UART3\_LDUM
  - STM8AF\_STM8S, [462](#)
- \_UART3\_LHDF
  - STM8AF\_STM8S, [462](#)
- \_UART3\_LHDIEN
  - STM8AF\_STM8S, [462](#)
- \_UART3\_LINEN
  - STM8AF\_STM8S, [462](#)
- \_UART3\_LSF
  - STM8AF\_STM8S, [463](#)
- \_UART3\_LSLV
  - STM8AF\_STM8S, [463](#)
- \_UART3\_NF
  - STM8AF\_STM8S, [463](#)
- \_UART3\_OR\_LHE
  - STM8AF\_STM8S, [463](#)
- \_UART3\_PE
  - STM8AF\_STM8S, [464](#)
- \_UART3\_PS
  - STM8AF\_STM8S, [464](#)
- \_UART3\_PCEN
  - STM8AF\_STM8S, [464](#)
- \_UART3\_PIEN
  - STM8AF\_STM8S, [464](#)
- \_UART3\_R8
  - STM8AF\_STM8S, [464](#)
- \_UART3\_REN
  - STM8AF\_STM8S, [465](#)
- \_UART3\_RIEN
  - STM8AF\_STM8S, [465](#)
- \_UART3\_RWU
  - STM8AF\_STM8S, [465](#)
- \_UART3\_RXNE
  - STM8AF\_STM8S, [465](#)
- \_UART3\_SR
  - STM8AF\_STM8S, [466](#)
- \_UART3\_SBK
  - STM8AF\_STM8S, [465](#)
- \_UART3\_SR\_RESET\_VALUE
  - STM8AF\_STM8S, [466](#)
- \_UART3\_STOP
  - STM8AF\_STM8S, [466](#)
- \_UART3\_STOP0
  - STM8AF\_STM8S, [466](#)
- \_UART3\_STOP1
  - STM8AF\_STM8S, [466](#)
- \_UART3\_T8
  - STM8AF\_STM8S, [467](#)
- \_UART3\_TC

STM8AF\_STM8S, [467](#)  
\_UART3\_TCIEN  
STM8AF\_STM8S, [467](#)  
\_UART3\_TEN  
STM8AF\_STM8S, [467](#)  
\_UART3\_TIEN  
STM8AF\_STM8S, [467](#)  
\_UART3\_TXE  
STM8AF\_STM8S, [468](#)  
\_UART3\_UARTD  
STM8AF\_STM8S, [468](#)  
\_UART3\_WAKE  
STM8AF\_STM8S, [468](#)  
\_UART4  
STM8AF\_STM8S, [468](#)  
\_UART4\_M  
STM8AF\_STM8S, [477](#)  
\_UART4\_ADD  
STM8AF\_STM8S, [468](#)  
\_UART4\_ADD0  
STM8AF\_STM8S, [469](#)  
\_UART4\_ADD1  
STM8AF\_STM8S, [469](#)  
\_UART4\_ADD2  
STM8AF\_STM8S, [469](#)  
\_UART4\_ADD3  
STM8AF\_STM8S, [469](#)  
\_UART4\_BRR1  
STM8AF\_STM8S, [469](#)  
\_UART4\_BRR1\_RESET\_VALUE  
STM8AF\_STM8S, [470](#)  
\_UART4\_BRR2  
STM8AF\_STM8S, [470](#)  
\_UART4\_BRR2\_RESET\_VALUE  
STM8AF\_STM8S, [470](#)  
\_UART4\_CKEN  
STM8AF\_STM8S, [470](#)  
\_UART4\_CPHA  
STM8AF\_STM8S, [470](#)  
\_UART4\_CPOL  
STM8AF\_STM8S, [471](#)  
\_UART4\_CR1  
STM8AF\_STM8S, [471](#)  
\_UART4\_CR1\_RESET\_VALUE  
STM8AF\_STM8S, [471](#)  
\_UART4\_CR2  
STM8AF\_STM8S, [471](#)  
\_UART4\_CR2\_RESET\_VALUE  
STM8AF\_STM8S, [471](#)  
\_UART4\_CR3  
STM8AF\_STM8S, [472](#)  
\_UART4\_CR3\_RESET\_VALUE  
STM8AF\_STM8S, [472](#)  
\_UART4\_CR4  
STM8AF\_STM8S, [472](#)  
\_UART4\_CR4\_RESET\_VALUE  
STM8AF\_STM8S, [472](#)  
\_UART4\_CR5  
STM8AF\_STM8S, [472](#)  
\_UART4\_CR5\_RESET\_VALUE  
STM8AF\_STM8S, [473](#)  
\_UART4\_CR6  
STM8AF\_STM8S, [473](#)  
\_UART4\_CR6\_RESET\_VALUE  
STM8AF\_STM8S, [473](#)  
\_UART4\_DR  
STM8AF\_STM8S, [473](#)  
\_UART4\_FE  
STM8AF\_STM8S, [473](#)  
\_UART4\_GTR  
STM8AF\_STM8S, [474](#)  
\_UART4\_GTR\_RESET\_VALUE  
STM8AF\_STM8S, [474](#)  
\_UART4\_HDSEL  
STM8AF\_STM8S, [474](#)  
\_UART4\_IDLE  
STM8AF\_STM8S, [474](#)  
\_UART4\_ILIEN  
STM8AF\_STM8S, [474](#)  
\_UART4\_IREN  
STM8AF\_STM8S, [475](#)  
\_UART4\_IRLP  
STM8AF\_STM8S, [475](#)  
\_UART4\_LASE  
STM8AF\_STM8S, [475](#)  
\_UART4\_LBCL  
STM8AF\_STM8S, [475](#)  
\_UART4\_LBDF  
STM8AF\_STM8S, [475](#)  
\_UART4\_LBDL  
STM8AF\_STM8S, [476](#)  
\_UART4\_LBDIEN  
STM8AF\_STM8S, [476](#)  
\_UART4\_LDUM  
STM8AF\_STM8S, [476](#)  
\_UART4\_LHDF  
STM8AF\_STM8S, [476](#)  
\_UART4\_LHDIEN  
STM8AF\_STM8S, [476](#)  
\_UART4\_LINEN  
STM8AF\_STM8S, [477](#)  
\_UART4\_LSF  
STM8AF\_STM8S, [477](#)  
\_UART4\_LSLV  
STM8AF\_STM8S, [477](#)  
\_UART4\_NF  
STM8AF\_STM8S, [478](#)  
\_UART4\_NACK  
STM8AF\_STM8S, [477](#)  
\_UART4\_OR\_LHE  
STM8AF\_STM8S, [478](#)  
\_UART4\_PE  
STM8AF\_STM8S, [478](#)  
\_UART4\_PS  
STM8AF\_STM8S, [479](#)  
\_UART4\_PCEN

- STM8AF\_STM8S, [478](#)
- \_UART4\_PIEN
  - STM8AF\_STM8S, [478](#)
- \_UART4\_PSCR
  - STM8AF\_STM8S, [479](#)
- \_UART4\_PSCR\_RESET\_VALUE
  - STM8AF\_STM8S, [479](#)
- \_UART4\_R8
  - STM8AF\_STM8S, [479](#)
- \_UART4\_REN
  - STM8AF\_STM8S, [479](#)
- \_UART4\_RIEN
  - STM8AF\_STM8S, [480](#)
- \_UART4\_RWU
  - STM8AF\_STM8S, [480](#)
- \_UART4\_RXNE
  - STM8AF\_STM8S, [480](#)
- \_UART4\_SR
  - STM8AF\_STM8S, [481](#)
- \_UART4\_SBK
  - STM8AF\_STM8S, [480](#)
- \_UART4\_SCEN
  - STM8AF\_STM8S, [480](#)
- \_UART4\_SR\_RESET\_VALUE
  - STM8AF\_STM8S, [481](#)
- \_UART4\_STOP
  - STM8AF\_STM8S, [481](#)
- \_UART4\_STOP0
  - STM8AF\_STM8S, [481](#)
- \_UART4\_STOP1
  - STM8AF\_STM8S, [481](#)
- \_UART4\_T8
  - STM8AF\_STM8S, [482](#)
- \_UART4\_TC
  - STM8AF\_STM8S, [482](#)
- \_UART4\_TCIEN
  - STM8AF\_STM8S, [482](#)
- \_UART4\_TEN
  - STM8AF\_STM8S, [482](#)
- \_UART4\_TIEN
  - STM8AF\_STM8S, [482](#)
- \_UART4\_TXE
  - STM8AF\_STM8S, [483](#)
- \_UART4\_UARTD
  - STM8AF\_STM8S, [483](#)
- \_UART4\_WAKE
  - STM8AF\_STM8S, [483](#)
- \_WWDG
  - STM8AF\_STM8S, [483](#)
- \_WWDG\_T
  - STM8AF\_STM8S, [484](#)
- \_WWDG\_W
  - STM8AF\_STM8S, [485](#)
- \_WWDG\_CR
  - STM8AF\_STM8S, [483](#)
- \_WWDG\_CR\_RESET\_VALUE
  - STM8AF\_STM8S, [484](#)
- \_WWDG\_T0
  - STM8AF\_STM8S, [484](#)
- \_WWDG\_T1
  - STM8AF\_STM8S, [484](#)
- \_WWDG\_T2
  - STM8AF\_STM8S, [485](#)
- \_WWDG\_T3
  - STM8AF\_STM8S, [485](#)
- \_WWDG\_T4
  - STM8AF\_STM8S, [485](#)
- \_WWDG\_T5
  - STM8AF\_STM8S, [485](#)
- \_WWDG\_T6
  - STM8AF\_STM8S, [485](#)
- \_WWDG\_W0
  - STM8AF\_STM8S, [486](#)
- \_WWDG\_W1
  - STM8AF\_STM8S, [486](#)
- \_WWDG\_W2
  - STM8AF\_STM8S, [486](#)
- \_WWDG\_W3
  - STM8AF\_STM8S, [486](#)
- \_WWDG\_W4
  - STM8AF\_STM8S, [486](#)
- \_WWDG\_W5
  - STM8AF\_STM8S, [487](#)
- \_WWDG\_W6
  - STM8AF\_STM8S, [487](#)
- \_WWDG\_WR
  - STM8AF\_STM8S, [487](#)
- \_WWDG\_WDGA
  - STM8AF\_STM8S, [487](#)
- \_WWDG\_WR\_RESET\_VALUE
  - STM8AF\_STM8S, [487](#)
- \_\_ADC1\_VECTOR\_\_
  - STM8AF\_STM8S, [125](#)
- \_\_AWU\_VECTOR\_\_
  - STM8AF\_STM8S, [126](#)
- \_\_CAN\_RX\_VECTOR\_\_
  - STM8AF\_STM8S, [126](#)
- \_\_CAN\_TX\_VECTOR\_\_
  - STM8AF\_STM8S, [126](#)
- \_\_CLK\_VECTOR\_\_
  - STM8AF\_STM8S, [126](#)
- \_\_FLASH\_VECTOR\_\_
  - STM8AF\_STM8S, [126](#)
- \_\_I2C\_VECTOR\_\_
  - STM8AF\_STM8S, [127](#)
- \_\_PORTA\_VECTOR\_\_
  - STM8AF\_STM8S, [127](#)
- \_\_PORTB\_VECTOR\_\_
  - STM8AF\_STM8S, [127](#)
- \_\_PORTC\_VECTOR\_\_
  - STM8AF\_STM8S, [127](#)
- \_\_PORTD\_VECTOR\_\_
  - STM8AF\_STM8S, [127](#)
- \_\_PORTE\_VECTOR\_\_
  - STM8AF\_STM8S, [127](#)
- \_\_PORTF\_VECTOR\_\_

- STM8AF\_STM8S, [128](#)
- \_\_SPI\_VECTOR\_\_
  - STM8AF\_STM8S, [128](#)
- \_\_TIM1\_CAPCOM\_VECTOR\_\_
  - STM8AF\_STM8S, [128](#)
- \_\_TIM1\_UPD\_OVF\_VECTOR\_\_
  - STM8AF\_STM8S, [128](#)
- \_\_TIM2\_CAPCOM\_VECTOR\_\_
  - STM8AF\_STM8S, [128](#)
- \_\_TIM2\_UPD\_OVF\_VECTOR\_\_
  - STM8AF\_STM8S, [128](#)
- \_\_TIM3\_CAPCOM\_VECTOR\_\_
  - STM8AF\_STM8S, [129](#)
- \_\_TIM3\_UPD\_OVF\_VECTOR\_\_
  - STM8AF\_STM8S, [129](#)
- \_\_TIM4\_UPD\_OVF\_VECTOR\_\_
  - STM8AF\_STM8S, [129](#)
- \_\_TIM5\_UPD\_OVF\_VECTOR\_\_
  - STM8AF\_STM8S, [129](#)
- \_\_TLI\_VECTOR\_\_
  - STM8AF\_STM8S, [129](#)
- \_\_UART1\_RXF\_VECTOR\_\_
  - STM8AF\_STM8S, [130](#)
- \_\_UART1\_TXE\_VECTOR\_\_
  - STM8AF\_STM8S, [130](#)
- \_\_UART2\_RXF\_VECTOR\_\_
  - STM8AF\_STM8S, [130](#)
- \_\_UART2\_TXE\_VECTOR\_\_
  - STM8AF\_STM8S, [130](#)
- ABOM
  - CAN\_t, [936](#)
- ABRQ
  - CAN\_t, [936](#)
- ACK
  - I2C\_t, [1012](#)
- ADC1\_AddressBase
  - STM8AF\_STM8S, [488–492](#)
- ADC1\_t, [891](#)
  - ADON, [896](#)
  - ALIGN, [896](#)
  - AWCRH, [897](#)
  - AWCRL, [897](#)
  - AWDIE, [897](#)
  - AWEN, [897](#)
  - AWSRH, [898](#)
  - AWSRL, [898](#)
  - AWD, [897](#)
  - AWS, [898](#)
  - CONT, [898](#)
  - CR1, [899](#)
  - CR2, [899](#)
  - CR3, [899](#)
  - CSR, [899](#)
  - CH, [898](#)
  - DATA, [899](#)
  - DB0RH, [900](#)
  - DB0RL, [900](#)
  - DB1RH, [901](#)
  - DB1RL, [901](#)
  - DB2RH, [901](#)
  - DB2RL, [901](#)
  - DB3RH, [901](#)
  - DB3RL, [901](#)
  - DB4RH, [902](#)
  - DB4RL, [902](#)
  - DB5RH, [902](#)
  - DB5RL, [902](#)
  - DB6RH, [902](#)
  - DB6RL, [902](#)
  - DB7RH, [903](#)
  - DB7RL, [903](#)
  - DB8RH, [903](#)
  - DB8RL, [903](#)
  - DB9RH, [903](#)
  - DB9RL, [903](#)
  - DBUF, [904](#)
  - DRH, [904](#)
  - DRL, [904](#)
  - EOCIE, [904](#)
  - EOC, [904](#)
  - EXTSEL, [904](#)
  - EXTTRIG, [905](#)
  - HTRH, [905](#)
  - HTRL, [905](#)
  - HT, [905](#)
  - LTRH, [906](#)
  - LTRL, [906](#)
  - LT, [905](#)
  - OVR, [906](#)
  - res, [906](#)
  - res2, [907](#)
  - res3, [907](#)
  - SCAN, [907](#)
  - SPSEL, [907](#)
  - TDRH, [908](#)
  - TDRL, [908](#)
  - TDH, [907](#)
  - TDL, [908](#)
- ADC2\_AddressBase
  - STM8AF\_STM8S, [492–499](#)
- ADC2\_t, [908](#)
  - ADON, [910](#)
  - ALIGN, [910](#)
  - CONT, [911](#)
  - CR1, [911](#)
  - CR2, [911](#)
  - CSR, [911](#)
  - CH, [910](#)
  - DATA, [911](#)
  - DRH, [912](#)
  - DRL, [912](#)
  - EOCIE, [912](#)
  - EOC, [912](#)
  - EXTSEL, [912](#)
  - EXTTRIG, [913](#)
  - res, [913](#)

- res2, [913](#)
- res3, [913](#)
- SPSEL, [914](#)
- TDRH, [914](#)
- TDRL, [914](#)
- TDH, [914](#)
- TDL, [914](#)
- ADD0
  - I2C\_t, [1012](#)
- ADD10
  - I2C\_t, [1012](#)
- ADDCONF
  - I2C\_t, [1012](#)
- ADDMODE
  - I2C\_t, [1013](#)
- ADDR
  - I2C\_t, [1013](#)
- ADON
  - ADC1\_t, [896](#)
  - ADC2\_t, [910](#)
- ADD
  - I2C\_t, [1012](#)
  - UART1\_t, [1160](#)
  - UART2\_t, [1175](#)
  - UART3\_t, [1190](#)
  - UART4\_t, [1204](#)
- AHALT
  - FLASH\_t, [1001](#)
- ALIGN
  - ADC1\_t, [896](#)
  - ADC2\_t, [910](#)
- ALST
  - CAN\_t, [936](#)
- AOE
  - TIM1\_t, [1059](#)
- APRE
  - AWU\_t, [916](#)
- APR
  - AWU\_t, [916](#)
- ARLO
  - I2C\_t, [1013](#)
- ARPE
  - TIM1\_t, [1059](#)
  - TIM2\_t, [1093](#)
  - TIM3\_t, [1112](#)
  - TIM4\_t, [1125](#)
  - TIM5\_t, [1134](#)
  - TIM6\_t, [1152](#)
- ARRH
  - TIM1\_t, [1059](#)
  - TIM2\_t, [1094](#)
  - TIM3\_t, [1112](#)
  - TIM5\_t, [1134](#)
- ARRL
  - TIM1\_t, [1060](#)
  - TIM2\_t, [1094](#)
  - TIM3\_t, [1113](#)
  - TIM5\_t, [1134](#)
- ARR
  - TIM1\_t, [1059](#)
  - TIM2\_t, [1094](#)
  - TIM3\_t, [1112](#)
  - TIM4\_t, [1125](#)
  - TIM5\_t, [1134](#)
  - TIM6\_t, [1153](#)
- AUX
  - CLK\_t, [986](#)
- AWCRH
  - ADC1\_t, [897](#)
- AWCRL
  - ADC1\_t, [897](#)
- AWDIE
  - ADC1\_t, [897](#)
- AWEN
  - ADC1\_t, [897](#)
- AWSRH
  - ADC1\_t, [898](#)
- AWSRL
  - ADC1\_t, [898](#)
- AWU\_AddressBase
  - STM8AF\_STM8S, [499–511](#)
- AWU\_t, [915](#)
  - APRE, [916](#)
  - APR, [916](#)
  - AWUEN, [916](#)
  - AWUTB, [916](#)
  - AWUF, [916](#)
  - CSR, [917](#)
  - MSR, [917](#)
  - res, [917](#)
  - res2, [917](#)
  - TBR, [917](#)
- AWUEN
  - AWU\_t, [916](#)
- AWUTB
  - AWU\_t, [916](#)
- AWUF
  - AWU\_t, [916](#)
- AWUM
  - CAN\_t, [937](#)
- AWD
  - ADC1\_t, [897](#)
- AWS
  - ADC1\_t, [898](#)
- AF
  - I2C\_t, [1013](#)
- AL
  - CFG\_t, [982](#)
- BDOE
  - SPI\_t, [1044](#)
- BDM
  - SPI\_t, [1044](#)
- BEEP\_AddressBase
  - STM8AF\_STM8S, [511–522](#)
- BEEP\_t, [918](#)
  - BEEPDIV, [918](#)

- BEEPEN, [918](#)
- BEEPSEL, [919](#)
- CSR, [919](#)
- BEEPDIV
  - BEEP\_t, [918](#)
- BEEPEN
  - BEEP\_t, [918](#)
- BEEPSEL
  - BEEP\_t, [919](#)
- BERR
  - I2C\_t, [1013](#)
- BIE
  - TIM1\_t, [1060](#)
- BIF
  - TIM1\_t, [1060](#)
- BKE
  - TIM1\_t, [1060](#)
- BKP
  - TIM1\_t, [1061](#)
- BKR
  - TIM1\_t, [1061](#)
- BOFIE
  - CAN\_t, [937](#)
- BOFF
  - CAN\_t, [937](#)
- BRR1
  - UART1\_t, [1160](#)
  - UART2\_t, [1175](#)
  - UART3\_t, [1191](#)
  - UART4\_t, [1204](#)
- BRR2
  - UART1\_t, [1161](#)
  - UART2\_t, [1175](#)
  - UART3\_t, [1191](#)
  - UART4\_t, [1204](#)
- BRP
  - CAN\_t, [937](#)
- BS1
  - CAN\_t, [937](#)
- BS2
  - CAN\_t, [938](#)
- BSY
  - SPI\_t, [1044](#)
- BTR1
  - CAN\_t, [938](#)
- BTR2
  - CAN\_t, [938](#)
- BTF
  - I2C\_t, [1014](#)
- BUSY
  - I2C\_t, [1014](#)
- BG
  - TIM1\_t, [1060](#)
- BR
  - SPI\_t, [1044](#)
- CAN\_AddressBase
  - STM8AF\_STM8S, [523–526](#)
- CAN\_t, [919](#)
- ABOM, [936](#)
- ABRQ, [936](#)
- ALST, [936](#)
- AWUM, [937](#)
- BOFIE, [937](#)
- BOFF, [937](#)
- BRP, [937](#)
- BS1, [937](#)
- BS2, [938](#)
- BTR1, [938](#)
- BTR2, [938](#)
- CODE, [938](#)
- DATA, [938](#)
- DGR, [940](#)
- DLC, [941](#)
- EIER, [941](#)
- EPVIE, [941](#)
- EPVF, [941](#)
- ERRIE, [942](#)
- ERRI, [941](#)
- ESR, [942](#)
- EWGIE, [942](#)
- EWGF, [942](#)
- EXID, [942](#)
- F0R1, [943](#)
- F0R2, [943](#)
- F0R3, [943](#)
- F0R4, [943](#)
- F0R5, [943](#)
- F0R6, [944](#)
- F0R7, [944](#)
- F0R8, [944](#)
- F1R1, [944](#)
- F1R2, [944](#)
- F1R3, [944](#)
- F1R4, [945](#)
- F1R5, [945](#)
- F1R6, [945](#)
- F1R7, [945](#)
- F1R8, [945](#)
- F2R1, [945](#)
- F2R2, [946](#)
- F2R3, [946](#)
- F2R4, [946](#)
- F2R5, [946](#)
- F2R6, [946](#)
- F2R7, [946](#)
- F2R8, [947](#)
- F3R1, [947](#)
- F3R2, [947](#)
- F3R3, [947](#)
- F3R4, [947](#)
- F3R5, [947](#)
- F3R6, [948](#)
- F3R7, [948](#)
- F3R8, [948](#)
- F4R1, [948](#)
- F4R2, [948](#)

F4R3, [948](#)  
F4R4, [949](#)  
F4R5, [949](#)  
F4R6, [949](#)  
F4R7, [949](#)  
F4R8, [949](#)  
F5R1, [949](#)  
F5R2, [950](#)  
F5R3, [950](#)  
F5R4, [950](#)  
F5R5, [950](#)  
F5R6, [950](#)  
F5R7, [950](#)  
F5R8, [951](#)  
FACT0, [951](#)  
FACT1, [951](#)  
FACT2, [951](#)  
FACT3, [951](#)  
FACT4, [952](#)  
FACT5, [952](#)  
FCR1, [952](#)  
FCR2, [952](#)  
FCR3, [952](#)  
FFIE, [953](#)  
FMH0, [953](#)  
FMH1, [953](#)  
FMH2, [953](#)  
FMH3, [953](#)  
FMH4, [954](#)  
FMH5, [954](#)  
FML0, [954](#)  
FML1, [954](#)  
FML2, [955](#)  
FML3, [955](#)  
FML4, [955](#)  
FML5, [955](#)  
FMPIE, [956](#)  
FMR1, [956](#)  
FMR2, [956](#)  
FMI, [954](#)  
FMP, [955](#)  
FOVIE, [956](#)  
FOVR, [956](#)  
FSC0, [957](#)  
FSC1, [957](#)  
FSC2, [957](#)  
FSC3, [957](#)  
FSC4, [957](#)  
FSC5, [958](#)  
FULL, [958](#)  
IDE, [958](#)  
IER, [958](#)  
INAK, [959](#)  
INRQ, [959](#)  
ID, [958](#)  
LBKM, [959](#)  
LECIE, [959](#)  
LEC, [959](#)  
LOW0, [960](#)  
LOW1, [960](#)  
LOW2, [960](#)  
MCSR, [960](#), [961](#)  
MCR, [960](#)  
MDAR1, [961](#)  
MDAR2, [962](#)  
MDAR3, [962](#), [963](#)  
MDAR4, [963](#)  
MDAR5, [964](#)  
MDAR6, [964](#), [965](#)  
MDAR7, [965](#)  
MDAR8, [966](#)  
MDLCR, [966](#), [967](#)  
MFMIR, [967](#)  
MIDR1, [967](#), [968](#)  
MIDR2, [968](#)  
MIDR3, [968](#), [969](#)  
MIDR4, [969](#), [970](#)  
MSR, [970](#)  
MTSRH, [970](#)  
MTSRL, [971](#)  
NART, [971](#)  
PAGE\_0, [972](#)  
PAGE\_1, [972](#)  
PAGE\_2, [972](#)  
PAGE\_3, [972](#)  
PAGE\_4, [972](#)  
PAGE\_5, [972](#)  
PAGE\_6, [973](#)  
PAGE\_7, [973](#)  
PSR, [973](#)  
Page, [971](#)  
PS, [973](#)  
RECR, [973](#)  
REC, [973](#)  
RFLM, [974](#)  
RFOM, [974](#)  
RFR, [975](#)  
RQCP0, [975](#)  
RQCP1, [975](#)  
RQCP2, [975](#)  
RQCP, [975](#)  
RTR, [976](#)  
res, [974](#)  
res2, [974](#)  
RX, [976](#)  
SAMP, [976](#)  
SILM, [976](#)  
SJW, [976](#)  
SLAK, [977](#)  
SLEEP, [977](#)  
TECR, [977](#)  
TERR, [977](#)  
TEC, [977](#)  
TGT, [978](#)  
TIME, [978](#)  
TME0, [978](#)

TME1, [978](#)  
 TME2, [978](#)  
 TMEIE, [979](#)  
 TPR, [979](#)  
 TSR, [979](#)  
 TTCM, [979](#)  
 TXFP, [980](#)  
 TXM2E, [980](#)  
 TXOK0, [980](#)  
 TXOK1, [980](#)  
 TXOK2, [981](#)  
 TXOK, [980](#)  
 TXRQ, [981](#)  
 TX, [979](#)  
 WKUIE, [981](#)  
 WKUI, [981](#)  
 CC1IE  
   TIM1\_t, [1061](#)  
   TIM2\_t, [1095](#)  
   TIM3\_t, [1113](#)  
   TIM5\_t, [1135](#)  
 CC1IF  
   TIM1\_t, [1062](#)  
   TIM2\_t, [1095](#)  
   TIM3\_t, [1113](#)  
   TIM5\_t, [1135](#)  
 CC1NE  
   TIM1\_t, [1062](#)  
 CC1NP  
   TIM1\_t, [1062](#)  
 CC1OF  
   TIM1\_t, [1062](#)  
   TIM2\_t, [1095](#)  
   TIM3\_t, [1114](#)  
   TIM5\_t, [1135](#)  
 CC1E  
   TIM1\_t, [1061](#)  
   TIM2\_t, [1094](#)  
   TIM3\_t, [1113](#)  
   TIM5\_t, [1135](#)  
 CC1G  
   TIM1\_t, [1061](#)  
   TIM2\_t, [1094](#)  
   TIM3\_t, [1113](#)  
   TIM5\_t, [1135](#)  
 CC1P  
   TIM1\_t, [1062](#)  
   TIM2\_t, [1095](#)  
   TIM3\_t, [1114](#)  
   TIM5\_t, [1136](#)  
 CC1S  
   TIM1\_t, [1063](#)  
   TIM2\_t, [1095](#)  
   TIM3\_t, [1114](#)  
   TIM5\_t, [1136](#)  
 CC2IE  
   TIM1\_t, [1063](#)  
   TIM2\_t, [1096](#)  
   TIM3\_t, [1115](#)  
   TIM5\_t, [1136](#)  
 CC2IF  
   TIM1\_t, [1063](#)  
   TIM2\_t, [1096](#)  
   TIM3\_t, [1115](#)  
   TIM5\_t, [1137](#)  
 CC2NE  
   TIM1\_t, [1064](#)  
 CC2NP  
   TIM1\_t, [1064](#)  
 CC2OF  
   TIM1\_t, [1064](#)  
   TIM2\_t, [1096](#)  
   TIM3\_t, [1115](#)  
   TIM5\_t, [1137](#)  
 CC2E  
   TIM1\_t, [1063](#)  
   TIM2\_t, [1096](#)  
   TIM3\_t, [1114](#)  
   TIM5\_t, [1136](#)  
 CC2G  
   TIM1\_t, [1063](#)  
   TIM2\_t, [1096](#)  
   TIM3\_t, [1114](#)  
   TIM5\_t, [1136](#)  
 CC2P  
   TIM1\_t, [1064](#)  
   TIM2\_t, [1097](#)  
   TIM3\_t, [1115](#)  
   TIM5\_t, [1137](#)  
 CC2S  
   TIM1\_t, [1064](#)  
   TIM2\_t, [1097](#)  
   TIM3\_t, [1115](#)  
   TIM5\_t, [1137](#)  
 CC3IE  
   TIM1\_t, [1065](#)  
   TIM2\_t, [1097](#)  
   TIM5\_t, [1138](#)  
 CC3IF  
   TIM1\_t, [1065](#)  
   TIM2\_t, [1098](#)  
   TIM5\_t, [1138](#)  
 CC3NE  
   TIM1\_t, [1065](#)  
 CC3NP  
   TIM1\_t, [1066](#)  
 CC3OF  
   TIM1\_t, [1066](#)  
   TIM2\_t, [1098](#)  
   TIM5\_t, [1138](#)  
 CC3E  
   TIM1\_t, [1065](#)  
   TIM2\_t, [1097](#)  
   TIM5\_t, [1137](#)  
 CC3G  
   TIM1\_t, [1065](#)



TIM2\_t, [1097](#)  
TIM5\_t, [1138](#)  
CC3P  
TIM1\_t, [1066](#)  
TIM2\_t, [1098](#)  
TIM5\_t, [1138](#)  
CC3S  
TIM1\_t, [1066](#)  
TIM2\_t, [1098](#)  
TIM5\_t, [1139](#)  
CC4IE  
TIM1\_t, [1067](#)  
CC4IF  
TIM1\_t, [1067](#)  
CC4OF  
TIM1\_t, [1067](#)  
CC4E  
TIM1\_t, [1066](#)  
CC4G  
TIM1\_t, [1067](#)  
CC4P  
TIM1\_t, [1067](#)  
CC4S  
TIM1\_t, [1068](#)  
CCER1  
TIM1\_t, [1068](#)  
TIM2\_t, [1098](#)  
TIM3\_t, [1116](#)  
TIM5\_t, [1139](#)  
CCER2  
TIM1\_t, [1068](#)  
TIM2\_t, [1099](#)  
TIM5\_t, [1139](#)  
CCMR1  
TIM1\_t, [1068](#)  
TIM2\_t, [1099](#)  
TIM3\_t, [1116](#)  
TIM5\_t, [1139](#)  
CCMR2  
TIM1\_t, [1068](#)  
TIM2\_t, [1099](#)  
TIM3\_t, [1116](#)  
TIM5\_t, [1139](#)  
CCMR3  
TIM1\_t, [1068](#)  
TIM2\_t, [1099](#)  
TIM5\_t, [1139](#)  
CCMR4  
TIM1\_t, [1069](#)  
CCOBSY  
CLK\_t, [986](#)  
CCOEN  
CLK\_t, [986](#)  
CCORDY  
CLK\_t, [987](#)  
CCOSEL  
CLK\_t, [987](#)  
CCOR  
CLK\_t, [987](#)  
CCPC  
TIM1\_t, [1069](#)  
TIM5\_t, [1140](#)  
CCR1  
TIM1\_t, [1069](#)  
TIM2\_t, [1099](#)  
TIM3\_t, [1116](#)  
TIM5\_t, [1140](#)  
CCR1H  
TIM1\_t, [1069](#)  
TIM2\_t, [1099](#)  
TIM3\_t, [1116](#)  
TIM5\_t, [1140](#)  
CCR1L  
TIM1\_t, [1069](#)  
TIM2\_t, [1100](#)  
TIM3\_t, [1117](#)  
TIM5\_t, [1140](#)  
CCR2  
TIM1\_t, [1070](#)  
TIM2\_t, [1100](#)  
TIM3\_t, [1117](#)  
TIM5\_t, [1140](#)  
CCR2H  
TIM1\_t, [1070](#)  
TIM2\_t, [1100](#)  
TIM3\_t, [1117](#)  
TIM5\_t, [1141](#)  
CCR2L  
TIM1\_t, [1070](#)  
TIM2\_t, [1100](#)  
TIM3\_t, [1117](#)  
TIM5\_t, [1141](#)  
CCR3  
TIM1\_t, [1070](#)  
TIM2\_t, [1100](#)  
TIM5\_t, [1141](#)  
CCR3H  
TIM1\_t, [1070](#)  
TIM2\_t, [1101](#)  
TIM5\_t, [1141](#)  
CCR3L  
TIM1\_t, [1071](#)  
TIM2\_t, [1101](#)  
TIM5\_t, [1141](#)  
CCR4  
TIM1\_t, [1071](#)  
CCR4H  
TIM1\_t, [1071](#)  
CCR4L  
TIM1\_t, [1071](#)  
CCRH  
I2C\_t, [1014](#)  
CCRL  
I2C\_t, [1014](#)  
CCR  
I2C\_t, [1014](#)

CEN  
   TIM1\_t, 1071  
   TIM2\_t, 1101  
   TIM3\_t, 1117  
   TIM4\_t, 1125  
   TIM5\_t, 1142  
   TIM6\_t, 1153  
 CFG\_AddressBase  
   STM8AF\_STM8S, 526–537  
 CFG\_t, 982  
   AL, 982  
   GCR, 982  
   res, 982  
   SWD, 983  
 CKDIVR  
   CLK\_t, 987  
 CKEN  
   UART1\_t, 1161  
   UART2\_t, 1175  
   UART4\_t, 1204  
 CKM  
   CLK\_t, 987  
 CLK\_AddressBase  
   STM8AF\_STM8S, 538–549  
 CLK\_t, 983  
   AUX, 986  
   CCOBSY, 986  
   CCOEN, 986  
   CCORDY, 987  
   CCOSEL, 987  
   CCOR, 987  
   CKDIVR, 987  
   CKM, 987  
   CMSR, 988  
   CPUDIV, 988  
   CSSDIE, 988  
   CSSEN, 988  
   CSSD, 988  
   CSSR, 989  
   ECKR, 989  
   FWU, 989  
   HSEEN, 989  
   HSERDY, 989  
   HSIDIV, 990  
   HSIEN, 990  
   HSIRDY, 990  
   HSITRIMR, 990  
   HSITRIM, 990  
   ICKR, 991  
   LSIEN, 991  
   LSIRDY, 991  
   PCKEN\_ADC, 991  
   PCKEN\_AWU, 991  
   PCKEN\_CAN, 992  
   PCKEN\_I2C, 992  
   PCKEN\_SPI, 992  
   PCKEN\_TIM1, 992  
   PCKEN\_TIM2\_TIM5, 992  
   PCKEN\_TIM3, 993  
   PCKEN\_TIM4\_TIM6, 993  
   PCKEN\_UART1, 993  
   PCKEN\_UART2, 993  
   PCKENR1, 993  
   PCKENR2, 994  
   REGAH, 994  
   res, 994  
   res2, 994, 995  
   SWBSY, 995  
   SWCR, 995  
   SWEN, 995  
   SWIEN, 996  
   SWIMCCR, 996  
   SWIMCLK, 996  
   SWIF, 996  
   SWI, 995  
   SWR, 996  
 CMSR  
   CLK\_t, 988  
 CMS  
   TIM1\_t, 1071  
 CNTRH  
   TIM1\_t, 1072  
   TIM2\_t, 1101  
   TIM3\_t, 1118  
   TIM5\_t, 1142  
 CNTRL  
   TIM1\_t, 1072  
   TIM2\_t, 1102  
   TIM3\_t, 1118  
   TIM5\_t, 1142  
 CNTR  
   TIM4\_t, 1126  
   TIM6\_t, 1153  
 CNT  
   TIM1\_t, 1072  
   TIM2\_t, 1101  
   TIM3\_t, 1117  
   TIM4\_t, 1126  
   TIM5\_t, 1142  
   TIM6\_t, 1153  
 CODE  
   CAN\_t, 938  
 COMIE  
   TIM1\_t, 1072  
 COMIF  
   TIM1\_t, 1073  
 COMG  
   TIM1\_t, 1072  
 COMS  
   TIM1\_t, 1073  
   TIM5\_t, 1142  
 CONT  
   ADC1\_t, 898  
   ADC2\_t, 911  
 CPHA  
   SPI\_t, 1045

- UART1\_t, [1161](#)
- UART2\_t, [1176](#)
- UART4\_t, [1205](#)
- CPOL
  - SPI\_t, [1045](#)
  - UART1\_t, [1161](#)
  - UART2\_t, [1176](#)
  - UART4\_t, [1205](#)
- CPUDIV
  - CLK\_t, [988](#)
- CR1
  - ADC1\_t, [899](#)
  - ADC2\_t, [911](#)
  - EXTI\_t, [997](#)
  - FLASH\_t, [1002](#)
  - I2C\_t, [1015](#)
  - PORT\_t, [1036](#)
  - SPI\_t, [1045](#)
  - TIM1\_t, [1073](#)
  - TIM2\_t, [1102](#)
  - TIM3\_t, [1118](#)
  - TIM5\_t, [1143](#)
  - TIM6\_t, [1154](#)
  - UART1\_t, [1161](#)
  - UART2\_t, [1176](#)
  - UART3\_t, [1191](#)
  - UART4\_t, [1205](#)
- CR2
  - ADC1\_t, [899](#)
  - ADC2\_t, [911](#)
  - EXTI\_t, [998](#)
  - FLASH\_t, [1002](#)
  - I2C\_t, [1015](#)
  - PORT\_t, [1036](#)
  - SPI\_t, [1045](#)
  - TIM1\_t, [1073](#)
  - TIM5\_t, [1143](#)
  - TIM6\_t, [1154](#)
  - UART1\_t, [1162](#)
  - UART2\_t, [1176](#)
  - UART3\_t, [1191](#)
  - UART4\_t, [1205](#)
- CR3
  - ADC1\_t, [899](#)
  - UART1\_t, [1162](#)
  - UART2\_t, [1176](#)
  - UART3\_t, [1191](#)
  - UART4\_t, [1205](#)
- CR4
  - UART1\_t, [1162](#)
  - UART2\_t, [1177](#)
  - UART3\_t, [1191](#)
  - UART4\_t, [1206](#)
- CR5
  - UART1\_t, [1162](#)
  - UART2\_t, [1177](#)
  - UART4\_t, [1206](#)
- CR6
  - UART2\_t, [1177](#)
  - UART3\_t, [1192](#)
  - UART4\_t, [1206](#)
- CRCERR
  - SPI\_t, [1046](#)
- CRCEN
  - SPI\_t, [1045](#)
- CRCNEXT
  - SPI\_t, [1046](#)
- CRCPOLY
  - SPI\_t, [1046](#)
- CRCPR
  - SPI\_t, [1046](#)
- CSSDIE
  - CLK\_t, [988](#)
- CSSSEN
  - CLK\_t, [988](#)
- CSSD
  - CLK\_t, [988](#)
- CSSR
  - CLK\_t, [989](#)
- CSR
  - ADC1\_t, [899](#)
  - ADC2\_t, [911](#)
  - AWU\_t, [917](#)
  - BEEP\_t, [919](#)
- CH
  - ADC1\_t, [898](#)
  - ADC2\_t, [910](#)
- CR
  - TIM4\_t, [1126](#)
  - WWDG\_t, [1217](#)
- DATA
  - ADC1\_t, [899](#)
  - ADC2\_t, [911](#)
  - CAN\_t, [938](#)
  - I2C\_t, [1015](#)
  - SPI\_t, [1046](#)
  - UART1\_t, [1162](#)
  - UART2\_t, [1177](#)
  - UART3\_t, [1192](#)
  - UART4\_t, [1206](#)
- DB0RH
  - ADC1\_t, [900](#)
- DB0RL
  - ADC1\_t, [900](#)
- DB1RH
  - ADC1\_t, [901](#)
- DB1RL
  - ADC1\_t, [901](#)
- DB2RH
  - ADC1\_t, [901](#)
- DB2RL
  - ADC1\_t, [901](#)
- DB3RH
  - ADC1\_t, [901](#)
- DB3RL
  - ADC1\_t, [901](#)

DB4RH  
     ADC1\_t, [902](#)  
 DB4RL  
     ADC1\_t, [902](#)  
 DB5RH  
     ADC1\_t, [902](#)  
 DB5RL  
     ADC1\_t, [902](#)  
 DB6RH  
     ADC1\_t, [902](#)  
 DB6RL  
     ADC1\_t, [902](#)  
 DB7RH  
     ADC1\_t, [903](#)  
 DB7RL  
     ADC1\_t, [903](#)  
 DB8RH  
     ADC1\_t, [903](#)  
 DB8RL  
     ADC1\_t, [903](#)  
 DB9RH  
     ADC1\_t, [903](#)  
 DB9RL  
     ADC1\_t, [903](#)  
 DBUF  
     ADC1\_t, [904](#)  
 DDR  
     PORT\_t, [1036](#)  
 DGR  
     CAN\_t, [940](#)  
 DISABLE\_INTERRUPTS  
     STM8AF\_STM8S, [549](#)  
 DIV\_0\_3  
     UART1\_t, [1162](#)  
     UART2\_t, [1177](#)  
     UART3\_t, [1192](#)  
     UART4\_t, [1206](#)  
 DIV\_12\_15  
     UART1\_t, [1163](#)  
     UART2\_t, [1177](#)  
     UART3\_t, [1192](#)  
     UART4\_t, [1206](#)  
 DIV\_4\_11  
     UART1\_t, [1163](#)  
     UART2\_t, [1178](#)  
     UART3\_t, [1192](#)  
     UART4\_t, [1207](#)  
 DIR  
     TIM1\_t, [1073](#)  
 DLC  
     CAN\_t, [941](#)  
 DM\_AddressBase  
     STM8AF\_STM8S, [549–561](#)  
 DRH  
     ADC1\_t, [904](#)  
     ADC2\_t, [912](#)  
 DRL  
     ADC1\_t, [904](#)  
     ADC2\_t, [912](#)  
 DTG  
     TIM1\_t, [1074](#)  
 DTR  
     TIM1\_t, [1074](#)  
 DUKR  
     FLASH\_t, [1002](#)  
 DUTY  
     I2C\_t, [1015](#)  
 DUK  
     FLASH\_t, [1002](#)  
 DUL  
     FLASH\_t, [1002](#)  
 DR  
     I2C\_t, [1015](#)  
     SPI\_t, [1047](#)  
     UART1\_t, [1163](#)  
     UART2\_t, [1178](#)  
     UART3\_t, [1193](#)  
     UART4\_t, [1207](#)  
 ECKR  
     CLK\_t, [989](#)  
 ECE  
     TIM1\_t, [1074](#)  
 EGR  
     TIM1\_t, [1074](#)  
     TIM2\_t, [1102](#)  
     TIM3\_t, [1118](#)  
     TIM4\_t, [1126](#)  
     TIM5\_t, [1143](#)  
     TIM6\_t, [1154](#)  
 EIER  
     CAN\_t, [941](#)  
 EMCf  
     RST\_t, [1040](#)  
 ENABLE\_INTERRUPTS  
     STM8AF\_STM8S, [561](#)  
 ENGC  
     I2C\_t, [1015](#)  
 ENTER\_HALT  
     STM8AF\_STM8S, [561](#)  
 EOCIE  
     ADC1\_t, [904](#)  
     ADC2\_t, [912](#)  
 EOC  
     ADC1\_t, [904](#)  
     ADC2\_t, [912](#)  
 EOP  
     FLASH\_t, [1002](#)  
 EPVIE  
     CAN\_t, [941](#)  
 EPVf  
     CAN\_t, [941](#)  
 ERASE  
     FLASH\_t, [1003](#)  
 ERRIE  
     CAN\_t, [942](#)  
     SPI\_t, [1047](#)

ERRI  
    CAN\_t, [941](#)  
ESR  
    CAN\_t, [942](#)  
ETPS  
    TIM1\_t, [1075](#)  
ETF  
    TIM1\_t, [1074](#)  
ETP  
    TIM1\_t, [1075](#)  
ETR  
    TIM1\_t, [1075](#)  
EWGIE  
    CAN\_t, [942](#)  
EWGF  
    CAN\_t, [942](#)  
EXID  
    CAN\_t, [942](#)  
EXTI\_AddressBase  
    STM8AF\_STM8S, [561–573](#)  
EXTI\_t, [997](#)  
    CR1, [997](#)  
    CR2, [998](#)  
    PAIS, [998](#)  
    PBIS, [998](#)  
    PCIS, [998](#)  
    PDIS, [998](#)  
    PEIS, [999](#)  
    res, [999](#)  
    TLIS, [999](#)  
EXTSEL  
    ADC1\_t, [904](#)  
    ADC2\_t, [912](#)  
EXTTRIG  
    ADC1\_t, [905](#)  
    ADC2\_t, [913](#)  
  
F0R1  
    CAN\_t, [943](#)  
F0R2  
    CAN\_t, [943](#)  
F0R3  
    CAN\_t, [943](#)  
F0R4  
    CAN\_t, [943](#)  
F0R5  
    CAN\_t, [943](#)  
F0R6  
    CAN\_t, [944](#)  
F0R7  
    CAN\_t, [944](#)  
F0R8  
    CAN\_t, [944](#)  
F1R1  
    CAN\_t, [944](#)  
F1R2  
    CAN\_t, [944](#)  
F1R3  
    CAN\_t, [944](#)  
F1R4  
    CAN\_t, [945](#)  
F1R5  
    CAN\_t, [945](#)  
F1R6  
    CAN\_t, [945](#)  
F1R7  
    CAN\_t, [945](#)  
F1R8  
    CAN\_t, [945](#)  
F2R1  
    CAN\_t, [945](#)  
F2R2  
    CAN\_t, [946](#)  
F2R3  
    CAN\_t, [946](#)  
F2R4  
    CAN\_t, [946](#)  
F2R5  
    CAN\_t, [946](#)  
F2R6  
    CAN\_t, [946](#)  
F2R7  
    CAN\_t, [946](#)  
F2R8  
    CAN\_t, [947](#)  
F3R1  
    CAN\_t, [947](#)  
F3R2  
    CAN\_t, [947](#)  
F3R3  
    CAN\_t, [947](#)  
F3R4  
    CAN\_t, [947](#)  
F3R5  
    CAN\_t, [947](#)  
F3R6  
    CAN\_t, [948](#)  
F3R7  
    CAN\_t, [948](#)  
F3R8  
    CAN\_t, [948](#)  
F4R1  
    CAN\_t, [948](#)  
F4R2  
    CAN\_t, [948](#)  
F4R3  
    CAN\_t, [948](#)  
F4R4  
    CAN\_t, [949](#)  
F4R5  
    CAN\_t, [949](#)  
F4R6  
    CAN\_t, [949](#)  
F4R7  
    CAN\_t, [949](#)  
F4R8  
    CAN\_t, [949](#)

F5R1  
     CAN\_t, [949](#)  
 F5R2  
     CAN\_t, [950](#)  
 F5R3  
     CAN\_t, [950](#)  
 F5R4  
     CAN\_t, [950](#)  
 F5R5  
     CAN\_t, [950](#)  
 F5R6  
     CAN\_t, [950](#)  
 F5R7  
     CAN\_t, [950](#)  
 F5R8  
     CAN\_t, [951](#)  
 FACT0  
     CAN\_t, [951](#)  
 FACT1  
     CAN\_t, [951](#)  
 FACT2  
     CAN\_t, [951](#)  
 FACT3  
     CAN\_t, [951](#)  
 FACT4  
     CAN\_t, [952](#)  
 FACT5  
     CAN\_t, [952](#)  
 FCR1  
     CAN\_t, [952](#)  
 FCR2  
     CAN\_t, [952](#)  
 FCR3  
     CAN\_t, [952](#)  
 FFIE  
     CAN\_t, [953](#)  
 FHWU  
     CLK\_t, [989](#)  
 FIX  
     FLASH\_t, [1003](#)  
 FLASH\_AddressBase  
     STM8AF\_STM8S, [573–585](#)  
 FLASH\_t, [999](#)  
     AHALT, [1001](#)  
     CR1, [1002](#)  
     CR2, [1002](#)  
     DUKR, [1002](#)  
     DUK, [1002](#)  
     DUL, [1002](#)  
     EOP, [1002](#)  
     ERASE, [1003](#)  
     FIX, [1003](#)  
     FPRG, [1003](#)  
     FPR, [1003](#)  
     HALT, [1003](#)  
     HVOFF, [1004](#)  
     IAPSR, [1004](#)  
     IE, [1004](#)  
     NCR2, [1004](#)  
     NERASE, [1004](#)  
     NFPRG, [1005](#)  
     NFPR, [1005](#)  
     NOPT, [1005](#)  
     NPRG, [1005](#)  
     NWPRG, [1006](#)  
     NWPB, [1005](#)  
     OPT, [1006](#)  
     PRG, [1006](#)  
     PUKR, [1006](#)  
     PUK, [1006](#)  
     PUL, [1007](#)  
     res, [1007](#)  
     res2, [1007](#)  
     WPRG, [1008](#)  
     WPB, [1008](#)  
     WR\_PG\_DIS, [1008](#)  
 FMH0  
     CAN\_t, [953](#)  
 FMH1  
     CAN\_t, [953](#)  
 FMH2  
     CAN\_t, [953](#)  
 FMH3  
     CAN\_t, [953](#)  
 FMH4  
     CAN\_t, [954](#)  
 FMH5  
     CAN\_t, [954](#)  
 FML0  
     CAN\_t, [954](#)  
 FML1  
     CAN\_t, [954](#)  
 FML2  
     CAN\_t, [955](#)  
 FML3  
     CAN\_t, [955](#)  
 FML4  
     CAN\_t, [955](#)  
 FML5  
     CAN\_t, [955](#)  
 FMPIE  
     CAN\_t, [956](#)  
 FMR1  
     CAN\_t, [956](#)  
 FMR2  
     CAN\_t, [956](#)  
 FMI  
     CAN\_t, [954](#)  
 FMP  
     CAN\_t, [955](#)  
 FOVIE  
     CAN\_t, [956](#)  
 FOVR  
     CAN\_t, [956](#)  
 FPRG  
     FLASH\_t, [1003](#)

FPR  
  FLASH\_t, 1003  
FREQR  
  I2C\_t, 1016  
FREQ  
  I2C\_t, 1016  
FSC0  
  CAN\_t, 957  
FSC1  
  CAN\_t, 957  
FSC2  
  CAN\_t, 957  
FSC3  
  CAN\_t, 957  
FSC4  
  CAN\_t, 957  
FSC5  
  CAN\_t, 958  
FULL  
  CAN\_t, 958  
FE  
  UART1\_t, 1163  
  UART2\_t, 1178  
  UART3\_t, 1193  
  UART4\_t, 1207  
FS  
  I2C\_t, 1016  
  
GCR  
  CFG\_t, 982  
GENCALL  
  I2C\_t, 1016  
GTR  
  UART1\_t, 1164  
  UART2\_t, 1178  
  UART4\_t, 1207  
GT  
  UART1\_t, 1163  
  UART2\_t, 1178  
  UART4\_t, 1207  
  
HALT  
  FLASH\_t, 1003  
HDSEL  
  UART1\_t, 1164  
  UART4\_t, 1208  
HSEEN  
  CLK\_t, 989  
HSERDY  
  CLK\_t, 989  
HSIDIV  
  CLK\_t, 990  
HSIEN  
  CLK\_t, 990  
HSIRDY  
  CLK\_t, 990  
HSITRIMR  
  CLK\_t, 990  
HSITRIM  
  CLK\_t, 990  
  
HTRH  
  ADC1\_t, 905  
HTRL  
  ADC1\_t, 905  
HVOFF  
  FLASH\_t, 1004  
HT  
  ADC1\_t, 905  
  
I2C\_AddressBase  
  STM8AF\_STM8S, 585–596  
I2C\_t, 1008  
  ACK, 1012  
  ADD0, 1012  
  ADD10, 1012  
  ADDCONF, 1012  
  ADDMODE, 1013  
  ADDR, 1013  
  ADD, 1012  
  ARLO, 1013  
  AF, 1013  
  BERR, 1013  
  BTF, 1014  
  BUSY, 1014  
  CCRH, 1014  
  CCRL, 1014  
  CCR, 1014  
  CR1, 1015  
  CR2, 1015  
  DATA, 1015  
  DUTY, 1015  
  DR, 1015  
  ENGCR, 1015  
  FREQR, 1016  
  FREQ, 1016  
  FS, 1016  
  GENCALL, 1016  
  ITBUFEN, 1016  
  ITERREN, 1017  
  ITEVTEN, 1017  
  ITR, 1017  
  MSL, 1017  
  NOSTRETCH, 1017  
  OARH, 1018  
  OARL, 1018  
  OVR, 1018  
  POS, 1018  
  PE, 1018  
  RXNE, 1019  
  res, 1019  
  res2, 1019  
  SR1, 1020  
  SR2, 1020  
  SR3, 1020  
  START, 1020  
  STOPF, 1021  
  STOP, 1020  
  SWRST, 1021

- SB, [1020](#)
- TRISER, [1021](#)
- TRISE, [1021](#)
- TRA, [1021](#)
- TXE, [1022](#)
- WUFH, [1022](#)
- IAPSR
  - FLASH\_t, [1004](#)
- IC1PSC
  - TIM1\_t, [1075](#)
  - TIM2\_t, [1102](#)
  - TIM3\_t, [1118](#)
  - TIM5\_t, [1143](#)
- IC1F
  - TIM1\_t, [1075](#)
  - TIM2\_t, [1102](#)
  - TIM3\_t, [1118](#)
  - TIM5\_t, [1143](#)
- IC2PSC
  - TIM1\_t, [1076](#)
  - TIM2\_t, [1103](#)
  - TIM3\_t, [1119](#)
  - TIM5\_t, [1144](#)
- IC2F
  - TIM1\_t, [1076](#)
  - TIM2\_t, [1102](#)
  - TIM3\_t, [1119](#)
  - TIM5\_t, [1143](#)
- IC3PSC
  - TIM1\_t, [1076](#)
  - TIM2\_t, [1103](#)
  - TIM5\_t, [1144](#)
- IC3F
  - TIM1\_t, [1076](#)
  - TIM2\_t, [1103](#)
  - TIM5\_t, [1144](#)
- IC4PSC
  - TIM1\_t, [1077](#)
- IC4F
  - TIM1\_t, [1076](#)
- ICKR
  - CLK\_t, [991](#)
- ICR
  - SPI\_t, [1047](#)
- IDLE
  - UART1\_t, [1164](#)
  - UART2\_t, [1179](#)
  - UART3\_t, [1193](#)
  - UART4\_t, [1208](#)
- IDE
  - CAN\_t, [958](#)
- IDR
  - PORT\_t, [1037](#)
- IER
  - CAN\_t, [958](#)
  - TIM1\_t, [1077](#)
  - TIM2\_t, [1103](#)
  - TIM3\_t, [1119](#)
  - TIM4\_t, [1126](#)
  - TIM5\_t, [1144](#)
  - TIM6\_t, [1154](#)
- ILIEN
  - UART1\_t, [1164](#)
  - UART2\_t, [1179](#)
  - UART3\_t, [1193](#)
  - UART4\_t, [1208](#)
- ILLOPF
  - RST\_t, [1040](#)
- INAK
  - CAN\_t, [959](#)
- INRQ
  - CAN\_t, [959](#)
- IREN
  - UART1\_t, [1164](#)
  - UART2\_t, [1179](#)
  - UART4\_t, [1208](#)
- IRLP
  - UART1\_t, [1165](#)
  - UART2\_t, [1179](#)
  - UART4\_t, [1208](#)
- ISR\_HANDLER\_TRAP
  - STM8AF\_STM8S, [597](#)
- ISR\_HANDLER
  - STM8AF\_STM8S, [596](#)
- ITBUFEN
  - I2C\_t, [1016](#)
- ITC\_AddressBase
  - STM8AF\_STM8S, [597–608](#)
- ITC\_t, [1022](#)
  - res, [1024](#)
  - SPR1, [1025](#)
  - SPR2, [1025](#)
  - SPR3, [1025](#)
  - SPR4, [1025](#)
  - SPR5, [1025](#)
  - SPR6, [1025](#)
  - SPR7, [1026](#)
  - SPR8, [1026](#)
  - VECT10SPR, [1026](#)
  - VECT11SPR, [1026](#)
  - VECT12SPR, [1026](#)
  - VECT13SPR, [1026](#)
  - VECT14SPR, [1027](#)
  - VECT15SPR, [1027](#)
  - VECT16SPR, [1027](#)
  - VECT17SPR, [1027](#)
  - VECT18SPR, [1027](#)
  - VECT19SPR, [1028](#)
  - VECT1SPR, [1028](#)
  - VECT20SPR, [1028](#)
  - VECT21SPR, [1028](#)
  - VECT22SPR, [1028](#)
  - VECT23SPR, [1029](#)
  - VECT24SPR, [1029](#)
  - VECT25SPR, [1029](#)
  - VECT26SPR, [1029](#)



- VECT27SPR, [1029](#)
- VECT28SPR, [1030](#)
- VECT29SPR, [1030](#)
- VECT2SPR, [1030](#)
- VECT3SPR, [1030](#)
- VECT4SPR, [1030](#)
- VECT5SPR, [1031](#)
- VECT6SPR, [1031](#)
- VECT7SPR, [1031](#)
- VECT8SPR, [1031](#)
- VECT9SPR, [1031](#)
- ITERREN
  - I2C\_t, [1017](#)
- ITEVTEN
  - I2C\_t, [1017](#)
- ITR
  - I2C\_t, [1017](#)
- IWDG\_AddressBase
  - STM8AF\_STM8S, [609–620](#)
- IWDG\_t, [1032](#)
  - KEY, [1032](#)
  - KR, [1033](#)
  - PRE, [1033](#)
  - PR, [1033](#)
  - RLR, [1034](#)
  - res, [1033](#)
  - RL, [1033](#)
- IWDGF
  - RST\_t, [1041](#)
- ID
  - CAN\_t, [958](#)
- IE
  - FLASH\_t, [1004](#)
- IN
  - TIM1\_t, [1077](#)
  - TIM2\_t, [1103](#), [1104](#)
  - TIM3\_t, [1119](#)
  - TIM5\_t, [1144](#), [1145](#)
- KEY
  - IWDG\_t, [1032](#)
- KR
  - IWDG\_t, [1033](#)
- LASE
  - UART2\_t, [1179](#)
  - UART3\_t, [1193](#)
  - UART4\_t, [1209](#)
- LBCL
  - UART1\_t, [1165](#)
  - UART2\_t, [1180](#)
  - UART4\_t, [1209](#)
- LBDIEN
  - UART1\_t, [1165](#)
  - UART2\_t, [1180](#)
  - UART3\_t, [1194](#)
  - UART4\_t, [1209](#)
- LBDF
  - UART1\_t, [1165](#)
- UART2\_t, [1180](#)
- UART3\_t, [1194](#)
- UART4\_t, [1209](#)
- LBDL
  - UART1\_t, [1165](#)
  - UART2\_t, [1180](#)
  - UART3\_t, [1194](#)
  - UART4\_t, [1209](#)
- LBKM
  - CAN\_t, [959](#)
- LDUM
  - UART2\_t, [1180](#)
  - UART3\_t, [1194](#)
  - UART4\_t, [1210](#)
- LECIE
  - CAN\_t, [959](#)
- LEC
  - CAN\_t, [959](#)
- LHDIEN
  - UART2\_t, [1181](#)
  - UART3\_t, [1195](#)
  - UART4\_t, [1210](#)
- LHDF
  - UART2\_t, [1181](#)
  - UART3\_t, [1194](#)
  - UART4\_t, [1210](#)
- LINEN
  - UART1\_t, [1166](#)
  - UART2\_t, [1181](#)
  - UART3\_t, [1195](#)
  - UART4\_t, [1210](#)
- LOCK
  - TIM1\_t, [1078](#)
- LOW0
  - CAN\_t, [960](#)
- LOW1
  - CAN\_t, [960](#)
- LOW2
  - CAN\_t, [960](#)
- LSBFIRST
  - SPI\_t, [1047](#)
- LSIEN
  - CLK\_t, [991](#)
- LSIRDY
  - CLK\_t, [991](#)
- LSLV
  - UART2\_t, [1181](#)
  - UART3\_t, [1195](#)
  - UART4\_t, [1211](#)
- LSF
  - UART2\_t, [1181](#)
  - UART3\_t, [1195](#)
  - UART4\_t, [1210](#)
- LTRH
  - ADC1\_t, [906](#)
- LTRL
  - ADC1\_t, [906](#)
- LT

ADC1\_t, [905](#)  
 M  
   UART1\_t, [1166](#)  
   UART2\_t, [1182](#)  
   UART3\_t, [1195](#)  
   UART4\_t, [1211](#)  
 MCSR  
   CAN\_t, [960](#), [961](#)  
 MCR  
   CAN\_t, [960](#)  
 MDAR1  
   CAN\_t, [961](#)  
 MDAR2  
   CAN\_t, [962](#)  
 MDAR3  
   CAN\_t, [962](#), [963](#)  
 MDAR4  
   CAN\_t, [963](#)  
 MDAR5  
   CAN\_t, [964](#)  
 MDAR6  
   CAN\_t, [964](#), [965](#)  
 MDAR7  
   CAN\_t, [965](#)  
 MDAR8  
   CAN\_t, [966](#)  
 MDLCR  
   CAN\_t, [966](#), [967](#)  
 MFMIR  
   CAN\_t, [967](#)  
 MIDR1  
   CAN\_t, [967](#), [968](#)  
 MIDR2  
   CAN\_t, [968](#)  
 MIDR3  
   CAN\_t, [968](#), [969](#)  
 MIDR4  
   CAN\_t, [969](#), [970](#)  
 MMS  
   TIM1\_t, [1078](#)  
   TIM5\_t, [1145](#)  
   TIM6\_t, [1154](#)  
 MODF  
   SPI\_t, [1047](#)  
 MOE  
   TIM1\_t, [1078](#)  
 MSTR  
   SPI\_t, [1048](#)  
 MSL  
   I2C\_t, [1017](#)  
 MSM  
   TIM1\_t, [1078](#)  
   TIM5\_t, [1145](#)  
   TIM6\_t, [1154](#)  
 MSR  
   AWU\_t, [917](#)  
   CAN\_t, [970](#)  
 MTSRH  
   CAN\_t, [970](#)  
 MTSRL  
   CAN\_t, [971](#)  
 NACK  
   UART1\_t, [1166](#)  
   UART2\_t, [1182](#)  
   UART4\_t, [1211](#)  
 NART  
   CAN\_t, [971](#)  
 NCR2  
   FLASH\_t, [1004](#)  
 NERASE  
   FLASH\_t, [1004](#)  
 NFPRG  
   FLASH\_t, [1005](#)  
 NFPR  
   FLASH\_t, [1005](#)  
 NOPT  
   FLASH\_t, [1005](#)  
 NOSTRETCH  
   I2C\_t, [1017](#)  
 NOP  
   STM8AF\_STM8S, [620](#)  
 NPRG  
   FLASH\_t, [1005](#)  
 NWPRG  
   FLASH\_t, [1006](#)  
 NWPB  
   FLASH\_t, [1005](#)  
 NF  
   UART1\_t, [1166](#)  
   UART2\_t, [1182](#)  
   UART3\_t, [1196](#)  
   UART4\_t, [1211](#)  
 OARH  
   I2C\_t, [1018](#)  
 OARL  
   I2C\_t, [1018](#)  
 OC1CE  
   TIM1\_t, [1078](#)  
 OC1FE  
   TIM1\_t, [1079](#)  
 OC1PE  
   TIM1\_t, [1079](#)  
   TIM2\_t, [1104](#)  
   TIM3\_t, [1120](#)  
   TIM5\_t, [1145](#)  
 OC1M  
   TIM1\_t, [1079](#)  
   TIM2\_t, [1104](#)  
   TIM3\_t, [1120](#)  
   TIM5\_t, [1145](#)  
 OC2CE  
   TIM1\_t, [1079](#)  
 OC2FE  
   TIM1\_t, [1079](#)  
 OC2PE

- TIM1\_t, [1080](#)
- TIM2\_t, [1104](#)
- TIM3\_t, [1120](#)
- TIM5\_t, [1146](#)
- OC2M
  - TIM1\_t, [1080](#)
  - TIM2\_t, [1104](#)
  - TIM3\_t, [1120](#)
  - TIM5\_t, [1146](#)
- OC3CE
  - TIM1\_t, [1080](#)
  - TIM2\_t, [1105](#)
- OC3FE
  - TIM1\_t, [1080](#)
- OC3PE
  - TIM1\_t, [1081](#)
  - TIM2\_t, [1105](#)
  - TIM5\_t, [1146](#)
- OC3M
  - TIM1\_t, [1080](#)
  - TIM2\_t, [1105](#)
  - TIM5\_t, [1146](#)
- OC4CE
  - TIM1\_t, [1081](#)
- OC4FE
  - TIM1\_t, [1081](#)
- OC4PE
  - TIM1\_t, [1081](#)
- OC4M
  - TIM1\_t, [1081](#)
- ODR
  - PORT\_t, [1037](#)
- OIS1
  - TIM1\_t, [1082](#)
- OIS1N
  - TIM1\_t, [1082](#)
- OIS2
  - TIM1\_t, [1082](#)
- OIS2N
  - TIM1\_t, [1082](#)
- OIS3
  - TIM1\_t, [1082](#)
- OIS3N
  - TIM1\_t, [1083](#)
- OIS4
  - TIM1\_t, [1083](#)
- OISR
  - TIM1\_t, [1083](#)
- OPT\_AddressBase
  - STM8AF\_STM8S, [620–632](#)
- OPM
  - TIM1\_t, [1083](#)
  - TIM2\_t, [1105](#)
  - TIM3\_t, [1120](#)
  - TIM4\_t, [1126](#)
  - TIM5\_t, [1146](#)
  - TIM6\_t, [1155](#)
- OPT
  - FLASH\_t, [1006](#)
- OSSI
  - TIM1\_t, [1083](#)
- OSSR
  - TIM1\_t, [1084](#)
- OUT
  - TIM1\_t, [1084](#)
  - TIM2\_t, [1105](#), [1106](#)
  - TIM3\_t, [1121](#)
  - TIM5\_t, [1147](#)
- OVR
  - ADC1\_t, [906](#)
  - I2C\_t, [1018](#)
  - SPI\_t, [1048](#)
- OR
  - UART1\_t, [1166](#)
  - UART2\_t, [1182](#)
  - UART3\_t, [1196](#)
  - UART4\_t, [1211](#)
- PAGE\_0
  - CAN\_t, [972](#)
- PAGE\_1
  - CAN\_t, [972](#)
- PAGE\_2
  - CAN\_t, [972](#)
- PAGE\_3
  - CAN\_t, [972](#)
- PAGE\_4
  - CAN\_t, [972](#)
- PAGE\_5
  - CAN\_t, [972](#)
- PAGE\_6
  - CAN\_t, [973](#)
- PAGE\_7
  - CAN\_t, [973](#)
- PAIS
  - EXTI\_t, [998](#)
- PBIS
  - EXTI\_t, [998](#)
- PCEN
  - UART1\_t, [1167](#)
  - UART2\_t, [1182](#)
  - UART3\_t, [1196](#)
  - UART4\_t, [1212](#)
- PCIS
  - EXTI\_t, [998](#)
- PCKEN\_ADC
  - CLK\_t, [991](#)
- PCKEN\_AWU
  - CLK\_t, [991](#)
- PCKEN\_CAN
  - CLK\_t, [992](#)
- PCKEN\_I2C
  - CLK\_t, [992](#)
- PCKEN\_SPI
  - CLK\_t, [992](#)
- PCKEN\_TIM1
  - CLK\_t, [992](#)

PCKEN\_TIM2\_TIM5  
     CLK\_t, [992](#)  
 PCKEN\_TIM3  
     CLK\_t, [993](#)  
 PCKEN\_TIM4\_TIM6  
     CLK\_t, [993](#)  
 PCKEN\_UART1  
     CLK\_t, [993](#)  
 PCKEN\_UART2  
     CLK\_t, [993](#)  
 PCKENR1  
     CLK\_t, [993](#)  
 PCKENR2  
     CLK\_t, [994](#)  
 PDIS  
     EXTI\_t, [998](#)  
 PEIS  
     EXTI\_t, [999](#)  
 PIEN  
     UART1\_t, [1167](#)  
     UART2\_t, [1183](#)  
     UART3\_t, [1196](#)  
     UART4\_t, [1212](#)  
 PIN0  
     PORT\_t, [1037](#)  
 PIN1  
     PORT\_t, [1037](#)  
 PIN2  
     PORT\_t, [1037](#)  
 PIN3  
     PORT\_t, [1038](#)  
 PIN4  
     PORT\_t, [1038](#)  
 PIN5  
     PORT\_t, [1038](#)  
 PIN6  
     PORT\_t, [1039](#)  
 PIN7  
     PORT\_t, [1039](#)  
 PORT\_t, [1034](#)  
     CR1, [1036](#)  
     CR2, [1036](#)  
     DDR, [1036](#)  
     IDR, [1037](#)  
     ODR, [1037](#)  
     PIN0, [1037](#)  
     PIN1, [1037](#)  
     PIN2, [1037](#)  
     PIN3, [1038](#)  
     PIN4, [1038](#)  
     PIN5, [1038](#)  
     PIN6, [1039](#)  
     PIN7, [1039](#)  
 PORTA\_AddressBase  
     STM8AF\_STM8S, [632–644](#)  
 PORTB\_AddressBase  
     STM8AF\_STM8S, [644–655](#)  
 PORTC\_AddressBase  
     STM8AF\_STM8S, [655–667](#)  
 PORTD\_AddressBase  
     STM8AF\_STM8S, [667–679](#)  
 PORTE\_AddressBase  
     STM8AF\_STM8S, [679–690](#)  
 PORTF\_AddressBase  
     STM8AF\_STM8S, [690–702](#)  
 PORTG\_AddressBase  
     STM8AF\_STM8S, [702–711](#)  
 PORTH\_AddressBase  
     STM8AF\_STM8S, [711–718](#)  
 PORTI\_AddressBase  
     STM8AF\_STM8S, [719–726](#)  
 POS  
     I2C\_t, [1018](#)  
 PRE  
     IWDG\_t, [1033](#)  
 PRG  
     FLASH\_t, [1006](#)  
 PSCRH  
     TIM1\_t, [1085](#)  
 PSCRL  
     TIM1\_t, [1085](#)  
 PSCR  
     TIM2\_t, [1106](#)  
     TIM3\_t, [1121](#)  
     TIM4\_t, [1127](#)  
     TIM5\_t, [1147](#)  
     TIM6\_t, [1155](#)  
     UART1\_t, [1168](#)  
     UART2\_t, [1183](#)  
     UART4\_t, [1213](#)  
 PSC  
     TIM1\_t, [1084](#)  
     TIM2\_t, [1106](#)  
     TIM3\_t, [1121](#)  
     TIM4\_t, [1127](#)  
     TIM5\_t, [1147](#)  
     TIM6\_t, [1155](#)  
     UART1\_t, [1167](#)  
     UART2\_t, [1183](#)  
     UART4\_t, [1212](#)  
 PSR  
     CAN\_t, [973](#)  
 PUKR  
     FLASH\_t, [1006](#)  
 PUK  
     FLASH\_t, [1006](#)  
 PUL  
     FLASH\_t, [1007](#)  
 Page  
     CAN\_t, [971](#)  
 pages/mainpage.txt, [1219](#)  
 PE  
     I2C\_t, [1018](#)  
     UART1\_t, [1167](#)  
     UART2\_t, [1183](#)  
     UART3\_t, [1196](#)

- UART4\_t, [1212](#)
- PR
  - IWDG\_t, [1033](#)
- PS
  - CAN\_t, [973](#)
  - UART1\_t, [1167](#)
  - UART2\_t, [1183](#)
  - UART3\_t, [1197](#)
  - UART4\_t, [1212](#)
- R8
  - UART1\_t, [1168](#)
  - UART2\_t, [1184](#)
  - UART3\_t, [1197](#)
  - UART4\_t, [1213](#)
- RCR
  - TIM1\_t, [1085](#)
- RECR
  - CAN\_t, [973](#)
- REGAH
  - CLK\_t, [994](#)
- REC
  - CAN\_t, [973](#)
- REN
  - UART1\_t, [1168](#)
  - UART2\_t, [1184](#)
  - UART3\_t, [1197](#)
  - UART4\_t, [1213](#)
- REP
  - TIM1\_t, [1085](#)
- RFLM
  - CAN\_t, [974](#)
- RFOM
  - CAN\_t, [974](#)
- RFR
  - CAN\_t, [975](#)
- RIEN
  - UART1\_t, [1169](#)
  - UART2\_t, [1185](#)
  - UART3\_t, [1198](#)
  - UART4\_t, [1214](#)
- RLR
  - IWDG\_t, [1034](#)
- RQCP0
  - CAN\_t, [975](#)
- RQCP1
  - CAN\_t, [975](#)
- RQCP2
  - CAN\_t, [975](#)
- RQCP
  - CAN\_t, [975](#)
- RST\_AddressBase
  - STM8AF\_STM8S, [726–737](#)
- RST\_t, [1040](#)
  - EMCF, [1040](#)
  - ILLOPF, [1040](#)
  - IWDGF, [1041](#)
  - res, [1041](#)
  - SWIMF, [1041](#)
  - SR, [1041](#)
  - WWDGF, [1041](#)
- RTR
  - CAN\_t, [976](#)
- RWU
  - UART1\_t, [1169](#)
  - UART2\_t, [1185](#)
  - UART3\_t, [1198](#)
  - UART4\_t, [1214](#)
- RXCRCR
  - SPI\_t, [1048](#)
- RXIE
  - SPI\_t, [1049](#)
- RXNE
  - I2C\_t, [1019](#)
  - SPI\_t, [1049](#)
  - UART1\_t, [1169](#)
  - UART2\_t, [1185](#)
  - UART3\_t, [1198](#)
  - UART4\_t, [1214](#)
- RXONLY
  - SPI\_t, [1049](#)
- res
  - ADC1\_t, [906](#)
  - ADC2\_t, [913](#)
  - AWU\_t, [917](#)
  - CAN\_t, [974](#)
  - CFG\_t, [982](#)
  - CLK\_t, [994](#)
  - EXTI\_t, [999](#)
  - FLASH\_t, [1007](#)
  - I2C\_t, [1019](#)
  - ITC\_t, [1024](#)
  - IWDG\_t, [1033](#)
  - RST\_t, [1041](#)
  - SPI\_t, [1048](#)
  - TIM1\_t, [1085](#)
  - TIM2\_t, [1106](#)
  - TIM3\_t, [1121](#)
  - TIM4\_t, [1127](#)
  - TIM5\_t, [1147](#)
  - TIM6\_t, [1155](#)
  - UART1\_t, [1168](#)
  - UART2\_t, [1184](#)
  - UART3\_t, [1197](#)
  - UART4\_t, [1213](#)
  - WWDG\_t, [1218](#)
- res2
  - ADC1\_t, [907](#)
  - ADC2\_t, [913](#)
  - AWU\_t, [917](#)
  - CAN\_t, [974](#)
  - CLK\_t, [994, 995](#)
  - FLASH\_t, [1007](#)
  - I2C\_t, [1019](#)
  - TIM1\_t, [1086](#)
  - TIM2\_t, [1106](#)
  - TIM3\_t, [1122](#)

TIM5\_t, [1148](#)  
 TIM6\_t, [1155](#)  
 UART1\_t, [1168](#)  
 UART2\_t, [1184](#)  
 UART3\_t, [1198](#)  
 UART4\_t, [1213](#)  
 res3  
   ADC1\_t, [907](#)  
   ADC2\_t, [913](#)  
   TIM1\_t, [1086](#)  
   TIM5\_t, [1148](#)  
   UART2\_t, [1184](#)  
 RL  
   IWDG\_t, [1033](#)  
 RX  
   CAN\_t, [976](#)  
 RxCRC  
   SPI\_t, [1048](#)  
 SAMP  
   CAN\_t, [976](#)  
 SBK  
   UART1\_t, [1169](#)  
   UART2\_t, [1185](#)  
   UART3\_t, [1198](#)  
   UART4\_t, [1214](#)  
 SCAN  
   ADC1\_t, [907](#)  
 SCEN  
   UART1\_t, [1169](#)  
   UART2\_t, [1185](#)  
   UART4\_t, [1214](#)  
 SILM  
   CAN\_t, [976](#)  
 SJW  
   CAN\_t, [976](#)  
 SLAK  
   CAN\_t, [977](#)  
 SLEEP  
   CAN\_t, [977](#)  
 SMCR  
   TIM1\_t, [1086](#)  
   TIM5\_t, [1148](#)  
   TIM6\_t, [1156](#)  
 SMS  
   TIM1\_t, [1086](#)  
   TIM5\_t, [1148](#)  
   TIM6\_t, [1156](#)  
 SPI\_AddressBase  
   STM8AF\_STM8S, [737–749](#)  
 SPI\_t, [1042](#)  
   BDOE, [1044](#)  
   BDM, [1044](#)  
   BSY, [1044](#)  
   BR, [1044](#)  
   CPHA, [1045](#)  
   CPOL, [1045](#)  
   CR1, [1045](#)  
   CR2, [1045](#)  
   CRCERR, [1046](#)  
   CRCEN, [1045](#)  
   CRCNEXT, [1046](#)  
   CRCPOLY, [1046](#)  
   CRCPR, [1046](#)  
   DATA, [1046](#)  
   DR, [1047](#)  
   ERRIE, [1047](#)  
   ICR, [1047](#)  
   LSBFIRST, [1047](#)  
   MODF, [1047](#)  
   MSTR, [1048](#)  
   OVR, [1048](#)  
   RXCRCR, [1048](#)  
   RXIE, [1049](#)  
   RXNE, [1049](#)  
   RXONLY, [1049](#)  
   res, [1048](#)  
   RxCRC, [1048](#)  
   SPE, [1049](#)  
   SSI, [1050](#)  
   SSM, [1050](#)  
   SR, [1049](#)  
   TXCRCR, [1050](#)  
   TXIE, [1051](#)  
   TXE, [1050](#)  
   TxCRC, [1050](#)  
   WKIE, [1051](#)  
   WKUP, [1051](#)  
 SPR1  
   ITC\_t, [1025](#)  
 SPR2  
   ITC\_t, [1025](#)  
 SPR3  
   ITC\_t, [1025](#)  
 SPR4  
   ITC\_t, [1025](#)  
 SPR5  
   ITC\_t, [1025](#)  
 SPR6  
   ITC\_t, [1025](#)  
 SPR7  
   ITC\_t, [1026](#)  
 SPR8  
   ITC\_t, [1026](#)  
 SPSEL  
   ADC1\_t, [907](#)  
   ADC2\_t, [914](#)  
 SPE  
   SPI\_t, [1049](#)  
 SR1  
   I2C\_t, [1020](#)  
   TIM1\_t, [1086](#)  
   TIM2\_t, [1107](#)  
   TIM3\_t, [1122](#)  
   TIM5\_t, [1148](#)  
 SR2  
   I2C\_t, [1020](#)

- TIM1\_t, [1087](#)
- TIM2\_t, [1107](#)
- TIM3\_t, [1122](#)
- TIM5\_t, [1149](#)
- SR3
- I2C\_t, [1020](#)
- SSI
- SPI\_t, [1050](#)
- SSM
- SPI\_t, [1050](#)
- START
- I2C\_t, [1020](#)
- STM8\_ADDR\_WIDTH
- STM8AF\_STM8S, [749](#)
- STM8\_EEPROM\_END
- STM8AF\_STM8S, [749](#)
- STM8\_EEPROM\_SIZE
- STM8AF\_STM8S, [749–761](#)
- STM8\_EEPROM\_START
- STM8AF\_STM8S, [761](#)
- STM8\_MEM\_POINTER\_T
- STM8AF\_STM8S, [761](#)
- STM8\_PFLASH\_END
- STM8AF\_STM8S, [762](#)
- STM8\_PFLASH\_SIZE
- STM8AF\_STM8S, [762–773](#)
- STM8\_PFLASH\_START
- STM8AF\_STM8S, [774](#)
- STM8\_RAM\_END
- STM8AF\_STM8S, [774](#)
- STM8\_RAM\_SIZE
- STM8AF\_STM8S, [774–786](#)
- STM8\_RAM\_START
- STM8AF\_STM8S, [786](#)
- STM8AF5268
- STM8AF\_STM8S, [786](#)
- STM8AF5269
- STM8AF\_STM8S, [786](#)
- STM8AF526x
- STM8AF\_STM8S, [786, 787](#)
- STM8AF5286
- STM8AF\_STM8S, [787](#)
- STM8AF5288
- STM8AF\_STM8S, [787](#)
- STM8AF5289
- STM8AF\_STM8S, [787](#)
- STM8AF528A
- STM8AF\_STM8S, [787](#)
- STM8AF528x
- STM8AF\_STM8S, [787, 788](#)
- STM8AF52A6
- STM8AF\_STM8S, [788](#)
- STM8AF52A8
- STM8AF\_STM8S, [788](#)
- STM8AF52A9
- STM8AF\_STM8S, [788](#)
- STM8AF52AA
- STM8AF\_STM8S, [789](#)
- STM8AF52Ax
- STM8AF\_STM8S, [789](#)
- STM8AF6213
- STM8AF\_STM8S, [789](#)
- STM8AF6213A
- STM8AF\_STM8S, [790](#)
- STM8AF621x
- STM8AF\_STM8S, [790](#)
- STM8AF6223
- STM8AF\_STM8S, [790](#)
- STM8AF6223A
- STM8AF\_STM8S, [790](#)
- STM8AF6226
- STM8AF\_STM8S, [790](#)
- STM8AF622x
- STM8AF\_STM8S, [791](#)
- STM8AF6246
- STM8AF\_STM8S, [791](#)
- STM8AF6248
- STM8AF\_STM8S, [791](#)
- STM8AF624x
- STM8AF\_STM8S, [791, 792](#)
- STM8AF6266
- STM8AF\_STM8S, [792](#)
- STM8AF6268
- STM8AF\_STM8S, [792](#)
- STM8AF6269
- STM8AF\_STM8S, [792](#)
- STM8AF626x
- STM8AF\_STM8S, [792, 793](#)
- STM8AF6286
- STM8AF\_STM8S, [793](#)
- STM8AF6288
- STM8AF\_STM8S, [793](#)
- STM8AF6289
- STM8AF\_STM8S, [793](#)
- STM8AF628A
- STM8AF\_STM8S, [793](#)
- STM8AF628x
- STM8AF\_STM8S, [793, 794](#)
- STM8AF62A6
- STM8AF\_STM8S, [794](#)
- STM8AF62A8
- STM8AF\_STM8S, [794](#)
- STM8AF62A9
- STM8AF\_STM8S, [794](#)
- STM8AF62AA
- STM8AF\_STM8S, [795](#)
- STM8AF62Ax
- STM8AF\_STM8S, [795](#)
- STM8AF6366
- STM8AF\_STM8S, [795](#)
- STM8AF636x
- STM8AF\_STM8S, [796](#)
- STM8AF6388
- STM8AF\_STM8S, [796](#)
- STM8AF638x
- STM8AF\_STM8S, [796](#)

STM8AF\_STM8S, 9  
   \_ADC1, 130  
   \_ADC1\_ADON, 131  
   \_ADC1\_ALIGN, 131  
   \_ADC1\_AWD, 132  
   \_ADC1\_AWCRH, 131  
   \_ADC1\_AWCRL, 131  
   \_ADC1\_AWCRH\_RESET\_VALUE, 131  
   \_ADC1\_AWCRL\_RESET\_VALUE, 132  
   \_ADC1\_AWDIE, 132  
   \_ADC1\_AWSRH, 132  
   \_ADC1\_AWSRL, 132  
   \_ADC1\_CH, 133  
   \_ADC1\_CH0, 133  
   \_ADC1\_CH1, 133  
   \_ADC1\_CH2, 133  
   \_ADC1\_CH3, 133  
   \_ADC1\_CONT, 134  
   \_ADC1\_CR1, 134  
   \_ADC1\_CR1\_RESET\_VALUE, 134  
   \_ADC1\_CR2, 134  
   \_ADC1\_CR2\_RESET\_VALUE, 134  
   \_ADC1\_CR3, 135  
   \_ADC1\_CR3\_RESET\_VALUE, 135  
   \_ADC1\_CSR, 135  
   \_ADC1\_CSR\_RESET\_VALUE, 135  
   \_ADC1\_DB0RH, 135  
   \_ADC1\_DB0RL, 136  
   \_ADC1\_DB1RH, 136  
   \_ADC1\_DB1RL, 136  
   \_ADC1\_DB2RH, 136  
   \_ADC1\_DB2RL, 136  
   \_ADC1\_DB3RH, 137  
   \_ADC1\_DB3RL, 137  
   \_ADC1\_DB4RH, 137  
   \_ADC1\_DB4RL, 137  
   \_ADC1\_DB5RH, 137  
   \_ADC1\_DB5RL, 138  
   \_ADC1\_DB6RH, 138  
   \_ADC1\_DB6RL, 138  
   \_ADC1\_DB7RH, 138  
   \_ADC1\_DB7RL, 138  
   \_ADC1\_DB8RH, 139  
   \_ADC1\_DB8RL, 139  
   \_ADC1\_DB9RH, 139  
   \_ADC1\_DB9RL, 139  
   \_ADC1\_DBUF, 139  
   \_ADC1\_DRH, 140  
   \_ADC1\_DRL, 140  
   \_ADC1\_EOC, 140  
   \_ADC1\_EOCIE, 140  
   \_ADC1\_EXTSEL, 140  
   \_ADC1\_EXTSEL0, 141  
   \_ADC1\_EXTSEL1, 141  
   \_ADC1\_EXTTRIG, 141  
   \_ADC1\_HTRH, 141  
   \_ADC1\_HTRL, 142  
   \_ADC1\_HTRH\_RESET\_VALUE, 141  
   \_ADC1\_HTRL\_RESET\_VALUE, 142  
   \_ADC1\_LTRH, 142  
   \_ADC1\_LTRL, 142  
   \_ADC1\_LTRH\_RESET\_VALUE, 142  
   \_ADC1\_LTRL\_RESET\_VALUE, 143  
   \_ADC1\_OVR, 143  
   \_ADC1\_SCAN, 143  
   \_ADC1\_SPSEL, 143  
   \_ADC1\_SPSEL0, 143  
   \_ADC1\_SPSEL1, 144  
   \_ADC1\_SPSEL2, 144  
   \_ADC1\_TDRH, 144  
   \_ADC1\_TDRL, 144  
   \_ADC1\_TDRH\_RESET\_VALUE, 144  
   \_ADC1\_TDRL\_RESET\_VALUE, 145  
   \_ADC2, 145  
   \_ADC2\_ADON, 145  
   \_ADC2\_ALIGN, 145  
   \_ADC2\_CH, 145  
   \_ADC2\_CH0, 146  
   \_ADC2\_CH1, 146  
   \_ADC2\_CH2, 146  
   \_ADC2\_CH3, 146  
   \_ADC2\_CONT, 146  
   \_ADC2\_CR1, 147  
   \_ADC2\_CR1\_RESET\_VALUE, 147  
   \_ADC2\_CR2, 147  
   \_ADC2\_CR2\_RESET\_VALUE, 147  
   \_ADC2\_CSR, 147  
   \_ADC2\_CSR\_RESET\_VALUE, 148  
   \_ADC2\_DRH, 148  
   \_ADC2\_DRL, 148  
   \_ADC2\_EOC, 148  
   \_ADC2\_EOCIE, 148  
   \_ADC2\_EXTSEL, 149  
   \_ADC2\_EXTSEL0, 149  
   \_ADC2\_EXTSEL1, 149  
   \_ADC2\_EXTTRIG, 149  
   \_ADC2\_SPSEL, 149  
   \_ADC2\_SPSEL0, 150  
   \_ADC2\_SPSEL1, 150  
   \_ADC2\_SPSEL2, 150  
   \_ADC2\_TDRH, 150  
   \_ADC2\_TDRL, 151  
   \_ADC2\_TDRH\_RESET\_VALUE, 150  
   \_ADC2\_TDRL\_RESET\_VALUE, 151  
   \_AWU, 151  
   \_AWU\_APR, 151  
   \_AWU\_APRE, 152  
   \_AWU\_APRE0, 152  
   \_AWU\_APRE1, 152  
   \_AWU\_APRE2, 152  
   \_AWU\_APRE3, 152  
   \_AWU\_APRE4, 153  
   \_AWU\_APRE5, 153  
   \_AWU\_APR\_RESET\_VALUE, 151  
   \_AWU\_AWUF, 153  
   \_AWU\_AWUEN, 153



[\\_AWU\\_AWUTB](#), 153  
[\\_AWU\\_AWUTB0](#), 154  
[\\_AWU\\_AWUTB1](#), 154  
[\\_AWU\\_AWUTB2](#), 154  
[\\_AWU\\_AWUTB3](#), 154  
[\\_AWU\\_CSR](#), 154  
[\\_AWU\\_CSR\\_RESET\\_VALUE](#), 155  
[\\_AWU\\_MSR](#), 155  
[\\_AWU\\_TBR](#), 155  
[\\_AWU\\_TBR\\_RESET\\_VALUE](#), 155  
[\\_BEEP](#), 155  
[\\_BEEP\\_BEEP\\_DIV](#), 156  
[\\_BEEP\\_BEEP\\_DIV0](#), 156  
[\\_BEEP\\_BEEP\\_DIV1](#), 156  
[\\_BEEP\\_BEEP\\_DIV2](#), 156  
[\\_BEEP\\_BEEP\\_DIV3](#), 156  
[\\_BEEP\\_BEEP\\_DIV4](#), 157  
[\\_BEEP\\_BEEPEN](#), 157  
[\\_BEEP\\_BEEPSEL](#), 157  
[\\_BEEP\\_BEEPSEL0](#), 157  
[\\_BEEP\\_BEEPSEL1](#), 157  
[\\_BEEP\\_CSR](#), 158  
[\\_BEEP\\_CSR\\_RESET\\_VALUE](#), 158  
[\\_BITS](#), 158  
[\\_CAN](#), 158  
[\\_CAN\\_ABOM](#), 158  
[\\_CAN\\_ABRQ](#), 159  
[\\_CAN\\_ALST](#), 159  
[\\_CAN\\_AWUM](#), 159  
[\\_CAN\\_BOFF](#), 159  
[\\_CAN\\_BOFIE](#), 159  
[\\_CAN\\_BRP](#), 160  
[\\_CAN\\_BRP0](#), 160  
[\\_CAN\\_BRP1](#), 160  
[\\_CAN\\_BRP2](#), 160  
[\\_CAN\\_BRP3](#), 160  
[\\_CAN\\_BRP4](#), 161  
[\\_CAN\\_BRP5](#), 161  
[\\_CAN\\_BS1](#), 161  
[\\_CAN\\_BS10](#), 161  
[\\_CAN\\_BS11](#), 161  
[\\_CAN\\_BS12](#), 162  
[\\_CAN\\_BS13](#), 162  
[\\_CAN\\_BS2](#), 162  
[\\_CAN\\_BS20](#), 162  
[\\_CAN\\_BS21](#), 162  
[\\_CAN\\_BS22](#), 163  
[\\_CAN\\_BTR1](#), 163  
[\\_CAN\\_BTR1\\_RESET\\_VALUE](#), 163  
[\\_CAN\\_BTR2](#), 163  
[\\_CAN\\_BTR2\\_RESET\\_VALUE](#), 163  
[\\_CAN\\_CODE](#), 164  
[\\_CAN\\_CODE0](#), 164  
[\\_CAN\\_CODE1](#), 164  
[\\_CAN\\_DGR](#), 164  
[\\_CAN\\_DGR\\_RESET\\_VALUE](#), 164  
[\\_CAN\\_DLC](#), 165  
[\\_CAN\\_DLC0](#), 165  
[\\_CAN\\_DLC1](#), 165  
[\\_CAN\\_DLC2](#), 165  
[\\_CAN\\_DLC3](#), 165  
[\\_CAN\\_EIER](#), 166  
[\\_CAN\\_EIER\\_RESET\\_VALUE](#), 166  
[\\_CAN\\_EPVF](#), 166  
[\\_CAN\\_EPVIE](#), 166  
[\\_CAN\\_ERRI](#), 166  
[\\_CAN\\_ERRIE](#), 167  
[\\_CAN\\_ESR](#), 167  
[\\_CAN\\_ESR\\_RESET\\_VALUE](#), 167  
[\\_CAN\\_EWGF](#), 167  
[\\_CAN\\_EWGIE](#), 167  
[\\_CAN\\_F0R1](#), 168  
[\\_CAN\\_F0R2](#), 168  
[\\_CAN\\_F0R3](#), 168  
[\\_CAN\\_F0R4](#), 168  
[\\_CAN\\_F0R5](#), 168  
[\\_CAN\\_F0R6](#), 169  
[\\_CAN\\_F0R7](#), 169  
[\\_CAN\\_F0R8](#), 169  
[\\_CAN\\_F1R1](#), 169  
[\\_CAN\\_F1R2](#), 169  
[\\_CAN\\_F1R3](#), 170  
[\\_CAN\\_F1R4](#), 170  
[\\_CAN\\_F1R5](#), 170  
[\\_CAN\\_F1R6](#), 170  
[\\_CAN\\_F1R7](#), 170  
[\\_CAN\\_F1R8](#), 171  
[\\_CAN\\_F2R1](#), 171  
[\\_CAN\\_F2R2](#), 171  
[\\_CAN\\_F2R3](#), 171  
[\\_CAN\\_F2R4](#), 171  
[\\_CAN\\_F2R5](#), 172  
[\\_CAN\\_F2R6](#), 172  
[\\_CAN\\_F2R7](#), 172  
[\\_CAN\\_F2R8](#), 172  
[\\_CAN\\_F3R1](#), 172  
[\\_CAN\\_F3R2](#), 173  
[\\_CAN\\_F3R3](#), 173  
[\\_CAN\\_F3R4](#), 173  
[\\_CAN\\_F3R5](#), 173  
[\\_CAN\\_F3R6](#), 173  
[\\_CAN\\_F3R7](#), 174  
[\\_CAN\\_F3R8](#), 174  
[\\_CAN\\_F4R1](#), 174  
[\\_CAN\\_F4R2](#), 174  
[\\_CAN\\_F4R3](#), 174  
[\\_CAN\\_F4R4](#), 175  
[\\_CAN\\_F4R5](#), 175  
[\\_CAN\\_F4R6](#), 175  
[\\_CAN\\_F4R7](#), 175  
[\\_CAN\\_F4R8](#), 175  
[\\_CAN\\_F5R1](#), 176  
[\\_CAN\\_F5R2](#), 176  
[\\_CAN\\_F5R3](#), 176  
[\\_CAN\\_F5R4](#), 176  
[\\_CAN\\_F5R5](#), 176

[\\_CAN\\_F5R6, 177](#)  
[\\_CAN\\_F5R7, 177](#)  
[\\_CAN\\_F5R8, 177](#)  
[\\_CAN\\_FACT0, 177](#)  
[\\_CAN\\_FACT1, 177](#)  
[\\_CAN\\_FACT2, 178](#)  
[\\_CAN\\_FACT3, 178](#)  
[\\_CAN\\_FACT4, 178](#)  
[\\_CAN\\_FACT5, 178](#)  
[\\_CAN\\_FCR1, 178](#)  
[\\_CAN\\_FCR2, 179](#)  
[\\_CAN\\_FCR3, 179](#)  
[\\_CAN\\_FCR\\_RESET\\_VALUE, 179](#)  
[\\_CAN\\_FFIE, 179](#)  
[\\_CAN\\_FMP, 182](#)  
[\\_CAN\\_FMH0, 179](#)  
[\\_CAN\\_FMH1, 180](#)  
[\\_CAN\\_FMH2, 180](#)  
[\\_CAN\\_FMH3, 180](#)  
[\\_CAN\\_FMH4, 180](#)  
[\\_CAN\\_FMH5, 180](#)  
[\\_CAN\\_FML0, 181](#)  
[\\_CAN\\_FML1, 181](#)  
[\\_CAN\\_FML2, 181](#)  
[\\_CAN\\_FML3, 181](#)  
[\\_CAN\\_FML4, 181](#)  
[\\_CAN\\_FML5, 182](#)  
[\\_CAN\\_FMP0, 182](#)  
[\\_CAN\\_FMP1, 182](#)  
[\\_CAN\\_FMPIE, 182](#)  
[\\_CAN\\_FMR1, 183](#)  
[\\_CAN\\_FMR1\\_RESET\\_VALUE, 183](#)  
[\\_CAN\\_FMR2, 183](#)  
[\\_CAN\\_FMR2\\_RESET\\_VALUE, 183](#)  
[\\_CAN\\_FOVR, 184](#)  
[\\_CAN\\_FOVIE, 183](#)  
[\\_CAN\\_FSC0, 184](#)  
[\\_CAN\\_FSC00, 184](#)  
[\\_CAN\\_FSC01, 184](#)  
[\\_CAN\\_FSC1, 184](#)  
[\\_CAN\\_FSC10, 185](#)  
[\\_CAN\\_FSC11, 185](#)  
[\\_CAN\\_FSC2, 185](#)  
[\\_CAN\\_FSC20, 185](#)  
[\\_CAN\\_FSC21, 185](#)  
[\\_CAN\\_FSC3, 186](#)  
[\\_CAN\\_FSC30, 186](#)  
[\\_CAN\\_FSC31, 186](#)  
[\\_CAN\\_FSC4, 186](#)  
[\\_CAN\\_FSC40, 186](#)  
[\\_CAN\\_FSC41, 187](#)  
[\\_CAN\\_FSC5, 187](#)  
[\\_CAN\\_FSC50, 187](#)  
[\\_CAN\\_FSC51, 187](#)  
[\\_CAN\\_FULL, 187](#)  
[\\_CAN\\_IDE, 188](#)  
[\\_CAN\\_IER, 188](#)  
[\\_CAN\\_IER\\_RESET\\_VALUE, 188](#)  
[\\_CAN\\_INAK, 188](#)  
[\\_CAN\\_INRQ, 188](#)  
[\\_CAN\\_LBKM, 189](#)  
[\\_CAN\\_LEC, 189](#)  
[\\_CAN\\_LEC0, 189](#)  
[\\_CAN\\_LEC1, 189](#)  
[\\_CAN\\_LEC2, 189](#)  
[\\_CAN\\_LECIE, 190](#)  
[\\_CAN\\_LOW0, 190](#)  
[\\_CAN\\_LOW1, 190](#)  
[\\_CAN\\_LOW2, 190](#)  
[\\_CAN\\_MCR, 190](#)  
[\\_CAN\\_MCR\\_RESET\\_VALUE, 191](#)  
[\\_CAN\\_MCSR, 191](#)  
[\\_CAN\\_MCSR\\_RESET\\_VALUE, 191](#)  
[\\_CAN\\_MDAR1, 191](#)  
[\\_CAN\\_MDAR2, 191](#)  
[\\_CAN\\_MDAR3, 192](#)  
[\\_CAN\\_MDAR4, 192](#)  
[\\_CAN\\_MDAR5, 192](#)  
[\\_CAN\\_MDAR6, 192](#)  
[\\_CAN\\_MDAR7, 192](#)  
[\\_CAN\\_MDAR8, 193](#)  
[\\_CAN\\_MDLCR, 193](#)  
[\\_CAN\\_MDLCR\\_RESET\\_VALUE, 193](#)  
[\\_CAN\\_MFMIR, 193](#)  
[\\_CAN\\_MFMIR\\_RESET\\_VALUE, 193](#)  
[\\_CAN\\_MIDR1, 194](#)  
[\\_CAN\\_MIDR2, 194](#)  
[\\_CAN\\_MIDR3, 194](#)  
[\\_CAN\\_MIDR4, 194](#)  
[\\_CAN\\_MSR, 194](#)  
[\\_CAN\\_MSR\\_RESET\\_VALUE, 195](#)  
[\\_CAN\\_MTSRH, 195](#)  
[\\_CAN\\_MTSRL, 195](#)  
[\\_CAN\\_NART, 195](#)  
[\\_CAN\\_PS, 195](#)  
[\\_CAN\\_PS0, 196](#)  
[\\_CAN\\_PS1, 196](#)  
[\\_CAN\\_PS2, 196](#)  
[\\_CAN\\_PSR, 196](#)  
[\\_CAN\\_PSR\\_RESET\\_VALUE, 196](#)  
[\\_CAN\\_RX, 199](#)  
[\\_CAN\\_RECR, 197](#)  
[\\_CAN\\_RECR\\_RESET\\_VALUE, 197](#)  
[\\_CAN\\_RFR, 197](#)  
[\\_CAN\\_RFLM, 197](#)  
[\\_CAN\\_RFOM, 197](#)  
[\\_CAN\\_RFR\\_RESET\\_VALUE, 198](#)  
[\\_CAN\\_RQCP, 198](#)  
[\\_CAN\\_RQCP0, 198](#)  
[\\_CAN\\_RQCP1, 198](#)  
[\\_CAN\\_RQCP2, 198](#)  
[\\_CAN\\_RTR, 199](#)  
[\\_CAN\\_RXS, 199](#)  
[\\_CAN\\_SAMP, 199](#)  
[\\_CAN\\_SILM, 199](#)  
[\\_CAN\\_SJW, 200](#)

[\\_CAN\\_SJW0](#), [200](#)  
[\\_CAN\\_SJW1](#), [200](#)  
[\\_CAN\\_SLAK](#), [200](#)  
[\\_CAN\\_SLEEP](#), [200](#)  
[\\_CAN\\_TX](#), [203](#)  
[\\_CAN\\_TECR](#), [201](#)  
[\\_CAN\\_TECR\\_RESET\\_VALUE](#), [201](#)  
[\\_CAN\\_TERR](#), [201](#)  
[\\_CAN\\_TGT](#), [201](#)  
[\\_CAN\\_TME0](#), [201](#)  
[\\_CAN\\_TME1](#), [202](#)  
[\\_CAN\\_TME2](#), [202](#)  
[\\_CAN\\_TMEIE](#), [202](#)  
[\\_CAN\\_TPR](#), [202](#)  
[\\_CAN\\_TPR\\_RESET\\_VALUE](#), [202](#)  
[\\_CAN\\_TSR](#), [203](#)  
[\\_CAN\\_TSR\\_RESET\\_VALUE](#), [203](#)  
[\\_CAN\\_TTCM](#), [203](#)  
[\\_CAN\\_TXFP](#), [203](#)  
[\\_CAN\\_TXM2E](#), [204](#)  
[\\_CAN\\_TXOK](#), [204](#)  
[\\_CAN\\_TXOK0](#), [204](#)  
[\\_CAN\\_TXOK1](#), [204](#)  
[\\_CAN\\_TXOK2](#), [204](#)  
[\\_CAN\\_TXRQ](#), [205](#)  
[\\_CAN\\_WKUI](#), [205](#)  
[\\_CAN\\_WKUIE](#), [205](#)  
[\\_CFG](#), [205](#)  
[\\_CFG\\_AL](#), [205](#)  
[\\_CFG\\_GCR](#), [206](#)  
[\\_CFG\\_GCR\\_RESET\\_VALUE](#), [206](#)  
[\\_CFG\\_SWD](#), [206](#)  
[\\_CLK](#), [206](#)  
[\\_CLK\\_ADC](#), [206](#)  
[\\_CLK\\_AUX](#), [207](#)  
[\\_CLK\\_AWU](#), [207](#)  
[\\_CLK\\_CAN](#), [207](#)  
[\\_CLK\\_CCOR](#), [208](#)  
[\\_CLK\\_CCOBSY](#), [207](#)  
[\\_CLK\\_CCOEN](#), [207](#)  
[\\_CLK\\_CCORDY](#), [208](#)  
[\\_CLK\\_CCOR\\_RESET\\_VALUE](#), [208](#)  
[\\_CLK\\_CCOSEL](#), [208](#)  
[\\_CLK\\_CCOSEL0](#), [208](#)  
[\\_CLK\\_CCOSEL1](#), [209](#)  
[\\_CLK\\_CCOSEL2](#), [209](#)  
[\\_CLK\\_CCOSEL3](#), [209](#)  
[\\_CLK\\_CKDIVR](#), [209](#)  
[\\_CLK\\_CKDIVR\\_RESET\\_VALUE](#), [209](#)  
[\\_CLK\\_CMSR](#), [210](#)  
[\\_CLK\\_CMSR\\_RESET\\_VALUE](#), [210](#)  
[\\_CLK\\_CPUDIV](#), [210](#)  
[\\_CLK\\_CPUDIV0](#), [210](#)  
[\\_CLK\\_CPUDIV1](#), [210](#)  
[\\_CLK\\_CPUDIV2](#), [211](#)  
[\\_CLK\\_CSSD](#), [211](#)  
[\\_CLK\\_CSSR](#), [211](#)  
[\\_CLK\\_CSSDIE](#), [211](#)  
[\\_CLK\\_CSSEN](#), [211](#)  
[\\_CLK\\_CSSR\\_RESET\\_VALUE](#), [212](#)  
[\\_CLK\\_ECKR](#), [212](#)  
[\\_CLK\\_ECKR\\_HSERDY](#), [212](#)  
[\\_CLK\\_ECKR\\_RESET\\_VALUE](#), [212](#)  
[\\_CLK\\_FHWU](#), [212](#)  
[\\_CLK\\_HSEEN](#), [213](#)  
[\\_CLK\\_HSIDIV](#), [213](#)  
[\\_CLK\\_HSIDIV0](#), [213](#)  
[\\_CLK\\_HSIDIV1](#), [213](#)  
[\\_CLK\\_HSIEN](#), [213](#)  
[\\_CLK\\_HSIRDY](#), [214](#)  
[\\_CLK\\_HSITRIM](#), [214](#)  
[\\_CLK\\_HSITRIM0](#), [214](#)  
[\\_CLK\\_HSITRIM1](#), [214](#)  
[\\_CLK\\_HSITRIM2](#), [214](#)  
[\\_CLK\\_HSITRIM3](#), [215](#)  
[\\_CLK\\_HSITRIMR](#), [215](#)  
[\\_CLK\\_HSITRIMR\\_RESET\\_VALUE](#), [215](#)  
[\\_CLK\\_I2C](#), [215](#)  
[\\_CLK\\_ICKR](#), [215](#)  
[\\_CLK\\_ICKR\\_RESET\\_VALUE](#), [216](#)  
[\\_CLK\\_LSIEN](#), [216](#)  
[\\_CLK\\_LSIRDY](#), [216](#)  
[\\_CLK\\_PCKENR1](#), [216](#)  
[\\_CLK\\_PCKENR1\\_RESET\\_VALUE](#), [216](#)  
[\\_CLK\\_PCKENR2](#), [217](#)  
[\\_CLK\\_PCKENR2\\_RESET\\_VALUE](#), [217](#)  
[\\_CLK\\_REGAH](#), [217](#)  
[\\_CLK\\_SPI](#), [217](#)  
[\\_CLK\\_SWR](#), [220](#)  
[\\_CLK\\_SWBSY](#), [217](#)  
[\\_CLK\\_SWCR](#), [218](#)  
[\\_CLK\\_SWCR\\_RESET\\_VALUE](#), [218](#)  
[\\_CLK\\_SWEN](#), [218](#)  
[\\_CLK\\_SWIF](#), [219](#)  
[\\_CLK\\_SWIEN](#), [219](#)  
[\\_CLK\\_SWIMCCR](#), [219](#)  
[\\_CLK\\_SWIMCCR\\_RESET\\_VALUE](#), [219](#)  
[\\_CLK\\_SWIMCLK](#), [220](#)  
[\\_CLK\\_SWI\\_HSE](#), [218](#)  
[\\_CLK\\_SWI\\_HSI](#), [218](#)  
[\\_CLK\\_SWI\\_LSI](#), [219](#)  
[\\_CLK\\_SWR\\_RESET\\_VALUE](#), [220](#)  
[\\_CLK\\_TIM1](#), [220](#)  
[\\_CLK\\_TIM2\\_TIM5](#), [220](#)  
[\\_CLK\\_TIM3](#), [221](#)  
[\\_CLK\\_TIM4\\_TIM6](#), [221](#)  
[\\_CLK\\_UART1](#), [221](#)  
[\\_CLK\\_UART2](#), [221](#)  
[\\_EXTI](#), [221](#)  
[\\_EXTI\\_CR1](#), [222](#)  
[\\_EXTI\\_CR1\\_RESET\\_VALUE](#), [222](#)  
[\\_EXTI\\_CR2](#), [222](#)  
[\\_EXTI\\_CR2\\_RESET\\_VALUE](#), [222](#)  
[\\_EXTI\\_PAIS](#), [222](#)  
[\\_EXTI\\_PAIS0](#), [223](#)  
[\\_EXTI\\_PAIS1](#), [223](#)

[\\_EXTI\\_PBIS, 223](#)  
[\\_EXTI\\_PBIS0, 223](#)  
[\\_EXTI\\_PBIS1, 223](#)  
[\\_EXTI\\_PCIS, 224](#)  
[\\_EXTI\\_PCIS0, 224](#)  
[\\_EXTI\\_PCIS1, 224](#)  
[\\_EXTI\\_PDIS, 224](#)  
[\\_EXTI\\_PDIS0, 224](#)  
[\\_EXTI\\_PDIS1, 225](#)  
[\\_EXTI\\_PEIS, 225](#)  
[\\_EXTI\\_PEIS0, 225](#)  
[\\_EXTI\\_PEIS1, 225](#)  
[\\_EXTI\\_TLIS, 225](#)  
[\\_FLASH, 226](#)  
[\\_FLASH\\_AHALT, 226](#)  
[\\_FLASH\\_CR1, 226](#)  
[\\_FLASH\\_CR1\\_RESET\\_VALUE, 226](#)  
[\\_FLASH\\_CR2, 226](#)  
[\\_FLASH\\_CR2\\_RESET\\_VALUE, 227](#)  
[\\_FLASH\\_DUL, 227](#)  
[\\_FLASH\\_DUKR, 227](#)  
[\\_FLASH\\_DUKR\\_RESET\\_VALUE, 227](#)  
[\\_FLASH\\_EOP, 227](#)  
[\\_FLASH\\_ERASE, 228](#)  
[\\_FLASH\\_FIX, 228](#)  
[\\_FLASH\\_FPR, 228](#)  
[\\_FLASH\\_FPRG, 228](#)  
[\\_FLASH\\_HALT, 228](#)  
[\\_FLASH\\_HVOFF, 229](#)  
[\\_FLASH\\_IE, 229](#)  
[\\_FLASH\\_IAPSR, 229](#)  
[\\_FLASH\\_IAPSR\\_RESET\\_VALUE, 229](#)  
[\\_FLASH\\_NCR2, 229](#)  
[\\_FLASH\\_NCR2\\_RESET\\_VALUE, 230](#)  
[\\_FLASH\\_NFPR, 230](#)  
[\\_FLASH\\_OPT, 230](#)  
[\\_FLASH\\_PRG, 230](#)  
[\\_FLASH\\_PUL, 231](#)  
[\\_FLASH\\_PUKR, 230](#)  
[\\_FLASH\\_PUKR\\_RESET\\_VALUE, 231](#)  
[\\_FLASH\\_WPB, 231](#)  
[\\_FLASH\\_WPB0, 231](#)  
[\\_FLASH\\_WPB1, 231](#)  
[\\_FLASH\\_WPB2, 232](#)  
[\\_FLASH\\_WPB3, 232](#)  
[\\_FLASH\\_WPB4, 232](#)  
[\\_FLASH\\_WPB5, 232](#)  
[\\_FLASH\\_WPRG, 232](#)  
[\\_FLASH\\_WR\\_PG\\_DIS, 233](#)  
[\\_GPIOA, 235](#)  
[\\_GPIOB, 236](#)  
[\\_GPIOC, 238](#)  
[\\_GPIOD, 239](#)  
[\\_GPIOE, 240](#)  
[\\_GPIOF, 241](#)  
[\\_GPIOG, 242](#)  
[\\_GPIOH, 244](#)  
[\\_GPIOI, 245](#)  
[\\_GPIOA\\_CR1, 235](#)  
[\\_GPIOA\\_CR2, 236](#)  
[\\_GPIOA\\_DDR, 236](#)  
[\\_GPIOA\\_IDR, 236](#)  
[\\_GPIOA\\_ODR, 236](#)  
[\\_GPIOB\\_CR1, 237](#)  
[\\_GPIOB\\_CR2, 237](#)  
[\\_GPIOB\\_DDR, 237](#)  
[\\_GPIOB\\_IDR, 237](#)  
[\\_GPIOB\\_ODR, 237](#)  
[\\_GPIOC\\_CR1, 238](#)  
[\\_GPIOC\\_CR2, 238](#)  
[\\_GPIOC\\_DDR, 238](#)  
[\\_GPIOC\\_IDR, 238](#)  
[\\_GPIOC\\_ODR, 239](#)  
[\\_GPIOD\\_CR1, 239](#)  
[\\_GPIOD\\_CR2, 239](#)  
[\\_GPIOD\\_DDR, 239](#)  
[\\_GPIOD\\_IDR, 240](#)  
[\\_GPIOD\\_ODR, 240](#)  
[\\_GPIOE\\_CR1, 240](#)  
[\\_GPIOE\\_CR2, 240](#)  
[\\_GPIOE\\_DDR, 241](#)  
[\\_GPIOE\\_IDR, 241](#)  
[\\_GPIOE\\_ODR, 241](#)  
[\\_GPIOF\\_CR1, 241](#)  
[\\_GPIOF\\_CR2, 242](#)  
[\\_GPIOF\\_DDR, 242](#)  
[\\_GPIOF\\_IDR, 242](#)  
[\\_GPIOF\\_ODR, 242](#)  
[\\_GPIOG\\_CR1, 243](#)  
[\\_GPIOG\\_CR2, 243](#)  
[\\_GPIOG\\_DDR, 243](#)  
[\\_GPIOG\\_IDR, 243](#)  
[\\_GPIOG\\_ODR, 243](#)  
[\\_GPIOH\\_CR1, 244](#)  
[\\_GPIOH\\_CR2, 244](#)  
[\\_GPIOH\\_DDR, 244](#)  
[\\_GPIOH\\_IDR, 244](#)  
[\\_GPIOH\\_ODR, 245](#)  
[\\_GPIOI\\_CR1, 245](#)  
[\\_GPIOI\\_CR2, 245](#)  
[\\_GPIOI\\_DDR, 245](#)  
[\\_GPIOI\\_IDR, 246](#)  
[\\_GPIOI\\_ODR, 246](#)  
[\\_GPIO\\_CR1\\_RESET\\_VALUE, 233](#)  
[\\_GPIO\\_CR2\\_RESET\\_VALUE, 233](#)  
[\\_GPIO\\_DDR\\_RESET\\_VALUE, 233](#)  
[\\_GPIO\\_ODR\\_RESET\\_VALUE, 233](#)  
[\\_GPIO\\_PIN0, 234](#)  
[\\_GPIO\\_PIN1, 234](#)  
[\\_GPIO\\_PIN2, 234](#)  
[\\_GPIO\\_PIN3, 234](#)  
[\\_GPIO\\_PIN4, 234](#)  
[\\_GPIO\\_PIN5, 235](#)  
[\\_GPIO\\_PIN6, 235](#)  
[\\_GPIO\\_PIN7, 235](#)  
[\\_I2C, 246](#)

[\\_I2C\\_AF](#), [249](#)  
[\\_I2C\\_ACK](#), [246](#)  
[\\_I2C\\_ADD0](#), [246](#)  
[\\_I2C\\_ADD1](#), [247](#)  
[\\_I2C\\_ADD10](#), [247](#)  
[\\_I2C\\_ADD2](#), [247](#)  
[\\_I2C\\_ADD3](#), [247](#)  
[\\_I2C\\_ADD4](#), [247](#)  
[\\_I2C\\_ADD5](#), [248](#)  
[\\_I2C\\_ADD6](#), [248](#)  
[\\_I2C\\_ADD7](#), [248](#)  
[\\_I2C\\_ADD8](#), [248](#)  
[\\_I2C\\_ADD9](#), [248](#)  
[\\_I2C\\_ADDR](#), [249](#)  
[\\_I2C\\_ADDCONF](#), [249](#)  
[\\_I2C\\_ADDMODE](#), [249](#)  
[\\_I2C\\_ADD\\_8\\_9](#), [249](#)  
[\\_I2C\\_ARLO](#), [250](#)  
[\\_I2C\\_BERR](#), [250](#)  
[\\_I2C\\_BTF](#), [250](#)  
[\\_I2C\\_BUSY](#), [250](#)  
[\\_I2C\\_CCR](#), [250](#)  
[\\_I2C\\_CCR0](#), [251](#)  
[\\_I2C\\_CCR1](#), [251](#)  
[\\_I2C\\_CCR2](#), [251](#)  
[\\_I2C\\_CCR3](#), [251](#)  
[\\_I2C\\_CCRH](#), [251](#)  
[\\_I2C\\_CCRL](#), [252](#)  
[\\_I2C\\_CCRH\\_RESET\\_VALUE](#), [252](#)  
[\\_I2C\\_CCRL\\_RESET\\_VALUE](#), [252](#)  
[\\_I2C\\_CR1](#), [252](#)  
[\\_I2C\\_CR1\\_RESET\\_VALUE](#), [252](#)  
[\\_I2C\\_CR2](#), [253](#)  
[\\_I2C\\_CR2\\_RESET\\_VALUE](#), [253](#)  
[\\_I2C\\_DR](#), [253](#)  
[\\_I2C\\_DR\\_RESET\\_VALUE](#), [253](#)  
[\\_I2C\\_DUTY](#), [253](#)  
[\\_I2C\\_ENGC](#), [254](#)  
[\\_I2C\\_FS](#), [256](#)  
[\\_I2C\\_FREQ](#), [254](#)  
[\\_I2C\\_FREQ0](#), [254](#)  
[\\_I2C\\_FREQ1](#), [254](#)  
[\\_I2C\\_FREQ2](#), [254](#)  
[\\_I2C\\_FREQ3](#), [255](#)  
[\\_I2C\\_FREQ4](#), [255](#)  
[\\_I2C\\_FREQ5](#), [255](#)  
[\\_I2C\\_FREQR](#), [255](#)  
[\\_I2C\\_FREQR\\_RESET\\_VALUE](#), [255](#)  
[\\_I2C\\_GENCALL](#), [256](#)  
[\\_I2C\\_ITR](#), [257](#)  
[\\_I2C\\_ITBUFEN](#), [256](#)  
[\\_I2C\\_ITERREN](#), [256](#)  
[\\_I2C\\_ITEVTEN](#), [256](#)  
[\\_I2C\\_ITR\\_RESET\\_VALUE](#), [257](#)  
[\\_I2C\\_MSL](#), [257](#)  
[\\_I2C\\_NOSTRETCH](#), [257](#)  
[\\_I2C\\_OARH](#), [257](#)  
[\\_I2C\\_OARL](#), [258](#)  
[\\_I2C\\_OARH\\_RESET\\_VALUE](#), [258](#)  
[\\_I2C\\_OARL\\_RESET\\_VALUE](#), [258](#)  
[\\_I2C\\_OVR](#), [258](#)  
[\\_I2C\\_PE](#), [258](#)  
[\\_I2C\\_POS](#), [259](#)  
[\\_I2C\\_RXNE](#), [259](#)  
[\\_I2C\\_SB](#), [259](#)  
[\\_I2C\\_SR1](#), [259](#)  
[\\_I2C\\_SR1\\_RESET\\_VALUE](#), [259](#)  
[\\_I2C\\_SR2](#), [260](#)  
[\\_I2C\\_SR2\\_RESET\\_VALUE](#), [260](#)  
[\\_I2C\\_SR3](#), [260](#)  
[\\_I2C\\_SR3\\_RESET\\_VALUE](#), [260](#)  
[\\_I2C\\_START](#), [260](#)  
[\\_I2C\\_STOP](#), [261](#)  
[\\_I2C\\_STOPF](#), [261](#)  
[\\_I2C\\_SWRST](#), [261](#)  
[\\_I2C\\_TRA](#), [261](#)  
[\\_I2C\\_TRISE](#), [261](#)  
[\\_I2C\\_TRISE0](#), [262](#)  
[\\_I2C\\_TRISE1](#), [262](#)  
[\\_I2C\\_TRISE2](#), [262](#)  
[\\_I2C\\_TRISE3](#), [262](#)  
[\\_I2C\\_TRISE4](#), [262](#)  
[\\_I2C\\_TRISE5](#), [263](#)  
[\\_I2C\\_TRISER](#), [263](#)  
[\\_I2C\\_TRISER\\_RESET\\_VALUE](#), [263](#)  
[\\_I2C\\_TXE](#), [263](#)  
[\\_I2C\\_WUFH](#), [263](#)  
[\\_ITC](#), [264](#)  
[\\_ITC\\_SPR1](#), [264](#)  
[\\_ITC\\_SPR1\\_RESET\\_VALUE](#), [264](#)  
[\\_ITC\\_SPR2](#), [264](#)  
[\\_ITC\\_SPR2\\_RESET\\_VALUE](#), [264](#)  
[\\_ITC\\_SPR3](#), [265](#)  
[\\_ITC\\_SPR3\\_RESET\\_VALUE](#), [265](#)  
[\\_ITC\\_SPR4](#), [265](#)  
[\\_ITC\\_SPR4\\_RESET\\_VALUE](#), [265](#)  
[\\_ITC\\_SPR5](#), [265](#)  
[\\_ITC\\_SPR5\\_RESET\\_VALUE](#), [266](#)  
[\\_ITC\\_SPR6](#), [266](#)  
[\\_ITC\\_SPR6\\_RESET\\_VALUE](#), [266](#)  
[\\_ITC\\_SPR7](#), [266](#)  
[\\_ITC\\_SPR7\\_RESET\\_VALUE](#), [266](#)  
[\\_ITC\\_SPR8](#), [267](#)  
[\\_ITC\\_SPR8\\_RESET\\_VALUE](#), [267](#)  
[\\_ITC\\_VECT10SPR](#), [267](#)  
[\\_ITC\\_VECT10SPR0](#), [267](#)  
[\\_ITC\\_VECT10SPR1](#), [267](#)  
[\\_ITC\\_VECT11SPR](#), [268](#)  
[\\_ITC\\_VECT11SPR0](#), [268](#)  
[\\_ITC\\_VECT11SPR1](#), [268](#)  
[\\_ITC\\_VECT12SPR](#), [268](#)  
[\\_ITC\\_VECT12SPR0](#), [268](#)  
[\\_ITC\\_VECT12SPR1](#), [269](#)  
[\\_ITC\\_VECT13SPR](#), [269](#)  
[\\_ITC\\_VECT13SPR0](#), [269](#)  
[\\_ITC\\_VECT13SPR1](#), [269](#)

[\\_ITC\\_VECT14SPR](#), 269  
[\\_ITC\\_VECT14SPR0](#), 270  
[\\_ITC\\_VECT14SPR1](#), 270  
[\\_ITC\\_VECT15SPR](#), 270  
[\\_ITC\\_VECT15SPR0](#), 270  
[\\_ITC\\_VECT15SPR1](#), 270  
[\\_ITC\\_VECT16SPR](#), 271  
[\\_ITC\\_VECT16SPR0](#), 271  
[\\_ITC\\_VECT16SPR1](#), 271  
[\\_ITC\\_VECT17SPR](#), 271  
[\\_ITC\\_VECT17SPR0](#), 271  
[\\_ITC\\_VECT17SPR1](#), 272  
[\\_ITC\\_VECT18SPR](#), 272  
[\\_ITC\\_VECT18SPR0](#), 272  
[\\_ITC\\_VECT18SPR1](#), 272  
[\\_ITC\\_VECT19SPR](#), 272  
[\\_ITC\\_VECT19SPR0](#), 273  
[\\_ITC\\_VECT19SPR1](#), 273  
[\\_ITC\\_VECT1SPR](#), 273  
[\\_ITC\\_VECT1SPR0](#), 273  
[\\_ITC\\_VECT1SPR1](#), 273  
[\\_ITC\\_VECT20SPR](#), 274  
[\\_ITC\\_VECT20SPR0](#), 274  
[\\_ITC\\_VECT20SPR1](#), 274  
[\\_ITC\\_VECT21SPR](#), 274  
[\\_ITC\\_VECT21SPR0](#), 274  
[\\_ITC\\_VECT21SPR1](#), 275  
[\\_ITC\\_VECT22SPR](#), 275  
[\\_ITC\\_VECT22SPR0](#), 275  
[\\_ITC\\_VECT22SPR1](#), 275  
[\\_ITC\\_VECT23SPR](#), 275  
[\\_ITC\\_VECT23SPR0](#), 276  
[\\_ITC\\_VECT23SPR1](#), 276  
[\\_ITC\\_VECT24SPR](#), 276  
[\\_ITC\\_VECT24SPR0](#), 276  
[\\_ITC\\_VECT24SPR1](#), 276  
[\\_ITC\\_VECT25SPR](#), 277  
[\\_ITC\\_VECT25SPR0](#), 277  
[\\_ITC\\_VECT25SPR1](#), 277  
[\\_ITC\\_VECT26SPR](#), 277  
[\\_ITC\\_VECT26SPR0](#), 277  
[\\_ITC\\_VECT26SPR1](#), 278  
[\\_ITC\\_VECT27SPR](#), 278  
[\\_ITC\\_VECT27SPR0](#), 278  
[\\_ITC\\_VECT27SPR1](#), 278  
[\\_ITC\\_VECT28SPR](#), 278  
[\\_ITC\\_VECT28SPR0](#), 279  
[\\_ITC\\_VECT28SPR1](#), 279  
[\\_ITC\\_VECT29SPR](#), 279  
[\\_ITC\\_VECT29SPR0](#), 279  
[\\_ITC\\_VECT29SPR1](#), 279  
[\\_ITC\\_VECT2SPR](#), 280  
[\\_ITC\\_VECT2SPR0](#), 280  
[\\_ITC\\_VECT2SPR1](#), 280  
[\\_ITC\\_VECT3SPR](#), 280  
[\\_ITC\\_VECT3SPR0](#), 280  
[\\_ITC\\_VECT3SPR1](#), 281  
[\\_ITC\\_VECT4SPR](#), 281  
[\\_ITC\\_VECT4SPR0](#), 281  
[\\_ITC\\_VECT4SPR1](#), 281  
[\\_ITC\\_VECT5SPR](#), 281  
[\\_ITC\\_VECT5SPR0](#), 282  
[\\_ITC\\_VECT5SPR1](#), 282  
[\\_ITC\\_VECT6SPR](#), 282  
[\\_ITC\\_VECT6SPR0](#), 282  
[\\_ITC\\_VECT6SPR1](#), 282  
[\\_ITC\\_VECT7SPR](#), 283  
[\\_ITC\\_VECT7SPR0](#), 283  
[\\_ITC\\_VECT7SPR1](#), 283  
[\\_ITC\\_VECT8SPR](#), 283  
[\\_ITC\\_VECT8SPR0](#), 283  
[\\_ITC\\_VECT8SPR1](#), 284  
[\\_ITC\\_VECT9SPR](#), 284  
[\\_ITC\\_VECT9SPR0](#), 284  
[\\_ITC\\_VECT9SPR1](#), 284  
[\\_IWDG](#), 284  
[\\_IWDG\\_KR](#), 285  
[\\_IWDG\\_KEY\\_ACCESS](#), 285  
[\\_IWDG\\_KEY\\_ENABLE](#), 285  
[\\_IWDG\\_KEY\\_REFRESH](#), 285  
[\\_IWDG\\_PR](#), 285  
[\\_IWDG\\_PRE](#), 286  
[\\_IWDG\\_PRE0](#), 286  
[\\_IWDG\\_PRE1](#), 286  
[\\_IWDG\\_PRE2](#), 286  
[\\_IWDG\\_PR\\_RESET\\_VALUE](#), 286  
[\\_IWDG\\_RLR](#), 287  
[\\_IWDG\\_RLR\\_RESET\\_VALUE](#), 287  
[\\_RST](#), 287  
[\\_RST\\_EMCF](#), 287  
[\\_RST\\_ILLOPF](#), 287  
[\\_RST\\_IWDGF](#), 288  
[\\_RST\\_SR](#), 288  
[\\_RST\\_SWIMF](#), 288  
[\\_RST\\_WWDGF](#), 288  
[\\_SFR](#), 288  
[\\_SPI](#), 289  
[\\_SPI\\_BR](#), 289  
[\\_SPI\\_BDM](#), 289  
[\\_SPI\\_BDOE](#), 289  
[\\_SPI\\_BR0](#), 289  
[\\_SPI\\_BR1](#), 290  
[\\_SPI\\_BR2](#), 290  
[\\_SPI\\_BSY](#), 290  
[\\_SPI\\_CPHA](#), 290  
[\\_SPI\\_CPOL](#), 290  
[\\_SPI\\_CR1](#), 291  
[\\_SPI\\_CR1\\_RESET\\_VALUE](#), 291  
[\\_SPI\\_CR2](#), 291  
[\\_SPI\\_CR2\\_RESET\\_VALUE](#), 291  
[\\_SPI\\_CRCEN](#), 291  
[\\_SPI\\_CRCERR](#), 292  
[\\_SPI\\_CRCNEXT](#), 292  
[\\_SPI\\_CRCPR](#), 292  
[\\_SPI\\_CRCPR\\_RESET\\_VALUE](#), 292  
[\\_SPI\\_DR](#), 292

[\\_SPI\\_DR\\_RESET\\_VALUE, 293](#)  
[\\_SPI\\_ERRIE, 293](#)  
[\\_SPI\\_ICR, 293](#)  
[\\_SPI\\_ICR\\_RESET\\_VALUE, 293](#)  
[\\_SPI\\_LSBFIRST, 293](#)  
[\\_SPI\\_MODF, 294](#)  
[\\_SPI\\_MSTR, 294](#)  
[\\_SPI\\_OVR, 294](#)  
[\\_SPI\\_RXCR, 294](#)  
[\\_SPI\\_RXCR\\_RESET\\_VALUE, 294](#)  
[\\_SPI\\_RXIE, 295](#)  
[\\_SPI\\_RXNE, 295](#)  
[\\_SPI\\_RXONLY, 295](#)  
[\\_SPI\\_SR, 295](#)  
[\\_SPI\\_SPE, 295](#)  
[\\_SPI\\_SR\\_RESET\\_VALUE, 296](#)  
[\\_SPI\\_SSI, 296](#)  
[\\_SPI\\_SSM, 296](#)  
[\\_SPI\\_TXE, 297](#)  
[\\_SPI\\_TXCR, 296](#)  
[\\_SPI\\_TXCR\\_RESET\\_VALUE, 296](#)  
[\\_SPI\\_TXIE, 297](#)  
[\\_SPI\\_WKIE, 297](#)  
[\\_SPI\\_WKUP, 297](#)  
[\\_TIM1, 297](#)  
[\\_TIM1\\_AOE, 298](#)  
[\\_TIM1\\_ARPE, 298](#)  
[\\_TIM1\\_ARRH, 298](#)  
[\\_TIM1\\_ARRL, 298](#)  
[\\_TIM1\\_ARRH\\_RESET\\_VALUE, 298](#)  
[\\_TIM1\\_ARRL\\_RESET\\_VALUE, 299](#)  
[\\_TIM1\\_BG, 299](#)  
[\\_TIM1\\_BIE, 299](#)  
[\\_TIM1\\_BIF, 299](#)  
[\\_TIM1\\_BKE, 299](#)  
[\\_TIM1\\_BKP, 300](#)  
[\\_TIM1\\_BKR, 300](#)  
[\\_TIM1\\_BKR\\_RESET\\_VALUE, 300](#)  
[\\_TIM1\\_CC1E, 300](#)  
[\\_TIM1\\_CC1G, 300](#)  
[\\_TIM1\\_CC1P, 302](#)  
[\\_TIM1\\_CC1S, 302](#)  
[\\_TIM1\\_CC1IE, 301](#)  
[\\_TIM1\\_CC1IF, 301](#)  
[\\_TIM1\\_CC1NE, 301](#)  
[\\_TIM1\\_CC1NP, 301](#)  
[\\_TIM1\\_CC1OF, 301](#)  
[\\_TIM1\\_CC1S0, 302](#)  
[\\_TIM1\\_CC1S1, 302](#)  
[\\_TIM1\\_CC2E, 302](#)  
[\\_TIM1\\_CC2G, 303](#)  
[\\_TIM1\\_CC2P, 304](#)  
[\\_TIM1\\_CC2S, 304](#)  
[\\_TIM1\\_CC2IE, 303](#)  
[\\_TIM1\\_CC2IF, 303](#)  
[\\_TIM1\\_CC2NE, 303](#)  
[\\_TIM1\\_CC2NP, 303](#)  
[\\_TIM1\\_CC2OF, 304](#)  
[\\_TIM1\\_CC2S0, 304](#)  
[\\_TIM1\\_CC2S1, 304](#)  
[\\_TIM1\\_CC3E, 305](#)  
[\\_TIM1\\_CC3G, 305](#)  
[\\_TIM1\\_CC3P, 306](#)  
[\\_TIM1\\_CC3S, 306](#)  
[\\_TIM1\\_CC3IE, 305](#)  
[\\_TIM1\\_CC3IF, 305](#)  
[\\_TIM1\\_CC3NE, 305](#)  
[\\_TIM1\\_CC3NP, 306](#)  
[\\_TIM1\\_CC3OF, 306](#)  
[\\_TIM1\\_CC3S0, 306](#)  
[\\_TIM1\\_CC3S1, 307](#)  
[\\_TIM1\\_CC4E, 307](#)  
[\\_TIM1\\_CC4G, 307](#)  
[\\_TIM1\\_CC4P, 308](#)  
[\\_TIM1\\_CC4S, 308](#)  
[\\_TIM1\\_CC4IE, 307](#)  
[\\_TIM1\\_CC4IF, 307](#)  
[\\_TIM1\\_CC4OF, 308](#)  
[\\_TIM1\\_CC4S0, 308](#)  
[\\_TIM1\\_CC4S1, 308](#)  
[\\_TIM1\\_CCER1, 309](#)  
[\\_TIM1\\_CCER1\\_RESET\\_VALUE, 309](#)  
[\\_TIM1\\_CCER2, 309](#)  
[\\_TIM1\\_CCER2\\_RESET\\_VALUE, 309](#)  
[\\_TIM1\\_CCMR1, 309](#)  
[\\_TIM1\\_CCMR1\\_RESET\\_VALUE, 310](#)  
[\\_TIM1\\_CCMR2, 310](#)  
[\\_TIM1\\_CCMR2\\_RESET\\_VALUE, 310](#)  
[\\_TIM1\\_CCMR3, 310](#)  
[\\_TIM1\\_CCMR3\\_RESET\\_VALUE, 310](#)  
[\\_TIM1\\_CCMR4, 311](#)  
[\\_TIM1\\_CCMR4\\_RESET\\_VALUE, 311](#)  
[\\_TIM1\\_CCPC, 311](#)  
[\\_TIM1\\_CCR1H, 311](#)  
[\\_TIM1\\_CCR1L, 312](#)  
[\\_TIM1\\_CCR1H\\_RESET\\_VALUE, 311](#)  
[\\_TIM1\\_CCR1L\\_RESET\\_VALUE, 312](#)  
[\\_TIM1\\_CCR2H, 312](#)  
[\\_TIM1\\_CCR2L, 312](#)  
[\\_TIM1\\_CCR2H\\_RESET\\_VALUE, 312](#)  
[\\_TIM1\\_CCR2L\\_RESET\\_VALUE, 313](#)  
[\\_TIM1\\_CCR3H, 313](#)  
[\\_TIM1\\_CCR3L, 313](#)  
[\\_TIM1\\_CCR3H\\_RESET\\_VALUE, 313](#)  
[\\_TIM1\\_CCR3L\\_RESET\\_VALUE, 313](#)  
[\\_TIM1\\_CCR4H, 314](#)  
[\\_TIM1\\_CCR4L, 314](#)  
[\\_TIM1\\_CCR4H\\_RESET\\_VALUE, 314](#)  
[\\_TIM1\\_CCR4L\\_RESET\\_VALUE, 314](#)  
[\\_TIM1\\_CEN, 314](#)  
[\\_TIM1\\_CMS, 315](#)  
[\\_TIM1\\_CMS0, 315](#)  
[\\_TIM1\\_CMS1, 315](#)  
[\\_TIM1\\_CNTRH, 315](#)  
[\\_TIM1\\_CNTRL, 316](#)  
[\\_TIM1\\_CNTRH\\_RESET\\_VALUE, 315](#)

[\\_TIM1\\_CNTRL\\_RESET\\_VALUE, 316](#)  
[\\_TIM1\\_COMG, 316](#)  
[\\_TIM1\\_COMS, 317](#)  
[\\_TIM1\\_COMIE, 316](#)  
[\\_TIM1\\_COMIF, 316](#)  
[\\_TIM1\\_CR1, 317](#)  
[\\_TIM1\\_CR1\\_RESET\\_VALUE, 317](#)  
[\\_TIM1\\_CR2, 317](#)  
[\\_TIM1\\_CR2\\_RESET\\_VALUE, 317](#)  
[\\_TIM1\\_DIR, 318](#)  
[\\_TIM1\\_DTR, 318](#)  
[\\_TIM1\\_DTR\\_RESET\\_VALUE, 318](#)  
[\\_TIM1\\_ECE, 318](#)  
[\\_TIM1\\_EGR, 318](#)  
[\\_TIM1\\_EGR\\_RESET\\_VALUE, 319](#)  
[\\_TIM1\\_ETF, 319](#)  
[\\_TIM1\\_ETP, 320](#)  
[\\_TIM1\\_ETR, 321](#)  
[\\_TIM1\\_ETF0, 319](#)  
[\\_TIM1\\_ETF1, 319](#)  
[\\_TIM1\\_ETF2, 319](#)  
[\\_TIM1\\_ETF3, 320](#)  
[\\_TIM1\\_ETPS, 320](#)  
[\\_TIM1\\_ETPS0, 320](#)  
[\\_TIM1\\_ETPS1, 320](#)  
[\\_TIM1\\_ETR\\_RESET\\_VALUE, 321](#)  
[\\_TIM1\\_IC1F, 321](#)  
[\\_TIM1\\_IC1F0, 321](#)  
[\\_TIM1\\_IC1F1, 321](#)  
[\\_TIM1\\_IC1F2, 322](#)  
[\\_TIM1\\_IC1F3, 322](#)  
[\\_TIM1\\_IC1PSC, 322](#)  
[\\_TIM1\\_IC1PSC0, 322](#)  
[\\_TIM1\\_IC1PSC1, 322](#)  
[\\_TIM1\\_IC2F, 323](#)  
[\\_TIM1\\_IC2F0, 323](#)  
[\\_TIM1\\_IC2F1, 323](#)  
[\\_TIM1\\_IC2F2, 323](#)  
[\\_TIM1\\_IC2F3, 323](#)  
[\\_TIM1\\_IC2PSC, 324](#)  
[\\_TIM1\\_IC2PSC0, 324](#)  
[\\_TIM1\\_IC2PSC1, 324](#)  
[\\_TIM1\\_IC3F, 324](#)  
[\\_TIM1\\_IC3F0, 324](#)  
[\\_TIM1\\_IC3F1, 325](#)  
[\\_TIM1\\_IC3F2, 325](#)  
[\\_TIM1\\_IC3F3, 325](#)  
[\\_TIM1\\_IC3PSC, 325](#)  
[\\_TIM1\\_IC3PSC0, 325](#)  
[\\_TIM1\\_IC3PSC1, 326](#)  
[\\_TIM1\\_IC4F, 326](#)  
[\\_TIM1\\_IC4F0, 326](#)  
[\\_TIM1\\_IC4F1, 326](#)  
[\\_TIM1\\_IC4F2, 326](#)  
[\\_TIM1\\_IC4F3, 327](#)  
[\\_TIM1\\_IC4PSC, 327](#)  
[\\_TIM1\\_IC4PSC0, 327](#)  
[\\_TIM1\\_IC4PSC1, 327](#)  
[\\_TIM1\\_IER, 327](#)  
[\\_TIM1\\_IER\\_RESET\\_VALUE, 328](#)  
[\\_TIM1\\_LOCK, 328](#)  
[\\_TIM1\\_LOCK0, 328](#)  
[\\_TIM1\\_LOCK1, 328](#)  
[\\_TIM1\\_MMS, 328](#)  
[\\_TIM1\\_MMS0, 329](#)  
[\\_TIM1\\_MMS1, 329](#)  
[\\_TIM1\\_MMS2, 329](#)  
[\\_TIM1\\_MOE, 329](#)  
[\\_TIM1\\_MSM, 329](#)  
[\\_TIM1\\_OC1M, 330](#)  
[\\_TIM1\\_OC1CE, 330](#)  
[\\_TIM1\\_OC1FE, 330](#)  
[\\_TIM1\\_OC1M0, 330](#)  
[\\_TIM1\\_OC1M1, 330](#)  
[\\_TIM1\\_OC1M2, 331](#)  
[\\_TIM1\\_OC1PE, 331](#)  
[\\_TIM1\\_OC2M, 331](#)  
[\\_TIM1\\_OC2CE, 331](#)  
[\\_TIM1\\_OC2FE, 331](#)  
[\\_TIM1\\_OC2M0, 332](#)  
[\\_TIM1\\_OC2M1, 332](#)  
[\\_TIM1\\_OC2M2, 332](#)  
[\\_TIM1\\_OC2PE, 332](#)  
[\\_TIM1\\_OC3M, 333](#)  
[\\_TIM1\\_OC3CE, 332](#)  
[\\_TIM1\\_OC3FE, 333](#)  
[\\_TIM1\\_OC3M0, 333](#)  
[\\_TIM1\\_OC3M1, 333](#)  
[\\_TIM1\\_OC3M2, 333](#)  
[\\_TIM1\\_OC3PE, 334](#)  
[\\_TIM1\\_OC4M, 334](#)  
[\\_TIM1\\_OC4CE, 334](#)  
[\\_TIM1\\_OC4FE, 334](#)  
[\\_TIM1\\_OC4M0, 334](#)  
[\\_TIM1\\_OC4M1, 335](#)  
[\\_TIM1\\_OC4M2, 335](#)  
[\\_TIM1\\_OC4PE, 335](#)  
[\\_TIM1\\_OIS1, 335](#)  
[\\_TIM1\\_OIS1N, 335](#)  
[\\_TIM1\\_OIS2, 336](#)  
[\\_TIM1\\_OIS2N, 336](#)  
[\\_TIM1\\_OIS3, 336](#)  
[\\_TIM1\\_OIS3N, 336](#)  
[\\_TIM1\\_OIS4, 336](#)  
[\\_TIM1\\_OISR, 337](#)  
[\\_TIM1\\_OISR\\_RESET\\_VALUE, 337](#)  
[\\_TIM1\\_OPM, 337](#)  
[\\_TIM1\\_OSSI, 337](#)  
[\\_TIM1\\_OSSR, 337](#)  
[\\_TIM1\\_PSCRH, 338](#)  
[\\_TIM1\\_PSCRL, 338](#)  
[\\_TIM1\\_PSCRH\\_RESET\\_VALUE, 338](#)  
[\\_TIM1\\_PSCRL\\_RESET\\_VALUE, 338](#)  
[\\_TIM1\\_RCR, 338](#)  
[\\_TIM1\\_RCR\\_RESET\\_VALUE, 339](#)  
[\\_TIM1\\_SMS, 339](#)



[\\_TIM1\\_SMCR, 339](#)  
[\\_TIM1\\_SMCR\\_RESET\\_VALUE, 339](#)  
[\\_TIM1\\_SMS0, 339](#)  
[\\_TIM1\\_SMS1, 340](#)  
[\\_TIM1\\_SMS2, 340](#)  
[\\_TIM1\\_SR1, 340](#)  
[\\_TIM1\\_SR1\\_RESET\\_VALUE, 340](#)  
[\\_TIM1\\_SR2, 340](#)  
[\\_TIM1\\_SR2\\_RESET\\_VALUE, 341](#)  
[\\_TIM1\\_TG, 341](#)  
[\\_TIM1\\_TS, 341](#)  
[\\_TIM1\\_TIE, 341](#)  
[\\_TIM1\\_TIF, 341](#)  
[\\_TIM1\\_TS0, 342](#)  
[\\_TIM1\\_TS1, 342](#)  
[\\_TIM1\\_TS2, 342](#)  
[\\_TIM1\\_UG, 342](#)  
[\\_TIM1\\_UDIS, 342](#)  
[\\_TIM1\\_UIE, 343](#)  
[\\_TIM1\\_UIF, 343](#)  
[\\_TIM1\\_URS, 343](#)  
[\\_TIM2, 343](#)  
[\\_TIM2\\_ARPE, 343](#)  
[\\_TIM2\\_ARRH, 344](#)  
[\\_TIM2\\_ARRL, 344](#)  
[\\_TIM2\\_ARRH\\_RESET\\_VALUE, 344](#)  
[\\_TIM2\\_ARRL\\_RESET\\_VALUE, 344](#)  
[\\_TIM2\\_CC1E, 344](#)  
[\\_TIM2\\_CC1G, 345](#)  
[\\_TIM2\\_CC1P, 345](#)  
[\\_TIM2\\_CC1S, 346](#)  
[\\_TIM2\\_CC1IE, 345](#)  
[\\_TIM2\\_CC1IF, 345](#)  
[\\_TIM2\\_CC1OF, 345](#)  
[\\_TIM2\\_CC1S0, 346](#)  
[\\_TIM2\\_CC1S1, 346](#)  
[\\_TIM2\\_CC2E, 346](#)  
[\\_TIM2\\_CC2G, 346](#)  
[\\_TIM2\\_CC2P, 347](#)  
[\\_TIM2\\_CC2S, 347](#)  
[\\_TIM2\\_CC2IE, 347](#)  
[\\_TIM2\\_CC2IF, 347](#)  
[\\_TIM2\\_CC2OF, 347](#)  
[\\_TIM2\\_CC2S0, 348](#)  
[\\_TIM2\\_CC2S1, 348](#)  
[\\_TIM2\\_CC3E, 348](#)  
[\\_TIM2\\_CC3G, 348](#)  
[\\_TIM2\\_CC3P, 349](#)  
[\\_TIM2\\_CC3S, 349](#)  
[\\_TIM2\\_CC3IE, 348](#)  
[\\_TIM2\\_CC3IF, 349](#)  
[\\_TIM2\\_CC3OF, 349](#)  
[\\_TIM2\\_CC3S0, 349](#)  
[\\_TIM2\\_CC3S1, 350](#)  
[\\_TIM2\\_CCER1, 350](#)  
[\\_TIM2\\_CCER1\\_RESET\\_VALUE, 350](#)  
[\\_TIM2\\_CCER2, 350](#)  
[\\_TIM2\\_CCER2\\_RESET\\_VALUE, 350](#)  
[\\_TIM2\\_CCMR1, 351](#)  
[\\_TIM2\\_CCMR1\\_RESET\\_VALUE, 351](#)  
[\\_TIM2\\_CCMR2, 351](#)  
[\\_TIM2\\_CCMR2\\_RESET\\_VALUE, 351](#)  
[\\_TIM2\\_CCMR3, 351](#)  
[\\_TIM2\\_CCMR3\\_RESET\\_VALUE, 352](#)  
[\\_TIM2\\_CCR1H, 352](#)  
[\\_TIM2\\_CCR1L, 352](#)  
[\\_TIM2\\_CCR1H\\_RESET\\_VALUE, 352](#)  
[\\_TIM2\\_CCR1L\\_RESET\\_VALUE, 352](#)  
[\\_TIM2\\_CCR2H, 353](#)  
[\\_TIM2\\_CCR2L, 353](#)  
[\\_TIM2\\_CCR2H\\_RESET\\_VALUE, 353](#)  
[\\_TIM2\\_CCR2L\\_RESET\\_VALUE, 353](#)  
[\\_TIM2\\_CCR3H, 353](#)  
[\\_TIM2\\_CCR3L, 354](#)  
[\\_TIM2\\_CCR3H\\_RESET\\_VALUE, 354](#)  
[\\_TIM2\\_CCR3L\\_RESET\\_VALUE, 354](#)  
[\\_TIM2\\_CEN, 354](#)  
[\\_TIM2\\_CNTRH, 354](#)  
[\\_TIM2\\_CNTRL, 355](#)  
[\\_TIM2\\_CNTRH\\_RESET\\_VALUE, 355](#)  
[\\_TIM2\\_CNTRL\\_RESET\\_VALUE, 355](#)  
[\\_TIM2\\_CR1, 355](#)  
[\\_TIM2\\_CR1\\_RESET\\_VALUE, 355](#)  
[\\_TIM2\\_EGR, 356](#)  
[\\_TIM2\\_EGR\\_RESET\\_VALUE, 356](#)  
[\\_TIM2\\_IC1F, 356](#)  
[\\_TIM2\\_IC1F0, 356](#)  
[\\_TIM2\\_IC1F1, 356](#)  
[\\_TIM2\\_IC1F2, 357](#)  
[\\_TIM2\\_IC1F3, 357](#)  
[\\_TIM2\\_IC1PSC, 357](#)  
[\\_TIM2\\_IC1PSC0, 357](#)  
[\\_TIM2\\_IC1PSC1, 357](#)  
[\\_TIM2\\_IC2F, 358](#)  
[\\_TIM2\\_IC2F0, 358](#)  
[\\_TIM2\\_IC2F1, 358](#)  
[\\_TIM2\\_IC2F2, 358](#)  
[\\_TIM2\\_IC2F3, 358](#)  
[\\_TIM2\\_IC2PSC, 359](#)  
[\\_TIM2\\_IC2PSC0, 359](#)  
[\\_TIM2\\_IC2PSC1, 359](#)  
[\\_TIM2\\_IC3F, 359](#)  
[\\_TIM2\\_IC3F0, 359](#)  
[\\_TIM2\\_IC3F1, 360](#)  
[\\_TIM2\\_IC3F2, 360](#)  
[\\_TIM2\\_IC3F3, 360](#)  
[\\_TIM2\\_IC3PSC, 360](#)  
[\\_TIM2\\_IC3PSC0, 360](#)  
[\\_TIM2\\_IC3PSC1, 361](#)  
[\\_TIM2\\_IER, 361](#)  
[\\_TIM2\\_IER\\_RESET\\_VALUE, 361](#)  
[\\_TIM2\\_OC1M, 361](#)  
[\\_TIM2\\_OC1M0, 361](#)  
[\\_TIM2\\_OC1M1, 362](#)  
[\\_TIM2\\_OC1M2, 362](#)  
[\\_TIM2\\_OC1PE, 362](#)

[\\_TIM2\\_OC2M](#), [362](#)  
[\\_TIM2\\_OC2M0](#), [362](#)  
[\\_TIM2\\_OC2M1](#), [363](#)  
[\\_TIM2\\_OC2M2](#), [363](#)  
[\\_TIM2\\_OC2PE](#), [363](#)  
[\\_TIM2\\_OC3M](#), [363](#)  
[\\_TIM2\\_OC3M0](#), [363](#)  
[\\_TIM2\\_OC3M1](#), [364](#)  
[\\_TIM2\\_OC3M2](#), [364](#)  
[\\_TIM2\\_OC3PE](#), [364](#)  
[\\_TIM2\\_OPM](#), [364](#)  
[\\_TIM2\\_PSC](#), [364](#)  
[\\_TIM2\\_PSC0](#), [365](#)  
[\\_TIM2\\_PSC1](#), [365](#)  
[\\_TIM2\\_PSC2](#), [365](#)  
[\\_TIM2\\_PSC3](#), [365](#)  
[\\_TIM2\\_PSCR](#), [365](#)  
[\\_TIM2\\_PSCR\\_RESET\\_VALUE](#), [366](#)  
[\\_TIM2\\_SR1](#), [366](#)  
[\\_TIM2\\_SR1\\_RESET\\_VALUE](#), [366](#)  
[\\_TIM2\\_SR2](#), [366](#)  
[\\_TIM2\\_SR2\\_RESET\\_VALUE](#), [366](#)  
[\\_TIM2\\_UG](#), [367](#)  
[\\_TIM2\\_UDIS](#), [367](#)  
[\\_TIM2\\_UIE](#), [367](#)  
[\\_TIM2\\_UIF](#), [367](#)  
[\\_TIM2\\_URS](#), [367](#)  
[\\_TIM3](#), [368](#)  
[\\_TIM3\\_ARPE](#), [368](#)  
[\\_TIM3\\_ARRH](#), [368](#)  
[\\_TIM3\\_ARRL](#), [368](#)  
[\\_TIM3\\_ARRH\\_RESET\\_VALUE](#), [368](#)  
[\\_TIM3\\_ARRL\\_RESET\\_VALUE](#), [369](#)  
[\\_TIM3\\_CC1E](#), [369](#)  
[\\_TIM3\\_CC1G](#), [369](#)  
[\\_TIM3\\_CC1P](#), [370](#)  
[\\_TIM3\\_CC1S](#), [370](#)  
[\\_TIM3\\_CC1IE](#), [369](#)  
[\\_TIM3\\_CC1IF](#), [369](#)  
[\\_TIM3\\_CC1OF](#), [370](#)  
[\\_TIM3\\_CC1S0](#), [370](#)  
[\\_TIM3\\_CC1S1](#), [370](#)  
[\\_TIM3\\_CC2E](#), [371](#)  
[\\_TIM3\\_CC2G](#), [371](#)  
[\\_TIM3\\_CC2P](#), [372](#)  
[\\_TIM3\\_CC2S](#), [372](#)  
[\\_TIM3\\_CC2IE](#), [371](#)  
[\\_TIM3\\_CC2IF](#), [371](#)  
[\\_TIM3\\_CC2OF](#), [371](#)  
[\\_TIM3\\_CC2S0](#), [372](#)  
[\\_TIM3\\_CC2S1](#), [372](#)  
[\\_TIM3\\_CCER1](#), [372](#)  
[\\_TIM3\\_CCER1\\_RESET\\_VALUE](#), [373](#)  
[\\_TIM3\\_CCMR1](#), [373](#)  
[\\_TIM3\\_CCMR1\\_RESET\\_VALUE](#), [373](#)  
[\\_TIM3\\_CCMR2](#), [373](#)  
[\\_TIM3\\_CCMR2\\_RESET\\_VALUE](#), [373](#)  
[\\_TIM3\\_CCR1H](#), [374](#)  
[\\_TIM3\\_CCR1L](#), [374](#)  
[\\_TIM3\\_CCR1H\\_RESET\\_VALUE](#), [374](#)  
[\\_TIM3\\_CCR1L\\_RESET\\_VALUE](#), [374](#)  
[\\_TIM3\\_CCR2H](#), [374](#)  
[\\_TIM3\\_CCR2L](#), [375](#)  
[\\_TIM3\\_CCR2H\\_RESET\\_VALUE](#), [375](#)  
[\\_TIM3\\_CCR2L\\_RESET\\_VALUE](#), [375](#)  
[\\_TIM3\\_CEN](#), [375](#)  
[\\_TIM3\\_CNTRH](#), [375](#)  
[\\_TIM3\\_CNTRL](#), [376](#)  
[\\_TIM3\\_CNTRH\\_RESET\\_VALUE](#), [376](#)  
[\\_TIM3\\_CNTRL\\_RESET\\_VALUE](#), [376](#)  
[\\_TIM3\\_CR1](#), [376](#)  
[\\_TIM3\\_CR1\\_RESET\\_VALUE](#), [376](#)  
[\\_TIM3\\_EGR](#), [377](#)  
[\\_TIM3\\_EGR\\_RESET\\_VALUE](#), [377](#)  
[\\_TIM3\\_IC1F](#), [377](#)  
[\\_TIM3\\_IC1F0](#), [377](#)  
[\\_TIM3\\_IC1F1](#), [377](#)  
[\\_TIM3\\_IC1F2](#), [378](#)  
[\\_TIM3\\_IC1F3](#), [378](#)  
[\\_TIM3\\_IC1PSC](#), [378](#)  
[\\_TIM3\\_IC1PSC0](#), [378](#)  
[\\_TIM3\\_IC1PSC1](#), [378](#)  
[\\_TIM3\\_IC2F](#), [379](#)  
[\\_TIM3\\_IC2F0](#), [379](#)  
[\\_TIM3\\_IC2F1](#), [379](#)  
[\\_TIM3\\_IC2F2](#), [379](#)  
[\\_TIM3\\_IC2F3](#), [379](#)  
[\\_TIM3\\_IC2PSC](#), [380](#)  
[\\_TIM3\\_IC2PSC0](#), [380](#)  
[\\_TIM3\\_IC2PSC1](#), [380](#)  
[\\_TIM3\\_IER](#), [380](#)  
[\\_TIM3\\_IER\\_RESET\\_VALUE](#), [380](#)  
[\\_TIM3\\_OC1M](#), [381](#)  
[\\_TIM3\\_OC1M0](#), [381](#)  
[\\_TIM3\\_OC1M1](#), [381](#)  
[\\_TIM3\\_OC1M2](#), [381](#)  
[\\_TIM3\\_OC1PE](#), [381](#)  
[\\_TIM3\\_OC2M](#), [382](#)  
[\\_TIM3\\_OC2M0](#), [382](#)  
[\\_TIM3\\_OC2M1](#), [382](#)  
[\\_TIM3\\_OC2M2](#), [382](#)  
[\\_TIM3\\_OC2PE](#), [382](#)  
[\\_TIM3\\_OPM](#), [383](#)  
[\\_TIM3\\_PSC](#), [383](#)  
[\\_TIM3\\_PSC0](#), [383](#)  
[\\_TIM3\\_PSC1](#), [383](#)  
[\\_TIM3\\_PSC2](#), [383](#)  
[\\_TIM3\\_PSC3](#), [384](#)  
[\\_TIM3\\_PSCR](#), [384](#)  
[\\_TIM3\\_PSCR\\_RESET\\_VALUE](#), [384](#)  
[\\_TIM3\\_SR1](#), [384](#)  
[\\_TIM3\\_SR1\\_RESET\\_VALUE](#), [384](#)  
[\\_TIM3\\_SR2](#), [385](#)  
[\\_TIM3\\_SR2\\_RESET\\_VALUE](#), [385](#)  
[\\_TIM3\\_UG](#), [385](#)  
[\\_TIM3\\_UDIS](#), [385](#)

[\\_TIM3\\_UIE, 385](#)  
[\\_TIM3\\_UIF, 386](#)  
[\\_TIM3\\_URS, 386](#)  
[\\_TIM4, 386](#)  
[\\_TIM4\\_ARR, 386](#)  
[\\_TIM4\\_ARPE, 386](#)  
[\\_TIM4\\_ARR\\_RESET\\_VALUE, 387](#)  
[\\_TIM4\\_CR, 387](#)  
[\\_TIM4\\_CEN, 387](#)  
[\\_TIM4\\_CNTR, 387](#)  
[\\_TIM4\\_CNTR\\_RESET\\_VALUE, 387](#)  
[\\_TIM4\\_CR\\_RESET\\_VALUE, 388](#)  
[\\_TIM4\\_EGR, 388](#)  
[\\_TIM4\\_EGR\\_RESET\\_VALUE, 388](#)  
[\\_TIM4\\_IER, 388](#)  
[\\_TIM4\\_IER\\_RESET\\_VALUE, 388](#)  
[\\_TIM4\\_OPM, 389](#)  
[\\_TIM4\\_PSC, 389](#)  
[\\_TIM4\\_PSC0, 389](#)  
[\\_TIM4\\_PSC1, 389](#)  
[\\_TIM4\\_PSC2, 389](#)  
[\\_TIM4\\_PSCR, 390](#)  
[\\_TIM4\\_PSCR\\_RESET\\_VALUE, 390](#)  
[\\_TIM4\\_SR, 390](#)  
[\\_TIM4\\_SR\\_RESET\\_VALUE, 390](#)  
[\\_TIM4\\_UG, 391](#)  
[\\_TIM4\\_UDIS, 390](#)  
[\\_TIM4\\_UIE, 391](#)  
[\\_TIM4\\_UIF, 391](#)  
[\\_TIM4\\_URS, 391](#)  
[\\_TIM5, 391](#)  
[\\_TIM5\\_ARPE, 392](#)  
[\\_TIM5\\_ARRH, 392](#)  
[\\_TIM5\\_ARRL, 392](#)  
[\\_TIM5\\_ARRH\\_RESET\\_VALUE, 392](#)  
[\\_TIM5\\_ARRL\\_RESET\\_VALUE, 392](#)  
[\\_TIM5\\_CC1E, 393](#)  
[\\_TIM5\\_CC1G, 393](#)  
[\\_TIM5\\_CC1P, 394](#)  
[\\_TIM5\\_CC1S, 394](#)  
[\\_TIM5\\_CC1IE, 393](#)  
[\\_TIM5\\_CC1IF, 393](#)  
[\\_TIM5\\_CC1OF, 393](#)  
[\\_TIM5\\_CC1S0, 394](#)  
[\\_TIM5\\_CC1S1, 394](#)  
[\\_TIM5\\_CC2E, 394](#)  
[\\_TIM5\\_CC2G, 395](#)  
[\\_TIM5\\_CC2P, 395](#)  
[\\_TIM5\\_CC2S, 396](#)  
[\\_TIM5\\_CC2IE, 395](#)  
[\\_TIM5\\_CC2IF, 395](#)  
[\\_TIM5\\_CC2OF, 395](#)  
[\\_TIM5\\_CC2S0, 396](#)  
[\\_TIM5\\_CC2S1, 396](#)  
[\\_TIM5\\_CC3E, 396](#)  
[\\_TIM5\\_CC3G, 396](#)  
[\\_TIM5\\_CC3P, 397](#)  
[\\_TIM5\\_CC3S, 397](#)  
[\\_TIM5\\_CC3IE, 397](#)  
[\\_TIM5\\_CC3IF, 397](#)  
[\\_TIM5\\_CC3OF, 397](#)  
[\\_TIM5\\_CC3S0, 398](#)  
[\\_TIM5\\_CC3S1, 398](#)  
[\\_TIM5\\_CCER1, 398](#)  
[\\_TIM5\\_CCER1\\_RESET\\_VALUE, 398](#)  
[\\_TIM5\\_CCER2, 398](#)  
[\\_TIM5\\_CCER2\\_RESET\\_VALUE, 399](#)  
[\\_TIM5\\_CCMR1, 399](#)  
[\\_TIM5\\_CCMR1\\_RESET\\_VALUE, 399](#)  
[\\_TIM5\\_CCMR2, 399](#)  
[\\_TIM5\\_CCMR2\\_RESET\\_VALUE, 399](#)  
[\\_TIM5\\_CCMR3, 400](#)  
[\\_TIM5\\_CCMR3\\_RESET\\_VALUE, 400](#)  
[\\_TIM5\\_CCPC, 400](#)  
[\\_TIM5\\_CCR1H, 400](#)  
[\\_TIM5\\_CCR1L, 401](#)  
[\\_TIM5\\_CCR1H\\_RESET\\_VALUE, 400](#)  
[\\_TIM5\\_CCR1L\\_RESET\\_VALUE, 401](#)  
[\\_TIM5\\_CCR2H, 401](#)  
[\\_TIM5\\_CCR2L, 401](#)  
[\\_TIM5\\_CCR2H\\_RESET\\_VALUE, 401](#)  
[\\_TIM5\\_CCR2L\\_RESET\\_VALUE, 402](#)  
[\\_TIM5\\_CCR3H, 402](#)  
[\\_TIM5\\_CCR3L, 402](#)  
[\\_TIM5\\_CCR3H\\_RESET\\_VALUE, 402](#)  
[\\_TIM5\\_CCR3L\\_RESET\\_VALUE, 402](#)  
[\\_TIM5\\_CEN, 403](#)  
[\\_TIM5\\_CNTRH, 403](#)  
[\\_TIM5\\_CNTRL, 403](#)  
[\\_TIM5\\_CNTRH\\_RESET\\_VALUE, 403](#)  
[\\_TIM5\\_CNTRL\\_RESET\\_VALUE, 403](#)  
[\\_TIM5\\_COMS, 404](#)  
[\\_TIM5\\_CR1, 404](#)  
[\\_TIM5\\_CR1\\_RESET\\_VALUE, 404](#)  
[\\_TIM5\\_CR2, 404](#)  
[\\_TIM5\\_CR2\\_RESET\\_VALUE, 404](#)  
[\\_TIM5\\_EGR, 405](#)  
[\\_TIM5\\_EGR\\_RESET\\_VALUE, 405](#)  
[\\_TIM5\\_IC1F, 405](#)  
[\\_TIM5\\_IC1F0, 405](#)  
[\\_TIM5\\_IC1F1, 405](#)  
[\\_TIM5\\_IC1F2, 406](#)  
[\\_TIM5\\_IC1F3, 406](#)  
[\\_TIM5\\_IC1PSC, 406](#)  
[\\_TIM5\\_IC1PSC0, 406](#)  
[\\_TIM5\\_IC1PSC1, 406](#)  
[\\_TIM5\\_IC2F, 407](#)  
[\\_TIM5\\_IC2F0, 407](#)  
[\\_TIM5\\_IC2F1, 407](#)  
[\\_TIM5\\_IC2F2, 407](#)  
[\\_TIM5\\_IC2F3, 407](#)  
[\\_TIM5\\_IC2PSC, 408](#)  
[\\_TIM5\\_IC2PSC0, 408](#)  
[\\_TIM5\\_IC2PSC1, 408](#)  
[\\_TIM5\\_IC3F, 408](#)  
[\\_TIM5\\_IC3F0, 408](#)

[\\_TIM5\\_IC3F1](#), [409](#)  
[\\_TIM5\\_IC3F2](#), [409](#)  
[\\_TIM5\\_IC3F3](#), [409](#)  
[\\_TIM5\\_IC3PSC](#), [409](#)  
[\\_TIM5\\_IC3PSC0](#), [409](#)  
[\\_TIM5\\_IC3PSC1](#), [410](#)  
[\\_TIM5\\_IER](#), [410](#)  
[\\_TIM5\\_IER\\_RESET\\_VALUE](#), [410](#)  
[\\_TIM5\\_MMS](#), [410](#)  
[\\_TIM5\\_MMS0](#), [410](#)  
[\\_TIM5\\_MMS1](#), [411](#)  
[\\_TIM5\\_MMS2](#), [411](#)  
[\\_TIM5\\_MSM](#), [411](#)  
[\\_TIM5\\_OC1M](#), [411](#)  
[\\_TIM5\\_OC1M0](#), [411](#)  
[\\_TIM5\\_OC1M1](#), [412](#)  
[\\_TIM5\\_OC1M2](#), [412](#)  
[\\_TIM5\\_OC1PE](#), [412](#)  
[\\_TIM5\\_OC2M](#), [412](#)  
[\\_TIM5\\_OC2M0](#), [412](#)  
[\\_TIM5\\_OC2M1](#), [413](#)  
[\\_TIM5\\_OC2M2](#), [413](#)  
[\\_TIM5\\_OC2PE](#), [413](#)  
[\\_TIM5\\_OC3M](#), [413](#)  
[\\_TIM5\\_OC3M0](#), [413](#)  
[\\_TIM5\\_OC3M1](#), [414](#)  
[\\_TIM5\\_OC3M2](#), [414](#)  
[\\_TIM5\\_OC3PE](#), [414](#)  
[\\_TIM5\\_OPM](#), [414](#)  
[\\_TIM5\\_PSC](#), [414](#)  
[\\_TIM5\\_PSC0](#), [415](#)  
[\\_TIM5\\_PSC1](#), [415](#)  
[\\_TIM5\\_PSC2](#), [415](#)  
[\\_TIM5\\_PSC3](#), [415](#)  
[\\_TIM5\\_PSCR](#), [415](#)  
[\\_TIM5\\_PSCR\\_RESET\\_VALUE](#), [416](#)  
[\\_TIM5\\_SMS](#), [416](#)  
[\\_TIM5\\_SMCR](#), [416](#)  
[\\_TIM5\\_SMCR\\_RESET\\_VALUE](#), [416](#)  
[\\_TIM5\\_SMS0](#), [416](#)  
[\\_TIM5\\_SMS1](#), [417](#)  
[\\_TIM5\\_SMS2](#), [417](#)  
[\\_TIM5\\_SR1](#), [417](#)  
[\\_TIM5\\_SR1\\_RESET\\_VALUE](#), [417](#)  
[\\_TIM5\\_SR2](#), [417](#)  
[\\_TIM5\\_SR2\\_RESET\\_VALUE](#), [418](#)  
[\\_TIM5\\_TG](#), [418](#)  
[\\_TIM5\\_TS](#), [418](#)  
[\\_TIM5\\_TIE](#), [418](#)  
[\\_TIM5\\_TIF](#), [418](#)  
[\\_TIM5\\_TS0](#), [419](#)  
[\\_TIM5\\_TS1](#), [419](#)  
[\\_TIM5\\_TS2](#), [419](#)  
[\\_TIM5\\_UG](#), [419](#)  
[\\_TIM5\\_UDIS](#), [419](#)  
[\\_TIM5\\_UIE](#), [420](#)  
[\\_TIM5\\_UIF](#), [420](#)  
[\\_TIM5\\_URS](#), [420](#)  
[\\_TIM6](#), [420](#)  
[\\_TIM6\\_ARR](#), [421](#)  
[\\_TIM6\\_ARPE](#), [420](#)  
[\\_TIM6\\_ARR\\_RESET\\_VALUE](#), [421](#)  
[\\_TIM6\\_CR](#), [422](#)  
[\\_TIM6\\_CEN](#), [421](#)  
[\\_TIM6\\_CNTR](#), [421](#)  
[\\_TIM6\\_CNTR\\_RESET\\_VALUE](#), [421](#)  
[\\_TIM6\\_CR\\_RESET\\_VALUE](#), [422](#)  
[\\_TIM6\\_EGR](#), [422](#)  
[\\_TIM6\\_EGR\\_RESET\\_VALUE](#), [422](#)  
[\\_TIM6\\_IER](#), [422](#)  
[\\_TIM6\\_IER\\_RESET\\_VALUE](#), [423](#)  
[\\_TIM6\\_MMS](#), [423](#)  
[\\_TIM6\\_MMS0](#), [423](#)  
[\\_TIM6\\_MMS1](#), [423](#)  
[\\_TIM6\\_MMS2](#), [423](#)  
[\\_TIM6\\_OPM](#), [424](#)  
[\\_TIM6\\_PSC](#), [424](#)  
[\\_TIM6\\_PSC0](#), [424](#)  
[\\_TIM6\\_PSC1](#), [424](#)  
[\\_TIM6\\_PSC2](#), [424](#)  
[\\_TIM6\\_PSCR](#), [425](#)  
[\\_TIM6\\_PSCR\\_RESET\\_VALUE](#), [425](#)  
[\\_TIM6\\_SR](#), [426](#)  
[\\_TIM6\\_SMS](#), [425](#)  
[\\_TIM6\\_SMS0](#), [425](#)  
[\\_TIM6\\_SMS1](#), [425](#)  
[\\_TIM6\\_SMS2](#), [426](#)  
[\\_TIM6\\_SR\\_RESET\\_VALUE](#), [426](#)  
[\\_TIM6\\_TS](#), [426](#)  
[\\_TIM6\\_TS0](#), [426](#)  
[\\_TIM6\\_TS1](#), [427](#)  
[\\_TIM6\\_TS2](#), [427](#)  
[\\_TIM6\\_UG](#), [427](#)  
[\\_TIM6\\_UDIS](#), [427](#)  
[\\_TIM6\\_UIE](#), [427](#)  
[\\_TIM6\\_UIF](#), [428](#)  
[\\_TIM6\\_URS](#), [428](#)  
[\\_UART1](#), [428](#)  
[\\_UART1\\_M](#), [435](#)  
[\\_UART1\\_ADD](#), [428](#)  
[\\_UART1\\_ADD0](#), [428](#)  
[\\_UART1\\_ADD1](#), [429](#)  
[\\_UART1\\_ADD2](#), [429](#)  
[\\_UART1\\_ADD3](#), [429](#)  
[\\_UART1\\_BRR1](#), [429](#)  
[\\_UART1\\_BRR1\\_RESET\\_VALUE](#), [429](#)  
[\\_UART1\\_BRR2](#), [430](#)  
[\\_UART1\\_BRR2\\_RESET\\_VALUE](#), [430](#)  
[\\_UART1\\_CKEN](#), [430](#)  
[\\_UART1\\_CPHA](#), [430](#)  
[\\_UART1\\_CPOL](#), [430](#)  
[\\_UART1\\_CR1](#), [431](#)  
[\\_UART1\\_CR1\\_RESET\\_VALUE](#), [431](#)  
[\\_UART1\\_CR2](#), [431](#)  
[\\_UART1\\_CR2\\_RESET\\_VALUE](#), [431](#)  
[\\_UART1\\_CR3](#), [431](#)

[\\_UART1\\_CR3\\_RESET\\_VALUE](#), 432  
[\\_UART1\\_CR4](#), 432  
[\\_UART1\\_CR4\\_RESET\\_VALUE](#), 432  
[\\_UART1\\_CR5](#), 432  
[\\_UART1\\_CR5\\_RESET\\_VALUE](#), 432  
[\\_UART1\\_DR](#), 433  
[\\_UART1\\_FE](#), 433  
[\\_UART1\\_GTR](#), 433  
[\\_UART1\\_GTR\\_RESET\\_VALUE](#), 433  
[\\_UART1\\_HDSEL](#), 433  
[\\_UART1\\_IDLE](#), 434  
[\\_UART1\\_ILIEN](#), 434  
[\\_UART1\\_IREN](#), 434  
[\\_UART1\\_IRLP](#), 434  
[\\_UART1\\_LBCL](#), 434  
[\\_UART1\\_LBDF](#), 435  
[\\_UART1\\_LBDL](#), 435  
[\\_UART1\\_LBDIEN](#), 435  
[\\_UART1\\_LINEN](#), 435  
[\\_UART1\\_NF](#), 436  
[\\_UART1\\_NACK](#), 436  
[\\_UART1\\_OR\\_LHE](#), 436  
[\\_UART1\\_PE](#), 436  
[\\_UART1\\_PS](#), 437  
[\\_UART1\\_PCEN](#), 436  
[\\_UART1\\_PIEN](#), 437  
[\\_UART1\\_PSCR](#), 437  
[\\_UART1\\_PSCR\\_RESET\\_VALUE](#), 437  
[\\_UART1\\_R8](#), 437  
[\\_UART1\\_REN](#), 438  
[\\_UART1\\_RIEN](#), 438  
[\\_UART1\\_RWU](#), 438  
[\\_UART1\\_RXNE](#), 438  
[\\_UART1\\_SR](#), 439  
[\\_UART1\\_SBK](#), 438  
[\\_UART1\\_SCEN](#), 439  
[\\_UART1\\_SR\\_RESET\\_VALUE](#), 439  
[\\_UART1\\_STOP](#), 439  
[\\_UART1\\_STOP0](#), 439  
[\\_UART1\\_STOP1](#), 440  
[\\_UART1\\_T8](#), 440  
[\\_UART1\\_TC](#), 440  
[\\_UART1\\_TCIEN](#), 440  
[\\_UART1\\_TEN](#), 440  
[\\_UART1\\_TIEN](#), 441  
[\\_UART1\\_TXE](#), 441  
[\\_UART1\\_UARTD](#), 441  
[\\_UART1\\_WAKE](#), 441  
[\\_UART2](#), 441  
[\\_UART2\\_M](#), 450  
[\\_UART2\\_ADD](#), 442  
[\\_UART2\\_ADD0](#), 442  
[\\_UART2\\_ADD1](#), 442  
[\\_UART2\\_ADD2](#), 442  
[\\_UART2\\_ADD3](#), 442  
[\\_UART2\\_BRR1](#), 443  
[\\_UART2\\_BRR1\\_RESET\\_VALUE](#), 443  
[\\_UART2\\_BRR2](#), 443  
[\\_UART2\\_BRR2\\_RESET\\_VALUE](#), 443  
[\\_UART2\\_CKEN](#), 443  
[\\_UART2\\_CPHA](#), 444  
[\\_UART2\\_CPOL](#), 444  
[\\_UART2\\_CR1](#), 444  
[\\_UART2\\_CR1\\_RESET\\_VALUE](#), 444  
[\\_UART2\\_CR2](#), 444  
[\\_UART2\\_CR2\\_RESET\\_VALUE](#), 445  
[\\_UART2\\_CR3](#), 445  
[\\_UART2\\_CR3\\_RESET\\_VALUE](#), 445  
[\\_UART2\\_CR4](#), 445  
[\\_UART2\\_CR4\\_RESET\\_VALUE](#), 445  
[\\_UART2\\_CR5](#), 446  
[\\_UART2\\_CR5\\_RESET\\_VALUE](#), 446  
[\\_UART2\\_CR6](#), 446  
[\\_UART2\\_CR6\\_RESET\\_VALUE](#), 446  
[\\_UART2\\_DR](#), 446  
[\\_UART2\\_FE](#), 447  
[\\_UART2\\_GTR](#), 447  
[\\_UART2\\_GTR\\_RESET\\_VALUE](#), 447  
[\\_UART2\\_IDLE](#), 447  
[\\_UART2\\_ILIEN](#), 447  
[\\_UART2\\_IREN](#), 448  
[\\_UART2\\_IRLP](#), 448  
[\\_UART2\\_LASE](#), 448  
[\\_UART2\\_LBCL](#), 448  
[\\_UART2\\_LBDF](#), 448  
[\\_UART2\\_LBDL](#), 449  
[\\_UART2\\_LBDIEN](#), 449  
[\\_UART2\\_LDUM](#), 449  
[\\_UART2\\_LHDF](#), 449  
[\\_UART2\\_LHDIEN](#), 449  
[\\_UART2\\_LINEN](#), 450  
[\\_UART2\\_LSF](#), 450  
[\\_UART2\\_LSLV](#), 450  
[\\_UART2\\_NF](#), 451  
[\\_UART2\\_NACK](#), 450  
[\\_UART2\\_OR\\_LHE](#), 451  
[\\_UART2\\_PE](#), 451  
[\\_UART2\\_PS](#), 452  
[\\_UART2\\_PCEN](#), 451  
[\\_UART2\\_PIEN](#), 451  
[\\_UART2\\_PSCR](#), 452  
[\\_UART2\\_PSCR\\_RESET\\_VALUE](#), 452  
[\\_UART2\\_R8](#), 452  
[\\_UART2\\_REN](#), 452  
[\\_UART2\\_RIEN](#), 453  
[\\_UART2\\_RWU](#), 453  
[\\_UART2\\_RXNE](#), 453  
[\\_UART2\\_SR](#), 454  
[\\_UART2\\_SBK](#), 453  
[\\_UART2\\_SCEN](#), 453  
[\\_UART2\\_SR\\_RESET\\_VALUE](#), 454  
[\\_UART2\\_STOP](#), 454  
[\\_UART2\\_STOP0](#), 454  
[\\_UART2\\_STOP1](#), 454  
[\\_UART2\\_T8](#), 455  
[\\_UART2\\_TC](#), 455

\_UART2\_TCEN, 455  
 \_UART2\_TEN, 455  
 \_UART2\_TIEN, 455  
 \_UART2\_TXE, 456  
 \_UART2\_UARTD, 456  
 \_UART2\_WAKE, 456  
 \_UART3, 456  
 \_UART3\_M, 463  
 \_UART3\_ADD, 456  
 \_UART3\_ADD0, 457  
 \_UART3\_ADD1, 457  
 \_UART3\_ADD2, 457  
 \_UART3\_ADD3, 457  
 \_UART3\_BRR1, 457  
 \_UART3\_BRR1\_RESET\_VALUE, 458  
 \_UART3\_BRR2, 458  
 \_UART3\_BRR2\_RESET\_VALUE, 458  
 \_UART3\_CR1, 458  
 \_UART3\_CR1\_RESET\_VALUE, 458  
 \_UART3\_CR2, 459  
 \_UART3\_CR2\_RESET\_VALUE, 459  
 \_UART3\_CR3, 459  
 \_UART3\_CR3\_RESET\_VALUE, 459  
 \_UART3\_CR4, 459  
 \_UART3\_CR4\_RESET\_VALUE, 460  
 \_UART3\_CR6, 460  
 \_UART3\_CR6\_RESET\_VALUE, 460  
 \_UART3\_DR, 460  
 \_UART3\_FE, 460  
 \_UART3\_IDLE, 461  
 \_UART3\_ILIEN, 461  
 \_UART3\_LASE, 461  
 \_UART3\_LBDF, 461  
 \_UART3\_LBDL, 462  
 \_UART3\_LBDIEN, 461  
 \_UART3\_LDUM, 462  
 \_UART3\_LHDF, 462  
 \_UART3\_LHDIEN, 462  
 \_UART3\_LINEN, 462  
 \_UART3\_LSF, 463  
 \_UART3\_LSLV, 463  
 \_UART3\_NF, 463  
 \_UART3\_OR\_LHE, 463  
 \_UART3\_PE, 464  
 \_UART3\_PS, 464  
 \_UART3\_PCEN, 464  
 \_UART3\_PIEN, 464  
 \_UART3\_R8, 464  
 \_UART3\_REN, 465  
 \_UART3\_RIEN, 465  
 \_UART3\_RWU, 465  
 \_UART3\_RXNE, 465  
 \_UART3\_SR, 466  
 \_UART3\_SBK, 465  
 \_UART3\_SR\_RESET\_VALUE, 466  
 \_UART3\_STOP, 466  
 \_UART3\_STOP0, 466  
 \_UART3\_STOP1, 466  
 \_UART3\_T8, 467  
 \_UART3\_TC, 467  
 \_UART3\_TCEN, 467  
 \_UART3\_TEN, 467  
 \_UART3\_TIEN, 467  
 \_UART3\_TXE, 468  
 \_UART3\_UARTD, 468  
 \_UART3\_WAKE, 468  
 \_UART4, 468  
 \_UART4\_M, 477  
 \_UART4\_ADD, 468  
 \_UART4\_ADD0, 469  
 \_UART4\_ADD1, 469  
 \_UART4\_ADD2, 469  
 \_UART4\_ADD3, 469  
 \_UART4\_BRR1, 469  
 \_UART4\_BRR1\_RESET\_VALUE, 470  
 \_UART4\_BRR2, 470  
 \_UART4\_BRR2\_RESET\_VALUE, 470  
 \_UART4\_CKEN, 470  
 \_UART4\_CPHA, 470  
 \_UART4\_CPOL, 471  
 \_UART4\_CR1, 471  
 \_UART4\_CR1\_RESET\_VALUE, 471  
 \_UART4\_CR2, 471  
 \_UART4\_CR2\_RESET\_VALUE, 471  
 \_UART4\_CR3, 472  
 \_UART4\_CR3\_RESET\_VALUE, 472  
 \_UART4\_CR4, 472  
 \_UART4\_CR4\_RESET\_VALUE, 472  
 \_UART4\_CR5, 472  
 \_UART4\_CR5\_RESET\_VALUE, 473  
 \_UART4\_CR6, 473  
 \_UART4\_CR6\_RESET\_VALUE, 473  
 \_UART4\_DR, 473  
 \_UART4\_FE, 473  
 \_UART4\_GTR, 474  
 \_UART4\_GTR\_RESET\_VALUE, 474  
 \_UART4\_HDSEL, 474  
 \_UART4\_IDLE, 474  
 \_UART4\_ILIEN, 474  
 \_UART4\_IREN, 475  
 \_UART4\_IRLP, 475  
 \_UART4\_LASE, 475  
 \_UART4\_LBCL, 475  
 \_UART4\_LBDF, 475  
 \_UART4\_LBDL, 476  
 \_UART4\_LBDIEN, 476  
 \_UART4\_LDUM, 476  
 \_UART4\_LHDF, 476  
 \_UART4\_LHDIEN, 476  
 \_UART4\_LINEN, 477  
 \_UART4\_LSF, 477  
 \_UART4\_LSLV, 477  
 \_UART4\_NF, 478  
 \_UART4\_NACK, 477  
 \_UART4\_OR\_LHE, 478  
 \_UART4\_PE, 478

[\\_UART4\\_PS](#), 479  
[\\_UART4\\_PCEN](#), 478  
[\\_UART4\\_PIEN](#), 478  
[\\_UART4\\_PSCR](#), 479  
[\\_UART4\\_PSCR\\_RESET\\_VALUE](#), 479  
[\\_UART4\\_R8](#), 479  
[\\_UART4\\_REN](#), 479  
[\\_UART4\\_RIEN](#), 480  
[\\_UART4\\_RWU](#), 480  
[\\_UART4\\_RXNE](#), 480  
[\\_UART4\\_SR](#), 481  
[\\_UART4\\_SBK](#), 480  
[\\_UART4\\_SCEN](#), 480  
[\\_UART4\\_SR\\_RESET\\_VALUE](#), 481  
[\\_UART4\\_STOP](#), 481  
[\\_UART4\\_STOP0](#), 481  
[\\_UART4\\_STOP1](#), 481  
[\\_UART4\\_T8](#), 482  
[\\_UART4\\_TC](#), 482  
[\\_UART4\\_TCIEN](#), 482  
[\\_UART4\\_TEN](#), 482  
[\\_UART4\\_TIEN](#), 482  
[\\_UART4\\_TXE](#), 483  
[\\_UART4\\_UARTD](#), 483  
[\\_UART4\\_WAKE](#), 483  
[\\_WWDG](#), 483  
[\\_WWDG\\_T](#), 484  
[\\_WWDG\\_W](#), 485  
[\\_WWDG\\_CR](#), 483  
[\\_WWDG\\_CR\\_RESET\\_VALUE](#), 484  
[\\_WWDG\\_T0](#), 484  
[\\_WWDG\\_T1](#), 484  
[\\_WWDG\\_T2](#), 484  
[\\_WWDG\\_T3](#), 485  
[\\_WWDG\\_T4](#), 485  
[\\_WWDG\\_T5](#), 485  
[\\_WWDG\\_T6](#), 485  
[\\_WWDG\\_W0](#), 486  
[\\_WWDG\\_W1](#), 486  
[\\_WWDG\\_W2](#), 486  
[\\_WWDG\\_W3](#), 486  
[\\_WWDG\\_W4](#), 486  
[\\_WWDG\\_W5](#), 487  
[\\_WWDG\\_W6](#), 487  
[\\_WWDG\\_WR](#), 487  
[\\_WWDG\\_WDGA](#), 487  
[\\_WWDG\\_WR\\_RESET\\_VALUE](#), 487  
[\\_\\_ADC1\\_VECTOR\\_\\_](#), 125  
[\\_\\_AWU\\_VECTOR\\_\\_](#), 126  
[\\_\\_CAN\\_RX\\_VECTOR\\_\\_](#), 126  
[\\_\\_CAN\\_TX\\_VECTOR\\_\\_](#), 126  
[\\_\\_CLK\\_VECTOR\\_\\_](#), 126  
[\\_\\_FLASH\\_VECTOR\\_\\_](#), 126  
[\\_\\_I2C\\_VECTOR\\_\\_](#), 127  
[\\_\\_PORTA\\_VECTOR\\_\\_](#), 127  
[\\_\\_PORTB\\_VECTOR\\_\\_](#), 127  
[\\_\\_PORTC\\_VECTOR\\_\\_](#), 127  
[\\_\\_PORTD\\_VECTOR\\_\\_](#), 127  
[\\_\\_PORTE\\_VECTOR\\_\\_](#), 127  
[\\_\\_PORTF\\_VECTOR\\_\\_](#), 128  
[\\_\\_SPI\\_VECTOR\\_\\_](#), 128  
[\\_\\_TIM1\\_CAPCOM\\_VECTOR\\_\\_](#), 128  
[\\_\\_TIM1\\_UPD\\_OVF\\_VECTOR\\_\\_](#), 128  
[\\_\\_TIM2\\_CAPCOM\\_VECTOR\\_\\_](#), 128  
[\\_\\_TIM2\\_UPD\\_OVF\\_VECTOR\\_\\_](#), 128  
[\\_\\_TIM3\\_CAPCOM\\_VECTOR\\_\\_](#), 129  
[\\_\\_TIM3\\_UPD\\_OVF\\_VECTOR\\_\\_](#), 129  
[\\_\\_TIM4\\_UPD\\_OVF\\_VECTOR\\_\\_](#), 129  
[\\_\\_TIM5\\_UPD\\_OVF\\_VECTOR\\_\\_](#), 129  
[\\_\\_TLI\\_VECTOR\\_\\_](#), 129  
[\\_\\_UART1\\_RXF\\_VECTOR\\_\\_](#), 130  
[\\_\\_UART1\\_TXE\\_VECTOR\\_\\_](#), 130  
[\\_\\_UART2\\_RXF\\_VECTOR\\_\\_](#), 130  
[\\_\\_UART2\\_TXE\\_VECTOR\\_\\_](#), 130  
[ADC1\\_AddressBase](#), 488–492  
[ADC2\\_AddressBase](#), 492–499  
[AWU\\_AddressBase](#), 499–511  
[BEEP\\_AddressBase](#), 511–522  
[CAN\\_AddressBase](#), 523–526  
[CFG\\_AddressBase](#), 526–537  
[CLK\\_AddressBase](#), 538–549  
[DISABLE\\_INTERRUPTS](#), 549  
[DM\\_AddressBase](#), 549–561  
[ENABLE\\_INTERRUPTS](#), 561  
[ENTER\\_HALT](#), 561  
[EXTI\\_AddressBase](#), 561–573  
[FLASH\\_AddressBase](#), 573–585  
[I2C\\_AddressBase](#), 585–596  
[ISR\\_HANDLER\\_TRAP](#), 597  
[ISR\\_HANDLER](#), 596  
[ITC\\_AddressBase](#), 597–608  
[IWDG\\_AddressBase](#), 609–620  
[NOP](#), 620  
[OPT\\_AddressBase](#), 620–632  
[PORTA\\_AddressBase](#), 632–644  
[PORTB\\_AddressBase](#), 644–655  
[PORTC\\_AddressBase](#), 655–667  
[PORTD\\_AddressBase](#), 667–679  
[PORTE\\_AddressBase](#), 679–690  
[PORTF\\_AddressBase](#), 690–702  
[PORTG\\_AddressBase](#), 702–711  
[PORTH\\_AddressBase](#), 711–718  
[PORTI\\_AddressBase](#), 719–726  
[RST\\_AddressBase](#), 726–737  
[SPI\\_AddressBase](#), 737–749  
[STM8\\_ADDR\\_WIDTH](#), 749  
[STM8\\_EEPROM\\_END](#), 749  
[STM8\\_EEPROM\\_SIZE](#), 749–761  
[STM8\\_EEPROM\\_START](#), 761  
[STM8\\_MEM\\_POINTER\\_T](#), 761  
[STM8\\_PFLASH\\_END](#), 762  
[STM8\\_PFLASH\\_SIZE](#), 762–773  
[STM8\\_PFLASH\\_START](#), 774  
[STM8\\_RAM\\_END](#), 774  
[STM8\\_RAM\\_SIZE](#), 774–786  
[STM8\\_RAM\\_START](#), 786

STM8AF5268, [786](#)  
 STM8AF5269, [786](#)  
 STM8AF526x, [786](#), [787](#)  
 STM8AF5286, [787](#)  
 STM8AF5288, [787](#)  
 STM8AF5289, [787](#)  
 STM8AF528A, [787](#)  
 STM8AF528x, [787](#), [788](#)  
 STM8AF52A6, [788](#)  
 STM8AF52A8, [788](#)  
 STM8AF52A9, [788](#)  
 STM8AF52AA, [789](#)  
 STM8AF52Ax, [789](#)  
 STM8AF6213, [789](#)  
 STM8AF6213A, [790](#)  
 STM8AF621x, [790](#)  
 STM8AF6223, [790](#)  
 STM8AF6223A, [790](#)  
 STM8AF6226, [790](#)  
 STM8AF622x, [791](#)  
 STM8AF6246, [791](#)  
 STM8AF6248, [791](#)  
 STM8AF624x, [791](#), [792](#)  
 STM8AF6266, [792](#)  
 STM8AF6268, [792](#)  
 STM8AF6269, [792](#)  
 STM8AF626x, [792](#), [793](#)  
 STM8AF6286, [793](#)  
 STM8AF6288, [793](#)  
 STM8AF6289, [793](#)  
 STM8AF628A, [793](#)  
 STM8AF628x, [793](#), [794](#)  
 STM8AF62A6, [794](#)  
 STM8AF62A8, [794](#)  
 STM8AF62A9, [794](#)  
 STM8AF62AA, [795](#)  
 STM8AF62Ax, [795](#)  
 STM8AF6366, [795](#)  
 STM8AF636x, [796](#)  
 STM8AF6388, [796](#)  
 STM8AF638x, [796](#)  
 STM8S001, [796](#)  
 STM8S001J3, [796](#)  
 STM8S003, [796](#), [797](#)  
 STM8S003F3, [797](#)  
 STM8S003K3, [797](#)  
 STM8S005, [797](#)  
 STM8S005C6, [797](#)  
 STM8S005K6, [798](#)  
 STM8S007, [798](#)  
 STM8S007C8, [798](#)  
 STM8S103, [798](#)  
 STM8S103F2, [799](#)  
 STM8S103F3, [799](#)  
 STM8S103K3, [799](#)  
 STM8S105, [799](#), [800](#)  
 STM8S105C4, [800](#)  
 STM8S105C6, [800](#)  
 STM8S105K4, [800](#)  
 STM8S105K6, [801](#)  
 STM8S105S4, [801](#)  
 STM8S105S6, [801](#)  
 STM8S207, [801–803](#)  
 STM8S207C6, [803](#)  
 STM8S207C8, [803](#)  
 STM8S207CB, [804](#)  
 STM8S207K6, [804](#)  
 STM8S207K8, [804](#)  
 STM8S207M8, [804](#)  
 STM8S207MB, [804](#)  
 STM8S207R6, [804](#)  
 STM8S207R8, [805](#)  
 STM8S207RB, [805](#)  
 STM8S207S6, [805](#)  
 STM8S207S8, [805](#)  
 STM8S207SB, [805](#)  
 STM8S208, [805–807](#)  
 STM8S208C6, [807](#)  
 STM8S208C8, [807](#)  
 STM8S208CB, [807](#)  
 STM8S208MB, [808](#)  
 STM8S208R6, [808](#)  
 STM8S208R8, [808](#)  
 STM8S208RB, [808](#)  
 STM8S208S6, [808](#)  
 STM8S208S8, [808](#)  
 STM8S208SB, [809](#)  
 STM8S903, [809](#)  
 STM8S903F3, [809](#)  
 STM8S903K3, [809](#)  
 SW\_RESET, [809](#)  
 TIM1\_AddressBase, [810–821](#)  
 TIM2\_AddressBase, [821–832](#)  
 TIM3\_AddressBase, [832–841](#)  
 TIM4\_AddressBase, [841–852](#)  
 TIM5\_AddressBase, [852, 853](#)  
 TIM6\_AddressBase, [853, 854](#)  
 TRIGGER\_TRAP, [854](#)  
 UART1\_AddressBase, [854–863](#)  
 UART2\_AddressBase, [863–865](#)  
 UART3\_AddressBase, [865–872](#)  
 UART4\_AddressBase, [872, 873](#)  
 UID\_AddressBase, [873–879](#)  
 WAIT\_FOR\_INTERRUPT, [879](#)  
 WWDG\_AddressBase, [879–890](#)  
 STM8S001  
     STM8AF\_STM8S, [796](#)  
 STM8S001J3  
     STM8AF\_STM8S, [796](#)  
 STM8S003  
     STM8AF\_STM8S, [796, 797](#)  
 STM8S003F3  
     STM8AF\_STM8S, [797](#)  
 STM8S003K3  
     STM8AF\_STM8S, [797](#)  
 STM8S005



STM8AF\_STM8S, [797](#)  
STM8S005C6  
STM8AF\_STM8S, [797](#)  
STM8S005K6  
STM8AF\_STM8S, [798](#)  
STM8S007  
STM8AF\_STM8S, [798](#)  
STM8S007C8  
STM8AF\_STM8S, [798](#)  
STM8S103  
STM8AF\_STM8S, [798](#)  
STM8S103F2  
STM8AF\_STM8S, [799](#)  
STM8S103F3  
STM8AF\_STM8S, [799](#)  
STM8S103K3  
STM8AF\_STM8S, [799](#)  
STM8S105  
STM8AF\_STM8S, [799](#), [800](#)  
STM8S105C4  
STM8AF\_STM8S, [800](#)  
STM8S105C6  
STM8AF\_STM8S, [800](#)  
STM8S105K4  
STM8AF\_STM8S, [800](#)  
STM8S105K6  
STM8AF\_STM8S, [801](#)  
STM8S105S4  
STM8AF\_STM8S, [801](#)  
STM8S105S6  
STM8AF\_STM8S, [801](#)  
STM8S207  
STM8AF\_STM8S, [801–803](#)  
STM8S207C6  
STM8AF\_STM8S, [803](#)  
STM8S207C8  
STM8AF\_STM8S, [803](#)  
STM8S207CB  
STM8AF\_STM8S, [804](#)  
STM8S207K6  
STM8AF\_STM8S, [804](#)  
STM8S207K8  
STM8AF\_STM8S, [804](#)  
STM8S207M8  
STM8AF\_STM8S, [804](#)  
STM8S207MB  
STM8AF\_STM8S, [804](#)  
STM8S207R6  
STM8AF\_STM8S, [804](#)  
STM8S207R8  
STM8AF\_STM8S, [805](#)  
STM8S207RB  
STM8AF\_STM8S, [805](#)  
STM8S207S6  
STM8AF\_STM8S, [805](#)  
STM8S207S8  
STM8AF\_STM8S, [805](#)  
STM8S207SB  
STM8AF\_STM8S, [805](#)  
STM8S208  
STM8AF\_STM8S, [805–807](#)  
STM8S208C6  
STM8AF\_STM8S, [807](#)  
STM8S208C8  
STM8AF\_STM8S, [807](#)  
STM8S208CB  
STM8AF\_STM8S, [807](#)  
STM8S208MB  
STM8AF\_STM8S, [808](#)  
STM8S208R6  
STM8AF\_STM8S, [808](#)  
STM8S208R8  
STM8AF\_STM8S, [808](#)  
STM8S208RB  
STM8AF\_STM8S, [808](#)  
STM8S208S6  
STM8AF\_STM8S, [808](#)  
STM8S208S8  
STM8AF\_STM8S, [808](#)  
STM8S208SB  
STM8AF\_STM8S, [809](#)  
STM8S903  
STM8AF\_STM8S, [809](#)  
STM8S903F3  
STM8AF\_STM8S, [809](#)  
STM8S903K3  
STM8AF\_STM8S, [809](#)  
STOPF  
I2C\_t, [1021](#)  
STOP  
I2C\_t, [1020](#)  
UART1\_t, [1170](#)  
UART2\_t, [1186](#)  
UART3\_t, [1199](#)  
UART4\_t, [1215](#)  
SW\_RESET  
STM8AF\_STM8S, [809](#)  
SWBSY  
CLK\_t, [995](#)  
SWCR  
CLK\_t, [995](#)  
SWEN  
CLK\_t, [995](#)  
SWIEN  
CLK\_t, [996](#)  
SWIMCCR  
CLK\_t, [996](#)  
SWIMCLK  
CLK\_t, [996](#)  
SWIMF  
RST\_t, [1041](#)  
SWIF  
CLK\_t, [996](#)  
SWRST  
I2C\_t, [1021](#)  
SWD

- CFG\_t, [983](#)
- SWI
  - CLK\_t, [995](#)
- SWR
  - CLK\_t, [996](#)
- SB
  - I2C\_t, [1020](#)
- SR
  - RST\_t, [1041](#)
  - SPI\_t, [1049](#)
  - TIM4\_t, [1127](#)
  - TIM6\_t, [1156](#)
  - UART1\_t, [1170](#)
  - UART2\_t, [1186](#)
  - UART3\_t, [1199](#)
  - UART4\_t, [1215](#)
- T
  - WWDG\_t, [1218](#)
- T8
  - UART1\_t, [1170](#)
  - UART2\_t, [1186](#)
  - UART3\_t, [1199](#)
  - UART4\_t, [1215](#)
- TBR
  - AWU\_t, [917](#)
- TCIEN
  - UART1\_t, [1170](#)
  - UART2\_t, [1186](#)
  - UART3\_t, [1199](#)
  - UART4\_t, [1215](#)
- TDRH
  - ADC1\_t, [908](#)
  - ADC2\_t, [914](#)
- TDRL
  - ADC1\_t, [908](#)
  - ADC2\_t, [914](#)
- TDH
  - ADC1\_t, [907](#)
  - ADC2\_t, [914](#)
- TDL
  - ADC1\_t, [908](#)
  - ADC2\_t, [914](#)
- TECR
  - CAN\_t, [977](#)
- TERR
  - CAN\_t, [977](#)
- TEC
  - CAN\_t, [977](#)
- TEN
  - UART1\_t, [1171](#)
  - UART2\_t, [1187](#)
  - UART3\_t, [1200](#)
  - UART4\_t, [1216](#)
- TGT
  - CAN\_t, [978](#)
- TIEN
  - UART1\_t, [1171](#)
  - UART2\_t, [1187](#)
- UART3\_t, [1200](#)
- UART4\_t, [1216](#)
- TIM1\_AddressBase
  - STM8AF\_STM8S, [810–821](#)
- TIM1\_t, [1051](#)
  - AOE, [1059](#)
  - ARPE, [1059](#)
  - ARRH, [1059](#)
  - ARRL, [1060](#)
  - ARR, [1059](#)
  - BIE, [1060](#)
  - BIF, [1060](#)
  - BKE, [1060](#)
  - BKP, [1061](#)
  - BKR, [1061](#)
  - BG, [1060](#)
  - CC1IE, [1061](#)
  - CC1IF, [1062](#)
  - CC1NE, [1062](#)
  - CC1NP, [1062](#)
  - CC1OF, [1062](#)
  - CC1E, [1061](#)
  - CC1G, [1061](#)
  - CC1P, [1062](#)
  - CC1S, [1063](#)
  - CC2IE, [1063](#)
  - CC2IF, [1063](#)
  - CC2NE, [1064](#)
  - CC2NP, [1064](#)
  - CC2OF, [1064](#)
  - CC2E, [1063](#)
  - CC2G, [1063](#)
  - CC2P, [1064](#)
  - CC2S, [1064](#)
  - CC3IE, [1065](#)
  - CC3IF, [1065](#)
  - CC3NE, [1065](#)
  - CC3NP, [1066](#)
  - CC3OF, [1066](#)
  - CC3E, [1065](#)
  - CC3G, [1065](#)
  - CC3P, [1066](#)
  - CC3S, [1066](#)
  - CC4IE, [1067](#)
  - CC4IF, [1067](#)
  - CC4OF, [1067](#)
  - CC4E, [1066](#)
  - CC4G, [1067](#)
  - CC4P, [1067](#)
  - CC4S, [1068](#)
  - CCER1, [1068](#)
  - CCER2, [1068](#)
  - CCMR1, [1068](#)
  - CCMR2, [1068](#)
  - CCMR3, [1068](#)
  - CCMR4, [1069](#)
  - CCPC, [1069](#)
  - CCR1, [1069](#)

CCR1H, [1069](#)  
CCR1L, [1069](#)  
CCR2, [1070](#)  
CCR2H, [1070](#)  
CCR2L, [1070](#)  
CCR3, [1070](#)  
CCR3H, [1070](#)  
CCR3L, [1071](#)  
CCR4, [1071](#)  
CCR4H, [1071](#)  
CCR4L, [1071](#)  
CEN, [1071](#)  
CMS, [1071](#)  
CNTRH, [1072](#)  
CNTRL, [1072](#)  
CNT, [1072](#)  
COMIE, [1072](#)  
COMIF, [1073](#)  
COMG, [1072](#)  
COMS, [1073](#)  
CR1, [1073](#)  
CR2, [1073](#)  
DIR, [1073](#)  
DTG, [1074](#)  
DTR, [1074](#)  
ECE, [1074](#)  
EGR, [1074](#)  
ETPS, [1075](#)  
ETF, [1074](#)  
ETP, [1075](#)  
ETR, [1075](#)  
IC1PSC, [1075](#)  
IC1F, [1075](#)  
IC2PSC, [1076](#)  
IC2F, [1076](#)  
IC3PSC, [1076](#)  
IC3F, [1076](#)  
IC4PSC, [1077](#)  
IC4F, [1076](#)  
IER, [1077](#)  
IN, [1077](#)  
LOCK, [1078](#)  
MMS, [1078](#)  
MOE, [1078](#)  
MSM, [1078](#)  
OC1CE, [1078](#)  
OC1FE, [1079](#)  
OC1PE, [1079](#)  
OC1M, [1079](#)  
OC2CE, [1079](#)  
OC2FE, [1079](#)  
OC2PE, [1080](#)  
OC2M, [1080](#)  
OC3CE, [1080](#)  
OC3FE, [1080](#)  
OC3PE, [1081](#)  
OC3M, [1080](#)  
OC4CE, [1081](#)  
OC4FE, [1081](#)  
OC4PE, [1081](#)  
OC4M, [1081](#)  
OIS1, [1082](#)  
OIS1N, [1082](#)  
OIS2, [1082](#)  
OIS2N, [1082](#)  
OIS3, [1082](#)  
OIS3N, [1083](#)  
OIS4, [1083](#)  
OISR, [1083](#)  
OPM, [1083](#)  
OSSI, [1083](#)  
OSSR, [1084](#)  
OUT, [1084](#)  
PSCRH, [1085](#)  
PSCRL, [1085](#)  
PSC, [1084](#)  
RCR, [1085](#)  
REP, [1085](#)  
res, [1085](#)  
res2, [1086](#)  
res3, [1086](#)  
SMCR, [1086](#)  
SMS, [1086](#)  
SR1, [1086](#)  
SR2, [1087](#)  
TIE, [1087](#)  
TIF, [1087](#)  
TG, [1087](#)  
TS, [1087](#)  
UDIS, [1088](#)  
UIE, [1088](#)  
UIF, [1088](#)  
URS, [1088](#)  
UG, [1088](#)  
TIM2\_AddressBase  
  STM8AF\_STM8S, [821–832](#)  
TIM2\_t, [1089](#)  
  ARPE, [1093](#)  
  ARRH, [1094](#)  
  ARRL, [1094](#)  
  ARR, [1094](#)  
  CC1IE, [1095](#)  
  CC1IF, [1095](#)  
  CC1OF, [1095](#)  
  CC1E, [1094](#)  
  CC1G, [1094](#)  
  CC1P, [1095](#)  
  CC1S, [1095](#)  
  CC2IE, [1096](#)  
  CC2IF, [1096](#)  
  CC2OF, [1096](#)  
  CC2E, [1096](#)  
  CC2G, [1096](#)  
  CC2P, [1097](#)  
  CC2S, [1097](#)  
  CC3IE, [1097](#)

- CC3IF, [1098](#)
- CC3OF, [1098](#)
- CC3E, [1097](#)
- CC3G, [1097](#)
- CC3P, [1098](#)
- CC3S, [1098](#)
- CCER1, [1098](#)
- CCER2, [1099](#)
- CCMR1, [1099](#)
- CCMR2, [1099](#)
- CCMR3, [1099](#)
- CCR1, [1099](#)
- CCR1H, [1099](#)
- CCR1L, [1100](#)
- CCR2, [1100](#)
- CCR2H, [1100](#)
- CCR2L, [1100](#)
- CCR3, [1100](#)
- CCR3H, [1101](#)
- CCR3L, [1101](#)
- CEN, [1101](#)
- CNTRH, [1101](#)
- CNTRL, [1102](#)
- CNT, [1101](#)
- CR1, [1102](#)
- EGR, [1102](#)
- IC1PSC, [1102](#)
- IC1F, [1102](#)
- IC2PSC, [1103](#)
- IC2F, [1102](#)
- IC3PSC, [1103](#)
- IC3F, [1103](#)
- IER, [1103](#)
- IN, [1103](#), [1104](#)
- OC1PE, [1104](#)
- OC1M, [1104](#)
- OC2PE, [1104](#)
- OC2M, [1104](#)
- OC3CE, [1105](#)
- OC3PE, [1105](#)
- OC3M, [1105](#)
- OPM, [1105](#)
- OUT, [1105](#), [1106](#)
- PSCR, [1106](#)
- PSC, [1106](#)
- res, [1106](#)
- res2, [1106](#)
- SR1, [1107](#)
- SR2, [1107](#)
- UDIS, [1107](#)
- UIE, [1107](#)
- UIF, [1108](#)
- URS, [1108](#)
- UG, [1107](#)
- TIM3\_AddressBase
  - STM8AF\_STM8S, [832–841](#)
- TIM3\_t, [1108](#)
- ARPE, [1112](#)
- ARRH, [1112](#)
- ARRL, [1113](#)
- ARR, [1112](#)
- CC1IE, [1113](#)
- CC1IF, [1113](#)
- CC1OF, [1114](#)
- CC1E, [1113](#)
- CC1G, [1113](#)
- CC1P, [1114](#)
- CC1S, [1114](#)
- CC2IE, [1115](#)
- CC2IF, [1115](#)
- CC2OF, [1115](#)
- CC2E, [1114](#)
- CC2G, [1114](#)
- CC2P, [1115](#)
- CC2S, [1115](#)
- CCER1, [1116](#)
- CCMR1, [1116](#)
- CCMR2, [1116](#)
- CCR1, [1116](#)
- CCR1H, [1116](#)
- CCR1L, [1117](#)
- CCR2, [1117](#)
- CCR2H, [1117](#)
- CCR2L, [1117](#)
- CEN, [1117](#)
- CNTRH, [1118](#)
- CNTRL, [1118](#)
- CNT, [1117](#)
- CR1, [1118](#)
- EGR, [1118](#)
- IC1PSC, [1118](#)
- IC1F, [1118](#)
- IC2PSC, [1119](#)
- IC2F, [1119](#)
- IER, [1119](#)
- IN, [1119](#)
- OC1PE, [1120](#)
- OC1M, [1120](#)
- OC2PE, [1120](#)
- OC2M, [1120](#)
- OPM, [1120](#)
- OUT, [1121](#)
- PSCR, [1121](#)
- PSC, [1121](#)
- res, [1121](#)
- res2, [1122](#)
- SR1, [1122](#)
- SR2, [1122](#)
- UDIS, [1122](#)
- UIE, [1123](#)
- UIF, [1123](#)
- URS, [1123](#)
- UG, [1122](#)
- TIM4\_AddressBase
  - STM8AF\_STM8S, [841–852](#)
- TIM4\_t, [1123](#)

- ARPE, [1125](#)
- ARR, [1125](#)
- CEN, [1125](#)
- CNTR, [1126](#)
- CNT, [1126](#)
- CR, [1126](#)
- EGR, [1126](#)
- IER, [1126](#)
- OPM, [1126](#)
- PSCR, [1127](#)
- PSC, [1127](#)
- res, [1127](#)
- SR, [1127](#)
- UDIS, [1127](#)
- UIE, [1128](#)
- UIF, [1128](#)
- URS, [1128](#)
- UG, [1128](#)
- TIM5\_AddressBase
  - STM8AF\_STM8S, [852](#), [853](#)
- TIM5\_t, [1129](#)
  - ARPE, [1134](#)
  - ARRH, [1134](#)
  - ARRL, [1134](#)
  - ARR, [1134](#)
  - CC1IE, [1135](#)
  - CC1IF, [1135](#)
  - CC1OF, [1135](#)
  - CC1E, [1135](#)
  - CC1G, [1135](#)
  - CC1P, [1136](#)
  - CC1S, [1136](#)
  - CC2IE, [1136](#)
  - CC2IF, [1137](#)
  - CC2OF, [1137](#)
  - CC2E, [1136](#)
  - CC2G, [1136](#)
  - CC2P, [1137](#)
  - CC2S, [1137](#)
  - CC3IE, [1138](#)
  - CC3IF, [1138](#)
  - CC3OF, [1138](#)
  - CC3E, [1137](#)
  - CC3G, [1138](#)
  - CC3P, [1138](#)
  - CC3S, [1139](#)
  - CCER1, [1139](#)
  - CCER2, [1139](#)
  - CCMR1, [1139](#)
  - CCMR2, [1139](#)
  - CCMR3, [1139](#)
  - CCPC, [1140](#)
  - CCR1, [1140](#)
  - CCR1H, [1140](#)
  - CCR1L, [1140](#)
  - CCR2, [1140](#)
  - CCR2H, [1141](#)
  - CCR2L, [1141](#)
  - CCR3, [1141](#)
  - CCR3H, [1141](#)
  - CCR3L, [1141](#)
  - CEN, [1142](#)
  - CNTRH, [1142](#)
  - CNTRL, [1142](#)
  - CNT, [1142](#)
  - COMS, [1142](#)
  - CR1, [1143](#)
  - CR2, [1143](#)
  - EGR, [1143](#)
  - IC1PSC, [1143](#)
  - IC1F, [1143](#)
  - IC2PSC, [1144](#)
  - IC2F, [1143](#)
  - IC3PSC, [1144](#)
  - IC3F, [1144](#)
  - IER, [1144](#)
  - IN, [1144](#), [1145](#)
  - MMS, [1145](#)
  - MSM, [1145](#)
  - OC1PE, [1145](#)
  - OC1M, [1145](#)
  - OC2PE, [1146](#)
  - OC2M, [1146](#)
  - OC3PE, [1146](#)
  - OC3M, [1146](#)
  - OPM, [1146](#)
  - OUT, [1147](#)
  - PSCR, [1147](#)
  - PSC, [1147](#)
  - res, [1147](#)
  - res2, [1148](#)
  - res3, [1148](#)
  - SMCR, [1148](#)
  - SMS, [1148](#)
  - SR1, [1148](#)
  - SR2, [1149](#)
  - TIE, [1149](#)
  - TIF, [1149](#)
  - TG, [1149](#)
  - TS, [1149](#)
  - UDIS, [1150](#)
  - UIE, [1150](#)
  - UIF, [1150](#)
  - URS, [1150](#)
  - UG, [1150](#)
- TIM6\_AddressBase
  - STM8AF\_STM8S, [853](#), [854](#)
- TIM6\_t, [1151](#)
  - ARPE, [1152](#)
  - ARR, [1153](#)
  - CEN, [1153](#)
  - CNTR, [1153](#)
  - CNT, [1153](#)
  - CR1, [1154](#)
  - CR2, [1154](#)
  - EGR, [1154](#)

- IER, [1154](#)
- MMS, [1154](#)
- MSM, [1154](#)
- OPM, [1155](#)
- PSCR, [1155](#)
- PSC, [1155](#)
- res, [1155](#)
- res2, [1155](#)
- SMCR, [1156](#)
- SMS, [1156](#)
- SR, [1156](#)
- TS, [1156](#)
- UDIS, [1156](#)
- UIE, [1157](#)
- UIF, [1157](#)
- URS, [1157](#)
- UG, [1157](#)
- TIME
  - CAN\_t, [978](#)
- TIE
  - TIM1\_t, [1087](#)
  - TIM5\_t, [1149](#)
- TIF
  - TIM1\_t, [1087](#)
  - TIM5\_t, [1149](#)
- TLIS
  - EXTI\_t, [999](#)
- TME0
  - CAN\_t, [978](#)
- TME1
  - CAN\_t, [978](#)
- TME2
  - CAN\_t, [978](#)
- TMEIE
  - CAN\_t, [979](#)
- TPR
  - CAN\_t, [979](#)
- TRIGGER\_TRAP
  - STM8AF\_STM8S, [854](#)
- TRISER
  - I2C\_t, [1021](#)
- TRISE
  - I2C\_t, [1021](#)
- TRA
  - I2C\_t, [1021](#)
- TSR
  - CAN\_t, [979](#)
- TTCM
  - CAN\_t, [979](#)
- TXCRCR
  - SPI\_t, [1050](#)
- TXFP
  - CAN\_t, [980](#)
- TXIE
  - SPI\_t, [1051](#)
- TXM2E
  - CAN\_t, [980](#)
- TXOK0
  - CAN\_t, [980](#)
- TXOK1
  - CAN\_t, [980](#)
- TXOK2
  - CAN\_t, [981](#)
- TXOK
  - CAN\_t, [980](#)
- TXRQ
  - CAN\_t, [981](#)
- TXE
  - I2C\_t, [1022](#)
  - SPI\_t, [1050](#)
  - UART1\_t, [1171](#)
  - UART2\_t, [1187](#)
  - UART3\_t, [1200](#)
  - UART4\_t, [1216](#)
- TC
  - UART1\_t, [1170](#)
  - UART2\_t, [1186](#)
  - UART3\_t, [1199](#)
  - UART4\_t, [1215](#)
- TG
  - TIM1\_t, [1087](#)
  - TIM5\_t, [1149](#)
- TS
  - TIM1\_t, [1087](#)
  - TIM5\_t, [1149](#)
  - TIM6\_t, [1156](#)
- TX
  - CAN\_t, [979](#)
- TxCRC
  - SPI\_t, [1050](#)
- UART1\_AddressBase
  - STM8AF\_STM8S, [854–863](#)
- UART1\_t, [1158](#)
  - ADD, [1160](#)
  - BRR1, [1160](#)
  - BRR2, [1161](#)
  - CKEN, [1161](#)
  - CPHA, [1161](#)
  - CPOL, [1161](#)
  - CR1, [1161](#)
  - CR2, [1162](#)
  - CR3, [1162](#)
  - CR4, [1162](#)
  - CR5, [1162](#)
  - DATA, [1162](#)
  - DIV\_0\_3, [1162](#)
  - DIV\_12\_15, [1163](#)
  - DIV\_4\_11, [1163](#)
  - DR, [1163](#)
  - FE, [1163](#)
  - GTR, [1164](#)
  - GT, [1163](#)
  - HDSEL, [1164](#)
  - IDLE, [1164](#)
  - ILIEN, [1164](#)
  - IREN, [1164](#)

- IRLP, [1165](#)
- LBCL, [1165](#)
- LBDIEN, [1165](#)
- LBDF, [1165](#)
- LBDL, [1165](#)
- LINEN, [1166](#)
- M, [1166](#)
- NACK, [1166](#)
- NF, [1166](#)
- OR, [1166](#)
- PCEN, [1167](#)
- PIEN, [1167](#)
- PSCR, [1168](#)
- PSC, [1167](#)
- PE, [1167](#)
- PS, [1167](#)
- R8, [1168](#)
- REN, [1168](#)
- RIEN, [1169](#)
- RWU, [1169](#)
- RXNE, [1169](#)
- res, [1168](#)
- res2, [1168](#)
- SBK, [1169](#)
- SCEN, [1169](#)
- STOP, [1170](#)
- SR, [1170](#)
- T8, [1170](#)
- TCIEN, [1170](#)
- TEN, [1171](#)
- TIEN, [1171](#)
- TXE, [1171](#)
- TC, [1170](#)
- UARTD, [1171](#)
- WAKE, [1171](#)
- UART2\_AddressBase
  - STM8AF\_STM8S, [863–865](#)
- UART2\_t, [1172](#)
  - ADD, [1175](#)
  - BRR1, [1175](#)
  - BRR2, [1175](#)
  - CKEN, [1175](#)
  - CPHA, [1176](#)
  - CPOL, [1176](#)
  - CR1, [1176](#)
  - CR2, [1176](#)
  - CR3, [1176](#)
  - CR4, [1177](#)
  - CR5, [1177](#)
  - CR6, [1177](#)
  - DATA, [1177](#)
  - DIV\_0\_3, [1177](#)
  - DIV\_12\_15, [1177](#)
  - DIV\_4\_11, [1178](#)
  - DR, [1178](#)
  - FE, [1178](#)
  - GTR, [1178](#)
  - GT, [1178](#)
  - IDLE, [1179](#)
  - ILIEN, [1179](#)
  - IREN, [1179](#)
  - IRLP, [1179](#)
  - LASE, [1179](#)
  - LBCL, [1180](#)
  - LBDIEN, [1180](#)
  - LBDF, [1180](#)
  - LBDL, [1180](#)
  - LDUM, [1180](#)
  - LHDIEN, [1181](#)
  - LHDF, [1181](#)
  - LINEN, [1181](#)
  - LSLV, [1181](#)
  - LSF, [1181](#)
  - M, [1182](#)
  - NACK, [1182](#)
  - NF, [1182](#)
  - OR, [1182](#)
  - PCEN, [1182](#)
  - PIEN, [1183](#)
  - PSCR, [1183](#)
  - PSC, [1183](#)
  - PE, [1183](#)
  - PS, [1183](#)
  - R8, [1184](#)
  - REN, [1184](#)
  - RIEN, [1185](#)
  - RWU, [1185](#)
  - RXNE, [1185](#)
  - res, [1184](#)
  - res2, [1184](#)
  - res3, [1184](#)
  - SBK, [1185](#)
  - SCEN, [1185](#)
  - STOP, [1186](#)
  - SR, [1186](#)
  - T8, [1186](#)
  - TCIEN, [1186](#)
  - TEN, [1187](#)
  - TIEN, [1187](#)
  - TXE, [1187](#)
  - TC, [1186](#)
  - UARTD, [1187](#)
  - WAKE, [1187](#)
- UART3\_AddressBase
  - STM8AF\_STM8S, [865–872](#)
- UART3\_t, [1188](#)
  - ADD, [1190](#)
  - BRR1, [1191](#)
  - BRR2, [1191](#)
  - CR1, [1191](#)
  - CR2, [1191](#)
  - CR3, [1191](#)
  - CR4, [1191](#)
  - CR6, [1192](#)
  - DATA, [1192](#)
  - DIV\_0\_3, [1192](#)

- DIV\_12\_15, [1192](#)
- DIV\_4\_11, [1192](#)
- DR, [1193](#)
- FE, [1193](#)
- IDLE, [1193](#)
- ILIEN, [1193](#)
- LASE, [1193](#)
- LBDIEN, [1194](#)
- LBDF, [1194](#)
- LBDL, [1194](#)
- LDUM, [1194](#)
- LHDIEN, [1195](#)
- LHDF, [1194](#)
- LINEN, [1195](#)
- LSLV, [1195](#)
- LSF, [1195](#)
- M, [1195](#)
- NF, [1196](#)
- OR, [1196](#)
- PCEN, [1196](#)
- PIEN, [1196](#)
- PE, [1196](#)
- PS, [1197](#)
- R8, [1197](#)
- REN, [1197](#)
- RIEN, [1198](#)
- RWU, [1198](#)
- RXNE, [1198](#)
- res, [1197](#)
- res2, [1198](#)
- SBK, [1198](#)
- STOP, [1199](#)
- SR, [1199](#)
- T8, [1199](#)
- TCIEN, [1199](#)
- TEN, [1200](#)
- TIEN, [1200](#)
- TXE, [1200](#)
- TC, [1199](#)
- UARTD, [1200](#)
- WAKE, [1200](#)
- UART4\_AddressBase
  - STM8AF\_STM8S, [872](#), [873](#)
- UART4\_t, [1201](#)
  - ADD, [1204](#)
  - BRR1, [1204](#)
  - BRR2, [1204](#)
  - CKEN, [1204](#)
  - CPHA, [1205](#)
  - CPOL, [1205](#)
  - CR1, [1205](#)
  - CR2, [1205](#)
  - CR3, [1205](#)
  - CR4, [1206](#)
  - CR5, [1206](#)
  - CR6, [1206](#)
  - DATA, [1206](#)
  - DIV\_0\_3, [1206](#)
  - DIV\_12\_15, [1206](#)
  - DIV\_4\_11, [1207](#)
  - DR, [1207](#)
  - FE, [1207](#)
  - GTR, [1207](#)
  - GT, [1207](#)
  - HDSEL, [1208](#)
  - IDLE, [1208](#)
  - ILIEN, [1208](#)
  - IREN, [1208](#)
  - IRLP, [1208](#)
  - LASE, [1209](#)
  - LBCL, [1209](#)
  - LBDIEN, [1209](#)
  - LBDF, [1209](#)
  - LBDL, [1209](#)
  - LDUM, [1210](#)
  - LHDIEN, [1210](#)
  - LHDF, [1210](#)
  - LINEN, [1210](#)
  - LSLV, [1211](#)
  - LSF, [1210](#)
  - M, [1211](#)
  - NACK, [1211](#)
  - NF, [1211](#)
  - OR, [1211](#)
  - PCEN, [1212](#)
  - PIEN, [1212](#)
  - PSCR, [1213](#)
  - PSC, [1212](#)
  - PE, [1212](#)
  - PS, [1212](#)
  - R8, [1213](#)
  - REN, [1213](#)
  - RIEN, [1214](#)
  - RWU, [1214](#)
  - RXNE, [1214](#)
  - res, [1213](#)
  - res2, [1213](#)
  - SBK, [1214](#)
  - SCEN, [1214](#)
  - STOP, [1215](#)
  - SR, [1215](#)
  - T8, [1215](#)
  - TCIEN, [1215](#)
  - TEN, [1216](#)
  - TIEN, [1216](#)
  - TXE, [1216](#)
  - TC, [1215](#)
  - UARTD, [1216](#)
  - WAKE, [1216](#)
- UARTD
  - UART1\_t, [1171](#)
  - UART2\_t, [1187](#)
  - UART3\_t, [1200](#)
  - UART4\_t, [1216](#)
- UDIS
  - TIM1\_t, [1088](#)



- TIM2\_t, [1107](#)
- TIM3\_t, [1122](#)
- TIM4\_t, [1127](#)
- TIM5\_t, [1150](#)
- TIM6\_t, [1156](#)
- UID\_AddressBase
  - STM8AF\_STM8S, [873–879](#)
- UIE
  - TIM1\_t, [1088](#)
  - TIM2\_t, [1107](#)
  - TIM3\_t, [1123](#)
  - TIM4\_t, [1128](#)
  - TIM5\_t, [1150](#)
  - TIM6\_t, [1157](#)
- UIF
  - TIM1\_t, [1088](#)
  - TIM2\_t, [1108](#)
  - TIM3\_t, [1123](#)
  - TIM4\_t, [1128](#)
  - TIM5\_t, [1150](#)
  - TIM6\_t, [1157](#)
- URS
  - TIM1\_t, [1088](#)
  - TIM2\_t, [1108](#)
  - TIM3\_t, [1123](#)
  - TIM4\_t, [1128](#)
  - TIM5\_t, [1150](#)
  - TIM6\_t, [1157](#)
- UG
  - TIM1\_t, [1088](#)
  - TIM2\_t, [1107](#)
  - TIM3\_t, [1122](#)
  - TIM4\_t, [1128](#)
  - TIM5\_t, [1150](#)
  - TIM6\_t, [1157](#)
- VECT10SPR
  - ITC\_t, [1026](#)
- VECT11SPR
  - ITC\_t, [1026](#)
- VECT12SPR
  - ITC\_t, [1026](#)
- VECT13SPR
  - ITC\_t, [1026](#)
- VECT14SPR
  - ITC\_t, [1027](#)
- VECT15SPR
  - ITC\_t, [1027](#)
- VECT16SPR
  - ITC\_t, [1027](#)
- VECT17SPR
  - ITC\_t, [1027](#)
- VECT18SPR
  - ITC\_t, [1027](#)
- VECT19SPR
  - ITC\_t, [1028](#)
- VECT1SPR
  - ITC\_t, [1028](#)
- VECT20SPR
  - ITC\_t, [1028](#)
- VECT21SPR
  - ITC\_t, [1028](#)
- VECT22SPR
  - ITC\_t, [1028](#)
- VECT23SPR
  - ITC\_t, [1029](#)
- VECT24SPR
  - ITC\_t, [1029](#)
- VECT25SPR
  - ITC\_t, [1029](#)
- VECT26SPR
  - ITC\_t, [1029](#)
- VECT27SPR
  - ITC\_t, [1029](#)
- VECT28SPR
  - ITC\_t, [1030](#)
- VECT29SPR
  - ITC\_t, [1030](#)
- VECT2SPR
  - ITC\_t, [1030](#)
- VECT3SPR
  - ITC\_t, [1030](#)
- VECT4SPR
  - ITC\_t, [1030](#)
- VECT5SPR
  - ITC\_t, [1031](#)
- VECT6SPR
  - ITC\_t, [1031](#)
- VECT7SPR
  - ITC\_t, [1031](#)
- VECT8SPR
  - ITC\_t, [1031](#)
- VECT9SPR
  - ITC\_t, [1031](#)
- W
  - WWDG\_t, [1218](#)
- WAIT\_FOR\_INTERRUPT
  - STM8AF\_STM8S, [879](#)
- WAKE
  - UART1\_t, [1171](#)
  - UART2\_t, [1187](#)
  - UART3\_t, [1200](#)
  - UART4\_t, [1216](#)
- WDGA
  - WWDG\_t, [1218](#)
- WKIE
  - SPI\_t, [1051](#)
- WKUIE
  - CAN\_t, [981](#)
- WKUI
  - CAN\_t, [981](#)
- WKUP
  - SPI\_t, [1051](#)
- WPRG
  - FLASH\_t, [1008](#)
- WPB
  - FLASH\_t, [1008](#)

WR\_PG\_DIS  
    FLASH\_t, [1008](#)  
WUFH  
    I2C\_t, [1022](#)  
WWDG\_AddressBase  
    STM8AF\_STM8S, [879–890](#)  
WWDG\_t, [1217](#)  
    CR, [1217](#)  
    res, [1218](#)  
    T, [1218](#)  
    W, [1218](#)  
    WDGA, [1218](#)  
    WR, [1218](#)  
WWDGF  
    RST\_t, [1041](#)  
WR  
    WWDG\_t, [1218](#)