VTS

1.0

Generated by Doxygen 1.8.16

1 Data Structure Index 1

1.1 Data Structures . . . 1

2 File Index 3

2.1 File List . . . 3

3 Data Structure Documentation 5

3.1 GPS_RMC Struct Reference . . . 5

3.1.1 Field Documentation . . . 5

3.1.1.1 Angle . . . 5

3.1.1.2 Date . . . 5

3.1.1.3 E_W . . . 6

3.1.1.4 Latitude . . . 6

3.1.1.5 Longitude . . . 6

3.1.1.6 N_S . . . 6

3.1.1.7 UTC_Time . . . 6

3.2 USART_type Struct Reference . . . 6

3.2.1 Field Documentation . . . 7

3.2.1.1 BRR . . . 7

3.2.1.2 CR1 . . . 7

3.2.1.3 CR2 . . . 7

3.2.1.4 CR3 . . . 7

3.2.1.5 DR . . . 7

3.2.1.6 GTPR . . . 7

3.2.1.7 SR . . . 7

4 File Documentation 9

4.1 Drivers/00 LIB/Bit_Math.h File Reference . . . 9

4.1.1 Detailed Description . . . 9

4.1.2 Macro Definition Documentation . . . 10

4.1.2.1 CLR_BIT . . . 10

4.1.2.2 GET_BIT . . . 10

4.1.2.3 SET_BIT . . . 10

4.1.2.4 TOG_BIT . . . 10

4.2 Drivers/00 LIB/Common_function.c File Reference . . . 10

4.2.1 Detailed Description . . . 11

4.2.2 Function Documentation . . . 11

4.2.2.1 _delay_ms() . . . 11

4.3 Drivers/00 LIB/Common_Function.h File Reference . . . 12

4.3.1 Detailed Description . . . 13

4.3.2 Function Documentation . . . 13

4.3.2.1 _delay_ms() . . . 13

4.4 Drivers/00 LIB/Std_Types.h File Reference . . . 13

4.4.1 Detailed Description . . . 14
4.4.2 Typedef Documentation . . . 14
4.4.2.1 f32 . . . 14
4.4.2.2 f64 . . . 14
4.4.2.3 s16 . . . 15
4.4.2.4 s32 . . . 15
4.4.2.5 s64 . . . 15
4.4.2.6 s8 . . . 15
4.4.2.7 u16 . . . 15
4.4.2.8 u32 . . . 15
4.4.2.9 u64 . . . 15
4.4.2.10 u8 . . . 15
4.5 Drivers/01 MCAL/00 RCC/RCC_config.h File Reference . . . 16
4.5.1 Detailed Description . . . 17
4.5.2 Macro Definition Documentation . . . 17
4.5.2.1 ADC_prescaler . . . 17
4.5.2.2 AHB_prescaler . . . 17
4.5.2.3 APB1_prescaler . . . 17
4.5.2.4 APB2_prescaler . . . 17
4.5.2.5 PLL_clocksource . . . 18
4.5.2.6 pll_Multiplication_factor . . . 18
4.5.2.7 RCC_SYS_CLK . . . 18
4.5.3 Enumeration Type Documentation . . . 18
4.5.3.1 anonymous enum . . . 18
4.5.3.2 anonymous enum . . . 18
4.5.3.3 anonymous enum . . . 19
4.5.3.4 anonymous enum . . . 19
4.5.3.5 clocksource . . . 19
4.5.3.6 pll_multiplication . . . 20
4.5.3.7 SYS_CLK . . . 20
4.6 Drivers/01 MCAL/00 RCC/RCC_interface.h File Reference . . . 20
4.6.1 Detailed Description . . . 22
4.6.2 Macro Definition Documentation . . . 23
4.6.2.1 RCC_AHB . . . 23
4.6.2.2 RCC_APB1 . . . 23
4.6.2.3 RCC_APB2 . . . 23
4.6.3 Enumeration Type Documentation . . . 23
4.6.3.1 anonymous enum . . . 23
4.6.3.2 anonymous enum . . . 23
4.6.3.3 anonymous enum . . . 24
4.6.3.4 prescalers . . . 25
4.6.3.5 RCC_cfgr . . . 25

4.6.3.6 RCC_cr . . . 26
4.6.4 Function Documentation . . . 26
4.6.4.1 RCC_ADC_SETprescaler() . . . 26
4.6.4.2 RCC_intalizeHSE() . . . 26
4.6.4.3 RCC_IntalizeHSI() . . . 27
4.6.4.4 RCC_intalizePLL() . . . 27
4.6.4.5 RCC_voidDisablePerClk() . . . 27
4.6.4.6 RCC_voidEnablePerClk() . . . 28
4.6.4.7 RCC_voidSetADCprescaler() . . . 28
4.6.4.8 RCC_voidSetAHBprescaler() . . . 28
4.6.4.9 RCC_voidSetAPB1prescaler() . . . 28
4.6.4.10 RCC_voidSetAPB2prescaler() . . . 29
4.6.4.11 RCC_voidSysClkInt() . . . 29
4.7 Drivers/01 MCAL/00 RCC/RCC_private.h File Reference . . . 29
4.7.1 Detailed Description . . . 30
4.7.2 Macro Definition Documentation . . . 31
4.7.2.1 RCC_AHBENR . . . 31
4.7.2.2 RCC_APB1ENR . . . 31
4.7.2.3 RCC_APB1RSTR . . . 31
4.7.2.4 RCC_APB2ENR . . . 31
4.7.2.5 RCC_APB2RSTR . . . 31
4.7.2.6 RCC_BASE_ADD . . . 31
4.7.2.7 RCC_BDCR . . . 31
4.7.2.8 RCC_CFGR . . . 32
4.7.2.9 RCC_CIR . . . 32
4.7.2.10 RCC_CR . . . 32
4.7.2.11 RCC_CSR . . . 32
4.8 Drivers/01 MCAL/00 RCC/RCC_program.c File Reference . . . 32
4.8.1 Detailed Description . . . 33
4.8.2 Function Documentation . . . 33
4.8.2.1 RCC_ADC_SETprescaler() . . . 33
4.8.2.2 RCC_intalizeHSE() . . . 34
4.8.2.3 RCC_IntalizeHSI() . . . 34
4.8.2.4 RCC_intalizePLL() . . . 34
4.8.2.5 RCC_voidDisablePerClk() . . . 34
4.8.2.6 RCC_voidEnablePerClk() . . . 35
4.8.2.7 RCC_voidSetADCprescaler() . . . 35
4.8.2.8 RCC_voidSetAHBprescaler() . . . 35
4.8.2.9 RCC_voidSetAPB1prescaler() . . . 35
4.8.2.10 RCC_voidSetAPB2prescaler() . . . 36
4.8.2.11 RCC_voidSysClkInt() . . . 36
4.9 Drivers/01 MCAL/01 GPIO/DIO_interface.h File Reference . . . 36

4.9.1 Detailed Description . . . 38
4.9.2 Macro Definition Documentation . . . 39
4.9.2.1 GPIO_HIGH . . . 39
4.9.2.2 GPIO_INPUT_ANALOG . . . 39
4.9.2.3 GPIO_INPUT_FLOAT . . . 39
4.9.2.4 GPIO_INPUT_PULL_UP_DOWN . . . 39
4.9.2.5 GPIO_LOW . . . 39
4.9.2.6 GPIO_OUTPUT_10MHZ_AFOD . . . 39
4.9.2.7 GPIO_OUTPUT_10MHZ_AFPP . . . 39
4.9.2.8 GPIO_OUTPUT_10MHZ_OD . . . 40
4.9.2.9 GPIO_OUTPUT_10MHZ_PP . . . 40
4.9.2.10 GPIO_OUTPUT_2MHZ_AFOD . . . 40
4.9.2.11 GPIO_OUTPUT_2MHZ_AFPP . . . 40
4.9.2.12 GPIO_OUTPUT_2MHZ_OD . . . 40
4.9.2.13 GPIO_OUTPUT_2MHZ_PP . . . 40
4.9.2.14 GPIO_OUTPUT_50MHZ_AFOD . . . 40
4.9.2.15 GPIO_OUTPUT_50MHZ_AFPP . . . 40
4.9.2.16 GPIO_OUTPUT_50MHZ_OD . . . 41
4.9.2.17 GPIO_OUTPUT_50MHZ_PP . . . 41
4.9.2.18 PIN0 . . . 41
4.9.2.19 PIN1 . . . 41
4.9.2.20 PIN10 . . . 41
4.9.2.21 PIN11 . . . 41
4.9.2.22 PIN12 . . . 41
4.9.2.23 PIN13 . . . 41
4.9.2.24 PIN14 . . . 42
4.9.2.25 PIN15 . . . 42
4.9.2.26 PIN2 . . . 42
4.9.2.27 PIN3 . . . 42
4.9.2.28 PIN4 . . . 42
4.9.2.29 PIN5 . . . 42
4.9.2.30 PIN6 . . . 42
4.9.2.31 PIN7 . . . 42
4.9.2.32 PIN8 . . . 43
4.9.2.33 PIN9 . . . 43
4.9.2.34 PORTA . . . 43
4.9.2.35 PORTB . . . 43
4.9.2.36 PORTC . . . 43
4.9.3 Function Documentation . . . 43
4.9.3.1 DIO_u8GetPinValue() . . . 43
4.9.3.2 DIO_voidSetPinDirection() . . . 44
4.9.3.3 DIO_voidSetPinValue() . . . 44

4.9.3.4 DIO_voidtogglepin() . . . 44
4.9.3.5 DIO_voidWriteValue() . . . 45
4.10 Drivers/01 MCAL/01 GPIO/DIO_private.h File Reference . . . 45
4.10.1 Detailed Description . . . 46
4.10.2 Macro Definition Documentation . . . 46
4.10.2.1 GPIOA_BASE_ADD . . . 46
4.10.2.2 GPIOA_BRR . . . 47
4.10.2.3 GPIOA_BSRR . . . 47
4.10.2.4 GPIOA_CRH . . . 47
4.10.2.5 GPIOA_CRL . . . 47
4.10.2.6 GPIOA_IDR . . . 47
4.10.2.7 GPIOA_LCKR . . . 47
4.10.2.8 GPIOA_ODR . . . 47
4.10.2.9 GPIOB_BASE_ADD . . . 47
4.10.2.10 GPIOB_BRR . . . 48
4.10.2.11 GPIOB_BSRR . . . 48
4.10.2.12 GPIOB_CRH . . . 48
4.10.2.13 GPIOB_CRL . . . 48
4.10.2.14 GPIOB_IDR . . . 48
4.10.2.15 GPIOB_LCKR . . . 48
4.10.2.16 GPIOB_ODR . . . 48
4.10.2.17 GPIOC_BASE_ADD . . . 48
4.10.2.18 GPIOC_BRR . . . 49
4.10.2.19 GPIOC_BSRR . . . 49
4.10.2.20 GPIOC_CRH . . . 49
4.10.2.21 GPIOC_CRL . . . 49
4.10.2.22 GPIOC_IDR . . . 49
4.10.2.23 GPIOC_LCKR . . . 49
4.10.2.24 GPIOC_ODR . . . 49
4.11 Drivers/01 MCAL/01 GPIO/DIO_program.c File Reference . . . 50
4.11.1 Detailed Description . . . 50
4.11.2 Function Documentation . . . 51
4.11.2.1 DIO_u8GetPinValue() . . . 51
4.11.2.2 DIO_voidSetPinDirection() . . . 51
4.11.2.3 DIO_voidSetPinValue() . . . 52
4.11.2.4 DIO_voidtogglepin() . . . 52
4.11.2.5 DIO_voidWriteValue() . . . 52
4.12 Drivers/01 MCAL/02 NVIC/NVIC_config.h File Reference . . . 53
4.12.1 Detailed Description . . . 53
4.12.2 Macro Definition Documentation . . . 53
4.12.2.1 NVIC_NUM_GRP_SUB . . . 54
4.13 Drivers/01 MCAL/02 NVIC/NVIC_interface.h File Reference . . . 54

4.13.1 Detailed Description . . . 54
4.13.2 Function Documentation . . . 55
4.13.2.1 NVIC_u8ReadActiveFlag() . . . 55
4.13.2.2 NVIC_voidClrPendingFlag() . . . 55
4.13.2.3 NVIC_voidDisablePerInt() . . . 55
4.13.2.4 NVIC_voidEnablePerInt() . . . 55
4.13.2.5 NVIC_voidInit() . . . 55
4.13.2.6 NVIC_voidSetPendingFlag() . . . 55
4.13.2.7 NVIC_voidSetSwIntPriority() . . . 56
4.14 Drivers/01 MCAL/02 NVIC/NVIC_private.h File Reference . . . 56
4.14.1 Detailed Description . . . 57
4.14.2 Macro Definition Documentation . . . 57
4.14.2.1 GRB_00_SUB_16 . . . 57
4.14.2.2 GRB_02_SUB_08 . . . 57
4.14.2.3 GRB_04_SUB_04 . . . 57
4.14.2.4 GRB_08_SUB_02 . . . 57
4.14.2.5 GRB_16_SUB_00 . . . 58
4.14.2.6 NVIC_BASE_ADD . . . 58
4.14.2.7 NVIC_IABR0 . . . 58
4.14.2.8 NVIC_IABR1 . . . 58
4.14.2.9 NVIC_ICER0 . . . 58
4.14.2.10 NVIC_ICER1 . . . 58
4.14.2.11 NVIC_ICPR0 . . . 58
4.14.2.12 NVIC_ICPR1 . . . 58
4.14.2.13 NVIC_IPR . . . 59
4.14.2.14 NVIC_ISER0 . . . 59
4.14.2.15 NVIC_ISER1 . . . 59
4.14.2.16 NVIC_ISPR0 . . . 59
4.14.2.17 NVIC_ISPR1 . . . 59
4.15 Drivers/01 MCAL/02 NVIC/NVIC_program.c File Reference . . . 59
4.15.1 Detailed Description . . . 60
4.15.2 Macro Definition Documentation . . . 60
4.15.2.1 SCB_AIRCR . . . 60
4.15.2.2 SCB_BASE_ADD . . . 60
4.15.3 Function Documentation . . . 61
4.15.3.1 NVIC_u8ReadActiveFlag() . . . 61
4.15.3.2 NVIC_voidClrPendingFlag() . . . 61
4.15.3.3 NVIC_voidDisablePerInt() . . . 61
4.15.3.4 NVIC_voidEnablePerInt() . . . 61
4.15.3.5 NVIC_voidInit() . . . 61
4.15.3.6 NVIC_voidSetPendingFlag() . . . 61
4.15.3.7 NVIC_voidSetSwIntPriority() . . . 62

4.16 Drivers/01 MCAL/06 USART/USART_config.h File Reference . . . 62
4.16.1 Detailed Description . . . 62
4.16.2 Macro Definition Documentation . . . 63
4.16.2.1 USART1_BAUDRATE . . . 63
4.16.2.2 USART1_DATA_LENGTH . . . 63
4.16.2.3 USART1_PARITY . . . 63
4.16.2.4 USART1_PARITY_TYPE . . . 63
4.16.2.5 USART1_STOP_LENGTH . . . 63
4.16.2.6 USART2_BAUDRATE . . . 64
4.16.2.7 USART2_DATA_LENGTH . . . 64
4.16.2.8 USART2_PARITY . . . 64
4.16.2.9 USART2_PARITY_TYPE . . . 64
4.16.2.10 USART2_STOP_LENGTH . . . 64
4.17 Drivers/01 MCAL/06 USART/USART_interface.h File Reference . . . 64
4.17.1 Detailed Description . . . 65
4.17.2 Function Documentation . . . 66
4.17.2.1 UART1_enableRX() . . . 66
4.17.2.2 UART1_enableTX() . . . 66
4.17.2.3 UART1_init() . . . 66
4.17.2.4 UART1_RXdata() . . . 66
4.17.2.5 UART1_StrReceiveString() . . . 67
4.17.2.6 UART1_TXdata() . . . 67
4.17.2.7 UART1_voidSendString() . . . 67
4.17.2.8 UART2_enableRX() . . . 68
4.17.2.9 UART2_enableTX() . . . 68
4.17.2.10 UART2_init() . . . 68
4.17.2.11 UART2_RX_CALLBACK() . . . 68
4.17.2.12 UART2_RXdata() . . . 69
4.17.2.13 UART2_StrReceiveString() . . . 69
4.17.2.14 UART2_TXdata() . . . 69
4.17.2.15 UART2_voidSendString() . . . 70
4.17.2.16 USART2_IRQHandler() . . . 70
4.18 Drivers/01 MCAL/06 USART/USART_private.h File Reference . . . 70
4.18.1 Detailed Description . . . 72
4.18.2 Macro Definition Documentation . . . 73
4.18.2.1 ADD . . . 73
4.18.2.2 BAUD_RATE_115200 . . . 73
4.18.2.3 BAUD_RATE_9600 . . . 73
4.18.2.4 CLKEN . . . 73
4.18.2.5 CPHA . . . 73
4.18.2.6 CPOL . . . 73
4.18.2.7 CTS . . . 73

4.18.2.8 CTSE . . . 74
4.18.2.9 CTSIE . . . 74
4.18.2.10 DATA_LEN_8 . . . 74
4.18.2.11 DATA_LEN_9 . . . 74
4.18.2.12 DMAR . . . 74
4.18.2.13 DMAT . . . 74
4.18.2.14 EIE . . . 74
4.18.2.15 FE . . . 74
4.18.2.16 GT . . . 75
4.18.2.17 HDSEL . . . 75
4.18.2.18 IDLE . . . 75
4.18.2.19 IDLEIE . . . 75
4.18.2.20 IREN . . . 75
4.18.2.21 IRLP . . . 75
4.18.2.22 LBCL . . . 75
4.18.2.23 LBD . . . 75
4.18.2.24 LBDIE . . . 76
4.18.2.25 LBDL . . . 76
4.18.2.26 LINEN . . . 76
4.18.2.27 M . . . 76
4.18.2.28 NACK . . . 76
4.18.2.29 NE . . . 76
4.18.2.30 ORE . . . 76
4.18.2.31 PARITY_EVEN . . . 76
4.18.2.32 PARITY_NO . . . 77
4.18.2.33 PARITY_ODD . . . 77
4.18.2.34 PARITY_YES . . . 77
4.18.2.35 PCE . . . 77
4.18.2.36 PE . . . 77
4.18.2.37 PEIE . . . 77
4.18.2.38 PS . . . 77
4.18.2.39 PSC . . . 77
4.18.2.40 RE . . . 78
4.18.2.41 RTSE . . . 78
4.18.2.42 RWU . . . 78
4.18.2.43 RXNE . . . 78
4.18.2.44 RXNEIE . . . 78
4.18.2.45 SBK . . . 78
4.18.2.46 SCEN . . . 78
4.18.2.47 STOP . . . 78
4.18.2.48 STOP_BIT_1 . . . 79
4.18.2.49 STOP_BIT_2 . . . 79

4.18.2.50 TC . . . 79
4.18.2.51 TCEIE . . . 79
4.18.2.52 TCIE . . . 79
4.18.2.53 TE . . . 79
4.18.2.54 TXE . . . 79
4.18.2.55 UE . . . 79
4.18.2.56 USART1 . . . 80
4.18.2.57 USART2 . . . 80
4.18.2.58 WAKe . . . 80
4.19 Drivers/01 MCAL/06 USART/USART_program.c File Reference . . . 80
4.19.1 Detailed Description . . . 81
4.19.2 Function Documentation . . . 82
4.19.2.1 UART1_enableRX() . . . 82
4.19.2.2 UART1_enableTX() . . . 82
4.19.2.3 UART1_init() . . . 82
4.19.2.4 UART1_RXdata() . . . 82
4.19.2.5 UART1_StrReceiveString() . . . 83
4.19.2.6 UART1_TXdata() . . . 83
4.19.2.7 UART1_voidSendString() . . . 83
4.19.2.8 UART2_enableRX() . . . 84
4.19.2.9 UART2_enableTX() . . . 84
4.19.2.10 UART2_init() . . . 84
4.19.2.11 UART2_RX_CALLBACK() . . . 84
4.19.2.12 UART2_RXdata() . . . 85
4.19.2.13 UART2_StrReceiveString() . . . 85
4.19.2.14 UART2_TXdata() . . . 85
4.19.2.15 UART2_voidSendString() . . . 86
4.19.2.16 USART2_IRQHandler() . . . 86
4.19.3 Variable Documentation . . . 86
4.19.3.1 UART2_PTR . . . 86
4.20 Drivers/02 ECU/Buttom/Buttom_Interface.h File Reference . . . 87
4.20.1 Function Documentation . . . 87
4.20.1.1 Buttom_u8GetValue() . . . 87
4.20.1.2 Buttom_voidInit() . . . 88
4.21 Drivers/02 ECU/Buttom/Buttom_program.c File Reference . . . 88
4.21.1 Function Documentation . . . 88
4.21.1.1 Buttom_u8GetValue() . . . 89
4.21.1.2 Buttom_voidInit() . . . 89
4.22 Drivers/02 ECU/GPS/GPS_Interface.h File Reference . . . 89
4.22.1 Detailed Description . . . 91
4.22.2 Macro Definition Documentation . . . 91
4.22.2.1 Buffer_size . . . 91

4.22.2.2 FRAME . . . 91
4.22.3 Typedef Documentation . . . 91
4.22.3.1 GPS_RMC_ST . . . 91
4.22.4 Function Documentation . . . 91
4.22.4.1 GPS_Decode_Buffer() . . . 92
4.22.4.2 GPS_Decode_Buffer_st() . . . 92
4.22.4.3 GPS_Frame_Concat() . . . 92
4.22.4.4 GPS_Init() . . . 92
4.22.5 Variable Documentation . . . 92
4.22.5.1 GPS_Complete_FRAME . . . 93
4.23 Drivers/02 ECU/GPS/GPS_Private.h File Reference . . . 93
4.23.1 Detailed Description . . . 93
4.23.2 Function Documentation . . . 94
4.23.2.1 GPS_Recieve_Buffer() . . . 94
4.23.3 Variable Documentation . . . 94
4.23.3.1 GPS_Buffer . . . 94
4.24 Drivers/02 ECU/GPS/GPS_Program.c File Reference . . . 94
4.24.1 Detailed Description . . . 95
4.24.2 Function Documentation . . . 95
4.24.2.1 GPS_Decode_Buffer_st() . . . 95
4.24.2.2 GPS_Frame_Concat() . . . 95
4.24.2.3 GPS_Init() . . . 96
4.24.2.4 GPS_Recieve_Buffer() . . . 96
4.24.3 Variable Documentation . . . 96
4.24.3.1 GPS_Complete_FRAME . . . 96
4.25 Drivers/02 ECU/GSM/GSM_Config.h File Reference . . . 97
4.25.1 Detailed Description . . . 97
4.25.2 Macro Definition Documentation . . . 97
4.25.2.1 NUMBER . . . 98
4.25.2.2 URL . . . 98
4.26 Drivers/02 ECU/GSM/GSM_Interface.h File Reference . . . 98
4.26.1 Detailed Description . . . 99
4.26.2 Function Documentation . . . 99
4.26.2.1 GSM_INIT() . . . 99
4.26.2.2 GSM_MakeCall() . . . 100
4.26.2.3 GSM_Post() . . . 100
4.26.2.4 GSM_SendSMS() . . . 101
4.27 Drivers/02 ECU/GSM/GSM_Private.h File Reference . . . 102
4.27.1 Detailed Description . . . 102
4.27.2 Macro Definition Documentation . . . 103
4.27.2.1 CALL_CONC [1/2] . . . 103
4.27.2.2 CALL_CONC [2/2] . . . 103

4.27.2.3 SMS_CONC [1/2] . . . 103
4.27.2.4 SMS_CONC [2/2] . . . 103
4.27.2.5 URL_CONC [1/2] . . . 103
4.27.2.6 URL_CONC [2/2] . . . 104
4.27.3 Function Documentation . . . 104
4.27.3.1 GSM_AdjustData_Json() . . . 104
4.27.3.2 GSM_INIT_GPRS_MODE() . . . 104
4.27.3.3 GSM_Init_HTTP() . . . 104
4.27.3.4 PlainToJSON() . . . 104
4.27.3.5 SMS_INIT() . . . 104
4.27.3.6 waitforDOWNLOAD() . . . 105
4.27.3.7 waitforHTTPACTION() . . . 105
4.27.3.8 waitforok() . . . 105
4.28 Drivers/02 ECU/GSM/GSM_Program.c File Reference . . . 105
4.28.1 Detailed Description . . . 106
4.28.2 Function Documentation . . . 106
4.28.2.1 GSM_AdjustData_Json() . . . 107
4.28.2.2 GSM_INIT() . . . 107
4.28.2.3 GSM_INIT_GPRS_MODE() . . . 108
4.28.2.4 GSM_Init_HTTP() . . . 108
4.28.2.5 GSM_MakeCall() . . . 109
4.28.2.6 GSM_Post() . . . 109
4.28.2.7 GSM_SendSMS() . . . 110
4.28.2.8 PlainToJSON() . . . 111
4.28.2.9 SMS_INIT() . . . 111
4.28.2.10 waitforDOWNLOAD() . . . 111
4.28.2.11 waitforHTTPACTION() . . . 112
4.28.2.12 waitforok() . . . 112
4.29 Drivers/02 ECU/IR/IR_Interface.h File Reference . . . 113
4.29.1 Function Documentation . . . 113
4.29.1.1 IR_u8GetValue() . . . 113
4.29.1.2 IR_voidInit() . . . 114
4.30 Drivers/02 ECU/IR/IR_program.c File Reference . . . 114
4.30.1 Function Documentation . . . 115
4.30.1.1 IR_u8GetValue() . . . 115
4.30.1.2 IR_voidInit() . . . 115
4.31 Src/main.c File Reference . . . 116
4.31.1 Detailed Description . . . 116
4.31.2 Function Documentation . . . 117
4.31.2.1 main() . . . 117
4.31.3 Variable Documentation . . . 117
4.31.3.1 RMC . . . 117

4.32 Src/syscalls.c File Reference . . . 118
4.32.1 Detailed Description . . . 119
4.32.2 Function Documentation . . . 119
4.32.2.1 __attribute__() . . . 119
4.32.2.2 __io_getchar() . . . 119
4.32.2.3 __io_putchar() . . . 120
4.32.2.4 _close() . . . 120
4.32.2.5 _execve() . . . 120
4.32.2.6 _exit() . . . 120
4.32.2.7 _fork() . . . 120
4.32.2.8 _fstat() . . . 121
4.32.2.9 _getpid() . . . 121
4.32.2.10 _isatty() . . . 121
4.32.2.11 _kill() . . . 121
4.32.2.12 _link() . . . 121
4.32.2.13 _lseek() . . . 121
4.32.2.14 _open() . . . 122
4.32.2.15 _stat() . . . 122
4.32.2.16 _times() . . . 122
4.32.2.17 _unlink() . . . 122
4.32.2.18 _wait() . . . 122
4.32.2.19 initialise_monitor_handles() . . . 122
4.32.3 Variable Documentation . . . 122
4.32.3.1 environ . . . 123
4.33 Src/sysmem.c File Reference . . . 123
4.33.1 Detailed Description . . . 123
4.33.2 Function Documentation . . . 124
4.33.2.1 _sbrk() . . . 124
4.33.3 Variable Documentation . . . 124
4.33.3.1 __sbrk_heap_end . . . 124

Index **125**

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

GPS_RMC 5
USART_type 6

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

Drivers/00 LIB/Bit_Math.h
Used to control bits . . . 9
Drivers/00 LIB/Common_function.c
Helper Fucntions . . . 10
Drivers/00 LIB/Common_Function.h
Contain APIs of common Function . . . 12
Drivers/00 LIB/Std_Types.h
Standard types to use in project . . . 13
Drivers/01 MCAL/00 RCC/RCC_config.h
RCC Confguration . . . 16
Drivers/01 MCAL/00 RCC/RCC_interface.h
RCC APIs and Global data . . . 20
Drivers/01 MCAL/00 RCC/RCC_private.h
RCC Private Data . . . 29
Drivers/01 MCAL/00 RCC/RCC_program.c
RCC Driver . . . 32
Drivers/01 MCAL/01 GPIO/DIO_interface.h
APIs of DIO . . . 36
Drivers/01 MCAL/01 GPIO/DIO_private.h
Private Data . . . 45
Drivers/01 MCAL/01 GPIO/DIO_program.c
DIO Implementaiton . . . 50
Drivers/01 MCAL/02 NVIC/NVIC_config.h
NVIC For ARM . . . 53
Drivers/01 MCAL/02 NVIC/NVIC_interface.h
NVIC APIs and Global data . . . 54
Drivers/01 MCAL/02 NVIC/NVIC_private.h
NVIC Private Data . . . 56
Drivers/01 MCAL/02 NVIC/NVIC_program.c
NVIC Driver Implementation . . . 59
Drivers/01 MCAL/06 USART/USART_config.h
Configuration of USART . . . 62
Drivers/01 MCAL/06 USART/USART_interface.h
APIs and Global data . . . 64
Drivers/01 MCAL/06 USART/USART_private.h
Private APIs and Data . . . 70

Drivers/01 MCAL/06 USART/USART_program.c
USART Driver Implementation . . . 80
Drivers/02 ECU/Buttom/Buttom_Interface.h . . . 87
Drivers/02 ECU/Buttom/Buttom_program.c . . . 88
Drivers/02 ECU/GPS/GPS_Interface.h
GPS APIs and Global data . . . 89
Drivers/02 ECU/GPS/GPS_Private.h
GPS Private Data . . . 93
Drivers/02 ECU/GPS/GPS_Program.c
Implement GPS Driver . . . 94
Drivers/02 ECU/GSM/GSM_Config.h
GSM Confguration of data . . . 97
Drivers/02 ECU/GSM/GSM_Interface.h
GSM APIs and Global Functions . . . 98
Drivers/02 ECU/GSM/GSM_Private.h
Private Data of GSM Module . . . 102
Drivers/02 ECU/GSM/GSM_Program.c
Used To Implement Functions of GSM . . . 105
Drivers/02 ECU/IR/IR_Interface.h . . . 113
Drivers/02 ECU/IR/IR_program.c . . . 114
Src/main.c
Main Function To test GPS Performance ON TTL . . . 116
Src/syscalls.c
STM32CubeIDE Minimal System calls file . . . 118
Src/sysmem.c
STM32CubeIDE System Memory calls file . . . 123

Chapter 3

Data Structure Documentation

3.1 GPS_RMC Struct Reference

`#include <GPS_Interface.h>`

Data Fields

- u8 UTC_Time [10]
- u8 Latitude [9]
- u8 Longitude [10]
- u8 Angle [6]
- u8 Date [6]
- u8 N_S
- u8 E_W

3.1.1 Field Documentation

3.1.1.1 Angle

`u8 GPS_RMC::Angle[6]`

3.1.1.2 Date

`u8 GPS_RMC::Date[6]`

3.1.1.3 E_W

```
u8 GPS_RMC::E_W
```

3.1.1.4 Latitude

```
u8 GPS_RMC::Latitude[9]
```

3.1.1.5 Longitude

```
u8 GPS_RMC::Longitude[10]
```

3.1.1.6 N_S

```
u8 GPS_RMC::N_S
```

3.1.1.7 UTC_Time

```
u8 GPS_RMC::UTC_Time[10]
```

The documentation for this struct was generated from the following file:

- Drivers/02 ECU/GPS/GPS_Interface.h

3.2 USART_type Struct Reference

```
#include <USART_private.h>
```

Data Fields

- u32 SR
- u32 DR
- u32 BRR
- u32 CR1
- u32 CR2
- u32 CR3
- u32 GTPR

3.2.1 Field Documentation

3.2.1.1 BRR

```
u32 USART_type::BRR
```

3.2.1.2 CR1

```
u32 USART_type::CR1
```

3.2.1.3 CR2

```
u32 USART_type::CR2
```

3.2.1.4 CR3

```
u32 USART_type::CR3
```

3.2.1.5 DR

```
u32 USART_type::DR
```

3.2.1.6 GTPR

```
u32 USART_type::GTPR
```

3.2.1.7 SR

```
u32 USART_type::SR
```

The documentation for this struct was generated from the following file:

- Drivers/01 MCAL/06 USART/USART_private.h

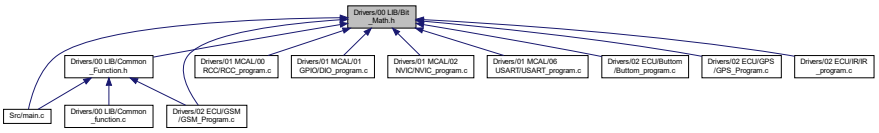
Chapter 4

File Documentation

4.1 Drivers/00 LIB/Bit_Math.h File Reference

used to control bits

This graph shows which files directly or indirectly include this file:



Macros

- #define SET_BIT(REG, BIT) (REG|= (1<<BIT))
- #define CLR_BIT(REG, BIT) (REG&= ~(1<<BIT))
- #define TOG_BIT(REG, BIT) (REG^= (1<<BIT))
- #define GET_BIT(REG, BIT) ((REG>>BIT)&0x01)

4.1.1 Detailed Description

used to control bits

Author

your name (you@domain.com)

Version

0.1

Date

2023-08-26

Copyright

4.1.2 Macro Definition Documentation

4.1.2.1 CLR_BIT

```
#define CLR_BIT(
              REG,
              BIT ) (REG&= ~(1<<BIT))
```

4.1.2.2 GET_BIT

```
#define GET_BIT(
              REG,
              BIT ) ((REG>>BIT)&0x01)
```

4.1.2.3 SET_BIT

```
#define SET_BIT(
              REG,
              BIT ) (REG|= (1<<BIT))
```

4.1.2.4 TOG_BIT

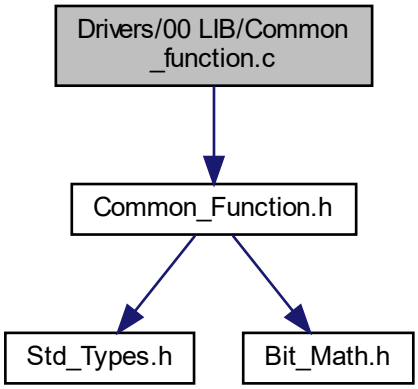
```
#define TOG_BIT(
              REG,
              BIT ) (REG^= (1<<BIT))
```

4.2 Drivers/00 LIB/Common_function.c File Reference

Helper Fucntions.

```
#include "Common_Function.h"
```

Include dependency graph for Common_function.c:



Functions

- void _delay_ms (u32 ticks)

  This Function is used to delay the task.

4.2.1 Detailed Description

Helper Fucntions.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.2.2 Function Documentation

4.2.2.1 _delay_ms()

```
void _delay_ms (
            u32 ticks )
```

This Function is used to delay the task.

Parameters

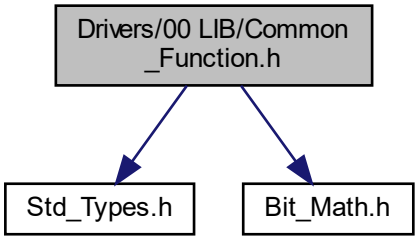
ticks	number of ms to wait

4.3 Drivers/00 LIB/Common_Function.h File Reference

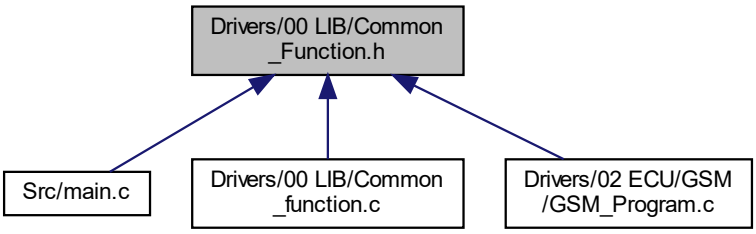
Contain APIs of common Function.

```
#include "Std_Types.h"
#include "Bit_Math.h"
```

Include dependency graph for Common_Function.h:



This graph shows which files directly or indirectly include this file:



Functions

- void _delay_ms (u32 ticks)

  This Function is used to delay the task.

4.3.1 Detailed Description

Contain APIs of common Function.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.3.2 Function Documentation

4.3.2.1 _delay_ms()

```
void _delay_ms (
            u32 ticks )
```

This Function is used to delay the task.

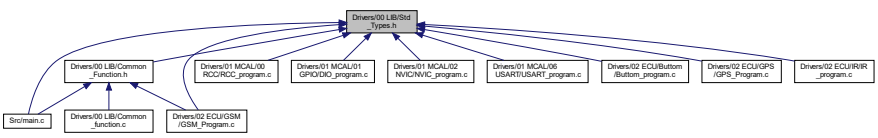
Parameters

ticks	number of ms to wait

4.4 Drivers/00 LIB/Std_Types.h File Reference

standard types to use in project

This graph shows which files directly or indirectly include this file:



Typedefs

- typedef unsigned char u8
- typedef unsigned short int u16
- typedef unsigned int u32
- typedef unsigned long int u64
- typedef float f32
- typedef double f64
- typedef signed char s8
- typedef signed short int s16
- typedef signed int s32
- typedef signed long int s64

4.4.1 Detailed Description

standard types to use in project

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.4.2 Typedef Documentation

4.4.2.1 f32

```
typedef float f32
```

4.4.2.2 f64

```
typedef double f64
```

4.4.2.3 s16

```
typedef signed short int s16
```

4.4.2.4 s32

```
typedef signed int s32
```

4.4.2.5 s64

```
typedef signed long int s64
```

4.4.2.6 s8

```
typedef signed char s8
```

4.4.2.7 u16

```
typedef unsigned short int u16
```

4.4.2.8 u32

```
typedef unsigned int u32
```

4.4.2.9 u64

```
typedef unsigned long int u64
```

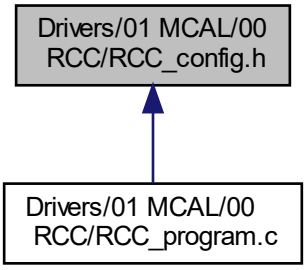
4.4.2.10 u8

```
typedef unsigned char u8
```

4.5 Drivers/01 MCAL/00 RCC/RCC_config.h File Reference

RCC Confguration.

This graph shows which files directly or indirectly include this file:



Macros

- #define RCC_SYS_CLK Rcc_HSI
- #define PLL_clocksource NO_PLL
- #define pll_Multiplication_factor pll_inputclockX2
- #define AHB_prescaler AHB_SysNotDiv
- #define APB1_prescaler APB1_HCLKNotDiv
- #define APB2_prescaler APB2_HCLKNotDiv
- #define ADC_prescaler ADC_APB2clkdiv2

Enumerations

- enum SYS_CLK { Rcc_HSI, Rcc_HSE, Rcc_PLL }
- enum clocksource { HALF_HSI, Full_HSE, Half_HSE, NO_PLL }
- enum pll_multiplication {
  pll_inputclockX2, pll_inputclockX3, pll_inputclockX4, pll_inputclockX5,
  pll_inputclockX6, pll_inputclockX7, pll_inputclockX8, pll_inputclockX9,
  pll_inputclockX10, pll_inputclockX11, pll_inputclockX12, pll_inputclockX13,
  pll_inputclockX14, pll_inputclockX15, pll_inputclockX16 }
- enum {
  AHB_SysNotDiv, AHB_SysDiv2, AHB_SysDiv4, AHB_SysDiv8,
  AHB_SysDiv16, AHB_SysDiv64, AHB_SysDiv128, AHB_SysDiv256,
  AHB_SysDiv512 }
- enum {
  APB1_HCLKNotDiv, APB1_HCLKDiv2, APB1_HCLKDiv4, APB1_HCLKDiv8,
  APB1_HCLKDiv16 }
- enum {
  APB2_HCLKNotDiv, APB2_HCLKDiv2, APB2_HCLKDiv4, APB2_HCLKDiv8,
  APB2_HCLKDiv16 }
- enum { ADC_APB2clkdiv2, ADC_APB2clkdiv4, ADC_APB2clkdiv6, ADC_APB2clkdiv8 }

4.5.1 Detailed Description

RCC Confguration.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.5.2 Macro Definition Documentation

4.5.2.1 ADC_prescaler

```
#define ADC_prescaler ADC_APB2clkdiv2
```

4.5.2.2 AHB_prescaler

```
#define AHB_prescaler AHB_SysNotDiv
```

4.5.2.3 APB1_prescaler

```
#define APB1_prescaler APB1_HCLKNotDiv
```

4.5.2.4 APB2_prescaler

```
#define APB2_prescaler APB2_HCLKNotDiv
```

4.5.2.5 PLL_clocksource

```
#define PLL_clocksource NO_PLL
```

4.5.2.6 pll_Multiplication_factor

```
#define pll_Multiplication_factor pll_inputclockX2
```

4.5.2.7 RCC_SYS_CLK

```
#define RCC_SYS_CLK Rcc_HSI
```

4.5.3 Enumeration Type Documentation

4.5.3.1 anonymous enum

```
anonymous enum
```

Enumerator

AHB_SysNotDiv	
AHB_SysDiv2	
AHB_SysDiv4	
AHB_SysDiv8	
AHB_SysDiv16	
AHB_SysDiv64	
AHB_SysDiv128	
AHB_SysDiv256	
AHB_SysDiv512	

4.5.3.2 anonymous enum

```
anonymous enum
```

Enumerator

APB1_HCLKNotDiv	

Enumerator

APB1_HCLKDiv2	
APB1_HCLKDiv4	
APB1_HCLKDiv8	
APB1_HCLKDiv16	

4.5.3.3 anonymous enum

```
anonymous enum
```

Enumerator

APB2_HCLKNotDiv	
APB2_HCLKDiv2	
APB2_HCLKDiv4	
APB2_HCLKDiv8	
APB2_HCLKDiv16	

4.5.3.4 anonymous enum

```
anonymous enum
```

Enumerator

ADC_APB2clkdiv2	
ADC_APB2clkdiv4	
ADC_APB2clkdiv6	
ADC_APB2clkdiv8	

4.5.3.5 clocksource

```
enum clocksource
```

Enumerator

HALF_HSI	
Full_HSE	
Half_HSE	
NO_PLL	

4.5.3.6 pll_multiplication

enum pll_multiplication

Enumerator

pll_inputclockX2	
pll_inputclockX3	
pll_inputclockX4	
pll_inputclockX5	
pll_inputclockX6	
pll_inputclockX7	
pll_inputclockX8	
pll_inputclockX9	
pll_inputclockX10	
pll_inputclockX11	
pll_inputclockX12	
pll_inputclockX13	
pll_inputclockX14	
pll_inputclockX15	
pll_inputclockX16	

4.5.3.7 SYS_CLK

enum SYS_CLK

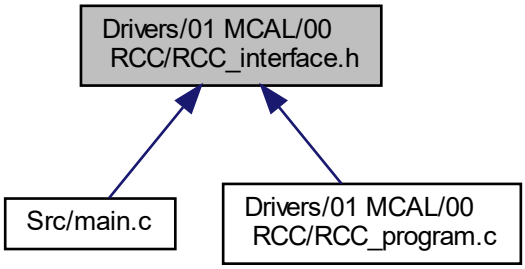
Enumerator

Rcc_HSI	
Rcc_HSE	
Rcc_PLL	

4.6 Drivers/01 MCAL/00 RCC/RCC_interface.h File Reference

RCC APIs and Global data.

This graph shows which files directly or indirectly include this file:



Macros

- #define RCC_AHB 0
- #define RCC_APB1 1
- #define RCC_APB2 2

Enumerations

- enum RCC_cr {
  HSION, HSIRDY, HSITRIM0 = 3, HSITRIM1,
  HSITRIM2, HSITRIM3, HSITRIM4, HSICAL0,
  HSICAL1, HSICAL2, HSICAL3, HSICAL4,
  HSICAL5, HSICAL6, HSICAL7, HSEON,
  HSERDY, HSEBYP, CSSON, PLLON = 24,
  PLLRDY }
- enum RCC_cfgr {
  SW0, SW1, SWS0, SWS1,
  HPRE0, HPRE1, HPRE2, HPRE3,
  PPRE10, PPRE11, PPRE12, PPRE20,
  PPRE21, PPRE22, ADCPRE0, ADCPRE1,
  PLLSRC, PLLXTPRE, PLLMUL0, PLLMUL1,
  PLLMUL2, PLLMUL3, USBPRE, MCO0 = 24,
  MCO1, MCO2 }
- enum {
  RCC_AHB_DMA1, RCC_AHB_DMA2, RCC_AHB_SRAM, RCC_AHB_FLITF =4,
  RCC_AHB_CRC =6, RCC_AHB_FSMC =8, RCC_AHB_SDIO =10 }
- enum {
  RCC_APB1_TIM2, RCC_APB1_TIM3, RCC_APB1_TIM4, RCC_APB1_TIM5,
  RCC_APB1_TIM6, RCC_APB1_TIM7, RCC_APB1_TIM12, RCC_APB1_TIM13,
  RCC_APB1_TIM14, RCC_APB1_WWWG =11, RCC_APB1_SPI2 =14, RCC_APB1_SPI3,
  RCC_APB1_USART2 =17, RCC_APB1_USART3, RCC_APB1_USART4, RCC_APB1_USART5,
  RCC_APB1_I2C1, RCC_APB1_I2C2, RCC_APB1_USB, RCC_APB1_CAN =25,
  RCC_APB1_BKP =27, RCC_APB1_PWR, RCC_APB1_DAC }

- enum {
  RCC_APB2_AFIO, RCC_APB2_DIOA =2, RCC_APB2_DIOB, RCC_APB2_DIOC,
  RCC_APB2_DIOD, RCC_APB2_DIOE, RCC_APB2_DIOF, RCC_APB2_DIOG,
  RCC_APB2_ADC1, RCC_APB2_ADC2, RCC_APB2_TIM1, RCC_APB2_SPI1,
  RCC_APB2_TIM8, RCC_APB2_USART1, RCC_APB2_ADC3, RCC_APB2_TIM9 =19,
  RCC_APB2_TIM10, RCC_APB2_TIM11 }
- enum prescalers { divided2, divided4, divided6, divided8 }

Functions

- void RCC_IntalizeHSI (void)

  Initialize Internal crystal.
- void RCC_intalizeHSE (void)

  Initilaize External Crystal.
- void RCC_intalizePLL (void)

  Initialize PLL.
- void RCC_voidSysClkInt (void)

  Initialize System Clock.
- void RCC_voidEnablePerClk (u8 BusId, u8 PerId)

  Enable Peripheral Clock.
- void RCC_voidDisablePerClk (u8 BusId, u8 PerId)

  Disalbe Peripheral Clock.
- void RCC_voidSetAHBprescaler (void)

  AHP Prescaler.
- void RCC_voidSetAPB1prescaler (void)

  APB1 Prescaller.
- void RCC_voidSetAPB2prescaler (void)

  APB2 Prescaller.
- void RCC_voidSetADCprescaler (void)

  ADC prescaller.
- void RCC_ADC_SETprescaler (u8 prescaler)

  ADC Prescaller.

4.6.1 Detailed Description

RCC APIs and Global data.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.6.2 Macro Definition Documentation

4.6.2.1 RCC_AHB

```
#define RCC_AHB 0
```

4.6.2.2 RCC_APB1

```
#define RCC_APB1 1
```

4.6.2.3 RCC_APB2

```
#define RCC_APB2 2
```

4.6.3 Enumeration Type Documentation

4.6.3.1 anonymous enum

```
anonymous enum
```

Enumerator

RCC_AHB_DMA1	
RCC_AHB_DMA2	
RCC_AHB_SRAM	
RCC_AHB_FLITF	
RCC_AHB_CRC	
RCC_AHB_FSMC	
RCC_AHB_SDIO	

4.6.3.2 anonymous enum

```
anonymous enum
```

Enumerator

RCC_APB1_TIM2	
RCC_APB1_TIM3	
RCC_APB1_TIM4	
RCC_APB1_TIM5	
RCC_APB1_TIM6	
RCC_APB1_TIM7	
RCC_APB1_TIM12	
RCC_APB1_TIM13	
RCC_APB1_TIM14	
RCC_APB1_WWWG	
RCC_APB1_SPI2	
RCC_APB1_SPI3	
RCC_APB1_USART2	
RCC_APB1_USART3	
RCC_APB1_USART4	
RCC_APB1_USART5	
RCC_APB1_I2C1	
RCC_APB1_I2C2	
RCC_APB1_USB	
RCC_APB1_CAN	
RCC_APB1_BKP	
RCC_APB1_PWR	
RCC_APB1_DAC	

4.6.3.3 anonymous enum

`anonymous enum`

Enumerator

RCC_APB2_AFIO	
RCC_APB2_DIOA	
RCC_APB2_DIOB	
RCC_APB2_DIOC	
RCC_APB2_DIOD	
RCC_APB2_DIOE	
RCC_APB2_DIOF	
RCC_APB2_DIOG	
RCC_APB2_ADC1	
RCC_APB2_ADC2	
RCC_APB2_TIM1	
RCC_APB2_SPI1	
RCC_APB2_TIM8	
RCC_APB2_USART1	
RCC_APB2_ADC3	
RCC_APB2_TIM9	
RCC_APB2_TIM10	
RCC_APB2_TIM11	

4.6.3.4 prescalers

enum prescalers

Enumerator

divided2	
divided4	
divided6	
divided8	

4.6.3.5 RCC_cfgr

enum RCC_cfgr

Enumerator

SW0	
SW1	
SWS0	
SWS1	
HPRE0	
HPRE1	
HPRE2	
HPRE3	
PPRE10	
PPRE11	
PPRE12	
PPRE20	
PPRE21	
PPRE22	
ADCPRE0	
ADCPRE1	
PLLSRC	
PLLXTPRE	
PLLMUL0	
PLLMUL1	
PLLMUL2	
PLLMUL3	
USBPRE	
MCO0	
MCO1	
MCO2	

4.6.3.6 RCC_cr

```
enum RCC_cr
```

Enumerator

HSION	
HSIRDY	
HSITRIM0	
HSITRIM1	
HSITRIM2	
HSITRIM3	
HSITRIM4	
HSICAL0	
HSICAL1	
HSICAL2	
HSICAL3	
HSICAL4	
HSICAL5	
HSICAL6	
HSICAL7	
HSEON	
HSERDY	
HSEBYP	
CSSON	
PLLON	
PLLRDY	

4.6.4 Function Documentation

4.6.4.1 RCC_ADC_SETprescaler()

```
void RCC_ADC_SETprescaler (
            u8 prescaler )
```

ADC Prescaller.

Parameters

prescaler	

4.6.4.2 RCC_intalizeHSE()

```
void RCC_intalizeHSE (
            void  )
```

Initilaize External Crystal.

4.6.4.3 RCC_IntalizeHSI()

```
void RCC_IntalizeHSI (
            void  )
```

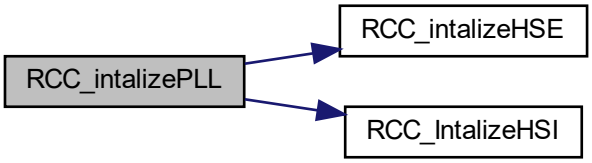
Initialize Internal crystal.

4.6.4.4 RCC_intalizePLL()

```
void RCC_intalizePLL (
            void  )
```

Initialize PLL.

Here is the call graph for this function:



4.6.4.5 RCC_voidDisablePerClk()

```
void RCC_voidDisablePerClk (
            u8 BusId,
            u8 PerId )
```

Disalbe Peripheral Clock.

Parameters

BusId	
PerId	

4.6.4.6 RCC_voidEnablePerClk()

```
void RCC_voidEnablePerClk (
            u8 BusId,
            u8 PerId )
```

Enable Peripheral Clock.

Parameters

BusId	
PerId	

4.6.4.7 RCC_voidSetADCprescaler()

```
void RCC_voidSetADCprescaler (
            void  )
```

ADC prescaller.

4.6.4.8 RCC_voidSetAHBprescaler()

```
void RCC_voidSetAHBprescaler (
            void  )
```

AHP Prescaler.

4.6.4.9 RCC_voidSetAPB1prescaler()

```
void RCC_voidSetAPB1prescaler (
            void  )
```

APB1 Prescaller.

4.6.4.10 RCC_voidSetAPB2prescaler()

```
void RCC_voidSetAPB2prescaler (
            void  )
```

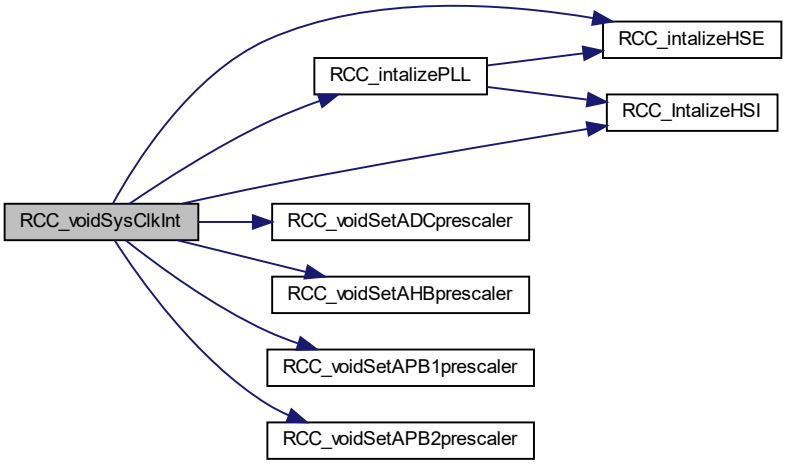
APB2 Prescaller.

4.6.4.11 RCC_voidSysClkInt()

```
void RCC_voidSysClkInt (
            void  )
```

Initialize System Clock.

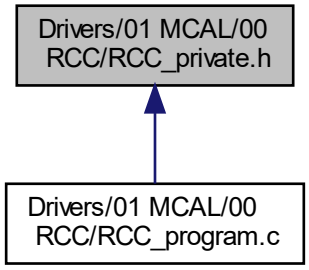
Here is the call graph for this function:



4.7 Drivers/01 MCAL/00 RCC/RCC_private.h File Reference

RCC Private Data.

This graph shows which files directly or indirectly include this file:



Macros

- #define RCC_BASE_ADD (0x40021000)
- #define RCC_CR *((u32 *)(RCC_BASE_ADD+0x00))
- #define RCC_CFGR *((u32 *)(RCC_BASE_ADD+0x04))
- #define RCC_CIR *((u32 *)(RCC_BASE_ADD+0x08))
- #define RCC_APB2RSTR *((u32 *)(RCC_BASE_ADD+0x0C))
- #define RCC_APB1RSTR *((u32 *)(RCC_BASE_ADD+0x10))
- #define RCC_AHBENR *((u32 *)(RCC_BASE_ADD+0x14))
- #define RCC_APB2ENR *((u32 *)(RCC_BASE_ADD+0x18))
- #define RCC_APB1ENR *((u32 *)(RCC_BASE_ADD+0x1C))
- #define RCC_BDCR *((u32 *)(RCC_BASE_ADD+0x20))
- #define RCC_CSR *((u32 *)(RCC_BASE_ADD+0x24))

4.7.1 Detailed Description

RCC Private Data.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.7.2 Macro Definition Documentation

4.7.2.1 RCC_AHBENR

```
#define RCC_AHBENR *((u32 *)(RCC_BASE_ADD+0x14))
```

4.7.2.2 RCC_APB1ENR

```
#define RCC_APB1ENR *((u32 *)(RCC_BASE_ADD+0x1C))
```

4.7.2.3 RCC_APB1RSTR

```
#define RCC_APB1RSTR *((u32 *)(RCC_BASE_ADD+0x10))
```

4.7.2.4 RCC_APB2ENR

```
#define RCC_APB2ENR *((u32 *)(RCC_BASE_ADD+0x18))
```

4.7.2.5 RCC_APB2RSTR

```
#define RCC_APB2RSTR *((u32 *)(RCC_BASE_ADD+0x0C))
```

4.7.2.6 RCC_BASE_ADD

```
#define RCC_BASE_ADD (0x40021000)
```

4.7.2.7 RCC_BDCR

```
#define RCC_BDCR *((u32 *)(RCC_BASE_ADD+0x20))
```

4.7.2.8 RCC_CFGR

```
#define RCC_CFGR *((u32 *)(RCC_BASE_ADD+0x04))
```

4.7.2.9 RCC_CIR

```
#define RCC_CIR *((u32 *)(RCC_BASE_ADD+0x08))
```

4.7.2.10 RCC_CR

```
#define RCC_CR *((u32 *)(RCC_BASE_ADD+0x00))
```

4.7.2.11 RCC_CSR

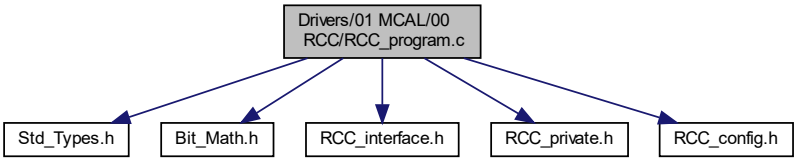
```
#define RCC_CSR *((u32 *)(RCC_BASE_ADD+0x24))
```

4.8 Drivers/01 MCAL/00 RCC/RCC_program.c File Reference

RCC Driver.

```
#include "Std_Types.h"
#include "Bit_Math.h"
#include "RCC_interface.h"
#include "RCC_private.h"
#include "RCC_config.h"
```

Include dependency graph for RCC_program.c:



Functions

- void RCC_IntalizeHSI (void)
  Initialize Internal crystal.
- void RCC_intalizeHSE (void)
  Initilaize External Crystal.
- void RCC_intalizePLL (void)
  Initialize PLL.
- void RCC_voidSysClkInt (void)
  Initialize System Clock.
- void RCC_voidEnablePerClk (u8 BusId, u8 PerId)
  Enable Peripheral Clock.
- void RCC_voidDisablePerClk (u8 BusId, u8 PerId)
  Disalbe Peripheral Clock.
- void RCC_ADC_SETprescaler (u8 prescaler)
  ADC Prescaller.
- void RCC_voidSetAHBprescaler (void)
  AHP Prescaler.
- void RCC_voidSetAPB1prescaler (void)
  APB1 Prescaller.
- void RCC_voidSetAPB2prescaler (void)
  APB2 Prescaller.
- void RCC_voidSetADCprescaler (void)
  ADC prescaller.

4.8.1 Detailed Description

RCC Driver.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.8.2 Function Documentation

4.8.2.1 RCC_ADC_SETprescaler()

```
void RCC_ADC_SETprescaler (
            u8 prescaler )
```

ADC Prescaller.

Parameters

prescaler	

4.8.2.2 RCC_intalizeHSE()

```
void RCC_intalizeHSE (
            void  )
```

Initilaize External Crystal.

4.8.2.3 RCC_IntalizeHSI()

```
void RCC_IntalizeHSI (
            void  )
```

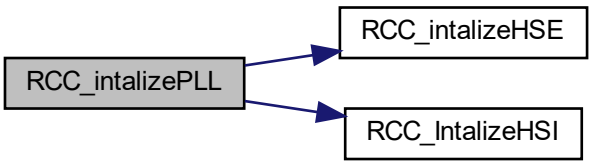
Initialize Internal crystal.

4.8.2.4 RCC_intalizePLL()

```
void RCC_intalizePLL (
            void  )
```

Initialize PLL.

Here is the call graph for this function:



4.8.2.5 RCC_voidDisablePerClk()

```
void RCC_voidDisablePerClk (
            u8 BusId,
            u8 PerId )
```

Disalbe Peripheral Clock.

Parameters

BusId	
PerId	

4.8.2.6 RCC_voidEnablePerClk()

```
void RCC_voidEnablePerClk (
            u8 BusId,
            u8 PerId )
```

Enable Peripheral Clock.

Parameters

BusId	
PerId	

4.8.2.7 RCC_voidSetADCprescaler()

```
void RCC_voidSetADCprescaler (
            void  )
```

ADC prescaller.

4.8.2.8 RCC_voidSetAHBprescaler()

```
void RCC_voidSetAHBprescaler (
            void  )
```

AHP Prescaler.

4.8.2.9 RCC_voidSetAPB1prescaler()

```
void RCC_voidSetAPB1prescaler (
            void  )
```

APB1 Prescaller.

4.8.2.10 RCC_voidSetAPB2prescaler()

```
void RCC_voidSetAPB2prescaler (
            void  )
```

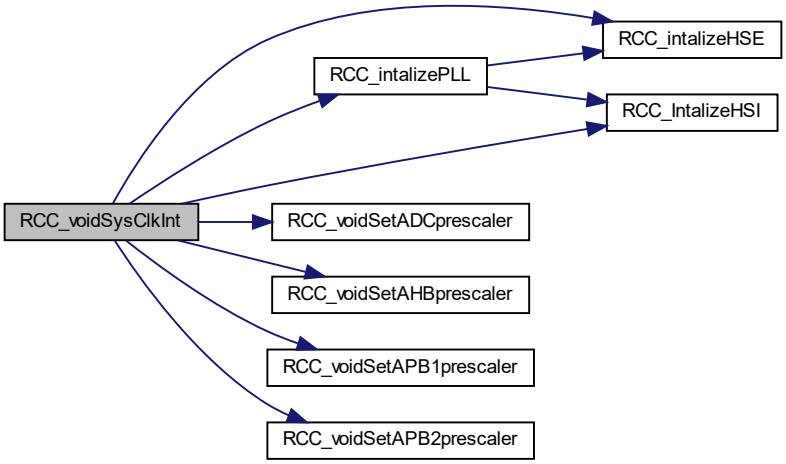
APB2 Prescaller.

4.8.2.11 RCC_voidSysClkInt()

```
void RCC_voidSysClkInt (
            void  )
```

Initialize System Clock.

Here is the call graph for this function:

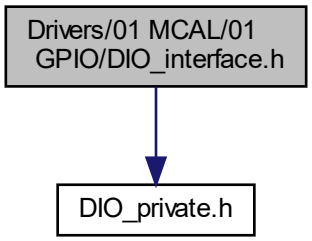


4.9 Drivers/01 MCAL/01 GPIO/DIO_interface.h File Reference

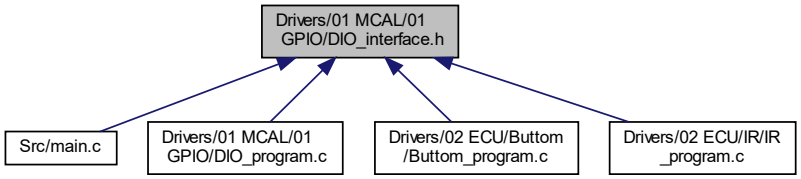
APIs of DIO.

```
#include "DIO_private.h"
```

Include dependency graph for DIO_interface.h:



This graph shows which files directly or indirectly include this file:



Macros

- #define PORTA 0
- #define PORTB 1
- #define PORTC 2
- #define PIN0 0
- #define PIN1 1
- #define PIN2 2
- #define PIN3 3
- #define PIN4 4
- #define PIN5 5
- #define PIN6 6
- #define PIN7 7
- #define PIN8 8
- #define PIN9 9
- #define PIN10 10
- #define PIN11 11
- #define PIN12 12
- #define PIN13 13
- #define PIN14 14
- #define PIN15 15
- #define GPIO_HIGH 1

- #define GPIO_LOW 0
- #define GPIO_INPUT_ANALOG 0b0000
- #define GPIO_INPUT_FLOAT 0b0100
- #define GPIO_INPUT_PULL_UP_DOWN 0b1000
- #define GPIO_OUTPUT_10MHZ_PP 0b0001
- #define GPIO_OUTPUT_10MHZ_OD 0b0101
- #define GPIO_OUTPUT_10MHZ_AFPP 0b1001
- #define GPIO_OUTPUT_10MHZ_AFOD 0b1101
- #define GPIO_OUTPUT_2MHZ_PP 0b0010
- #define GPIO_OUTPUT_2MHZ_OD 0b0110
- #define GPIO_OUTPUT_2MHZ_AFPP 0b1010
- #define GPIO_OUTPUT_2MHZ_AFOD 0b1110
- #define GPIO_OUTPUT_50MHZ_PP 0b0011
- #define GPIO_OUTPUT_50MHZ_OD 0b0111
- #define GPIO_OUTPUT_50MHZ_AFPP 0b1011
- #define GPIO_OUTPUT_50MHZ_AFOD 0b1111

Functions

- void DIO_voidSetPinDirection (u8 port, u8 pin, u8 Direction)

  Set Direction of DIO.
- void DIO_voidSetPinValue (u8 port, u8 pin, u8 Value)

  Set PIN Value.
- void DIO_voidtogglepin (u8 port, u8 pin)

  Toggle PIN State.
- u8 DIO_u8GetPinValue (u8 port, u8 pin)

  return PIN State
- void DIO_voidWriteValue (u8 port, u8 startpin, u8 data)

  write value on PORT

4.9.1 Detailed Description

APIs of DIO.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.9.2 Macro Definition Documentation

4.9.2.1 GPIO_HIGH

```
#define GPIO_HIGH 1
```

4.9.2.2 GPIO_INPUT_ANALOG

```
#define GPIO_INPUT_ANALOG 0b0000
```

4.9.2.3 GPIO_INPUT_FLOAT

```
#define GPIO_INPUT_FLOAT 0b0100
```

4.9.2.4 GPIO_INPUT_PULL_UP_DOWN

```
#define GPIO_INPUT_PULL_UP_DOWN 0b1000
```

4.9.2.5 GPIO_LOW

```
#define GPIO_LOW 0
```

4.9.2.6 GPIO_OUTPUT_10MHZ_AFOD

```
#define GPIO_OUTPUT_10MHZ_AFOD 0b1101
```

4.9.2.7 GPIO_OUTPUT_10MHZ_AFPP

```
#define GPIO_OUTPUT_10MHZ_AFPP 0b1001
```

4.9.2.8 GPIO_OUTPUT_10MHZ_OD

```
#define GPIO_OUTPUT_10MHZ_OD 0b0101
```

4.9.2.9 GPIO_OUTPUT_10MHZ_PP

```
#define GPIO_OUTPUT_10MHZ_PP 0b0001
```

4.9.2.10 GPIO_OUTPUT_2MHZ_AFOD

```
#define GPIO_OUTPUT_2MHZ_AFOD 0b1110
```

4.9.2.11 GPIO_OUTPUT_2MHZ_AFPP

```
#define GPIO_OUTPUT_2MHZ_AFPP 0b1010
```

4.9.2.12 GPIO_OUTPUT_2MHZ_OD

```
#define GPIO_OUTPUT_2MHZ_OD 0b0110
```

4.9.2.13 GPIO_OUTPUT_2MHZ_PP

```
#define GPIO_OUTPUT_2MHZ_PP 0b0010
```

4.9.2.14 GPIO_OUTPUT_50MHZ_AFOD

```
#define GPIO_OUTPUT_50MHZ_AFOD 0b1111
```

4.9.2.15 GPIO_OUTPUT_50MHZ_AFPP

```
#define GPIO_OUTPUT_50MHZ_AFPP 0b1011
```

4.9.2.16 GPIO_OUTPUT_50MHZ_OD

```
#define GPIO_OUTPUT_50MHZ_OD 0b0111
```

4.9.2.17 GPIO_OUTPUT_50MHZ_PP

```
#define GPIO_OUTPUT_50MHZ_PP 0b0011
```

4.9.2.18 PIN0

```
#define PIN0 0
```

4.9.2.19 PIN1

```
#define PIN1 1
```

4.9.2.20 PIN10

```
#define PIN10 10
```

4.9.2.21 PIN11

```
#define PIN11 11
```

4.9.2.22 PIN12

```
#define PIN12 12
```

4.9.2.23 PIN13

```
#define PIN13 13
```

4.9.2.24 PIN14

```
#define PIN14 14
```

4.9.2.25 PIN15

```
#define PIN15 15
```

4.9.2.26 PIN2

```
#define PIN2 2
```

4.9.2.27 PIN3

```
#define PIN3 3
```

4.9.2.28 PIN4

```
#define PIN4 4
```

4.9.2.29 PIN5

```
#define PIN5 5
```

4.9.2.30 PIN6

```
#define PIN6 6
```

4.9.2.31 PIN7

```
#define PIN7 7
```

4.9.2.32 PIN8

```
#define PIN8 8
```

4.9.2.33 PIN9

```
#define PIN9 9
```

4.9.2.34 PORTA

```
#define PORTA 0
```

4.9.2.35 PORTB

```
#define PORTB 1
```

4.9.2.36 PORTC

```
#define PORTC 2
```

4.9.3 Function Documentation

4.9.3.1 DIO_u8GetPinValue()

```
u8 DIO_u8GetPinValue (
            u8 port,
            u8 pin )
```

return PIN State

Parameters

port	
pin	

Returns

u8

4.9.3.2 DIO_voidSetPinDirection()

```
void DIO_voidSetPinDirection (
            u8 port,
            u8 pin,
            u8 Direction )
```

Set Direction of DIO.

Parameters

port	
pin	
Direction	

4.9.3.3 DIO_voidSetPinValue()

```
void DIO_voidSetPinValue (
            u8 port,
            u8 pin,
            u8 Value )
```

Set PIN Value.

Parameters

port	
pin	
Value	

4.9.3.4 DIO_voidtogglepin()

```
void DIO_voidtogglepin (
            u8 port,
            u8 pin )
```

Toggle PIN State.

Parameters

port	
pin	

4.9.3.5 DIO_voidWriteValue()

```
void DIO_voidWriteValue (
            u8 port,
            u8 startpin,
            u8 data )
```

write value on PORT

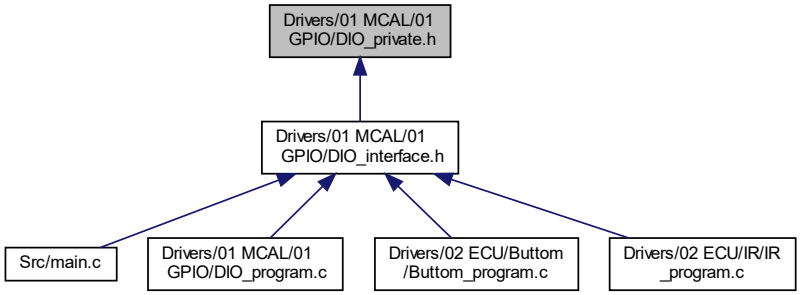
Parameters

port	
startpin	
data	

4.10 Drivers/01 MCAL/01 GPIO/DIO_private.h File Reference

Private Data.

This graph shows which files directly or indirectly include this file:



Macros

- #define GPIOA_BASE_ADD (0x40010800)
- #define GPIOA_CRL ∗((u32 ∗)(GPIOA_BASE_ADD+0x00))
- #define GPIOA_CRH ∗((u32 ∗)(GPIOA_BASE_ADD+0x04))
- #define GPIOA_IDR ∗((u32 ∗)(GPIOA_BASE_ADD+0x08))

- #define GPIOA_ODR *((u32 *)(GPIOA_BASE_ADD+0x0C))
- #define GPIOA_BSRR *((u32 *)(GPIOA_BASE_ADD+0x10))
- #define GPIOA_BRR *((u32 *)(GPIOA_BASE_ADD+0x14))
- #define GPIOA_LCKR *((u32 *)(GPIOA_BASE_ADD+0x18))
- #define GPIOB_BASE_ADD (0x40010C00)
- #define GPIOB_CRL *((u32 *)(GPIOB_BASE_ADD+0x00))
- #define GPIOB_CRH *((u32 *)(GPIOB_BASE_ADD+0x04))
- #define GPIOB_IDR *((u32 *)(GPIOB_BASE_ADD+0x08))
- #define GPIOB_ODR *((u32 *)(GPIOB_BASE_ADD+0x0C))
- #define GPIOB_BSRR *((u32 *)(GPIOB_BASE_ADD+0x10))
- #define GPIOB_BRR *((u32 *)(GPIOB_BASE_ADD+0x14))
- #define GPIOB_LCKR *((u32 *)(GPIOB_BASE_ADD+0x18))
- #define GPIOC_BASE_ADD (0x40011000)
- #define GPIOC_CRL *((u32 *)(GPIOC_BASE_ADD+0x00))
- #define GPIOC_CRH *((u32 *)(GPIOC_BASE_ADD+0x04))
- #define GPIOC_IDR *((u32 *)(GPIOC_BASE_ADD+0x08))
- #define GPIOC_ODR *((u32 *)(GPIOC_BASE_ADD+0x0C))
- #define GPIOC_BSRR *((u32 *)(GPIOC_BASE_ADD+0x10))
- #define GPIOC_BRR *((u32 *)(GPIOC_BASE_ADD+0x14))
- #define GPIOC_LCKR *((u32 *)(GPIOC_BASE_ADD+0x18))

4.10.1 Detailed Description

Private Data.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.10.2 Macro Definition Documentation

4.10.2.1 GPIOA_BASE_ADD

```
#define GPIOA_BASE_ADD (0x40010800)
```

4.10.2.2 GPIOA_BRR

```
#define GPIOA_BRR *((u32 *)(GPIOA_BASE_ADD+0x14))
```

4.10.2.3 GPIOA_BSRR

```
#define GPIOA_BSRR *((u32 *)(GPIOA_BASE_ADD+0x10))
```

4.10.2.4 GPIOA_CRH

```
#define GPIOA_CRH *((u32 *)(GPIOA_BASE_ADD+0x04))
```

4.10.2.5 GPIOA_CRL

```
#define GPIOA_CRL *((u32 *)(GPIOA_BASE_ADD+0x00))
```

4.10.2.6 GPIOA_IDR

```
#define GPIOA_IDR *((u32 *)(GPIOA_BASE_ADD+0x08))
```

4.10.2.7 GPIOA_LCKR

```
#define GPIOA_LCKR *((u32 *)(GPIOA_BASE_ADD+0x18))
```

4.10.2.8 GPIOA_ODR

```
#define GPIOA_ODR *((u32 *)(GPIOA_BASE_ADD+0x0C))
```

4.10.2.9 GPIOB_BASE_ADD

```
#define GPIOB_BASE_ADD (0x40010C00)
```

4.10.2.10 GPIOB_BRR

```
#define GPIOB_BRR *((u32 *)(GPIOB_BASE_ADD+0x14))
```

4.10.2.11 GPIOB_BSRR

```
#define GPIOB_BSRR *((u32 *)(GPIOB_BASE_ADD+0x10))
```

4.10.2.12 GPIOB_CRH

```
#define GPIOB_CRH *((u32 *)(GPIOB_BASE_ADD+0x04))
```

4.10.2.13 GPIOB_CRL

```
#define GPIOB_CRL *((u32 *)(GPIOB_BASE_ADD+0x00))
```

4.10.2.14 GPIOB_IDR

```
#define GPIOB_IDR *((u32 *)(GPIOB_BASE_ADD+0x08))
```

4.10.2.15 GPIOB_LCKR

```
#define GPIOB_LCKR *((u32 *)(GPIOB_BASE_ADD+0x18))
```

4.10.2.16 GPIOB_ODR

```
#define GPIOB_ODR *((u32 *)(GPIOB_BASE_ADD+0x0C))
```

4.10.2.17 GPIOC_BASE_ADD

```
#define GPIOC_BASE_ADD (0x40011000)
```

4.10.2.18 GPIOC_BRR

```
#define GPIOC_BRR *((u32 *)(GPIOC_BASE_ADD+0x14))
```

4.10.2.19 GPIOC_BSRR

```
#define GPIOC_BSRR *((u32 *)(GPIOC_BASE_ADD+0x10))
```

4.10.2.20 GPIOC_CRH

```
#define GPIOC_CRH *((u32 *)(GPIOC_BASE_ADD+0x04))
```

4.10.2.21 GPIOC_CRL

```
#define GPIOC_CRL *((u32 *)(GPIOC_BASE_ADD+0x00))
```

4.10.2.22 GPIOC_IDR

```
#define GPIOC_IDR *((u32 *)(GPIOC_BASE_ADD+0x08))
```

4.10.2.23 GPIOC_LCKR

```
#define GPIOC_LCKR *((u32 *)(GPIOC_BASE_ADD+0x18))
```

4.10.2.24 GPIOC_ODR

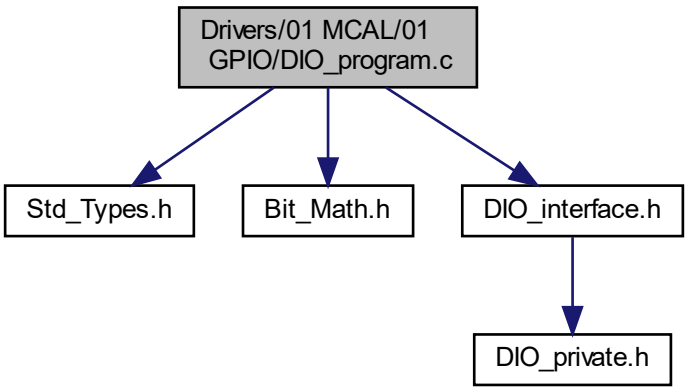
```
#define GPIOC_ODR *((u32 *)(GPIOC_BASE_ADD+0x0C))
```

4.11 Drivers/01 MCAL/01 GPIO/DIO_program.c File Reference

DIO Implementaiton.

```
#include "Std_Types.h"
#include "Bit_Math.h"
#include "DIO_interface.h"
```

Include dependency graph for DIO_program.c:



Functions

- void DIO_voidSetPinDirection (u8 port, u8 pin, u8 Direction)

  Set Direction of DIO.
- void DIO_voidSetPinValue (u8 port, u8 pin, u8 Value)

  Set PIN Value.
- void DIO_voidtogglepin (u8 port, u8 pin)

  Toggle PIN State.
- u8 DIO_u8GetPinValue (u8 port, u8 pin)

  return PIN State
- void DIO_voidWriteValue (u8 port, u8 startpin, u8 data)

  write value on PORT

4.11.1 Detailed Description

DIO Implementaiton.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.11.2 Function Documentation

4.11.2.1 DIO_u8GetPinValue()

```
u8 DIO_u8GetPinValue (
            u8 port,
            u8 pin )
```

return PIN State

Parameters

port	
pin	

Returns

u8

4.11.2.2 DIO_voidSetPinDirection()

```
void DIO_voidSetPinDirection (
            u8 port,
            u8 pin,
            u8 Direction )
```

Set Direction of DIO.

Parameters

port	
pin	
Direction	

4.11.2.3 DIO_voidSetPinValue()

```
void DIO_voidSetPinValue (
            u8 port,
            u8 pin,
            u8 Value )
```

Set PIN Value.

Parameters

port	
pin	
Value	

4.11.2.4 DIO_voidtogglepin()

```
void DIO_voidtogglepin (
            u8 port,
            u8 pin )
```

Toggle PIN State.

Parameters

port	
pin	

4.11.2.5 DIO_voidWriteValue()

```
void DIO_voidWriteValue (
            u8 port,
            u8 startpin,
            u8 data )
```

write value on PORT

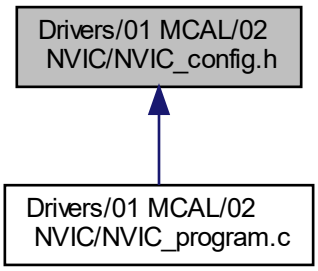
Parameters

port	
startpin	
data	

4.12 Drivers/01 MCAL/02 NVIC/NVIC_config.h File Reference

NVIC For ARM.

This graph shows which files directly or indirectly include this file:



Macros

- #define NVIC_NUM_GRP_SUB GRB_04_SUB_04

4.12.1 Detailed Description

NVIC For ARM.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.12.2 Macro Definition Documentation

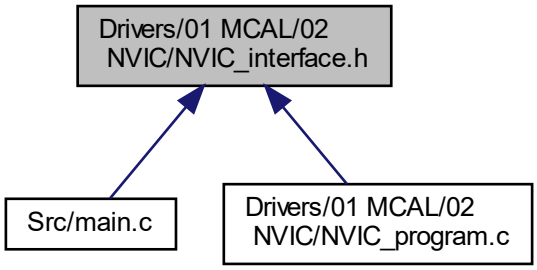
4.12.2.1 NVIC_NUM_GRP_SUB

```
#define NVIC_NUM_GRP_SUB GRB_04_SUB_04
```

4.13 Drivers/01 MCAL/02 NVIC/NVIC_interface.h File Reference

NVIC APIs and Global data.

This graph shows which files directly or indirectly include this file:



Functions

- void NVIC_voidInit ()
- void NVIC_voidEnablePerInt (u8 IntNum)
- void NVIC_voidDisablePerInt (u8 IntNum)
- void NVIC_voidSetPendingFlag (u8 IntNum)
- void NVIC_voidClrPendingFlag (u8 IntNum)
- u8 NVIC_u8ReadActiveFlag (u8 IntNum)
- void NVIC_voidSetSwIntPriority (u8 IntNum, u8 priority)

4.13.1 Detailed Description

NVIC APIs and Global data.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.13.2 Function Documentation

4.13.2.1 NVIC_u8ReadActiveFlag()

```
u8 NVIC_u8ReadActiveFlag (
            u8 IntNum )
```

4.13.2.2 NVIC_voidClrPendingFlag()

```
void NVIC_voidClrPendingFlag (
            u8 IntNum )
```

4.13.2.3 NVIC_voidDisablePerInt()

```
void NVIC_voidDisablePerInt (
            u8 IntNum )
```

4.13.2.4 NVIC_voidEnablePerInt()

```
void NVIC_voidEnablePerInt (
            u8 IntNum )
```

4.13.2.5 NVIC_voidInit()

```
void NVIC_voidInit ( )
```

4.13.2.6 NVIC_voidSetPendingFlag()

```
void NVIC_voidSetPendingFlag (
            u8 IntNum )
```

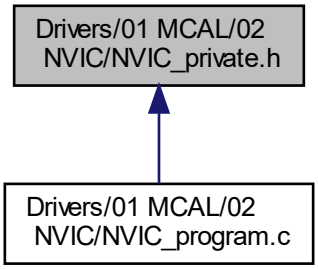
4.13.2.7 NVIC_voidSetSwIntPriority()

```
void NVIC_voidSetSwIntPriority (
            u8 IntNum,
            u8 priority )
```

4.14 Drivers/01 MCAL/02 NVIC/NVIC_private.h File Reference

NVIC Private Data.

This graph shows which files directly or indirectly include this file:



Macros

- #define NVIC_BASE_ADD (0xE000E100)
- #define NVIC_ISER0 *((u32 *)(NVIC_BASE_ADD + 0x000))
- #define NVIC_ISER1 *((u32 *)(NVIC_BASE_ADD + 0x004))
- #define NVIC_ICER0 *((u32 *)(NVIC_BASE_ADD + 0x080))
- #define NVIC_ICER1 *((u32 *)(NVIC_BASE_ADD + 0x084))
- #define NVIC_ISPR0 *((u32 *)(NVIC_BASE_ADD + 0x100))
- #define NVIC_ISPR1 *((u32 *)(NVIC_BASE_ADD + 0x104))
- #define NVIC_ICPR0 *((u32 *)(NVIC_BASE_ADD + 0x180))
- #define NVIC_ICPR1 *((u32 *)(NVIC_BASE_ADD + 0x184))
- #define NVIC_IABR0 *((u32 *)(NVIC_BASE_ADD + 0x200))
- #define NVIC_IABR1 *((u32 *)(NVIC_BASE_ADD + 0x204))
- #define NVIC_IPR ((u8 *)(NVIC_BASE_ADD + 0x300))
- #define GRB_16_SUB_00 (0x05FA0300)
- #define GRB_08_SUB_02 (0x05FA0400)
- #define GRB_04_SUB_04 (0x05FA0500)
- #define GRB_02_SUB_08 (0x05FA0600)
- #define GRB_00_SUB_16 (0x05FA0700)

4.14.1 Detailed Description

NVIC Private Data.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.14.2 Macro Definition Documentation

4.14.2.1 GRB_00_SUB_16

```
#define GRB_00_SUB_16 (0x05FA0700)
```

4.14.2.2 GRB_02_SUB_08

```
#define GRB_02_SUB_08 (0x05FA0600)
```

4.14.2.3 GRB_04_SUB_04

```
#define GRB_04_SUB_04 (0x05FA0500)
```

4.14.2.4 GRB_08_SUB_02

```
#define GRB_08_SUB_02 (0x05FA0400)
```

4.14.2.5 GRB_16_SUB_00

```
#define GRB_16_SUB_00 (0x05FA0300)
```

4.14.2.6 NVIC_BASE_ADD

```
#define NVIC_BASE_ADD (0xE000E100)
```

4.14.2.7 NVIC_IABR0

```
#define NVIC_IABR0 *((u32 *)(NVIC_BASE_ADD + 0x200))
```

4.14.2.8 NVIC_IABR1

```
#define NVIC_IABR1 *((u32 *)(NVIC_BASE_ADD + 0x204))
```

4.14.2.9 NVIC_ICER0

```
#define NVIC_ICER0 *((u32 *)(NVIC_BASE_ADD + 0x080))
```

4.14.2.10 NVIC_ICER1

```
#define NVIC_ICER1 *((u32 *)(NVIC_BASE_ADD + 0x084))
```

4.14.2.11 NVIC_ICPR0

```
#define NVIC_ICPR0 *((u32 *)(NVIC_BASE_ADD + 0x180))
```

4.14.2.12 NVIC_ICPR1

```
#define NVIC_ICPR1 *((u32 *)(NVIC_BASE_ADD + 0x184))
```

4.14.2.13 NVIC_IPR

```
#define NVIC_IPR ((u8 *)(NVIC_BASE_ADD + 0x300))
```

4.14.2.14 NVIC_ISER0

```
#define NVIC_ISER0 *((u32 *)(NVIC_BASE_ADD + 0x000))
```

4.14.2.15 NVIC_ISER1

```
#define NVIC_ISER1 *((u32 *)(NVIC_BASE_ADD + 0x004))
```

4.14.2.16 NVIC_ISPR0

```
#define NVIC_ISPR0 *((u32 *)(NVIC_BASE_ADD + 0x100))
```

4.14.2.17 NVIC_ISPR1

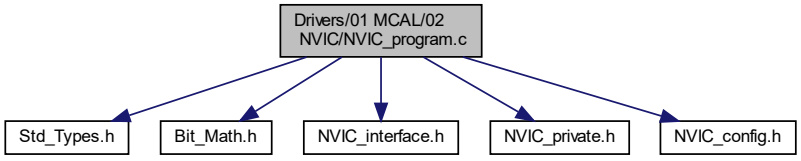
```
#define NVIC_ISPR1 *((u32 *)(NVIC_BASE_ADD + 0x104))
```

4.15 Drivers/01 MCAL/02 NVIC/NVIC_program.c File Reference

NVIC Driver Implementation.

```
#include "Std_Types.h"
#include "Bit_Math.h"
#include "NVIC_interface.h"
#include "NVIC_private.h"
#include "NVIC_config.h"
```

Include dependency graph for NVIC_program.c:



Macros

- #define SCB_BASE_ADD (0xE000ED00)
- #define SCB_AIRCR *((u32 *)(SCB_BASE_ADD+0x0C))

Functions

- void NVIC_voidInit ()
- void NVIC_voidEnablePerInt (u8 IntNum)
- void NVIC_voidDisablePerInt (u8 IntNum)
- void NVIC_voidSetPendingFlag (u8 IntNum)
- void NVIC_voidClrPendingFlag (u8 IntNum)
- u8 NVIC_u8ReadActiveFlag (u8 IntNum)
- void NVIC_voidSetSwIntPriority (u8 IntNum, u8 priority)

4.15.1 Detailed Description

NVIC Driver Implementation.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.15.2 Macro Definition Documentation

4.15.2.1 SCB_AIRCR

```
#define SCB_AIRCR *((u32 *)(SCB_BASE_ADD+0x0C))
```

4.15.2.2 SCB_BASE_ADD

```
#define SCB_BASE_ADD (0xE000ED00)
```

4.15.3 Function Documentation

4.15.3.1 NVIC_u8ReadActiveFlag()

```
u8 NVIC_u8ReadActiveFlag (
            u8 IntNum )
```

4.15.3.2 NVIC_voidClrPendingFlag()

```
void NVIC_voidClrPendingFlag (
            u8 IntNum )
```

4.15.3.3 NVIC_voidDisablePerInt()

```
void NVIC_voidDisablePerInt (
            u8 IntNum )
```

4.15.3.4 NVIC_voidEnablePerInt()

```
void NVIC_voidEnablePerInt (
            u8 IntNum )
```

4.15.3.5 NVIC_voidInit()

```
void NVIC_voidInit ( )
```

4.15.3.6 NVIC_voidSetPendingFlag()

```
void NVIC_voidSetPendingFlag (
            u8 IntNum )
```

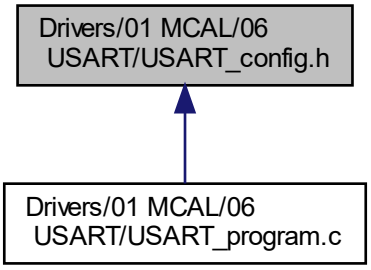
4.15.3.7 NVIC_voidSetSwIntPriority()

```
void NVIC_voidSetSwIntPriority (
             u8 IntNum,
             u8 priority )
```

4.16 Drivers/01 MCAL/06 USART/USART_config.h File Reference

configuration of USART

This graph shows which files directly or indirectly include this file:



Macros

- #define USART1_BAUDRATE BAUD_RATE_9600
- #define USART1_PARITY PARITY_NO
- #define USART1_PARITY_TYPE PARITY_EVEN
- #define USART1_DATA_LENGTH DATA_LEN_8
- #define USART1_STOP_LENGTH STOP_BIT_1
- #define USART2_BAUDRATE BAUD_RATE_9600
- #define USART2_PARITY PARITY_NO
- #define USART2_PARITY_TYPE PARITY_EVEN
- #define USART2_DATA_LENGTH DATA_LEN_8
- #define USART2_STOP_LENGTH STOP_BIT_1

4.16.1 Detailed Description

configuration of USART

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.16.2 Macro Definition Documentation

4.16.2.1 USART1_BAUDRATE

```
#define USART1_BAUDRATE BAUD_RATE_9600
```

4.16.2.2 USART1_DATA_LENGTH

```
#define USART1_DATA_LENGTH DATA_LEN_8
```

4.16.2.3 USART1_PARITY

```
#define USART1_PARITY PARITY_NO
```

4.16.2.4 USART1_PARITY_TYPE

```
#define USART1_PARITY_TYPE PARITY_EVEN
```

4.16.2.5 USART1_STOP_LENGTH

```
#define USART1_STOP_LENGTH STOP_BIT_1
```

4.16.2.6 USART2_BAUDRATE

```
#define USART2_BAUDRATE BAUD_RATE_9600
```

4.16.2.7 USART2_DATA_LENGTH

```
#define USART2_DATA_LENGTH DATA_LEN_8
```

4.16.2.8 USART2_PARITY

```
#define USART2_PARITY PARITY_NO
```

4.16.2.9 USART2_PARITY_TYPE

```
#define USART2_PARITY_TYPE PARITY_EVEN
```

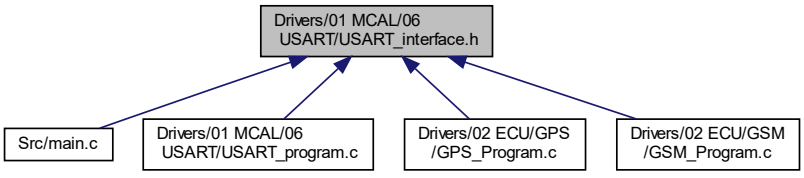
4.16.2.10 USART2_STOP_LENGTH

```
#define USART2_STOP_LENGTH STOP_BIT_1
```

4.17 Drivers/01 MCAL/06 USART/USART_interface.h File Reference

APIs and Global data.

This graph shows which files directly or indirectly include this file:



Functions

- void UART1_init (void)
  Initialized UART.
- void UART1_enableTX (void)
  Enable Data Transmition.
- void UART1_enableRX (void)
  Enable data Recieve.
- void UART1_TXdata (u32 data)
  Transmint Data.
- u32 UART1_RXdata (void)
  Recieve Data.
- void UART1_voidSendString (char const *str)
  Send String.
- u8 *const UART1_StrReceiveString (void)
  recieve string
- void UART2_init (void)
  Uart2 Inialization.
- void UART2_enableTX (void)
  Enable TX of UART2.
- void UART2_enableRX (void)
  Enable Recieve data.
- void UART2_TXdata (u32 data)
  Transmit data.
- u32 UART2_RXdata (void)
  Recieve data.
- void UART2_voidSendString (char const *str)
  Send string.
- u8 *const UART2_StrReceiveString (void)
  Recieve string.
- void UART2_RX_CALLBACK (void(*func)(void))
  Call Back Function To Handle ISR.
- void USART2_IRQHandler (void)
  ISR of UART2.

4.17.1 Detailed Description

APIs and Global data.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.17.2 Function Documentation

4.17.2.1 UART1_enableRX()

```
void UART1_enableRX (
             void  )
```

Enable data Recieve.

4.17.2.2 UART1_enableTX()

```
void UART1_enableTX (
             void  )
```

Enable Data Transmition.

4.17.2.3 UART1_init()

```
void UART1_init (
             void  )
```

Initialized UART.

4.17.2.4 UART1_RXdata()

```
u32 UART1_RXdata (
             void  )
```

Recieve Data.

Returns

u32 Data to be recieverd

4.17.2.5 UART1_StrReceiveString()

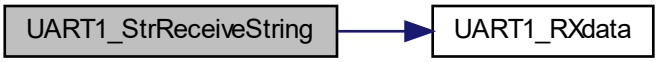
```
u8* const UART1_StrReceiveString (
             void  )
```

recieve string

Returns

u8* const

Here is the call graph for this function:



4.17.2.6 UART1_TXdata()

```
void UART1_TXdata (
             u32 data )
```

Transmint Data.

Parameters

data	data to be transmitter

4.17.2.7 UART1_voidSendString()

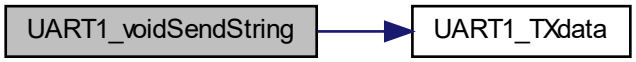
```
void UART1_voidSendString (
             char const * str )
```

Send String.

Parameters

str	

Here is the call graph for this function:



4.17.2.8 UART2_enableRX()

```
void UART2_enableRX (
             void  )
```

Enable Recieve data.

4.17.2.9 UART2_enableTX()

```
void UART2_enableTX (
             void  )
```

Enable TX of UART2.

4.17.2.10 UART2_init()

```
void UART2_init (
             void  )
```

Uart2 Inialization.

4.17.2.11 UART2_RX_CALLBACK()

```
void UART2_RX_CALLBACK (
             void(*)(void) func )
```

Call Back Function To Handle ISR.

Parameters

func	

4.17.2.12 UART2_RXdata()

```
u32 UART2_RXdata (
            void  )
```

Recieve data.

Returns

u32

4.17.2.13 UART2_StrReceiveString()

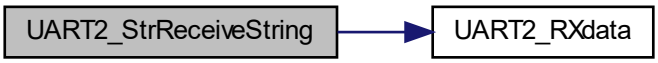
```
u8* const UART2_StrReceiveString (
            void  )
```

Recieve string.

Returns

u8∗ const

Here is the call graph for this function:



4.17.2.14 UART2_TXdata()

```
void UART2_TXdata (
            u32 data )
```

Transmit data.

Parameters

data	

4.17.2.15 UART2_voidSendString()

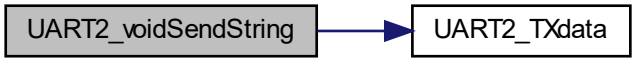
```
void UART2_voidSendString (
            char const * str )
```

Send string.

Parameters

str	

Here is the call graph for this function:



4.17.2.16 USART2_IRQHandler()

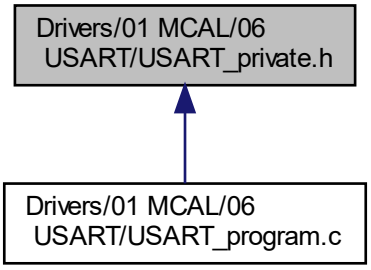
```
void USART2_IRQHandler (
            void  )
```

ISR of UART2.

4.18 Drivers/01 MCAL/06 USART/USART_private.h File Reference

Private APIs and Data.

This graph shows which files directly or indirectly include this file:



Data Structures

- struct USART_type

Macros

- #define USART1 ((volatile USART_type *)(0x40013800))
- #define USART2 ((volatile USART_type *)(0x40004400))
- #define BAUD_RATE_9600 0x341
- #define BAUD_RATE_115200 0x45
- #define PARITY_NO 0
- #define PARITY_YES 1
- #define PARITY_EVEN 0
- #define PARITY_ODD 1
- #define DATA_LEN_8 1
- #define DATA_LEN_9 0
- #define STOP_BIT_1 0b00
- #define STOP_BIT_2 0b10
- #define PE 0
- #define FE 1
- #define NE 2
- #define ORE 3
- #define IDLE 4
- #define RXNE 5
- #define TC 6
- #define TXE 7
- #define LBD 8
- #define CTS 9
- #define SBK 0
- #define RWU 1
- #define RE 2
- #define TE 3
- #define IDLEIE 4
- #define RXNEIE 5
- #define TCIE 6

- #define TCEIE 7
- #define PEIE 8
- #define PS 9
- #define PCE 10
- #define WAKe 11
- #define M 12
- #define UE 13
- #define ADD 0
- #define LBDL 5
- #define LBDIE 6
- #define LBCL 8
- #define CPHA 9
- #define CPOL 10
- #define CLKEN 11
- #define STOP 12
- #define LINEN 14
- #define EIE 0
- #define IREN 1
- #define IRLP 2
- #define HDSEL 3
- #define NACK 4
- #define SCEN 5
- #define DMAR 6
- #define DMAT 7
- #define RTSE 8
- #define CTSE 9
- #define CTSIE 10
- #define PSC 0
- #define GT 8

4.18.1 Detailed Description

Private APIs and Data.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.18.2 Macro Definition Documentation

4.18.2.1 ADD

```
#define ADD 0
```

4.18.2.2 BAUD_RATE_115200

```
#define BAUD_RATE_115200 0x45
```

4.18.2.3 BAUD_RATE_9600

```
#define BAUD_RATE_9600 0x341
```

4.18.2.4 CLKEN

```
#define CLKEN 11
```

4.18.2.5 CPHA

```
#define CPHA 9
```

4.18.2.6 CPOL

```
#define CPOL 10
```

4.18.2.7 CTS

```
#define CTS 9
```

4.18.2.8 CTSE

```
#define CTSE 9
```

4.18.2.9 CTSIE

```
#define CTSIE 10
```

4.18.2.10 DATA_LEN_8

```
#define DATA_LEN_8 1
```

4.18.2.11 DATA_LEN_9

```
#define DATA_LEN_9 0
```

4.18.2.12 DMAR

```
#define DMAR 6
```

4.18.2.13 DMAT

```
#define DMAT 7
```

4.18.2.14 EIE

```
#define EIE 0
```

4.18.2.15 FE

```
#define FE 1
```

4.18.2.16 GT

```
#define GT 8
```

4.18.2.17 HDSEL

```
#define HDSEL 3
```

4.18.2.18 IDLE

```
#define IDLE 4
```

4.18.2.19 IDLEIE

```
#define IDLEIE 4
```

4.18.2.20 IREN

```
#define IREN 1
```

4.18.2.21 IRLP

```
#define IRLP 2
```

4.18.2.22 LBCL

```
#define LBCL 8
```

4.18.2.23 LBD

```
#define LBD 8
```

4.18.2.24 LBDIE

```
#define LBDIE 6
```

4.18.2.25 LBDL

```
#define LBDL 5
```

4.18.2.26 LINEN

```
#define LINEN 14
```

4.18.2.27 M

```
#define M 12
```

4.18.2.28 NACK

```
#define NACK 4
```

4.18.2.29 NE

```
#define NE 2
```

4.18.2.30 ORE

```
#define ORE 3
```

4.18.2.31 PARITY_EVEN

```
#define PARITY_EVEN 0
```

4.18.2.32 PARITY_NO

```
#define PARITY_NO 0
```

4.18.2.33 PARITY_ODD

```
#define PARITY_ODD 1
```

4.18.2.34 PARITY_YES

```
#define PARITY_YES 1
```

4.18.2.35 PCE

```
#define PCE 10
```

4.18.2.36 PE

```
#define PE 0
```

4.18.2.37 PEIE

```
#define PEIE 8
```

4.18.2.38 PS

```
#define PS 9
```

4.18.2.39 PSC

```
#define PSC 0
```

4.18.2.40 RE

```
#define RE 2
```

4.18.2.41 RTSE

```
#define RTSE 8
```

4.18.2.42 RWU

```
#define RWU 1
```

4.18.2.43 RXNE

```
#define RXNE 5
```

4.18.2.44 RXNEIE

```
#define RXNEIE 5
```

4.18.2.45 SBK

```
#define SBK 0
```

4.18.2.46 SCEN

```
#define SCEN 5
```

4.18.2.47 STOP

```
#define STOP 12
```

4.18.2.48 STOP_BIT_1

```
#define STOP_BIT_1 0b00
```

4.18.2.49 STOP_BIT_2

```
#define STOP_BIT_2 0b10
```

4.18.2.50 TC

```
#define TC 6
```

4.18.2.51 TCEIE

```
#define TCEIE 7
```

4.18.2.52 TCIE

```
#define TCIE 6
```

4.18.2.53 TE

```
#define TE 3
```

4.18.2.54 TXE

```
#define TXE 7
```

4.18.2.55 UE

```
#define UE 13
```

4.18.2.56 USART1

```
#define USART1 ((volatile USART_type *)(0x40013800))
```

4.18.2.57 USART2

```
#define USART2 ((volatile USART_type *)(0x40004400))
```

4.18.2.58 WAKe

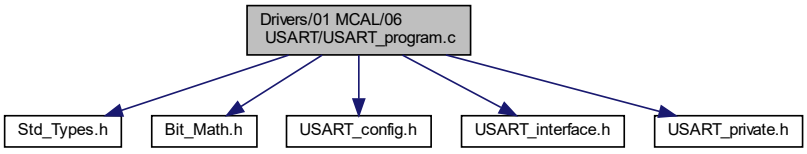
```
#define WAKe 11
```

4.19 Drivers/01 MCAL/06 USART/USART_program.c File Reference

USART Driver Implementation.

```
#include "Std_Types.h"
#include "Bit_Math.h"
#include "USART_config.h"
#include "USART_interface.h"
#include "USART_private.h"
```

Include dependency graph for USART_program.c:



Functions

- void UART1_init (void)

  Initialized UART.
- void UART1_enableTX (void)

  Enable Data Transmition.
- void UART1_enableRX (void)

  Enable data Recieve.
- void UART1_TXdata (u32 data)

  Transmint Data.
- u32 UART1_RXdata (void)

Recieve Data.

- void UART1_voidSendString (char const *str)

  Send String.

- u8 *const UART1_StrReceiveString (void)

  recieve string

- void UART2_init (void)

  Uart2 Inialization.

- void UART2_enableTX (void)

  Enable TX of UART2.

- void UART2_enableRX (void)

  Enable Recieve data.

- void UART2_TXdata (u32 data)

  Transmit data.

- u32 UART2_RXdata (void)

  Recieve data.

- void UART2_voidSendString (char const *str)

  Send string.

- u8 *const UART2_StrReceiveString (void)

  Recieve string.

- void UART2_RX_CALLBACK (void(*func)(void))

  Call Back Function To Handle ISR.

- void USART2_IRQHandler (void)

  ISR of UART2.

Variables

- void(* UART2_PTR)(void)=0

4.19.1 Detailed Description

USART Driver Implementation.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.19.2 Function Documentation

4.19.2.1 UART1_enableRX()

```
void UART1_enableRX (
             void  )
```

Enable data Recieve.

4.19.2.2 UART1_enableTX()

```
void UART1_enableTX (
             void  )
```

Enable Data Transmition.

4.19.2.3 UART1_init()

```
void UART1_init (
             void  )
```

Initialized UART.

4.19.2.4 UART1_RXdata()

```
u32 UART1_RXdata (
             void  )
```

Recieve Data.

Returns

u32 Data to be recieverd

4.19.2.5 UART1_StrReceiveString()

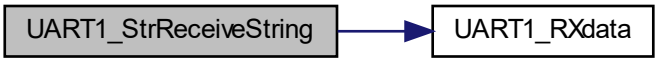
```
u8* const UART1_StrReceiveString (
             void  )
```

recieve string

Returns

u8* const

Here is the call graph for this function:



4.19.2.6 UART1_TXdata()

```
void UART1_TXdata (
             u32 data )
```

Transmint Data.

Parameters

data	data to be transmitter

4.19.2.7 UART1_voidSendString()

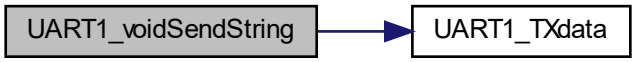
```
void UART1_voidSendString (
             char const * str )
```

Send String.

Parameters

str	

Here is the call graph for this function:



4.19.2.8 UART2_enableRX()

```
void UART2_enableRX (
             void  )
```

Enable Recieve data.

4.19.2.9 UART2_enableTX()

```
void UART2_enableTX (
             void  )
```

Enable TX of UART2.

4.19.2.10 UART2_init()

```
void UART2_init (
             void  )
```

Uart2 Inialization.

4.19.2.11 UART2_RX_CALLBACK()

```
void UART2_RX_CALLBACK (
             void(*)(void) func )
```

Call Back Function To Handle ISR.

Parameters

func	

4.19.2.12 UART2_RXdata()

```
u32 UART2_RXdata (
            void  )
```

Recieve data.

Returns

u32

4.19.2.13 UART2_StrReceiveString()

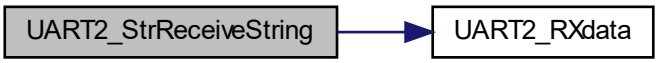
```
u8* const UART2_StrReceiveString (
            void  )
```

Recieve string.

Returns

u8∗ const

Here is the call graph for this function:



4.19.2.14 UART2_TXdata()

```
void UART2_TXdata (
            u32 data )
```

Transmit data.

Parameters

data	

4.19.2.15 UART2_voidSendString()

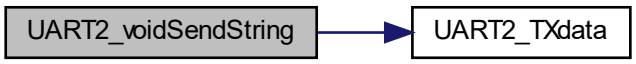
```
void UART2_voidSendString (
            char const * str )
```

Send string.

Parameters

str	

Here is the call graph for this function:



4.19.2.16 USART2_IRQHandler()

```
void USART2_IRQHandler (
            void  )
```

ISR of UART2.

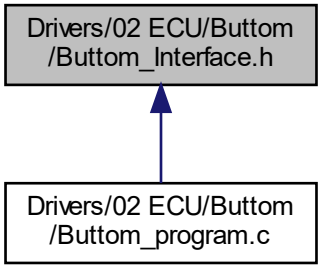
4.19.3 Variable Documentation

4.19.3.1 UART2_PTR

```
void(* UART2_PTR) (void)=0
```

4.20 Drivers/02 ECU/Buttom/Buttom_Interface.h File Reference

This graph shows which files directly or indirectly include this file:



Functions

- void Buttom_voidInit (u8 PORT, u8 PIN)
- u8 Buttom_u8GetValue (u8 PORT, u8 PIN)

4.20.1 Function Documentation

4.20.1.1 Buttom_u8GetValue()

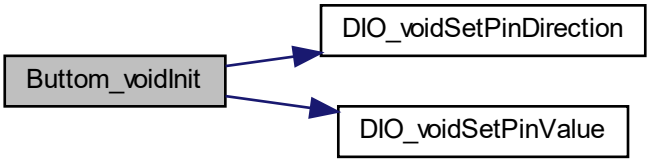
```
u8 Buttom_u8GetValue (
            u8 PORT,
            u8 PIN )
```

Here is the call graph for this function:

4.20.1.2 Buttom_voidInit()

```
void Buttom_voidInit (
            u8 PORT,
            u8 PIN )
```

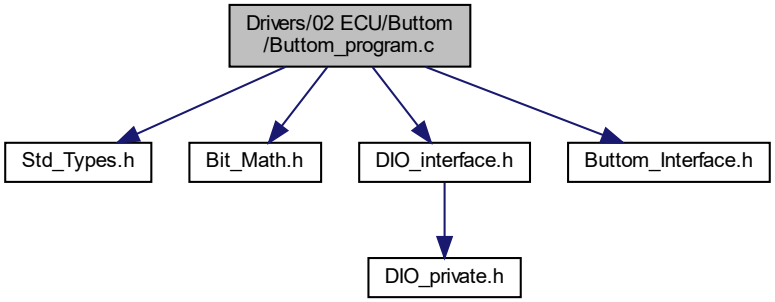
Here is the call graph for this function:



4.21 Drivers/02 ECU/Buttom/Buttom_program.c File Reference

```
#include "Std_Types.h"
#include "Bit_Math.h"
#include "DIO_interface.h"
#include "Buttom_Interface.h"
```

Include dependency graph for Buttom_program.c:



Functions

- void Buttom_voidInit (u8 PORT, u8 PIN)
- u8 Buttom_u8GetValue (u8 PORT, u8 PIN)

4.21.1 Function Documentation

4.21.1.1 Buttom_u8GetValue()

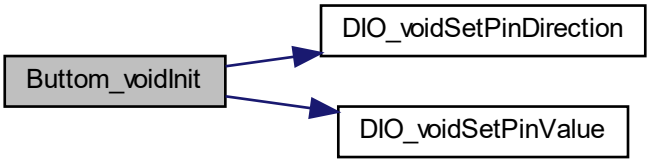
```
u8 Buttom_u8GetValue (
            u8 PORT,
            u8 PIN )
```

Here is the call graph for this function:

4.21.1.2 Buttom_voidInit()

```
void Buttom_voidInit (
            u8 PORT,
            u8 PIN )
```

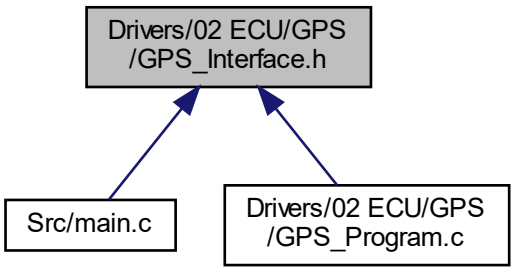
Here is the call graph for this function:



4.22 Drivers/02 ECU/GPS/GPS_Interface.h File Reference

GPS APIs and Global data.

This graph shows which files directly or indirectly include this file:



Data Structures

- struct GPS_RMC

Macros

- #define Buffer_size 80
- #define FRAME "GPRMC"

Typedefs

- typedef struct GPS_RMC GPS_RMC_ST

Functions

- void GPS_Init (void)
- void GPS_Decode_Buffer (u8 ∗longitude, u8 ∗latidude)
- void GPS_Decode_Buffer_st (GPS_RMC_ST ∗RMC)

  GPS Decode Data.
- void GPS_Frame_Concat (GPS_RMC_ST ∗RMC, char ∗buffer)

Variables

- u8 GPS_Complete_FRAME

  GPS Initialization.

4.22.1 Detailed Description

GPS APIs and Global data.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.22.2 Macro Definition Documentation

4.22.2.1 Buffer_size

```
#define Buffer_size 80
```

4.22.2.2 FRAME

```
#define FRAME "GPRMC"
```

4.22.3 Typedef Documentation

4.22.3.1 GPS_RMC_ST

```
typedef struct GPS_RMC GPS_RMC_ST
```

4.22.4 Function Documentation

4.22.4.1 GPS_Decode_Buffer()

```
void GPS_Decode_Buffer (
            u8 * longitude,
            u8 * latidude )
```

4.22.4.2 GPS_Decode_Buffer_st()

```
void GPS_Decode_Buffer_st (
            GPS_RMC_ST * RMC )
```

GPS Decode Data.

Parameters

RMC	Buffer Where Data Decoded

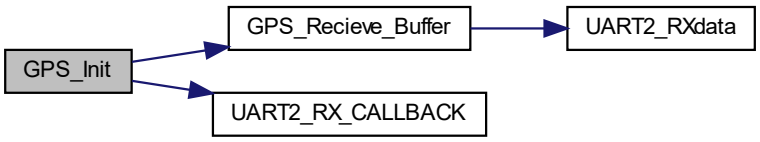
4.22.4.3 GPS_Frame_Concat()

```
void GPS_Frame_Concat (
            GPS_RMC_ST * RMC,
            char * buffer )
```

4.22.4.4 GPS_Init()

```
void GPS_Init (
            void  )
```

Here is the call graph for this function:



4.22.5 Variable Documentation

4.22.5.1 GPS_Complete_FRAME

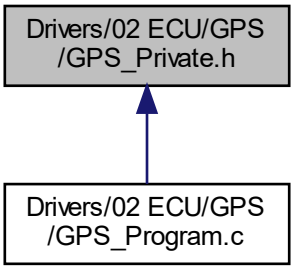
```
u8 GPS_Complete_FRAME
```

GPS Initialization.

4.23 Drivers/02 ECU/GPS/GPS_Private.h File Reference

GPS Private Data.

This graph shows which files directly or indirectly include this file:



Functions

- static void GPS_Recieve_Buffer (void)

Variables

- static u8 GPS_Buffer [Buffer_size]

4.23.1 Detailed Description

GPS Private Data.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.23.2 Function Documentation

4.23.2.1 GPS_Recieve_Buffer()

```
static void GPS_Recieve_Buffer (
            void  )  [static]
```

4.23.3 Variable Documentation

4.23.3.1 GPS_Buffer

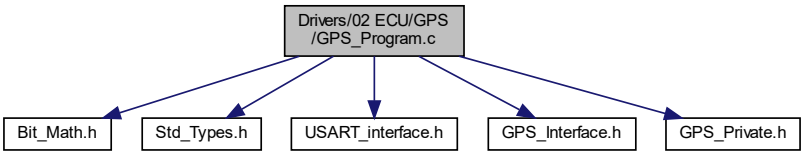
```
u8 GPS_Buffer[Buffer_size]  [static]
```

4.24 Drivers/02 ECU/GPS/GPS_Program.c File Reference

Implement GPS Driver.

```
#include "Bit_Math.h"
#include "Std_Types.h"
#include "USART_interface.h"
#include "GPS_Interface.h"
#include "GPS_Private.h"
```

Include dependency graph for GPS_Program.c:



Functions

- void GPS_Init (void)
- void GPS_Decode_Buffer_st (GPS_RMC_ST *RMC)

  GPS Decode Data.
- void GPS_Frame_Concat (GPS_RMC_ST *RMC, char *buffer)
- static void GPS_Recieve_Buffer (void)

  GPS Recive from UART.

Variables

- u8 GPS_Complete_FRAME = 0
  GPS Initialization.

4.24.1 Detailed Description

Implement GPS Driver.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.24.2 Function Documentation

4.24.2.1 GPS_Decode_Buffer_st()

```
void GPS_Decode_Buffer_st (
            GPS_RMC_ST * RMC )
```

GPS Decode Data.

Parameters

RMC	Buffer Where Data Decoded

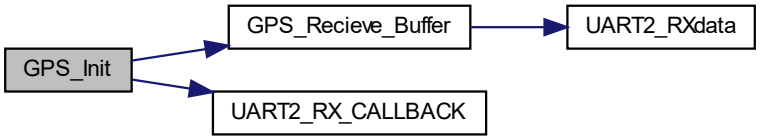
4.24.2.2 GPS_Frame_Concat()

```
void GPS_Frame_Concat (
            GPS_RMC_ST * RMC,
            char * buffer )
```

4.24.2.3 GPS_Init()

```
void GPS_Init (
            void  )
```

Here is the call graph for this function:

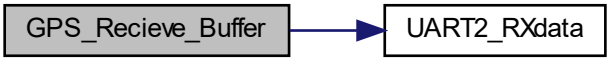


4.24.2.4 GPS_Recieve_Buffer()

```
static void GPS_Recieve_Buffer (
            void  )  [static]
```

GPS Recive from UART.

Here is the call graph for this function:



4.24.3 Variable Documentation

4.24.3.1 GPS_Complete_FRAME

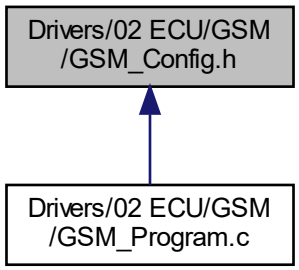
```
u8 GPS_Complete_FRAME = 0
```

GPS Initialization.

4.25 Drivers/02 ECU/GSM/GSM_Config.h File Reference

GSM Confguration of data.

This graph shows which files directly or indirectly include this file:



Macros

- #define NUMBER "01094488675"
- #define URL "http://127.0.0.1:8090/api/collections/points/records"

4.25.1 Detailed Description

GSM Confguration of data.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

4.25.2 Macro Definition Documentation

4.25.2.1 NUMBER

```
#define NUMBER "01094488675"
```

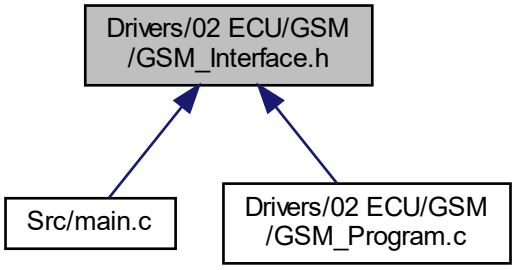
4.25.2.2 URL

```
#define URL "http://127.0.0.1:8090/api/collections/points/records"
```

4.26 Drivers/02 ECU/GSM/GSM_Interface.h File Reference

GSM APIs and Global Functions.

This graph shows which files directly or indirectly include this file:



Functions

- void GSM_INIT (void)

  Initialization Of GSM.
- void GSM_Post (char const ∗type, char const ∗data, char ∗buffer)

  GSM Send Data to Server.
- void GSM_MakeCall (void)

  Make A CAll to configured Numbed.
- void GSM_SendSMS (char const ∗SMS)

  Send SMS To Number.

4.26.1 Detailed Description

GSM APIs and Global Functions.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

Copyright (c) 2023

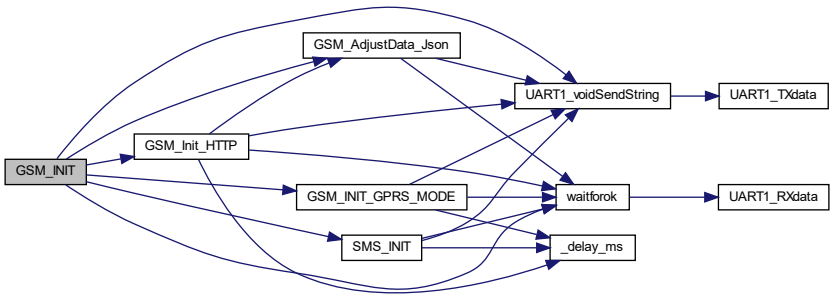
4.26.2 Function Documentation

4.26.2.1 GSM_INIT()

```
void GSM_INIT (
            void  )
```

Initialization Of GSM.

Here is the call graph for this function:

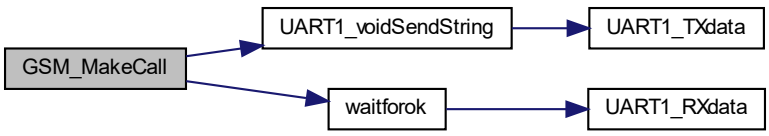


4.26.2.2 GSM_MakeCall()

```
void GSM_MakeCall (
             void  )
```

Make A CAll to configured Numbed.

Here is the call graph for this function:



4.26.2.3 GSM_Post()

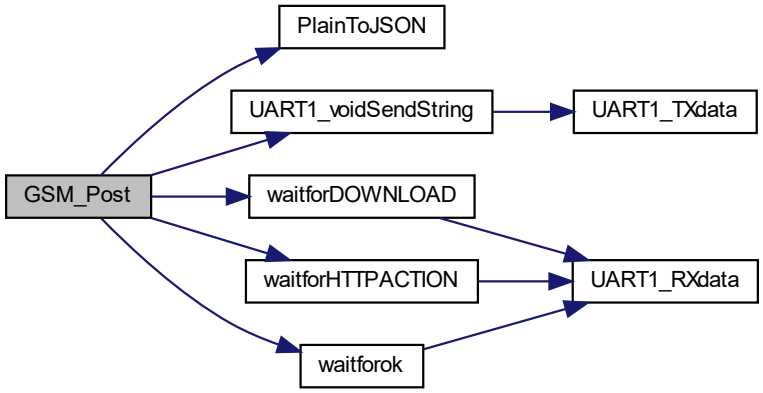
```
void GSM_Post (
             char const * type,
             char const * data,
             char * buffer )
```

GSM Send Data to Server.

Parameters

type	Type of data to send
data	data to send
buffer	Buffer to send data from it

Here is the call graph for this function:



4.26.2.4 GSM_SendSMS()

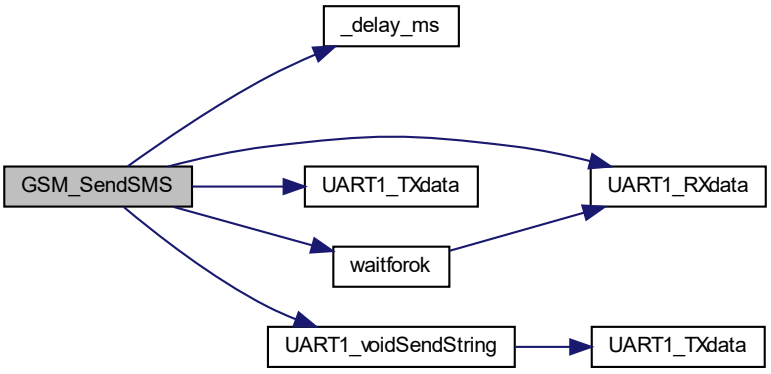
```
void GSM_SendSMS (
            char const * SMS )
```

Send SMS To Number.

Parameters

SMS	Content of SMS

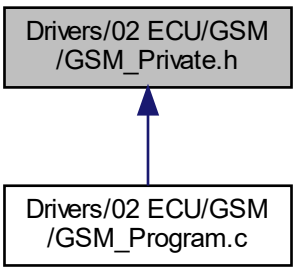
Here is the call graph for this function:



4.27 Drivers/02 ECU/GSM/GSM_Private.h File Reference

Private Data of GSM Module.

This graph shows which files directly or indirectly include this file:



Macros

- #define CALL_CONC "ATD" NUMBER "\r\n"
- #define SMS_CONC "AT+CMGS=\"" NUMBER ";\r\n"
- #define URL_CONC "AT+HTTPPARA=\"URL\",\"" URL "\"\r\n"
- #define CALL_CONC "ATD" NUMBER "\r\n"
- #define SMS_CONC "AT+CMGS=\"" NUMBER ";\r\n"
- #define URL_CONC "AT+HTTPPARA=\"URL\",\"" URL "\"\r\n"

Functions

- static void waitforok (void)
- static void waitforDOWNLOAD (void)
- static void waitforHTTPACTION (void)
- static void GSM_INIT_GPRS_MODE (void)
- static void GSM_Init_HTTP (void)
- static void GSM_AdjustData_Json (void)
- static void PlainToJSON (char const *type, char const *data, char *buffer)
- static void SMS_INIT (void)

4.27.1 Detailed Description

Private Data of GSM Module.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.27.2 Macro Definition Documentation

4.27.2.1 CALL_CONC [1/2]

```
#define CALL_CONC "ATD" NUMBER "\r\n"
```

4.27.2.2 CALL_CONC [2/2]

```
#define CALL_CONC "ATD" NUMBER "\r\n"
```

4.27.2.3 SMS_CONC [1/2]

```
#define SMS_CONC "AT+CMGS=\"" NUMBER ";\r\n"
```

4.27.2.4 SMS_CONC [2/2]

```
#define SMS_CONC "AT+CMGS=\"" NUMBER ";\r\n"
```

4.27.2.5 URL_CONC [1/2]

```
#define URL_CONC "AT+HTTPPARA=\"URL\",\"" URL "\"\r\n"
```

4.27.2.6 URL_CONC [2/2]

```
#define URL_CONC "AT+HTTPPARA=\"URL\",\"" URL "\"\r\n"
```

4.27.3 Function Documentation

4.27.3.1 GSM_AdjustData_Json()

```
static void GSM_AdjustData_Json (
            void  )  [static]
```

4.27.3.2 GSM_INIT_GPRS_MODE()

```
static void GSM_INIT_GPRS_MODE (
            void  )  [static]
```

4.27.3.3 GSM_Init_HTTP()

```
static void GSM_Init_HTTP (
            void  )  [static]
```

4.27.3.4 PlainToJSON()

```
static void PlainToJSON (
            char const * type,
            char const * data,
            char * buffer )  [static]
```

4.27.3.5 SMS_INIT()

```
static void SMS_INIT (
            void  )  [static]
```

4.27.3.6 waitforDOWNLOAD()

```
static void waitforDOWNLOAD (
            void  )  [static]
```

4.27.3.7 waitforHTTPACTION()

```
static void waitforHTTPACTION (
            void  )  [static]
```

4.27.3.8 waitforok()

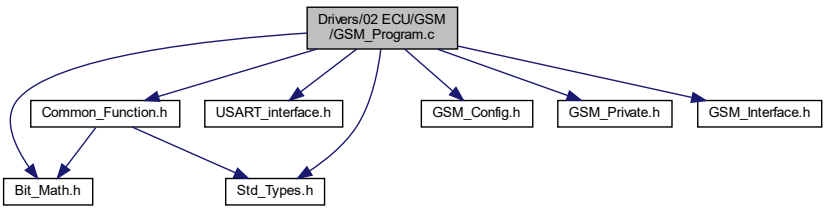
```
static void waitforok (
            void  )  [static]
```

4.28 Drivers/02 ECU/GSM/GSM_Program.c File Reference

Used To Implement Functions of GSM.

```
#include "Bit_Math.h"
#include "Std_Types.h"
#include "USART_interface.h"
#include "Common_Function.h"
#include "GSM_Config.h"
#include "GSM_Private.h"
#include "GSM_Interface.h"
```

Include dependency graph for GSM_Program.c:



Functions

- void GSM_INIT (void)

  Initialization Of GSM.
- void GSM_Post (char const ∗type, char const ∗data, char ∗buffer)

  GSM Send Data to Server.
- void GSM_MakeCall (void)

  Make A CAll to configured Numbed.
- void GSM_SendSMS (char const ∗SMS)

  Send SMS To Number.
- static void GSM_INIT_GPRS_MODE (void)

  Initialization of GPRS Mode.
- static void GSM_Init_HTTP (void)

  Initialization of HTTP.
- static void GSM_AdjustData_Json (void)

  Adjust Data format to be Written In Json.
- static void waitforHTTPACTION (void)

  Wait for HTTPACTION.
- static void waitforok (void)

  Wait for ok.
- static void waitforDOWNLOAD (void)

  wait for Download
- static void PlainToJSON (char const ∗type, char const ∗data, char ∗buffer)

  Convert Data to json Format.
- static void SMS_INIT (void)

  Send SMS Initialization.

4.28.1 Detailed Description

Used To Implement Functions of GSM.

Author

Mohamed Dwedar (MohamedDwedar2699@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

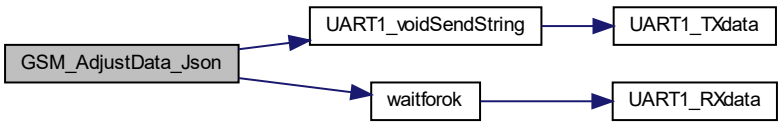
4.28.2 Function Documentation

4.28.2.1 GSM_AdjustData_Json()

```
static void GSM_AdjustData_Json (
             void  )  [static]
```

Adjust Data format to be Written In Json.

Here is the call graph for this function:

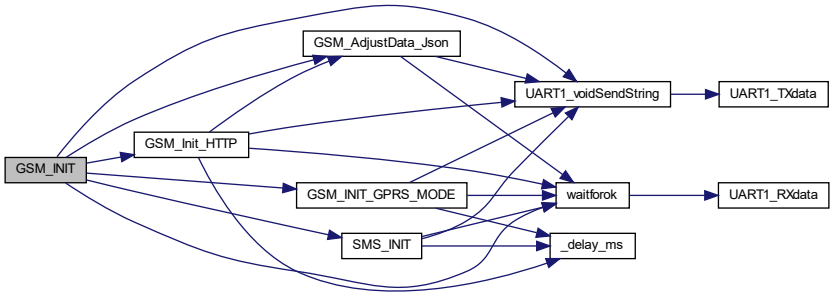


4.28.2.2 GSM_INIT()

```
void GSM_INIT (
             void  )
```

Initialization Of GSM.

Here is the call graph for this function:

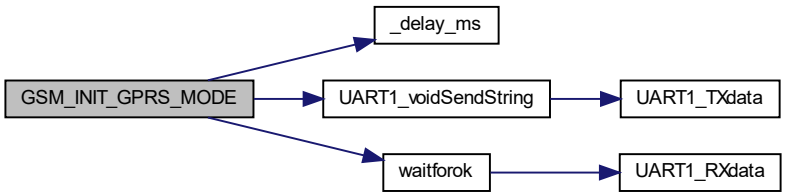


4.28.2.3 GSM_INIT_GPRS_MODE()

```
static void GSM_INIT_GPRS_MODE (
            void  )  [static]
```

Initialization of GPRS Mode.

Here is the call graph for this function:

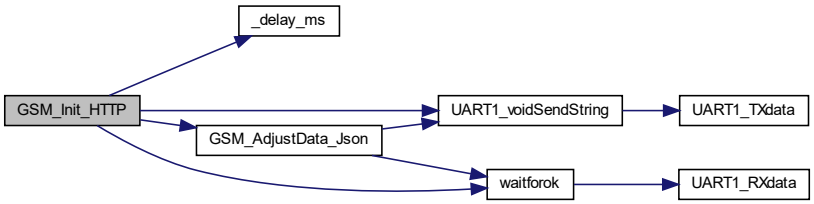


4.28.2.4 GSM_Init_HTTP()

```
static void GSM_Init_HTTP (
            void  )  [static]
```

Initialization of HTTP.

Here is the call graph for this function:

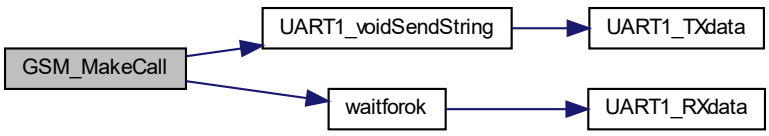


4.28.2.5 GSM_MakeCall()

```
void GSM_MakeCall (
             void  )
```

Make A CAll to configured Numbed.

Here is the call graph for this function:



4.28.2.6 GSM_Post()

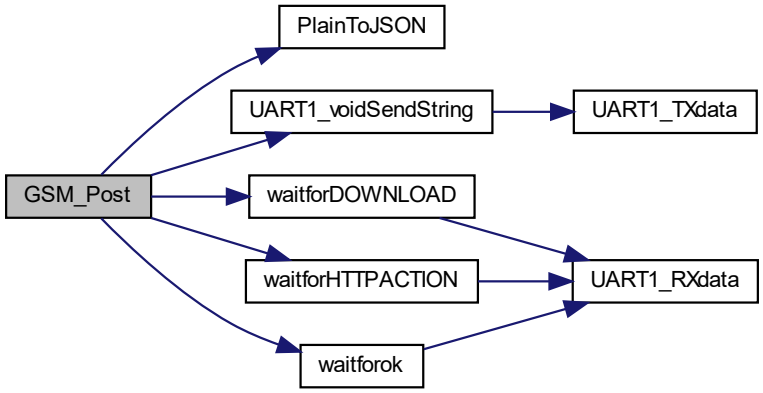
```
void GSM_Post (
             char const * type,
             char const * data,
             char * buffer )
```

GSM Send Data to Server.

Parameters

type	Type of data to send
data	data to send
buffer	Buffer to send data from it

Here is the call graph for this function:



4.28.2.7 GSM_SendSMS()

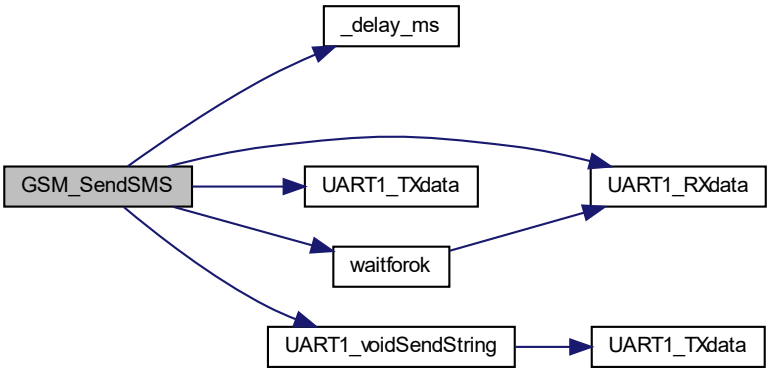
```
void GSM_SendSMS (
            char const * SMS )
```

Send SMS To Number.

Parameters

SMS	Content of SMS

Here is the call graph for this function:



4.28.2.8 PlainToJSON()

```
static void PlainToJSON (
            char const * type,
            char const * data,
            char * buffer )  [static]
```

Convert Data to json Format.

Parameters

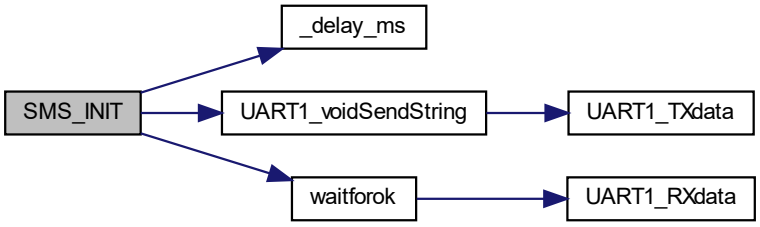
type	Type of data
data	data to content
buffer	buffer to save data

4.28.2.9 SMS_INIT()

```
static void SMS_INIT (
            void  )  [static]
```

Send SMS Initialization.

Here is the call graph for this function:

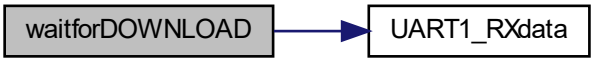


4.28.2.10 waitforDOWNLOAD()

```
static void waitforDOWNLOAD (
            void  )  [static]
```

wait for Download

Here is the call graph for this function:

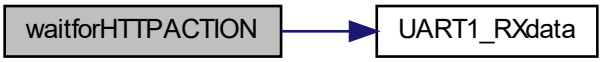


4.28.2.11 waitforHTTPACTION()

```
static void waitforHTTPACTION (
            void  )  [static]
```

Wait for HTTPACTION.

Here is the call graph for this function:

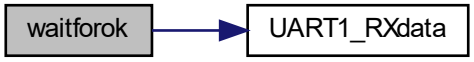


4.28.2.12 waitforok()

```
static void waitforok (
            void  )  [static]
```

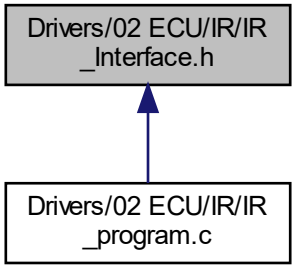
Wait for ok.

Here is the call graph for this function:



4.29 Drivers/02 ECU/IR/IR_Interface.h File Reference

This graph shows which files directly or indirectly include this file:



Functions

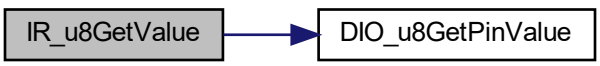
- void IR_voidInit (u8 PORT, u8 PIN)
- u8 IR_u8GetValue (u8 PORT, u8 PIN)

4.29.1 Function Documentation

4.29.1.1 IR_u8GetValue()

```
u8 IR_u8GetValue (
              u8 PORT,
              u8 PIN )
```

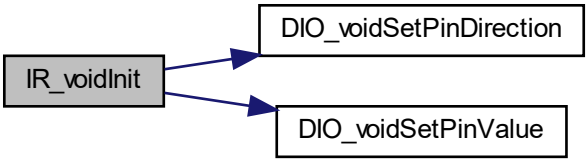
Here is the call graph for this function:



4.29.1.2 IR_voidInit()

```
void IR_voidInit (
             u8 PORT,
             u8 PIN )
```

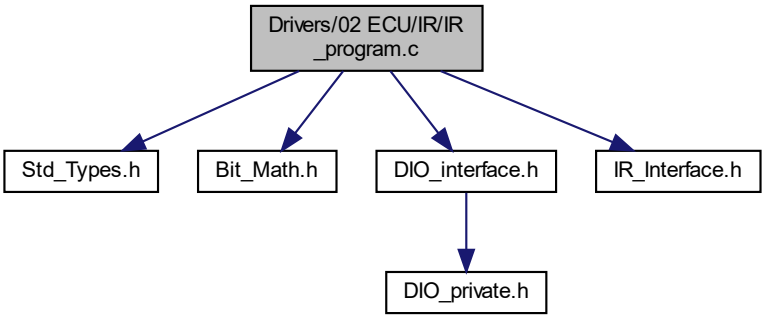
Here is the call graph for this function:



4.30 Drivers/02 ECU/IR/IR_program.c File Reference

```
#include "Std_Types.h"
#include "Bit_Math.h"
#include "DIO_interface.h"
#include "IR_Interface.h"
```

Include dependency graph for IR_program.c:



Functions

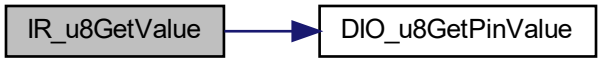
- void IR_voidInit (u8 PORT, u8 PIN)
- u8 IR_u8GetValue (u8 PORT, u8 PIN)

4.30.1 Function Documentation

4.30.1.1 IR_u8GetValue()

```
u8 IR_u8GetValue (
              u8 PORT,
              u8 PIN )
```

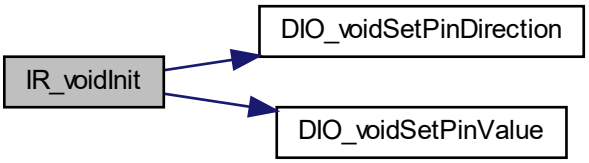
Here is the call graph for this function:



4.30.1.2 IR_voidInit()

```
void IR_voidInit (
              u8 PORT,
              u8 PIN )
```

Here is the call graph for this function:

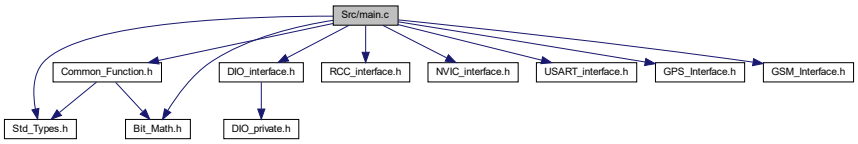


4.31 Src/main.c File Reference

Main Function To test GPS Performance ON TTL.

```
#include "Std_Types.h"
#include "Bit_Math.h"
#include "Common_Function.h"
#include "DIO_interface.h"
#include "RCC_interface.h"
#include "NVIC_interface.h"
#include "USART_interface.h"
#include "GPS_Interface.h"
#include "GSM_Interface.h"
```

Include dependency graph for main.c:



Functions

- int main (void)

  Main APP.

Variables

- GPS_RMC_ST RMC

4.31.1 Detailed Description

Main Function To test GPS Performance ON TTL.

Author

Mohamed Dwedar (MohamedDwedar@gmail.com)

Version

0.1

Date

2023-08-26

Copyright

4.31.2 Function Documentation

4.31.2.1 main()

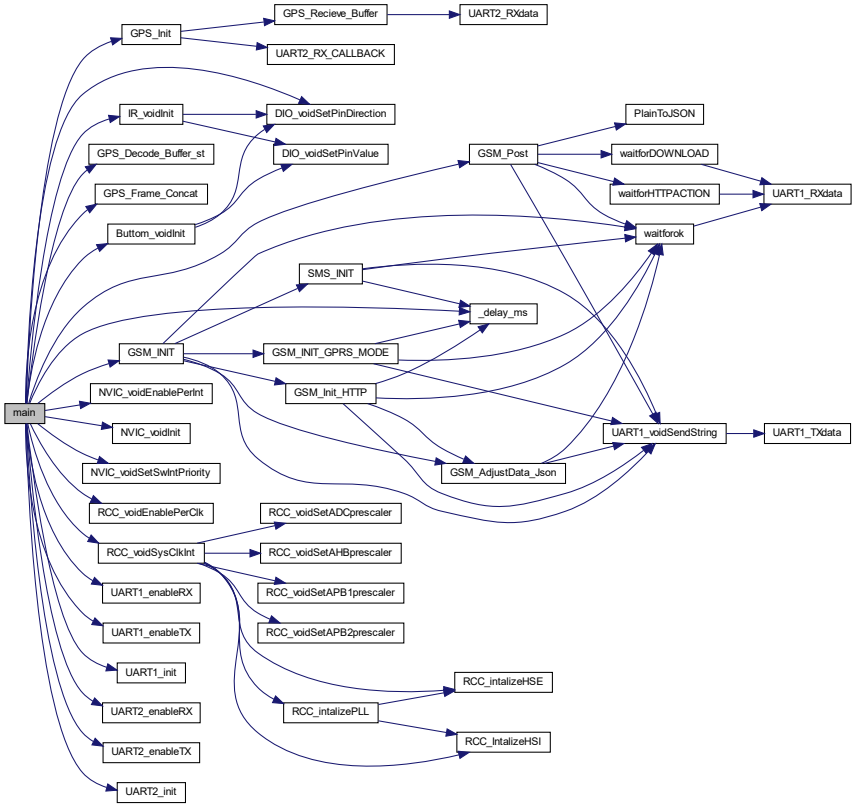
```
int main (
            void  )
```

Main APP.

Returns

int

Here is the call graph for this function:



4.31.3 Variable Documentation

4.31.3.1 RMC

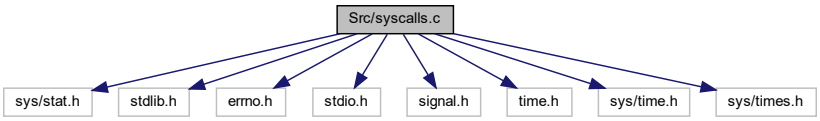
```
GPS_RMC_ST RMC
```

4.32 Src/syscalls.c File Reference

STM32CubeIDE Minimal System calls file.

```
#include <sys/stat.h>
#include <stdlib.h>
#include <errno.h>
#include <stdio.h>
#include <signal.h>
#include <time.h>
#include <sys/time.h>
#include <sys/times.h>
```

Include dependency graph for syscalls.c:



Functions

- int __io_putchar (int ch) __attribute__((weak))
- int __io_getchar (void)
- void initialise_monitor_handles ()
- int _getpid (void)
- int _kill (int pid, int sig)
- void _exit (int status)
- __attribute__ ((weak))
- int _close (int file)
- int _fstat (int file, struct stat *st)
- int _isatty (int file)
- int _lseek (int file, int ptr, int dir)
- int _open (char *path, int flags,...)
- int _wait (int *status)
- int _unlink (char *name)
- int _times (struct tms *buf)
- int _stat (char *file, struct stat *st)
- int _link (char *old, char *new)
- int _fork (void)
- int _execve (char *name, char **argv, char **env)

Variables

- char ** environ = __env

4.32.1 Detailed Description

STM32CubeIDE Minimal System calls file.

Author

Auto-generated by STM32CubeIDE

```
For more information about which c-functions
need which of these lowlevel functions
please consult the Newlib libc-manual
```

Attention

4.32.2 Function Documentation

4.32.2.1 __attribute__()

```
__attribute__ (
             (weak)  )
```

Here is the call graph for this function:

4.32.2.2 __io_getchar()

```
int __io_getchar (
             void  )
```

4.32.2.3 __io_putchar()

```
int __io_putchar (
            int ch )
```

4.32.2.4 _close()

```
int _close (
            int file )
```

4.32.2.5 _execve()

```
int _execve (
            char * name,
            char ** argv,
            char ** env )
```

4.32.2.6 _exit()

```
void _exit (
            int status )
```

Here is the call graph for this function:

4.32.2.7 _fork()

```
int _fork (
            void  )
```

4.32.2.8 _fstat()

```
int _fstat (
            int file,
            struct stat * st )
```

4.32.2.9 _getpid()

```
int _getpid (
            void  )
```

4.32.2.10 _isatty()

```
int _isatty (
            int file )
```

4.32.2.11 _kill()

```
int _kill (
            int pid,
            int sig )
```

4.32.2.12 _link()

```
int _link (
            char * old,
            char * new )
```

4.32.2.13 _lseek()

```
int _lseek (
            int file,
            int ptr,
            int dir )
```

4.32.2.14 _open()

```
int _open (
              char * path,
              int flags,
               ...  )
```

4.32.2.15 _stat()

```
int _stat (
              char * file,
              struct stat * st )
```

4.32.2.16 _times()

```
int _times (
              struct tms * buf )
```

4.32.2.17 _unlink()

```
int _unlink (
              char * name )
```

4.32.2.18 _wait()

```
int _wait (
              int * status )
```

4.32.2.19 initialise_monitor_handles()

```
void initialise_monitor_handles ( )
```

4.32.3 Variable Documentation

4.32.3.1 environ

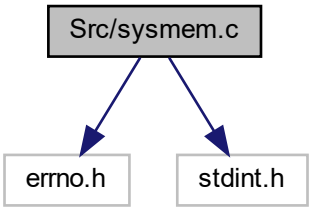
```
char** environ = __env
```

4.33 Src/sysmem.c File Reference

STM32CubeIDE System Memory calls file.

```
#include <errno.h>
#include <stdint.h>
```

Include dependency graph for sysmem.c:



Functions

- void * _sbrk (ptrdiff_t incr)

  _sbrk() allocates memory to the newlib heap and is used by malloc and others from the C library

Variables

- static uint8_t * __sbrk_heap_end = NULL

4.33.1 Detailed Description

STM32CubeIDE System Memory calls file.

Author

Generated by STM32CubeIDE

```
For more information about which C functions
need which of these lowlevel functions
please consult the newlib libc manual
```

Attention

4.33.2 Function Documentation

4.33.2.1 _sbrk()

```
void* _sbrk (
            ptrdiff_t incr )
```

_sbrk() allocates memory to the newlib heap and is used by malloc and others from the C library

```
* ##########################################################################
* #  .data  #  .bss  #       newlib heap       #          MSP stack          #
* #         #        #                         # Reserved by _Min_Stack_Size #
* ##########################################################################
* ^-- RAM start      ^-- _end                             _estack, RAM end --^
*
```

This implementation starts allocating at the '_end' linker symbol The '_Min_Stack_Size' linker symbol reserves a memory for the MSP stack The implementation considers '_estack' linker symbol to be RAM end NOTE: If the MSP stack, at any point during execution, grows larger than the reserved size, please increase the '_Min_Stack_Size'.

Parameters

incr	Memory size

Returns

Pointer to allocated memory

4.33.3 Variable Documentation

4.33.3.1 __sbrk_heap_end

```
uint8_t* __sbrk_heap_end = NULL  [static]
```

Pointer to the current high watermark of the heap usage

Index

__attribute__
    syscalls.c, 119
__io_getchar
    syscalls.c, 119
__io_putchar
    syscalls.c, 119
__sbrk_heap_end
    sysmem.c, 124
_close
    syscalls.c, 120
_delay_ms
    Common_function.c, 11
    Common_Function.h, 13
_execve
    syscalls.c, 120
_exit
    syscalls.c, 120
_fork
    syscalls.c, 120
_fstat
    syscalls.c, 120
_getpid
    syscalls.c, 121
_isatty
    syscalls.c, 121
_kill
    syscalls.c, 121
_link
    syscalls.c, 121
_lseek
    syscalls.c, 121
_open
    syscalls.c, 121
_sbrk
    sysmem.c, 124
_stat
    syscalls.c, 122
_times
    syscalls.c, 122
_unlink
    syscalls.c, 122
_wait
    syscalls.c, 122

ADC_APB2clkdiv2
    RCC_config.h, 19
ADC_APB2clkdiv4
    RCC_config.h, 19
ADC_APB2clkdiv6
    RCC_config.h, 19
ADC_APB2clkdiv8
    RCC_config.h, 19
ADC_prescaler
    RCC_config.h, 17
ADCPRE0
    RCC_interface.h, 25
ADCPRE1
    RCC_interface.h, 25
ADD
    USART_private.h, 73
AHB_prescaler
    RCC_config.h, 17
AHB_SysDiv128
    RCC_config.h, 18
AHB_SysDiv16
    RCC_config.h, 18
AHB_SysDiv2
    RCC_config.h, 18
AHB_SysDiv256
    RCC_config.h, 18
AHB_SysDiv4
    RCC_config.h, 18
AHB_SysDiv512
    RCC_config.h, 18
AHB_SysDiv64
    RCC_config.h, 18
AHB_SysDiv8
    RCC_config.h, 18
AHB_SysNotDiv
    RCC_config.h, 18
Angle
    GPS_RMC, 5
APB1_HCLKDiv16
    RCC_config.h, 19
APB1_HCLKDiv2
    RCC_config.h, 19
APB1_HCLKDiv4
    RCC_config.h, 19
APB1_HCLKDiv8
    RCC_config.h, 19
APB1_HCLKNotDiv
    RCC_config.h, 18
APB1_prescaler
    RCC_config.h, 17
APB2_HCLKDiv16
    RCC_config.h, 19
APB2_HCLKDiv2
    RCC_config.h, 19
APB2_HCLKDiv4

RCC_config.h, 19
APB2_HCLKDiv8
RCC_config.h, 19
APB2_HCLKNotDiv
RCC_config.h, 19
APB2_prescaler
RCC_config.h, 17

BAUD_RATE_115200
USART_private.h, 73
BAUD_RATE_9600
USART_private.h, 73
Bit_Math.h
CLR_BIT, 10
GET_BIT, 10
SET_BIT, 10
TOG_BIT, 10
BRR
USART_type, 7
Buffer_size
GPS_Interface.h, 91
Buttom_Interface.h
Buttom_u8GetValue, 87
Buttom_voidInit, 87
Buttom_program.c
Buttom_u8GetValue, 88
Buttom_voidInit, 89
Buttom_u8GetValue
Buttom_Interface.h, 87
Buttom_program.c, 88
Buttom_voidInit
Buttom_Interface.h, 87
Buttom_program.c, 89

CALL_CONC
GSM_Private.h, 103
CLKEN
USART_private.h, 73
clocksource
RCC_config.h, 19
CLR_BIT
Bit_Math.h, 10
Common_function.c
_delay_ms, 11
Common_Function.h
_delay_ms, 13
CPHA
USART_private.h, 73
CPOL
USART_private.h, 73
CR1
USART_type, 7
CR2
USART_type, 7
CR3
USART_type, 7
CSSON
RCC_interface.h, 26
CTS
USART_private.h, 73
CTSE
USART_private.h, 73
CTSIE
USART_private.h, 74

DATA_LEN_8
USART_private.h, 74
DATA_LEN_9
USART_private.h, 74
Date
GPS_RMC, 5
DIO_interface.h
DIO_u8GetPinValue, 43
DIO_voidSetPinDirection, 44
DIO_voidSetPinValue, 44
DIO_voidtogglepin, 44
DIO_voidWriteValue, 45
GPIO_HIGH, 39
GPIO_INPUT_ANALOG, 39
GPIO_INPUT_FLOAT, 39
GPIO_INPUT_PULL_UP_DOWN, 39
GPIO_LOW, 39
GPIO_OUTPUT_10MHZ_AFOD, 39
GPIO_OUTPUT_10MHZ_AFPP, 39
GPIO_OUTPUT_10MHZ_OD, 39
GPIO_OUTPUT_10MHZ_PP, 40
GPIO_OUTPUT_2MHZ_AFOD, 40
GPIO_OUTPUT_2MHZ_AFPP, 40
GPIO_OUTPUT_2MHZ_OD, 40
GPIO_OUTPUT_2MHZ_PP, 40
GPIO_OUTPUT_50MHZ_AFOD, 40
GPIO_OUTPUT_50MHZ_AFPP, 40
GPIO_OUTPUT_50MHZ_OD, 40
GPIO_OUTPUT_50MHZ_PP, 41
PIN0, 41
PIN1, 41
PIN10, 41
PIN11, 41
PIN12, 41
PIN13, 41
PIN14, 41
PIN15, 42
PIN2, 42
PIN3, 42
PIN4, 42
PIN5, 42
PIN6, 42
PIN7, 42
PIN8, 42
PIN9, 43
PORTA, 43
PORTB, 43
PORTC, 43
DIO_private.h
GPIOA_BASE_ADD, 46
GPIOA_BRR, 46
GPIOA_BSRR, 47
GPIOA_CRH, 47

GPIOA_CRL, 47
GPIOA_IDR, 47
GPIOA_LCKR, 47
GPIOA_ODR, 47
GPIOB_BASE_ADD, 47
GPIOB_BRR, 47
GPIOB_BSRR, 48
GPIOB_CRH, 48
GPIOB_CRL, 48
GPIOB_IDR, 48
GPIOB_LCKR, 48
GPIOB_ODR, 48
GPIOC_BASE_ADD, 48
GPIOC_BRR, 48
GPIOC_BSRR, 49
GPIOC_CRH, 49
GPIOC_CRL, 49
GPIOC_IDR, 49
GPIOC_LCKR, 49
GPIOC_ODR, 49
DIO_program.c
DIO_u8GetPinValue, 51
DIO_voidSetPinDirection, 51
DIO_voidSetPinValue, 52
DIO_voidtogglepin, 52
DIO_voidWriteValue, 52
DIO_u8GetPinValue
DIO_interface.h, 43
DIO_program.c, 51
DIO_voidSetPinDirection
DIO_interface.h, 44
DIO_program.c, 51
DIO_voidSetPinValue
DIO_interface.h, 44
DIO_program.c, 52
DIO_voidtogglepin
DIO_interface.h, 44
DIO_program.c, 52
DIO_voidWriteValue
DIO_interface.h, 45
DIO_program.c, 52
divided2
RCC_interface.h, 25
divided4
RCC_interface.h, 25
divided6
RCC_interface.h, 25
divided8
RCC_interface.h, 25
DMAR
USART_private.h, 74
DMAT
USART_private.h, 74
DR
USART_type, 7
Drivers/00 LIB/Bit_Math.h, 9
Drivers/00 LIB/Common_function.c, 10
Drivers/00 LIB/Common_Function.h, 12
Drivers/00 LIB/Std_Types.h, 13
Drivers/01 MCAL/00 RCC/RCC_config.h, 16
Drivers/01 MCAL/00 RCC/RCC_interface.h, 20
Drivers/01 MCAL/00 RCC/RCC_private.h, 29
Drivers/01 MCAL/00 RCC/RCC_program.c, 32
Drivers/01 MCAL/01 GPIO/DIO_interface.h, 36
Drivers/01 MCAL/01 GPIO/DIO_private.h, 45
Drivers/01 MCAL/01 GPIO/DIO_program.c, 50
Drivers/01 MCAL/02 NVIC/NVIC_config.h, 53
Drivers/01 MCAL/02 NVIC/NVIC_interface.h, 54
Drivers/01 MCAL/02 NVIC/NVIC_private.h, 56
Drivers/01 MCAL/02 NVIC/NVIC_program.c, 59
Drivers/01 MCAL/06 USART/USART_config.h, 62
Drivers/01 MCAL/06 USART/USART_interface.h, 64
Drivers/01 MCAL/06 USART/USART_private.h, 70
Drivers/01 MCAL/06 USART/USART_program.c, 80
Drivers/02 ECU/Buttom/Buttom_Interface.h, 87
Drivers/02 ECU/Buttom/Buttom_program.c, 88
Drivers/02 ECU/GPS/GPS_Interface.h, 89
Drivers/02 ECU/GPS/GPS_Private.h, 93
Drivers/02 ECU/GPS/GPS_Program.c, 94
Drivers/02 ECU/GSM/GSM_Config.h, 97
Drivers/02 ECU/GSM/GSM_Interface.h, 98
Drivers/02 ECU/GSM/GSM_Private.h, 102
Drivers/02 ECU/GSM/GSM_Program.c, 105
Drivers/02 ECU/IR/IR_Interface.h, 113
Drivers/02 ECU/IR/IR_program.c, 114

E_W
GPS_RMC, 5
EIE
USART_private.h, 74
environ
syscalls.c, 122

f32
Std_Types.h, 14
f64
Std_Types.h, 14
FE
USART_private.h, 74
FRAME
GPS_Interface.h, 91
Full_HSE
RCC_config.h, 19

GET_BIT
Bit_Math.h, 10
GPIO_HIGH
DIO_interface.h, 39
GPIO_INPUT_ANALOG
DIO_interface.h, 39
GPIO_INPUT_FLOAT
DIO_interface.h, 39
GPIO_INPUT_PULL_UP_DOWN
DIO_interface.h, 39
GPIO_LOW
DIO_interface.h, 39
GPIO_OUTPUT_10MHZ_AFOD

DIO_interface.h, 39
GPIO_OUTPUT_10MHZ_AFPP
DIO_interface.h, 39
GPIO_OUTPUT_10MHZ_OD
DIO_interface.h, 39
GPIO_OUTPUT_10MHZ_PP
DIO_interface.h, 40
GPIO_OUTPUT_2MHZ_AFOD
DIO_interface.h, 40
GPIO_OUTPUT_2MHZ_AFPP
DIO_interface.h, 40
GPIO_OUTPUT_2MHZ_OD
DIO_interface.h, 40
GPIO_OUTPUT_2MHZ_PP
DIO_interface.h, 40
GPIO_OUTPUT_50MHZ_AFOD
DIO_interface.h, 40
GPIO_OUTPUT_50MHZ_AFPP
DIO_interface.h, 40
GPIO_OUTPUT_50MHZ_OD
DIO_interface.h, 40
GPIO_OUTPUT_50MHZ_PP
DIO_interface.h, 41
GPIOA_BASE_ADD
DIO_private.h, 46
GPIOA_BRR
DIO_private.h, 46
GPIOA_BSRR
DIO_private.h, 47
GPIOA_CRH
DIO_private.h, 47
GPIOA_CRL
DIO_private.h, 47
GPIOA_IDR
DIO_private.h, 47
GPIOA_LCKR
DIO_private.h, 47
GPIOA_ODR
DIO_private.h, 47
GPIOB_BASE_ADD
DIO_private.h, 47
GPIOB_BRR
DIO_private.h, 47
GPIOB_BSRR
DIO_private.h, 48
GPIOB_CRH
DIO_private.h, 48
GPIOB_CRL
DIO_private.h, 48
GPIOB_IDR
DIO_private.h, 48
GPIOB_LCKR
DIO_private.h, 48
GPIOB_ODR
DIO_private.h, 48
GPIOC_BASE_ADD
DIO_private.h, 48
GPIOC_BRR
DIO_private.h, 48
GPIOC_BSRR
DIO_private.h, 49
GPIOC_CRH
DIO_private.h, 49
GPIOC_CRL
DIO_private.h, 49
GPIOC_IDR
DIO_private.h, 49
GPIOC_LCKR
DIO_private.h, 49
GPIOC_ODR
DIO_private.h, 49
GPS_Buffer
GPS_Private.h, 94
GPS_Complete_FRAME
GPS_Interface.h, 92
GPS_Program.c, 96
GPS_Decode_Buffer
GPS_Interface.h, 91
GPS_Decode_Buffer_st
GPS_Interface.h, 92
GPS_Program.c, 95
GPS_Frame_Concat
GPS_Interface.h, 92
GPS_Program.c, 95
GPS_Init
GPS_Interface.h, 92
GPS_Program.c, 96
GPS_Interface.h
Buffer_size, 91
FRAME, 91
GPS_Complete_FRAME, 92
GPS_Decode_Buffer, 91
GPS_Decode_Buffer_st, 92
GPS_Frame_Concat, 92
GPS_Init, 92
GPS_RMC_ST, 91
GPS_Private.h
GPS_Buffer, 94
GPS_Recieve_Buffer, 94
GPS_Program.c
GPS_Complete_FRAME, 96
GPS_Decode_Buffer_st, 95
GPS_Frame_Concat, 95
GPS_Init, 96
GPS_Recieve_Buffer, 96
GPS_Recieve_Buffer
GPS_Private.h, 94
GPS_Program.c, 96
GPS_RMC, 5
Angle, 5
Date, 5
E_W, 5
Latitude, 6
Longitude, 6
N_S, 6
UTC_Time, 6

GPS_RMC_ST
GPS_Interface.h, 91
GRB_00_SUB_16
NVIC_private.h, 57
GRB_02_SUB_08
NVIC_private.h, 57
GRB_04_SUB_04
NVIC_private.h, 57
GRB_08_SUB_02
NVIC_private.h, 57
GRB_16_SUB_00
NVIC_private.h, 57
GSM_AdjustData_Json
GSM_Private.h, 104
GSM_Program.c, 106
GSM_Config.h
NUMBER, 97
URL, 98
GSM_INIT
GSM_Interface.h, 99
GSM_Program.c, 107
GSM_INIT_GPRS_MODE
GSM_Private.h, 104
GSM_Program.c, 107
GSM_Init_HTTP
GSM_Private.h, 104
GSM_Program.c, 108
GSM_Interface.h
GSM_INIT, 99
GSM_MakeCall, 99
GSM_Post, 100
GSM_SendSMS, 101
GSM_MakeCall
GSM_Interface.h, 99
GSM_Program.c, 108
GSM_Post
GSM_Interface.h, 100
GSM_Program.c, 109
GSM_Private.h
CALL_CONC, 103
GSM_AdjustData_Json, 104
GSM_INIT_GPRS_MODE, 104
GSM_Init_HTTP, 104
PlainToJSON, 104
SMS_CONC, 103
SMS_INIT, 104
URL_CONC, 103
waitforDOWNLOAD, 104
waitforHTTPACTION, 105
waitforok, 105
GSM_Program.c
GSM_AdjustData_Json, 106
GSM_INIT, 107
GSM_INIT_GPRS_MODE, 107
GSM_Init_HTTP, 108
GSM_MakeCall, 108
GSM_Post, 109
GSM_SendSMS, 110
PlainToJSON, 111
SMS_INIT, 111
waitforDOWNLOAD, 111
waitforHTTPACTION, 112
waitforok, 112
GSM_SendSMS
GSM_Interface.h, 101
GSM_Program.c, 110
GT
USART_private.h, 74
GTPR
USART_type, 7

Half_HSE
RCC_config.h, 19
HALF_HSI
RCC_config.h, 19
HDSEL
USART_private.h, 75
HPRE0
RCC_interface.h, 25
HPRE1
RCC_interface.h, 25
HPRE2
RCC_interface.h, 25
HPRE3
RCC_interface.h, 25
HSEBYP
RCC_interface.h, 26
HSEON
RCC_interface.h, 26
HSERDY
RCC_interface.h, 26
HSICAL0
RCC_interface.h, 26
HSICAL1
RCC_interface.h, 26
HSICAL2
RCC_interface.h, 26
HSICAL3
RCC_interface.h, 26
HSICAL4
RCC_interface.h, 26
HSICAL5
RCC_interface.h, 26
HSICAL6
RCC_interface.h, 26
HSICAL7
RCC_interface.h, 26
HSION
RCC_interface.h, 26
HSIRDY
RCC_interface.h, 26
HSITRIM0
RCC_interface.h, 26
HSITRIM1
RCC_interface.h, 26
HSITRIM2
RCC_interface.h, 26

HSITRIM3
RCC_interface.h, 26
HSITRIM4
RCC_interface.h, 26

IDLE
USART_private.h, 75
IDLEIE
USART_private.h, 75
initialise_monitor_handles
syscalls.c, 122
IR_Interface.h
IR_u8GetValue, 113
IR_voidInit, 113
IR_program.c
IR_u8GetValue, 115
IR_voidInit, 115
IR_u8GetValue
IR_Interface.h, 113
IR_program.c, 115
IR_voidInit
IR_Interface.h, 113
IR_program.c, 115
IREN
USART_private.h, 75
IRLP
USART_private.h, 75

Latitude
GPS_RMC, 6
LBCL
USART_private.h, 75
LBD
USART_private.h, 75
LBDIE
USART_private.h, 75
LBDL
USART_private.h, 76
LINEN
USART_private.h, 76
Longitude
GPS_RMC, 6

M
USART_private.h, 76
main
main.c, 117
main.c
main, 117
RMC, 117
MCO0
RCC_interface.h, 25
MCO1
RCC_interface.h, 25
MCO2
RCC_interface.h, 25

N_S
GPS_RMC, 6
NACK
USART_private.h, 76
NE
USART_private.h, 76
NO_PLL
RCC_config.h, 19
NUMBER
GSM_Config.h, 97
NVIC_BASE_ADD
NVIC_private.h, 58
NVIC_config.h
NVIC_NUM_GRP_SUB, 53
NVIC_IABR0
NVIC_private.h, 58
NVIC_IABR1
NVIC_private.h, 58
NVIC_ICER0
NVIC_private.h, 58
NVIC_ICER1
NVIC_private.h, 58
NVIC_ICPR0
NVIC_private.h, 58
NVIC_ICPR1
NVIC_private.h, 58
NVIC_interface.h
NVIC_u8ReadActiveFlag, 55
NVIC_voidClrPendingFlag, 55
NVIC_voidDisablePerInt, 55
NVIC_voidEnablePerInt, 55
NVIC_voidInit, 55
NVIC_voidSetPendingFlag, 55
NVIC_voidSetSwIntPriority, 55
NVIC_IPR
NVIC_private.h, 58
NVIC_ISER0
NVIC_private.h, 59
NVIC_ISER1
NVIC_private.h, 59
NVIC_ISPR0
NVIC_private.h, 59
NVIC_ISPR1
NVIC_private.h, 59
NVIC_NUM_GRP_SUB
NVIC_config.h, 53
NVIC_private.h
GRB_00_SUB_16, 57
GRB_02_SUB_08, 57
GRB_04_SUB_04, 57
GRB_08_SUB_02, 57
GRB_16_SUB_00, 57
NVIC_BASE_ADD, 58
NVIC_IABR0, 58
NVIC_IABR1, 58
NVIC_ICER0, 58
NVIC_ICER1, 58
NVIC_ICPR0, 58
NVIC_ICPR1, 58
NVIC_IPR, 58

NVIC_ISER0, 59
NVIC_ISER1, 59
NVIC_ISPR0, 59
NVIC_ISPR1, 59
NVIC_program.c
NVIC_u8ReadActiveFlag, 61
NVIC_voidClrPendingFlag, 61
NVIC_voidDisablePerInt, 61
NVIC_voidEnablePerInt, 61
NVIC_voidInit, 61
NVIC_voidSetPendingFlag, 61
NVIC_voidSetSwIntPriority, 61
SCB_AIRCR, 60
SCB_BASE_ADD, 60
NVIC_u8ReadActiveFlag
NVIC_interface.h, 55
NVIC_program.c, 61
NVIC_voidClrPendingFlag
NVIC_interface.h, 55
NVIC_program.c, 61
NVIC_voidDisablePerInt
NVIC_interface.h, 55
NVIC_program.c, 61
NVIC_voidEnablePerInt
NVIC_interface.h, 55
NVIC_program.c, 61
NVIC_voidInit
NVIC_interface.h, 55
NVIC_program.c, 61
NVIC_voidSetPendingFlag
NVIC_interface.h, 55
NVIC_program.c, 61
NVIC_voidSetSwIntPriority
NVIC_interface.h, 55
NVIC_program.c, 61

ORE
USART_private.h, 76

PARITY_EVEN
USART_private.h, 76
PARITY_NO
USART_private.h, 76
PARITY_ODD
USART_private.h, 77
PARITY_YES
USART_private.h, 77
PCE
USART_private.h, 77
PE
USART_private.h, 77
PEIE
USART_private.h, 77
PIN0
DIO_interface.h, 41
PIN1
DIO_interface.h, 41
PIN10
DIO_interface.h, 41
PIN11
DIO_interface.h, 41
PIN12
DIO_interface.h, 41
PIN13
DIO_interface.h, 41
PIN14
DIO_interface.h, 41
PIN15
DIO_interface.h, 42
PIN2
DIO_interface.h, 42
PIN3
DIO_interface.h, 42
PIN4
DIO_interface.h, 42
PIN5
DIO_interface.h, 42
PIN6
DIO_interface.h, 42
PIN7
DIO_interface.h, 42
PIN8
DIO_interface.h, 42
PIN9
DIO_interface.h, 43
PlainToJSON
GSM_Private.h, 104
GSM_Program.c, 111
PLL_clocksource
RCC_config.h, 17
pll_inputclockX10
RCC_config.h, 20
pll_inputclockX11
RCC_config.h, 20
pll_inputclockX12
RCC_config.h, 20
pll_inputclockX13
RCC_config.h, 20
pll_inputclockX14
RCC_config.h, 20
pll_inputclockX15
RCC_config.h, 20
pll_inputclockX16
RCC_config.h, 20
pll_inputclockX2
RCC_config.h, 20
pll_inputclockX3
RCC_config.h, 20
pll_inputclockX4
RCC_config.h, 20
pll_inputclockX5
RCC_config.h, 20
pll_inputclockX6
RCC_config.h, 20
pll_inputclockX7
RCC_config.h, 20
pll_inputclockX8

RCC_config.h, 20
pll_inputclockX9
RCC_config.h, 20
pll_multiplication
RCC_config.h, 20
pll_Multiplication_factor
RCC_config.h, 18
PLLMUL0
RCC_interface.h, 25
PLLMUL1
RCC_interface.h, 25
PLLMUL2
RCC_interface.h, 25
PLLMUL3
RCC_interface.h, 25
PLLON
RCC_interface.h, 26
PLLRDY
RCC_interface.h, 26
PLLSRC
RCC_interface.h, 25
PLLXTPRE
RCC_interface.h, 25
PORTA
DIO_interface.h, 43
PORTB
DIO_interface.h, 43
PORTC
DIO_interface.h, 43
PPRE10
RCC_interface.h, 25
PPRE11
RCC_interface.h, 25
PPRE12
RCC_interface.h, 25
PPRE20
RCC_interface.h, 25
PPRE21
RCC_interface.h, 25
PPRE22
RCC_interface.h, 25
prescalers
RCC_interface.h, 25
PS
USART_private.h, 77
PSC
USART_private.h, 77

RCC_ADC_SETprescaler
RCC_interface.h, 26
RCC_program.c, 33
RCC_AHB
RCC_interface.h, 23
RCC_AHB_CRC
RCC_interface.h, 23
RCC_AHB_DMA1
RCC_interface.h, 23
RCC_AHB_DMA2
RCC_interface.h, 23
RCC_AHB_FLITF
RCC_interface.h, 23
RCC_AHB_FSMC
RCC_interface.h, 23
RCC_AHB_SDIO
RCC_interface.h, 23
RCC_AHB_SRAM
RCC_interface.h, 23
RCC_AHBENR
RCC_private.h, 31
RCC_APB1
RCC_interface.h, 23
RCC_APB1_BKP
RCC_interface.h, 24
RCC_APB1_CAN
RCC_interface.h, 24
RCC_APB1_DAC
RCC_interface.h, 24
RCC_APB1_I2C1
RCC_interface.h, 24
RCC_APB1_I2C2
RCC_interface.h, 24
RCC_APB1_PWR
RCC_interface.h, 24
RCC_APB1_SPI2
RCC_interface.h, 24
RCC_APB1_SPI3
RCC_interface.h, 24
RCC_APB1_TIM12
RCC_interface.h, 24
RCC_APB1_TIM13
RCC_interface.h, 24
RCC_APB1_TIM14
RCC_interface.h, 24
RCC_APB1_TIM2
RCC_interface.h, 24
RCC_APB1_TIM3
RCC_interface.h, 24
RCC_APB1_TIM4
RCC_interface.h, 24
RCC_APB1_TIM5
RCC_interface.h, 24
RCC_APB1_TIM6
RCC_interface.h, 24
RCC_APB1_TIM7
RCC_interface.h, 24
RCC_APB1_USART2
RCC_interface.h, 24
RCC_APB1_USART3
RCC_interface.h, 24
RCC_APB1_USART4
RCC_interface.h, 24
RCC_APB1_USART5
RCC_interface.h, 24
RCC_APB1_USB
RCC_interface.h, 24
RCC_APB1_WWWG
RCC_interface.h, 24

RCC_APB1ENR
RCC_private.h, 31
RCC_APB1RSTR
RCC_private.h, 31
RCC_APB2
RCC_interface.h, 23
RCC_APB2_ADC1
RCC_interface.h, 24
RCC_APB2_ADC2
RCC_interface.h, 24
RCC_APB2_ADC3
RCC_interface.h, 24
RCC_APB2_AFIO
RCC_interface.h, 24
RCC_APB2_DIOA
RCC_interface.h, 24
RCC_APB2_DIOB
RCC_interface.h, 24
RCC_APB2_DIOC
RCC_interface.h, 24
RCC_APB2_DIOD
RCC_interface.h, 24
RCC_APB2_DIOE
RCC_interface.h, 24
RCC_APB2_DIOF
RCC_interface.h, 24
RCC_APB2_DIOG
RCC_interface.h, 24
RCC_APB2_SPI1
RCC_interface.h, 24
RCC_APB2_TIM1
RCC_interface.h, 24
RCC_APB2_TIM10
RCC_interface.h, 24
RCC_APB2_TIM11
RCC_interface.h, 24
RCC_APB2_TIM8
RCC_interface.h, 24
RCC_APB2_TIM9
RCC_interface.h, 24
RCC_APB2_USART1
RCC_interface.h, 24
RCC_APB2ENR
RCC_private.h, 31
RCC_APB2RSTR
RCC_private.h, 31
RCC_BASE_ADD
RCC_private.h, 31
RCC_BDCR
RCC_private.h, 31
RCC_CFGR
RCC_private.h, 31
RCC_cfgr
RCC_interface.h, 25
RCC_CIR
RCC_private.h, 32
RCC_config.h
ADC_APB2clkdiv2, 19
ADC_APB2clkdiv4, 19
ADC_APB2clkdiv6, 19
ADC_APB2clkdiv8, 19
ADC_prescaler, 17
AHB_prescaler, 17
AHB_SysDiv128, 18
AHB_SysDiv16, 18
AHB_SysDiv2, 18
AHB_SysDiv256, 18
AHB_SysDiv4, 18
AHB_SysDiv512, 18
AHB_SysDiv64, 18
AHB_SysDiv8, 18
AHB_SysNotDiv, 18
APB1_HCLKDiv16, 19
APB1_HCLKDiv2, 19
APB1_HCLKDiv4, 19
APB1_HCLKDiv8, 19
APB1_HCLKNotDiv, 18
APB1_prescaler, 17
APB2_HCLKDiv16, 19
APB2_HCLKDiv2, 19
APB2_HCLKDiv4, 19
APB2_HCLKDiv8, 19
APB2_HCLKNotDiv, 19
APB2_prescaler, 17
clocksource, 19
Full_HSE, 19
Half_HSE, 19
HALF_HSI, 19
NO_PLL, 19
PLL_clocksource, 17
pll_inputclockX10, 20
pll_inputclockX11, 20
pll_inputclockX12, 20
pll_inputclockX13, 20
pll_inputclockX14, 20
pll_inputclockX15, 20
pll_inputclockX16, 20
pll_inputclockX2, 20
pll_inputclockX3, 20
pll_inputclockX4, 20
pll_inputclockX5, 20
pll_inputclockX6, 20
pll_inputclockX7, 20
pll_inputclockX8, 20
pll_inputclockX9, 20
pll_multiplication, 20
pll_Multiplication_factor, 18
Rcc_HSE, 20
Rcc_HSI, 20
Rcc_PLL, 20
RCC_SYS_CLK, 18
SYS_CLK, 20
RCC_CR
RCC_private.h, 32
RCC_cr
RCC_interface.h, 25

RCC_CSR
RCC_private.h, 32
Rcc_HSE
RCC_config.h, 20
Rcc_HSI
RCC_config.h, 20
RCC_intalizeHSE
RCC_interface.h, 26
RCC_program.c, 34
RCC_IntalizeHSI
RCC_interface.h, 27
RCC_program.c, 34
RCC_intalizePLL
RCC_interface.h, 27
RCC_program.c, 34
RCC_interface.h
ADCPRE0, 25
ADCPRE1, 25
CSSON, 26
divided2, 25
divided4, 25
divided6, 25
divided8, 25
HPRE0, 25
HPRE1, 25
HPRE2, 25
HPRE3, 25
HSEBYP, 26
HSEON, 26
HSERDY, 26
HSICAL0, 26
HSICAL1, 26
HSICAL2, 26
HSICAL3, 26
HSICAL4, 26
HSICAL5, 26
HSICAL6, 26
HSICAL7, 26
HSION, 26
HSIRDY, 26
HSITRIM0, 26
HSITRIM1, 26
HSITRIM2, 26
HSITRIM3, 26
HSITRIM4, 26
MCO0, 25
MCO1, 25
MCO2, 25
PLLMUL0, 25
PLLMUL1, 25
PLLMUL2, 25
PLLMUL3, 25
PLLON, 26
PLLRDY, 26
PLLSRC, 25
PLLXTPRE, 25
PPRE10, 25
PPRE11, 25
PPRE12, 25
PPRE20, 25
PPRE21, 25
PPRE22, 25
prescalers, 25
RCC_ADC_SETprescaler, 26
RCC_AHB, 23
RCC_AHB_CRC, 23
RCC_AHB_DMA1, 23
RCC_AHB_DMA2, 23
RCC_AHB_FLITF, 23
RCC_AHB_FSMC, 23
RCC_AHB_SDIO, 23
RCC_AHB_SRAM, 23
RCC_APB1, 23
RCC_APB1_BKP, 24
RCC_APB1_CAN, 24
RCC_APB1_DAC, 24
RCC_APB1_I2C1, 24
RCC_APB1_I2C2, 24
RCC_APB1_PWR, 24
RCC_APB1_SPI2, 24
RCC_APB1_SPI3, 24
RCC_APB1_TIM12, 24
RCC_APB1_TIM13, 24
RCC_APB1_TIM14, 24
RCC_APB1_TIM2, 24
RCC_APB1_TIM3, 24
RCC_APB1_TIM4, 24
RCC_APB1_TIM5, 24
RCC_APB1_TIM6, 24
RCC_APB1_TIM7, 24
RCC_APB1_USART2, 24
RCC_APB1_USART3, 24
RCC_APB1_USART4, 24
RCC_APB1_USART5, 24
RCC_APB1_USB, 24
RCC_APB1_WWWG, 24
RCC_APB2, 23
RCC_APB2_ADC1, 24
RCC_APB2_ADC2, 24
RCC_APB2_ADC3, 24
RCC_APB2_AFIO, 24
RCC_APB2_DIOA, 24
RCC_APB2_DIOB, 24
RCC_APB2_DIOC, 24
RCC_APB2_DIOD, 24
RCC_APB2_DIOE, 24
RCC_APB2_DIOF, 24
RCC_APB2_DIOG, 24
RCC_APB2_SPI1, 24
RCC_APB2_TIM1, 24
RCC_APB2_TIM10, 24
RCC_APB2_TIM11, 24
RCC_APB2_TIM8, 24
RCC_APB2_TIM9, 24
RCC_APB2_USART1, 24
RCC_cfgr, 25

RCC_cr, 25
RCC_intalizeHSE, 26
RCC_IntalizeHSI, 27
RCC_intalizePLL, 27
RCC_voidDisablePerClk, 27
RCC_voidEnablePerClk, 28
RCC_voidSetADCprescaler, 28
RCC_voidSetAHBprescaler, 28
RCC_voidSetAPB1prescaler, 28
RCC_voidSetAPB2prescaler, 28
RCC_voidSysClkInt, 29
SW0, 25
SW1, 25
SWS0, 25
SWS1, 25
USBPRE, 25
Rcc_PLL
RCC_config.h, 20
RCC_private.h
RCC_AHBENR, 31
RCC_APB1ENR, 31
RCC_APB1RSTR, 31
RCC_APB2ENR, 31
RCC_APB2RSTR, 31
RCC_BASE_ADD, 31
RCC_BDCR, 31
RCC_CFGR, 31
RCC_CIR, 32
RCC_CR, 32
RCC_CSR, 32
RCC_program.c
RCC_ADC_SETprescaler, 33
RCC_intalizeHSE, 34
RCC_IntalizeHSI, 34
RCC_intalizePLL, 34
RCC_voidDisablePerClk, 34
RCC_voidEnablePerClk, 35
RCC_voidSetADCprescaler, 35
RCC_voidSetAHBprescaler, 35
RCC_voidSetAPB1prescaler, 35
RCC_voidSetAPB2prescaler, 35
RCC_voidSysClkInt, 36
RCC_SYS_CLK
RCC_config.h, 18
RCC_voidDisablePerClk
RCC_interface.h, 27
RCC_program.c, 34
RCC_voidEnablePerClk
RCC_interface.h, 28
RCC_program.c, 35
RCC_voidSetADCprescaler
RCC_interface.h, 28
RCC_program.c, 35
RCC_voidSetAHBprescaler
RCC_interface.h, 28
RCC_program.c, 35
RCC_voidSetAPB1prescaler
RCC_interface.h, 28
RCC_program.c, 35
RCC_voidSetAPB2prescaler
RCC_interface.h, 28
RCC_program.c, 35
RCC_voidSysClkInt
RCC_interface.h, 29
RCC_program.c, 36
RE
USART_private.h, 77
RMC
main.c, 117
RTSE
USART_private.h, 78
RWU
USART_private.h, 78
RXNE
USART_private.h, 78
RXNEIE
USART_private.h, 78

s16
Std_Types.h, 14
s32
Std_Types.h, 15
s64
Std_Types.h, 15
s8
Std_Types.h, 15
SBK
USART_private.h, 78
SCB_AIRCR
NVIC_program.c, 60
SCB_BASE_ADD
NVIC_program.c, 60
SCEN
USART_private.h, 78
SET_BIT
Bit_Math.h, 10
SMS_CONC
GSM_Private.h, 103
SMS_INIT
GSM_Private.h, 104
GSM_Program.c, 111
SR
USART_type, 7
Src/main.c, 116
Src/syscalls.c, 118
Src/sysmem.c, 123
Std_Types.h
f32, 14
f64, 14
s16, 14
s32, 15
s64, 15
s8, 15
u16, 15
u32, 15
u64, 15
u8, 15

STOP
USART_private.h, 78
STOP_BIT_1
USART_private.h, 78
STOP_BIT_2
USART_private.h, 79
SW0
RCC_interface.h, 25
SW1
RCC_interface.h, 25
SWS0
RCC_interface.h, 25
SWS1
RCC_interface.h, 25
SYS_CLK
RCC_config.h, 20
syscalls.c
__attribute__, 119
__io_getchar, 119
__io_putchar, 119
_close, 120
_execve, 120
_exit, 120
_fork, 120
_fstat, 120
_getpid, 121
_isatty, 121
_kill, 121
_link, 121
_lseek, 121
_open, 121
_stat, 122
_times, 122
_unlink, 122
_wait, 122
environ, 122
initialise_monitor_handles, 122
sysmem.c
__sbrk_heap_end, 124
_sbrk, 124

TC
USART_private.h, 79
TCEIE
USART_private.h, 79
TCIE
USART_private.h, 79
TE
USART_private.h, 79
TOG_BIT
Bit_Math.h, 10
TXE
USART_private.h, 79

u16
Std_Types.h, 15
u32
Std_Types.h, 15
u64
Std_Types.h, 15
u8
Std_Types.h, 15
UART1_enableRX
USART_interface.h, 66
USART_program.c, 82
UART1_enableTX
USART_interface.h, 66
USART_program.c, 82
UART1_init
USART_interface.h, 66
USART_program.c, 82
UART1_RXdata
USART_interface.h, 66
USART_program.c, 82
UART1_StrReceiveString
USART_interface.h, 66
USART_program.c, 82
UART1_TXdata
USART_interface.h, 67
USART_program.c, 83
UART1_voidSendString
USART_interface.h, 67
USART_program.c, 83
UART2_enableRX
USART_interface.h, 68
USART_program.c, 84
UART2_enableTX
USART_interface.h, 68
USART_program.c, 84
UART2_init
USART_interface.h, 68
USART_program.c, 84
UART2_PTR
USART_program.c, 86
UART2_RX_CALLBACK
USART_interface.h, 68
USART_program.c, 84
UART2_RXdata
USART_interface.h, 69
USART_program.c, 85
UART2_StrReceiveString
USART_interface.h, 69
USART_program.c, 85
UART2_TXdata
USART_interface.h, 69
USART_program.c, 85
UART2_voidSendString
USART_interface.h, 70
USART_program.c, 86
UE
USART_private.h, 79
URL
GSM_Config.h, 98
URL_CONC
GSM_Private.h, 103
USART1
USART_private.h, 79

USART1_BAUDRATE
USART_config.h, 63
USART1_DATA_LENGTH
USART_config.h, 63
USART1_PARITY
USART_config.h, 63
USART1_PARITY_TYPE
USART_config.h, 63
USART1_STOP_LENGTH
USART_config.h, 63
USART2
USART_private.h, 80
USART2_BAUDRATE
USART_config.h, 63
USART2_DATA_LENGTH
USART_config.h, 64
USART2_IRQHandler
USART_interface.h, 70
USART_program.c, 86
USART2_PARITY
USART_config.h, 64
USART2_PARITY_TYPE
USART_config.h, 64
USART2_STOP_LENGTH
USART_config.h, 64
USART_config.h
USART1_BAUDRATE, 63
USART1_DATA_LENGTH, 63
USART1_PARITY, 63
USART1_PARITY_TYPE, 63
USART1_STOP_LENGTH, 63
USART2_BAUDRATE, 63
USART2_DATA_LENGTH, 64
USART2_PARITY, 64
USART2_PARITY_TYPE, 64
USART2_STOP_LENGTH, 64
USART_interface.h
UART1_enableRX, 66
UART1_enableTX, 66
UART1_init, 66
UART1_RXdata, 66
UART1_StrReceiveString, 66
UART1_TXdata, 67
UART1_voidSendString, 67
UART2_enableRX, 68
UART2_enableTX, 68
UART2_init, 68
UART2_RX_CALLBACK, 68
UART2_RXdata, 69
UART2_StrReceiveString, 69
UART2_TXdata, 69
UART2_voidSendString, 70
USART2_IRQHandler, 70
USART_private.h
ADD, 73
BAUD_RATE_115200, 73
BAUD_RATE_9600, 73
CLKEN, 73
CPHA, 73
CPOL, 73
CTS, 73
CTSE, 73
CTSIE, 74
DATA_LEN_8, 74
DATA_LEN_9, 74
DMAR, 74
DMAT, 74
EIE, 74
FE, 74
GT, 74
HDSEL, 75
IDLE, 75
IDLEIE, 75
IREN, 75
IRLP, 75
LBCL, 75
LBD, 75
LBDIE, 75
LBDL, 76
LINEN, 76
M, 76
NACK, 76
NE, 76
ORE, 76
PARITY_EVEN, 76
PARITY_NO, 76
PARITY_ODD, 77
PARITY_YES, 77
PCE, 77
PE, 77
PEIE, 77
PS, 77
PSC, 77
RE, 77
RTSE, 78
RWU, 78
RXNE, 78
RXNEIE, 78
SBK, 78
SCEN, 78
STOP, 78
STOP_BIT_1, 78
STOP_BIT_2, 79
TC, 79
TCEIE, 79
TCIE, 79
TE, 79
TXE, 79
UE, 79
USART1, 79
USART2, 80
WAKe, 80
USART_program.c
UART1_enableRX, 82
UART1_enableTX, 82
UART1_init, 82

UART1_RXdata, 82
UART1_StrReceiveString, 82
UART1_TXdata, 83
UART1_voidSendString, 83
UART2_enableRX, 84
UART2_enableTX, 84
UART2_init, 84
UART2_PTR, 86
UART2_RX_CALLBACK, 84
UART2_RXdata, 85
UART2_StrReceiveString, 85
UART2_TXdata, 85
UART2_voidSendString, 86
USART2_IRQHandler, 86
USART_type, 6
BRR, 7
CR1, 7
CR2, 7
CR3, 7
DR, 7
GTPR, 7
SR, 7
USBPRE
RCC_interface.h, 25
UTC_Time
GPS_RMC, 6

waitforDOWNLOAD
GSM_Private.h, 104
GSM_Program.c, 111
waitforHTTPACTION
GSM_Private.h, 105
GSM_Program.c, 112
waitforok
GSM_Private.h, 105
GSM_Program.c, 112
WAKe
USART_private.h, 80