

My Project

Generated by Doxygen 1.8.13

Contents

1	Hierarchical Index	1
1.1	Class Hierarchy	1
2	Class Index	3
2.1	Class List	3
3	File Index	5
3.1	File List	5
4	Class Documentation	7
4.1	Ant Class Reference	7
4.1.1	Constructor & Destructor Documentation	7
4.1.1.1	Ant()	7
4.1.2	Member Data Documentation	7
4.1.2.1	dir	8
4.1.2.2	x	8
4.1.2.3	y	8
4.2	AntThread Class Reference	8
4.2.1	Constructor & Destructor Documentation	8
4.2.1.1	AntThread()	9
4.2.1.2	~AntThread()	9
4.2.2	Member Function Documentation	9
4.2.2.1	finish()	9
4.2.2.2	isRunnung()	9
4.2.2.3	pause()	9

4.2.2.4	run()	9
4.2.2.5	step	10
4.3	AntWin Class Reference	10
4.3.1	Constructor & Destructor Documentation	10
4.3.1.1	AntWin()	11
4.3.1.2	~AntWin()	11
4.3.2	Member Function Documentation	11
4.3.2.1	closeEvent()	11
4.3.2.2	keyPressEvent()	11
4.3.2.3	paintEvent()	11
4.3.2.4	step	12
4.3.3	Member Data Documentation	12
4.3.3.1	antThread	12
4.4	qt_meta_stringdata_AntThread_t Struct Reference	12
4.4.1	Member Data Documentation	12
4.4.1.1	data	12
4.4.1.2	stringdata0	12
4.5	qt_meta_stringdata_AntWin_t Struct Reference	13
4.5.1	Member Data Documentation	13
4.5.1.1	data	13
4.5.1.2	stringdata0	13

5 File Documentation	15
5.1 ant.h File Reference	15
5.2 antthread.cpp File Reference	15
5.3 antthread.h File Reference	15
5.4 antwin.cpp File Reference	15
5.5 antwin.h File Reference	16
5.6 main.cpp File Reference	16
5.6.1 Function Documentation	16
5.6.1.1 main()	16
5.7 moc_antthread.cpp File Reference	16
5.7.1 Macro Definition Documentation	17
5.7.1.1 QT_MOC_LITERAL	17
5.8 moc_antwin.cpp File Reference	17
5.8.1 Macro Definition Documentation	17
5.8.1.1 QT_MOC_LITERAL	18
5.9 moc_predefs.h File Reference	18
5.9.1 Macro Definition Documentation	25
5.9.1.1 __amd64	25
5.9.1.2 __amd64__	25
5.9.1.3 __ATOMIC_ACQ_REL	25
5.9.1.4 __ATOMIC_ACQUIRE	26
5.9.1.5 __ATOMIC_CONSUME	26
5.9.1.6 __ATOMIC_HLE_ACQUIRE	26
5.9.1.7 __ATOMIC_HLE_RELEASE	26
5.9.1.8 __ATOMIC_RELAXED	26
5.9.1.9 __ATOMIC_RELEASE	26
5.9.1.10 __ATOMIC_SEQ_CST	26
5.9.1.11 __BIGGEST_ALIGNMENT__	26
5.9.1.12 __BYTE_ORDER__	27
5.9.1.13 __CHAR16_TYPE__	27

5.9.1.14	<code>__CHAR32_TYPE__</code>	27
5.9.1.15	<code>__CHAR_BIT__</code>	27
5.9.1.16	<code>__code_model_small__</code>	27
5.9.1.17	<code>__cplusplus</code>	27
5.9.1.18	<code>__cpp_aggregate_nsdmi</code>	27
5.9.1.19	<code>__cpp_alias_templates</code>	27
5.9.1.20	<code>__cpp_attributes</code>	28
5.9.1.21	<code>__cpp_binary_literals</code>	28
5.9.1.22	<code>__cpp_constexpr</code>	28
5.9.1.23	<code>__cpp_decltype</code>	28
5.9.1.24	<code>__cpp_decltype_auto</code>	28
5.9.1.25	<code>__cpp_delegating_constructors</code>	28
5.9.1.26	<code>__cpp_digit_separators</code>	28
5.9.1.27	<code>__cpp_exceptions</code>	28
5.9.1.28	<code>__cpp_generic_lambdas</code>	29
5.9.1.29	<code>__cpp_hex_float</code>	29
5.9.1.30	<code>__cpp_inheriting_constructors</code>	29
5.9.1.31	<code>__cpp_init_captures</code>	29
5.9.1.32	<code>__cpp_initializer_lists</code>	29
5.9.1.33	<code>__cpp_lambdas</code>	29
5.9.1.34	<code>__cpp_nsdmi</code>	29
5.9.1.35	<code>__cpp_range_based_for</code>	29
5.9.1.36	<code>__cpp_raw_strings</code>	30
5.9.1.37	<code>__cpp_ref_qualifiers</code>	30
5.9.1.38	<code>__cpp_return_type_deduction</code>	30
5.9.1.39	<code>__cpp_rtti</code>	30
5.9.1.40	<code>__cpp_runtime_arrays</code>	30
5.9.1.41	<code>__cpp_rvalue_reference</code>	30
5.9.1.42	<code>__cpp_rvalue_references</code>	30
5.9.1.43	<code>__cpp_sized_deallocation</code>	30

5.9.1.44	__cpp_static_assert	31
5.9.1.45	__cpp_threadsafe_static_init	31
5.9.1.46	__cpp_unicode_characters	31
5.9.1.47	__cpp_unicode_literals	31
5.9.1.48	__cpp_user_defined_literals	31
5.9.1.49	__cpp_variable_templates	31
5.9.1.50	__cpp_variadic_templates	31
5.9.1.51	__DBL_DECIMAL_DIG__	31
5.9.1.52	__DBL_DENORM_MIN__	32
5.9.1.53	__DBL_DIG__	32
5.9.1.54	__DBL_EPSILON__	32
5.9.1.55	__DBL_HAS_DENORM__	32
5.9.1.56	__DBL_HAS_INFINITY__	32
5.9.1.57	__DBL_HAS_QUIET_NAN__	32
5.9.1.58	__DBL_MANT_DIG__	32
5.9.1.59	__DBL_MAX_10_EXP__	32
5.9.1.60	__DBL_MAX__	33
5.9.1.61	__DBL_MAX_EXP__	33
5.9.1.62	__DBL_MIN_10_EXP__	33
5.9.1.63	__DBL_MIN__	33
5.9.1.64	__DBL_MIN_EXP__	33
5.9.1.65	__DEC128_EPSILON__	33
5.9.1.66	__DEC128_MANT_DIG__	33
5.9.1.67	__DEC128_MAX__	33
5.9.1.68	__DEC128_MAX_EXP__	34
5.9.1.69	__DEC128_MIN__	34
5.9.1.70	__DEC128_MIN_EXP__	34
5.9.1.71	__DEC128_SUBNORMAL_MIN__	34
5.9.1.72	__DEC32_EPSILON__	34
5.9.1.73	__DEC32_MANT_DIG__	34

5.9.1.74	__DEC32_MAX__	34
5.9.1.75	__DEC32_MAX_EXP__	34
5.9.1.76	__DEC32_MIN__	35
5.9.1.77	__DEC32_MIN_EXP__	35
5.9.1.78	__DEC32_SUBNORMAL_MIN__	35
5.9.1.79	__DEC64_EPSILON__	35
5.9.1.80	__DEC64_MANT_DIG__	35
5.9.1.81	__DEC64_MAX__	35
5.9.1.82	__DEC64_MAX_EXP__	35
5.9.1.83	__DEC64_MIN__	35
5.9.1.84	__DEC64_MIN_EXP__	36
5.9.1.85	__DEC64_SUBNORMAL_MIN__	36
5.9.1.86	__DEC_EVAL_METHOD__	36
5.9.1.87	__DECIMAL_BID_FORMAT__	36
5.9.1.88	__DECIMAL_DIG__	36
5.9.1.89	__DEPRECATED	36
5.9.1.90	__ELF__	36
5.9.1.91	__EXCEPTIONS	36
5.9.1.92	__FINITE_MATH_ONLY__	37
5.9.1.93	__FLOAT_WORD_ORDER__	37
5.9.1.94	__FLT128_DECIMAL_DIG__	37
5.9.1.95	__FLT128_DENORM_MIN__	37
5.9.1.96	__FLT128_DIG__	37
5.9.1.97	__FLT128_EPSILON__	37
5.9.1.98	__FLT128_HAS_DENORM__	37
5.9.1.99	__FLT128_HAS_INFINITY__	37
5.9.1.100	__FLT128_HAS_QUIET_NAN__	38
5.9.1.101	__FLT128_MANT_DIG__	38
5.9.1.102	__FLT128_MAX_10_EXP__	38
5.9.1.103	__FLT128_MAX__	38

5.9.1.104 __FLT128_MAX_EXP__	38
5.9.1.105 __FLT128_MIN_10_EXP__	38
5.9.1.106 __FLT128_MIN__	38
5.9.1.107 __FLT128_MIN_EXP__	38
5.9.1.108 __FLT32_DECIMAL_DIG__	39
5.9.1.109 __FLT32_DENORM_MIN__	39
5.9.1.110 __FLT32_DIG__	39
5.9.1.111 __FLT32_EPSILON__	39
5.9.1.112 __FLT32_HAS_DENORM__	39
5.9.1.113 __FLT32_HAS_INFINITY__	39
5.9.1.114 __FLT32_HAS_QUIET_NAN__	39
5.9.1.115 __FLT32_MANT_DIG__	39
5.9.1.116 __FLT32_MAX_10_EXP__	40
5.9.1.117 __FLT32_MAX__	40
5.9.1.118 __FLT32_MAX_EXP__	40
5.9.1.119 __FLT32_MIN_10_EXP__	40
5.9.1.120 __FLT32_MIN__	40
5.9.1.121 __FLT32_MIN_EXP__	40
5.9.1.122 __FLT32X_DECIMAL_DIG__	40
5.9.1.123 __FLT32X_DENORM_MIN__	40
5.9.1.124 __FLT32X_DIG__	41
5.9.1.125 __FLT32X_EPSILON__	41
5.9.1.126 __FLT32X_HAS_DENORM__	41
5.9.1.127 __FLT32X_HAS_INFINITY__	41
5.9.1.128 __FLT32X_HAS_QUIET_NAN__	41
5.9.1.129 __FLT32X_MANT_DIG__	41
5.9.1.130 __FLT32X_MAX_10_EXP__	41
5.9.1.131 __FLT32X_MAX__	41
5.9.1.132 __FLT32X_MAX_EXP__	42
5.9.1.133 __FLT32X_MIN_10_EXP__	42

5.9.1.134 __FLT32X_MIN__	42
5.9.1.135 __FLT32X_MIN_EXP__	42
5.9.1.136 __FLT64_DECIMAL_DIG__	42
5.9.1.137 __FLT64_DENORM_MIN__	42
5.9.1.138 __FLT64_DIG__	42
5.9.1.139 __FLT64_EPSILON__	42
5.9.1.140 __FLT64_HAS_DENORM__	43
5.9.1.141 __FLT64_HAS_INFINITY__	43
5.9.1.142 __FLT64_HAS_QUIET_NAN__	43
5.9.1.143 __FLT64_MANT_DIG__	43
5.9.1.144 __FLT64_MAX_10_EXP__	43
5.9.1.145 __FLT64_MAX__	43
5.9.1.146 __FLT64_MAX_EXP__	43
5.9.1.147 __FLT64_MIN_10_EXP__	43
5.9.1.148 __FLT64_MIN__	44
5.9.1.149 __FLT64_MIN_EXP__	44
5.9.1.150 __FLT64X_DECIMAL_DIG__	44
5.9.1.151 __FLT64X_DENORM_MIN__	44
5.9.1.152 __FLT64X_DIG__	44
5.9.1.153 __FLT64X_EPSILON__	44
5.9.1.154 __FLT64X_HAS_DENORM__	44
5.9.1.155 __FLT64X_HAS_INFINITY__	44
5.9.1.156 __FLT64X_HAS_QUIET_NAN__	45
5.9.1.157 __FLT64X_MANT_DIG__	45
5.9.1.158 __FLT64X_MAX_10_EXP__	45
5.9.1.159 __FLT64X_MAX__	45
5.9.1.160 __FLT64X_MAX_EXP__	45
5.9.1.161 __FLT64X_MIN_10_EXP__	45
5.9.1.162 __FLT64X_MIN__	45
5.9.1.163 __FLT64X_MIN_EXP__	45

5.9.1.164	__FLT_DECIMAL_DIG__	46
5.9.1.165	__FLT_DENORM_MIN__	46
5.9.1.166	__FLT_DIG__	46
5.9.1.167	__FLT_EPSILON__	46
5.9.1.168	__FLT_EVAL_METHOD__	46
5.9.1.169	__FLT_EVAL_METHOD_TS_18661_3__	46
5.9.1.170	__FLT_HAS_DENORM__	46
5.9.1.171	__FLT_HAS_INFINITY__	46
5.9.1.172	__FLT_HAS_QUIET_NAN__	47
5.9.1.173	__FLT_MANT_DIG__	47
5.9.1.174	__FLT_MAX_10_EXP__	47
5.9.1.175	__FLT_MAX__	47
5.9.1.176	__FLT_MAX_EXP__	47
5.9.1.177	__FLT_MIN_10_EXP__	47
5.9.1.178	__FLT_MIN__	47
5.9.1.179	__FLT_MIN_EXP__	47
5.9.1.180	__FLT_RADIX__	48
5.9.1.181	__FXSR__	48
5.9.1.182	__GCC_ASM_FLAG_OUTPUTS__	48
5.9.1.183	__GCC_ATOMIC_BOOL_LOCK_FREE	48
5.9.1.184	__GCC_ATOMIC_CHAR16_T_LOCK_FREE	48
5.9.1.185	__GCC_ATOMIC_CHAR32_T_LOCK_FREE	48
5.9.1.186	__GCC_ATOMIC_CHAR_LOCK_FREE	48
5.9.1.187	__GCC_ATOMIC_INT_LOCK_FREE	48
5.9.1.188	__GCC_ATOMIC_LLONG_LOCK_FREE	49
5.9.1.189	__GCC_ATOMIC_LONG_LOCK_FREE	49
5.9.1.190	__GCC_ATOMIC_POINTER_LOCK_FREE	49
5.9.1.191	__GCC_ATOMIC_SHORT_LOCK_FREE	49
5.9.1.192	__GCC_ATOMIC_TEST_AND_SET_TRUEVAL	49
5.9.1.193	__GCC_ATOMIC_WCHAR_T_LOCK_FREE	49

5.9.1.194	<code>__GCC_HAVE_DWARF2_CFI_ASM</code>	49
5.9.1.195	<code>__GCC_HAVE_SYNC_COMPARE_AND_SWAP_1</code>	49
5.9.1.196	<code>__GCC_HAVE_SYNC_COMPARE_AND_SWAP_2</code>	50
5.9.1.197	<code>__GCC_HAVE_SYNC_COMPARE_AND_SWAP_4</code>	50
5.9.1.198	<code>__GCC_HAVE_SYNC_COMPARE_AND_SWAP_8</code>	50
5.9.1.199	<code>__GCC_IEC_559</code>	50
5.9.1.200	<code>__GCC_IEC_559_COMPLEX</code>	50
5.9.1.201	<code>__GLIBCXX_BITSIZE_INT_N_0</code>	50
5.9.1.202	<code>__GLIBCXX_TYPE_INT_N_0</code>	50
5.9.1.203	<code>__gnu_linux__</code>	50
5.9.1.204	<code>__GNUC__</code>	51
5.9.1.205	<code>__GNUC_MINOR__</code>	51
5.9.1.206	<code>__GNUC_PATCHLEVEL__</code>	51
5.9.1.207	<code>__GNUC_STDC_INLINE__</code>	51
5.9.1.208	<code>__GNUG__</code>	51
5.9.1.209	<code>__GXX_ABI_VERSION</code>	51
5.9.1.210	<code>__GXX_EXPERIMENTAL_CXX0X__</code>	51
5.9.1.211	<code>__GXX_RTTI</code>	51
5.9.1.212	<code>__GXX_WEAK__</code>	52
5.9.1.213	<code>__has_include</code>	52
5.9.1.214	<code>__has_include_next</code>	52
5.9.1.215	<code>__INT16_C</code>	52
5.9.1.216	<code>__INT16_MAX__</code>	52
5.9.1.217	<code>__INT16_TYPE__</code>	52
5.9.1.218	<code>__INT32_C</code>	52
5.9.1.219	<code>__INT32_MAX__</code>	53
5.9.1.220	<code>__INT32_TYPE__</code>	53
5.9.1.221	<code>__INT64_C</code>	53
5.9.1.222	<code>__INT64_MAX__</code>	53
5.9.1.223	<code>__INT64_TYPE__</code>	53

5.9.1.224 __INT8_C	53
5.9.1.225 __INT8_MAX__	53
5.9.1.226 __INT8_TYPE__	54
5.9.1.227 __INT_FAST16_MAX__	54
5.9.1.228 __INT_FAST16_TYPE__	54
5.9.1.229 __INT_FAST16_WIDTH__	54
5.9.1.230 __INT_FAST32_MAX__	54
5.9.1.231 __INT_FAST32_TYPE__	54
5.9.1.232 __INT_FAST32_WIDTH__	54
5.9.1.233 __INT_FAST64_MAX__	54
5.9.1.234 __INT_FAST64_TYPE__	55
5.9.1.235 __INT_FAST64_WIDTH__	55
5.9.1.236 __INT_FAST8_MAX__	55
5.9.1.237 __INT_FAST8_TYPE__	55
5.9.1.238 __INT_FAST8_WIDTH__	55
5.9.1.239 __INT_LEAST16_MAX__	55
5.9.1.240 __INT_LEAST16_TYPE__	55
5.9.1.241 __INT_LEAST16_WIDTH__	55
5.9.1.242 __INT_LEAST32_MAX__	56
5.9.1.243 __INT_LEAST32_TYPE__	56
5.9.1.244 __INT_LEAST32_WIDTH__	56
5.9.1.245 __INT_LEAST64_MAX__	56
5.9.1.246 __INT_LEAST64_TYPE__	56
5.9.1.247 __INT_LEAST64_WIDTH__	56
5.9.1.248 __INT_LEAST8_MAX__	56
5.9.1.249 __INT_LEAST8_TYPE__	56
5.9.1.250 __INT_LEAST8_WIDTH__	57
5.9.1.251 __INT_MAX__	57
5.9.1.252 __INT_WIDTH__	57
5.9.1.253 __INTMAX_C	57

5.9.1.254 __INTMAX_MAX__	57
5.9.1.255 __INTMAX_TYPE__	57
5.9.1.256 __INTMAX_WIDTH__	57
5.9.1.257 __INTPTR_MAX__	58
5.9.1.258 __INTPTR_TYPE__	58
5.9.1.259 __INTPTR_WIDTH__	58
5.9.1.260 __k8	58
5.9.1.261 __k8__	58
5.9.1.262 __LDBL_DECIMAL_DIG__	58
5.9.1.263 __LDBL_DENORM_MIN__	58
5.9.1.264 __LDBL_DIG__	58
5.9.1.265 __LDBL_EPSILON__	59
5.9.1.266 __LDBL_HAS_DENORM__	59
5.9.1.267 __LDBL_HAS_INFINITY__	59
5.9.1.268 __LDBL_HAS_QUIET_NAN__	59
5.9.1.269 __LDBL_MANT_DIG__	59
5.9.1.270 __LDBL_MAX_10_EXP__	59
5.9.1.271 __LDBL_MAX__	59
5.9.1.272 __LDBL_MAX_EXP__	59
5.9.1.273 __LDBL_MIN_10_EXP__	60
5.9.1.274 __LDBL_MIN__	60
5.9.1.275 __LDBL_MIN_EXP__	60
5.9.1.276 __linux	60
5.9.1.277 __linux__	60
5.9.1.278 __LONG_LONG_MAX__	60
5.9.1.279 __LONG_LONG_WIDTH__	60
5.9.1.280 __LONG_MAX__	60
5.9.1.281 __LONG_WIDTH__	61
5.9.1.282 __LP64__	61
5.9.1.283 __MMX__	61

5.9.1.284	<code>__OPTIMIZE__</code>	61
5.9.1.285	<code>__ORDER_BIG_ENDIAN__</code>	61
5.9.1.286	<code>__ORDER_LITTLE_ENDIAN__</code>	61
5.9.1.287	<code>__ORDER_PDP_ENDIAN__</code>	61
5.9.1.288	<code>__pic__</code>	61
5.9.1.289	<code>__PIC__</code>	62
5.9.1.290	<code>__pie__</code>	62
5.9.1.291	<code>__PIE__</code>	62
5.9.1.292	<code>__PRAGMA_REDEFINE_EXTNAME</code>	62
5.9.1.293	<code>__PTRDIFF_MAX__</code>	62
5.9.1.294	<code>__PTRDIFF_TYPE__</code>	62
5.9.1.295	<code>__PTRDIFF_WIDTH__</code>	62
5.9.1.296	<code>__REGISTER_PREFIX__</code>	62
5.9.1.297	<code>__SCHAR_MAX__</code>	63
5.9.1.298	<code>__SCHAR_WIDTH__</code>	63
5.9.1.299	<code>__SEG_FS</code>	63
5.9.1.300	<code>__SEG_GS</code>	63
5.9.1.301	<code>__SHRT_MAX__</code>	63
5.9.1.302	<code>__SHRT_WIDTH__</code>	63
5.9.1.303	<code>__SIG_ATOMIC_MAX__</code>	63
5.9.1.304	<code>__SIG_ATOMIC_MIN__</code>	63
5.9.1.305	<code>__SIG_ATOMIC_TYPE__</code>	64
5.9.1.306	<code>__SIG_ATOMIC_WIDTH__</code>	64
5.9.1.307	<code>__SIZE_MAX__</code>	64
5.9.1.308	<code>__SIZE_TYPE__</code>	64
5.9.1.309	<code>__SIZE_WIDTH__</code>	64
5.9.1.310	<code>__SIZEOF_DOUBLE__</code>	64
5.9.1.311	<code>__SIZEOF_FLOAT128__</code>	64
5.9.1.312	<code>__SIZEOF_FLOAT80__</code>	64
5.9.1.313	<code>__SIZEOF_FLOAT__</code>	65

5.9.1.314	<code>__SIZEOF_INT128__</code>	65
5.9.1.315	<code>__SIZEOF_INT__</code>	65
5.9.1.316	<code>__SIZEOF_LONG__</code>	65
5.9.1.317	<code>__SIZEOF_LONG_DOUBLE__</code>	65
5.9.1.318	<code>__SIZEOF_LONG_LONG__</code>	65
5.9.1.319	<code>__SIZEOF_POINTER__</code>	65
5.9.1.320	<code>__SIZEOF_PTRDIFF_T__</code>	65
5.9.1.321	<code>__SIZEOF_SHORT__</code>	66
5.9.1.322	<code>__SIZEOF_SIZE_T__</code>	66
5.9.1.323	<code>__SIZEOF_WCHAR_T__</code>	66
5.9.1.324	<code>__SIZEOF_WINT_T__</code>	66
5.9.1.325	<code>__SSE2__</code>	66
5.9.1.326	<code>__SSE2_MATH__</code>	66
5.9.1.327	<code>__SSE__</code>	66
5.9.1.328	<code>__SSE_MATH__</code>	66
5.9.1.329	<code>__SSP_STRONG__</code>	67
5.9.1.330	<code>__STDC__</code>	67
5.9.1.331	<code>__STDC_HOSTED__</code>	67
5.9.1.332	<code>__STDC_IEC_559__</code>	67
5.9.1.333	<code>__STDC_IEC_559_COMPLEX__</code>	67
5.9.1.334	<code>__STDC_ISO_10646__</code>	67
5.9.1.335	<code>__STDC_NO_THREADS__</code>	67
5.9.1.336	<code>__STDC_UTF_16__</code>	67
5.9.1.337	<code>__STDC_UTF_32__</code>	68
5.9.1.338	<code>__UINT16_C</code>	68
5.9.1.339	<code>__UINT16_MAX__</code>	68
5.9.1.340	<code>__UINT16_TYPE__</code>	68
5.9.1.341	<code>__UINT32_C</code>	68
5.9.1.342	<code>__UINT32_MAX__</code>	68
5.9.1.343	<code>__UINT32_TYPE__</code>	68

5.9.1.344	<code>__UINT64_C</code>	69
5.9.1.345	<code>__UINT64_MAX</code>	69
5.9.1.346	<code>__UINT64_TYPE</code>	69
5.9.1.347	<code>__UINT8_C</code>	69
5.9.1.348	<code>__UINT8_MAX</code>	69
5.9.1.349	<code>__UINT8_TYPE</code>	69
5.9.1.350	<code>__UINT_FAST16_MAX</code>	69
5.9.1.351	<code>__UINT_FAST16_TYPE</code>	70
5.9.1.352	<code>__UINT_FAST32_MAX</code>	70
5.9.1.353	<code>__UINT_FAST32_TYPE</code>	70
5.9.1.354	<code>__UINT_FAST64_MAX</code>	70
5.9.1.355	<code>__UINT_FAST64_TYPE</code>	70
5.9.1.356	<code>__UINT_FAST8_MAX</code>	70
5.9.1.357	<code>__UINT_FAST8_TYPE</code>	70
5.9.1.358	<code>__UINT_LEAST16_MAX</code>	70
5.9.1.359	<code>__UINT_LEAST16_TYPE</code>	71
5.9.1.360	<code>__UINT_LEAST32_MAX</code>	71
5.9.1.361	<code>__UINT_LEAST32_TYPE</code>	71
5.9.1.362	<code>__UINT_LEAST64_MAX</code>	71
5.9.1.363	<code>__UINT_LEAST64_TYPE</code>	71
5.9.1.364	<code>__UINT_LEAST8_MAX</code>	71
5.9.1.365	<code>__UINT_LEAST8_TYPE</code>	71
5.9.1.366	<code>__UINTMAX_C</code>	71
5.9.1.367	<code>__UINTMAX_MAX</code>	72
5.9.1.368	<code>__UINTMAX_TYPE</code>	72
5.9.1.369	<code>__UINTPTR_MAX</code>	72
5.9.1.370	<code>__UINTPTR_TYPE</code>	72
5.9.1.371	<code>__unix</code>	72
5.9.1.372	<code>__unix</code>	72
5.9.1.373	<code>__USER_LABEL_PREFIX</code>	72

5.9.1.374 __VERSION__	72
5.9.1.375 __WCHAR_MAX__	73
5.9.1.376 __WCHAR_MIN__	73
5.9.1.377 __WCHAR_TYPE__	73
5.9.1.378 __WCHAR_WIDTH__	73
5.9.1.379 __WINT_MAX__	73
5.9.1.380 __WINT_MIN__	73
5.9.1.381 __WINT_TYPE__	73
5.9.1.382 __WINT_WIDTH__	73
5.9.1.383 __x86_64	74
5.9.1.384 __x86_64__	74
5.9.1.385 _FORTIFY_SOURCE	74
5.9.1.386 _GNU_SOURCE	74
5.9.1.387 _LP64	74
5.9.1.388 _STDC_PREDEF_H	74
5.9.1.389 linux	74
5.9.1.390 unix	74

Index**75**

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Ant	7
QMainWindow	
AntWin	10
qt_meta_stringdata_AntThread_t	12
qt_meta_stringdata_AntWin_t	13
QThread	
AntThread	8

Chapter 2

Class Index

2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Ant	7
AntThread	8
AntWin	10
qt_meta_stringdata_AntThread_t	12
qt_meta_stringdata_AntWin_t	13

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

ant.h	15
antthread.cpp	15
antthread.h	15
antwin.cpp	15
antwin.h	16
main.cpp	16
moc_antthread.cpp	16
moc_antwin.cpp	17
moc_predefs.h	18

Chapter 4

Class Documentation

4.1 Ant Class Reference

```
#include <ant.h>
```

Collaboration diagram for Ant:

Public Member Functions

- [Ant](#) (int [x](#), int [y](#))

Public Attributes

- int [x](#)
- int [y](#)
- int [dir](#)

4.1.1 Constructor & Destructor Documentation

4.1.1.1 Ant()

```
Ant::Ant (
    int x,
    int y ) [inline]
```

4.1.2 Member Data Documentation

4.1.2.1 dir

```
int Ant::dir
```

4.1.2.2 x

```
int Ant::x
```

4.1.2.3 y

```
int Ant::y
```

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

- [ant.h](#)

4.2 AntThread Class Reference

```
#include <antthread.h>
```

Inheritance diagram for AntThread:

Collaboration diagram for AntThread:

Signals

- void [step](#) (const int &)

Public Member Functions

- [AntThread](#) ([Ants](#) *ants, int ***grids, int width, int height, int delay, int numAnts, int pheromone, int nbr↔ Pheromone, int evaporation, int min, int max, int cellAntMax)
- [~AntThread](#) ()
- void [run](#) ()
- void [finish](#) ()
- void [pause](#) ()
- bool [isRunnung](#) ()

4.2.1 Constructor & Destructor Documentation

4.2.1.1 AntThread()

```
AntThread::AntThread (
    Ants * ants,
    int *** grids,
    int width,
    int height,
    int delay,
    int numAnts,
    int pheromone,
    int nbrPheromone,
    int evaporation,
    int min,
    int max,
    int cellAntMax )
```

4.2.1.2 ~AntThread()

```
AntThread::~AntThread ( )
```

4.2.2 Member Function Documentation

4.2.2.1 finish()

```
void AntThread::finish ( ) [inline]
```

4.2.2.2 isRunnung()

```
bool AntThread::isRunnung ( ) [inline]
```

4.2.2.3 pause()

```
void AntThread::pause ( ) [inline]
```

4.2.2.4 run()

```
void AntThread::run ( )
```

4.2.2.5 step

```
void AntThread::step (  
    const int & _t1 ) [signal]
```

The documentation for this class was generated from the following files:

- [antthread.h](#)
- [antthread.cpp](#)
- [moc_antthread.cpp](#)

4.3 AntWin Class Reference

```
#include <antwin.h>
```

Inheritance diagram for AntWin:

Collaboration diagram for AntWin:

Public Slots

- void [step](#) (const int &)

Public Member Functions

- [AntWin](#) (int width=100, int height=75, int delay=120, int numAnts=100, int pheromone=10, int nbh↔ Pheromon=3, int evaporation=2, int cellDef=1, int min=2, int max=50, int cellAntMax=4, QWidget *parent=0)
- void [closeEvent](#) (QCloseEvent *event)
- void [keyPressEvent](#) (QKeyEvent *event)
- virtual [~AntWin](#) ()
- void [paintEvent](#) (QPaintEvent *)

Public Attributes

- [AntThread](#) * [antThread](#)

4.3.1 Constructor & Destructor Documentation

4.3.1.1 AntWin()

```
AntWin::AntWin (
    int width = 100,
    int height = 75,
    int delay = 120,
    int numAnts = 100,
    int pheromone = 10,
    int nbhPheromon = 3,
    int evaporation = 2,
    int cellDef = 1,
    int min = 2,
    int max = 50,
    int cellAntMax = 4,
    QWidget * parent = 0 )
```

4.3.1.2 ~AntWin()

```
AntWin::~AntWin ( ) [virtual]
```

4.3.2 Member Function Documentation

4.3.2.1 closeEvent()

```
void AntWin::closeEvent (
    QCloseEvent * event ) [inline]
```

4.3.2.2 keyPressEvent()

```
void AntWin::keyPressEvent (
    QKeyEvent * event ) [inline]
```

4.3.2.3 paintEvent()

```
void AntWin::paintEvent (
    QPaintEvent * )
```

4.3.2.4 step

```
void AntWin::step (
    const int & gridIdx ) [slot]
```

4.3.3 Member Data Documentation

4.3.3.1 antThread

```
AntThread\* AntWin::antThread
```

The documentation for this class was generated from the following files:

- [antwin.h](#)
- [antwin.cpp](#)

4.4 qt_meta_stringdata_AntThread_t Struct Reference

Collaboration diagram for qt_meta_stringdata_AntThread_t:

Public Attributes

- QByteArrayData [data](#) [3]
- char [stringdata0](#) [16]

4.4.1 Member Data Documentation

4.4.1.1 data

```
QByteArrayData qt_meta_stringdata_AntThread_t::data[3]
```

4.4.1.2 stringdata0

```
char qt_meta_stringdata_AntThread_t::stringdata0[16]
```

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

- [moc_antthread.cpp](#)

4.5 qt_meta_stringdata_AntWin_t Struct Reference

Collaboration diagram for qt_meta_stringdata_AntWin_t:

Public Attributes

- QByteArrayData [data](#) [3]
- char [stringdata0](#) [13]

4.5.1 Member Data Documentation

4.5.1.1 data

```
QByteArrayData qt_meta_stringdata_AntWin_t::data[3]
```

4.5.1.2 stringdata0

```
char qt_meta_stringdata_AntWin_t::stringdata0[13]
```

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

- [moc_antwin.cpp](#)

Chapter 5

File Documentation

5.1 ant.h File Reference

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

5.2 antthread.cpp File Reference

```
#include "antthread.h"  
#include <QDebug>  
#include <cmath>  
#include <QDateTime>  
Include dependency graph for antthread.cpp:
```

5.3 antthread.h File Reference

```
#include <QThread>  
#include "ant.h"  
Include dependency graph for antthread.h: This graph shows which files directly or indirectly include this file:
```

Classes

- class [AntThread](#)

5.4 antwin.cpp File Reference

```
#include "antwin.h"  
#include <QDebug>  
Include dependency graph for antwin.cpp:
```

5.5 antwin.h File Reference

```
#include <QMainWindow>
#include <QPainter>
#include <QString>
#include <QCloseEvent>
#include "antthread.h"
#include "ant.h"
```

Include dependency graph for antwin.h: This graph shows which files directly or indirectly include this file:

Classes

- class [AntWin](#)

5.6 main.cpp File Reference

```
#include <QApplication>
#include <QDesktopWidget>
#include <QDebug>
#include <QDateTime>
#include <QCommandLineOption>
#include <QCommandLineParser>
#include "antwin.h"
```

Include dependency graph for main.cpp:

Functions

- int [main](#) (int argc, char *argv[])

5.6.1 Function Documentation

5.6.1.1 main()

```
int main (
    int argc,
    char * argv[ ] )
```

5.7 moc_antthread.cpp File Reference

```
#include <memory>
#include "antthread.h"
#include <QtCore/qbytearray.h>
#include <QtCore/qmetatype.h>
```

Include dependency graph for moc_antthread.cpp:

Classes

- struct [qt_meta_stringdata_AntThread_t](#)

Macros

- #define [QT_MOC_LITERAL](#)(idx, ofs, len)

5.7.1 Macro Definition Documentation

5.7.1.1 QT_MOC_LITERAL

```
#define QT_MOC_LITERAL(  
    idx,  
    ofs,  
    len )
```

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \  
    qptrdiff(offsetof(qt\_meta\_stringdata\_AntThread\_t, stringdata0) + ofs \  
        - idx * sizeof(QByteArrayData)) \  
    )
```

5.8 moc_antwin.cpp File Reference

```
#include <memory>  
#include "antwin.h"  
#include <QtCore/qbytearray.h>  
#include <QtCore/qmetatype.h>  
Include dependency graph for moc_antwin.cpp:
```

Classes

- struct [qt_meta_stringdata_AntWin_t](#)

Macros

- #define [QT_MOC_LITERAL](#)(idx, ofs, len)

5.8.1 Macro Definition Documentation

5.8.1.1 QT_MOC_LITERAL

```
#define QT_MOC_LITERAL(  
    idx,  
    ofs,  
    len )
```

Value:

```
Q_STATIC_BYTE_ARRAY_DATA_HEADER_INITIALIZER_WITH_OFFSET(len, \  
    qptrdiff(offsetof(qt_meta_stringdata_AntWin_t, stringdata0) + ofs \  
        - idx * sizeof(QByteArrayData)) \  
    )
```

5.9 moc_predefs.h File Reference

Macros

- `#define __SSP_STRONG__ 3`
- `#define __DBL_MIN_EXP__ (-1021)`
- `#define __FLT32X_MAX_EXP__ 1024`
- `#define __cpp_attributes 200809`
- `#define __UINT_LEAST16_MAX__ 0xffff`
- `#define __ATOMIC_ACQUIRE 2`
- `#define __FLT128_MAX_10_EXP__ 4932`
- `#define __FLT_MIN__ 1.17549435082228750796873653722224568e-38F`
- `#define __GCC_IEC_559_COMPLEX 2`
- `#define __cpp_aggregate_nsdmi 201304`
- `#define __UINT_LEAST8_TYPE__ unsigned char`
- `#define __SIZEOF_FLOAT80__ 16`
- `#define __INTMAX_C(c) c ## L`
- `#define __CHAR_BIT__ 8`
- `#define __UINT8_MAX__ 0xff`
- `#define __WINT_MAX__ 0xffffffffU`
- `#define __FLT32_MIN_EXP__ (-125)`
- `#define __cpp_static_assert 200410`
- `#define __ORDER_LITTLE_ENDIAN__ 1234`
- `#define __SIZE_MAX__ 0xffffffffffffffffUL`
- `#define __WCHAR_MAX__ 0x7fffffff`
- `#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_1 1`
- `#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_2 1`
- `#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_4 1`
- `#define __DBL_DENORM_MIN__ double(4.94065645841246544176568792868221372e-324L)`
- `#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_8 1`
- `#define __GCC_ATOMIC_CHAR_LOCK_FREE 2`
- `#define __GCC_IEC_559 2`
- `#define __FLT32X_DECIMAL_DIG__ 17`
- `#define __FLT_EVAL_METHOD__ 0`
- `#define __unix__ 1`
- `#define __cpp_binary_literals 201304`
- `#define __FLT64_DECIMAL_DIG__ 17`
- `#define __GCC_ATOMIC_CHAR32_T_LOCK_FREE 2`
- `#define __x86_64 1`

- [#define __cpp_variadic_templates](#) 200704
- [#define __UINT_FAST64_MAX__](#) 0xffffffffffffffffUL
- [#define __SIG_ATOMIC_TYPE__](#) int
- [#define __DBL_MIN_10_EXP__](#) (-307)
- [#define __FINITE_MATH_ONLY__](#) 0
- [#define __cpp_variable_templates](#) 201304
- [#define __GNUC_PATCHLEVEL__](#) 0
- [#define __FLT32_HAS_DENORM__](#) 1
- [#define __UINT_FAST8_MAX__](#) 0xff
- [#define __has_include\(STR\)](#) __has_include__(STR)
- [#define __DEC64_MAX_EXP__](#) 385
- [#define __INT8_C\(c\)](#) c
- [#define __INT_LEAST8_WIDTH__](#) 8
- [#define __UINT_LEAST64_MAX__](#) 0xffffffffffffffffUL
- [#define __SHRT_MAX__](#) 0x7fff
- [#define __LDBL_MAX__](#) 1.18973149535723176502126385303097021e+4932L
- [#define __FLT64X_MAX_10_EXP__](#) 4932
- [#define __UINT_LEAST8_MAX__](#) 0xff
- [#define __GCC_ATOMIC_BOOL_LOCK_FREE](#) 2
- [#define __FLT128_DENORM_MIN__](#) 6.47517511943802511092443895822764655e-4966F128
- [#define __UINTMAX_TYPE__](#) long unsigned int
- [#define __linux](#) 1
- [#define __DEC32_EPSILON__](#) 1E-6DF
- [#define __FLT_EVAL_METHOD_TS_18661_3__](#) 0
- [#define __OPTIMIZE__](#) 1
- [#define __unix](#) 1
- [#define __UINT32_MAX__](#) 0xffffffffU
- [#define __GXX_EXPERIMENTAL_CXX0X__](#) 1
- [#define __LDBL_MAX_EXP__](#) 16384
- [#define __FLT128_MIN_EXP__](#) (-16381)
- [#define __WINT_MIN__](#) 0U
- [#define __linux__](#) 1
- [#define __FLT128_MIN_10_EXP__](#) (-4931)
- [#define __INT_LEAST16_WIDTH__](#) 16
- [#define __SCHAR_MAX__](#) 0x7f
- [#define __FLT128_MANT_DIG__](#) 113
- [#define __WCHAR_MIN__](#) (-__WCHAR_MAX__ - 1)
- [#define __INT64_C\(c\)](#) c ## L
- [#define __DBL_DIG__](#) 15
- [#define __GCC_ATOMIC_POINTER_LOCK_FREE](#) 2
- [#define __FLT64X_MANT_DIG__](#) 64
- [#define __FORTIFY_SOURCE](#) 2
- [#define __SIZEOF_INT__](#) 4
- [#define __SIZEOF_POINTER__](#) 8
- [#define __GCC_ATOMIC_CHAR16_T_LOCK_FREE](#) 2
- [#define __USER_LABEL_PREFIX__](#)
- [#define __FLT64X_EPSILON__](#) 1.08420217248550443400745280086994171e-19F64x
- [#define __STDC_HOSTED__](#) 1
- [#define __LDBL_HAS_INFINITY__](#) 1
- [#define __FLT32_DIG__](#) 6
- [#define __FLT_EPSILON__](#) 1.19209289550781250000000000000000000e-7F
- [#define __GXX_WEAK__](#) 1
- [#define __SHRT_WIDTH__](#) 16
- [#define __LDBL_MIN__](#) 3.36210314311209350626267781732175260e-4932L
- [#define __DEC32_MAX__](#) 9.999999E96DF

- `#define __STDC_NO_THREADS__ 1`
- `#define __ATOMIC_HLE_ACQUIRE 65536`
- `#define __FLT32_HAS_QUIET_NAN__ 1`
- `#define __GNU__ 7`
- `#define __LONG_LONG_MAX__ 0x7fffffffffffffffLL`
- `#define __SIZEOF_SIZE_T__ 8`
- `#define __cpp_rvalue_reference 200610`
- `#define __cpp_nsdmi 200809`
- `#define __FLT64X_MIN_EXP__ (-16381)`
- `#define __SIZEOF_WINT_T__ 4`
- `#define __LONG_LONG_WIDTH__ 64`
- `#define __cpp_initializer_lists 200806`
- `#define __FLT32_MAX_EXP__ 128`
- `#define __cpp_hex_float 201603`
- `#define __GCC_HAVE_DWARF2_CFI_ASM 1`
- `#define __GXX_ABI_VERSION 1011`
- `#define __FLT128_HAS_INFINITY__ 1`
- `#define __FLT_MIN_EXP__ (-125)`
- `#define __cpp_lambdas 200907`
- `#define __FLT64X_HAS_QUIET_NAN__ 1`
- `#define __INT_FAST64_TYPE__ long int`
- `#define __FLT64_DENORM_MIN__ 4.94065645841246544176568792868221372e-324F64`
- `#define __DBL_MIN__ double(2.22507385850720138309023271733240406e-308L)`
- `#define __PIE__ 2`
- `#define __LP64__ 1`
- `#define __FLT32X_EPSILON__ 2.22044604925031308084726333618164062e-16F32x`
- `#define __DECIMAL_BID_FORMAT__ 1`
- `#define __FLT64_MIN_10_EXP__ (-307)`
- `#define __FLT64X_DECIMAL_DIG__ 21`
- `#define __DEC128_MIN__ 1E-6143DL`
- `#define __REGISTER_PREFIX__`
- `#define __UINT16_MAX__ 0xffff`
- `#define __DBL_HAS_DENORM__ 1`
- `#define __FLT32_MIN__ 1.17549435082228750796873653722224568e-38F32`
- `#define __UINT8_TYPE__ unsigned char`
- `#define __FLT_MANT_DIG__ 24`
- `#define __LDBL_DECIMAL_DIG__ 21`
- `#define __VERSION__ "7.5.0"`
- `#define __UINT64_C(c) c ## UL`
- `#define __cpp_unicode_characters 200704`
- `#define __STDC_PREDEF_H 1`
- `#define __cpp_decltype_auto 201304`
- `#define __GCC_ATOMIC_INT_LOCK_FREE 2`
- `#define __FLT128_MAX_EXP__ 16384`
- `#define __FLT32_MANT_DIG__ 24`
- `#define __FLOAT_WORD_ORDER__ __ORDER_LITTLE_ENDIAN__`
- `#define __STDC_IEC_559_COMPLEX__ 1`
- `#define __FLT128_HAS_DENORM__ 1`
- `#define __FLT128_DIG__ 33`
- `#define __SCHAR_WIDTH__ 8`
- `#define __INT32_C(c) c`
- `#define __DEC64_EPSILON__ 1E-15DD`
- `#define __ORDER_PDP_ENDIAN__ 3412`
- `#define __DEC128_MIN_EXP__ (-6142)`
- `#define __FLT32_MAX_10_EXP__ 38`

- `#define __DEC32_MAX_EXP__ 97`
- `#define __INT_FAST8_MAX__ 0x7f`
- `#define __FLT128_MAX__ 1.18973149535723176508575932662800702e+4932F128`
- `#define __INTPTR_MAX__ 0x7fffffffffffffffL`
- `#define linux 1`
- `#define __cpp_range_based_for 200907`
- `#define __FLT64_HAS_QUIET_NAN__ 1`
- `#define __FLT32_MIN_10_EXP__ (-37)`
- `#define __SSE2__ 1`
- `#define __EXCEPTIONS 1`
- `#define __LDBL_MANT_DIG__ 64`
- `#define __DBL_HAS_QUIET_NAN__ 1`
- `#define __FLT64_HAS_INFINITY__ 1`
- `#define __FLT64X_MAX__ 1.18973149535723176502126385303097021e+4932F64x`
- `#define __SIG_ATOMIC_MIN__ (-__SIG_ATOMIC_MAX__ - 1)`
- `#define __code_model_small__ 1`
- `#define __cpp_return_type_deduction 201304`
- `#define __k8__ 1`
- `#define __INTPTR_TYPE__ long int`
- `#define __UINT16_TYPE__ short unsigned int`
- `#define __WCHAR_TYPE__ int`
- `#define __SIZEOF_FLOAT__ 4`
- `#define __pic__ 2`
- `#define __UINTPTR_MAX__ 0xffffffffffffffffUL`
- `#define __INT_FAST64_WIDTH__ 64`
- `#define __DEC64_MIN_EXP__ (-382)`
- `#define __cpp_decltype 200707`
- `#define __FLT32_DECIMAL_DIG__ 9`
- `#define __INT_FAST64_MAX__ 0x7fffffffffffffffL`
- `#define __GCC_ATOMIC_TEST_AND_SET_TRUEVAL 1`
- `#define __FLT_DIG__ 6`
- `#define __FLT64X_MAX_EXP__ 16384`
- `#define __UINT_FAST64_TYPE__ long unsigned int`
- `#define __INT_MAX__ 0x7fffffff`
- `#define __amd64__ 1`
- `#define __INT64_TYPE__ long int`
- `#define __FLT_MAX_EXP__ 128`
- `#define __ORDER_BIG_ENDIAN__ 4321`
- `#define __DBL_MANT_DIG__ 53`
- `#define __cpp_inheriting_constructors 201511`
- `#define __SIZEOF_FLOAT128__ 16`
- `#define __INT_LEAST64_MAX__ 0x7fffffffffffffffL`
- `#define __DEC64_MIN__ 1E-383DD`
- `#define __WINT_TYPE__ unsigned int`
- `#define __UINT_LEAST32_TYPE__ unsigned int`
- `#define __SIZEOF_SHORT__ 2`
- `#define __SSE__ 1`
- `#define __LDBL_MIN_EXP__ (-16381)`
- `#define __FLT64_MAX__ 1.79769313486231570814527423731704357e+308F64`
- `#define __WINT_WIDTH__ 32`
- `#define __INT_LEAST8_MAX__ 0x7f`
- `#define __FLT32X_MAX_10_EXP__ 308`
- `#define __SIZEOF_INT128__ 16`
- `#define __LDBL_MAX_10_EXP__ 4932`
- `#define __ATOMIC_RELAXED 0`

- #define `__DBL_EPSILON__` double(2.22044604925031308084726333618164062e-16L)
- #define `__FLT128_MIN__` 3.36210314311209350626267781732175260e-4932F128
- #define `__LP64` 1
- #define `__UINT8_C(c)` c
- #define `__FLT64_MAX_EXP__` 1024
- #define `__INT_LEAST32_TYPE__` int
- #define `__SIZEOF_WCHAR_T__` 4
- #define `__FLT128_HAS_QUIET_NAN__` 1
- #define `__INT_FAST8_TYPE__` signed char
- #define `__FLT64X_MIN__` 3.36210314311209350626267781732175260e-4932F64x
- #define `__GNUC_STDC_INLINE__` 1
- #define `__FLT64_HAS_DENORM__` 1
- #define `__FLT32_EPSILON__` 1.19209289550781250000000000000000000000e-7F32
- #define `__DBL_DECIMAL_DIG__` 17
- #define `__STDC_UTF_32__` 1
- #define `__INT_FAST8_WIDTH__` 8
- #define `__FXSR__` 1
- #define `__DEC_EVAL_METHOD__` 2
- #define `__FLT32X_MAX__` 1.79769313486231570814527423731704357e+308F32x
- #define `__cpp_runtime_arrays` 198712
- #define `__UINT64_TYPE__` long unsigned int
- #define `__UINT32_C(c)` c ## U
- #define `__INTMAX_MAX__` 0x7fffffffffffffffL
- #define `__cpp_alias_templates` 200704
- #define `__BYTE_ORDER__` `__ORDER_LITTLE_ENDIAN__`
- #define `__FLT_DENORM_MIN__` 1.40129846432481707092372958328991613e-45F
- #define `__INT8_MAX__` 0x7f
- #define `__LONG_WIDTH__` 64
- #define `__PIC__` 2
- #define `__UINT_FAST32_TYPE__` long unsigned int
- #define `__CHAR32_TYPE__` unsigned int
- #define `__FLT_MAX__` 3.40282346638528859811704183484516925e+38F
- #define `__cpp_constexpr` 201304
- #define `__INT32_TYPE__` int
- #define `__SIZEOF_DOUBLE__` 8
- #define `__cpp_exceptions` 199711
- #define `__FLT_MIN_10_EXP__` (-37)
- #define `__FLT64_MIN__` 2.22507385850720138309023271733240406e-308F64
- #define `__INT_LEAST32_WIDTH__` 32
- #define `__INTMAX_TYPE__` long int
- #define `__DEC128_MAX_EXP__` 6145
- #define `__FLT32X_HAS_QUIET_NAN__` 1
- #define `__ATOMIC_CONSUME` 1
- #define `__GNUC_MINOR__` 5
- #define `__GLIBCXX_TYPE_INT_N_0__` int128
- #define `__INT_FAST16_WIDTH__` 64
- #define `__UINTMAX_MAX__` 0xffffffffffffffffUL
- #define `__DEC32_MANT_DIG__` 7
- #define `__FLT32X_DENORM_MIN__` 4.94065645841246544176568792868221372e-324F32x
- #define `__DBL_MAX_10_EXP__` 308
- #define `__LDBL_DENORM_MIN__` 3.64519953188247460252840593361941982e-4951L
- #define `__INT16_C(c)` c
- #define `__cpp_generic_lambdas` 201304
- #define `__STDC__` 1
- #define `__FLT32X_DIG__` 15

- `#define __PTRDIFF_TYPE__` long int
- `#define __ATOMIC_SEQ_CST` 5
- `#define __UINT32_TYPE__` unsigned int
- `#define __FLT32X_MIN_10_EXP__` (-307)
- `#define __UINTPTR_TYPE__` long unsigned int
- `#define __DEC64_SUBNORMAL_MIN__` 0.0000000000000001E-383DD
- `#define __DEC128_MANT_DIG__` 34
- `#define __LDBL_MIN_10_EXP__` (-4931)
- `#define __FLT128_EPSILON__` 1.92592994438723585305597794258492732e-34F128
- `#define __SSE_MATH__` 1
- `#define __SIZEOF_LONG_LONG__` 8
- `#define __cpp_user_defined_literals` 200809
- `#define __FLT128_DECIMAL_DIG__` 36
- `#define __GCC_ATOMIC_LLONG_LOCK_FREE` 2
- `#define __FLT32X_MIN__` 2.22507385850720138309023271733240406e-308F32x
- `#define __LDBL_DIG__` 18
- `#define __FLT_DECIMAL_DIG__` 9
- `#define __UINT_FAST16_MAX__` 0xffffffffffffUL
- `#define __GCC_ATOMIC_SHORT_LOCK_FREE` 2
- `#define __INT_LEAST64_WIDTH__` 64
- `#define __UINT_FAST8_TYPE__` unsigned char
- `#define __GNU_SOURCE` 1
- `#define __cpp_init_captures` 201304
- `#define __ATOMIC_ACQ_REL` 4
- `#define __ATOMIC_RELEASE` 3

5.9.1 Macro Definition Documentation

5.9.1.1 `__amd64`

```
#define __amd64 1
```

5.9.1.2 `__amd64__`

```
#define __amd64__ 1
```

5.9.1.3 `__ATOMIC_ACQ_REL`

```
#define __ATOMIC_ACQ_REL 4
```

5.9.1.4 `__ATOMIC_ACQUIRE`

```
#define __ATOMIC_ACQUIRE 2
```

5.9.1.5 `__ATOMIC_CONSUME`

```
#define __ATOMIC_CONSUME 1
```

5.9.1.6 `__ATOMIC_HLE_ACQUIRE`

```
#define __ATOMIC_HLE_ACQUIRE 65536
```

5.9.1.7 `__ATOMIC_HLE_RELEASE`

```
#define __ATOMIC_HLE_RELEASE 131072
```

5.9.1.8 `__ATOMIC_RELAXED`

```
#define __ATOMIC_RELAXED 0
```

5.9.1.9 `__ATOMIC_RELEASE`

```
#define __ATOMIC_RELEASE 3
```

5.9.1.10 `__ATOMIC_SEQ_CST`

```
#define __ATOMIC_SEQ_CST 5
```

5.9.1.11 `__BIGGEST_ALIGNMENT__`

```
#define __BIGGEST_ALIGNMENT__ 16
```

5.9.1.12 `__BYTE_ORDER__`

```
#define __BYTE_ORDER__ \_\_ORDER\_LITTLE\_ENDIAN\_\_
```

5.9.1.13 `__CHAR16_TYPE__`

```
#define __CHAR16_TYPE__ short unsigned int
```

5.9.1.14 `__CHAR32_TYPE__`

```
#define __CHAR32_TYPE__ unsigned int
```

5.9.1.15 `__CHAR_BIT__`

```
#define __CHAR_BIT__ 8
```

5.9.1.16 `__code_model_small__`

```
#define __code_model_small__ 1
```

5.9.1.17 `__cplusplus`

```
#define __cplusplus 201402L
```

5.9.1.18 `__cpp_aggregate_nsdmi`

```
#define __cpp_aggregate_nsdmi 201304
```

5.9.1.19 `__cpp_alias_templates`

```
#define __cpp_alias_templates 200704
```

5.9.1.20 __cpp_attributes

```
#define __cpp_attributes 200809
```

5.9.1.21 __cpp_binary_literals

```
#define __cpp_binary_literals 201304
```

5.9.1.22 __cpp_constexpr

```
#define __cpp_constexpr 201304
```

5.9.1.23 __cpp_decltype

```
#define __cpp_decltype 200707
```

5.9.1.24 __cpp_decltype_auto

```
#define __cpp_decltype_auto 201304
```

5.9.1.25 __cpp_delegating_constructors

```
#define __cpp_delegating_constructors 200604
```

5.9.1.26 __cpp_digit_separators

```
#define __cpp_digit_separators 201309
```

5.9.1.27 __cpp_exceptions

```
#define __cpp_exceptions 199711
```

5.9.1.28 __cpp_generic_lambdas

```
#define __cpp_generic_lambdas 201304
```

5.9.1.29 __cpp_hex_float

```
#define __cpp_hex_float 201603
```

5.9.1.30 __cpp_inheriting_constructors

```
#define __cpp_inheriting_constructors 201511
```

5.9.1.31 __cpp_init_captures

```
#define __cpp_init_captures 201304
```

5.9.1.32 __cpp_initializer_lists

```
#define __cpp_initializer_lists 200806
```

5.9.1.33 __cpp_lambdas

```
#define __cpp_lambdas 200907
```

5.9.1.34 __cpp_nsdmi

```
#define __cpp_nsdmi 200809
```

5.9.1.35 __cpp_range_based_for

```
#define __cpp_range_based_for 200907
```

5.9.1.36 __cpp_raw_strings

```
#define __cpp_raw_strings 200710
```

5.9.1.37 __cpp_ref_qualifiers

```
#define __cpp_ref_qualifiers 200710
```

5.9.1.38 __cpp_return_type_deduction

```
#define __cpp_return_type_deduction 201304
```

5.9.1.39 __cpp_rtti

```
#define __cpp_rtti 199711
```

5.9.1.40 __cpp_runtime_arrays

```
#define __cpp_runtime_arrays 198712
```

5.9.1.41 __cpp_rvalue_reference

```
#define __cpp_rvalue_reference 200610
```

5.9.1.42 __cpp_rvalue_references

```
#define __cpp_rvalue_references 200610
```

5.9.1.43 __cpp_sized_deallocation

```
#define __cpp_sized_deallocation 201309
```


5.9.1.44 __cpp_static_assert

```
#define __cpp_static_assert 200410
```

5.9.1.45 __cpp_threadsafe_static_init

```
#define __cpp_threadsafe_static_init 200806
```

5.9.1.46 __cpp_unicode_characters

```
#define __cpp_unicode_characters 200704
```

5.9.1.47 __cpp_unicode_literals

```
#define __cpp_unicode_literals 200710
```

5.9.1.48 __cpp_user_defined_literals

```
#define __cpp_user_defined_literals 200809
```

5.9.1.49 __cpp_variable_templates

```
#define __cpp_variable_templates 201304
```

5.9.1.50 __cpp_variadic_templates

```
#define __cpp_variadic_templates 200704
```

5.9.1.51 __DBL_DECIMAL_DIG__

```
#define __DBL_DECIMAL_DIG__ 17
```

5.9.1.52 __DBL_DENORM_MIN__

```
#define __DBL_DENORM_MIN__ double(4.94065645841246544176568792868221372e-324L)
```

5.9.1.53 __DBL_DIG__

```
#define __DBL_DIG__ 15
```

5.9.1.54 __DBL_EPSILON__

```
#define __DBL_EPSILON__ double(2.22044604925031308084726333618164062e-16L)
```

5.9.1.55 __DBL_HAS_DENORM__

```
#define __DBL_HAS_DENORM__ 1
```

5.9.1.56 __DBL_HAS_INFINITY__

```
#define __DBL_HAS_INFINITY__ 1
```

5.9.1.57 __DBL_HAS_QUIET_NAN__

```
#define __DBL_HAS_QUIET_NAN__ 1
```

5.9.1.58 __DBL_MANT_DIG__

```
#define __DBL_MANT_DIG__ 53
```

5.9.1.59 __DBL_MAX_10_EXP__

```
#define __DBL_MAX_10_EXP__ 308
```

```
#define DBL_MAX double(1.79769313486231570814527423731704357e+308L)
```

```
#define DBL_MAX_EXP 1024
```

```
#define DBL_MIN_10_EXP (-307)
```

```
#define DBL_MIN double(2.22507385850720138309023271733240406e-308L)
```

```
#define __DBL_MIN_EXP__ (-1021)
```

```
#define __DEC128_EPSILON__ 1E-33DL
```

```
#define __DEC128_MANT_DIG__ 34
```

```
#define __DEC128_MAX__ 9.999999999999999999999999999E6144DL
```

```
#define __DEC128_MAX_EXP__ 6145
```

```
#define __DEC128_MIN__ 1E-6143DL
```

```
#define __DEC128_MIN_EXP__ (-6142)
```

[illegible]

```
#define __DEC32_EPSILON__ 1E-6DF
```

```
#define __DEC32_MANT_DIG__ 7
```

```
#define __DEC32_MAX__ 9.999999E96DF
```

```
#define __DEC32_MAX_EXP__ 97
```

5.9.1.76 __DEC32_MIN__

```
#define __DEC32_MIN__ 1E-95DF
```

5.9.1.77 __DEC32_MIN_EXP__

```
#define __DEC32_MIN_EXP__ (-94)
```

5.9.1.78 __DEC32_SUBNORMAL_MIN__

```
#define __DEC32_SUBNORMAL_MIN__ 0.000001E-95DF
```

5.9.1.79 __DEC64_EPSILON__

```
#define __DEC64_EPSILON__ 1E-15DD
```

5.9.1.80 __DEC64_MANT_DIG__

```
#define __DEC64_MANT_DIG__ 16
```

5.9.1.81 __DEC64_MAX__

```
#define __DEC64_MAX__ 9.999999999999999E384DD
```

5.9.1.82 __DEC64_MAX_EXP__

```
#define __DEC64_MAX_EXP__ 385
```

5.9.1.83 __DEC64_MIN__

```
#define __DEC64_MIN__ 1E-383DD
```

5.9.1.84 __DEC64_MIN_EXP__

```
#define __DEC64_MIN_EXP__ (-382)
```

5.9.1.85 __DEC64_SUBNORMAL_MIN__

```
#define __DEC64_SUBNORMAL_MIN__ 0.000000000000001E-383DD
```

5.9.1.86 __DEC_EVAL_METHOD__

```
#define __DEC_EVAL_METHOD__ 2
```

5.9.1.87 __DECIMAL_BID_FORMAT__

```
#define __DECIMAL_BID_FORMAT__ 1
```

5.9.1.88 __DECIMAL_DIG__

```
#define __DECIMAL_DIG__ 21
```

5.9.1.89 __DEPRECATED

```
#define __DEPRECATED 1
```

5.9.1.90 __ELF__

```
#define __ELF__ 1
```

5.9.1.91 __EXCEPTIONS

```
#define __EXCEPTIONS 1
```

5.9.1.92 __FINITE_MATH_ONLY__

```
#define __FINITE_MATH_ONLY__ 0
```

5.9.1.93 __FLOAT_WORD_ORDER__

```
#define __FLOAT_WORD_ORDER__ __ORDER_LITTLE_ENDIAN__
```

5.9.1.94 __FLT128_DECIMAL_DIG__

```
#define __FLT128_DECIMAL_DIG__ 36
```

5.9.1.95 __FLT128_DENORM_MIN__

```
#define __FLT128_DENORM_MIN__ 6.47517511943802511092443895822764655e-4966F128
```

5.9.1.96 __FLT128_DIG__

```
#define __FLT128_DIG__ 33
```

5.9.1.97 __FLT128_EPSILON__

```
#define __FLT128_EPSILON__ 1.92592994438723585305597794258492732e-34F128
```

5.9.1.98 __FLT128_HAS_DENORM__

```
#define __FLT128_HAS_DENORM__ 1
```

5.9.1.99 __FLT128_HAS_INFINITY__

```
#define __FLT128_HAS_INFINITY__ 1
```

5.9.1.100 __FLT128_HAS_QUIET_NAN__

```
#define __FLT128_HAS_QUIET_NAN__ 1
```

5.9.1.101 __FLT128_MANT_DIG__

```
#define __FLT128_MANT_DIG__ 113
```

5.9.1.102 __FLT128_MAX_10_EXP__

```
#define __FLT128_MAX_10_EXP__ 4932
```

5.9.1.103 __FLT128_MAX__

```
#define __FLT128_MAX__ 1.18973149535723176508575932662800702e+4932F128
```

5.9.1.104 __FLT128_MAX_EXP__

```
#define __FLT128_MAX_EXP__ 16384
```

5.9.1.105 __FLT128_MIN_10_EXP__

```
#define __FLT128_MIN_10_EXP__ (-4931)
```

5.9.1.106 __FLT128_MIN__

```
#define __FLT128_MIN__ 3.36210314311209350626267781732175260e-4932F128
```

5.9.1.107 __FLT128_MIN_EXP__

```
#define __FLT128_MIN_EXP__ (-16381)
```


5.9.1.108 __FLT32_DECIMAL_DIG__

```
#define __FLT32_DECIMAL_DIG__ 9
```

5.9.1.109 __FLT32_DENORM_MIN__

```
#define __FLT32_DENORM_MIN__ 1.40129846432481707092372958328991613e-45F32
```

5.9.1.110 __FLT32_DIG__

```
#define __FLT32_DIG__ 6
```

5.9.1.111 __FLT32_EPSILON__

```
#define __FLT32_EPSILON__ 1.1920928955078125000000000000000000000000e-7F32
```

5.9.1.112 __FLT32_HAS_DENORM__

```
#define __FLT32_HAS_DENORM__ 1
```

5.9.1.113 __FLT32_HAS_INFINITY__

```
#define __FLT32_HAS_INFINITY__ 1
```

5.9.1.114 __FLT32_HAS_QUIET_NAN__

```
#define __FLT32_HAS_QUIET_NAN__ 1
```

5.9.1.115 __FLT32_MANT_DIG__

```
#define __FLT32_MANT_DIG__ 24
```

5.9.1.116 __FLT32_MAX_10_EXP__

```
#define __FLT32_MAX_10_EXP__ 38
```

5.9.1.117 __FLT32_MAX__

```
#define __FLT32_MAX__ 3.40282346638528859811704183484516925e+38F32
```

5.9.1.118 __FLT32_MAX_EXP__

```
#define __FLT32_MAX_EXP__ 128
```

5.9.1.119 __FLT32_MIN_10_EXP__

```
#define __FLT32_MIN_10_EXP__ (-37)
```

5.9.1.120 __FLT32_MIN__

```
#define __FLT32_MIN__ 1.17549435082228750796873653722224568e-38F32
```

5.9.1.121 __FLT32_MIN_EXP__

```
#define __FLT32_MIN_EXP__ (-125)
```

5.9.1.122 __FLT32X_DECIMAL_DIG__

```
#define __FLT32X_DECIMAL_DIG__ 17
```

5.9.1.123 __FLT32X_DENORM_MIN__

```
#define __FLT32X_DENORM_MIN__ 4.94065645841246544176568792868221372e-324F32x
```

5.9.1.124 __FLT32X_DIG__

```
#define __FLT32X_DIG__ 15
```

5.9.1.125 __FLT32X_EPSILON__

```
#define __FLT32X_EPSILON__ 2.22044604925031308084726333618164062e-16F32x
```

5.9.1.126 __FLT32X_HAS_DENORM__

```
#define __FLT32X_HAS_DENORM__ 1
```

5.9.1.127 __FLT32X_HAS_INFINITY__

```
#define __FLT32X_HAS_INFINITY__ 1
```

5.9.1.128 __FLT32X_HAS_QUIET_NAN__

```
#define __FLT32X_HAS_QUIET_NAN__ 1
```

5.9.1.129 __FLT32X_MANT_DIG__

```
#define __FLT32X_MANT_DIG__ 53
```

5.9.1.130 __FLT32X_MAX_10_EXP__

```
#define __FLT32X_MAX_10_EXP__ 308
```

5.9.1.131 __FLT32X_MAX__

```
#define __FLT32X_MAX__ 1.79769313486231570814527423731704357e+308F32x
```

5.9.1.132 __FLT32X_MAX_EXP__

```
#define __FLT32X_MAX_EXP__ 1024
```

5.9.1.133 __FLT32X_MIN_10_EXP__

```
#define __FLT32X_MIN_10_EXP__ (-307)
```

5.9.1.134 __FLT32X_MIN__

```
#define __FLT32X_MIN__ 2.22507385850720138309023271733240406e-308F32x
```

5.9.1.135 __FLT32X_MIN_EXP__

```
#define __FLT32X_MIN_EXP__ (-1021)
```

5.9.1.136 __FLT64_DECIMAL_DIG__

```
#define __FLT64_DECIMAL_DIG__ 17
```

5.9.1.137 __FLT64_DENORM_MIN__

```
#define __FLT64_DENORM_MIN__ 4.94065645841246544176568792868221372e-324F64
```

5.9.1.138 __FLT64_DIG__

```
#define __FLT64_DIG__ 15
```

5.9.1.139 __FLT64_EPSILON__

```
#define __FLT64_EPSILON__ 2.22044604925031308084726333618164062e-16F64
```

5.9.1.140 __FLT64_HAS_DENORM__

```
#define __FLT64_HAS_DENORM__ 1
```

5.9.1.141 __FLT64_HAS_INFINITY__

```
#define __FLT64_HAS_INFINITY__ 1
```

5.9.1.142 __FLT64_HAS_QUIET_NAN__

```
#define __FLT64_HAS_QUIET_NAN__ 1
```

5.9.1.143 __FLT64_MANT_DIG__

```
#define __FLT64_MANT_DIG__ 53
```

5.9.1.144 __FLT64_MAX_10_EXP__

```
#define __FLT64_MAX_10_EXP__ 308
```

5.9.1.145 __FLT64_MAX__

```
#define __FLT64_MAX__ 1.79769313486231570814527423731704357e+308F64
```

5.9.1.146 __FLT64_MAX_EXP__

```
#define __FLT64_MAX_EXP__ 1024
```

5.9.1.147 __FLT64_MIN_10_EXP__

```
#define __FLT64_MIN_10_EXP__ (-307)
```

5.9.1.148 __FLT64_MIN__

```
#define __FLT64_MIN__ 2.22507385850720138309023271733240406e-308F64
```

5.9.1.149 __FLT64_MIN_EXP__

```
#define __FLT64_MIN_EXP__ (-1021)
```

5.9.1.150 __FLT64X_DECIMAL_DIG__

```
#define __FLT64X_DECIMAL_DIG__ 21
```

5.9.1.151 __FLT64X_DENORM_MIN__

```
#define __FLT64X_DENORM_MIN__ 3.64519953188247460252840593361941982e-4951F64x
```

5.9.1.152 __FLT64X_DIG__

```
#define __FLT64X_DIG__ 18
```

5.9.1.153 __FLT64X_EPSILON__

```
#define __FLT64X_EPSILON__ 1.08420217248550443400745280086994171e-19F64x
```

5.9.1.154 __FLT64X_HAS_DENORM__

```
#define __FLT64X_HAS_DENORM__ 1
```

5.9.1.155 __FLT64X_HAS_INFINITY__

```
#define __FLT64X_HAS_INFINITY__ 1
```

5.9.1.156 __FLT64X_HAS_QUIET_NAN__

```
#define __FLT64X_HAS_QUIET_NAN__ 1
```

5.9.1.157 __FLT64X_MANT_DIG__

```
#define __FLT64X_MANT_DIG__ 64
```

5.9.1.158 __FLT64X_MAX_10_EXP__

```
#define __FLT64X_MAX_10_EXP__ 4932
```

5.9.1.159 __FLT64X_MAX__

```
#define __FLT64X_MAX__ 1.18973149535723176502126385303097021e+4932F64x
```

5.9.1.160 __FLT64X_MAX_EXP__

```
#define __FLT64X_MAX_EXP__ 16384
```

5.9.1.161 __FLT64X_MIN_10_EXP__

```
#define __FLT64X_MIN_10_EXP__ (-4931)
```

5.9.1.162 __FLT64X_MIN__

```
#define __FLT64X_MIN__ 3.36210314311209350626267781732175260e-4932F64x
```

5.9.1.163 __FLT64X_MIN_EXP__

```
#define __FLT64X_MIN_EXP__ (-16381)
```

5.9.1.164 __FLT_DECIMAL_DIG__

```
#define __FLT_DECIMAL_DIG__ 9
```

5.9.1.165 __FLT_DENORM_MIN__

```
#define __FLT_DENORM_MIN__ 1.40129846432481707092372958328991613e-45F
```

5.9.1.166 __FLT_DIG__

```
#define __FLT_DIG__ 6
```

5.9.1.167 __FLT_EPSILON__

```
#define __FLT_EPSILON__ 1.192092895507812500000000000000000000e-7F
```

5.9.1.168 __FLT_EVAL_METHOD__

```
#define __FLT_EVAL_METHOD__ 0
```

5.9.1.169 __FLT_EVAL_METHOD_TS_18661_3__

```
#define __FLT_EVAL_METHOD_TS_18661_3__ 0
```

5.9.1.170 __FLT_HAS_DENORM__

```
#define __FLT_HAS_DENORM__ 1
```

5.9.1.171 __FLT_HAS_INFINITY__

```
#define __FLT_HAS_INFINITY__ 1
```


5.9.1.172 __FLT_HAS_QUIET_NAN__

```
#define __FLT_HAS_QUIET_NAN__ 1
```

5.9.1.173 __FLT_MANT_DIG__

```
#define __FLT_MANT_DIG__ 24
```

5.9.1.174 __FLT_MAX_10_EXP__

```
#define __FLT_MAX_10_EXP__ 38
```

5.9.1.175 __FLT_MAX__

```
#define __FLT_MAX__ 3.40282346638528859811704183484516925e+38F
```

5.9.1.176 __FLT_MAX_EXP__

```
#define __FLT_MAX_EXP__ 128
```

5.9.1.177 __FLT_MIN_10_EXP__

```
#define __FLT_MIN_10_EXP__ (-37)
```

5.9.1.178 __FLT_MIN__

```
#define __FLT_MIN__ 1.1754943508222875079687365372224568e-38F
```

5.9.1.179 __FLT_MIN_EXP__

```
#define __FLT_MIN_EXP__ (-125)
```

5.9.1.180 __FLT_RADIX__

```
#define __FLT_RADIX__ 2
```

5.9.1.181 __FXSR__

```
#define __FXSR__ 1
```

5.9.1.182 __GCC_ASM_FLAG_OUTPUTS__

```
#define __GCC_ASM_FLAG_OUTPUTS__ 1
```

5.9.1.183 __GCC_ATOMIC_BOOL_LOCK_FREE

```
#define __GCC_ATOMIC_BOOL_LOCK_FREE 2
```

5.9.1.184 __GCC_ATOMIC_CHAR16_T_LOCK_FREE

```
#define __GCC_ATOMIC_CHAR16_T_LOCK_FREE 2
```

5.9.1.185 __GCC_ATOMIC_CHAR32_T_LOCK_FREE

```
#define __GCC_ATOMIC_CHAR32_T_LOCK_FREE 2
```

5.9.1.186 __GCC_ATOMIC_CHAR_LOCK_FREE

```
#define __GCC_ATOMIC_CHAR_LOCK_FREE 2
```

5.9.1.187 __GCC_ATOMIC_INT_LOCK_FREE

```
#define __GCC_ATOMIC_INT_LOCK_FREE 2
```

5.9.1.188 __GCC_ATOMIC_LLONG_LOCK_FREE

```
#define __GCC_ATOMIC_LLONG_LOCK_FREE 2
```

5.9.1.189 __GCC_ATOMIC_LONG_LOCK_FREE

```
#define __GCC_ATOMIC_LONG_LOCK_FREE 2
```

5.9.1.190 __GCC_ATOMIC_POINTER_LOCK_FREE

```
#define __GCC_ATOMIC_POINTER_LOCK_FREE 2
```

5.9.1.191 __GCC_ATOMIC_SHORT_LOCK_FREE

```
#define __GCC_ATOMIC_SHORT_LOCK_FREE 2
```

5.9.1.192 __GCC_ATOMIC_TEST_AND_SET_TRUEVAL

```
#define __GCC_ATOMIC_TEST_AND_SET_TRUEVAL 1
```

5.9.1.193 __GCC_ATOMIC_WCHAR_T_LOCK_FREE

```
#define __GCC_ATOMIC_WCHAR_T_LOCK_FREE 2
```

5.9.1.194 __GCC_HAVE_DWARF2_CFI_ASM

```
#define __GCC_HAVE_DWARF2_CFI_ASM 1
```

5.9.1.195 __GCC_HAVE_SYNC_COMPARE_AND_SWAP_1

```
#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_1 1
```

5.9.1.196 __GCC_HAVE_SYNC_COMPARE_AND_SWAP_2

```
#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_2 1
```

5.9.1.197 __GCC_HAVE_SYNC_COMPARE_AND_SWAP_4

```
#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_4 1
```

5.9.1.198 __GCC_HAVE_SYNC_COMPARE_AND_SWAP_8

```
#define __GCC_HAVE_SYNC_COMPARE_AND_SWAP_8 1
```

5.9.1.199 __GCC_IEC_559

```
#define __GCC_IEC_559 2
```

5.9.1.200 __GCC_IEC_559_COMPLEX

```
#define __GCC_IEC_559_COMPLEX 2
```

5.9.1.201 __GLIBCXX_BITSIZE_INT_N_0

```
#define __GLIBCXX_BITSIZE_INT_N_0 128
```

5.9.1.202 __GLIBCXX_TYPE_INT_N_0

```
#define __GLIBCXX_TYPE_INT_N_0 __int128
```

5.9.1.203 __gnu_linux__

```
#define __gnu_linux__ 1
```

5.9.1.204 __GNUC__

```
#define __GNUC__ 7
```

5.9.1.205 __GNUC_MINOR__

```
#define __GNUC_MINOR__ 5
```

5.9.1.206 __GNUC_PATCHLEVEL__

```
#define __GNUC_PATCHLEVEL__ 0
```

5.9.1.207 __GNUC_STDC_INLINE__

```
#define __GNUC_STDC_INLINE__ 1
```

5.9.1.208 __GNUG__

```
#define __GNUG__ 7
```

5.9.1.209 __GXX_ABI_VERSION

```
#define __GXX_ABI_VERSION 1011
```

5.9.1.210 __GXX_EXPERIMENTAL_CXX0X__

```
#define __GXX_EXPERIMENTAL_CXX0X__ 1
```

5.9.1.211 __GXX_RTTI

```
#define __GXX_RTTI 1
```

5.9.1.212 __GXX_WEAK__

```
#define __GXX_WEAK__ 1
```

5.9.1.213 __has_include

```
#define __has_include(  
    STR ) __has_include__(STR)
```

5.9.1.214 __has_include_next

```
#define __has_include_next(  
    STR ) __has_include_next__(STR)
```

5.9.1.215 __INT16_C

```
#define __INT16_C(  
    c ) c
```

5.9.1.216 __INT16_MAX__

```
#define __INT16_MAX__ 0x7fff
```

5.9.1.217 __INT16_TYPE__

```
#define __INT16_TYPE__ short int
```

5.9.1.218 __INT32_C

```
#define __INT32_C(  
    c ) c
```

5.9.1.219 __INT32_MAX__

```
#define __INT32_MAX__ 0x7fffffff
```

5.9.1.220 __INT32_TYPE__

```
#define __INT32_TYPE__ int
```

5.9.1.221 __INT64_C

```
#define __INT64_C(  
    c ) c ## L
```

5.9.1.222 __INT64_MAX__

```
#define __INT64_MAX__ 0xffffffffffffffffL
```

5.9.1.223 __INT64_TYPE__

```
#define __INT64_TYPE__ long int
```

5.9.1.224 __INT8_C

```
#define __INT8_C(  
    c ) c
```

5.9.1.225 __INT8_MAX__

```
#define __INT8_MAX__ 0x7f
```

5.9.1.226 __INT8_TYPE__

```
#define __INT8_TYPE__ signed char
```

5.9.1.227 __INT_FAST16_MAX__

```
#define __INT_FAST16_MAX__ 0x7fffffffffffffffL
```

5.9.1.228 __INT_FAST16_TYPE__

```
#define __INT_FAST16_TYPE__ long int
```

5.9.1.229 __INT_FAST16_WIDTH__

```
#define __INT_FAST16_WIDTH__ 64
```

5.9.1.230 __INT_FAST32_MAX__

```
#define __INT_FAST32_MAX__ 0x7fffffffffffffffL
```

5.9.1.231 __INT_FAST32_TYPE__

```
#define __INT_FAST32_TYPE__ long int
```

5.9.1.232 __INT_FAST32_WIDTH__

```
#define __INT_FAST32_WIDTH__ 64
```

5.9.1.233 __INT_FAST64_MAX__

```
#define __INT_FAST64_MAX__ 0x7fffffffffffffffL
```


5.9.1.234 __INT_FAST64_TYPE__

```
#define __INT_FAST64_TYPE__ long int
```

5.9.1.235 __INT_FAST64_WIDTH__

```
#define __INT_FAST64_WIDTH__ 64
```

5.9.1.236 __INT_FAST8_MAX__

```
#define __INT_FAST8_MAX__ 0x7f
```

5.9.1.237 __INT_FAST8_TYPE__

```
#define __INT_FAST8_TYPE__ signed char
```

5.9.1.238 __INT_FAST8_WIDTH__

```
#define __INT_FAST8_WIDTH__ 8
```

5.9.1.239 __INT_LEAST16_MAX__

```
#define __INT_LEAST16_MAX__ 0x7fff
```

5.9.1.240 __INT_LEAST16_TYPE__

```
#define __INT_LEAST16_TYPE__ short int
```

5.9.1.241 __INT_LEAST16_WIDTH__

```
#define __INT_LEAST16_WIDTH__ 16
```

5.9.1.242 __INT_LEAST32_MAX__

```
#define __INT_LEAST32_MAX__ 0x7fffffff
```

5.9.1.243 __INT_LEAST32_TYPE__

```
#define __INT_LEAST32_TYPE__ int
```

5.9.1.244 __INT_LEAST32_WIDTH__

```
#define __INT_LEAST32_WIDTH__ 32
```

5.9.1.245 __INT_LEAST64_MAX__

```
#define __INT_LEAST64_MAX__ 0x7fffffffffffffffL
```

5.9.1.246 __INT_LEAST64_TYPE__

```
#define __INT_LEAST64_TYPE__ long int
```

5.9.1.247 __INT_LEAST64_WIDTH__

```
#define __INT_LEAST64_WIDTH__ 64
```

5.9.1.248 __INT_LEAST8_MAX__

```
#define __INT_LEAST8_MAX__ 0x7f
```

5.9.1.249 __INT_LEAST8_TYPE__

```
#define __INT_LEAST8_TYPE__ signed char
```

5.9.1.250 __INT_LEAST8_WIDTH__

```
#define __INT_LEAST8_WIDTH__ 8
```

5.9.1.251 __INT_MAX__

```
#define __INT_MAX__ 0x7fffffff
```

5.9.1.252 __INT_WIDTH__

```
#define __INT_WIDTH__ 32
```

5.9.1.253 __INTMAX_C

```
#define __INTMAX_C(  
    c ) c ## L
```

5.9.1.254 __INTMAX_MAX__

```
#define __INTMAX_MAX__ 0x7fffffffffffffffL
```

5.9.1.255 __INTMAX_TYPE__

```
#define __INTMAX_TYPE__ long int
```

5.9.1.256 __INTMAX_WIDTH__

```
#define __INTMAX_WIDTH__ 64
```

5.9.1.257 __INTPTR_MAX__

```
#define __INTPTR_MAX__ 0x7fffffffffffffffL
```

5.9.1.258 __INTPTR_TYPE__

```
#define __INTPTR_TYPE__ long int
```

5.9.1.259 __INTPTR_WIDTH__

```
#define __INTPTR_WIDTH__ 64
```

5.9.1.260 __k8

```
#define __k8 1
```

5.9.1.261 __k8__

```
#define __k8__ 1
```

5.9.1.262 __LDBL_DECIMAL_DIG__

```
#define __LDBL_DECIMAL_DIG__ 21
```

5.9.1.263 __LDBL_DENORM_MIN__

```
#define __LDBL_DENORM_MIN__ 3.64519953188247460252840593361941982e-4951L
```

5.9.1.264 __LDBL_DIG__

```
#define __LDBL_DIG__ 18
```

5.9.1.265 __LDBL_EPSILON__

```
#define __LDBL_EPSILON__ 1.08420217248550443400745280086994171e-19L
```

5.9.1.266 __LDBL_HAS_DENORM__

```
#define __LDBL_HAS_DENORM__ 1
```

5.9.1.267 __LDBL_HAS_INFINITY__

```
#define __LDBL_HAS_INFINITY__ 1
```

5.9.1.268 __LDBL_HAS_QUIET_NAN__

```
#define __LDBL_HAS_QUIET_NAN__ 1
```

5.9.1.269 __LDBL_MANT_DIG__

```
#define __LDBL_MANT_DIG__ 64
```

5.9.1.270 __LDBL_MAX_10_EXP__

```
#define __LDBL_MAX_10_EXP__ 4932
```

5.9.1.271 __LDBL_MAX__

```
#define __LDBL_MAX__ 1.18973149535723176502126385303097021e+4932L
```

5.9.1.272 __LDBL_MAX_EXP__

```
#define __LDBL_MAX_EXP__ 16384
```

5.9.1.273 __LDBL_MIN_10_EXP__

```
#define __LDBL_MIN_10_EXP__ (-4931)
```

5.9.1.274 __LDBL_MIN__

```
#define __LDBL_MIN__ 3.36210314311209350626267781732175260e-4932L
```

5.9.1.275 __LDBL_MIN_EXP__

```
#define __LDBL_MIN_EXP__ (-16381)
```

5.9.1.276 __linux

```
#define __linux 1
```

5.9.1.277 __linux__

```
#define __linux__ 1
```

5.9.1.278 __LONG_LONG_MAX__

```
#define __LONG_LONG_MAX__ 0x7fffffffffffffffffLL
```

5.9.1.279 __LONG_LONG_WIDTH__

```
#define __LONG_LONG_WIDTH__ 64
```

5.9.1.280 __LONG_MAX__

```
#define __LONG_MAX__ 0x7fffffffffffffffL
```

5.9.1.281 __LONG_WIDTH__

```
#define __LONG_WIDTH__ 64
```

5.9.1.282 __LP64__

```
#define __LP64__ 1
```

5.9.1.283 __MMX__

```
#define __MMX__ 1
```

5.9.1.284 __OPTIMIZE__

```
#define __OPTIMIZE__ 1
```

5.9.1.285 __ORDER_BIG_ENDIAN__

```
#define __ORDER_BIG_ENDIAN__ 4321
```

5.9.1.286 __ORDER_LITTLE_ENDIAN__

```
#define __ORDER_LITTLE_ENDIAN__ 1234
```

5.9.1.287 __ORDER_PDP_ENDIAN__

```
#define __ORDER_PDP_ENDIAN__ 3412
```

5.9.1.288 __pic__

```
#define __pic__ 2
```

5.9.1.289 __PIC__

```
#define __PIC__ 2
```

5.9.1.290 __pie__

```
#define __pie__ 2
```

5.9.1.291 __PIE__

```
#define __PIE__ 2
```

5.9.1.292 __PRAGMA_REDEFINE_EXTNAME

```
#define __PRAGMA_REDEFINE_EXTNAME 1
```

5.9.1.293 __PTRDIFF_MAX__

```
#define __PTRDIFF_MAX__ 0x7fffffffffffffffL
```

5.9.1.294 __PTRDIFF_TYPE__

```
#define __PTRDIFF_TYPE__ long int
```

5.9.1.295 __PTRDIFF_WIDTH__

```
#define __PTRDIFF_WIDTH__ 64
```

5.9.1.296 __REGISTER_PREFIX__

```
#define __REGISTER_PREFIX__
```


5.9.1.297 __SCHAR_MAX__

```
#define __SCHAR_MAX__ 0x7f
```

5.9.1.298 __SCHAR_WIDTH__

```
#define __SCHAR_WIDTH__ 8
```

5.9.1.299 __SEG_FS

```
#define __SEG_FS 1
```

5.9.1.300 __SEG_GS

```
#define __SEG_GS 1
```

5.9.1.301 __SHRT_MAX__

```
#define __SHRT_MAX__ 0x7fff
```

5.9.1.302 __SHRT_WIDTH__

```
#define __SHRT_WIDTH__ 16
```

5.9.1.303 __SIG_ATOMIC_MAX__

```
#define __SIG_ATOMIC_MAX__ 0x7fffffff
```

5.9.1.304 __SIG_ATOMIC_MIN__

```
#define __SIG_ATOMIC_MIN__ (-__SIG_ATOMIC_MAX__ - 1)
```

5.9.1.305 __SIG_ATOMIC_TYPE__

```
#define __SIG_ATOMIC_TYPE__ int
```

5.9.1.306 __SIG_ATOMIC_WIDTH__

```
#define __SIG_ATOMIC_WIDTH__ 32
```

5.9.1.307 __SIZE_MAX__

```
#define __SIZE_MAX__ 0xffffffffffffffffUL
```

5.9.1.308 __SIZE_TYPE__

```
#define __SIZE_TYPE__ long unsigned int
```

5.9.1.309 __SIZE_WIDTH__

```
#define __SIZE_WIDTH__ 64
```

5.9.1.310 __SIZEOF_DOUBLE__

```
#define __SIZEOF_DOUBLE__ 8
```

5.9.1.311 __SIZEOF_FLOAT128__

```
#define __SIZEOF_FLOAT128__ 16
```

5.9.1.312 __SIZEOF_FLOAT80__

```
#define __SIZEOF_FLOAT80__ 16
```

5.9.1.313 __SIZEOF_FLOAT__

```
#define __SIZEOF_FLOAT__ 4
```

5.9.1.314 __SIZEOF_INT128__

```
#define __SIZEOF_INT128__ 16
```

5.9.1.315 __SIZEOF_INT__

```
#define __SIZEOF_INT__ 4
```

5.9.1.316 __SIZEOF_LONG__

```
#define __SIZEOF_LONG__ 8
```

5.9.1.317 __SIZEOF_LONG_DOUBLE__

```
#define __SIZEOF_LONG_DOUBLE__ 16
```

5.9.1.318 __SIZEOF_LONG_LONG__

```
#define __SIZEOF_LONG_LONG__ 8
```

5.9.1.319 __SIZEOF_POINTER__

```
#define __SIZEOF_POINTER__ 8
```

5.9.1.320 __SIZEOF_PTRDIFF_T__

```
#define __SIZEOF_PTRDIFF_T__ 8
```

5.9.1.321 __SIZEOF_SHORT__

```
#define __SIZEOF_SHORT__ 2
```

5.9.1.322 __SIZEOF_SIZE_T__

```
#define __SIZEOF_SIZE_T__ 8
```

5.9.1.323 __SIZEOF_WCHAR_T__

```
#define __SIZEOF_WCHAR_T__ 4
```

5.9.1.324 __SIZEOF_WINT_T__

```
#define __SIZEOF_WINT_T__ 4
```

5.9.1.325 __SSE2__

```
#define __SSE2__ 1
```

5.9.1.326 __SSE2_MATH__

```
#define __SSE2_MATH__ 1
```

5.9.1.327 __SSE__

```
#define __SSE__ 1
```

5.9.1.328 __SSE_MATH__

```
#define __SSE_MATH__ 1
```

5.9.1.329 __SSP_STRONG__

```
#define __SSP_STRONG__ 3
```

5.9.1.330 __STDC__

```
#define __STDC__ 1
```

5.9.1.331 __STDC_HOSTED__

```
#define __STDC_HOSTED__ 1
```

5.9.1.332 __STDC_IEC_559__

```
#define __STDC_IEC_559__ 1
```

5.9.1.333 __STDC_IEC_559_COMPLEX__

```
#define __STDC_IEC_559_COMPLEX__ 1
```

5.9.1.334 __STDC_ISO_10646__

```
#define __STDC_ISO_10646__ 201706L
```

5.9.1.335 __STDC_NO_THREADS__

```
#define __STDC_NO_THREADS__ 1
```

5.9.1.336 __STDC_UTF_16__

```
#define __STDC_UTF_16__ 1
```

5.9.1.337 __STDC_UTF_32__

```
#define __STDC_UTF_32__ 1
```

5.9.1.338 __UINT16_C

```
#define __UINT16_C(  
    c ) c
```

5.9.1.339 __UINT16_MAX__

```
#define __UINT16_MAX__ 0xffff
```

5.9.1.340 __UINT16_TYPE__

```
#define __UINT16_TYPE__ short unsigned int
```

5.9.1.341 __UINT32_C

```
#define __UINT32_C(  
    c ) c ## U
```

5.9.1.342 __UINT32_MAX__

```
#define __UINT32_MAX__ 0xffffffffU
```

5.9.1.343 __UINT32_TYPE__

```
#define __UINT32_TYPE__ unsigned int
```

5.9.1.344 __UINT64_C

```
#define __UINT64_C(  
    c ) c ## UL
```

5.9.1.345 __UINT64_MAX__

```
#define __UINT64_MAX__ 0xffffffffffffffffUL
```

5.9.1.346 __UINT64_TYPE__

```
#define __UINT64_TYPE__ long unsigned int
```

5.9.1.347 __UINT8_C

```
#define __UINT8_C(  
    c ) c
```

5.9.1.348 __UINT8_MAX__

```
#define __UINT8_MAX__ 0xff
```

5.9.1.349 __UINT8_TYPE__

```
#define __UINT8_TYPE__ unsigned char
```

5.9.1.350 __UINT_FAST16_MAX__

```
#define __UINT_FAST16_MAX__ 0xffffffffffffffffUL
```

5.9.1.351 __UINT_FAST16_TYPE__

```
#define __UINT_FAST16_TYPE__ long unsigned int
```

5.9.1.352 __UINT_FAST32_MAX__

```
#define __UINT_FAST32_MAX__ 0xffffffffffffffffUL
```

5.9.1.353 __UINT_FAST32_TYPE__

```
#define __UINT_FAST32_TYPE__ long unsigned int
```

5.9.1.354 __UINT_FAST64_MAX__

```
#define __UINT_FAST64_MAX__ 0xffffffffffffffffUL
```

5.9.1.355 __UINT_FAST64_TYPE__

```
#define __UINT_FAST64_TYPE__ long unsigned int
```

5.9.1.356 __UINT_FAST8_MAX__

```
#define __UINT_FAST8_MAX__ 0xff
```

5.9.1.357 __UINT_FAST8_TYPE__

```
#define __UINT_FAST8_TYPE__ unsigned char
```

5.9.1.358 __UINT_LEAST16_MAX__

```
#define __UINT_LEAST16_MAX__ 0xffff
```


5.9.1.359 __UINT_LEAST16_TYPE__

```
#define __UINT_LEAST16_TYPE__ short unsigned int
```

5.9.1.360 __UINT_LEAST32_MAX__

```
#define __UINT_LEAST32_MAX__ 0xffffffffU
```

5.9.1.361 __UINT_LEAST32_TYPE__

```
#define __UINT_LEAST32_TYPE__ unsigned int
```

5.9.1.362 __UINT_LEAST64_MAX__

```
#define __UINT_LEAST64_MAX__ 0xffffffffffffffffUL
```

5.9.1.363 __UINT_LEAST64_TYPE__

```
#define __UINT_LEAST64_TYPE__ long unsigned int
```

5.9.1.364 __UINT_LEAST8_MAX__

```
#define __UINT_LEAST8_MAX__ 0xff
```

5.9.1.365 __UINT_LEAST8_TYPE__

```
#define __UINT_LEAST8_TYPE__ unsigned char
```

5.9.1.366 __UINTMAX_C

```
#define __UINTMAX_C(  
    c ) c ## UL
```

5.9.1.367 __UINTMAX_MAX__

```
#define __UINTMAX_MAX__ 0xffffffffffffffffUL
```

5.9.1.368 __UINTMAX_TYPE__

```
#define __UINTMAX_TYPE__ long unsigned int
```

5.9.1.369 __UINTPTR_MAX__

```
#define __UINTPTR_MAX__ 0xffffffffffffffffUL
```

5.9.1.370 __UINTPTR_TYPE__

```
#define __UINTPTR_TYPE__ long unsigned int
```

5.9.1.371 __unix

```
#define __unix 1
```

5.9.1.372 __unix__

```
#define __unix__ 1
```

5.9.1.373 __USER_LABEL_PREFIX__

```
#define __USER_LABEL_PREFIX__
```

5.9.1.374 __VERSION__

```
#define __VERSION__ "7.5.0"
```

5.9.1.375 __WCHAR_MAX__

```
#define __WCHAR_MAX__ 0x7fffffff
```

5.9.1.376 __WCHAR_MIN__

```
#define __WCHAR_MIN__ (-__WCHAR_MAX__ - 1)
```

5.9.1.377 __WCHAR_TYPE__

```
#define __WCHAR_TYPE__ int
```

5.9.1.378 __WCHAR_WIDTH__

```
#define __WCHAR_WIDTH__ 32
```

5.9.1.379 __WINT_MAX__

```
#define __WINT_MAX__ 0xffffffffU
```

5.9.1.380 __WINT_MIN__

```
#define __WINT_MIN__ 0U
```

5.9.1.381 __WINT_TYPE__

```
#define __WINT_TYPE__ unsigned int
```

5.9.1.382 __WINT_WIDTH__

```
#define __WINT_WIDTH__ 32
```

5.9.1.383 __x86_64

```
#define __x86_64 1
```

5.9.1.384 __x86_64__

```
#define __x86_64__ 1
```

5.9.1.385 _FORTIFY_SOURCE

```
#define _FORTIFY_SOURCE 2
```

5.9.1.386 _GNU_SOURCE

```
#define _GNU_SOURCE 1
```

5.9.1.387 _LP64

```
#define _LP64 1
```

5.9.1.388 _STDC_PREDEF_H

```
#define _STDC_PREDEF_H 1
```

5.9.1.389 linux

```
#define linux 1
```

5.9.1.390 unix

```
#define unix 1
```

Index

`__FORTIFY_SOURCE`
 [moc_predefs.h, 74](#)

`__GNU_SOURCE`
 [moc_predefs.h, 74](#)

`__LP64`
 [moc_predefs.h, 74](#)

`__STDC_PREDEF_H`
 [moc_predefs.h, 74](#)

`__ATOMIC_ACQUIRE`
 [moc_predefs.h, 25](#)

`__ATOMIC_ACQ_REL`
 [moc_predefs.h, 25](#)

`__ATOMIC_CONSUME`
 [moc_predefs.h, 26](#)

`__ATOMIC_HLE_ACQUIRE`
 [moc_predefs.h, 26](#)

`__ATOMIC_HLE_RELEASE`
 [moc_predefs.h, 26](#)

`__ATOMIC_RELAXED`
 [moc_predefs.h, 26](#)

`__ATOMIC_RELEASE`
 [moc_predefs.h, 26](#)

`__ATOMIC_SEQ_CST`
 [moc_predefs.h, 26](#)

`__BIGGEST_ALIGNMENT__`
 [moc_predefs.h, 26](#)

`__BYTE_ORDER__`
 [moc_predefs.h, 26](#)

`__CHAR16_TYPE__`
 [moc_predefs.h, 27](#)

`__CHAR32_TYPE__`
 [moc_predefs.h, 27](#)

`__CHAR_BIT__`
 [moc_predefs.h, 27](#)

`__DBL_DECIMAL_DIG__`
 [moc_predefs.h, 31](#)

`__DBL_DENORM_MIN__`
 [moc_predefs.h, 31](#)

`__DBL_DIG__`
 [moc_predefs.h, 32](#)

`__DBL_EPSILON__`
 [moc_predefs.h, 32](#)

`__DBL_HAS_DENORM__`
 [moc_predefs.h, 32](#)

`__DBL_HAS_INFINITY__`
 [moc_predefs.h, 32](#)

`__DBL_HAS_QUIET_NAN__`
 [moc_predefs.h, 32](#)

`__DBL_MANT_DIG__`
 [moc_predefs.h, 32](#)

`__DBL_MAX_10_EXP__`
 [moc_predefs.h, 32](#)

`__DBL_MAX_EXP__`
 [moc_predefs.h, 33](#)

`__DBL_MAX__`
 [moc_predefs.h, 32](#)

`__DBL_MIN_10_EXP__`
 [moc_predefs.h, 33](#)

`__DBL_MIN_EXP__`
 [moc_predefs.h, 33](#)

`__DBL_MIN__`
 [moc_predefs.h, 33](#)

`__DEC128_EPSILON__`
 [moc_predefs.h, 33](#)

`__DEC128_MANT_DIG__`
 [moc_predefs.h, 33](#)

`__DEC128_MAX_EXP__`
 [moc_predefs.h, 33](#)

`__DEC128_MAX__`
 [moc_predefs.h, 33](#)

`__DEC128_MIN_EXP__`
 [moc_predefs.h, 34](#)

`__DEC128_MIN__`
 [moc_predefs.h, 34](#)

`__DEC128_SUBNORMAL_MIN__`
 [moc_predefs.h, 34](#)

`__DEC32_EPSILON__`
 [moc_predefs.h, 34](#)

`__DEC32_MANT_DIG__`
 [moc_predefs.h, 34](#)

`__DEC32_MAX_EXP__`
 [moc_predefs.h, 34](#)

`__DEC32_MAX__`
 [moc_predefs.h, 34](#)

`__DEC32_MIN_EXP__`
 [moc_predefs.h, 35](#)

`__DEC32_MIN__`
 [moc_predefs.h, 34](#)

`__DEC32_SUBNORMAL_MIN__`
 [moc_predefs.h, 35](#)

`__DEC64_EPSILON__`
 [moc_predefs.h, 35](#)

`__DEC64_MANT_DIG__`
 [moc_predefs.h, 35](#)

`__DEC64_MAX_EXP__`
 [moc_predefs.h, 35](#)

`__DEC64_MAX__`
 [moc_predefs.h, 35](#)

__DEC64_MIN_EXP__	__FLT32X_HAS_DENORM__
moc_predefs.h, 35	moc_predefs.h, 41
__DEC64_MIN__	__FLT32X_HAS_INFINITY__
moc_predefs.h, 35	moc_predefs.h, 41
__DEC64_SUBNORMAL_MIN__	__FLT32X_HAS_QUIET_NAN__
moc_predefs.h, 36	moc_predefs.h, 41
__DECIMAL_BID_FORMAT__	__FLT32X_MANT_DIG__
moc_predefs.h, 36	moc_predefs.h, 41
__DECIMAL_DIG__	__FLT32X_MAX_10_EXP__
moc_predefs.h, 36	moc_predefs.h, 41
__DEC_EVAL_METHOD__	__FLT32X_MAX_EXP__
moc_predefs.h, 36	moc_predefs.h, 41
__DEPRECATED	__FLT32X_MAX__
moc_predefs.h, 36	moc_predefs.h, 41
__ELF__	__FLT32X_MIN_10_EXP__
moc_predefs.h, 36	moc_predefs.h, 42
__EXCEPTIONS	__FLT32X_MIN_EXP__
moc_predefs.h, 36	moc_predefs.h, 42
__FINITE_MATH_ONLY__	__FLT32X_MIN__
moc_predefs.h, 36	moc_predefs.h, 42
__FLOAT_WORD_ORDER__	__FLT32_DECIMAL_DIG__
moc_predefs.h, 37	moc_predefs.h, 38
__FLT128_DECIMAL_DIG__	__FLT32_DENORM_MIN__
moc_predefs.h, 37	moc_predefs.h, 39
__FLT128_DENORM_MIN__	__FLT32_DIG__
moc_predefs.h, 37	moc_predefs.h, 39
__FLT128_DIG__	__FLT32_EPSILON__
moc_predefs.h, 37	moc_predefs.h, 39
__FLT128_EPSILON__	__FLT32_HAS_DENORM__
moc_predefs.h, 37	moc_predefs.h, 39
__FLT128_HAS_DENORM__	__FLT32_HAS_INFINITY__
moc_predefs.h, 37	moc_predefs.h, 39
__FLT128_HAS_INFINITY__	__FLT32_HAS_QUIET_NAN__
moc_predefs.h, 37	moc_predefs.h, 39
__FLT128_HAS_QUIET_NAN__	__FLT32_MANT_DIG__
moc_predefs.h, 37	moc_predefs.h, 39
__FLT128_MANT_DIG__	__FLT32_MAX_10_EXP__
moc_predefs.h, 38	moc_predefs.h, 39
__FLT128_MAX_10_EXP__	__FLT32_MAX_EXP__
moc_predefs.h, 38	moc_predefs.h, 40
__FLT128_MAX_EXP__	__FLT32_MAX__
moc_predefs.h, 38	moc_predefs.h, 40
__FLT128_MAX__	__FLT32_MIN_10_EXP__
moc_predefs.h, 38	moc_predefs.h, 40
__FLT128_MIN_10_EXP__	__FLT32_MIN_EXP__
moc_predefs.h, 38	moc_predefs.h, 40
__FLT128_MIN_EXP__	__FLT32_MIN__
moc_predefs.h, 38	moc_predefs.h, 40
__FLT128_MIN__	__FLT64X_DECIMAL_DIG__
moc_predefs.h, 38	moc_predefs.h, 44
__FLT32X_DECIMAL_DIG__	__FLT64X_DENORM_MIN__
moc_predefs.h, 40	moc_predefs.h, 44
__FLT32X_DENORM_MIN__	__FLT64X_DIG__
moc_predefs.h, 40	moc_predefs.h, 44
__FLT32X_DIG__	__FLT64X_EPSILON__
moc_predefs.h, 40	moc_predefs.h, 44
__FLT32X_EPSILON__	__FLT64X_HAS_DENORM__
moc_predefs.h, 41	moc_predefs.h, 44

- `__FLT64X_HAS_INFINITY__`
 moc_predefs.h, [44](#)
- `__FLT64X_HAS_QUIET_NAN__`
 moc_predefs.h, [44](#)
- `__FLT64X_MANT_DIG__`
 moc_predefs.h, [45](#)
- `__FLT64X_MAX_10_EXP__`
 moc_predefs.h, [45](#)
- `__FLT64X_MAX_EXP__`
 moc_predefs.h, [45](#)
- `__FLT64X_MAX__`
 moc_predefs.h, [45](#)
- `__FLT64X_MIN_10_EXP__`
 moc_predefs.h, [45](#)
- `__FLT64X_MIN_EXP__`
 moc_predefs.h, [45](#)
- `__FLT64X_MIN__`
 moc_predefs.h, [45](#)
- `__FLT64_DECIMAL_DIG__`
 moc_predefs.h, [42](#)
- `__FLT64_DENORM_MIN__`
 moc_predefs.h, [42](#)
- `__FLT64_DIG__`
 moc_predefs.h, [42](#)
- `__FLT64_EPSILON__`
 moc_predefs.h, [42](#)
- `__FLT64_HAS_DENORM__`
 moc_predefs.h, [42](#)
- `__FLT64_HAS_INFINITY__`
 moc_predefs.h, [43](#)
- `__FLT64_HAS_QUIET_NAN__`
 moc_predefs.h, [43](#)
- `__FLT64_MANT_DIG__`
 moc_predefs.h, [43](#)
- `__FLT64_MAX_10_EXP__`
 moc_predefs.h, [43](#)
- `__FLT64_MAX_EXP__`
 moc_predefs.h, [43](#)
- `__FLT64_MAX__`
 moc_predefs.h, [43](#)
- `__FLT64_MIN_10_EXP__`
 moc_predefs.h, [43](#)
- `__FLT64_MIN_EXP__`
 moc_predefs.h, [44](#)
- `__FLT64_MIN__`
 moc_predefs.h, [43](#)
- `__FLT_DECIMAL_DIG__`
 moc_predefs.h, [45](#)
- `__FLT_DENORM_MIN__`
 moc_predefs.h, [46](#)
- `__FLT_DIG__`
 moc_predefs.h, [46](#)
- `__FLT_EPSILON__`
 moc_predefs.h, [46](#)
- `__FLT_EVAL_METHOD_TS_18661_3__`
 moc_predefs.h, [46](#)
- `__FLT_EVAL_METHOD__`
 moc_predefs.h, [46](#)
- `__FLT_HAS_DENORM__`
 moc_predefs.h, [46](#)
- `__FLT_HAS_INFINITY__`
 moc_predefs.h, [46](#)
- `__FLT_HAS_QUIET_NAN__`
 moc_predefs.h, [46](#)
- `__FLT_MANT_DIG__`
 moc_predefs.h, [47](#)
- `__FLT_MAX_10_EXP__`
 moc_predefs.h, [47](#)
- `__FLT_MAX_EXP__`
 moc_predefs.h, [47](#)
- `__FLT_MAX__`
 moc_predefs.h, [47](#)
- `__FLT_MIN_10_EXP__`
 moc_predefs.h, [47](#)
- `__FLT_MIN_EXP__`
 moc_predefs.h, [47](#)
- `__FLT_MIN__`
 moc_predefs.h, [47](#)
- `__FLT_RADIX__`
 moc_predefs.h, [47](#)
- `__FXSR__`
 moc_predefs.h, [48](#)
- `__GCC_ASM_FLAG_OUTPUTS__`
 moc_predefs.h, [48](#)
- `__GCC_ATOMIC_BOOL_LOCK_FREE`
 moc_predefs.h, [48](#)
- `__GCC_ATOMIC_CHAR16_T_LOCK_FREE`
 moc_predefs.h, [48](#)
- `__GCC_ATOMIC_CHAR32_T_LOCK_FREE`
 moc_predefs.h, [48](#)
- `__GCC_ATOMIC_CHAR_LOCK_FREE`
 moc_predefs.h, [48](#)
- `__GCC_ATOMIC_INT_LOCK_FREE`
 moc_predefs.h, [48](#)
- `__GCC_ATOMIC_LLONG_LOCK_FREE`
 moc_predefs.h, [48](#)
- `__GCC_ATOMIC_LONG_LOCK_FREE`
 moc_predefs.h, [49](#)
- `__GCC_ATOMIC_POINTER_LOCK_FREE`
 moc_predefs.h, [49](#)
- `__GCC_ATOMIC_SHORT_LOCK_FREE`
 moc_predefs.h, [49](#)
- `__GCC_ATOMIC_TEST_AND_SET_TRUEVAL`
 moc_predefs.h, [49](#)
- `__GCC_ATOMIC_WCHAR_T_LOCK_FREE`
 moc_predefs.h, [49](#)
- `__GCC_HAVE_DWARF2_CFI_ASM`
 moc_predefs.h, [49](#)
- `__GCC_HAVE_SYNC_COMPARE_AND_SWAP_1`
 moc_predefs.h, [49](#)
- `__GCC_HAVE_SYNC_COMPARE_AND_SWAP_2`
 moc_predefs.h, [49](#)
- `__GCC_HAVE_SYNC_COMPARE_AND_SWAP_4`
 moc_predefs.h, [50](#)
- `__GCC_HAVE_SYNC_COMPARE_AND_SWAP_8`
 moc_predefs.h, [50](#)

__GCC_IEC_559	__INTPTR_MAX__
moc_predefs.h, 50	moc_predefs.h, 57
__GCC_IEC_559_COMPLEX	__INTPTR_TYPE__
moc_predefs.h, 50	moc_predefs.h, 58
__GLIBCXX_BITSIZE_INT_N_0	__INTPTR_WIDTH__
moc_predefs.h, 50	moc_predefs.h, 58
__GLIBCXX_TYPE_INT_N_0	__INT_FAST16_MAX__
moc_predefs.h, 50	moc_predefs.h, 54
__GNUC_MINOR__	__INT_FAST16_TYPE__
moc_predefs.h, 51	moc_predefs.h, 54
__GNUC_PATCHLEVEL__	__INT_FAST16_WIDTH__
moc_predefs.h, 51	moc_predefs.h, 54
__GNUC_STDC_INLINE__	__INT_FAST32_MAX__
moc_predefs.h, 51	moc_predefs.h, 54
__GNUC__	__INT_FAST32_TYPE__
moc_predefs.h, 50	moc_predefs.h, 54
__GNUG__	__INT_FAST32_WIDTH__
moc_predefs.h, 51	moc_predefs.h, 54
__GXX_ABI_VERSION	__INT_FAST64_MAX__
moc_predefs.h, 51	moc_predefs.h, 54
__GXX_EXPERIMENTAL_CXX0X__	__INT_FAST64_TYPE__
moc_predefs.h, 51	moc_predefs.h, 54
__GXX_RTTI	__INT_FAST64_WIDTH__
moc_predefs.h, 51	moc_predefs.h, 55
__GXX_WEAK__	__INT_FAST8_MAX__
moc_predefs.h, 51	moc_predefs.h, 55
__INT16_C	__INT_FAST8_TYPE__
moc_predefs.h, 52	moc_predefs.h, 55
__INT16_MAX__	__INT_FAST8_WIDTH__
moc_predefs.h, 52	moc_predefs.h, 55
__INT16_TYPE__	__INT_LEAST16_MAX__
moc_predefs.h, 52	moc_predefs.h, 55
__INT32_C	__INT_LEAST16_TYPE__
moc_predefs.h, 52	moc_predefs.h, 55
__INT32_MAX__	__INT_LEAST16_WIDTH__
moc_predefs.h, 52	moc_predefs.h, 55
__INT32_TYPE__	__INT_LEAST32_MAX__
moc_predefs.h, 53	moc_predefs.h, 55
__INT64_C	__INT_LEAST32_TYPE__
moc_predefs.h, 53	moc_predefs.h, 56
__INT64_MAX__	__INT_LEAST32_WIDTH__
moc_predefs.h, 53	moc_predefs.h, 56
__INT64_TYPE__	__INT_LEAST64_MAX__
moc_predefs.h, 53	moc_predefs.h, 56
__INT8_C	__INT_LEAST64_TYPE__
moc_predefs.h, 53	moc_predefs.h, 56
__INT8_MAX__	__INT_LEAST64_WIDTH__
moc_predefs.h, 53	moc_predefs.h, 56
__INT8_TYPE__	__INT_LEAST8_MAX__
moc_predefs.h, 53	moc_predefs.h, 56
__INTMAX_C	__INT_LEAST8_TYPE__
moc_predefs.h, 57	moc_predefs.h, 56
__INTMAX_MAX__	__INT_LEAST8_WIDTH__
moc_predefs.h, 57	moc_predefs.h, 56
__INTMAX_TYPE__	__INT_MAX__
moc_predefs.h, 57	moc_predefs.h, 57
__INTMAX_WIDTH__	__INT_WIDTH__
moc_predefs.h, 57	moc_predefs.h, 57

- `__LDBL_DECIMAL_DIG__`
moc_predefs.h, [58](#)
- `__LDBL_DENORM_MIN__`
moc_predefs.h, [58](#)
- `__LDBL_DIG__`
moc_predefs.h, [58](#)
- `__LDBL_EPSILON__`
moc_predefs.h, [58](#)
- `__LDBL_HAS_DENORM__`
moc_predefs.h, [59](#)
- `__LDBL_HAS_INFINITY__`
moc_predefs.h, [59](#)
- `__LDBL_HAS_QUIET_NAN__`
moc_predefs.h, [59](#)
- `__LDBL_MANT_DIG__`
moc_predefs.h, [59](#)
- `__LDBL_MAX_10_EXP__`
moc_predefs.h, [59](#)
- `__LDBL_MAX_EXP__`
moc_predefs.h, [59](#)
- `__LDBL_MAX__`
moc_predefs.h, [59](#)
- `__LDBL_MIN_10_EXP__`
moc_predefs.h, [59](#)
- `__LDBL_MIN_EXP__`
moc_predefs.h, [60](#)
- `__LDBL_MIN__`
moc_predefs.h, [60](#)
- `__LONG_LONG_MAX__`
moc_predefs.h, [60](#)
- `__LONG_LONG_WIDTH__`
moc_predefs.h, [60](#)
- `__LONG_MAX__`
moc_predefs.h, [60](#)
- `__LONG_WIDTH__`
moc_predefs.h, [60](#)
- `__LP64__`
moc_predefs.h, [61](#)
- `__MMX__`
moc_predefs.h, [61](#)
- `__OPTIMIZE__`
moc_predefs.h, [61](#)
- `__ORDER_BIG_ENDIAN__`
moc_predefs.h, [61](#)
- `__ORDER_LITTLE_ENDIAN__`
moc_predefs.h, [61](#)
- `__ORDER_PDP_ENDIAN__`
moc_predefs.h, [61](#)
- `__PIC__`
moc_predefs.h, [61](#)
- `__PIE__`
moc_predefs.h, [62](#)
- `__PRAGMA_REDEFINE_EXTNAME`
moc_predefs.h, [62](#)
- `__PTRDIFF_MAX__`
moc_predefs.h, [62](#)
- `__PTRDIFF_TYPE__`
moc_predefs.h, [62](#)
- `__PTRDIFF_WIDTH__`
moc_predefs.h, [62](#)
- `__REGISTER_PREFIX__`
moc_predefs.h, [62](#)
- `__SCHAR_MAX__`
moc_predefs.h, [62](#)
- `__SCHAR_WIDTH__`
moc_predefs.h, [63](#)
- `__SEG_FS`
moc_predefs.h, [63](#)
- `__SEG_GS`
moc_predefs.h, [63](#)
- `__SHRT_MAX__`
moc_predefs.h, [63](#)
- `__SHRT_WIDTH__`
moc_predefs.h, [63](#)
- `__SIG_ATOMIC_MAX__`
moc_predefs.h, [63](#)
- `__SIG_ATOMIC_MIN__`
moc_predefs.h, [63](#)
- `__SIG_ATOMIC_TYPE__`
moc_predefs.h, [63](#)
- `__SIG_ATOMIC_WIDTH__`
moc_predefs.h, [64](#)
- `__SIZEOF_DOUBLE__`
moc_predefs.h, [64](#)
- `__SIZEOF_FLOAT128__`
moc_predefs.h, [64](#)
- `__SIZEOF_FLOAT80__`
moc_predefs.h, [64](#)
- `__SIZEOF_FLOAT__`
moc_predefs.h, [64](#)
- `__SIZEOF_INT128__`
moc_predefs.h, [65](#)
- `__SIZEOF_INT__`
moc_predefs.h, [65](#)
- `__SIZEOF_LONG_DOUBLE__`
moc_predefs.h, [65](#)
- `__SIZEOF_LONG_LONG__`
moc_predefs.h, [65](#)
- `__SIZEOF_LONG__`
moc_predefs.h, [65](#)
- `__SIZEOF_POINTER__`
moc_predefs.h, [65](#)
- `__SIZEOF_PTRDIFF_T__`
moc_predefs.h, [65](#)
- `__SIZEOF_SHORT__`
moc_predefs.h, [65](#)
- `__SIZEOF_SIZE_T__`
moc_predefs.h, [66](#)
- `__SIZEOF_WCHAR_T__`
moc_predefs.h, [66](#)
- `__SIZEOF_WINT_T__`
moc_predefs.h, [66](#)
- `__SIZE_MAX__`
moc_predefs.h, [64](#)
- `__SIZE_TYPE__`
moc_predefs.h, [64](#)

<code>__SIZE_WIDTH__</code>	<code>__UINTPTR_MAX__</code>
moc_predefs.h, 64	moc_predefs.h, 72
<code>__SSE2_MATH__</code>	<code>__UINTPTR_TYPE__</code>
moc_predefs.h, 66	moc_predefs.h, 72
<code>__SSE2__</code>	<code>__UINT_FAST16_MAX__</code>
moc_predefs.h, 66	moc_predefs.h, 69
<code>__SSE_MATH__</code>	<code>__UINT_FAST16_TYPE__</code>
moc_predefs.h, 66	moc_predefs.h, 69
<code>__SSE__</code>	<code>__UINT_FAST32_MAX__</code>
moc_predefs.h, 66	moc_predefs.h, 70
<code>__SSP_STRONG__</code>	<code>__UINT_FAST32_TYPE__</code>
moc_predefs.h, 66	moc_predefs.h, 70
<code>__STDC_HOSTED__</code>	<code>__UINT_FAST64_MAX__</code>
moc_predefs.h, 67	moc_predefs.h, 70
<code>__STDC_IEC_559_COMPLEX__</code>	<code>__UINT_FAST64_TYPE__</code>
moc_predefs.h, 67	moc_predefs.h, 70
<code>__STDC_IEC_559__</code>	<code>__UINT_FAST8_MAX__</code>
moc_predefs.h, 67	moc_predefs.h, 70
<code>__STDC_ISO_10646__</code>	<code>__UINT_FAST8_TYPE__</code>
moc_predefs.h, 67	moc_predefs.h, 70
<code>__STDC_NO_THREADS__</code>	<code>__UINT_LEAST16_MAX__</code>
moc_predefs.h, 67	moc_predefs.h, 70
<code>__STDC_UTF_16__</code>	<code>__UINT_LEAST16_TYPE__</code>
moc_predefs.h, 67	moc_predefs.h, 70
<code>__STDC_UTF_32__</code>	<code>__UINT_LEAST32_MAX__</code>
moc_predefs.h, 67	moc_predefs.h, 71
<code>__STDC__</code>	<code>__UINT_LEAST32_TYPE__</code>
moc_predefs.h, 67	moc_predefs.h, 71
<code>__UINT16_C</code>	<code>__UINT_LEAST64_MAX__</code>
moc_predefs.h, 68	moc_predefs.h, 71
<code>__UINT16_MAX__</code>	<code>__UINT_LEAST64_TYPE__</code>
moc_predefs.h, 68	moc_predefs.h, 71
<code>__UINT16_TYPE__</code>	<code>__UINT_LEAST8_MAX__</code>
moc_predefs.h, 68	moc_predefs.h, 71
<code>__UINT32_C</code>	<code>__UINT_LEAST8_TYPE__</code>
moc_predefs.h, 68	moc_predefs.h, 71
<code>__UINT32_MAX__</code>	<code>__USER_LABEL_PREFIX__</code>
moc_predefs.h, 68	moc_predefs.h, 72
<code>__UINT32_TYPE__</code>	<code>__VERSION__</code>
moc_predefs.h, 68	moc_predefs.h, 72
<code>__UINT64_C</code>	<code>__WCHAR_MAX__</code>
moc_predefs.h, 68	moc_predefs.h, 72
<code>__UINT64_MAX__</code>	<code>__WCHAR_MIN__</code>
moc_predefs.h, 69	moc_predefs.h, 73
<code>__UINT64_TYPE__</code>	<code>__WCHAR_TYPE__</code>
moc_predefs.h, 69	moc_predefs.h, 73
<code>__UINT8_C</code>	<code>__WCHAR_WIDTH__</code>
moc_predefs.h, 69	moc_predefs.h, 73
<code>__UINT8_MAX__</code>	<code>__WINT_MAX__</code>
moc_predefs.h, 69	moc_predefs.h, 73
<code>__UINT8_TYPE__</code>	<code>__WINT_MIN__</code>
moc_predefs.h, 69	moc_predefs.h, 73
<code>__UINTMAX_C</code>	<code>__WINT_TYPE__</code>
moc_predefs.h, 71	moc_predefs.h, 73
<code>__UINTMAX_MAX__</code>	<code>__WINT_WIDTH__</code>
moc_predefs.h, 71	moc_predefs.h, 73
<code>__UINTMAX_TYPE__</code>	<code>__amd64</code>
moc_predefs.h, 72	moc_predefs.h, 25

- __amd64__
 - moc_predefs.h, 25
- __code_model_small__
 - moc_predefs.h, 27
- __cplusplus
 - moc_predefs.h, 27
- __cpp_aggregate_nsdmi
 - moc_predefs.h, 27
- __cpp_alias_templates
 - moc_predefs.h, 27
- __cpp_attributes
 - moc_predefs.h, 27
- __cpp_binary_literals
 - moc_predefs.h, 28
- __cpp_constexpr
 - moc_predefs.h, 28
- __cpp_decltype
 - moc_predefs.h, 28
- __cpp_decltype_auto
 - moc_predefs.h, 28
- __cpp_delegating_constructors
 - moc_predefs.h, 28
- __cpp_digit_separators
 - moc_predefs.h, 28
- __cpp_exceptions
 - moc_predefs.h, 28
- __cpp_generic_lambdas
 - moc_predefs.h, 28
- __cpp_hex_float
 - moc_predefs.h, 29
- __cpp_inheriting_constructors
 - moc_predefs.h, 29
- __cpp_init_captures
 - moc_predefs.h, 29
- __cpp_initializer_lists
 - moc_predefs.h, 29
- __cpp_lambdas
 - moc_predefs.h, 29
- __cpp_nsdmi
 - moc_predefs.h, 29
- __cpp_range_based_for
 - moc_predefs.h, 29
- __cpp_raw_strings
 - moc_predefs.h, 29
- __cpp_ref_qualifiers
 - moc_predefs.h, 30
- __cpp_return_type_deduction
 - moc_predefs.h, 30
- __cpp_rtti
 - moc_predefs.h, 30
- __cpp_runtime_arrays
 - moc_predefs.h, 30
- __cpp_rvalue_reference
 - moc_predefs.h, 30
- __cpp_rvalue_references
 - moc_predefs.h, 30
- __cpp_sized_deallocation
 - moc_predefs.h, 30
- __cpp_static_assert
 - moc_predefs.h, 30
- __cpp_threadsafe_static_init
 - moc_predefs.h, 31
- __cpp_unicode_characters
 - moc_predefs.h, 31
- __cpp_unicode_literals
 - moc_predefs.h, 31
- __cpp_user_defined_literals
 - moc_predefs.h, 31
- __cpp_variable_templates
 - moc_predefs.h, 31
- __cpp_variadic_templates
 - moc_predefs.h, 31
- __gnu_linux__
 - moc_predefs.h, 50
- __has_include
 - moc_predefs.h, 52
- __has_include_next
 - moc_predefs.h, 52
- __k8
 - moc_predefs.h, 58
- __k8__
 - moc_predefs.h, 58
- __linux
 - moc_predefs.h, 60
- __linux__
 - moc_predefs.h, 60
- __pic__
 - moc_predefs.h, 61
- __pie__
 - moc_predefs.h, 62
- __unix
 - moc_predefs.h, 72
- __unix__
 - moc_predefs.h, 72
- __x86_64
 - moc_predefs.h, 73
- __x86_64__
 - moc_predefs.h, 74
- ~AntThread
 - AntThread, 9
- ~AntWin
 - AntWin, 11
- Ant, 7
 - Ant, 7
 - dir, 7
 - x, 8
 - y, 8
- ant.h, 15
- AntThread, 8
 - ~AntThread, 9
 - AntThread, 8
 - finish, 9
 - isRunning, 9
 - pause, 9
 - run, 9
 - step, 9

- antThread
 - AntWin, [12](#)
- AntWin, [10](#)
 - ~AntWin, [11](#)
 - antThread, [12](#)
 - AntWin, [10](#)
 - closeEvent, [11](#)
 - keyPressEvent, [11](#)
 - paintEvent, [11](#)
 - step, [11](#)
- antthread.cpp, [15](#)
- antthread.h, [15](#)
- antwin.cpp, [15](#)
- antwin.h, [16](#)
- closeEvent
 - AntWin, [11](#)
- data
 - qt_meta_stringdata_AntThread_t, [12](#)
 - qt_meta_stringdata_AntWin_t, [13](#)
- dir
 - Ant, [7](#)
- finish
 - AntThread, [9](#)
- isRunnung
 - AntThread, [9](#)
- keyPressEvent
 - AntWin, [11](#)
- linux
 - moc_predefs.h, [74](#)
- main
 - main.cpp, [16](#)
- main.cpp, [16](#)
 - main, [16](#)
- moc_antthread.cpp, [16](#)
 - QT_MOC_LITERAL, [17](#)
- moc_antwin.cpp, [17](#)
 - QT_MOC_LITERAL, [17](#)
- moc_predefs.h, [18](#)
 - _FORTIFY_SOURCE, [74](#)
 - _GNU_SOURCE, [74](#)
 - _LP64, [74](#)
 - _STDC_PREDEF_H, [74](#)
 - _ATOMIC_ACQUIRE, [25](#)
 - _ATOMIC_ACQ_REL, [25](#)
 - _ATOMIC_CONSUME, [26](#)
 - _ATOMIC_HLE_ACQUIRE, [26](#)
 - _ATOMIC_HLE_RELEASE, [26](#)
 - _ATOMIC_RELAXED, [26](#)
 - _ATOMIC_RELEASE, [26](#)
 - _ATOMIC_SEQ_CST, [26](#)
 - _BIGGEST_ALIGNMENT__, [26](#)
 - _BYTE_ORDER__, [26](#)
 - _CHAR16_TYPE__, [27](#)
 - _CHAR32_TYPE__, [27](#)
 - _CHAR_BIT__, [27](#)
 - _DBL_DECIMAL_DIG__, [31](#)
 - _DBL_DENORM_MIN__, [31](#)
 - _DBL_DIG__, [32](#)
 - _DBL_EPSILON__, [32](#)
 - _DBL_HAS_DENORM__, [32](#)
 - _DBL_HAS_INFINITY__, [32](#)
 - _DBL_HAS_QUIET_NAN__, [32](#)
 - _DBL_MANT_DIG__, [32](#)
 - _DBL_MAX_10_EXP__, [32](#)
 - _DBL_MAX_EXP__, [33](#)
 - _DBL_MAX__, [32](#)
 - _DBL_MIN_10_EXP__, [33](#)
 - _DBL_MIN_EXP__, [33](#)
 - _DBL_MIN__, [33](#)
 - _DEC128_EPSILON__, [33](#)
 - _DEC128_MANT_DIG__, [33](#)
 - _DEC128_MAX_EXP__, [33](#)
 - _DEC128_MAX__, [33](#)
 - _DEC128_MIN_EXP__, [34](#)
 - _DEC128_MIN__, [34](#)
 - _DEC128_SUBNORMAL_MIN__, [34](#)
 - _DEC32_EPSILON__, [34](#)
 - _DEC32_MANT_DIG__, [34](#)
 - _DEC32_MAX_EXP__, [34](#)
 - _DEC32_MAX__, [34](#)
 - _DEC32_MIN_EXP__, [35](#)
 - _DEC32_MIN__, [34](#)
 - _DEC32_SUBNORMAL_MIN__, [35](#)
 - _DEC64_EPSILON__, [35](#)
 - _DEC64_MANT_DIG__, [35](#)
 - _DEC64_MAX_EXP__, [35](#)
 - _DEC64_MAX__, [35](#)
 - _DEC64_MIN_EXP__, [35](#)
 - _DEC64_MIN__, [35](#)
 - _DEC64_SUBNORMAL_MIN__, [36](#)
 - _DECIMAL_BID_FORMAT__, [36](#)
 - _DECIMAL_DIG__, [36](#)
 - _DEC_EVAL_METHOD__, [36](#)
 - _DEPRECATED, [36](#)
 - _ELF__, [36](#)
 - _EXCEPTIONS, [36](#)
 - _FINITE_MATH_ONLY__, [36](#)
 - _FLOAT_WORD_ORDER__, [37](#)
 - _FLT128_DECIMAL_DIG__, [37](#)
 - _FLT128_DENORM_MIN__, [37](#)
 - _FLT128_DIG__, [37](#)
 - _FLT128_EPSILON__, [37](#)
 - _FLT128_HAS_DENORM__, [37](#)
 - _FLT128_HAS_INFINITY__, [37](#)
 - _FLT128_HAS_QUIET_NAN__, [37](#)
 - _FLT128_MANT_DIG__, [38](#)
 - _FLT128_MAX_10_EXP__, [38](#)
 - _FLT128_MAX_EXP__, [38](#)
 - _FLT128_MAX__, [38](#)
 - _FLT128_MIN_10_EXP__, [38](#)
 - _FLT128_MIN_EXP__, [38](#)

- [__FLT128_MIN__, 38](#)
- [__FLT32X_DECIMAL_DIG__, 40](#)
- [__FLT32X_DENORM_MIN__, 40](#)
- [__FLT32X_DIG__, 40](#)
- [__FLT32X_EPSILON__, 41](#)
- [__FLT32X_HAS_DENORM__, 41](#)
- [__FLT32X_HAS_INFINITY__, 41](#)
- [__FLT32X_HAS_QUIET_NAN__, 41](#)
- [__FLT32X_MANT_DIG__, 41](#)
- [__FLT32X_MAX_10_EXP__, 41](#)
- [__FLT32X_MAX_EXP__, 41](#)
- [__FLT32X_MAX__, 41](#)
- [__FLT32X_MIN_10_EXP__, 42](#)
- [__FLT32X_MIN_EXP__, 42](#)
- [__FLT32X_MIN__, 42](#)
- [__FLT32_DECIMAL_DIG__, 38](#)
- [__FLT32_DENORM_MIN__, 39](#)
- [__FLT32_DIG__, 39](#)
- [__FLT32_EPSILON__, 39](#)
- [__FLT32_HAS_DENORM__, 39](#)
- [__FLT32_HAS_INFINITY__, 39](#)
- [__FLT32_HAS_QUIET_NAN__, 39](#)
- [__FLT32_MANT_DIG__, 39](#)
- [__FLT32_MAX_10_EXP__, 39](#)
- [__FLT32_MAX_EXP__, 40](#)
- [__FLT32_MAX__, 40](#)
- [__FLT32_MIN_10_EXP__, 40](#)
- [__FLT32_MIN_EXP__, 40](#)
- [__FLT32_MIN__, 40](#)
- [__FLT64X_DECIMAL_DIG__, 44](#)
- [__FLT64X_DENORM_MIN__, 44](#)
- [__FLT64X_DIG__, 44](#)
- [__FLT64X_EPSILON__, 44](#)
- [__FLT64X_HAS_DENORM__, 44](#)
- [__FLT64X_HAS_INFINITY__, 44](#)
- [__FLT64X_HAS_QUIET_NAN__, 44](#)
- [__FLT64X_MANT_DIG__, 45](#)
- [__FLT64X_MAX_10_EXP__, 45](#)
- [__FLT64X_MAX_EXP__, 45](#)
- [__FLT64X_MAX__, 45](#)
- [__FLT64X_MIN_10_EXP__, 45](#)
- [__FLT64X_MIN_EXP__, 45](#)
- [__FLT64X_MIN__, 45](#)
- [__FLT64_DECIMAL_DIG__, 42](#)
- [__FLT64_DENORM_MIN__, 42](#)
- [__FLT64_DIG__, 42](#)
- [__FLT64_EPSILON__, 42](#)
- [__FLT64_HAS_DENORM__, 42](#)
- [__FLT64_HAS_INFINITY__, 43](#)
- [__FLT64_HAS_QUIET_NAN__, 43](#)
- [__FLT64_MANT_DIG__, 43](#)
- [__FLT64_MAX_10_EXP__, 43](#)
- [__FLT64_MAX_EXP__, 43](#)
- [__FLT64_MAX__, 43](#)
- [__FLT64_MIN_10_EXP__, 43](#)
- [__FLT64_MIN_EXP__, 44](#)
- [__FLT64_MIN__, 43](#)
- [__FLT_DECIMAL_DIG__, 45](#)
- [__FLT_DENORM_MIN__, 46](#)
- [__FLT_DIG__, 46](#)
- [__FLT_EPSILON__, 46](#)
- [__FLT_EVAL_METHOD_TS_18661_3__, 46](#)
- [__FLT_EVAL_METHOD__, 46](#)
- [__FLT_HAS_DENORM__, 46](#)
- [__FLT_HAS_INFINITY__, 46](#)
- [__FLT_HAS_QUIET_NAN__, 46](#)
- [__FLT_MANT_DIG__, 47](#)
- [__FLT_MAX_10_EXP__, 47](#)
- [__FLT_MAX_EXP__, 47](#)
- [__FLT_MAX__, 47](#)
- [__FLT_MIN_10_EXP__, 47](#)
- [__FLT_MIN_EXP__, 47](#)
- [__FLT_MIN__, 47](#)
- [__FLT_RADIX__, 47](#)
- [FXSR__, 48](#)
- [GCC_ASM_FLAG_OUTPUTS__, 48](#)
- [GCC_ATOMIC_BOOL_LOCK_FREE, 48](#)
- [GCC_ATOMIC_CHAR16_T_LOCK_FREE, 48](#)
- [GCC_ATOMIC_CHAR32_T_LOCK_FREE, 48](#)
- [GCC_ATOMIC_CHAR_LOCK_FREE, 48](#)
- [GCC_ATOMIC_INT_LOCK_FREE, 48](#)
- [GCC_ATOMIC_LLONG_LOCK_FREE, 48](#)
- [GCC_ATOMIC_LONG_LOCK_FREE, 49](#)
- [GCC_ATOMIC_POINTER_LOCK_FREE, 49](#)
- [GCC_ATOMIC_SHORT_LOCK_FREE, 49](#)
- [GCC_ATOMIC_TEST_AND_SET_TRUEVAL, 49](#)
- [GCC_ATOMIC_WCHAR_T_LOCK_FREE, 49](#)
- [GCC_HAVE_DWARF2_CFI_ASM, 49](#)
- [GCC_HAVE_SYNC_COMPARE_AND_SWAP_1, 49](#)
- [GCC_HAVE_SYNC_COMPARE_AND_SWAP_2, 49](#)
- [GCC_HAVE_SYNC_COMPARE_AND_SWAP_4, 50](#)
- [GCC_HAVE_SYNC_COMPARE_AND_SWAP_8, 50](#)
- [GCC_IEC_559, 50](#)
- [GCC_IEC_559_COMPLEX, 50](#)
- [GLIBCXX_BITSIZE_INT_N_0, 50](#)
- [GLIBCXX_TYPE_INT_N_0, 50](#)
- [GNUC_MINOR__, 51](#)
- [GNUC_PATCHLEVEL__, 51](#)
- [GNUC_STDC_INLINE__, 51](#)
- [GNUC__, 50](#)
- [GNUG__, 51](#)
- [GXX_ABI_VERSION, 51](#)
- [GXX_EXPERIMENTAL_CXX0X__, 51](#)
- [GXX_RTTI, 51](#)
- [GXX_WEAK__, 51](#)
- [__INT16_C__, 52](#)
- [__INT16_MAX__, 52](#)
- [__INT16_TYPE__, 52](#)
- [__INT32_C__, 52](#)
- [__INT32_MAX__, 52](#)
- [__INT32_TYPE__, 53](#)

- [__INT64_C](#), 53
- [__INT64_MAX](#), 53
- [__INT64_TYPE](#), 53
- [__INT8_C](#), 53
- [__INT8_MAX](#), 53
- [__INT8_TYPE](#), 53
- [__INTMAX_C](#), 57
- [__INTMAX_MAX](#), 57
- [__INTMAX_TYPE](#), 57
- [__INTMAX_WIDTH](#), 57
- [__INTPTR_MAX](#), 57
- [__INTPTR_TYPE](#), 58
- [__INTPTR_WIDTH](#), 58
- [__INT_FAST16_MAX](#), 54
- [__INT_FAST16_TYPE](#), 54
- [__INT_FAST16_WIDTH](#), 54
- [__INT_FAST32_MAX](#), 54
- [__INT_FAST32_TYPE](#), 54
- [__INT_FAST32_WIDTH](#), 54
- [__INT_FAST64_MAX](#), 54
- [__INT_FAST64_TYPE](#), 54
- [__INT_FAST64_WIDTH](#), 55
- [__INT_FAST8_MAX](#), 55
- [__INT_FAST8_TYPE](#), 55
- [__INT_FAST8_WIDTH](#), 55
- [__INT_LEAST16_MAX](#), 55
- [__INT_LEAST16_TYPE](#), 55
- [__INT_LEAST16_WIDTH](#), 55
- [__INT_LEAST32_MAX](#), 55
- [__INT_LEAST32_TYPE](#), 56
- [__INT_LEAST32_WIDTH](#), 56
- [__INT_LEAST64_MAX](#), 56
- [__INT_LEAST64_TYPE](#), 56
- [__INT_LEAST64_WIDTH](#), 56
- [__INT_LEAST8_MAX](#), 56
- [__INT_LEAST8_TYPE](#), 56
- [__INT_LEAST8_WIDTH](#), 56
- [__INT_MAX](#), 57
- [__INT_WIDTH](#), 57
- [__LDBL_DECIMAL_DIG](#), 58
- [__LDBL_DENORM_MIN](#), 58
- [__LDBL_DIG](#), 58
- [__LDBL_EPSILON](#), 58
- [__LDBL_HAS_DENORM](#), 59
- [__LDBL_HAS_INFINITY](#), 59
- [__LDBL_HAS_QUIET_NAN](#), 59
- [__LDBL_MANT_DIG](#), 59
- [__LDBL_MAX_10_EXP](#), 59
- [__LDBL_MAX_EXP](#), 59
- [__LDBL_MAX](#), 59
- [__LDBL_MIN_10_EXP](#), 59
- [__LDBL_MIN_EXP](#), 60
- [__LDBL_MIN](#), 60
- [__LONG_LONG_MAX](#), 60
- [__LONG_LONG_WIDTH](#), 60
- [__LONG_MAX](#), 60
- [__LONG_WIDTH](#), 60
- [__LP64](#), 61
- [__MMX](#), 61
- [__OPTIMIZE](#), 61
- [__ORDER_BIG_ENDIAN](#), 61
- [__ORDER_LITTLE_ENDIAN](#), 61
- [__ORDER_PDP_ENDIAN](#), 61
- [__PIC](#), 61
- [__PIE](#), 62
- [__PRAGMA_REDEFINE_EXTNAME](#), 62
- [__PTRDIFF_MAX](#), 62
- [__PTRDIFF_TYPE](#), 62
- [__PTRDIFF_WIDTH](#), 62
- [__REGISTER_PREFIX](#), 62
- [__SCHAR_MAX](#), 62
- [__SCHAR_WIDTH](#), 63
- [__SEG_FS](#), 63
- [__SEG_GS](#), 63
- [__SHRT_MAX](#), 63
- [__SHRT_WIDTH](#), 63
- [__SIG_ATOMIC_MAX](#), 63
- [__SIG_ATOMIC_MIN](#), 63
- [__SIG_ATOMIC_TYPE](#), 63
- [__SIG_ATOMIC_WIDTH](#), 64
- [__SIZEOF_DOUBLE](#), 64
- [__SIZEOF_FLOAT128](#), 64
- [__SIZEOF_FLOAT80](#), 64
- [__SIZEOF_FLOAT](#), 64
- [__SIZEOF_INT128](#), 65
- [__SIZEOF_INT](#), 65
- [__SIZEOF_LONG_DOUBLE](#), 65
- [__SIZEOF_LONG_LONG](#), 65
- [__SIZEOF_LONG](#), 65
- [__SIZEOF_POINTER](#), 65
- [__SIZEOF_PTRDIFF_T](#), 65
- [__SIZEOF_SHORT](#), 65
- [__SIZEOF_SIZE_T](#), 66
- [__SIZEOF_WCHAR_T](#), 66
- [__SIZEOF_WINT_T](#), 66
- [__SIZE_MAX](#), 64
- [__SIZE_TYPE](#), 64
- [__SIZE_WIDTH](#), 64
- [__SSE2_MATH](#), 66
- [__SSE2](#), 66
- [__SSE_MATH](#), 66
- [__SSE](#), 66
- [__SSP_STRONG](#), 66
- [__STDC_HOSTED](#), 67
- [__STDC_IEC_559_COMPLEX](#), 67
- [__STDC_IEC_559](#), 67
- [__STDC_ISO_10646](#), 67
- [__STDC_NO_THREADS](#), 67
- [__STDC_UTF_16](#), 67
- [__STDC_UTF_32](#), 67
- [__STDC](#), 67
- [__UINT16_C](#), 68
- [__UINT16_MAX](#), 68
- [__UINT16_TYPE](#), 68
- [__UINT32_C](#), 68
- [__UINT32_MAX](#), 68

- [__UINT32_TYPE__, 68](#)
- [__UINT64_C, 68](#)
- [__UINT64_MAX__, 69](#)
- [__UINT64_TYPE__, 69](#)
- [__UINT8_C, 69](#)
- [__UINT8_MAX__, 69](#)
- [__UINT8_TYPE__, 69](#)
- [__UINTMAX_C, 71](#)
- [__UINTMAX_MAX__, 71](#)
- [__UINTMAX_TYPE__, 72](#)
- [__UINTPTR_MAX__, 72](#)
- [__UINTPTR_TYPE__, 72](#)
- [__UINT_FAST16_MAX__, 69](#)
- [__UINT_FAST16_TYPE__, 69](#)
- [__UINT_FAST32_MAX__, 70](#)
- [__UINT_FAST32_TYPE__, 70](#)
- [__UINT_FAST64_MAX__, 70](#)
- [__UINT_FAST64_TYPE__, 70](#)
- [__UINT_FAST8_MAX__, 70](#)
- [__UINT_FAST8_TYPE__, 70](#)
- [__UINT_LEAST16_MAX__, 70](#)
- [__UINT_LEAST16_TYPE__, 70](#)
- [__UINT_LEAST32_MAX__, 71](#)
- [__UINT_LEAST32_TYPE__, 71](#)
- [__UINT_LEAST64_MAX__, 71](#)
- [__UINT_LEAST64_TYPE__, 71](#)
- [__UINT_LEAST8_MAX__, 71](#)
- [__UINT_LEAST8_TYPE__, 71](#)
- [__USER_LABEL_PREFIX__, 72](#)
- [__VERSION__, 72](#)
- [__WCHAR_MAX__, 72](#)
- [__WCHAR_MIN__, 73](#)
- [__WCHAR_TYPE__, 73](#)
- [__WCHAR_WIDTH__, 73](#)
- [__WINT_MAX__, 73](#)
- [__WINT_MIN__, 73](#)
- [__WINT_TYPE__, 73](#)
- [__WINT_WIDTH__, 73](#)
- [__amd64, 25](#)
- [__amd64__, 25](#)
- [__code_model_small__, 27](#)
- [__cplusplus, 27](#)
- [__cpp_aggregate_nsdmi, 27](#)
- [__cpp_alias_templates, 27](#)
- [__cpp_attributes, 27](#)
- [__cpp_binary_literals, 28](#)
- [__cpp_constexpr, 28](#)
- [__cpp_decltype, 28](#)
- [__cpp_decltype_auto, 28](#)
- [__cpp_delegating_constructors, 28](#)
- [__cpp_digit_separators, 28](#)
- [__cpp_exceptions, 28](#)
- [__cpp_generic_lambdas, 28](#)
- [__cpp_hex_float, 29](#)
- [__cpp_inheriting_constructors, 29](#)
- [__cpp_init_captures, 29](#)
- [__cpp_initializer_lists, 29](#)
- [__cpp_lambdas, 29](#)
- [__cpp_nsdmi, 29](#)
- [__cpp_range_based_for, 29](#)
- [__cpp_raw_strings, 29](#)
- [__cpp_ref_qualifiers, 30](#)
- [__cpp_return_type_deduction, 30](#)
- [__cpp_rtti, 30](#)
- [__cpp_runtime_arrays, 30](#)
- [__cpp_rvalue_reference, 30](#)
- [__cpp_rvalue_references, 30](#)
- [__cpp_sized_deallocation, 30](#)
- [__cpp_static_assert, 30](#)
- [__cpp_threadsafe_static_init, 31](#)
- [__cpp_unicode_characters, 31](#)
- [__cpp_unicode_literals, 31](#)
- [__cpp_user_defined_literals, 31](#)
- [__cpp_variable_templates, 31](#)
- [__cpp_variadic_templates, 31](#)
- [__gnu_linux__, 50](#)
- [__has_include, 52](#)
- [__has_include_next, 52](#)
- [__k8, 58](#)
- [__k8__, 58](#)
- [__linux, 60](#)
- [__linux__, 60](#)
- [__pic__, 61](#)
- [__pie__, 62](#)
- [__unix, 72](#)
- [__unix__, 72](#)
- [__x86_64, 73](#)
- [__x86_64__, 74](#)
- [linux, 74](#)
- [unix, 74](#)
- [paintEvent](#)
 - [AntWin, 11](#)
- [pause](#)
 - [AntThread, 9](#)
- [QT_MOC_LITERAL](#)
 - [moc_antthread.cpp, 17](#)
 - [moc_antwin.cpp, 17](#)
- [qt_meta_stringdata_AntThread_t, 12](#)
 - [data, 12](#)
 - [stringdata0, 12](#)
- [qt_meta_stringdata_AntWin_t, 13](#)
 - [data, 13](#)
 - [stringdata0, 13](#)
- [run](#)
 - [AntThread, 9](#)
- [step](#)
 - [AntThread, 9](#)
 - [AntWin, 11](#)
- [stringdata0](#)
 - [qt_meta_stringdata_AntThread_t, 12](#)
 - [qt_meta_stringdata_AntWin_t, 13](#)
- [unix](#)

moc_predefs.h, [74](#)

x

 Ant, [8](#)

y

 Ant, [8](#)