Target & Compiler

Target processor type, compiler behavior

Set the Target and Compiler options to match your run time environment. Polyspace® uses these options to interpret variables sizes, language extensions, and compiler behavior. If you cannot match your compiler and target environment with these options, contact MathWorks® support.

Polyspace Analysis Options

Language

`Source code language (-lang)`	Specify language of source files
`Respect C90 standard (-no-language-extensions)`	Restrict analysis to C language specified in ANSI C standard
`C++11 extensions (-cpp11-extension)`	Allow C++11 language extensions

Compiler

Required	
`Compiler (-compiler)`	Specify the compiler that you use to build your source code
`Target processor type (-target)`	Specify size of data types and endianness by using predefined target processor list

Optional	
`Management of size_t (-size-t-type-is)`	Specify the underlying data type of size_t
`Management of wchar_t (-wchar-t-type-is)`	Specify the underlying data type of wchar_t
`Block char16/32_t types (-no-uliterals)`	Disable Polyspace definitions for char16_t or char32_t
`Sfr type support (-sfr-types)`	Specify sizes of sfr types for code developed with Keil or IAR compilers
`Pack alignment value (-pack-alignment-value)`	Specify default structure packing alignment for code developed in Visual C++
`Ignore pragma pack directives (-ignore-pragma-pack)`	Ignore #pragma pack directives

Advanced

`Division round down (-div-round-down)`	Round down quotients from division or modulus of negative numbers instead of rounding up
`Enum type definition (-enum-type-definition)`	Specify how to represent an enum with a base type
`Signed right shift (-logical-signed-right-shift)`	Specify how to treat the sign bit for logical right shifts on signed variables

Examples and How To

Specify Generic Target Processors

If your application is designed for a custom target processor, you can configure many basic characteristics of the target by selecting the selecting the mcpu... **(Advanced)**

Modify Predefined Target Processor Attributes

You can modify certain attributes of the predefined target processors.

Analyze Keil or IAR Compiled Code

Find which Keil or IAR-specific extensions of the C language standard are supported for verification.

Concepts

Address Alignment

Polyspace software handles address alignment by calculating sizeof and alignments.

Supported C++ 2011 Extensions

The following table list which C++ 2011 standards Polyspace can analyze.