

Synopsys ARC MetaWare Debugger

Abstract

This document describes the usage of the Synopsys ARC MetaWare debugger as target system. The minimum required version is K-2015.06. TESSY supports the ARC controller family.

Please note: Since TESSY version 4.2.12 the communication has changed for this target. Please adapt TEE Attribute Slave Call, if you updated your TESSY installation (See 2).

Please note: Since TESSY version 4.2.12 there is a new master.cmd script file available. It is essential to use this new master.cmd file. Do not remove #BREAKPOINT or #OUTPUT_BUFFER from the new script file.

Table of Contents

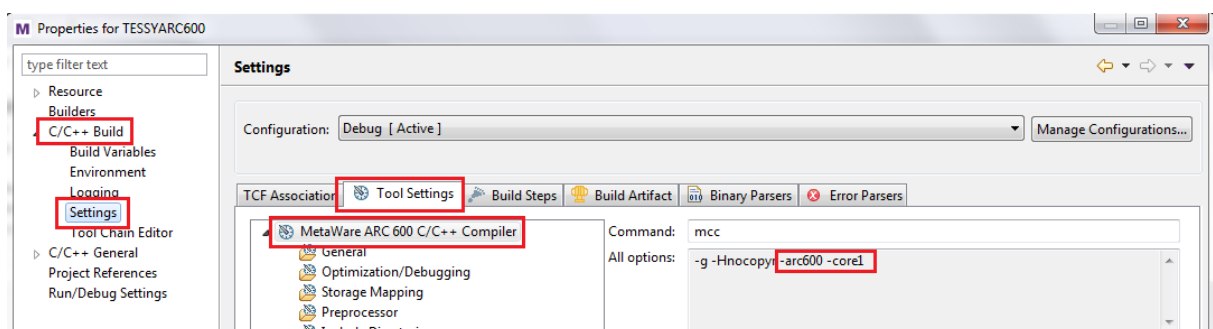
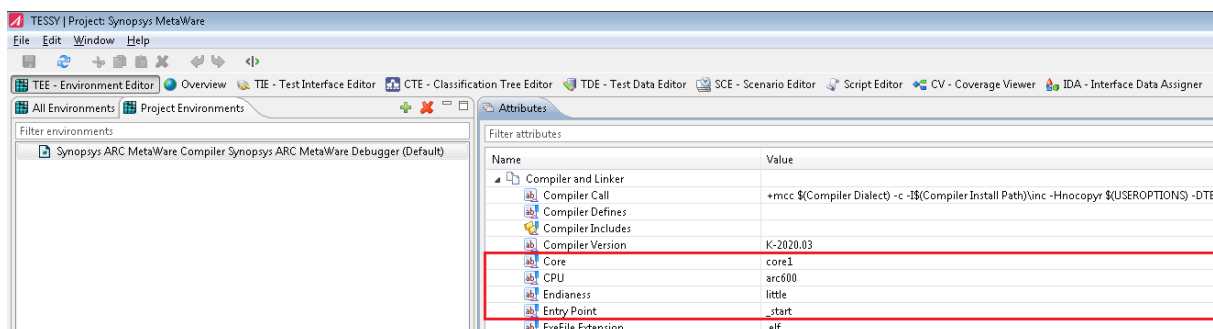
Abstract	1
1 ARC MetaWare Debugger.....	2
2 TESSY Environment Settings.....	2
3 Test Run.....	5
3.1 Parallel execution.....	5
3.2 Non-interactive test run	5
3.3 Interactive Debugging	5

1 ARC MetaWare Debugger

The communication between TESSY and ARC MetaWare is based on the debugger's scripting language. TESSY's master launches the ARC MetaWare command line debugger with a master script file which is located in TESSY's installation directory and pointed to by the TEE attribute **Init Script**. The script instructs the debugger to read and write data from and into the running target binary. Actually, the script is just a template which is modified by TESSY's master before the debugger runs it. You may add commands to the beginning of the master script file but you should not alter the basic procedures following the **while** control flow statement.

2 TESSY Environment Settings

There are a few attributes within the TEE which have to be checked. Choose the desired core, CPU, endianness, and entry point at compiler level of your file node. The correct values can be found in your ARC MetaWare project's **Properties** dialog as show below. Since TESSY version 4.2.12 the TEE attributes **Slave Call** and **Slave Debug Call** have changed. The executable directory is now `$(MODULEPATH)` instead of `$(TESSY_TESTAREA)\testarea` and the script to be read is now `master.cmd` instead of `$(Init Script)`.



Properties for TESSYARC600

type filter text

- Resource
- Builders
 - C/C++ Build
 - Build Variables
 - Environment
 - Logging
 - Settings
 - Tool Chain Editor
- C/C++ General
- Project References
- Run/Debug Settings

Settings

Configuration: Debug [Active] Manage Configurations...

TCF Association: **Tool Settings** Build Steps Build Artifact Binary Parsers Error Parsers

- MetaWare ARC 600 C/C++ Compiler
 - General
 - Optimization/Debugging
 - Storage Mapping
 - Preprocessor
 - Include Directories
 - Language Compatibility
 - C++
 - Esoteric Toggles
 - Processor/Extensions
 - DSP Extensions
 - Memory Model
 - Overlay Support
 - Special Register Widths
 - ARCompact Assembler
 - Assembler Preprocessor
 - General
 - Include paths
 - ARCompact Specific
 - DSP Support
 - MetaWare Linker for ARCompact**
 - Runtime Libraries**
 - Search Path
 - Additional Object Files & Libraries
 - Command files
 - Map Listing
 - Output Generation
 - Miscellaneous
 - Extension settings (inherited from Compiler)
 - Overlay Support

Named library from buildlib tool (-Hlib=)

Specify Heap Size (values are power of 2 such as 256,512,1K,...,16M)

Heap Size (-Hheap=)

Specify Stack Size (values are power of 2 such as 256,512,1K,...,128K)

Stack Size (-Hstack=)

Link in the Hostlink Libraries

Include profiling code (-pg)

Endian: **Little Endian**

Link in default startup code (-Hnocrt)

Link in standard C/C++ library (-Hnolib)

Include RTOS-specific thread library: (None)

Runtime C++ library: MetaWare C++ library

Timer for hardware profiling: (None)

Compact Single-precision floating point libraries (-Xspfp_compact)

Fast Single-precision floating point libraries (-Xspfp_fast)

Compact Double-precision floating point libraries (-Xdpfp_compact)

Fast Double-precision floating point libraries (-Xdpfp_fast)

Real Time Counter(-Xrtc)

Compile/Link with Compact Lib(-Hcl)

Linker's INITDATA command-file feature (-Hcrt_initcopy)

Zero the BSS section(-Hcrt_initbss)

Invalidate the data cache(-Hcrt_invcache)

Initialize the timer module(-Hcrt_inittimer)

Argc and argv to main() and call exit() upon return(-Hcrt_argv)

Use a minimal implementation of printf rather than the standard version(-Hcrt_pfmim)

Link in additional object files located in the library(-Hcrt_pfnofit)

Restore Defaults Apply

OK Cancel

Properties for TESSYARC600

type filter text

- Resource
- Builders
 - C/C++ Build
 - Build Variables
 - Environment
 - Logging
 - Settings
 - Tool Chain Editor
- C/C++ General
- Project References
- Run/Debug Settings

Settings

Configuration: Debug [Active] Manage Configurations...

TCF Association: **Tool Settings** Build Steps Build Artifact Binary Parsers Error Parsers

- MetaWare ARC 600 C/C++ Compiler
 - General
 - Optimization/Debugging
 - Storage Mapping
 - Preprocessor
 - Include Directories
 - Language Compatibility
 - C++
 - Esoteric Toggles
 - Processor/Extensions
 - DSP Extensions
 - Memory Model
 - Overlay Support
 - Special Register Widths
 - ARCompact Assembler
 - Assembler Preprocessor
 - General
 - Include paths
 - ARCompact Specific
 - DSP Support
 - MetaWare Linker for ARCompact**
 - Runtime Libraries
 - Search Path
 - Additional Object Files & Libraries
 - Command files
 - Map Listing
 - Output Generation
 - Miscellaneous**
 - Extension settings (inherited from Compiler)
 - Overlay Support

Display copyright (-q)

Additional options for linking (one per line)

Search libraries together as a group (-Bgrouplib)

Entry point (-e) _start

Permit duplicate symbols with warning (-zdup)

Enable run-time checking for ARC extensions (-Hxcheck)

Disable run-time checking for ARC extensions (-Hnoxcheck)

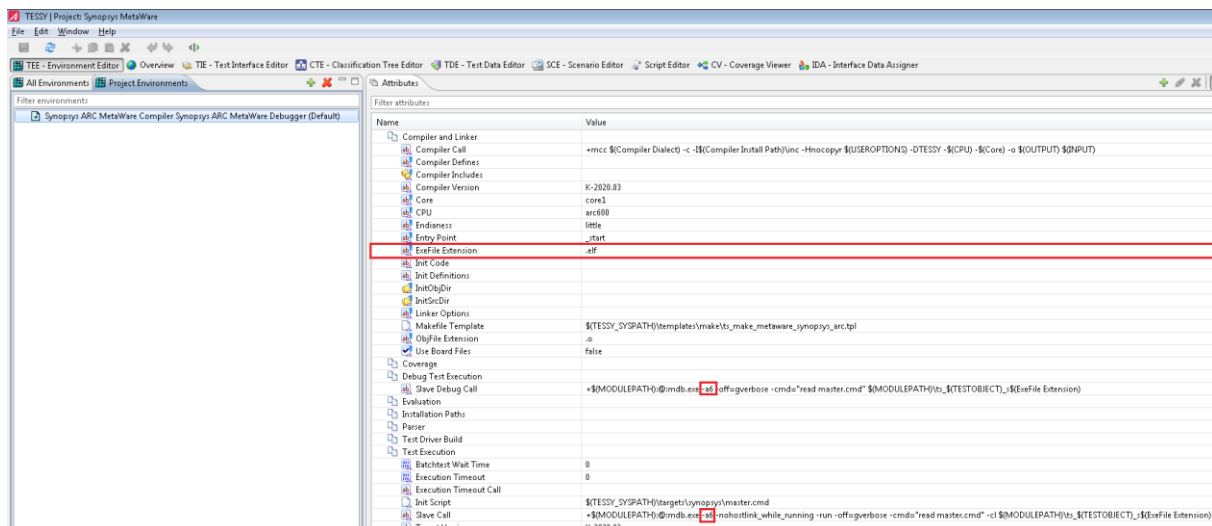
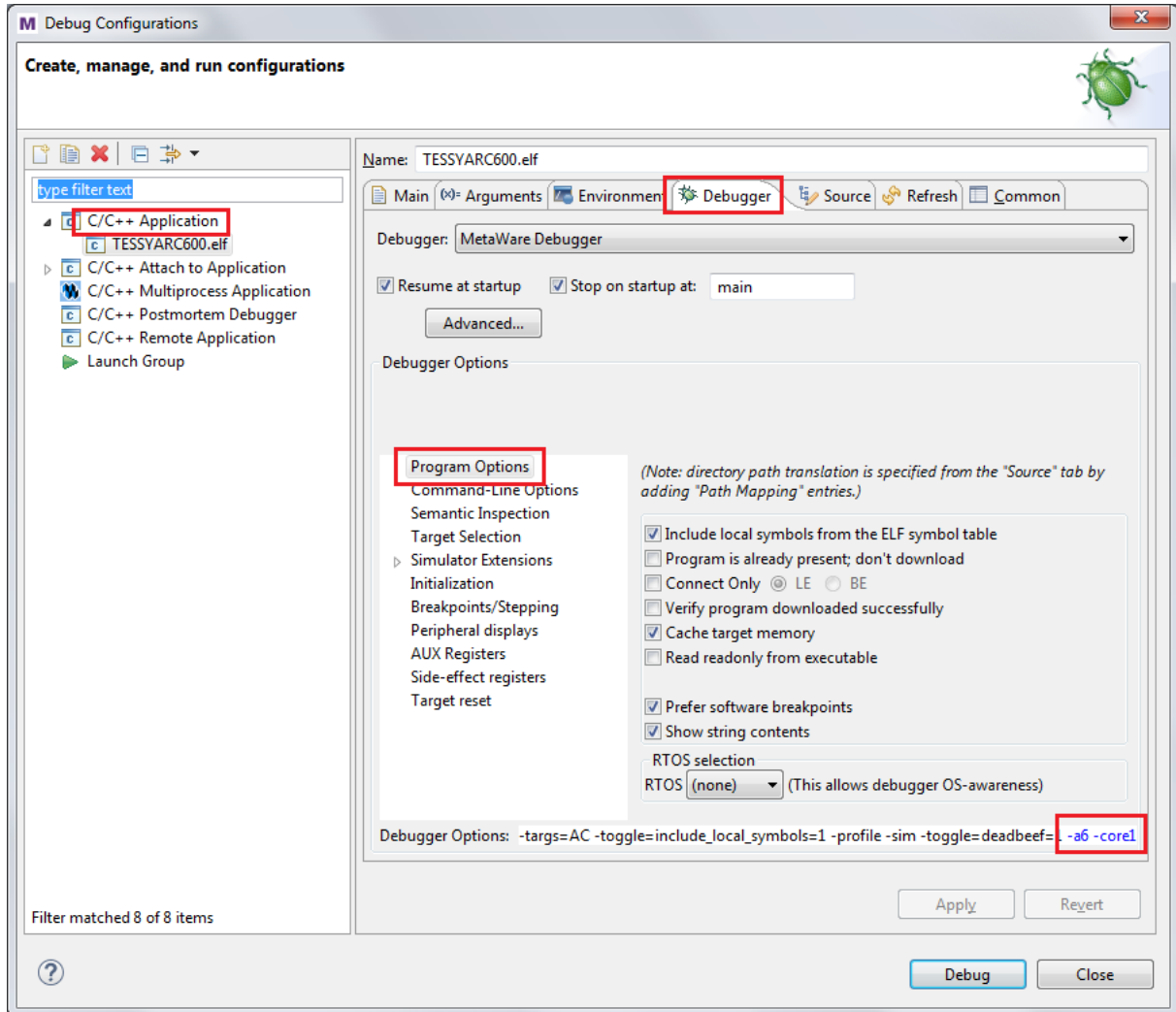
Exclude -Hnoxcheck stub from link (-Hnoxcheck_obj)

Restore Defaults Apply

OK Cancel

TESSY Application Notes

Adjust the executable file extension unless it is `.elf` and check the debugger options. Make sure to use the correct '-a' option which can be determined by the ARC MetaWare **Debug Configurations** dialog as shown below.



3 Test Run

There are two different calls to the ARC MetaWare's debugger. One is used for the ordinary test run the other is used for the interactive debugging of the test object.

3.1 Parallel execution

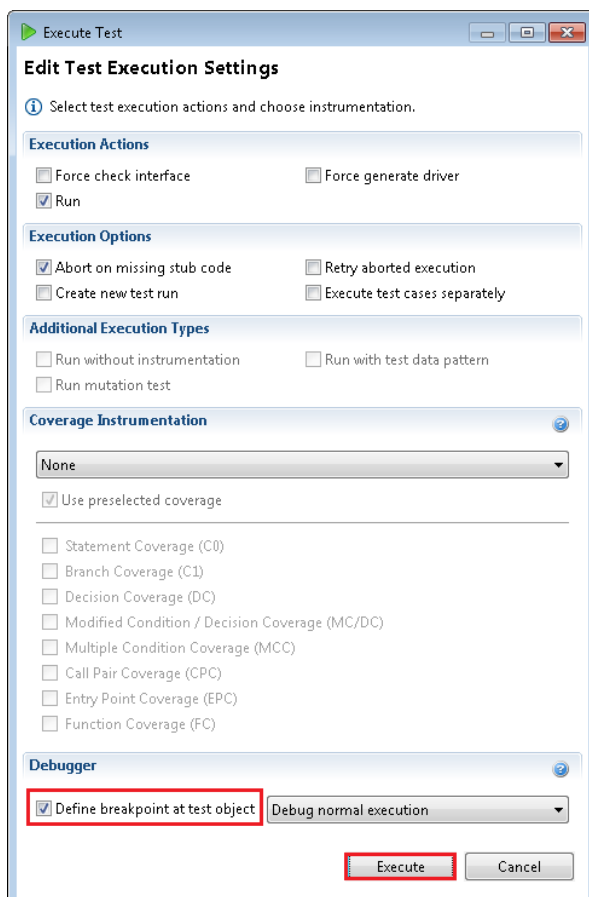
By default, 15 test object tests run in parallel. To disable this feature or to alter the number of tests to be run in parallel, adjust TEE attribute **Target Concurrency** to the desired value. **1** means to disable the feature. It strongly depends on the capacity of your PC to decide which value is the best one. Too high a value may slow down your PC and consequently prolong the execution time of your batch test.

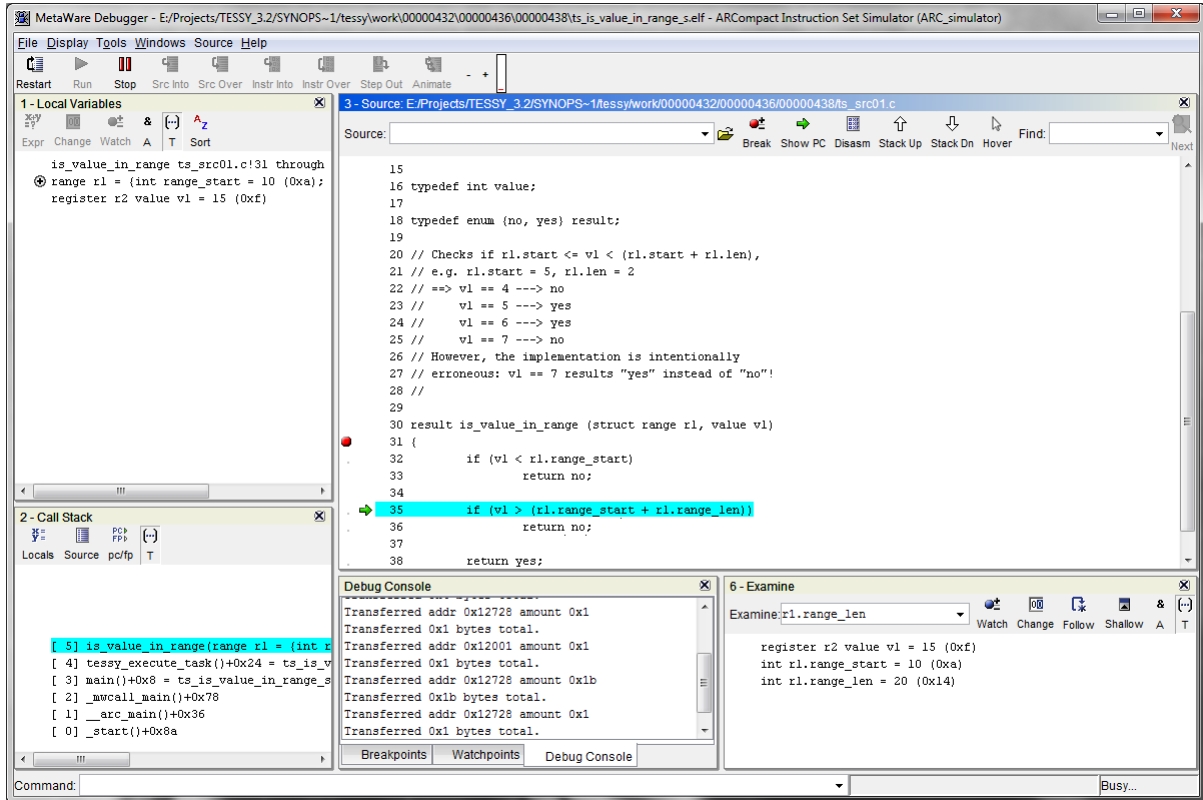
3.2 Non-interactive test run

The non-interactive test run is command line based only. Thus, no ARC MetaWare IDE debugger will be launched. You will see messages coming from the command line debugger which are issued each time memory has been read or written within TESSY's **Console** view.

3.3 Interactive Debugging

For interactive debugging open TESSY's **Execute Test** dialog and select **Define Breakpoint at Test Object** as shown below. Clicking **Execute** will start the test run and let the ARC MetaWare IDE debugger halt at your test object containing your test data. The global variables `TS_CURRENT_TESTCASE` and `TS_CURRENT_TESTSTEP` contain the test case respectively the test step number.





You may step through your code by selecting a source line and opening the context menu as shown below and by clicking **Run to cursor F11** to let the debugger halt at the selected line.

