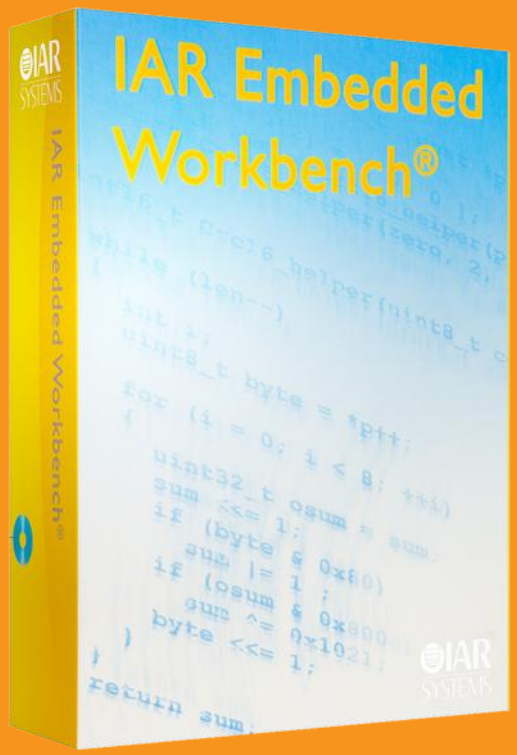
One tool chain for 2,000+ devices: IAR Embedded Workbench for ARM - Highlights



Recent enhancements

Version 6.10 – October 2010

- Full C++ (exceptions & RTTI)
- C99 compliance
- Cortex-A5
- Power debugging
- CMSIS SVD

Version 6.30 – October 2011

- JTAGjet-Trace integration
- Speed optimizations
- Stack usage analysis
- Improved inline assembler
- Graphical event log for Cortex-M3/M4
- CMSIS 2.10 is integrated
- Support for TI Stellaris ICDI

Version 6.20 – April 2011

- Subversion
- Power debugging enhancements
- CMSIS DSP library for Cortex-M3/M4
- Cortex-A8/A9
- Automatic selection of printf/scanf formatter
- Virtual Function Elimination (VFE)

Version 6.40 – June 2012 (Overview)

IDE

- Editor
- Source browser

Compiler

- Speed optimizations
- Enhanced stack usage analysis
- Enhanced Inline assembler

Debugger

- I-jet support
- Improved ITM event plot
- Memory configuration
- Custom SFR window

General

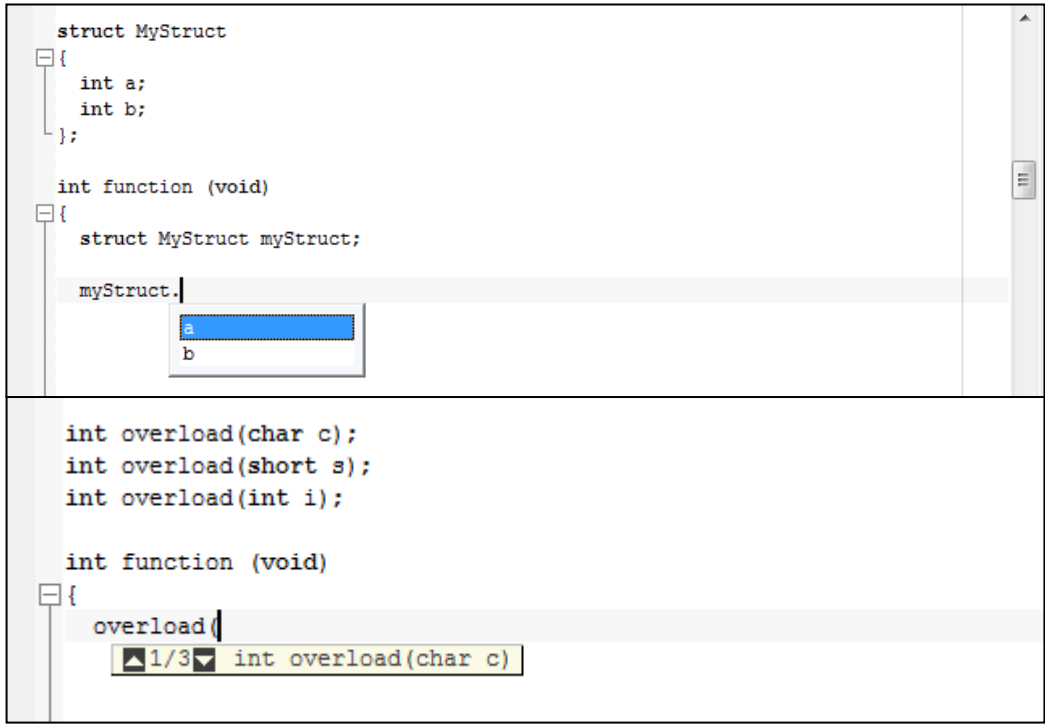
- Support for ARM Cortex-M0+, R5, R7, A7, A15
- CMSIS 3.01
- New devices

Latest EWARM Enhancements Details

- IDE
- Compiler
- Linker
- Debugger

New editor

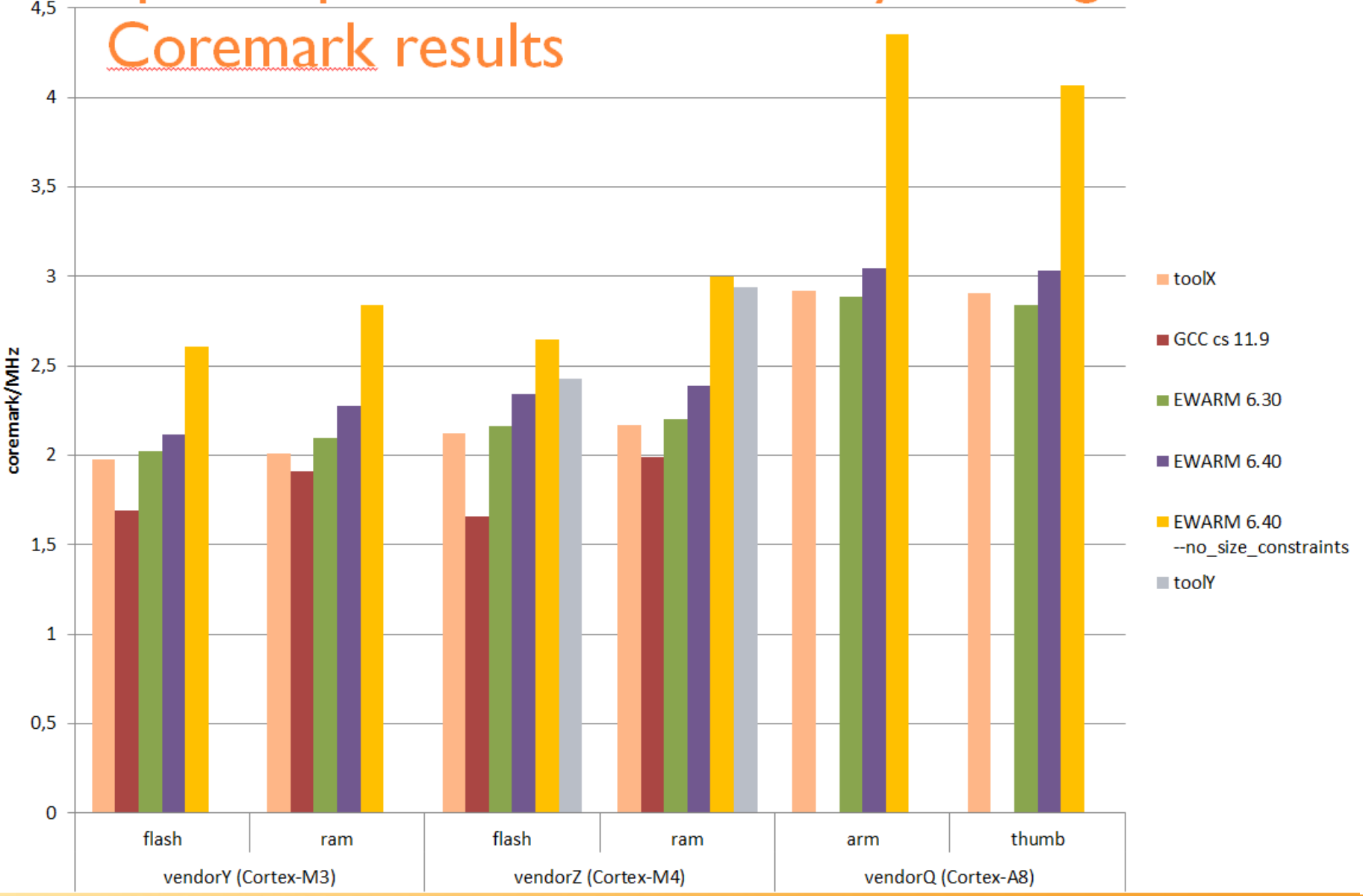
- Code and comment folding
- Word completion
- Code completion
  - When you type ., ->, or :: after a class or object name, the editor shows a list of symbols that are available in a class
- Parameter hint
  - Start typing the first parenthesis after a function name, and the editor suggests parameters as tooltip information
- Block select (Alt-mark)
- Block indent (mark text + tab)



Latest EWARM Enhancements Details

- IDE
- Compiler
- Debugger
- General

Speed optimizations - Industry leading Coremark results



New core support

- Cortex-M0+
- Cortex-R5
- Cortex-R7
- Cortex-A7
- Cortex-A15

CMSIS 3.01

- Latest version of ARM CMSIS integrated

Latest EWARM Enhancements Details

- IDE
- Compiler
- Linker
- Debugger

Stack usage analysis (introduced in 6.30)

- Under the right circumstances, the linker can calculate the maximum stack usage for each call graph root.
- Stack usage chapter in the linker map file, listing of maximum stack depth for each call graph root.
- The compiler will generate stack info for each C function, but if there are indirect calls, you must supply a list of possible functions that can be called from each calling function.
  - Two new pragmas:
    - #pragma calls=*function*[, *function*...]
    - #pragma call_graph_root[=*category*]
  - Stack usage control file

Stack usage analysis result

```
**************************************************************
*** STACK USAGE ***
Program entry
  __iar_program_start: 0x00000319

  Maximum call chain                          112 bytes
    __iar_program_start                         0
    __cmain                                     0
    main                                        8
    DoForegroundProcess                         8
    PutFib                                     16
    putchar                                    16
    __write                                     8
    __dwrite                                    8
    __iar_sh_stdout                            24
    __iar_get_ttio                             24
    __iar_lookup_ttioh                          0
```

Stack usage analysis enhancement (6.40)

- `check that` directive in linker configuration file to check that the stack usage calculated by the linker does not exceed the stack space allocated by the user
- C++ source code is now supported
- Improved support with multi-file compilation (`--mfc`)
- Support for recursion
- Log output option that produces a text representation of the call graph

Latest EWARM Enhancements Details

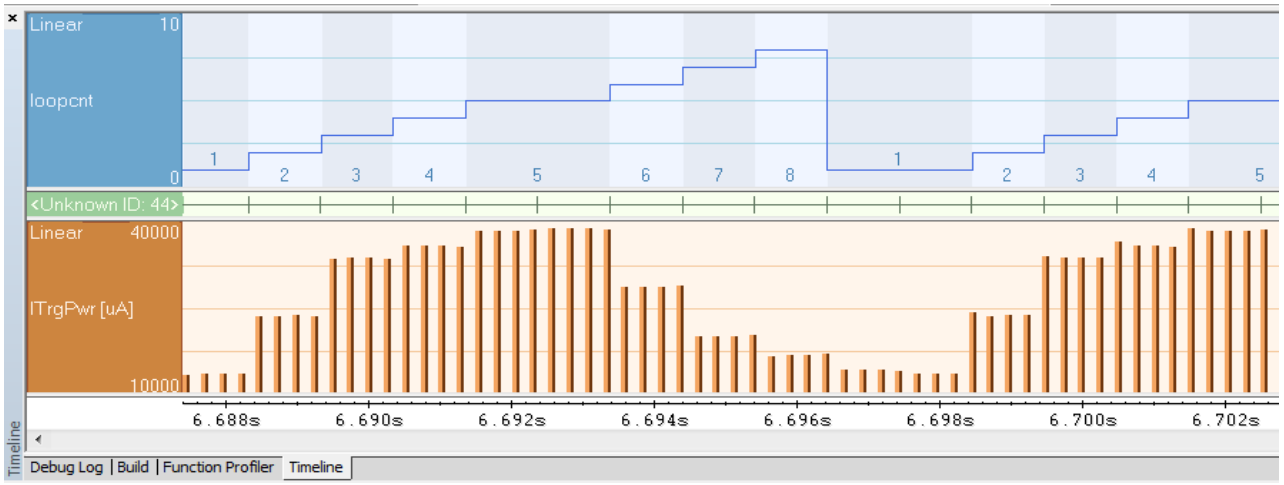
- IDE
- Compiler
- Debugger
- General

I-jet

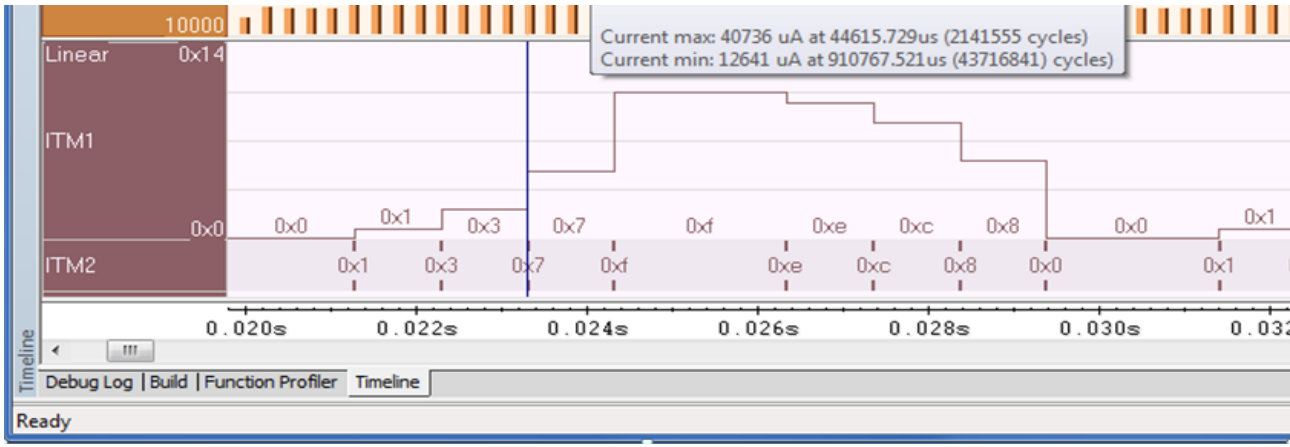
- Powered entirely by the USB port
- Hi-speed USB 2.0 interface (480Mbps)
- Supports ARM7/9/11, Cortex-M/R/A
- Target power can be supplied from I-jet (5V@400mA)with overload protection
- JTAG and Serial Wire Debug (SWD) clocks up to 32 MHz
- SWO frequency up to 60 MHz (UART and Manchester encoding)
- Target power consumption can be measured at 200 kHz with high accuracy (200uA)
- Supports target voltage range from 1.65V to 5V

Enhanced power measurements with I-jet

- Improved plot functions in Timeline window

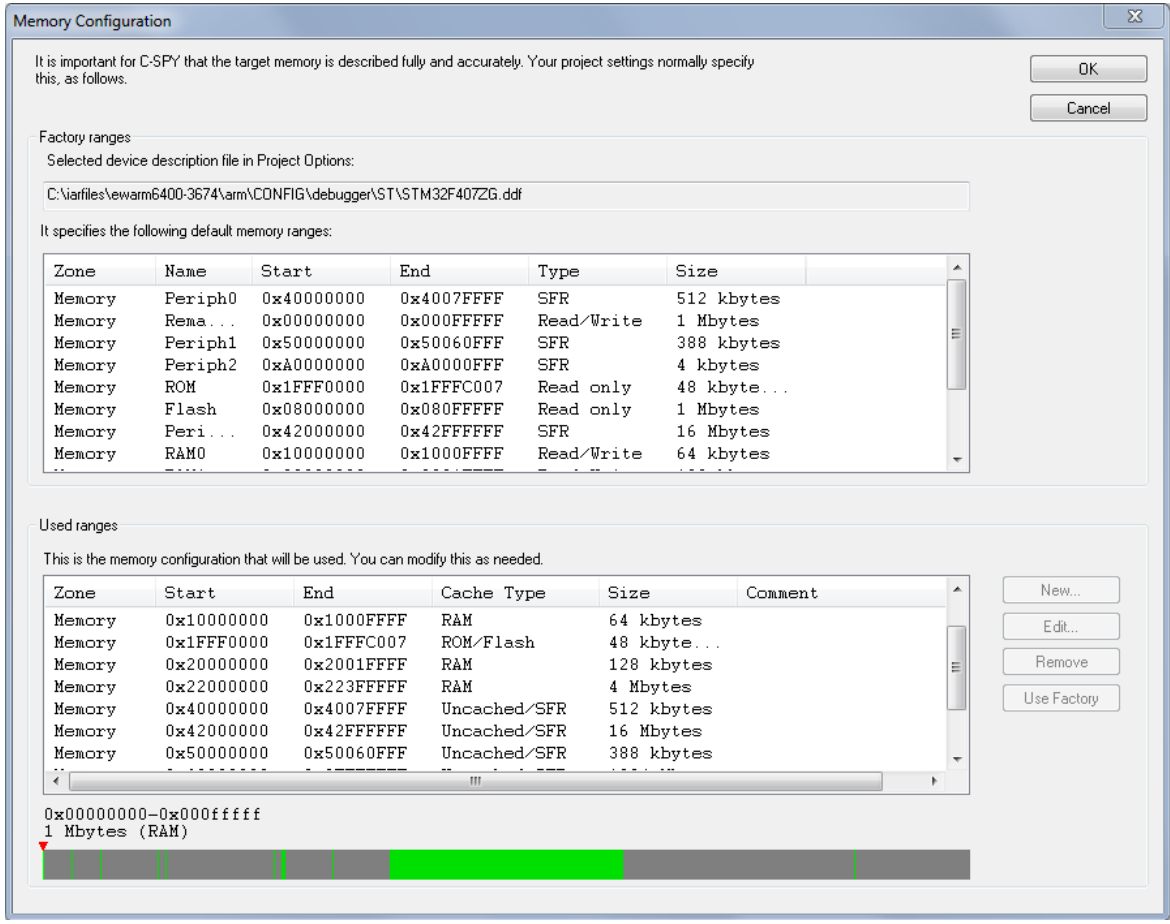


- Improved ITM event plot in Timeline window



Memory configuration framework

- Gives the developer the ability to see at-a-glance the memory map of their MCU
- Currently exists for I-jet —other drivers will be added in the future
- Adds safety against illegal memory accesses



www.iar.com/ewarm