MCUXpresso IDE v11.1.0
Overview of Main Enhancements and Changes

DECEMBER 2019

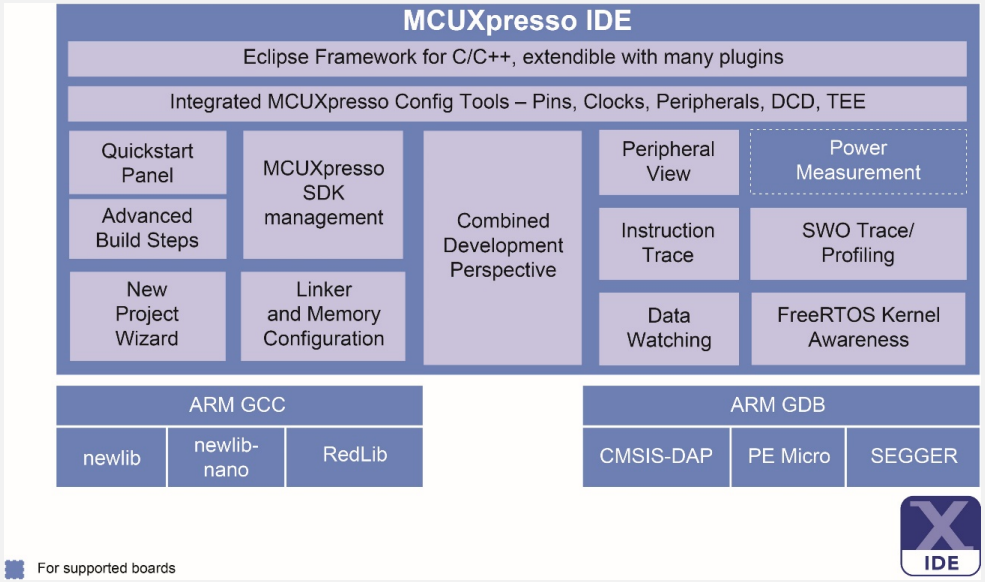
PUBLIC

SECURE CONNECTIONS
FOR A SMARTER WORLD

Learn more at: www.nxp.com/mcuxpresso/ide

MCUXpresso IDE

Free Eclipse / GCC-based Development



- Feature-rich, unlimited code size, optimized for ease-of-use, based on industry standard Eclipse framework for NXP's Kinetis and LPC MCUs and i.MX RT crossover processors
- Application development with Eclipse and GCC-based IDE for advanced editing, compiling and debugging
- Supports custom development boards, Freedom, Tower and LPCXpresso boards, and i.MX RT evaluation kits with debug probes from NXP, P&E and SEGGER
- Free: Full Featured, unlimited Code Size, no special activation needed, community based support, advanced trace capabilities, MTB and ETB instruction trace
- Works in conjunction with MCUXpresso Config Tools and MCUXpresso SDK to provide complete development environment

LPC | Kinetis | i.MX RT

MCUXpresso Software and Tools

for General Purpose MCUs and Crossover processors

December 2019 Release

MCUXpresso IDE

Edit, compile, debug and optimize in an intuitive and powerful IDE

v11.1.0

MCUXpresso SDK

Runtime software including peripheral drivers, middleware, RTOS, demos and more

v2.7

MCUXpresso Config Tools

Online and desktop tool suite for system configuration and optimization

v7

MCUXpresso Secure Provisioning Tool

Graphical and command line tool for securely provisioning and programming MCUs with secure boot

New in December 2019

MCUXpresso IDE v11.1.0 : Supported platforms

- Windows 7 / 8 / 10 (64-bit only)
  - During 2020, we will migrate to supporting Win10 only
- Ubuntu 16.04 LTS / 18.04 LTS (64-bit only)
  - Some updates made to product and installer to allow better compatibility with newer versions of Ubuntu going forward (but supported platforms currently remain unchanged)
- macOS 10.12 to 10.14
  - IDE has not yet been notarized for installation on macOS 10.15, so no formal Catalina support currently
    - But see MCUXpresso IDE Community blog for latest information ….

Moving from a previous version of MCUXpresso IDE

- Projects
  - Projects from earlier versions of MCUXpresso IDE will import straight into v11.1.0
    - SDKs installed by earlier IDE will remain accessible
  - Any existing workspace should be backed up before being used
    - Preferably, create a new workspace and import projects from old workspace or from version control
    - Projects / workspaces created, edited or simply loaded into IDE v11.1.0 should not then be used with earlier releases
- Launch configurations
  - It is recommended that any existing .launch files from earlier IDE releases are deleted, and IDE v11.1.0 used to recreate them
  - In some circumstances, the IDE may prompt you about doing this automatically

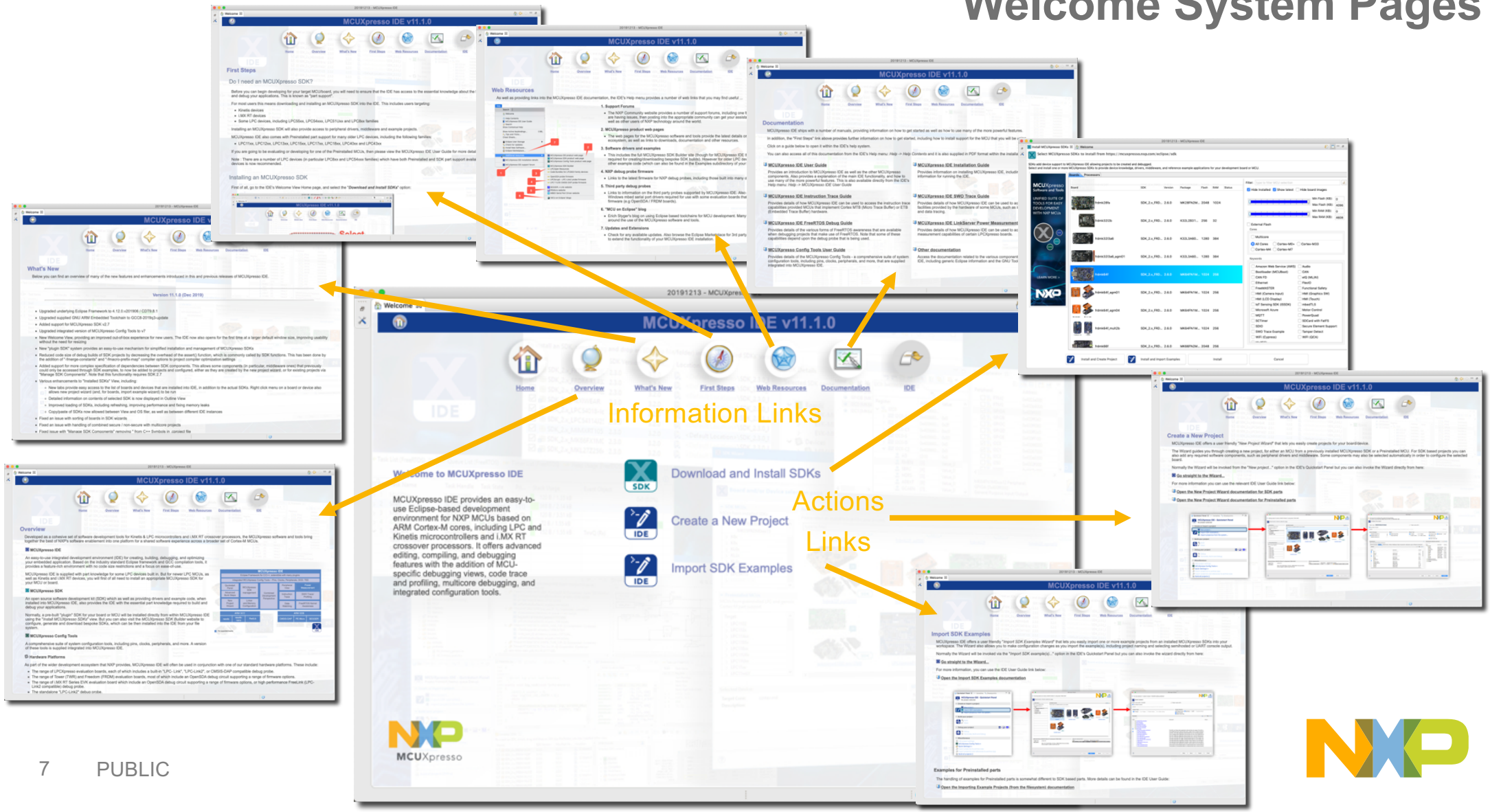
Underlying tool version updates

- Underlying IDE framework migrated to Eclipse 4.12.0.v201906 / CDT9.8.1
  - Also increased default java memory settings, meaning the IDE may now use up to 2GB by default
- Underlying GCC package migrated to ARM's GCC8-2019q3-update
- Updated integrated version of MCUXpresso Config Tools to v7
- MCUXpresso SDK support extended to include latest SDK v2.7
- Updated supplied version of P&E Micro debug probe plugin to v4.4.1
- Updated supplied version of SEGGER J-Link software to v6.54c

New Welcome System : Improved Out of box Experience

- With MCUXpresso IDE v11.1.0 we are introducing new functionality to help educate and guide new users in the initial use of the IDE. This includes simplified selection and installation of SDKs and the importing / creation of initial projects
  - Full window Welcome System
  - Larger default IDE window
    - more usable without resizing

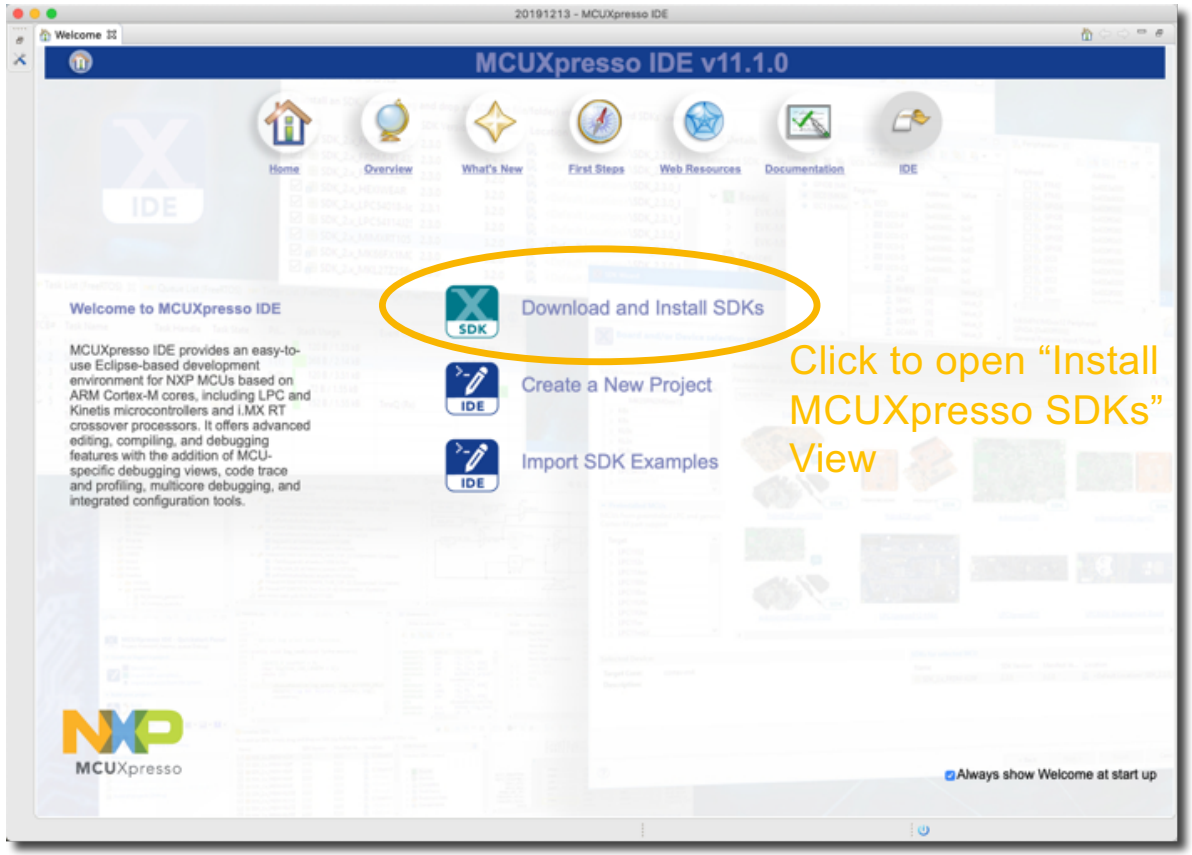
Welcome System Pages



“Plugin” SDK Installation

- MCUXpresso IDE v11.1.0 introduces a new mechanism to allow users to directly download and install pre-generated and configured “Plugin” SDKs from within the IDE itself

- For most users this replaces the need to create, download and install bespoke “file system” SDKs from SDK Builder (mcuxpresso.nxp.com) website



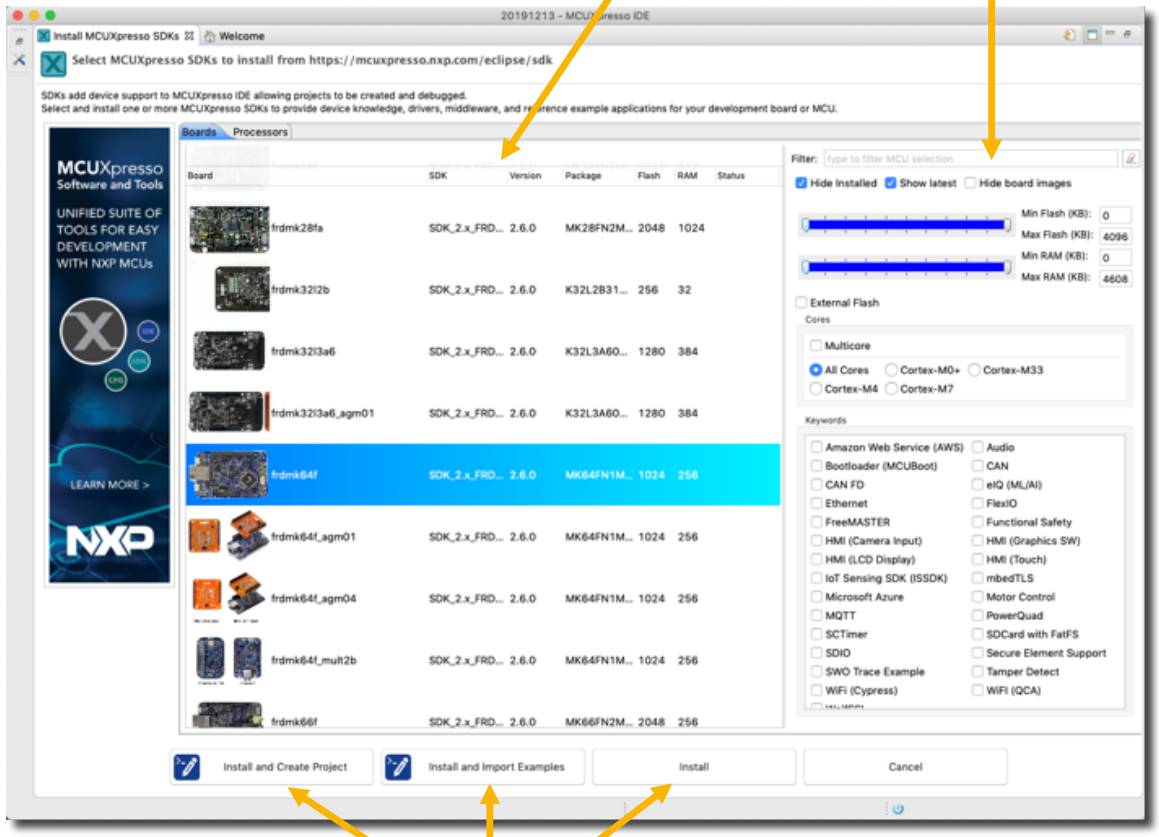
- Note that some SDK elements from 3rd parties are only available via SDK Builder due to required license click through agreements

“Install MCUXpresso SDKs” View

- Displays the set of available SDKs
  - Can sort, search and filter to quickly find the required SDK
- Select required SDK, then choose install option :
  - Just download and install SDK
  - Install and then run new project wizard
  - Install and then run import examples wizard
- Status will flag when a compatible “plugin” SDK update is available to be installed
- Option to display SDKs that have already been installed, with status showing type:
  - A “file system” SDK is already installed
  - A “plugin” SDK is already installed

List of SDKs

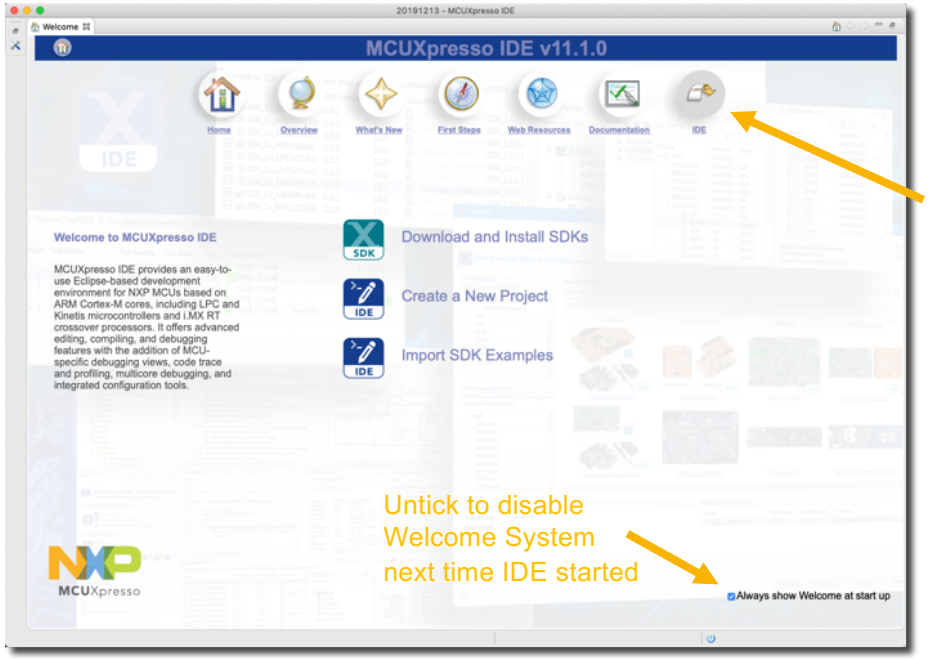
Various filtering options



Install selected SDK(s), with optional, follow-on actions

Welcome System : Note for existing users

- Although Welcome System opens by default when starting IDE (with option to disable), existing IDE front end is all still in place underneath
  - Just close the Welcome System to jump to the main IDE screen that you will recognize from previous versions
  - "File system" SDKs can still be installed as per previous releases if required
    - Log in to mcuxpresso.nxp.com
    - Select/configure SDK, then download
    - Drag'n'Drop SDK into IDE
  - Welcome System can be reopened from main IDE toolbar if required



Click on "IDE" to close Welcome System and switch to main IDE window

Untick to disable Welcome System next time IDE started

Reduced code size of debug builds of SDK projects

- Code size of debug builds decreased by reducing the overhead of the assert() function, which is commonly called by SDK functions
  - Implemented by addition "-fmerge-constants" and "-fmacro-prefix-map" options to project compiler optimization settings

```
Memory region         Used Size  Region Size  %age Used
   PROGRAM_FLASH:        9504 B       32640 B     29.12%
      BOOT_FLASH:          0 GB        128 B      0.00%
            SRAM:       1020 B         4 KB     24.90%
Finished building target: lpcxpresso804_usart_terminal.axf
```

1548 bytes saved

```
Memory region         Used Size  Region Size  %age Used
   PROGRAM_FLASH:        7956 B       32640 B     24.38%
      BOOT_FLASH:          0 GB        128 B      0.00%
            SRAM:       1020 B         4 KB     24.90%
Finished building target: lpcxpresso804_usart_terminal.axf
```

```
Memory region         Used Size  Region Size  %age Used
   PROGRAM_FLASH:       78128 B          1 MB      7.45%
      SRAM_UPPER:       62416 B        192 KB     31.75%
      SRAM_LOWER:          0 GB         64 KB      0.00%
        FLEX_RAM:          0 GB          4 KB      0.00%
Finished building target: frdmk64f_lwip_dhcp_bm.axf
```

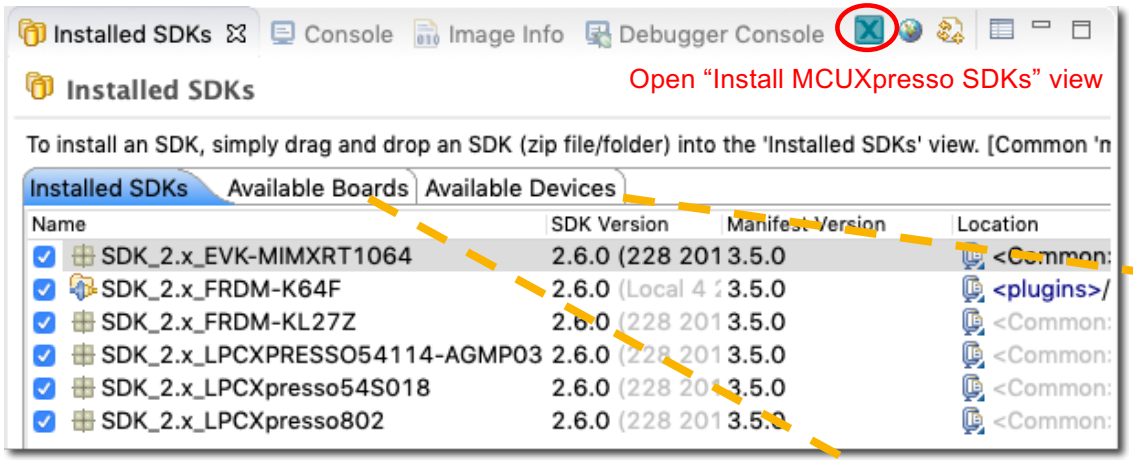
3136 bytes saved

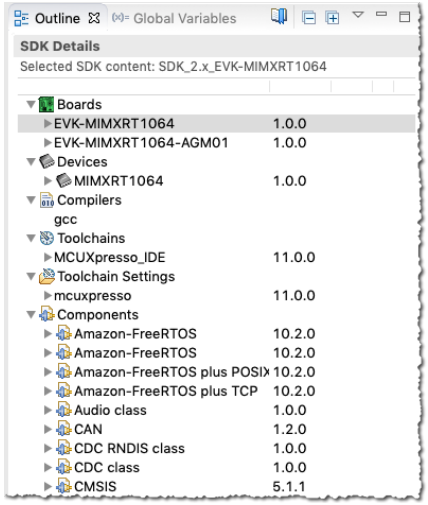
```
Memory region         Used Size  Region Size  %age Used
   PROGRAM_FLASH:       74992 B          1 MB      7.15%
      SRAM_UPPER:       62416 B        192 KB     31.75%
      SRAM_LOWER:          0 GB         64 KB      0.00%
        FLEX_RAM:          0 GB          4 KB      0.00%
Finished building target: frdmk64f_lwip_dhcp_bm.axf
```

Savings project specific, depending upon level of use of assert().
But particularly helpful on smaller footprint devices.

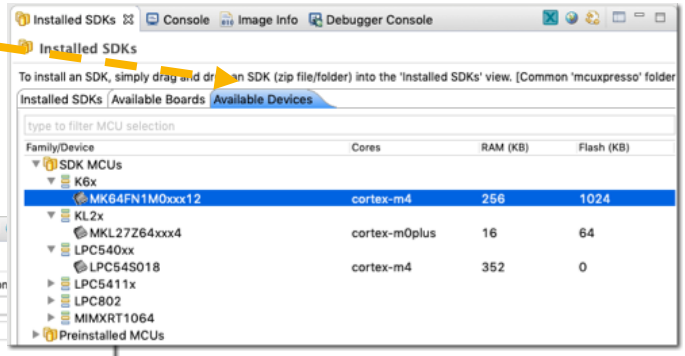
Installed SDKs View Improvements

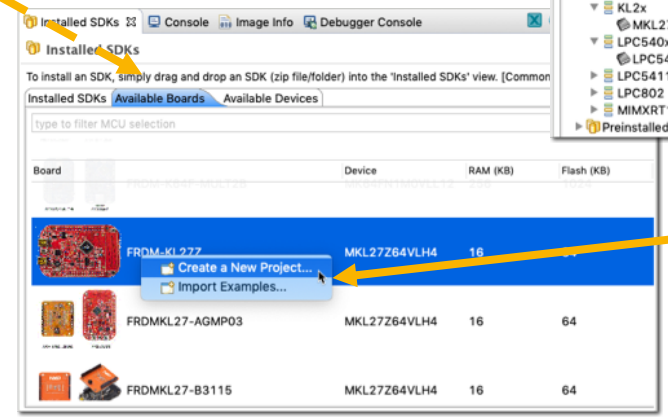
New tabs provide easy access to the list of boards and devices that are installed into IDE, in addition to the actual SDKs





Detailed information on contents of selected SDK is now displayed in Outline View



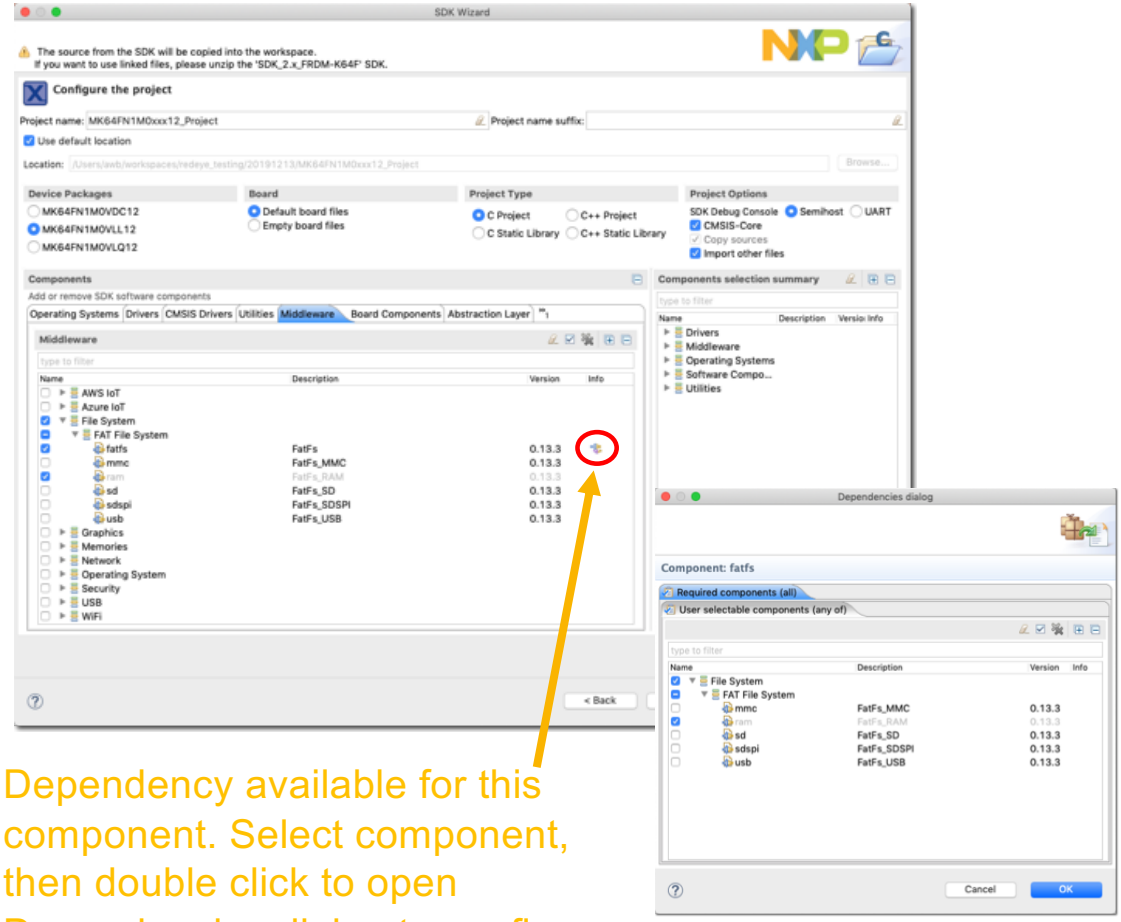


Right click menu on a board or device allows new project wizard (and, for boards, import example wizard) to be run

- Other enhancements include
  - Improved loading of SDKs, including refreshing, improving performance and fixing memory leaks
  - Copy/paste of SDKs now allowed between View and OS filer, as well as between different IDE instances

Support for SDK component “complex” dependencies

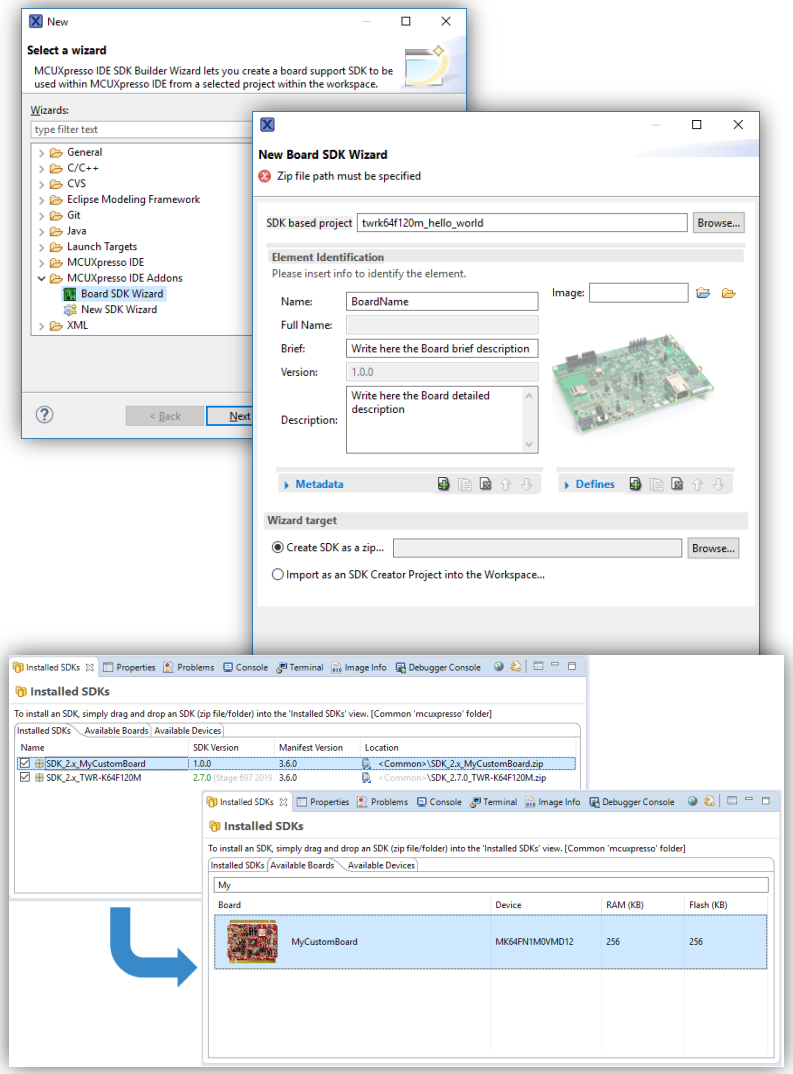
- Added support for more complex specification of dependencies between SDK components
- Allows some components (in particular, middleware ones) that previously could only be accessed through SDK examples, to now be added to projects and configured in new project wizard or using "Manage SDK Components"
- Note that this functionality requires SDK 2.7



Dependency available for this component. Select component, then double click to open Dependencies dialog to configure

SDK Board Creator

- MCUXpresso SDKs are supplied configured for use with standard NXP evaluation boards
- Once you have your own board, you can use the MCUXpresso Config Tools to make appropriate project modifications for pins, clocks and other configuration settings
- The Board SDK Creator functionality inside MCUXpresso IDE provides a means of generating a secondary SDK to contain these configuration settings for your board, that can be used in conjunction with the original SDK
  - This can then be used when creating new projects directly for your board and also be easily shared with other members of your team or customers



See blog on MCUXpresso IDE Community for more information

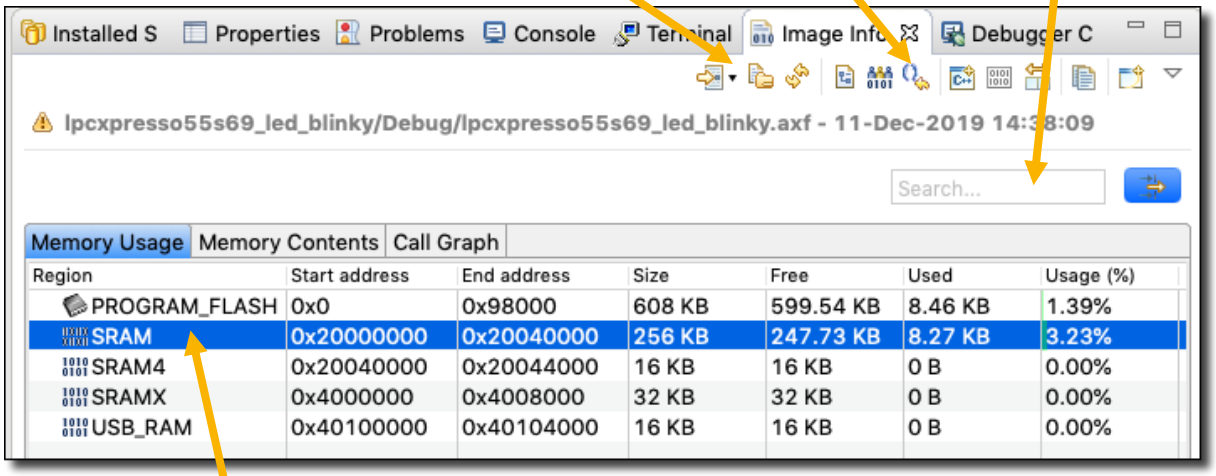
Build Analysis: Image Info View enhancements

- The Image Info view provides an easy way of analysing the output of a project build:
  - Overall memory usage
  - Content of memory regions
  - Static call graph, including stack usage information
- In IDE v11.1.0, various enhancements have been made – as per the screenshot – as well as:
  - Further improvements to load times
  - Improved support for projects outside the workspace

"Load" from project button now provides a drop-down to allow direct loading of info from multiple build configurations into multiple Image Info Views (allowing for easier comparisons)

"Open 'main' symbol" button implemented on toolbar

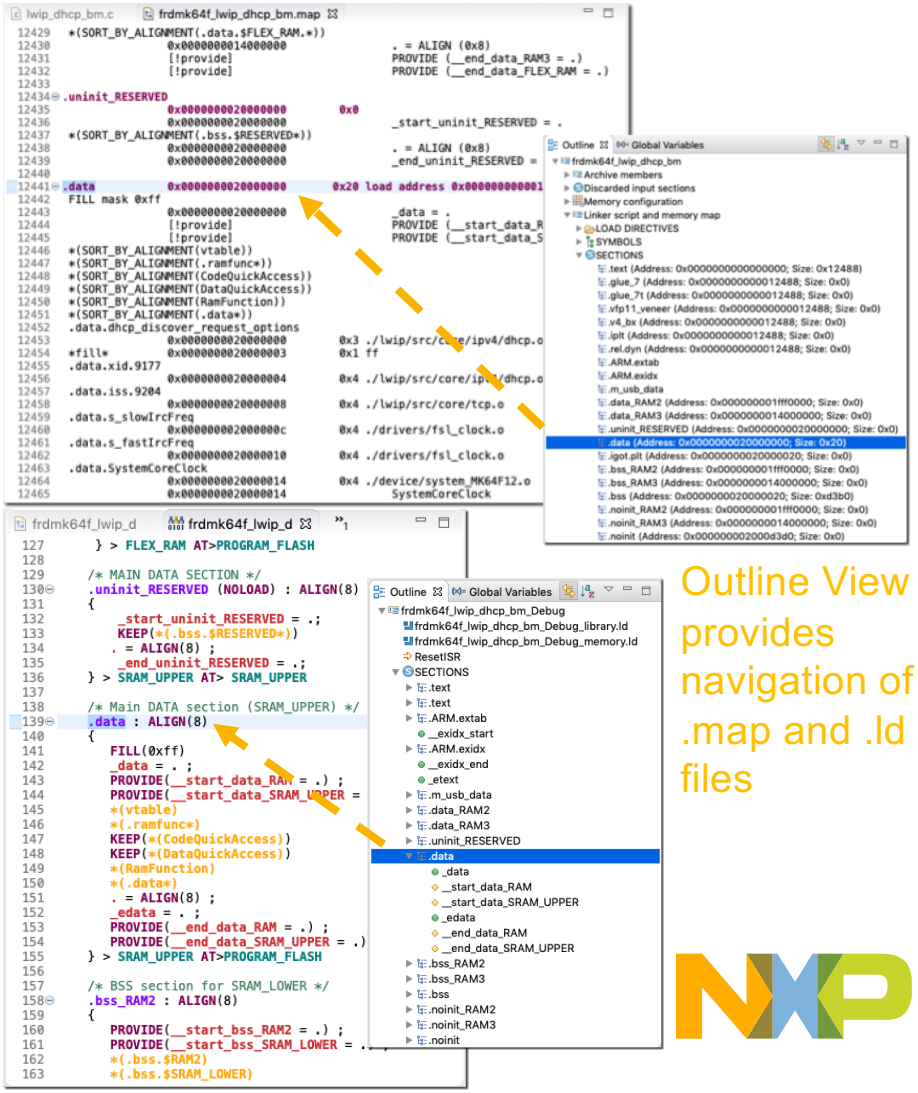
Regular expressions now supported in "Search..." box



Added ability to double-click on memory region in Memory Usage tab and jump to that region in the Memory Contents tab

Build Analysis: File awareness enhancements

- “File awareness" for linker map files, linker scripts and linker script templates provides navigation of file contents via the Outline view, along with syntax coloring and folding of elements within editor view
- Various improvements and fixes to map file awareness, including:
  - Improved handling of C++ projects
  - Fixed an issue with handling expressions containing parentheses
- Various improvements and fixes to linker script file awareness, including
  - Improved handling of libraries specified in script



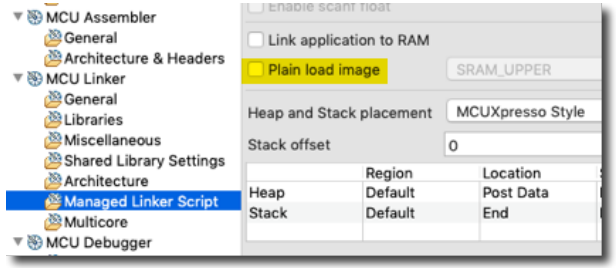
Outline View provides navigation of .map and .ld files

Managed Linker Script system changes

- Various improvements including …
  - Standardized set of symbols now generated for sections
  - Fixed an issue with LMA being incorrectly set in some circumstances for sections that do not have a real load address (e.g. .bss).
    - Although this did not prevent the generated image from working, it could cause, for example, erroneous information in Image Info View
  - Plain load image functionality now supported for RT6xx devices
  - LinkServer FreeRTOS debug config rodata now placed by script, when appropriate

```
.data_RAM2 : ALIGN(8)
{
    FILL(0xff)
    PROVIDE(__start_data_RAM2 = .) ;
    PROVIDE(__start_data_SRAM_LOWER = .) ;
    *(.ramfunc.$RAM2)
    *(.ramfunc.$SRAM_LOWER)
    *(.data.$RAM2)
    *(.data.$SRAM_LOWER)
    *(.data.$RAM2.*)
    *(.data.$SRAM_LOWER.*)
    . = ALIGN(8) ;
    PROVIDE(__end_data_RAM2 = .) ;
    PROVIDE(__end_data_SRAM_LOWER = .) ;
} > SRAM_LOWER AT>PROGRAM_FLASH
```

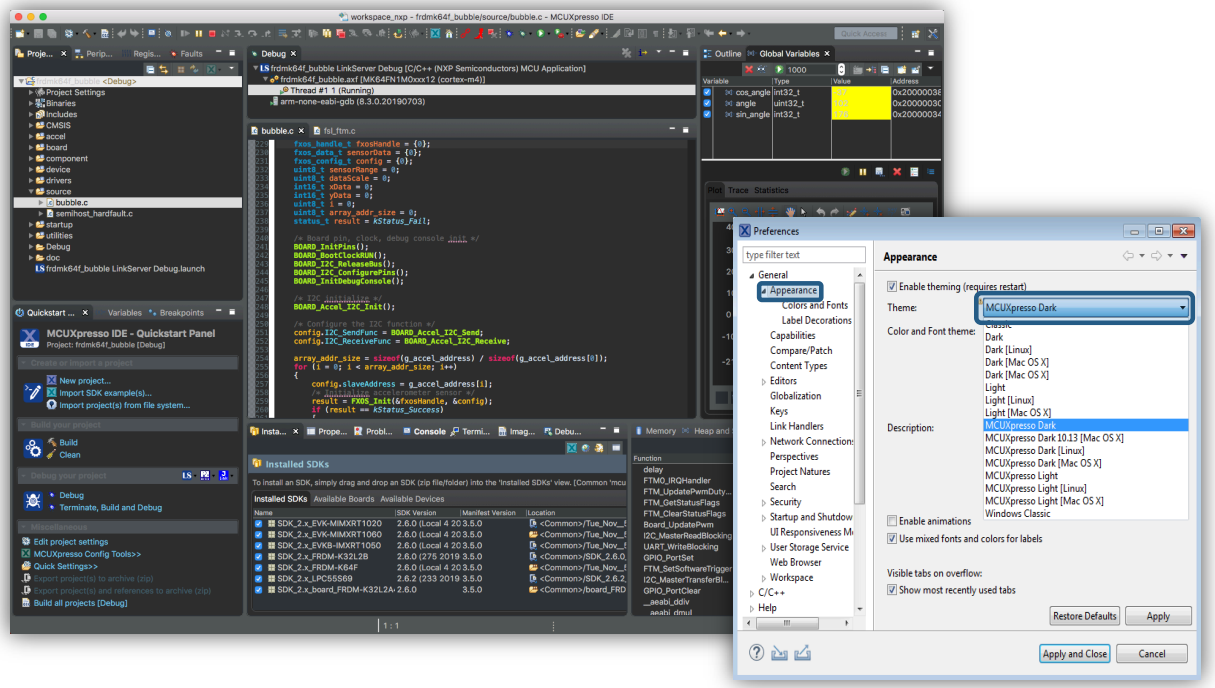
```
/* MAIN BSS SECTION */
.bss : ALIGN(8)
{
    _bss = .;
    PROVIDE(__start_bss_RAM = .) ;
    PROVIDE(__start_bss_SRAM_UPPER = .) ;
    *(.bss*)
    *(COMMON)
    . = ALIGN(8) ;
    _ebss = .;
    PROVIDE(__end_bss_RAM = .) ;
    PROVIDE(__end_bss_SRAM_UPPER = .) ;
    PROVIDE(end = .);
} > SRAM_UPPER AT> SRAM_UPPER
```

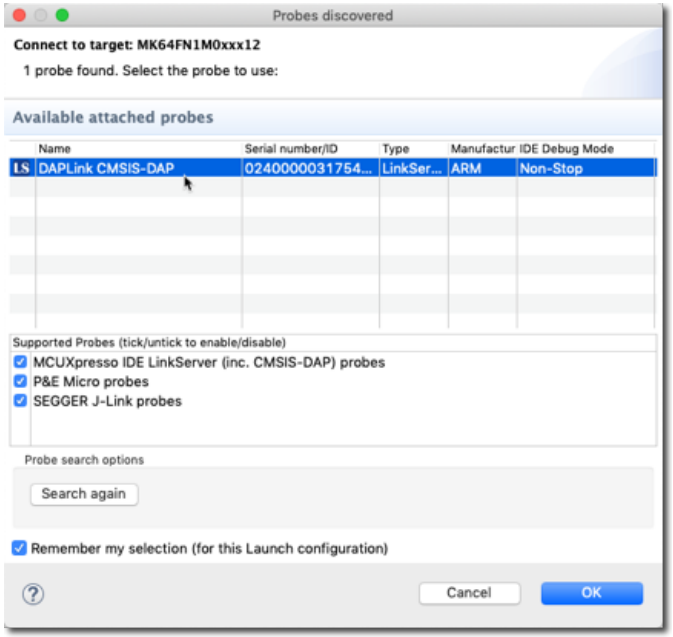



```
.text : ALIGN(8)
{
  *(.text*)
  KEEP(*freertos*/tasks.o(.rodata*)) /* FreeRTOS Debug Config */
  *(.rodata .rodata.* .constdata .constdata.*)
  . = ALIGN(8);
} > PROGRAM_FLASH
```

General Improvements

- Added MCUXpresso Dark Theme, providing better support for IDE's tailored set of Views being used in dark mode





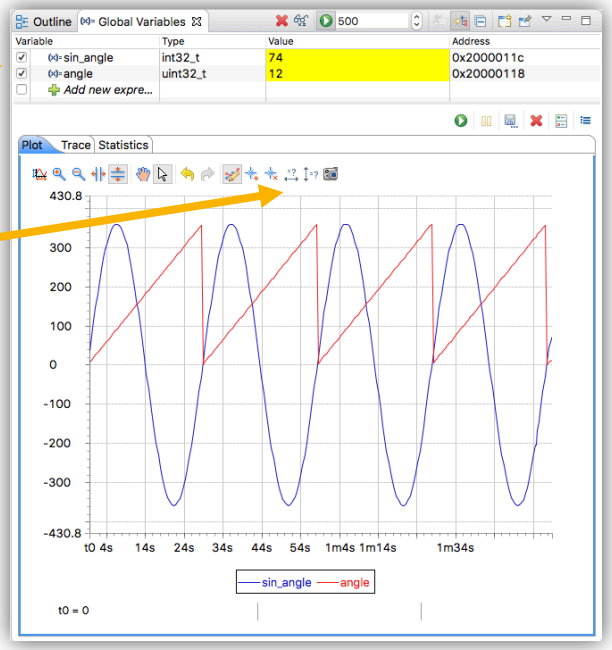
- Required debug probe can now be selected by double-clicking on it in the "Probes discovered" dialog, rather than having to use OK

Runtime Analysis : Graphing Enhancements

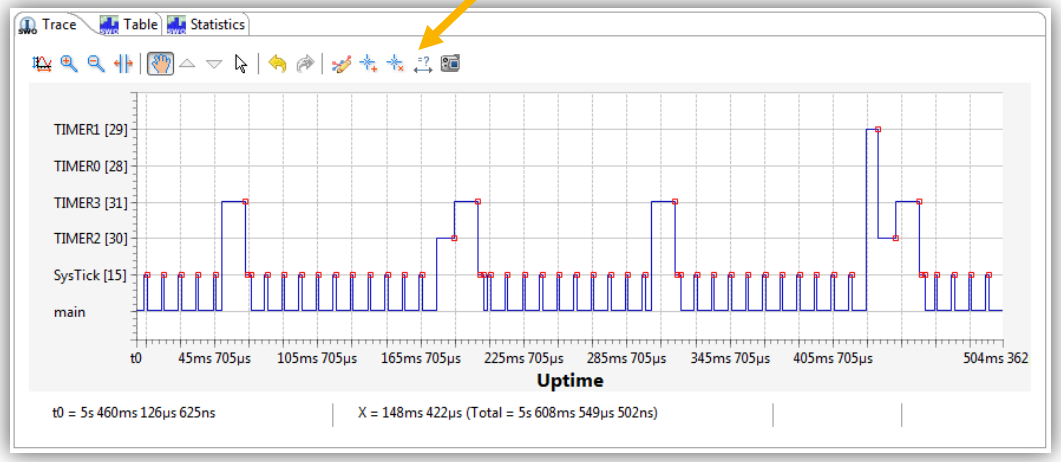
- Single graphing technology now used by both SWO Interrupts View and Global Variables View, providing a base for further future enhancements
- New tool to allow easy time measurement between points on graphs
- Global Variables View now uses tick boxes to allow more controlled selection of variables to be graphed

Tick box selection for graphing

Measurement tool



Measurement tool



SWO Interrupts View

LinkServer Flash Driver enhancements

- Most LinkServer flash drivers now implement a "verify-same" hashing mechanism to remove the need to erase/reprogram when the flash sector being programmed already has the required content
  - Provides noticeable improvement in debug startup time when carrying out multiple debug sessions of large, non-changing images, particularly via slower debug probes

First debug session

```
Flash variant 'K 64 FTFE Generic 4K' detected (1MB = 256*4K at 0x0)
Writing 77248 bytes to address 0x00000000 in Flash
00001000 done   5% (4096 out of 77248)
00002000 done  10% (8192 out of 77248)
00003000 done  15% (12288 out of 77248)
00004000 done  21% (16384 out of 77248)
00005000 done  26% (20480 out of 77248)
00006000 done  31% (24576 out of 77248)
00007000 done  37% (28672 out of 77248)
00008000 done  42% (32768 out of 77248)
00009000 done  47% (36864 out of 77248)
0000A000 done  53% (40960 out of 77248)
0000B000 done  58% (45056 out of 77248)
0000C000 done  63% (49152 out of 77248)
0000D000 done  68% (53248 out of 77248)
0000E000 done  74% (57344 out of 77248)
0000F000 done  79% (61440 out of 77248)
00010000 done  84% (65536 out of 77248)
00011000 done  90% (69632 out of 77248)
00012000 done  95% (73728 out of 77248)
00013000 done 100% (77824 out of 77248)
Sectors written: 19, unchanged: 0, total: 19
Erased/Wrote sector  0-18 with 77248 bytes in 3214msec
Closing flash driver FTFE_4K.cfx
Flash Write Done
Flash Program Summary: 77248 bytes in 3.21 seconds (23.47 KB/sec)
Starting execution using system reset and halt target
Stopped (Was Reset)  [Reset from Unknown]
Stopped: Breakpoint #1
```

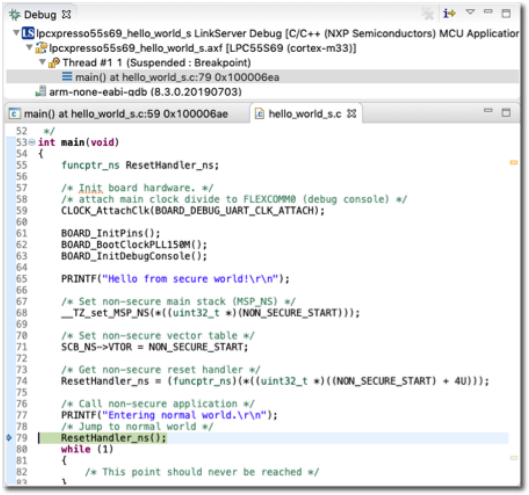
Second debug session

```
GDB nonstop mode enabled
Opening flash driver FTFE_4K.cfx (already resident)
Sending VECTRESET to run flash driver
Flash variant 'K 64 FTFE Generic 4K' detected (1MB = 256*4K at 0x0)
Writing 77248 bytes to address 0x00000000 in Flash
Sectors written: 0, unchanged: 19, total: 19
Erased/Wrote sector  0-18 with 77248 bytes in 69msec
Closing flash driver FTFE_4K.cfx
Flash Write Done
Flash Program Summary: 77248 bytes in 0.07 seconds (1093.30 KB/sec)
Starting execution using system reset and halt target
Stopped (Was Reset)  [Reset from Unknown]
Stopped: Breakpoint #1
```

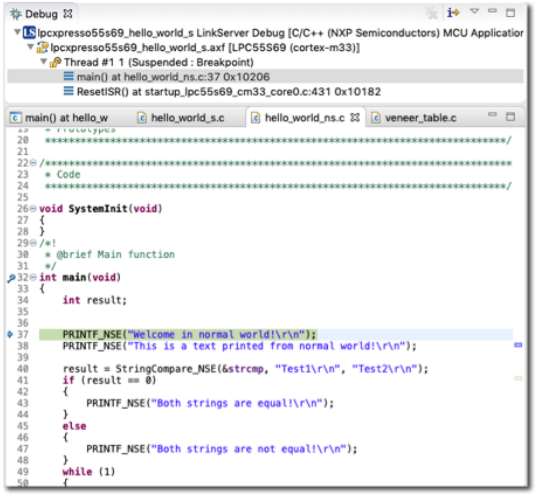
Example “debug messages” logs from FRDM-K64 with OpenSDA DAP-Link probe

Improved Cortex-M33 Secure / Non-Secure projects (LPC55xx)

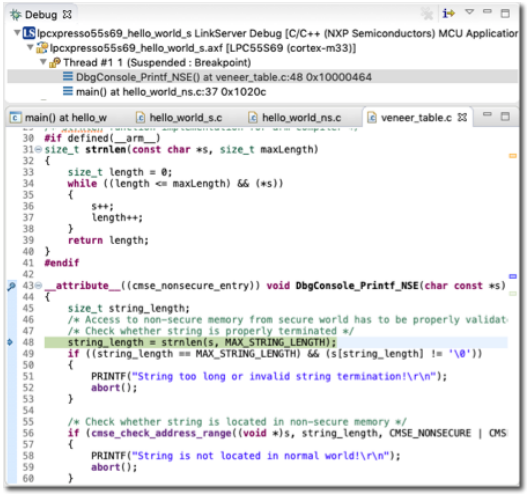
- Can now set source level breakpoints in CMSE functions on Secure side, to allow debugging of these functions when they are called from Non-Secure code



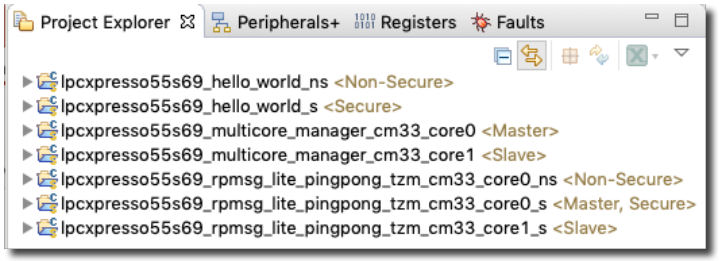
Secure to Non-Secure



Call back into Secure



- Fixed an issue with handling of combined secure / non-secure with multicore projects
- Project Explorer View now implements project decorators for Secure / Non-secure projects
  - Also Master / Slave project decorators (for all multicore devices)



Other debug improvements

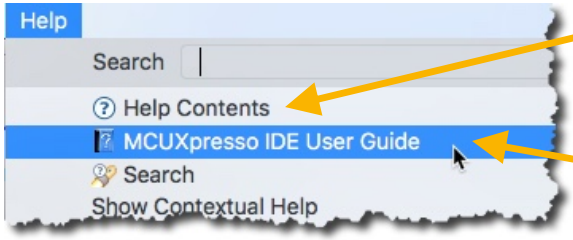
- Various fixes and enhancements to the Peripherals View, in particularly some registers being marked as inaccessible when single stepping
- Improved handling of certain variable types, along with signed versus unsigned variables, within the Global Variables View
- Fixed an issue with display of large arrays in Global Variables View with SEGGER and P&E Micro debug connections, when Live variables are enabled
- Fixed an issue with Disassembly View going blank after Restart, with LinkServer debug connections
- Fixed an issue with decoding correct stack pointer to be used in Faults View
- NXP LPC-LINK2 CMSIS-DAP firmware soft-loaded by IDE updated to v5.361

For more information …

- Extensive information can be found in the supplied MCUXpresso IDE v11.1 documentation, which can be
  - Accessed via the IDE's built-in Help System
  - Found in PDF format inside the product installation directory
  - Downloaded from NXP website

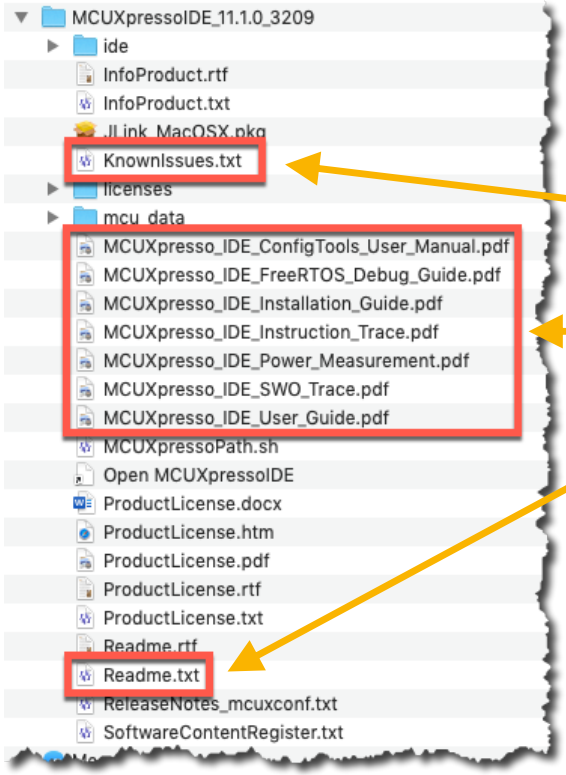
- Release notes containing a more extensive list of changes in MCUXpresso IDE v11.1 can also be found in the product installation directory
  - Or in "What's New" page of Welcome System

- For general product information and links to the product installers and documentation, visit: http://www.nxp.com/mcuxpresso/ide



Open docs in built-in Help System

Open User Guide



Browse to manuals and release notes inside IDE install directory

SECURE CONNECTIONS
FOR A SMARTER WORLD