Using the ARM Fast Model-based Cortex-M33 IoT Kit FVP

MDK Version 5

AN302, Summer 2017, V 1.1 Ken Havens ken.havens@arm.com

Contents

Introduction

This document describes a step by step process on how to use the ARM Fast Model based Cortex-M33 IoT Kit FVP with the MDK toolchain. While we've tested these implementations, there will be updates to the tools and FPGA images. Expect differences between these and follow-on implementations.

Prerequisites

All the ARMv8-M support is now in our standard MDK product as of version 5.23 and later. Just perform the normal MDK installation.

To build and run the examples, you'll need the "CMSIS.5.0.1 (2017-02-03)" pack and "V2M-MPS2_IOTKit_BSP 1.3.0 (2017-03-10) packs available via the pack installer in MDK. These packs have the latest support for the MPS2+ (V2M-MPS2-0318C) board running the Cortex-M33 IoT FPGA image.

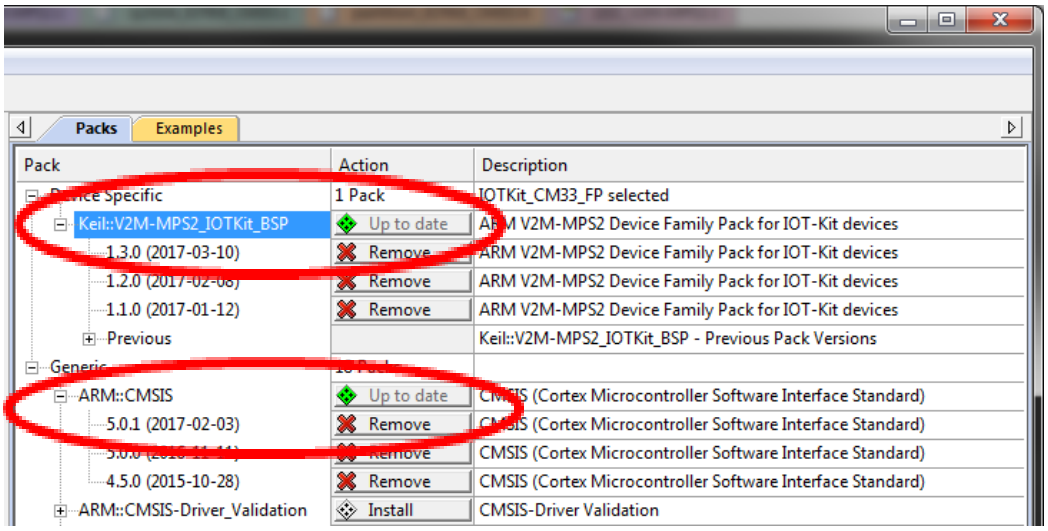
The example project introduced in this document requires the Windows built-in telnet.exe to be activated. For some Windows version, such as Windows 10, telnet.exe is not activated by default. After enabling the Windows built-in telnet.exe, it can be found via the path C:\Windows\System32\telnet.exe.

Using the ARM Fast Model based Cortex-M33 IoT Kit FVP with MDK

Verify the Pack Installation

1. Let's double check that the CMSIS.5.0.1" and "V2M-MPS2_IOTKit_BSP 1.3.0" packs are both installed properly. We'll use the pack installer from MDK. Click on the pack installer icon from the icon bar…

2. Then make sure you have the latest packs loaded, CMSIS.5.0.1-dev5.pack and V2M-MPS2_IOTKit_BSP.1.3.0.pack packs by checking the versions listed and that the “Up to date” button is shown next to each one…

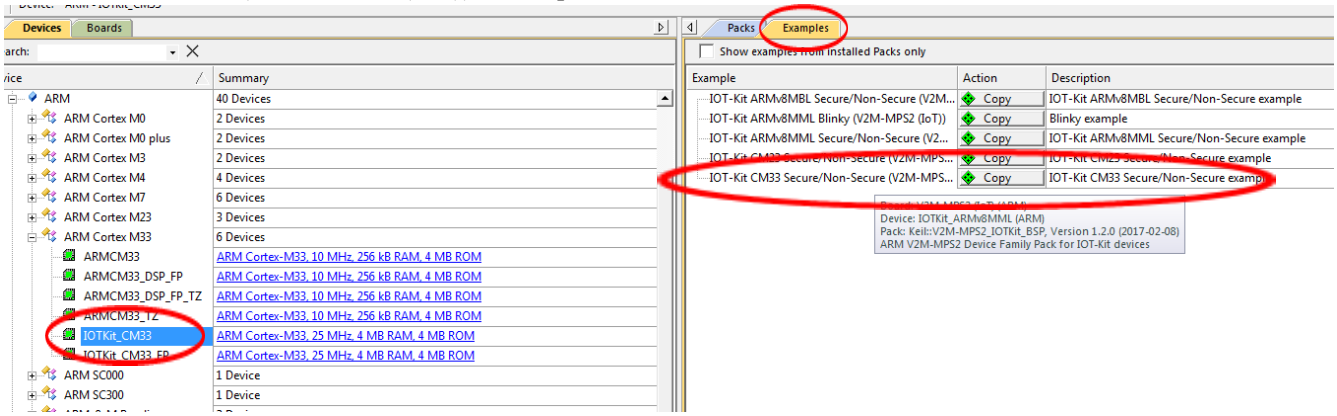


Copy, Update the Target and Run the Example Application

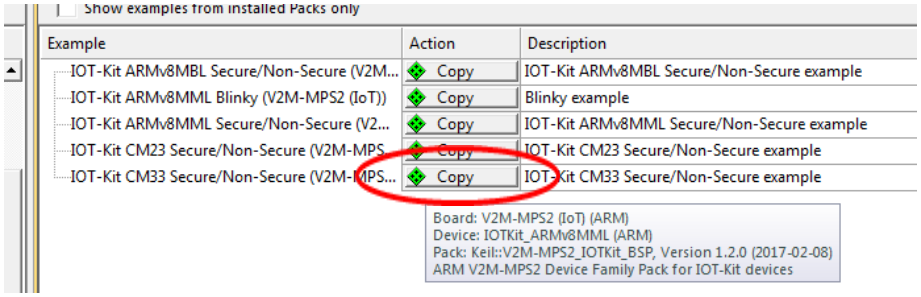
Once you have these packs loaded you can go to the examples tab and export the latest "TrustZone for ARMv8-M RTOS" (µVision Simulator)" example…

1. On the left side, click the "Devices" tab, then select "ARM" then "ARM Cortex-M33" then "IOTKit_CM33"as the device.

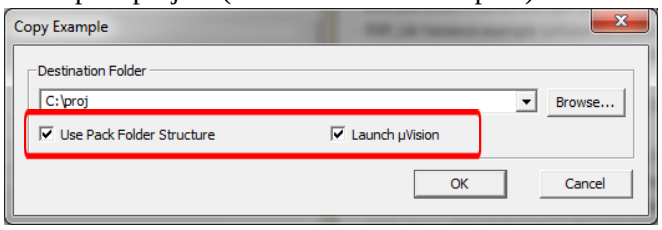
2. On the right side, switch to the "Examples" tab. If needed, scroll down till you see the "IOT-Kit CM33 Secure/Non-Secure (V2M-MPS2 (IoT))" example...



Click the "Copy" button next to the "IOT-Kit CM33 Secure/Non-Secure (V2M-MPS2 (IoT))" example project…



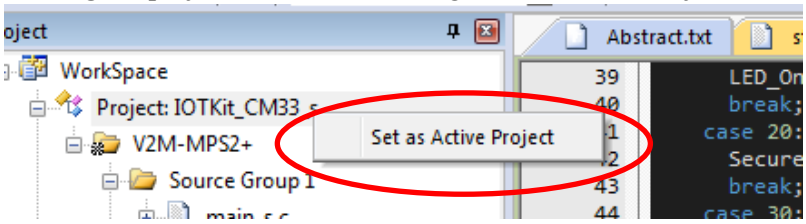
3. …for running the example application, it is more convenient to check the box to 'Launch µVision' with the copied project (Note the destination path)…



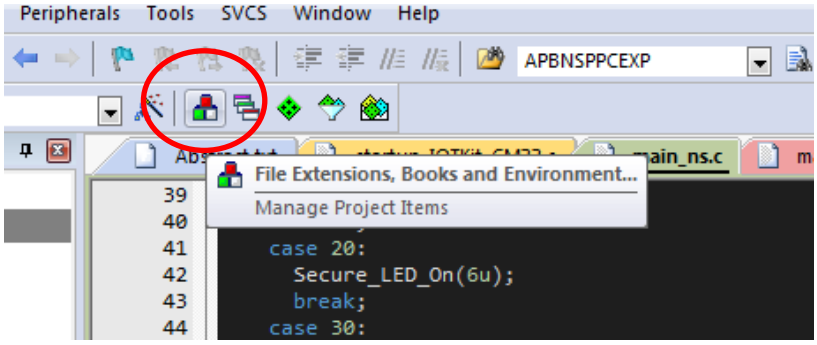
4. Close the Pack Installer and, if you didn't select the 'Launch µVision' check box, open the newly downloaded project in MDK by double clicking the "IOTKit_CM33_s_ns.uvmpw" multi-project file in the folder located in the following directory path noted above…
<load point>\proj\Boards\ARM\V2M-MPS2\IOTKit_CM33\IOTKit_CM33_S_NS\

Adding the ARM Fast Model based Cortex-M33 IoT Kit FVP as a debug target

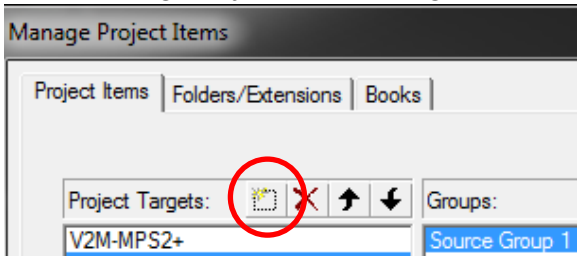
1. First, check to make sure secure project, "IOTKIT_CM33_s", is the current active project by right clicking the project name and selecting "Set as Active Project" button…



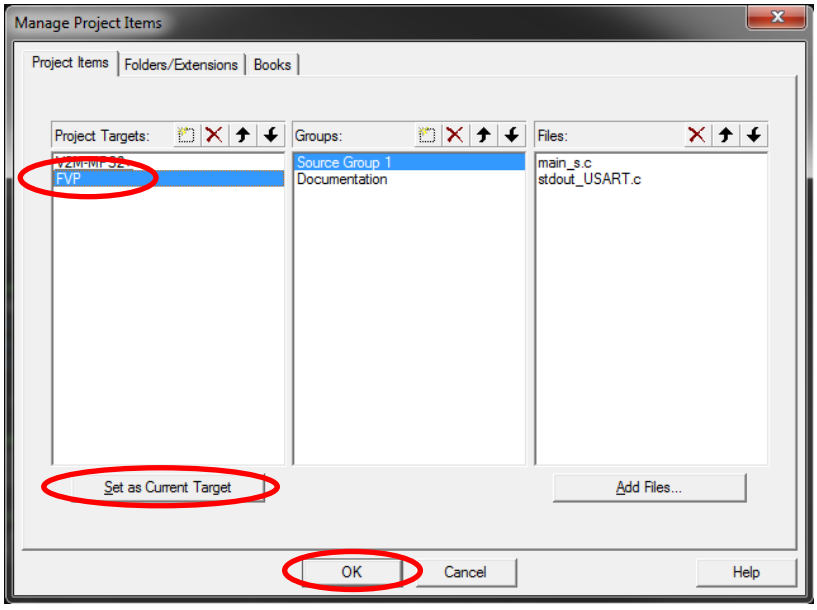
2. To add additional targets to the project, click the "Manage Project Items" icon near the top of the µVision window…



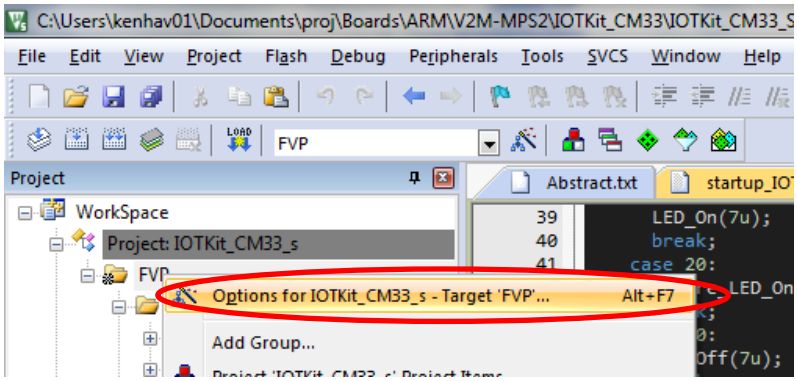
3. In the "Manage Project Items" dialog box, add a new target by clicking the "New (Insert)" button…



4. This will add a blank line to fill in the name of the new target. In the example below I typed "FVP" then enter. Once the new name is highlighted, click the "Set as Current Target" button at the bottom of the dialog and then the "OK" button…

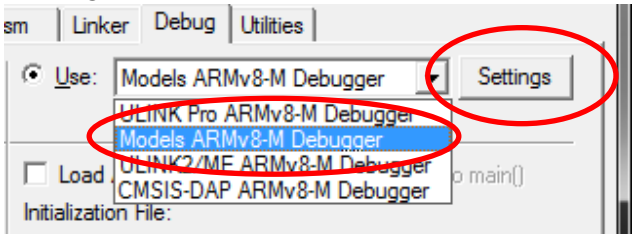


5. Now the FVP target should be seen as the current target in the project tab as below. As the MPS2 settings were copied over when we created the new target, we need to update the debug settings to accommodate using the FVP. Right click on the "FVP" target in the Project window and select "Options for Target for IOTKit_CM33_s – Target FVP" as shown below…

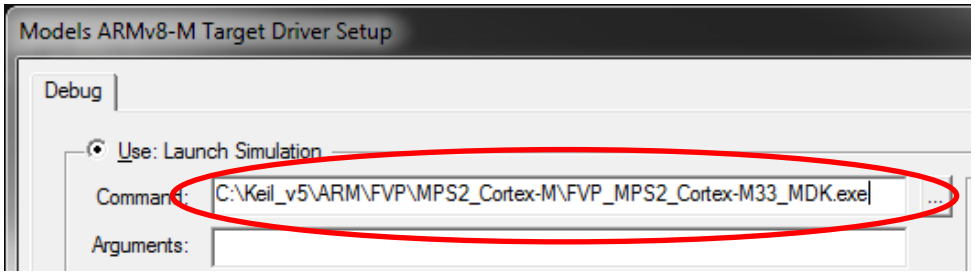


6. In the "Options for Target FVP" dialog, click the "Debug" tab to open the debug settings view…

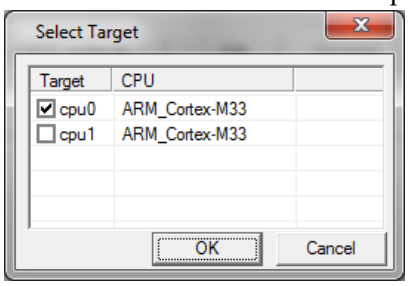
7. In the drop down selection for the debugger, select "Models ARMv8-M Debugger", then click the "Settings" button next to it…



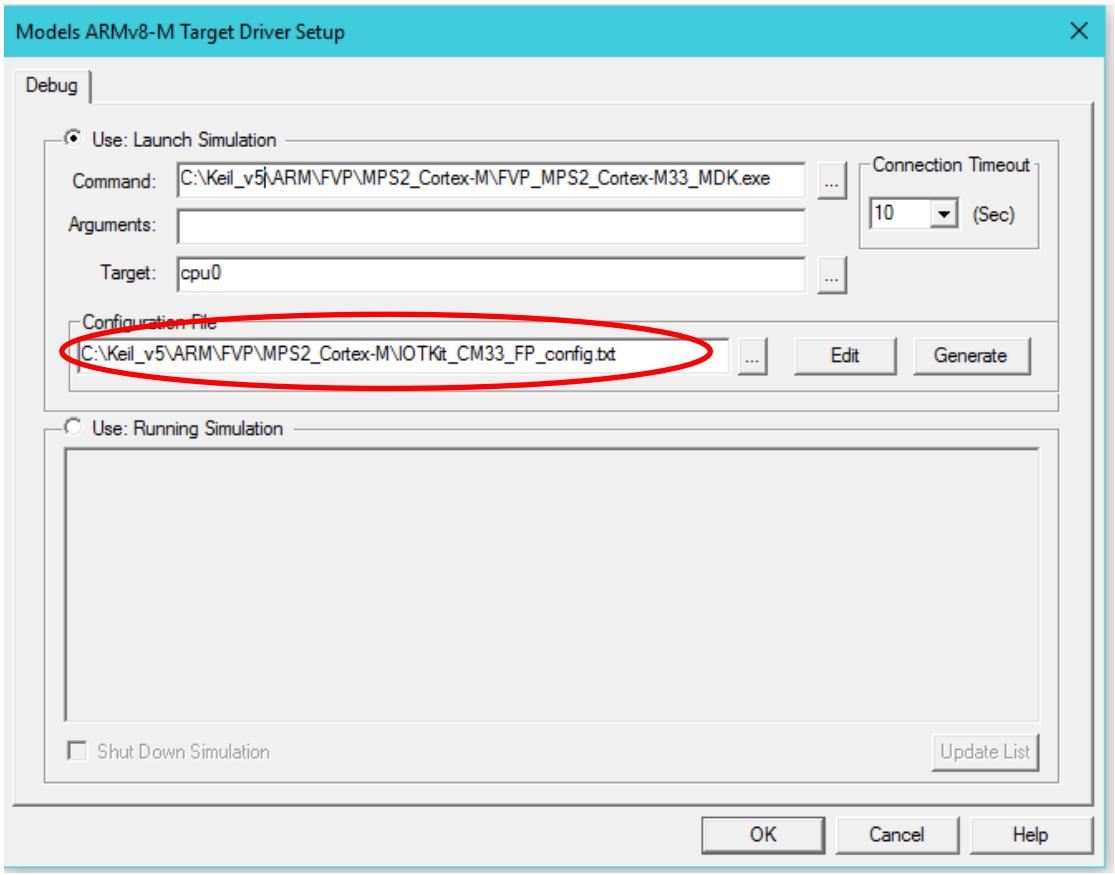
8. You should have the "Models ARMv8-M Target Driver Setup" dialog box open now. Here we need to make a few entries. Keep in mind your paths may vary based on your system preferences. First let's point to the actual FVP executable, "FVP_MPS2_Cortex-M33_MDK.exe" by clicking the ellipsis button … to browse to the file location in the MDK install directories similar to the path below…



9. Next we need to update the “Target:” box with the CPU we want to connect to which in this case is “cpu0”. Click on the ellipsis button at the right of the “Target:” box to display the available CPU selections available and select “cpu0” then click “OK”…

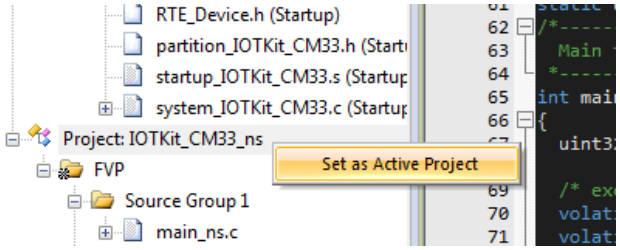


10. In the “Configuration File” box, select the “IOTKit_CM33_FP_config.txt” configuration file from the same location of the file FVP_MPS2_Cortex-M33_MDK.exe mentioned above, which is similar to the below…



If your setup box is similar to the above, click “OK” on both dialog boxes to save the changes and close them out.

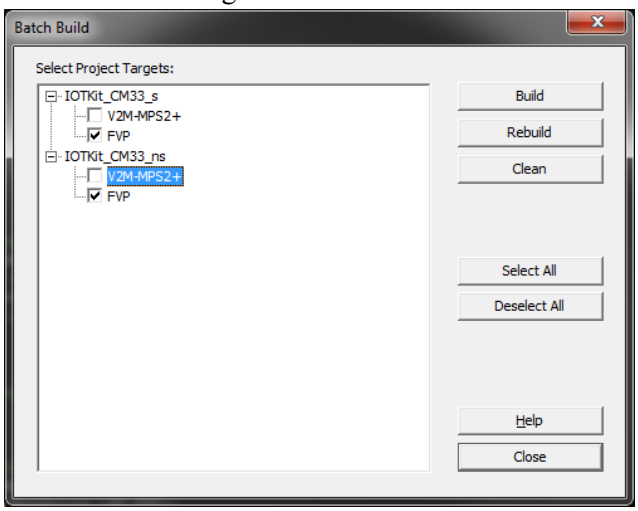
11. Repeat this same procedure for the non-secure project, “IOTKit_CM33_ns”, by right clicking the project name and selecting “Set as Active Project” button on the non-secure project…



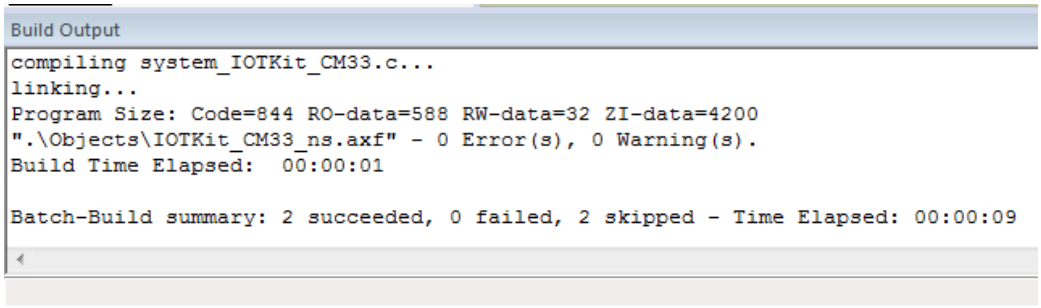
12. Follow the same process as in steps 2 through 10 above. Note that both projects can use the same configuration file in step 10.

Let's build the projects and debug the FVP target

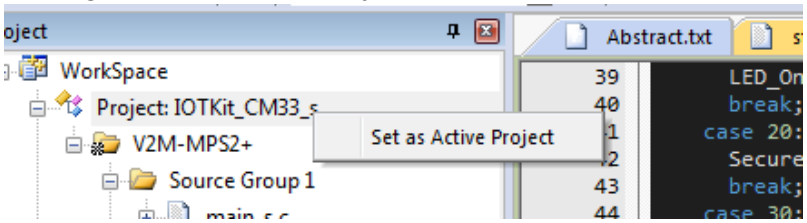
1. Click the "Batch Build" button  on the icon bar near the top left of the MDK window to open the "Batch Build" dialog box. In the below example I've deselected the "V2M-MPS2+" projects to save on build time. This is optional and only affects the amount of build time. In this simple project the build time is minimal but, can become more significant as your project becomes more complex. In this case make sure the "FVP" targets are selected for both the secure and non-secure targets. Then click "Rebuild"…



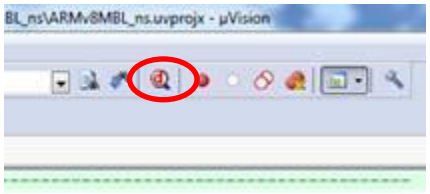
2. In the "Build Output Window at the bottom you should see both projects built…



3. Now let's start a debug session using the FVP. First make sure you start the debug session from the secure project. This is important in our example as the secure project initializes the memory and security settings. Again, right click the secure project, "IOTKIT_CM33_s", and make it the active project by clicking the "Set as Active Project" button…

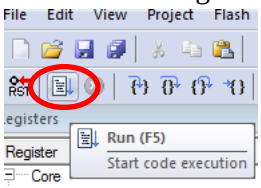


4. Click the Debug button at the top to enter the debug session…

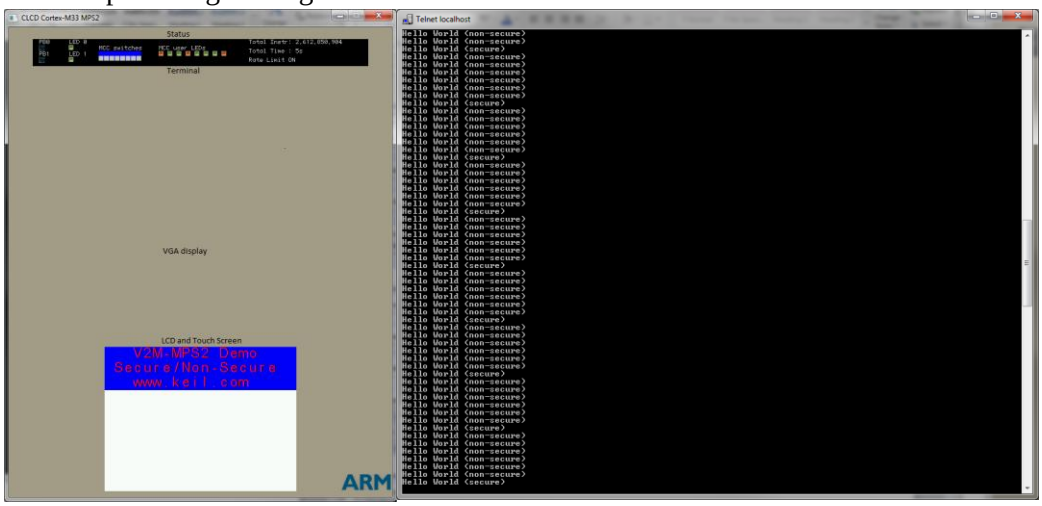


5. At this point you should see the FVP open in another window on your machine similar to the below screen capture…

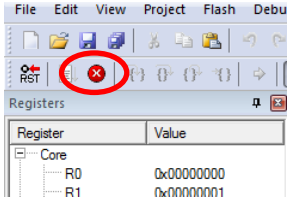
6. Back in the debug session, click run...



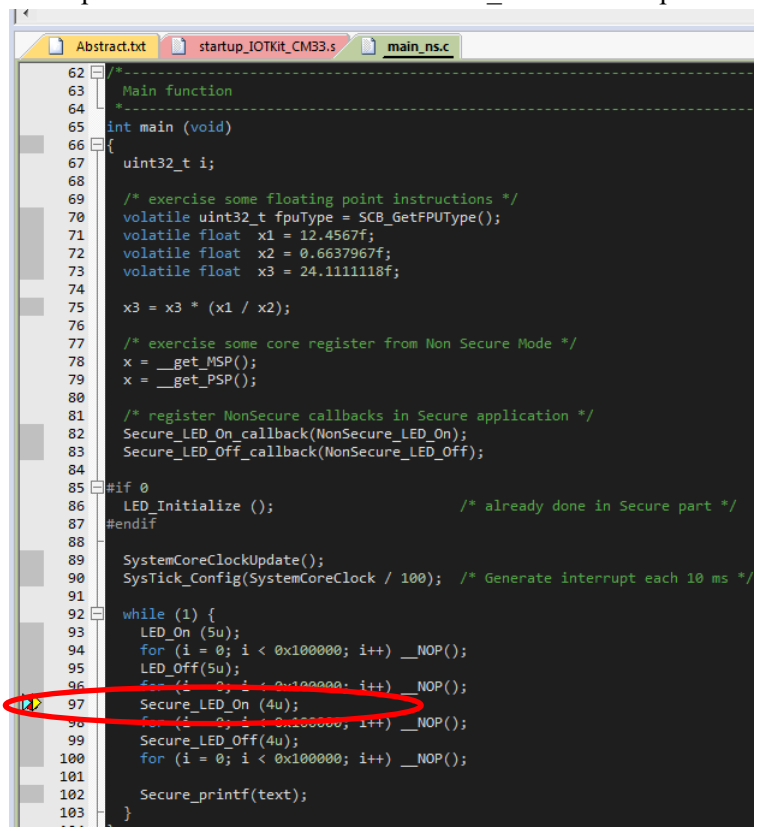
7. You should see the FVP window's LCD screen update, the LED lights begin to blink and another Telnet window providing messages via a UART connection from the FVP…



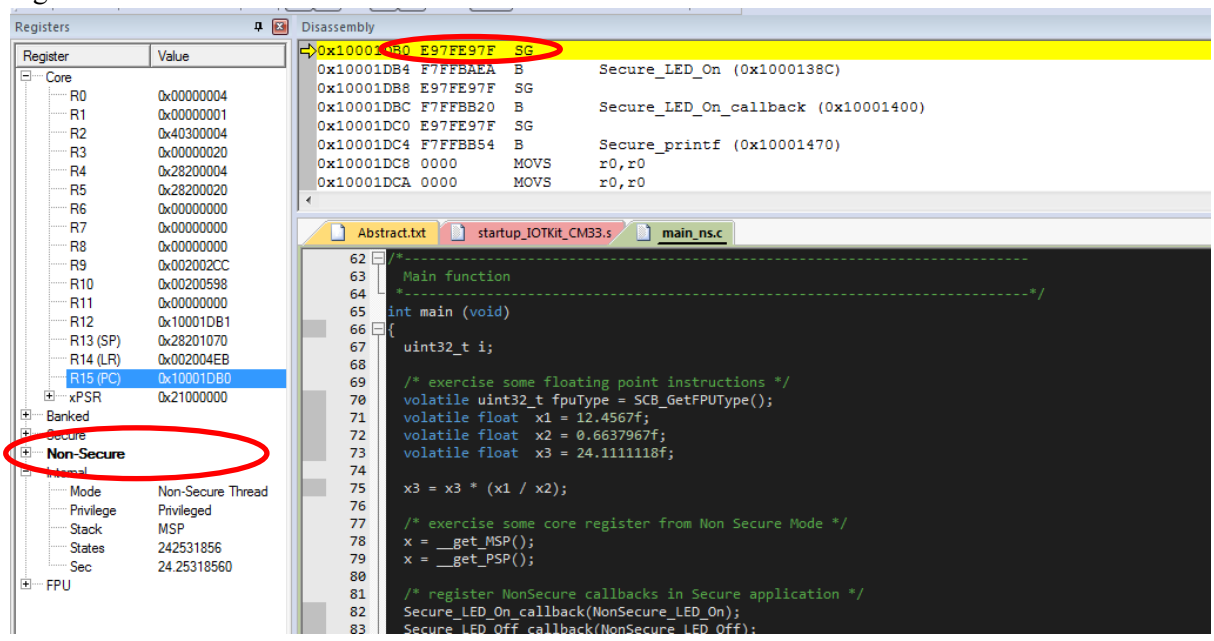
8. Press stop…



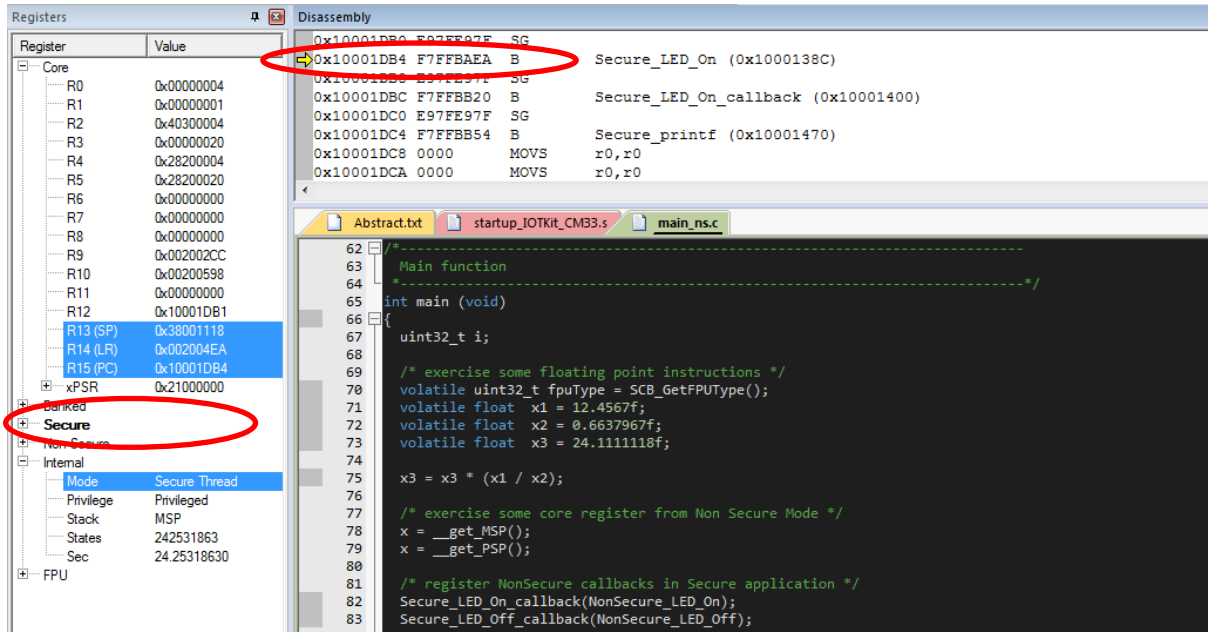
9. As an example let's walk through the process of going from non-secure to secure and back, by placing a break point at or near line 97 in the main_ns.c file then press "Run" to stop at that breakpoint…



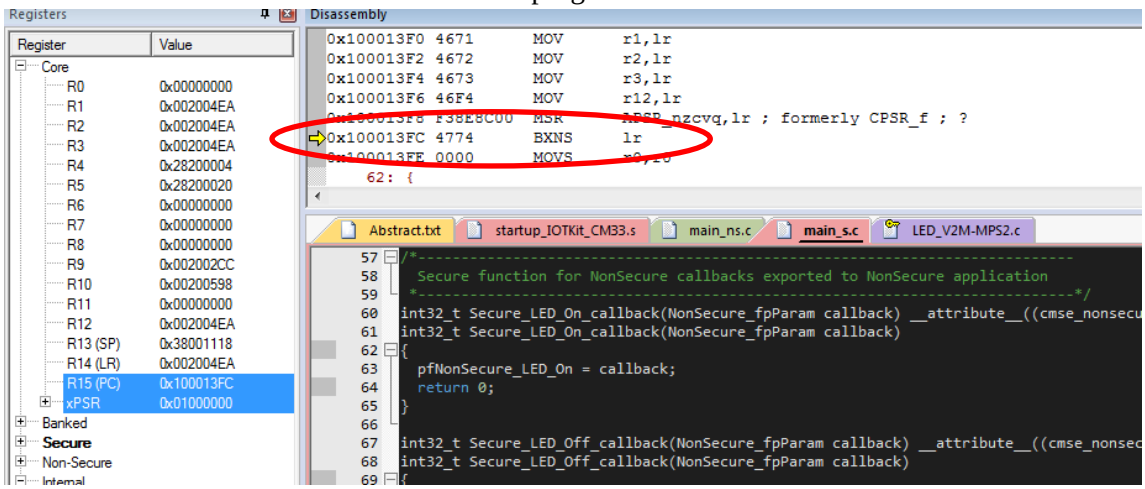
10. From this point, click into the "Disassembly" window and single step (F11) through the code. After around five steps you'll see the "SG" instruction which is the secure gate instruction. After executing that instruction you should see the debugger update from "Non-Secure Thread" to "Secure Thread" in the register window…



After the “SG” Instruction…



11. Continuing the single step in the disassembly window for approximately 23 steps and you should see the branch back to non-secure instruction “BXNS” which when executed will take you back to the “Non-Secure Thread” state in the non-secure main program.



12. More information about CMSIS-Core for ARMv8-M can be found in the included documentation in the Manage RTE component “CMSIS-CORE for Cortex-M, SC000, SC300, ARMv8-M” or by browsing to it in your installation at…
    <install point> /ARM/PACK/ARM/CMSIS/5.0.1/CMSIS/Documentation/Core/html/index.html
13. For more information about the Cortex-M33 IoTKit FPV look in the following directories of the Cortex-M33_IoT_kit_2_0.zip file…
    \Cortex-M33_IoT_kit_2_0\app_notes\AN505\docs\ DAI0505A_example_iot_kit_subsystem_for_v2m_mps2.pdf
    \Cortex-M33_IoT_kit_2_0\boards\Docs\ARMv8-M_IoT_Kit_UG.pdf
    While these both are for the MPS2 FPGA image, they are also applicable to the FVP

This is a very simple example but I hope it helps you understand Secure and Non-secure operations better.

Some Notes

1. Note that the MDK tools do two incremental loads before starting, one each for the secure and non-secure domains.

This is a very simple example but I hope it helps you understand Secure and Non-secure operations better.