

CCS: Download and Debug via JTAG with one mouse click

How to create a target configuration for your UCD / JTAG emulator pair

(You can use an existing target configuration if you have one created already).

In CCS select “View -> Target Configurations”

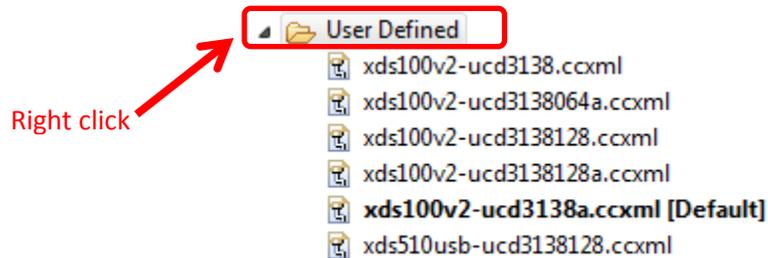


Figure 1

Right click on “User Defined” (see Figure 1), Select “New Target Configuration”

The pop up window (Figure 2) will appear.

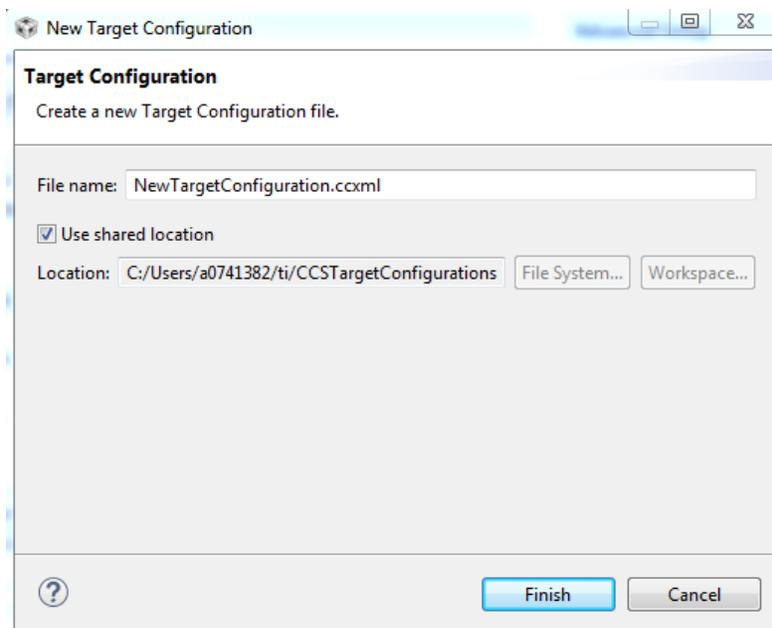


Figure 2

Give the configuration a meaningful name for your UCD device / JTAG emulator combination,

Click “Finish”.

Now you need to associate a UCD device and JTAG emulator with the configuration.

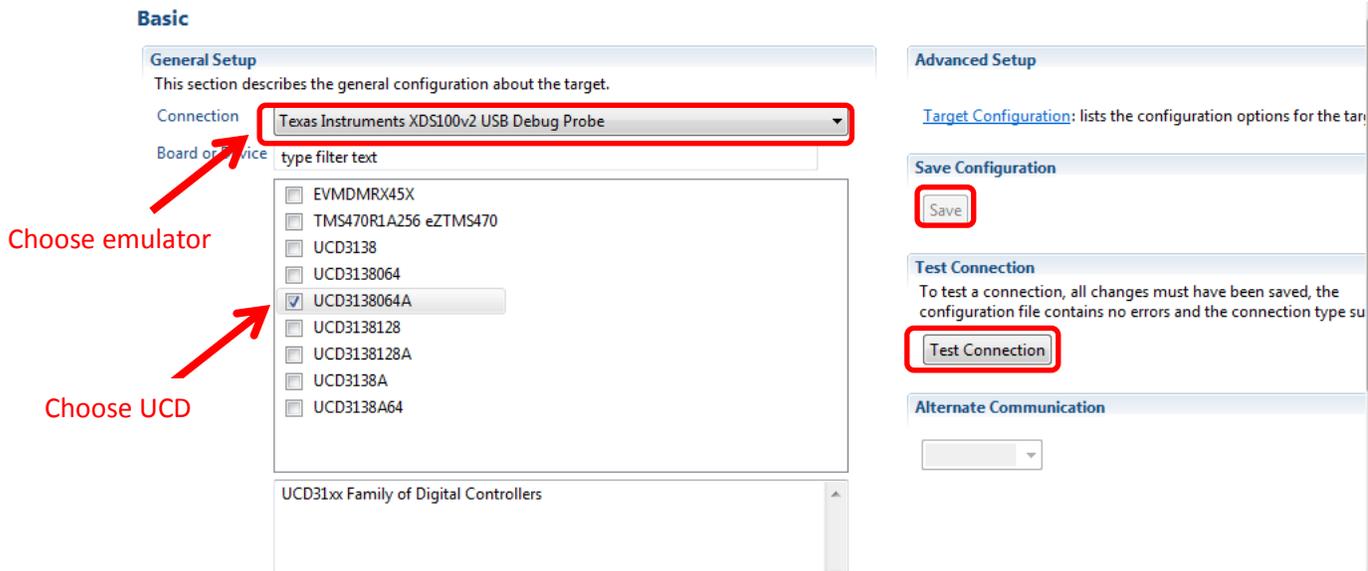


Figure 3

You should now see the window in Figure 3. Under "Connection", choose your JTAG emulator. Under "Board or Device", choose your UCD device type. Click "Save". Click "Test Connection" to check the physical connection from CCS to your UCD device via the JTAG emulator. A pop up menu will appear (Figure 4), scroll to the bottom and you should see the text highlighted in red if the test passes.

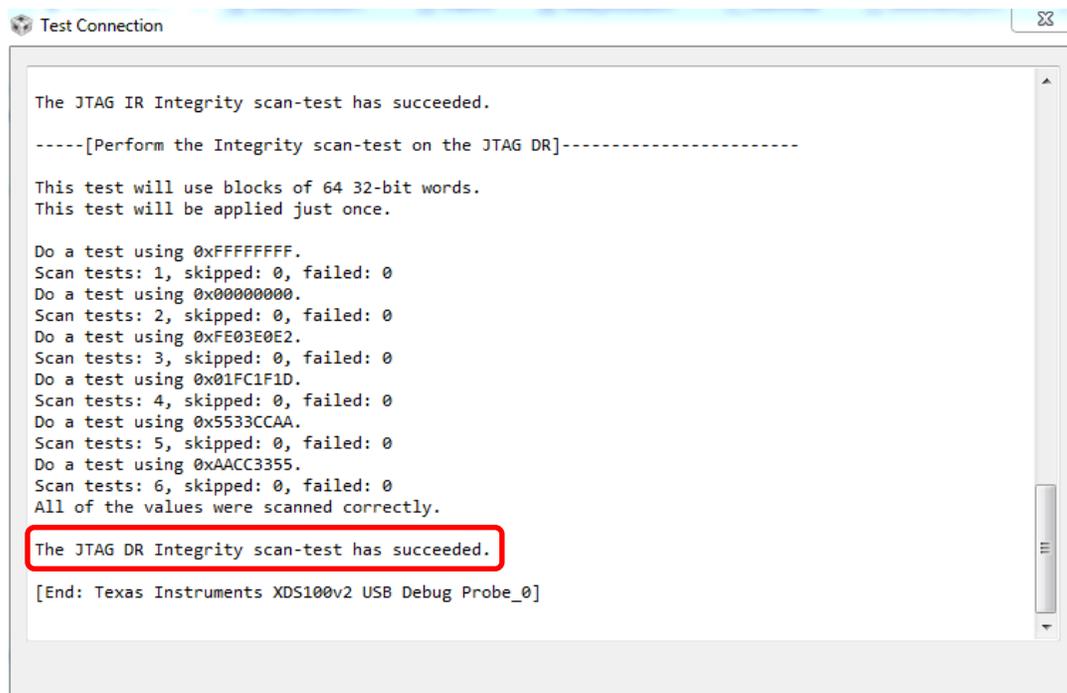


Figure 4

Associating a target configuration with a project

You now want to associate this target configuration with a project.

Go back to the user defined menu (see Figure 1). Right click on the configuration you wish to associate with a project, in the pop up menu that appears select “Link File To Project”, and then select the project you wish to link this configuration to.

In my case, I’ve linked the configuration “xds100v2-ucd3138064a.ccxml” to the project “bootflash”.

Go to the project explorer window and expand the project view (see Figure 5).

You should see the JTAG configuration that you just linked to your project (i.e. the .ccxml file) as part of your project view (highlighted in red at the bottom of Figure 5).

Right click on the .ccxml file, and select “Set as active target configuration”

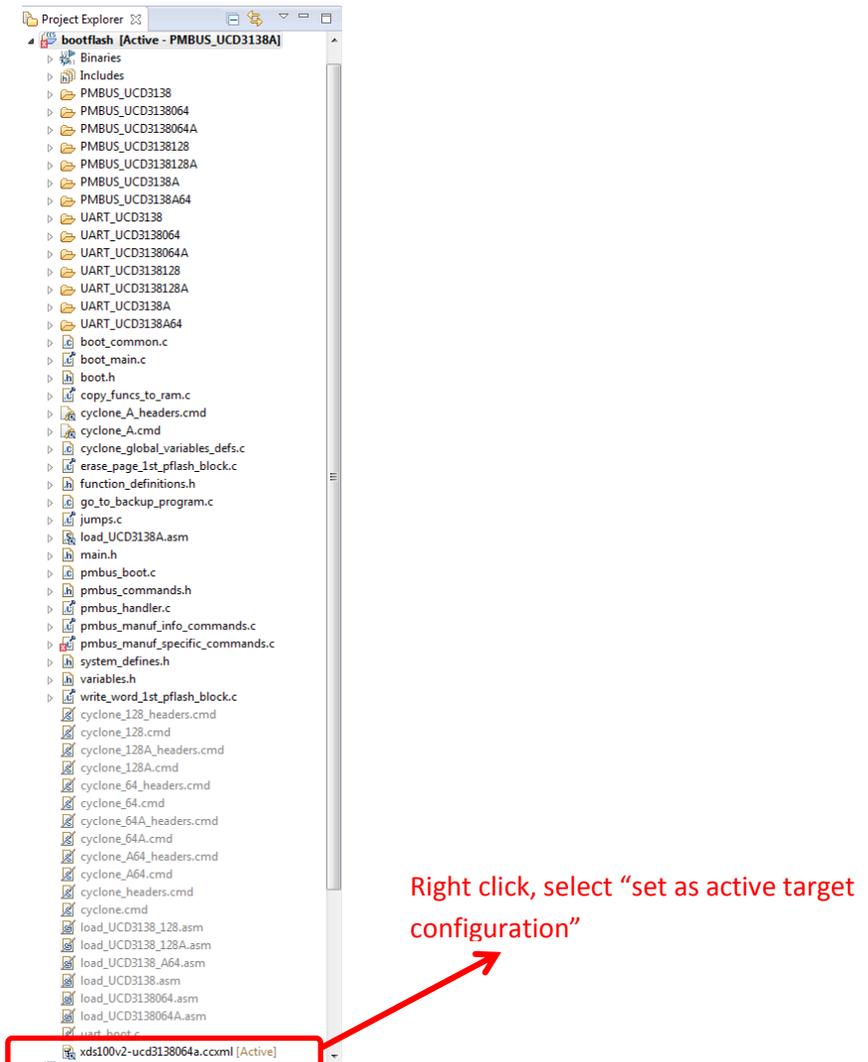


Figure 5

Now go back to the project explorer window (Figure 6). Right click on the project you have just linked to, and select “Properties”. Or press “Alt + Enter” together.

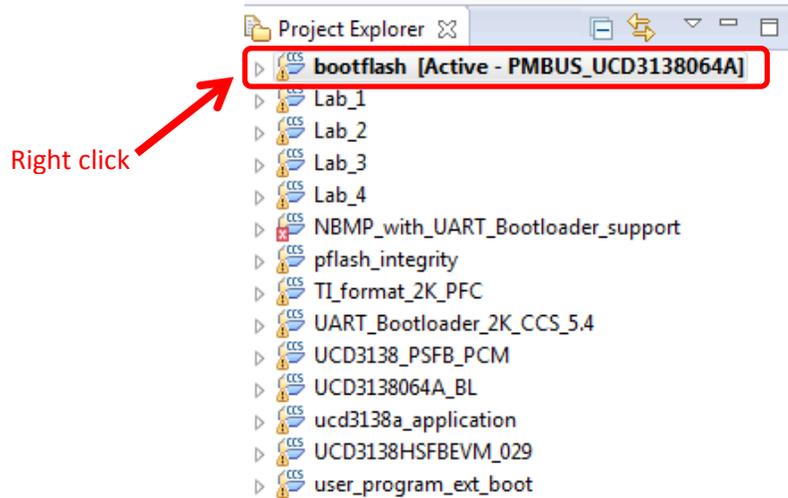


Figure 6

In the window that appears, select “debug”. Go to “Program/Memory Load Options”. Ensure that “Reset the target on a connect” is not selected. Select the option “Reset the target on a program load or restart”

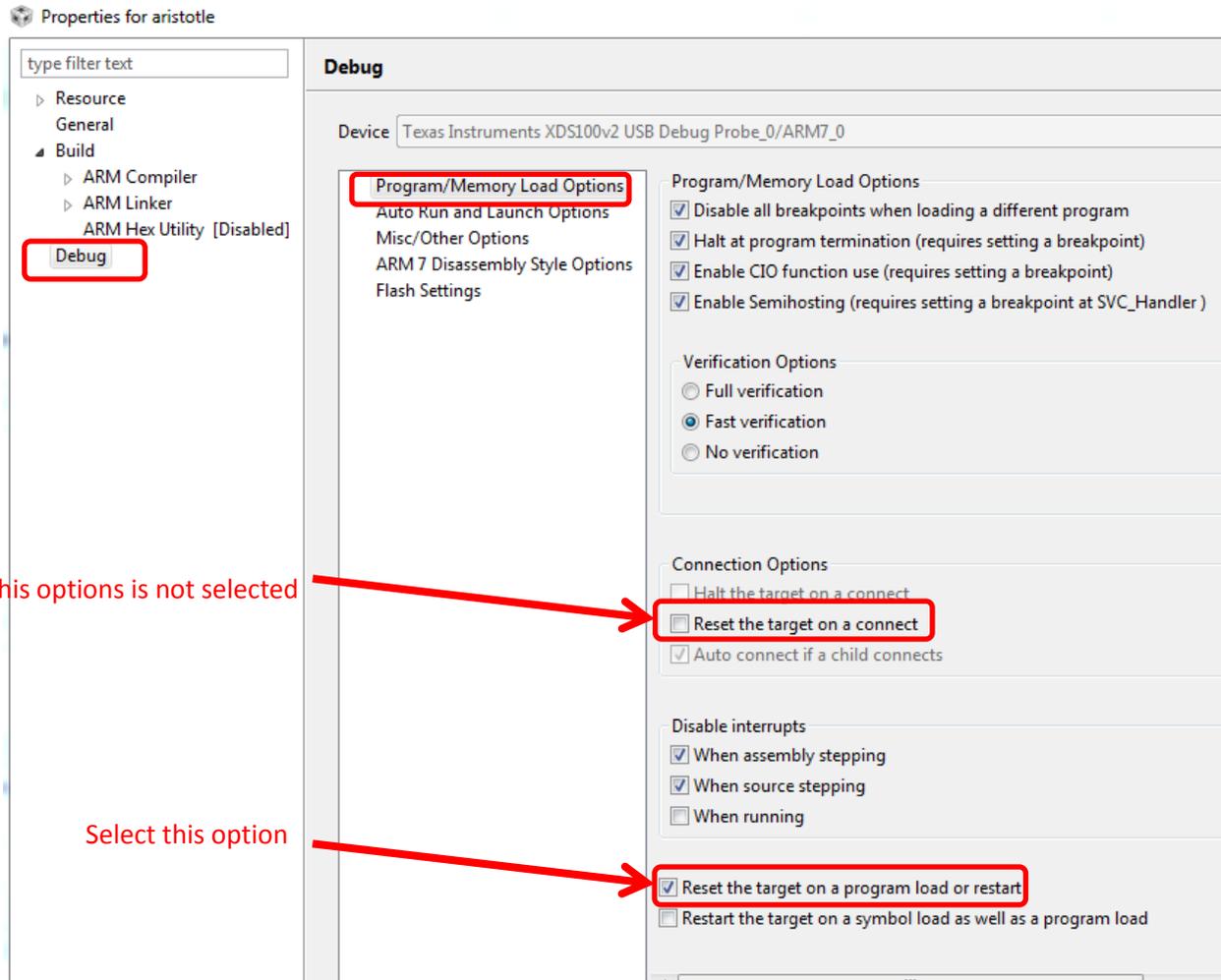


Figure 7

Go to “Auto Run and Launch options”. Clear the text in the text box opposite “Run to Symbol” (see Figure 8, and uncheck the checkbox “on a program load or restart”

Under “Launch options” select the checkbox “Connect to the target on debugger startup”

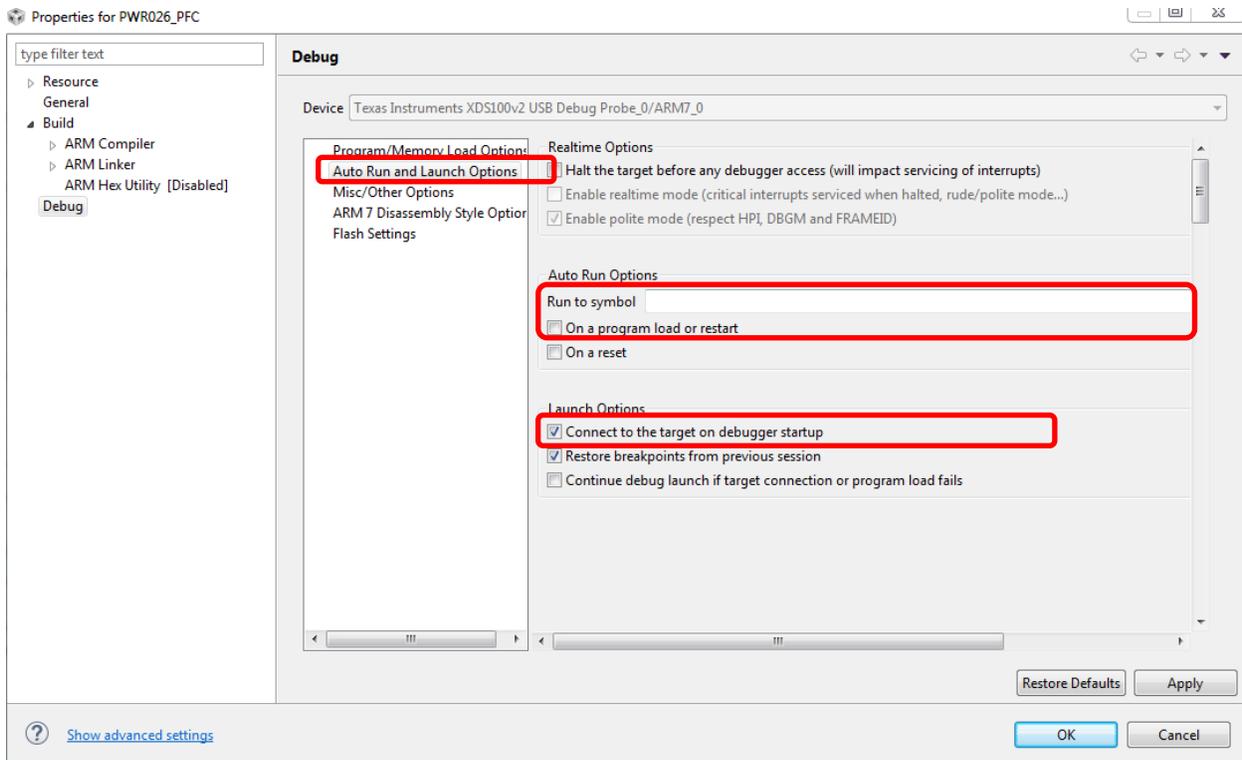


Figure 8

Go to “Flash Settings” (see Figure 9).

Select the settings you want for loading firmware (like block number, write checksum, boot support, data flash settings etc.).

Ensure that “Perform Blank Check before Write” is unchecked. This is required, as if this is checked, the flash programming routine does a blank check before erasing the flash area that is to be programmed. If the part has not been pre-erased the blank check will fail.

If you uncheck this checkbox, the blank check is executed, but the area of flash is still erased before programming. Selecting “Perform checksum based verification after load program” will ensure that the code has been downloaded successfully anyway. In a production environment, we recommend that you check “perform blank check before write”. You will most likely be using Uniflash anyway in a production environment, where you will be explicitly pre-erasing the flash before programming.

Check “Write Checksum” if you wish to write the checksum, so that the device remains in FLASH mode after a Power-On / Reset (POR). Not that if you choose to write the checksum, you will lost the ability to communicate over JTAG after the next POR unless the following line is at the beginning of your program (select one of the four options below based on the device you are using). This is because if a valid checksum is present, the ROM code changes the relevant bit in the IOMUX register to prevent JTAG comms (this was added as a security feature)

```

// enable JTAG: all devices
MiscAnalogRegs.IOMUX.all = 0;

// enable JTAG: UCD3138, UCD3138064
MiscAnalogRegs.IOMUX.bit.JTAG_CLK_MUX_SEL = 0;

// enable JTAG: UCD3138A64, UCD3138128
MiscAnalogRegs.IOMUX.all &= 0xFD; // clear bit 2

// enable JTAG: UCD3138128A
MiscAnalogRegs.IOMUX.bit.JTAG_MUX_SEL = 0;

```

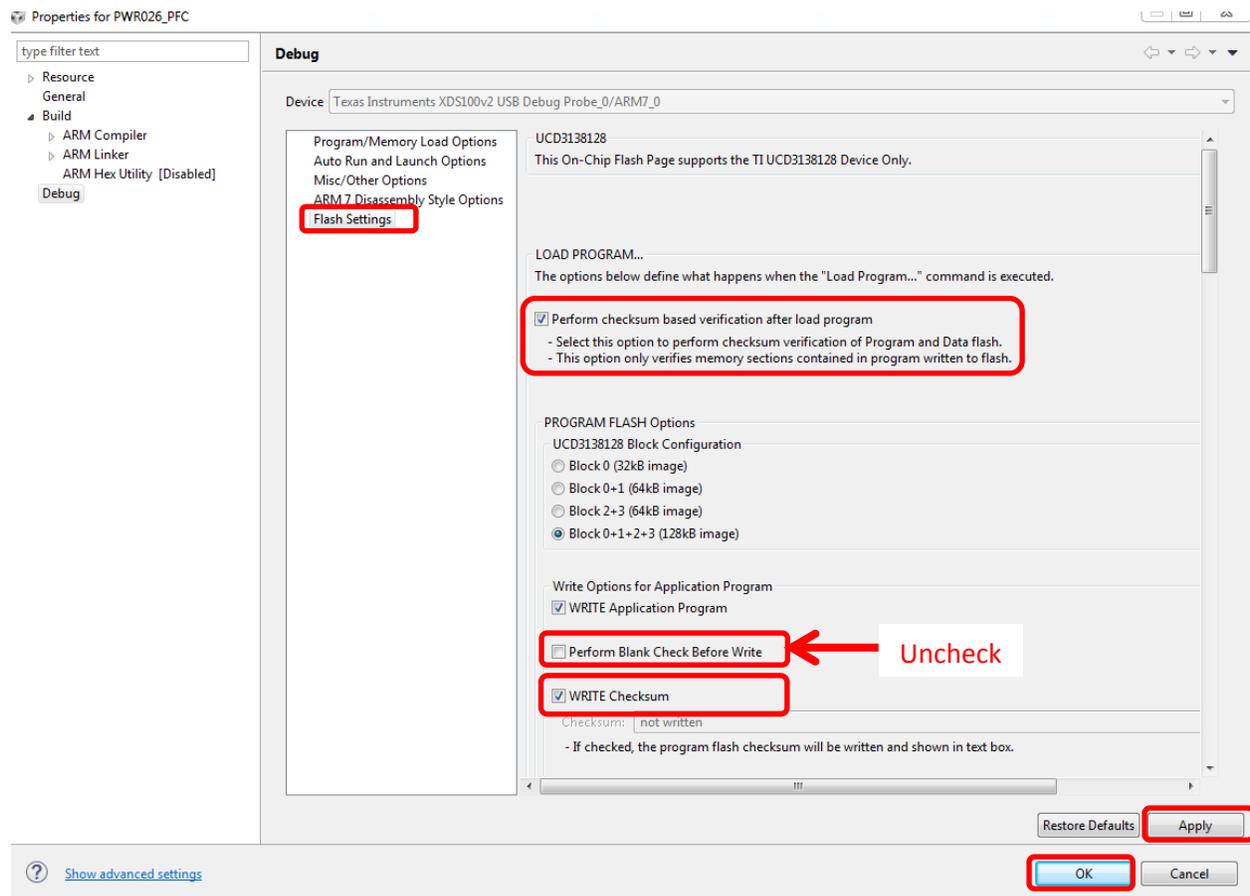


Figure 9

Scroll to the bottom of the “Flash setting page”. If you wish, select “Enable Verbose Mode” to see information on the console during download (see Figure 10).

To save your settings, click “Apply”, then “OK”. All of the above only needs to be done once.

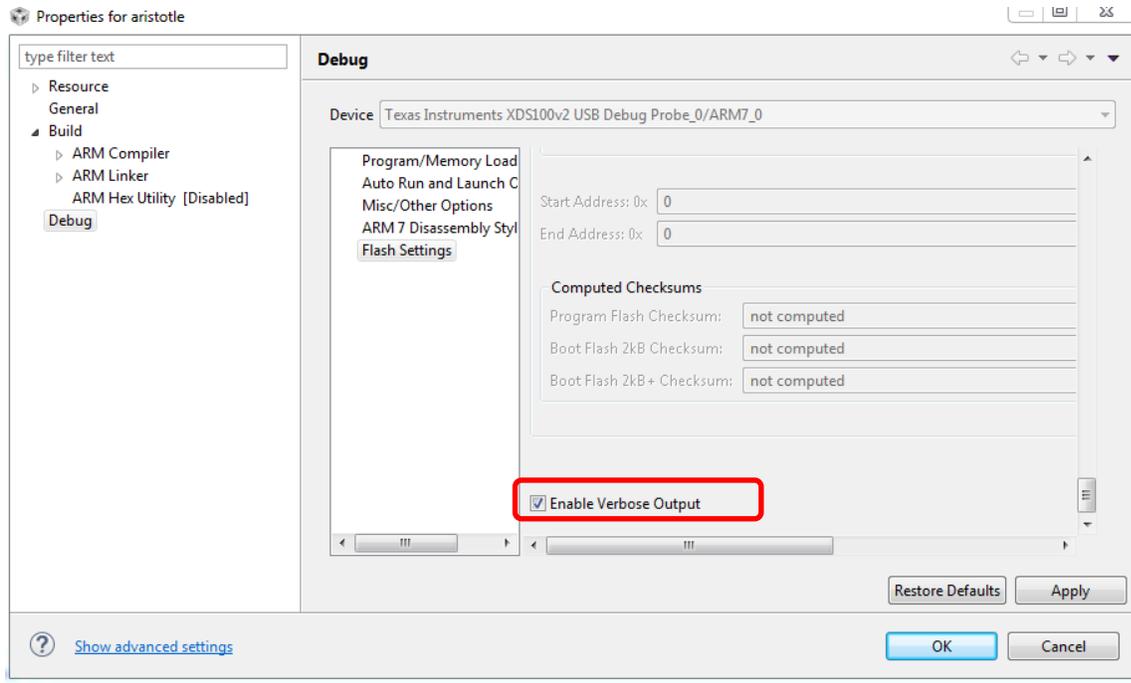


Figure 10

Download and debug with one click

To build, download and start the JTAG debugger, select the project in the project explorer window and left click on the bug icon in the toolbar (see Figure 11). Or just press **F11**.

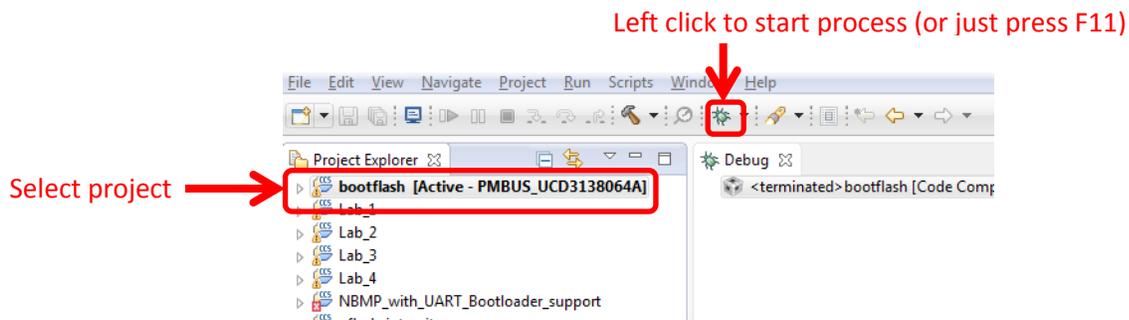


Figure 11

Now you can use the JTAG debugger to run, single step, set breakpoints, look at memory, look at registers, use disassembly view, etc. (Figure 12)



Figure 12