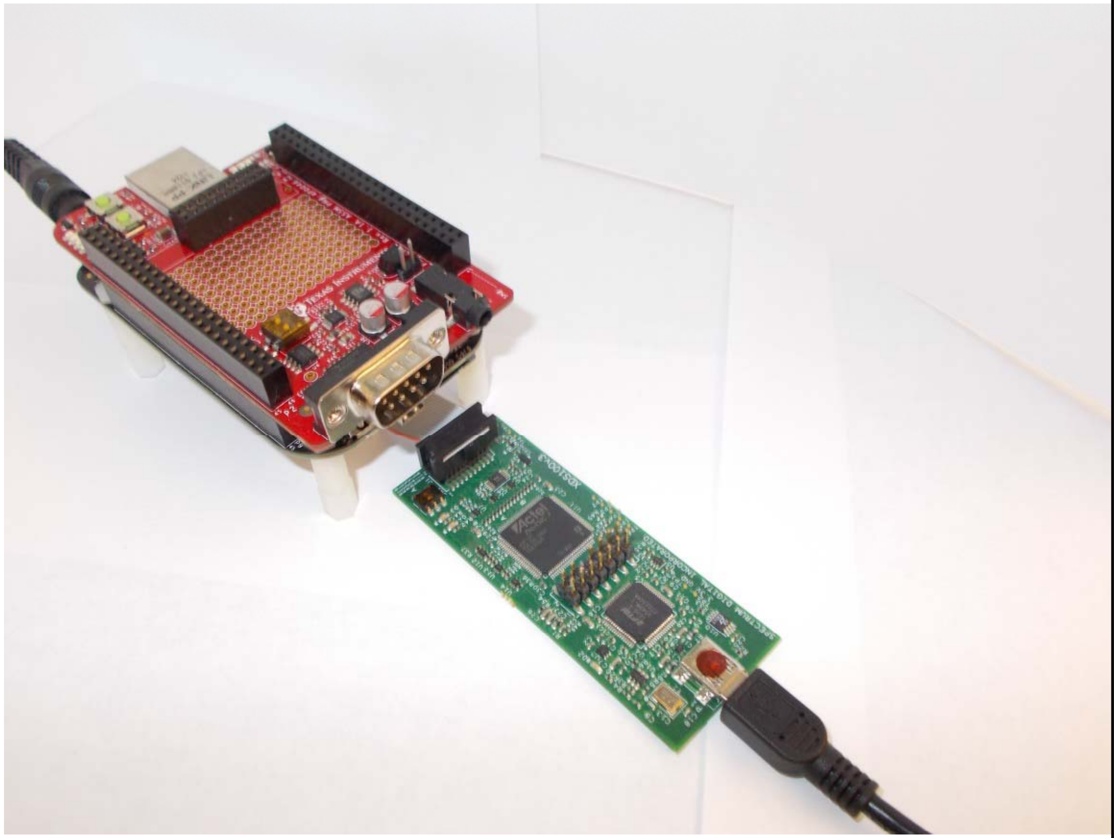
Debug PRU with Code Composer Studio

George Mock

Hardware: BeagleBone Black

- BeagleBone Black (BBB)
  - AM335x device
  - 2 PRU’s embedded in AM335x
  - Add JTAG header to P2 on back
    - Optional standoffs
- Debug Probe
  - XDS100 or XDS200
  - Screen shots show XDS100v3
- PRU Cape
  - Add-on board for learning about PRU
- BBB only info is marked

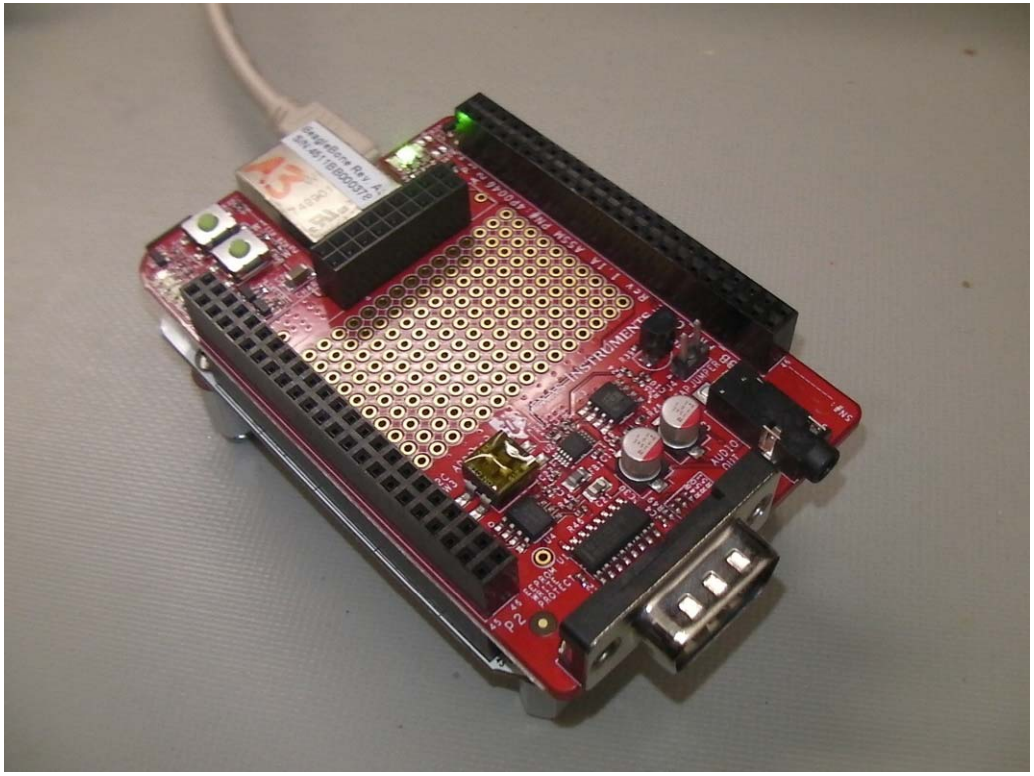
BBB Only



Hardware: BeagleBone White

- BeagleBone White (BBW)
  - AM335x device
  - 2 PRU's embedded in AM335x
  - More expensive than BBB
- Debug Probe
  - XDS100v2 built-in
- PRU Cape
  - Add-on board for learning about PRU
- BBW only info is marked

BBW Only

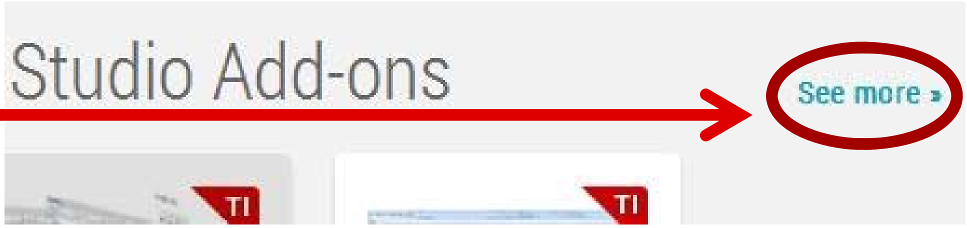


Lab Conventions

- Lab steps are in black and numbered for easier reference
    1. …
    2. …
- Descriptive text is written in blue

Install Software

- CCS version 6.1
- Install PRU C compiler
  1. Select *View→CCS App Center*
  2. To right of **Code Composer Studio Add-ons**, click on **See more >**
  3. Select **PRU Compiler**
     - Version 2.1.0 or later
  4. Click on **Install Software**
     - In the upper left corner. Scroll up if needed.



Workaround: Avoid Linux on BBW

BBW Only

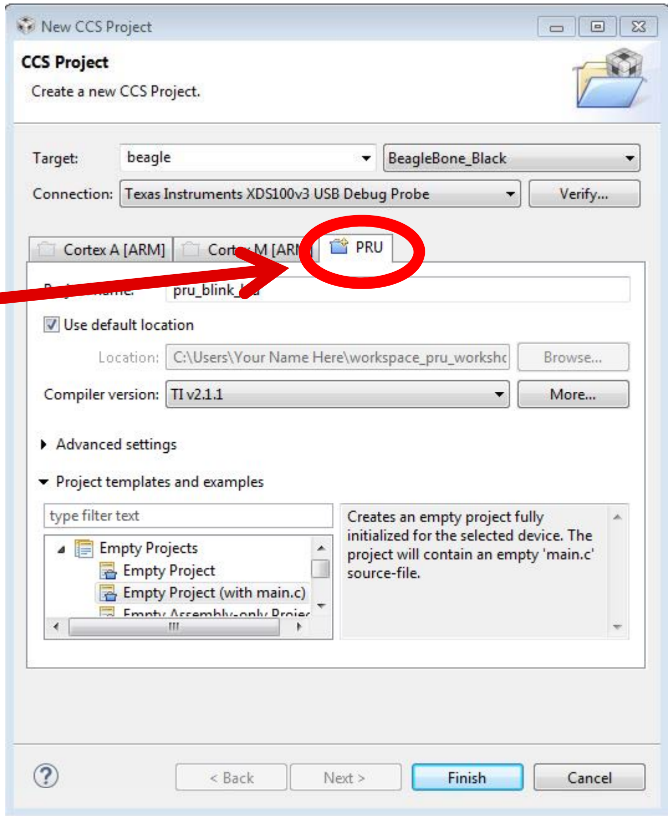
- Prevent Linux from running
- Eject micro SD card before power up
- OK for Linux to run on BBB

BBB: Create Blink LED Project

BBB
Only

From CCS Edit perspective

1. Go to *File→New→CCS Project*
2. Set **Target** to *BeagleBone_Black*
3. Set **Connection** to your debug probe
4. Select **PRU** tab
5. Enter project name
   - This workshop uses *pru_blink_led*
6. Leave **Project templates and examples** at default
   - Empty Project (with main.c)
7. Click **Finish**
   - CCS creates the project



Target: beagle | BeagleBone_Black

Connection: Texas Instruments XDS100v3 USB Debug Probe

Project name: pru_blink_led

☑ Use default location

Location: C:\Users\Your Name Here\workspace_pru_workshc

Compiler version: TI v2.1.1

Project templates and examples

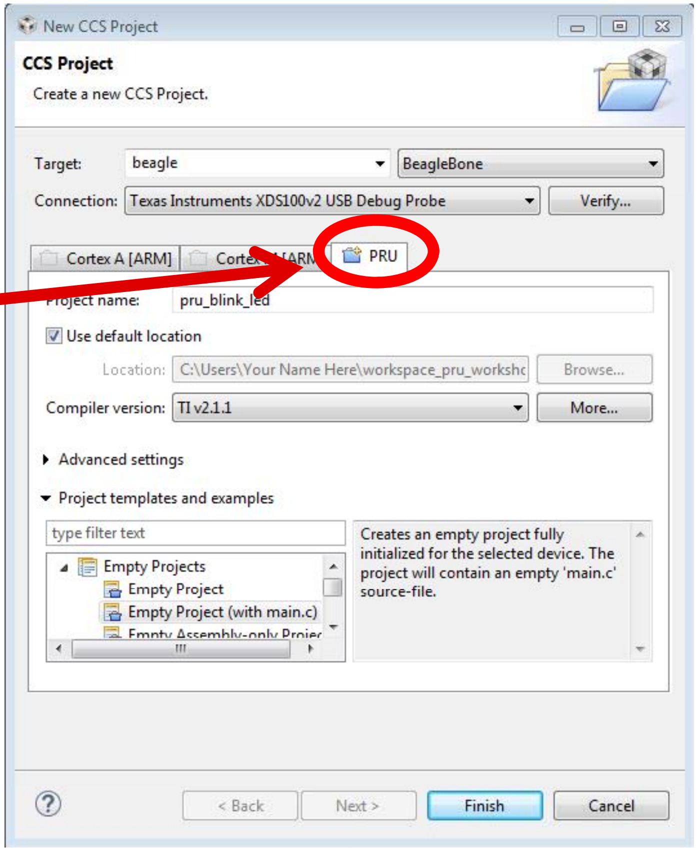
Creates an empty project fully initialized for the selected device. The project will contain an empty 'main.c' source-file.

BBW: Create Blink LED Project

From CCS Edit perspective

1. Go to *File→New→CCS Project*
2. Set **Target** to *BeagleBone*
3. Set **Connection** to XDS100v2 USB
4. Select **PRU** tab
5. Enter project name
   - This workshop uses *pru_blink_led*
6. Leave **Project templates and examples** at default
   - Empty Project (with main.c)
7. Click **Finish**
   - CCS creates the project

BBW Only



Create a new CCS Project.

Target: beagle — BeagleBone

Connection: Texas Instruments XDS100v2 USB Debug Probe

Project name: pru_blink_led

☑ Use default location

Location: C:\Users\Your Name Here\workspace_pru_workshc

Compiler version: TI v2.1.1

- Empty Projects
  - Empty Project
  - Empty Project (with main.c)
  - Empty Assembly-only Projec

Creates an empty project fully initialized for the selected device. The project will contain an empty 'main.c' source-file.

Replace All Code in main.c with These Lines

```
#include <stdint.h>    // for uint32_t
volatile register uint32_t __R30;

int main(void)
{
        uint32_t gpo;

        while (1)
        {
                gpo = __R30;
                gpo ^= 0xF;
                __R30 = gpo;
                // 10 to the 8th == half-second delay
                __delay_cycles(100000000);
        }
}
```

Build

1. Save changes to main.c with control-S
2. Click build icon 🔨

Correct any errors found.

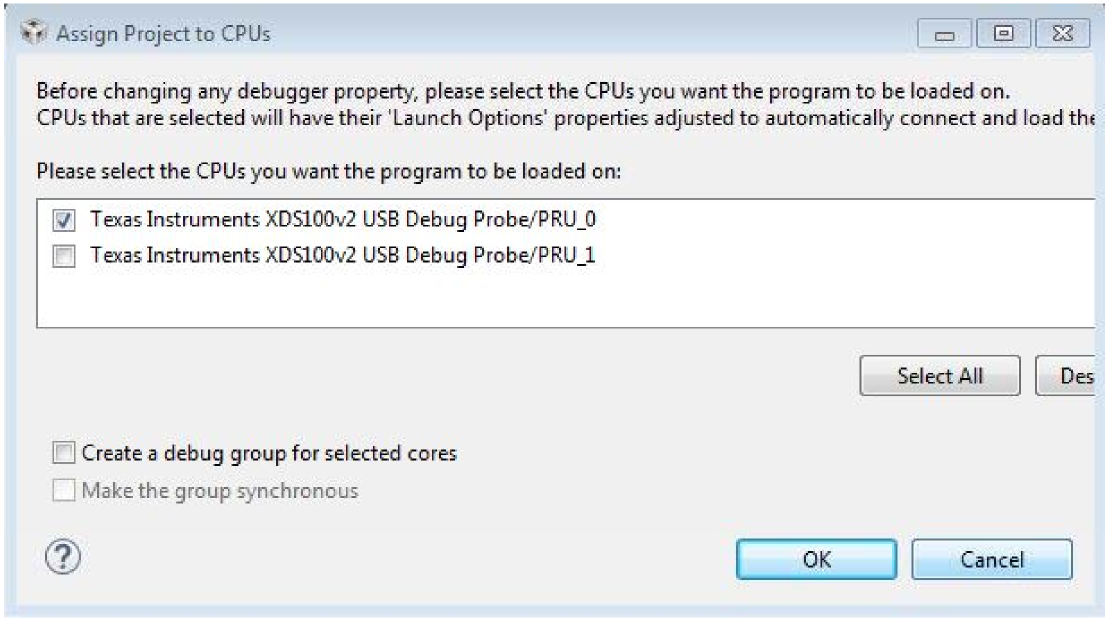
Debug Configuration Described

- It is tricky to access a deeply embedded CPU on a complicated SOC
- When you start a debug session, CCS automatically creates a *Debug Configuration*
  - Contains detailed info about how to start up the debugger part of CCS
- Accessing the PRU requires adding a special script to the Debug Configuration

Create Initial Debug Configuration

1. Right click the project name and choose **Properties**
2. In left pane highlight **Debug**
3. CCS asks which PRU core to load. Check *PRU_0*. Clear *PRU_1*.
4. Click **OK**
5. Click **OK** again

Adding the script is next.



Before changing any debugger property, please select the CPUs you want the program to be loaded on.
CPUs that are selected will have their 'Launch Options' properties adjusted to automatically connect and load the

Please select the CPUs you want the program to be loaded on:

☑ Texas Instruments XDS100v2 USB Debug Probe/PRU_0
□ Texas Instruments XDS100v2 USB Debug Probe/PRU_1

□ Create a debug group for selected cores
□ Make the group synchronous

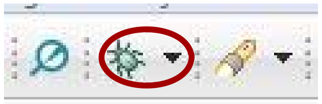
Add PRU Startup Script to Project

- More initialization needs to happen when debugger launches
  - Some on Debug Access Port (DAP), some on PRU
- Cannot perform with GEL
  - GEL script can only be associated with one CPU
- Can control from a Debug Server Scripting (DSS) script
  - Can perform operations on all CPU's

1. Download example DSS script
   - http://processors.wiki.ti.com/index.php/Debug_Configuration_Initialization_Scripts
   - File named bbb_pru_startup.js
2. If Project Explorer view is not present, select *View→Project Explorer*
3. Paste bbb_pru_startup.js into pru_blink_led project next to main.c
   - CCS Project Explorer view works just like Windows Explorer

Add Startup Script to Debug Configuration

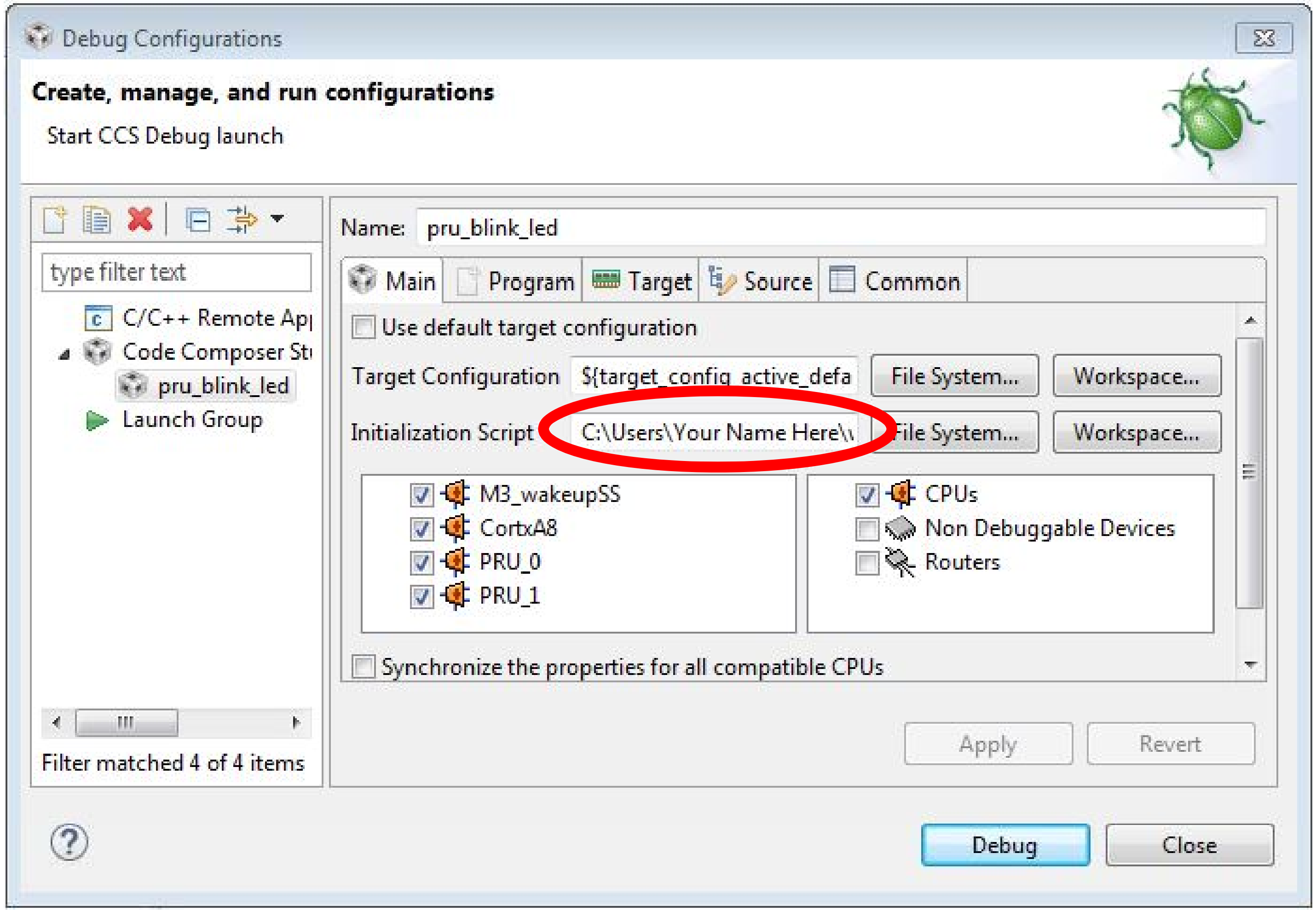
1. Click down arrow next to Debug icon and select **Debug Configurations**



2. In left pane, expand *Code Composer Studio* entry
3. Highlight the *pru_blink_led* configuration
4. Next to **Initialization Script** click **Workspace**
5. Open tree for *pru_blink_led* and select *bbb_pru_startup.js*
6. Click **OK**
7. Click **Apply**
8. Compare with screen shot in next slide
9. Click **Close**

This DSS script runs every time debugger is launched.

Debug Configuration



Debug Configurations

Create, manage, and run configurations

Start CCS Debug launch

Name: pru_blink_led

□ Use default target configuration

Target Configuration: ${target_config_active_defa

Initialization Script: C:\Users\Your Name Here\\

☑ M3_wakeupSS
☑ CortxA8
☑ PRU_0
☑ PRU_1

☑ CPUs
□ Non Debuggable Devices
□ Routers

□ Synchronize the properties for all compatible CPUs

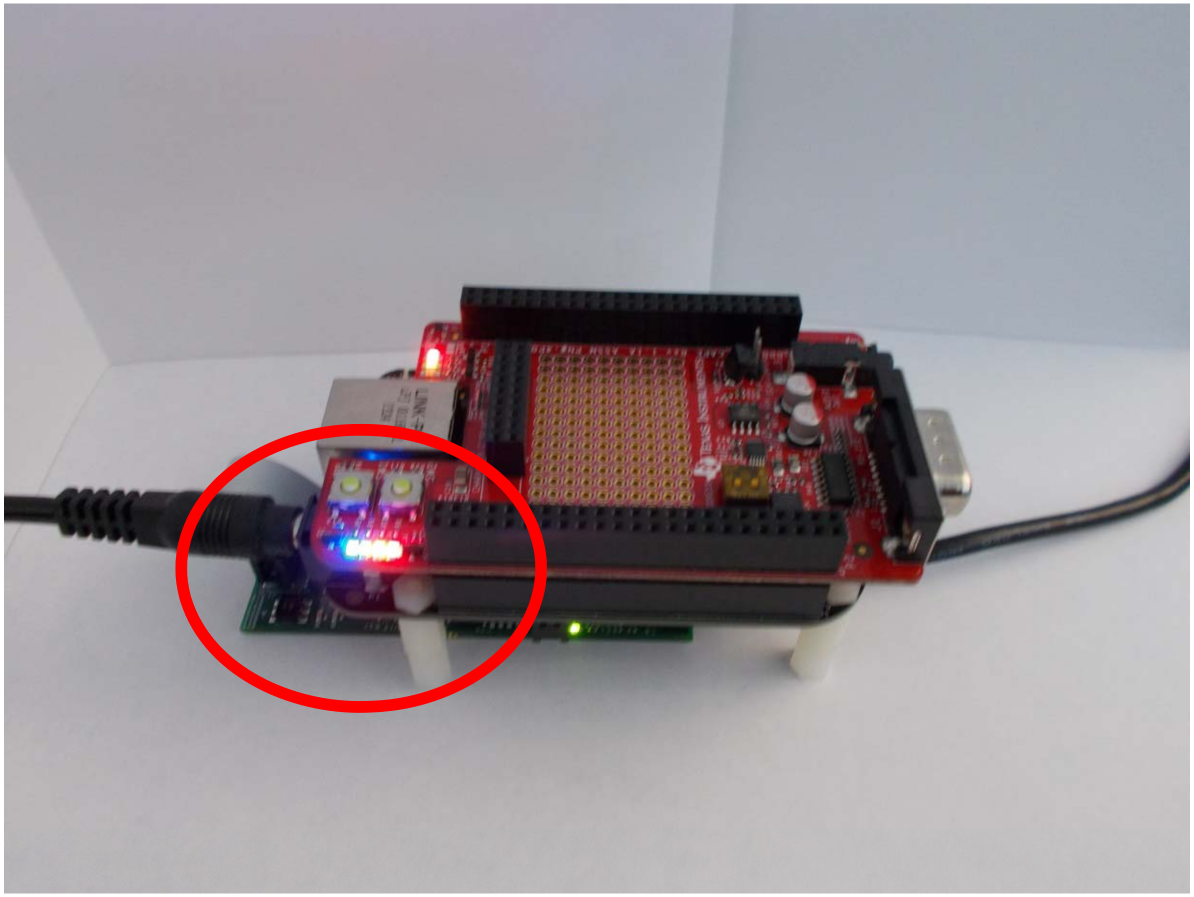
Execute the Code

1. Click Debug icon
2. Click Resume Execution icon
3. See the LED’s blink!
4. Click Suspend to halt execution

Ignore any error message about: invalid register index 8

This is a known issue which will be fixed.

Blinking LED



For More Information

- E2E Forum
  - http://e2e.ti.com
  - Community forum. Search, ask, answer questions.
- TI Processors Wiki
  - http://processors.wiki.ti.com
  - Technical articles on practical topics
- PRU Software Support Package
  - http://www.ti.com/tool/pru-swpkg
  - Framework and examples for developing PRU software