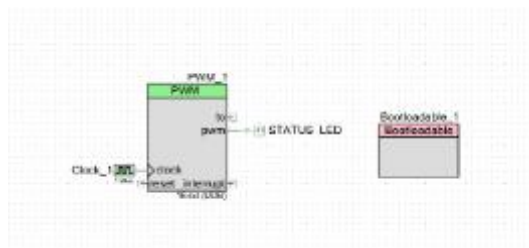


PSoC 4 Pioneer Kit Community Project#036 – What? I can use the PSoC 5LP too?

We all know that we use a PSoC 5LP device on the Pioneer kit as a programmer, debugger, and USB-I2C/UART bridge. What I wanted to showcase today is that this PSoC 5LP is not limited to the programming and bridging functionality. In the PSoC 5LP device we have included a USB Bootloader for field upgrades. Due to this feature you have the ability to bootload your own custom PSoC 5LP code into the device. By doing this you overwrite the existing PSoC 5LP functionality for your custom project. Don't worry about overwriting the PSoC 5LP code, you can easily restore the PSoC 5LP to its factory setting as a programmer and debugger. This bootloader allows you the opportunity to try out the PSoC 5LP device and enable your Pioneer kit as a dual ARM development platform.

The PSoC 5LP firmware is a single HEX file but has two components, the Bootloader Code and the Application Code. The application code is the programming, debugging, and bridging functionality we've see in our previous examples. The bootloader code governs the application code space, ensuring that the application code is correct and providing updates through serial interfaces (USB in this case). In this example we are creating new application code that will be sent to the PSoC 5LP device.

What I want to stress with this example is that while you purchased the Pioneer kit for the ARM PSoC 4 device but we have also given you access to our ARM Cortex M3 PSoC 5LP device. This makes the kit a dual chip kit for you to use and develop applications.



Forum Post Attachments:

At the bottom of this post we are including the following items:

- Example Project Zip File
- Zip File of Images
 - Project Schematic
 - Component Configurations

Components Used:

The user can download the example project at the bottom of this post. The project uses the following list of Creator Components:

- PWM
- Bootloadable
- CyPins
- CyClock

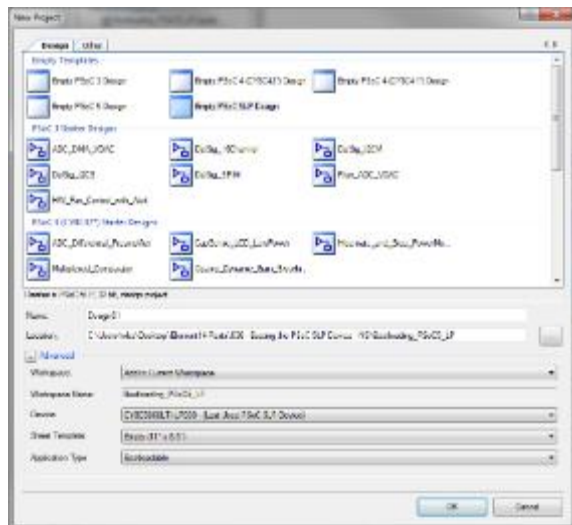
The components are configured by right clicking on the component in your Top Design schematic view and selecting **Configure**. Please enable the following selections in the Configuration windows for the listed components above.

Firmware Description:

The main.c firmware is included in the example project. Please review the commented sections for more details.

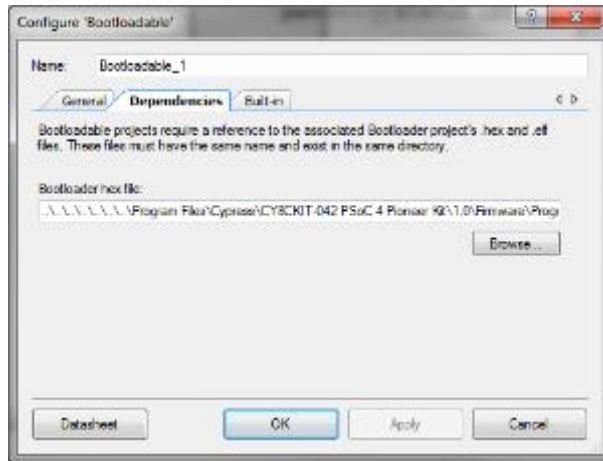
In this example we are showcasing the bootloadable characteristics of the PSoC 5LP. Using this bootloader the user can develop custom code targeting the PSoC 5LP device. The key point of this example is that you will lose the programming and debugging functionality of the device if you bootstrap custom application code. We will also show you how to recover the PSoC 5LP device to its factory settings.

To create our example we will use PSoC Creator. Open PSoC Creator and create a new project. Select an empty Design, name the project, select the CY8C5868LTI-LP039 device, and select Bootloadable under the application type.

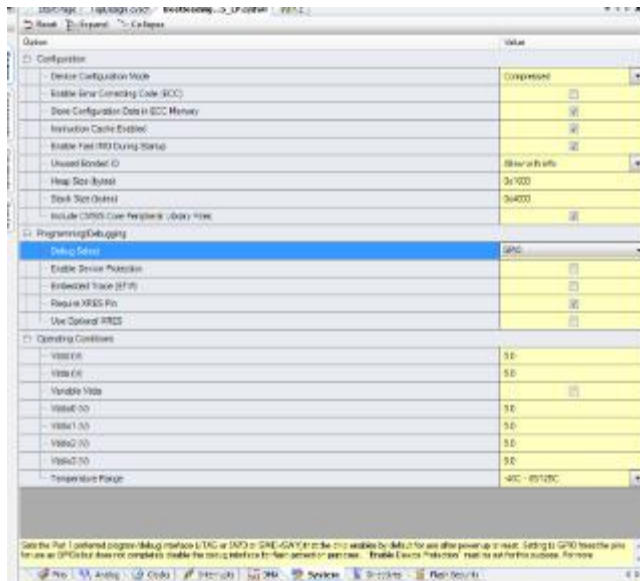


In our example we are going to create a simple blinking LED using a PWM. The example project below includes project design, components and routing. The only addition to our project is to add the 'Bootloadable' component. We must double click on this component to open the component options menu and select the dependencies tab. In this tab we must add the Bootloader hex file path. You can navigate to the kit install path:

```
...\\Program Files\\Cypress\\CY8CKIT-042 PSoC 4 Pioneer Kit\\[Kit  
Version]\\Firmware\\Programmer\\KitProg_Bootloader\\KitProg_Bootloader.hex
```



We need to ensure that the bootloadable file we are trying to create matches the same form and structure as the bootloader that is on the PSoC 5LP. This will allow us to boot in our application code over the USB interface. Next go to the DWR selections for 'System' and configure your settings:



Next we will need to add in a little code in our main.c file. This code will start the PWM. In this example I've implemented the PWM and LED control in hardware so there will be no need for any C code to control the PWM or LED. All we need is a single line of code to start the PWM. Build and compile this example.

```

10
11
12 // =====
13
14 #include <device.h>
15
16 void main()
17 {
18     /* Place your initialization/startup code here (e.g. MyInit_Start()) */
19     PWM_1_Start();
20     /* CyGlobalIntEnable: */ /* Document this line to enable global interrupts. */
21     for(;;)
22     {
23         /* Place your application code here. */
24     }
25 }
26
27 /* END OF FILE */
28

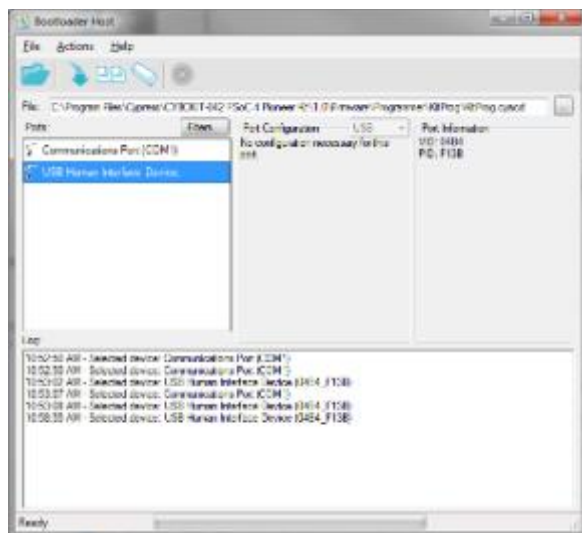
```

Once the project is built we will want to launch the Bootloader Host tool within PSoC Creator. This application will allow us to bootload our code into the PSoC 5LP device. Navigate to *Tools>Bootloader Host*. This will launch the tool. We are going to need to add a VID and PID ID to the filter. Please select the Filter button and then add in 0x4B4 for the VID and 0xF13B for the PID. This will allow us to identify the Bootloader on the PSoC 5LP.



Next we will want to connect the Pioneer kit to the PC. We will want to initiate the Bootloader sequence on the PSoC 5LP device so ensure that you are holding down the SW1 (RESET) button as you plug in the USB cable. Once the kit is plugged in release the SW1 switch. You will notice that the green status LED will be blinking once every second. This is how you know the PSoC 5LP is in Bootloader mode and waiting to be updated by the Host tool.

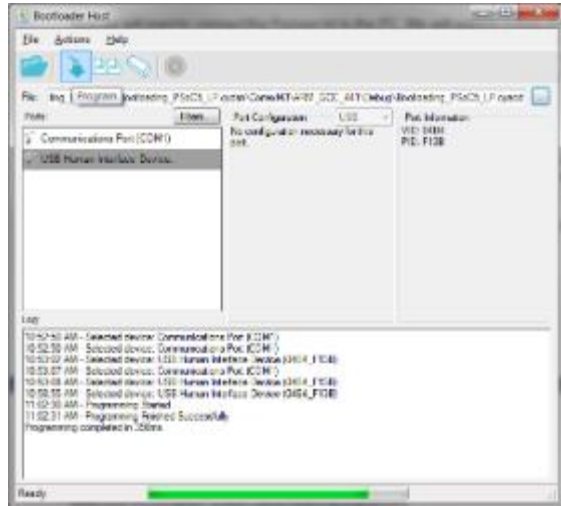
You will see a USB Human Interface Device in the port window. Select that port option, this is the PSoC 5LP bootloader.



Next we will want to select our new application code. Navigate up to the open folder button and select your new code file:

[Example]\Bootloading_PSoC5_LP\Bootloading_PSoC5_LP.cydsn\CortexM3\ARM_GCC_441\Debug\Bootloading_PSoC5_LP.cyacd

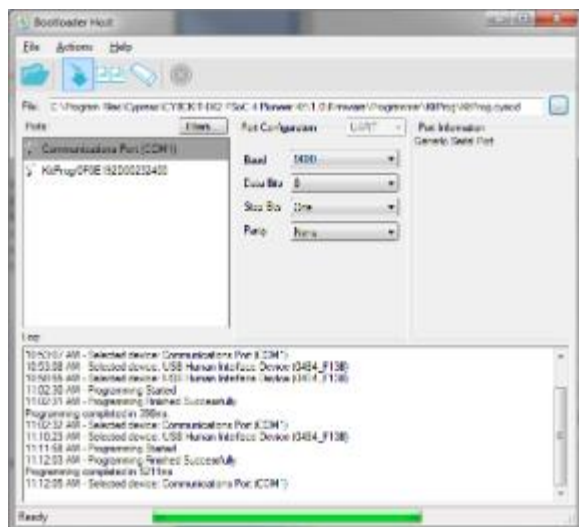
Next click the program button (downward facing arrow). The program will immediately bootload into the PSoC 5LP device over the USB connection. After the bootloading was complete you will see the Status LED blinking very fast. This is our new application on the PSoC 5LP device! I call it Rave Party LED!



The next steps will show you how to recover the PSoC 5LP device to its factory settings, to re-enable the programming, debugging, and bridging functionality. Again configure the Pioneer kit into bootloader mode by holding down the SW1 button and then replugging the kit into the PC using the USB cable. After releasing the SW1 switch the green status LED will blink at the 1 pulse per second rate, you are in bootloader mode. Next open the Bootloader Host tool, select the USB HID device from the port window. Next you will want to open the original application code source file. Click the open file button and navigate to the following file:

\\CY8CKIT-042 PSoC 4 Pioneer Kit\1.0\Firmware\Programmer\KitProg\KitProg.cyacd

Click the program button and see the kit updated to the factory firmware.



Hardware Connections:

There are no hardware connections other than connecting the Pioneer Board to the PC using the USB cable.

Test Your Project:

After bootloading your example into the PSoC 5LP device you will see that the Status LED will be blinking very fast. You can expand on the Pioneer kit by creating custom PSoC 5LP examples. Please take some time to review the Pioneer Kit user guide for more information on what PSoC 5LP features and pins are available to users. The user guide can be found here:

www.cypress.com/CY8CKit-042

I hope this example can help you in your design.

<http://www.element14.com/community/message/80725>