Week 2 Lab

CC3501

This week you'll read a light sensor and display the reading using pulse width modulation.

Task Description

The FRDM development board has a visible light sensor connected to one of the analog-to-digital converter (ADC) inputs. Your task is to read the light intensity using that sensor and display it using a light emitting diode (LED).

It is up to you to design how you want to present the light intensity. For example, you might use a single colour only and modulate the intensity, or you may prefer to switch from one colour to another as the intensity changes. Yet another option would be to change the speed at which the light appears to blink. A more sophisticated approach might be a smooth transition along a colour spectrum, for example from blue to green to yellow to red. Design an appropriate scheme and then implement it.

You must test the signal received from the light sensor under conditions of maximum indoor brightness and also minimal brightness (e.g. when the sensor is covered). The useful dynamic range of the sensor does not correspond to the full range of the ADC input and so you need to scale the measurement accordingly. You may find the debugger helps you inspect the raw ADC values under different conditions. Another option would be to connect a multimeter to the sensor output (which is exposed on the J10 header).

Isolating the light sensor—The light sensor is connected to an ADC pin but it is also exposed on the J10 header. Suppose you wanted to isolate the light sensor and use that J10 pin for something else. Studying the circuit schematic and looking carefully at the board itself, how would you do this?

Suggested Resources

You may find it convenient to use some of the following Processor Expert components[1]:

- ADC
- BitIO
- PWM

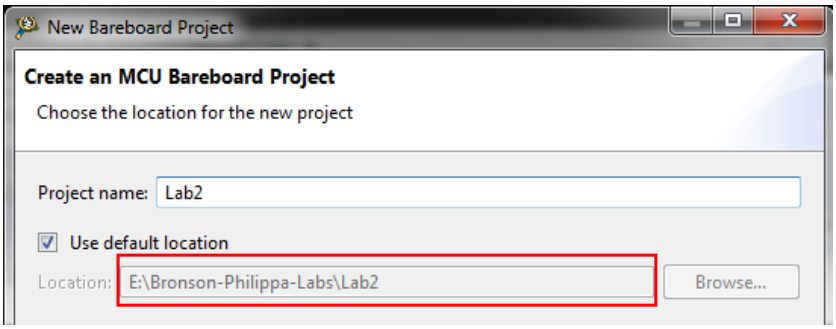
If you use any Processor Expert components, make sure you read the corresponding documentation carefully. Alternatively, if you are directly manipulating registers, you will need to find the appropriate sections of the reference manual.

[1] It is not mandatory that you use these components or even to use Processor Expert. You should develop your own professional judgement as to what tools are likely to be the most appropriate for any given task.

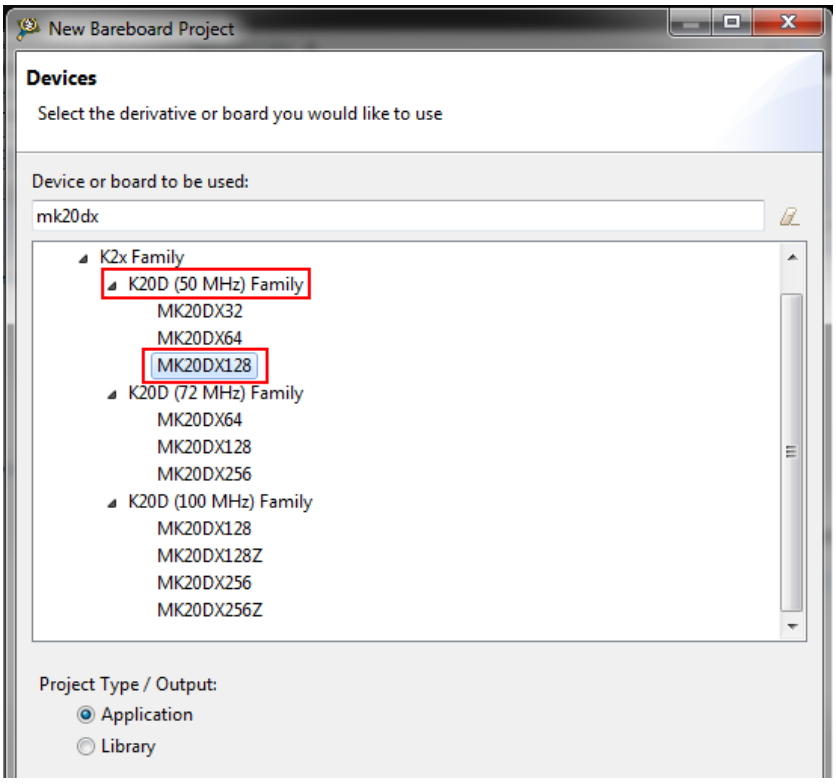
Creating the CodeWarrior Project

Follow these steps to create a new CodeWarrior project.

1. Start up CodeWarrior.
2. Select File -> New -> **Bareboard Project**.
3. Name your project "**Lab2**". **Ensure that the location is inside the Git repository that you cloned in the first week.**



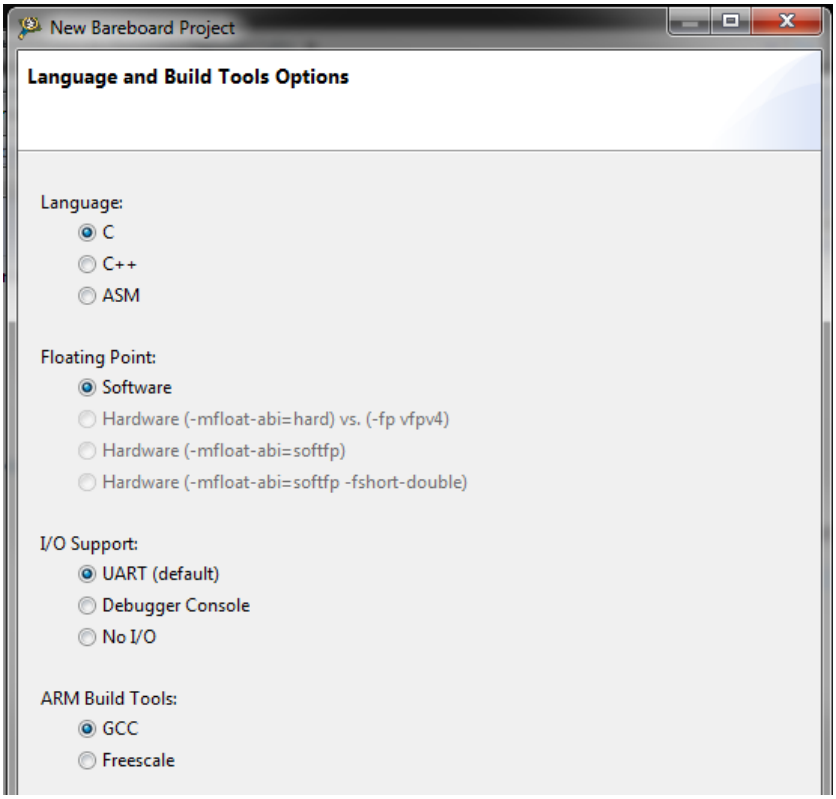
4. Under `K20D (50 MHz) Family`, select the **`MK20DX128`** processor type.



Workspace Directory— In the first week you should have cloned a Git repository into a USB drive, and created a CodeWarrior workspace in this directory. Open up the same workspace as before.

5. Select the **OpenSDA** debugging and ensure the other options are deselected.

6. Select the **C language**.



7. Ensure that the "Rapid Application Development" option is set to "**Processor Expert**".

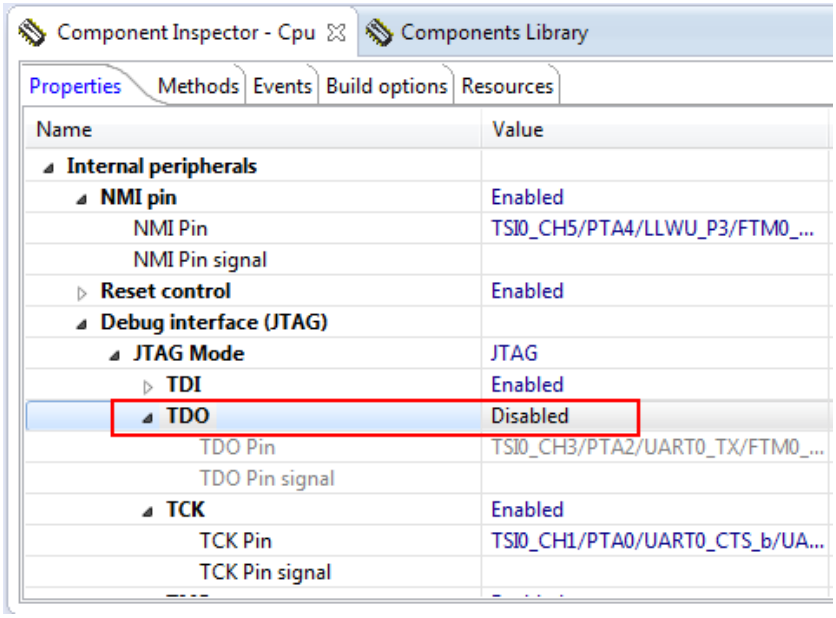
Clocking the Port Control module

Various components of the microprocessor are not clocked by default. If you use Processor Expert then the generated code will automatically provide a clock for all the necessary components. However, **if you are not using Processor Expert, then you must manually configure a clock for the Port Control module**. You can clock the Port Control module using the following C code, placed at the top of your main function:

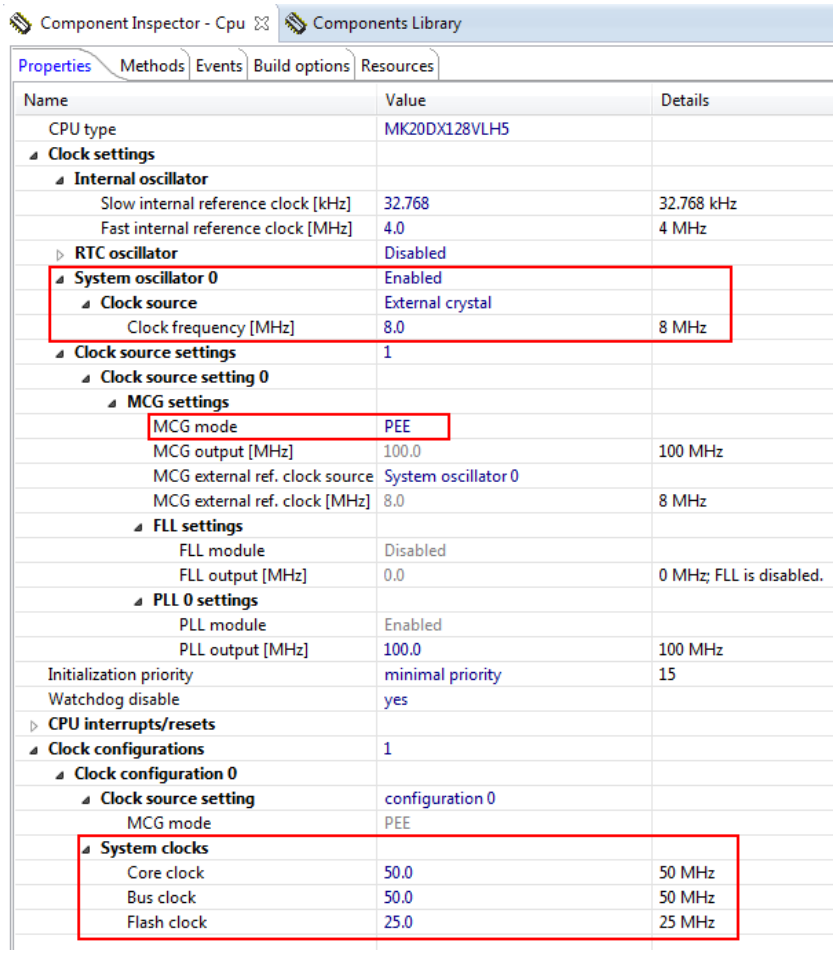
```
// Enable the clock for the port control module
SIM_SCGC5 |= 0x3e00;
```

Processor Expert Tips

- By default, PTA2 (the pin connected to the blue LED) is allocated to the processor's JTAG debug interface. To use it for GPIO or PWM, you need to first disable the relevant JTAG pin in the Component Inspector for the CPU:



- To run the CPU at its maximum clock speed (50 MHz), you need to configure the system oscillator to use the external crystal and then set the multi-clock generator to Phase Locked Loop External Engaged (PEE) mode:



Assessment

To complete this lab task, you must demonstrate to your prac tutor:

- A working board where the light intensity is visually displayed on the LED in an appropriate manner. A good test is to cover the light sensor with a piece of paper and watch the transition. Note that the transition from maximum brightness to minimum brightness must be clearly visible, i.e. you must scale the ADC result appropriately.
- Your GitHub webpage showing your Lab 2 code uploaded to your repository.