ELEC 3300

LAB 2: I/O INTERFACE, INTERRUPT FUNCTION OF ARM

A. OBJECTIVE:

1. To familiarize yourself with the MINI-V3 Development Board.
2. To understand the basic input/output of the microcontroller.
3. To understand the difference of polling and interrupt.
4. To refresh your knowledge of using the CRO.

B. PRE-LAB ASSIGNMENT:

1. Study the information about MINI-V3 Development Board from the course website.
2. Study the information about Fire Debugger from the course website.
3. Study the tutorial information related to LAB2.

C. LAB SETUP DETAILS

1. Download the LAB2.zip from the course website and unzip it.
2. Open Keil. Go Project → Open Project... Navigate to the project file for this lab, the project file should be under …/LAB2/Project/RVMDK/LAB2.uvprojx.
3. Connect the Fire Debugger according to the information about Fire debugger. Make sure that the Green LED of the Fire Debugger is ON.

D. EXPERIMENT

In this LAB, there are 6 tasks.

Task 1 to 3 are for initialization of the device.

Task 1 – You need to configure the RGB LED as the output.
Task 2 – You need to configure the 2 keys K1, K2, as the input.
Task 3 – You need to configure the interrupt function of K1.

Task 4 to 6 are implementation of the LAB functions.

Task 4 – Square wave generation and measurement
Task 5 – In the while(1) loop of main.c, comment the Task 4 code, and implement an if else statement to check if K2 is pressed, it will light up Blue LED, else Blue LED will be turned off.
Task 6 – Open the file stm32f10x_it.c, modify the function
`void EXTI0_IRQHandler(void)`
so that whenever K1 is pressed, the Green LED will toggle.

E. PROCEDURE

We will be using one RGB LED, button K1, K2 as the input, please refer to the schematics from the MINI-V3 Development Board.

Task 1 – You need to configure the RGB LED as the output in the main.c under the function
`void RGBLED_CONFIG(void)`

```
/* Task 1: Configure the RGB LED as output */
```

PIN	Connection	Type	Speed
PB.5	RGBLED-R	Output PP	50MHz
PB.0	RGBLED-G	Output PP	50MHz
PB.1	RGBLED-B	Output PP	50MHz

Task 2 – You need to configure the K1, K2 as the input in the main.c under the function
`void KEY_CONFIG(void)`

```
/* Task 2: Configure K1 and K2 as input */
```

PIN	Connection	Type	Trigger	Interrupt
PA.0	KEY1	Input (Floating)	Falling Edge	EXT0
PC.13	KEY2	Input (Floating)		

Task 3 – You need to configure the interrupt function of KEY1 under the function
`void EXTI_CONFIG(void)`

```
/* Task 3:  Configure the interrupt function of K1 */
```

Task 4 – Square wave generation and measurement.
In the while(1) loop, generate a waveform by the following idea.

```
while(1)
{
     Reset PB.5 to 0;
     Delayus(1); // This function is given
     Set PB.5 to 1;
     Delayus(1); // This function is given
}
```

Please note that you need to implement the Reset and Set by information from Tutorial 2.

1. Save your main.c. Right click the Project on the workspace and choose Options for Target 'LAB2'

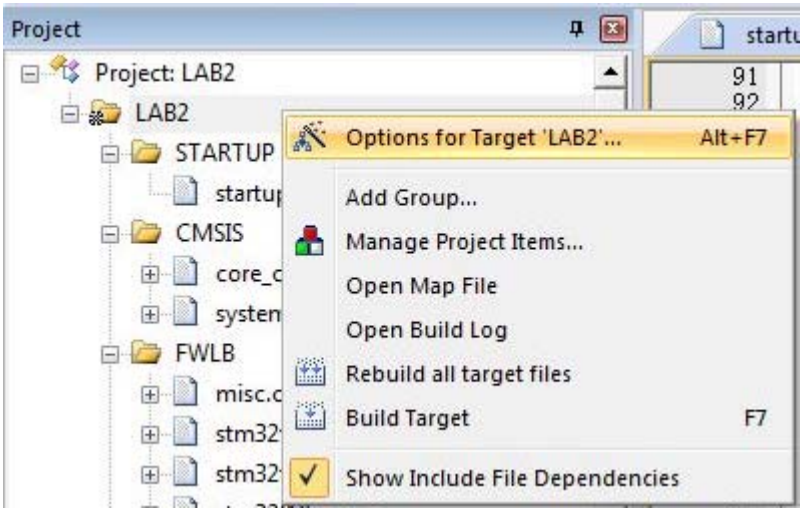

Figure 1: Project Option

2. In the Option, under C/C++ Tab, make sure your Optimization is Level 0. Click OK

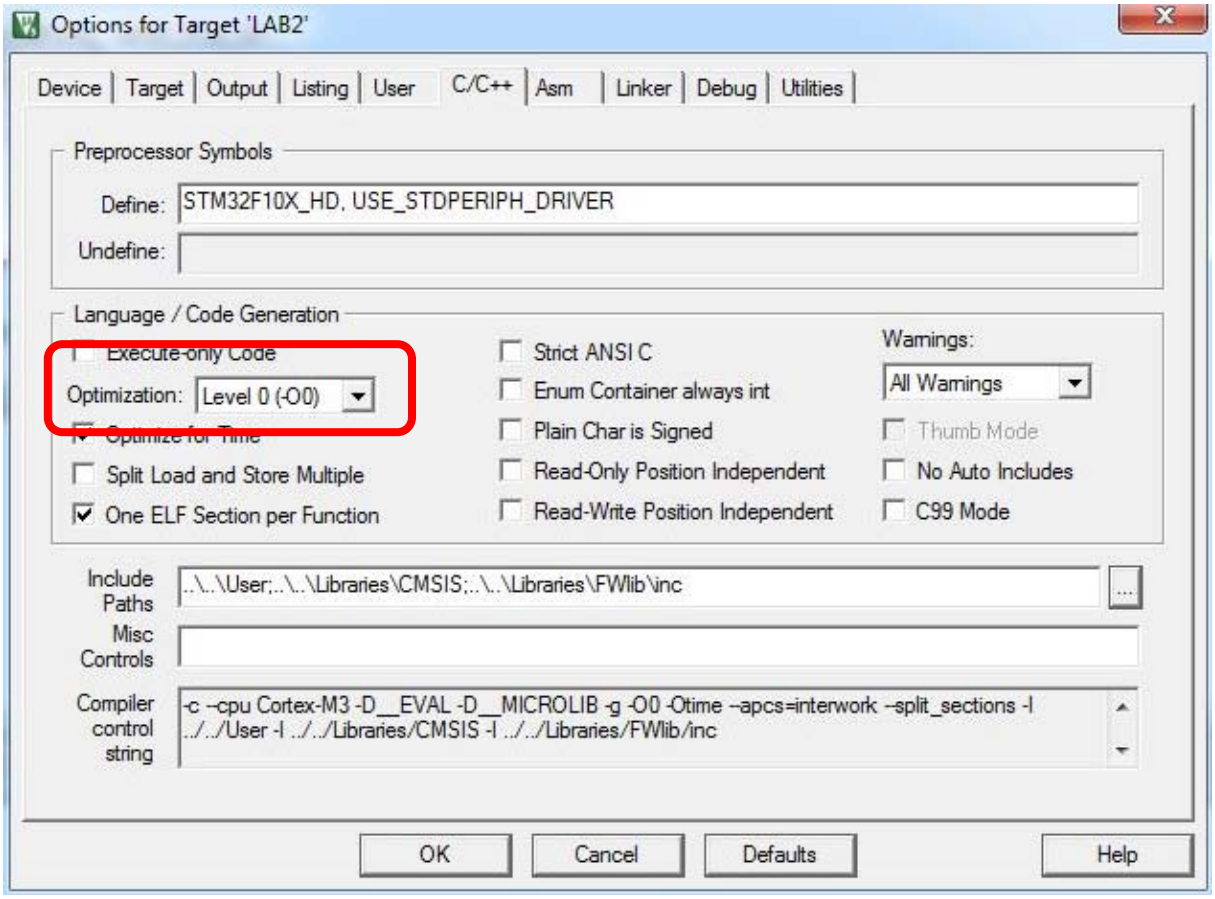

Figure 2: C/C++ Option – Optimization Level 0

3. Build your project by Choosing Project → Rebuild all Target File.

4. Locate the PB.5 on the board. Using the connection wires provided, connect the output to the CRO.

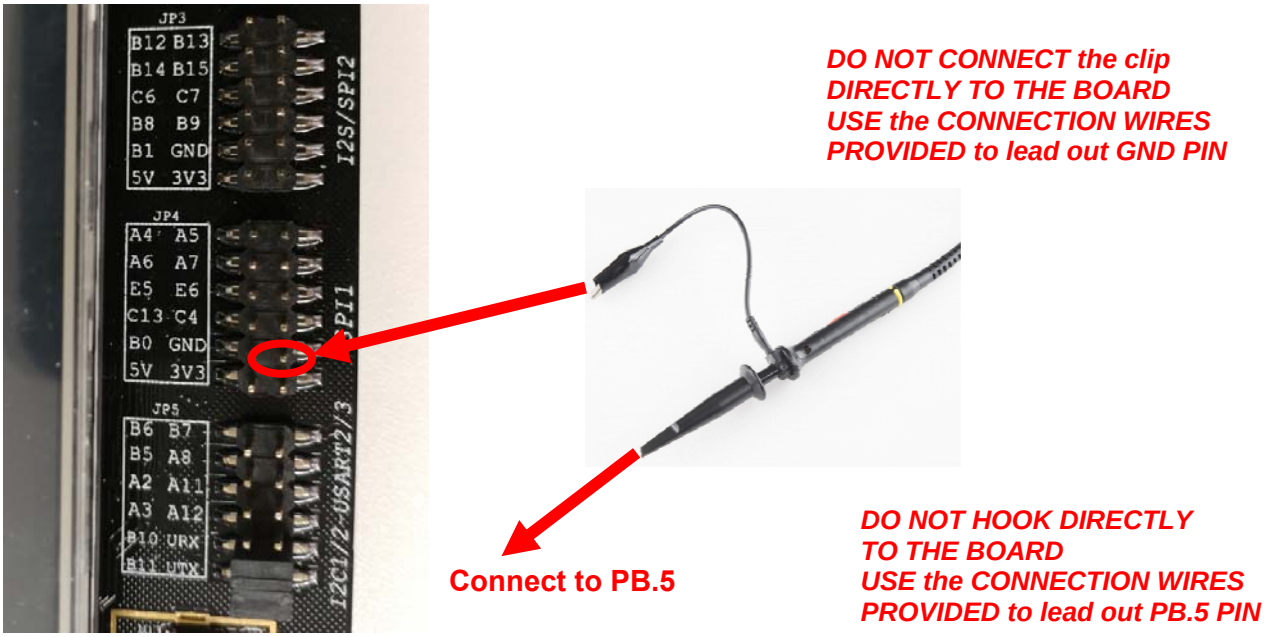

Figure 3: Output Pins on the right side of the Development Board

5. Start a Debug session by Debug → Start/Stop Debug Session

6. Run the Program by Debug → Run

7. Check the waveform and answer the following questions.

   What is the High time and Low time of your waveform?

   High time is the time between the wave start rising form the low level and start of falling from high level

   Low time is the time between the wave start falling from high level to start of rising form the low level

   High time: _____ Low time: _____

   How long is the delay generated by the function `Delayus(1)`? _____

   What C instruction that you Reset the pin level to 0? What is/are the corresponding assembly instruction(s)

   _____

   What C instruction that you Set the pin level to 1? What is/are the corresponding assembly instruction(s)

   _____

8. Go back to step 2, under C/C++ Tab, make sure your Optimization is Level 3. Click OK

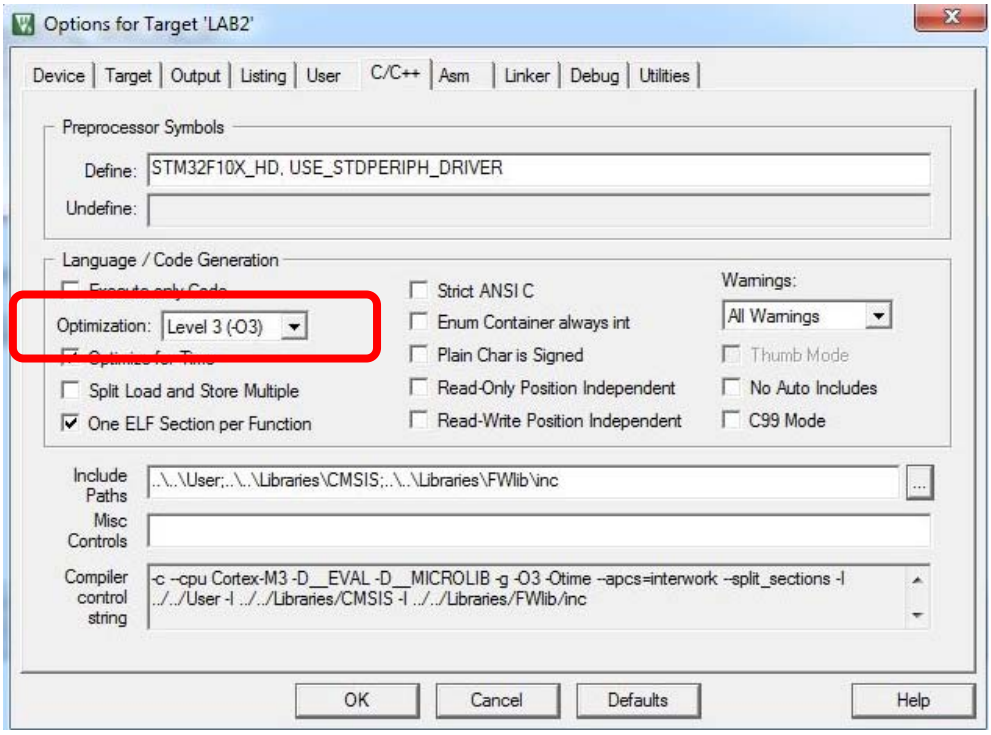

Figure 4: C/C++ Option – Optimization Level 3

9. Rebuild all your target files, download and run the project. Check the waveform and answer following questions.

   What is the High time and Low time of your waveform?

   High time: _____ Low time: _____

   How long is the delay generated by the function `Delayus(1)?` _____

   Is the Delayus(1) generate the same amount of Delay as in Optimization Level 0 ? _____

   Why you think it differs ? _____

   What C instruction that you Reset the pin level to 0? What is/are the corresponding assembly instruction(s)

   _____

   What C instruction that you Set the pin level to 1? What is/are the corresponding assembly instruction(s)

   _____

10. Compare the High time and Low time you recorded in either optimization level, is

    High time > Low time OR High time = Low time OR High time < Low time

    Can you explain why ? _____

Task 5 – Comment all your Task 4 code.

In the while (1) loop, implement an if else statement to check if K2 is pressed, it will light up the Blue LED, else the light will be turned off.

Rebuild, download the run your program. Show, your result to TA

Task 6 – Open the file stm32f10x_it.c, locate the function

```
void EXTI0_IRQHandler(void)
```

Modify the function so that whenever K1 is pressed, the Green LED will toggle.

Rebuild, download the run your program. Show, your result to TA that when you pressing K1, Green LED will toggle. At the same time, your Task 5 Code should still work.