Addressing debugging issue for lab-4

Problem: Disappearing the yellow arrow in the main program

You may or may not experience the issue when debugging lab-4 code. If you encountered the problem, use the suggestion work-around to make it easier to debug your code.

<u>**Description**</u>: After compiling the program (Build button or F7 key), you click on the 'Start/Stop Debug Session' menu or hit Control-F5 key to switch to 'Debug Mode'. Then you would see a yellow arrow pointing to the first executable line of code in the Reset_Handler procedure in the startup_LPC17xx.s file.

From this point on, every time you click on the 'Step' menu or press F11 key, one line of code is executed and the yellow arrow moves to the next line.

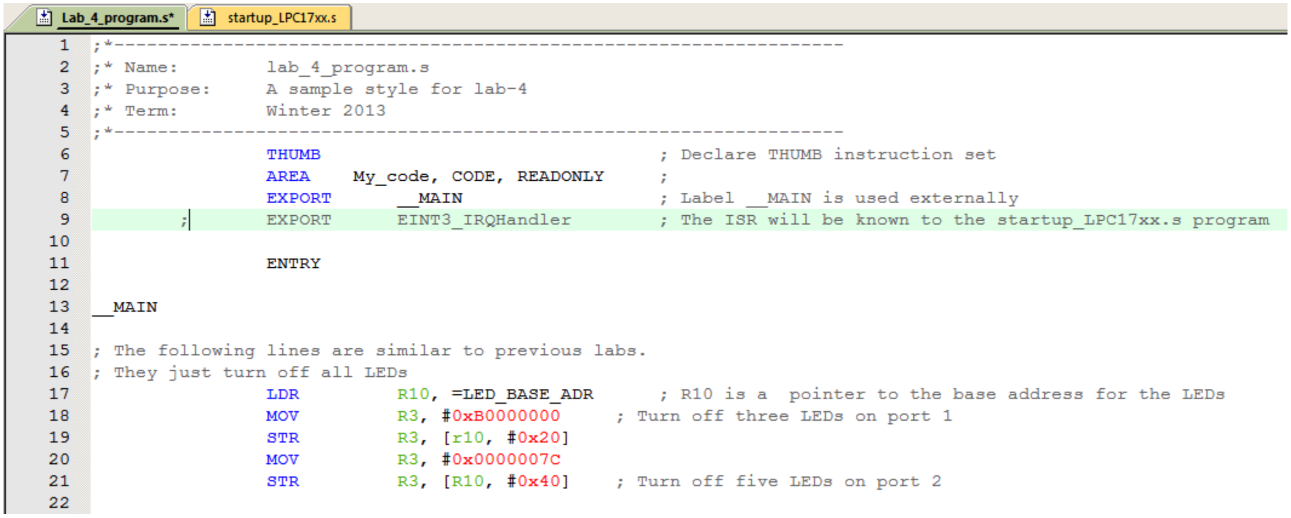
But the yellow arrow disappears when the program is shifted from the startup_LPC17xx.s file to the main program file. Not being able to see this yellow arrow makes it difficult to debug your code.

The following is a work-around that helps you avoid this issue.

Solution: Moving the EINT3_IRQHandler to the startup program

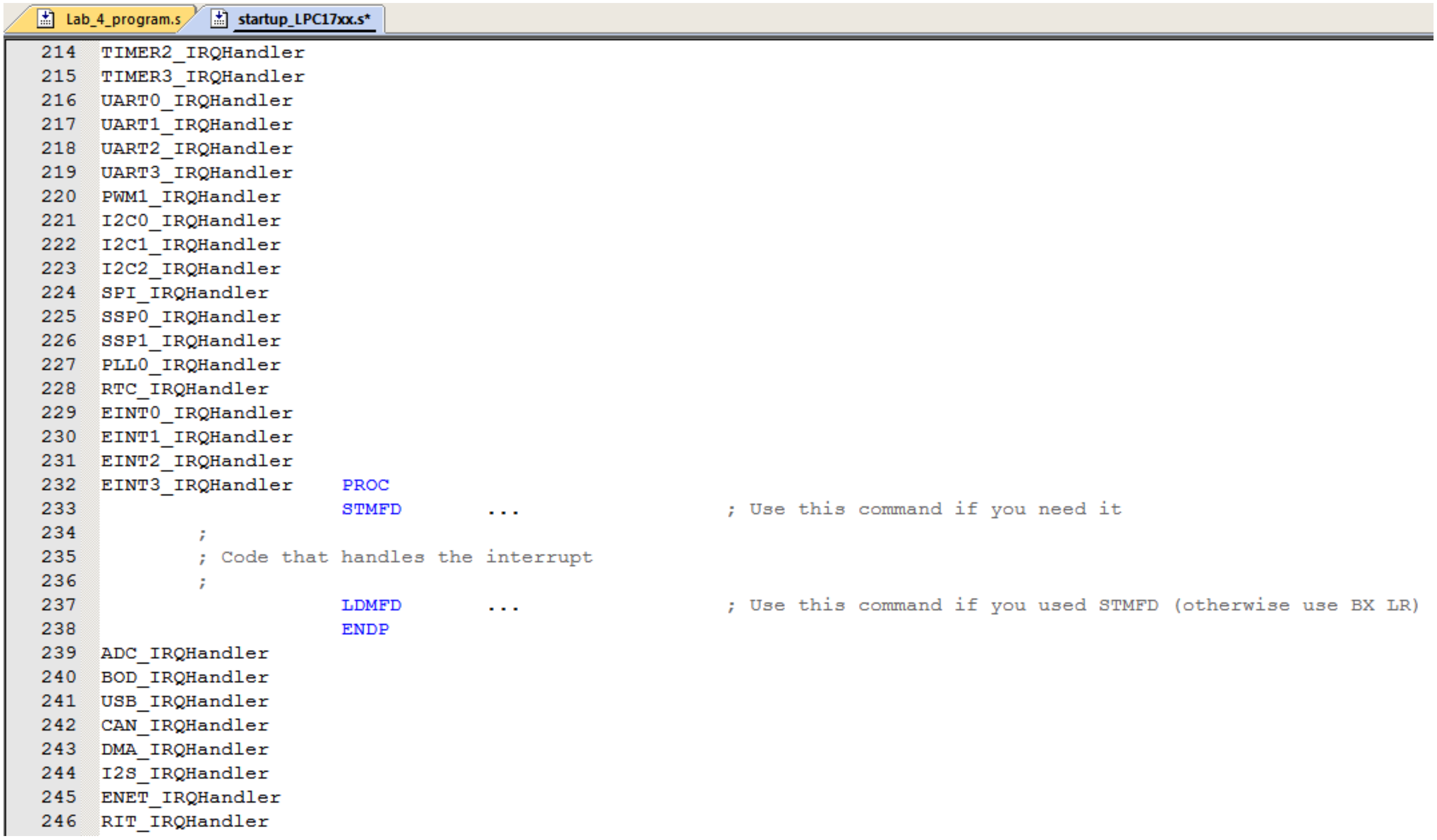
Moving the ISR of EINT3_IRQHandler from the main program to the startup_LPC17xx.s file should solve the problem. Here is what you will need to do:

1- Comment out line 9 of the given template code for lab-4.



```
 1  ;*-----------------------------------------------------------------
 2  ;* Name:          lab_4_program.s
 3  ;* Purpose:       A sample style for lab-4
 4  ;* Term:          Winter 2013
 5  ;*-----------------------------------------------------------------
 6                    THUMB                                ; Declare THUMB instruction set
 7                    AREA    My_code, CODE, READONLY      ;
 8                    EXPORT        __MAIN                 ; Label __MAIN is used externally
 9              ;     EXPORT        EINT3_IRQHandler       ; The ISR will be known to the startup_LPC17xx.s program
10
11                    ENTRY
12
13  __MAIN
14
15  ; The following lines are similar to previous labs.
16  ; They just turn off all LEDs
17                    LDR           R10, =LED_BASE_ADR     ; R10 is a  pointer to the base address for the LEDs
18                    MOV           R3, #0xB0000000     ; Turn off three LEDs on port 1
19                    STR           R3, [r10, #0x20]
20                    MOV           R3, #0x0000007C
21                    STR           R3, [R10, #0x40]    ; Turn off five LEDs on port 2
22
```

2- Move the procedure for the EINT3_IRQHandler to line 232 of the startup_LPC17xx.s file.



```
214  TIMER2_IRQHandler
215  TIMER3_IRQHandler
216  UART0_IRQHandler
217  UART1_IRQHandler
218  UART2_IRQHandler
219  UART3_IRQHandler
220  PWM1_IRQHandler
221  I2C0_IRQHandler
222  I2C1_IRQHandler
223  I2C2_IRQHandler
224  SPI_IRQHandler
225  SSP0_IRQHandler
226  SSP1_IRQHandler
227  PLL0_IRQHandler
228  RTC_IRQHandler
229  EINT0_IRQHandler
230  EINT1_IRQHandler
231  EINT2_IRQHandler
232  EINT3_IRQHandler    PROC
233                      STMFD       ...                 ; Use this command if you need it
234          ;
235          ; Code that handles the interrupt
236          ;
237                      LDMFD       ...                 ; Use this command if you used STMFD (otherwise use BX LR)
238                      ENDP
239  ADC_IRQHandler
240  BOD_IRQHandler
241  USB_IRQHandler
242  CAN_IRQHandler
243  DMA_IRQHandler
244  I2S_IRQHandler
245  ENET_IRQHandler
246  RIT_IRQHandler
```

By implementing the above two steps you should be able to debug your code similar to previous labs with no problem. Report any problem to Lab Instructor (rkeshava@uwaterloo.ca).