Embedded Systems (CSCE 4114)

Lab 7

Zack Fravel

12/4/16

zpfravel@uark.edu

? What happens if you press Key0 or Key1?

Key0: Decreases the left shift speed of the LED by 1Hz

Key1: Increases the left shift speed of the LED by 1Hz

? What do variables a, b, c, d, e, f mean?

a : unsigned binary value to be shown passed to LED PIO

b: timer frequency counter (convert to milliseconds)

c : timing frequency dividing factor [1, 16] Hz

d : edge detector for KEY0 and KEY1

e : timer register value (50,000)

f : timer frequency (50,000,000) (in clock cycles).

? What is the purpose of functions f1, f2, f3 and f4? Provide a line by line explanation.

f1:
- Sets button register for Key0
- Sets button register for Key1
- sets the register to call f3 whenever a button interrupt is received

f2:
- sets ‘e’e to the first 16 bits of the timer register value and shifts 16 left
- adds the last 16 bits to ‘e’
- prints the value of ‘e’ to the console
- prints the value of ‘f’ to the console

f3:
- creates integer ‘d’
- sets ‘d’ to the value of the button register that sent the interrupt request
- checks if ‘d’ == 1 (KEY0)
    - set ‘c’ = the result of a conditional checking if ‘c’ is greater than 1, if so then it decrements ‘c’ if not it remains 1.
- checks if ‘d’ == 2 (KEY1)
    - set ‘c’ = the result of a conditional checking of ‘c’ is less than 16, if so then it decrements ‘c’ if not it remains 16.

f4: - Writes to the timer register 0 all 0's
- increment 'b'
- checks if b > 1000/c (timer frequency in milliseconds)
    - if so, sets 'a' = 'a' << 1
    - checks if ('a' == 256)
        - if so, sets 'a' = 1
    - Writes the value of 'a' to the LED register
    - set 'b' = 0

? What is the execution flow of this program?

- Program declares two main functions f1 and f2
- declares variables
    a = 128
    b = 0
    c = 8
- main method
    - calls f1()
    - calls f2()
    - runs while(1) (infinite loop)
    - return0

The program once it goes through f1 and f2 once gets kicked into a self repeating loop, once f3 and f4 are called by f1 and f2 respectively. The program uses the software CPU's timer and its register values to create a timer that works in millisecond resolution. The timer counts 50,000,000 clock cycles 50,000 times (0.001 second). Every time this count happens, an interrupt request kicks off f4. Whenever f4 runs, it sets the timer back to 0 and increments 'b' and checks whether or not 'b' is greater than the value the user desires for the shifting of the LED's. f4 runs every millisecond and shifts the LED's based on the rate set by variable 'c.' Once the LED's finally shift it sets 'b' back to 0. f1 sets the buttons and sets up f3 to be kicked off every time a button interrupt request is received. Whenever f3 receives an interrupt, it checks which button register it received it from and sets the variable 'c' accordingly.