NAME: _____ STUDENT #: _____

EECE 259: Introduction to Microcomputers **Lecture Quiz** **Mar 30, 2011**

An **Alarm System** does several things at the same time. First, it blinks a **warning light** on LEDG0 (500ms on, 500ms off) if the system is armed but there is no alarm. Second, every 10ms it checks to make sure that no alarm has been triggered. An alarm is triggered if the system is armed and the value of **any sensor** on SW[9:4] changes from its previous state; these keys must be polled every 10ms. Third, an alarm is always triggered immediately via interrupt if the **panic buttons** on KEY3 or KEY2 are pressed. Fourth, a triggered alarm causes the alarm lights on LEDR to blink (250ms on, 250ms off). Fifth, once triggered, an alarm is cleared by setting the alarm password on SW[3..0] and then pressing the clear button on KEY1. Sixth, as the alarm is cleared, if the password remains on SW[3:0], then the system is NOT armed so no alarm can be triggered and LEDG0 remains off; if the value of SW[3:0] ever changes becomes armed; if set back to the password it only unarms by KEY1.

```
#include "259macros.h"

/* global variables */
enum Modes { IDLE, ARMED, ALARM };
enum Modes mode = IDLE;
int sensors;
int password = 0xB;
int count = 0;
int red   = 0;
int green = 0;
int oldpw = 0xB;

int main(...)
{
    initInterrupts();
    enableCounterIRQ( 10*ONE_MS ,cntrISR);
    enableKeyIRQ(  8 | 4 | 2  , keyISR );

    /* write your code below */
    while(1) {

        int sw = *pSWITCH;
        int pw = sw & 0xf;

        if( mode==IDLE && pw!=oldpw ) {
                disableInterrupts();
                mode = ARMED;
                sensors = sw & 0x7f0;
                enableInterrupts();
        }

        oldpw = sw & 0xf;


    }

 }

/* ISR should be called every 10ms  */
void cntrISR()
{
    /* remember: no waiting in here */
    *pCOUNTER_STATUS = 1; // clear irq

    count++;
    if( mode==ARMED && (count%50==0) ) {
            green = !green;
            count = 0;
    }

    int sw = *pSWITCH & 0x7f0;
    if( mode==ARMED && sw!=sensors ) {
            mode  = ALARM;
            green = 0;
            count = 0;
    }

    if( mode==ALARM && (count%25==0) ) {
            red   = !red;
            count = 0;
    }
    *pLEDR = red ? 0x3ff : 0; // ALL LEDs
    *pLEDG = green;
}

void keyISR()
{
    /* remember: no waiting in here */
    int key = *pKEY_EDGECAPTURE;
    *pKEY_EDGECAPTURE = 0;
    if( key & (8|4) ) {
            mode = ALARM;
            red  = 1;
    } else if( key & 2 ) {
            key = *pSWITCH;
            if( (key&0xf)==password) {
                    mode = IDLE;
                    red  = 0; green = 0;
            }
    }
}
```