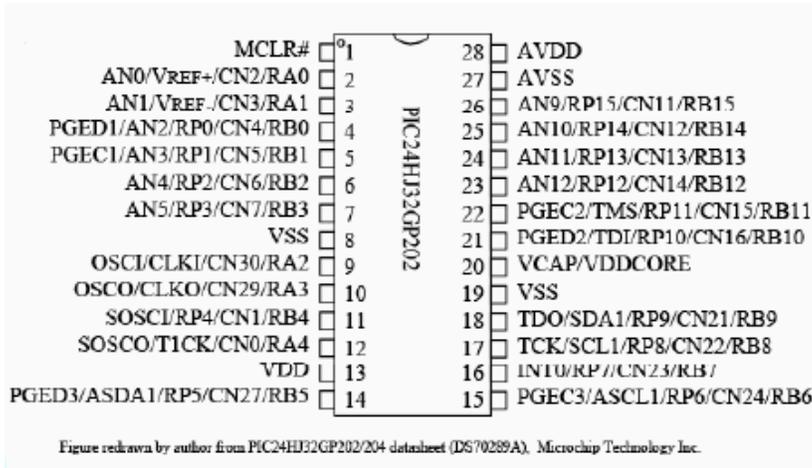


UNCA CSCI 255 Practice Exam 3

Ch 8 --- PIC24 hardware systems

1. Describe at least 7 of the functionalities associated with the physical pins of the PIC24, as shown below.



2. The following 3 registers are part of the parallel IO system: TRIS, LAT, and PORT (the individual bits of the PORT register are named `_RBx` and `_RAx`, e.g., `_RB14`). What purpose does each of these registers serve in the parallel IO system?
3. Describe the functionality of each Macro or inline function listed below:
 - a. `ENABLE_RB15_PULLUP()`
 - b. `ENABLE_RB13_OPENDRAIN()`
 - c. `CONFIG_RB8_AS_DIG_OD_OUTPUT()`
4. Given a C function that tests the functionality of a XOR gate as an example, write a C function that tests the functionality of an AND gate. The function must exhaustively apply all input combinations to the external gate. If the gate produces the wrong output for any input combination, the function terminates and returns *false*. If the gate produces the correct output for all input combinations, then the function should return *true*. Use a 1 us delay between applying the gate inputs and reading the gate outputs. The function prototype is:

```
unit8 testGate();
```

Questions 5 through 8 assume that the PIC24 is configured as shown in the Figure 1 and running the program listed below that figure.

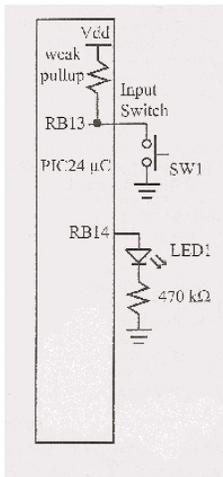


Figure 1

```
#define CONFIG_LED1() CONFIG_RB14_AS_DIG_OUTPUT()
#define LED1 _LATB14

inline void CONFIG_SW1() {
    CONFIG_RB13_AS_DIG_INPUT();
    ENABLE_RB13_PULLUP();
}

*** #define SW1 _RB13
#define SW1_PRESSED() (SW1==0)
#define SW1_RELEASED() (SW1==1)

int main (void) {
    configBasic(HELLO_MSG);
*** CONFIG_SW1();
    CONFIG_LED1();
    DELAY_US(1);
    LED1 = 0;
    while (1) {
*** while (SW1_RELEASED())
        doHeartbeat();
*** DELAY_MS(15);
        while (SW1_PRESSED())
            doHeartbeat();
        DELAY_MS(15);
*** LED1 = !LED1;
    }
}
```

5. Explain the functionality of each stated line in the program above.
6. Does the program above implement polled IO?
7. What will happen each time the pushbutton switch attached to pin RB13 is depressed and released?

8. What purpose is served by the weak pull-up resistor on pin RB13? Why is it necessary to have this configuration when running the program provided above.

Ch 9 --- PIC24 interrupts

1. What is an *Interrupt Service Routine* and how is it executed?
2. What is a *trap* in the context of this course?
3. The following 3 registers are part of the Interrupt Control system: IFSx (e.g., _CNIF for the Change Notification interrupt), IPCx (e.g., _CNIP for the Change Notification interrupt), and IECx (e.g., _CNIE for the Change Notification interrupt). What purpose does each of these registers serve in the Interrupt Control system?

Questions 4 through 11 have not been discussed in pervious lectures; they will be covered for the first time in this review.

4. Assume that RB13 of the PIC24 has a pushbutton switch connected to it as shown in problems 5-8 above. Assume that the change notification interrupt for RB13 is enabled (which is CN13). For the following code snippet, how many times does the _CNInterrupt Interrupt Service Routine (ISR) execute if the pushbutton is pressed and released one time? Assume no switch bounce and explain your answer.

```
void _ISR_CNInterrupt(void) {
    DISABLE_RB13_CN_INTERRUPT();
    _CNIF = 0;
}

int main (void) {
    CONFIG_RB13_AS_DIG_INPUT();
    ENABLE_RB13_CN_INTERRUPT();
    _CNIF = 0;
    _CNIP = 2;
    _CNIE = 1;
    while (1);
}
```

9. Explain the functionality of each stated line in the function below.

```
#define ISR_PERIOD    15    // in ms
void configTimer3(void) {
    T2CONbits.T32 = 0;
    T3CON = T3_OFF | T3_IDLE_CON | T3_GATE_OFF
           | T3_SOURCE_INT
           | T3_PS_1_64 ; //results in T3CON= 0x0020
    *** PR3 = msToU16Ticks(ISR_PERIOD, getTimerPrescale(T3CONbits))-1;
    *** TMR3 = 0; //CLEAR TIMER, I.E., REGISTER HOLDING THE TICK COUNT
    *** _T3IF = 0;
    *** _T3IP = 1;
    *** _T3IE = 1;
    *** T3CONbits.TON = 1;
}
```

10. Explain the functionality of each stated line in the program below.

```
#define CONFIG_LED() CONFIG_RB14_AS_DIG_OUTPUT()
#define LED _LATB14

#define SW1 _RB13
inline void CONFIG_SW1() {
    CONFIG_RB13_AS_DIG_INPUT();
    ENABLE_RB13_PULLUP();
}

*** volatile uint8 SWvalue = 0;
*** volatile uint8 oldSWvalue = 0;

//Interrupt Service Routine for Timer3
*** void _ISRFAST _T3Interrupt (void) {
***     _T3IF = 0;
***     SWvalue = SW1;
***     if(SWvalue == 1 && oldSWvalue == 0) {
***         LED = !LED;
***     }
***     oldSWvalue = SWvalue;
*** }

#define ISR_PERIOD 15 // in ms
void configTimer3(void) {
    T2CONbits.T32 = 0; // 32-bit mode off
    T3CON = T3_OFF | T3_IDLE_CON | T3_GATE_OFF
           | T3_SOURCE_INT
           | T3_PS_1_64 ; //results in T3CON= 0x0020
    PR3=msToU16Ticks(ISR_PERIOD, getTimerPrescale(T3CONbits)) - 1;
    TMR3 = 0;
    _T3IF = 0;
    _T3IP = 1;
    _T3IE = 1;
    T3CONbits.TON = 1;
}

int main (void) {
    configBasic(HELLO_MSG);
    CONFIG_SW1();
    CONFIG_LED();
    configTimer3();
    while (1) {
        doHeartbeat();
    }
}
```

11. Assuming that the PIC24 is configured as shown for questions 5-8 of Ch 8 and running the program listed in question 10 above, what will happened each time the pushbutton switch attached to pin RB13 is depressed and released?