

Microcontroller pin configuration is shown engineering essay

[Engineering](#)



**ASSIGN
BUSTER**

Abstract

This report is about pic16f84a microcontroller programming that how we write a program and how to burn it. In this report there is little introduction about controller and most of code are described. This report is on subject digital electronics designs. There are different tasks which we have to complete by writing their coding and check the program by burn it into microcontroller. We use pic16f84a microcontroller for programming in lab session of digital electronics fundamentals and MPLAB IDE v8.33 and HI-TECH compiler software for our lab. Pic16f84a is 18 pins microcontroller pin configuration is shown. Vss is used to supply 5v and Vdd is used as ground. There are four interrupts External RB0/INT pin TMR0 timer overflow PORTB <7: 4> interrupt-on-change Data EEPROM write complete. RB6, RB7, Vss, Vdd and MCLR used to burn the program.

I/O ports:

RA0 to RA4 pins of porta and RB0 to RB7 pins of portb are used as input output in microcontroller. In lab1 our first task is to use microcontroller RB0 to RB4 pins as output which are connected to leds and RA0 to RA4 pins as input through programming in MPLAB. When we write full program then compile it to check the errors download it into microcontroller.

First task code:

```
#include // header file#define LED1 PORTBbits. RB0 // define portb pins as LED# so no need to write address of complete pin in main program#define LED2 PORTBbits. RB1#define LED3 PORTBbits. RB2#define LED4 PORTBbits.
```

```
RB3#define LED5 PORTBbits. RB4void delay (void); // delay function but no  
need in this taskvoid main(void) //main function
```

```
{  
TRISA= 0xFF; //define porta as inputTRISB= 0x00; //define portb as  
outputLED1= 0; //all portb initial value is 0LED2= 0; LED3= 0; LED4= 0;  
LED5= 0; PORTA= PORTB; PORTA= 0x00; //input is initially zero  
delay();  
while(1) //loop when true loop will run infinitely  
  
{  
if(PORTAbits. RA0== 1) //to turn on led1 give PORTA value ' 1' means turn  
on switch which is connected to RA0 pin in if statement means when RA0 is  
when go into if body and do that else do what is in else body  
  
{  
LED1= 1;  
  
}  
else LED1= 0; if(PORTAbits. RA1== 1) //to turn on led2{LED2= 1;}else  
LED2= 0; if(PORTAbits. RA2== 1) //to turn on led3{LED3= 1;}else LED3= 0;  
if(PORTAbits. RA3== 1) //led4 turn on{LED4= 1;}else LED4= 0; if(PORTAbits.  
RA4== 1) //led5 turn on{LED5= 1;}else LED5= 0;  
  
}  
  
}  
void delay (void) //delay function
```

```

{
int n; //define integerfor (n= 0; n <50; n++); //for loop which run unless n=
50 then go back to program where delay is call

}

```

Task two:

In task two we have to transfer data from Port B to Port A.

Task two codes:

```

#include #define LED1 PORTAbits. RA0 //in this task we define Porta pins as
LED# because we want to show output on porta#define LED2 PORTAbits.
RA1#define LED3 PORTAbits. RA2#define LED4 PORTAbits. RA3#define
LED5 PORTAbits. RA4void delay (void); void main(void)

```

```

{
TRISA= 0x00; //initialize porta as outputTRISB= 0xFF; //initialize portb as
inputLED1= 0; //all leds initial value is set to 0LED2= 0; LED3= 0; LED4= 0;
LED5= 0; while(1)

```

```

{
if(PORTBbits. RB0== 1) //portb use as switch write statement in if
function{ LED1= 1; //turn on led1 on RA0

```

```

}
else LED1= 0; if(PORTBbits. RB1== 1)

```

```

{
LED2= 1;

```

```

}
else LED2= 0; if(PORTBbits. RB2== 1){ LED3= 1;

}
else LED3= 0; if(PORTBbits. RB3== 1){LED4= 1;}else LED4= 0;
if(PORTBbits. RB4== 1)

{
LED5= 1;

}
else LED5= 0;}

}

```

Task three:

In task three we have to turn on light on portb with delay of 1 second for this task we modify previous program and burn it into microcontroller.

Task three codes:

```

#include #define LED1 PORTBbits. RB0 //define portb pin as LED# #define
LED2 PORTBbits. RB1 #define LED3 PORTBbits. RB2 #define LED4 PORTBbits.
RB3 #define LED5 PORTBbits. RB4 void delay (int d); void main(void)

{
TRISB= 0x00; //initialize as output LED1= 0; LED2= 0; LED3= 0; LED4= 0;
LED5= 0; while(1) //while loop run endlessly

```

```

{
LED1= 1; //turn on led1delay(66); //delay of 1 secondLED1= 0; //turn off led
1LED2= 1; //turn on led2delay(66); //delay of 1 secondLED2= 0; //turn off
led2LED3= 1; delay(66); LED3= 0; LED4= 1; delay(66); LED4= 0; LED5= 1;
delay(66); LED5= 0;

}

}

int d; void delay (int d) //delay function int d is integer value which we write
in delay fuction

{
int b; int n;

{
for (b= 0; b <1000; b++);

}}

```

Task four:

Write a program on traffic light signalSequence is shown in picture

Four task codes:

```

#include #define GREEN1 PORTBbits. RB7 //define portb pin 7 as
green1#define AMBER1 PORTBbits. RB6 //define portb pin 6 as
amber1#define RED1 PORTBbits. RB5 //define portb pin 5 as red1#define
GREEN2 PORTBbits. RB4 //define portb pin 4 as green2#define AMBER2
PORTBbits. RB3 //define portb pin 3 as amber2#define RED2 PORTBbits.

```

```
RB2 //define portb pin 2 as red2
void delay (int d); //delay function
void
main(void) //main function

{
TRISB= 0x00; //initialize trisb as output value written in
hexadecimal
GREEN1= 0; //initial valuse of all leds are set to 0
AMBER1= 0;
RED1= 0; GREEN2= 0; AMBER2= 0; RED2= 0; while(1) //while loop

{
PORTB= 0b00110000; //green2 and red1 are on
delay(262); //delay of 4
seconds
PORTB= 0b00101000; //red1 amber2 are on
delay(131); //delay of 2
seconds
PORTB= 0b00100100; //red1 and red 2 are on
delay(132); PORTB=
0b00110100; //red1, red2 and amber1 are on
delay(132); PORTB=
0b10000100; //red2 and green1 are on
delay(262); PORTB=
0b01000100; //red2 and amber1 are on
delay(132); PORTB=
0b00100100; //red2 and red1 are on
delay(132); PORTB= 0b00101100; //red2,
red1 and amber2 are on
delay(132);

}

}

int d; void delay (int d) //delay function

{

int b; int n;

{

for (b= 0; b <1000; b++); //for loop
```

```
}}
```

Task five:

We modify previous program by using switches RA0 and RA1 on the development board so that the lights remain in the r1 g2 state until the RA0 switch is pressed and remains in the g1 r2 state until the RA1 switch is pressed.

Task five codes:

```
#include #define GREEN1 PORTBbits. RB7#define AMBER1 PORTBbits.
RB6#define RED1 PORTBbits. RB5#define GREEN2 PORTBbits. RB4#define
AMBER2 PORTBbits. RB3#define RED2 PORTBbits. RB2#define S1 PORTBbits.
RA1 //define#define S2 PORTBbits. RA0void delay (int d); void main(void)

{
TRISB= 0b00000000; //portb as outputTRISA= 0x03; //initialize porta two pin
RA1 and RA0 as inputPORTB= 0x00; GREEN1= 0; //initial value is 0AMBER1=
0; RED1= 0; GREEN2= 0; AMBER2= 0; RED2= 0; while(1)

{
PORTB= 0b00110000; //green1 and red2 are ondelay(262);
//delaywhile(S1== 0); //when sensor 1 is zero remain in this statePORTB=
0b00101000;//red1 amber2 are ondelay(131); PORTB= 0b00100100;//red1
and red 2 are ondelay(132); PORTB= 0b00110100;//red1, red2 and amber1
are ondelay(132); PORTB= 0b10000100;//red2 and green1 are ondelay(262);
while(S2== 0); //remain in this state until s2 is equal to 1PORTB=
0b01000100;//red2 and amber1 are ondelay(132); PORTB=
```



```
0b00100100;//red2 and red1 are ondelay(132); PORTB= 0b00101100;//red2,
red1 and amber2 are ondelay(132);
```

```
}
```

```
}
```

```
int d; void delay (int d)
```

```
{
```

```
int b; int n;
```

```
{
```

```
for (b= 0; b <1000; b++);
```

```
}
```

```
}
```

Lab 2:

In second lab our first task is to run motor clockwise and anticlockwise and stop. We use RA4 and RA3 to determine it's these states. When the value on these pins are same then motor will be stop and run in clock wise direction when value on RA3 is one and RA4 is zero when these values are reverse then motor will run in anti-clockwise direction. We use IR sensor to count its rotations. It is connected to RA1 pin of microcontroller. The microcontroller can monitor the rotation of the motor by detecting the rising or falling edges on this input pin. The steps which we follow in this exercise are Start motorDetect three falling edges on pin RA1Stop motorWait for about half a secondStart motor in reverse directionDetect three falling edges on pin RA1Stop motorWait for about half a second

<https://assignbuster.com/microcontroller-pin-configuration-is-shown-engineering-essay/>

Task 1 code:

```

#include #define RA3 PORTAbits. RA3#define RA4 PORTAbits. RA4void
delay(void); //delay functionvoid edge(void); //edge function to count rising
or falling edgevoid main( void)

{
// initialise Port A and Port BTRISA = 0b00011; ; /* set tris B register */TRISB
= 0x00; ; /* tris B all o/p */PORTA = 0b00000 ; /* set Port A register to 0
*/INTCON= 0X00; //initial value of intcon register is set to zero
int b; for(b= 0;
b <10; b++) //for loop which will run 10 times

{
PORTB= b+1; //increment in port b value till to run 10 times for loopRA3= 0 ;
//run anticlock wiseRA4= 1; edge(); RA3= 0; //stopRA4= 0; delay(); //delay
about secondRA3= 1; //run clockwise directionRA4= 0; edge(); RA3= 0;
//stopRA4= 0; delay();

}
while(1); //after running for loop 10 time it will stuck here} // end of mainvoid
edge(void) //edge function which will count three edges

{
int a; for(a= 0; a <3; a++)

{
while(PORTAbits. RA1== 1); while(PORTAbits. RA1== 0);

```

```
}}
```

```
void delay(void) //delay function using inner interrupt of microcontroller to
generate delay of one second
```

```
{
```

```
TMR0= 0x0c; //timer0 initial valueOPTION_REG= 0x87; //set option register
value and use prescalarwhile(INTCONbits. T0IF== 0); //program will remain
in this while loop until flag is set highINTCONbits. T0IF= 0; //set flag to zero
```

```
}
```

Task two of lab2:

```
#include #define RA3 PORTAbits. RA3 //define porta pins#define RA4
PORTAbits. RA4#define RB0 PORTBbits. RB0 //define port b pins#define
RB1PORTBbits. RB1#define RB2 PORTBbits. RB2#define RB3 PORTBbits.
RB3#define RB4 PORTBbits. RB4#define RB5 PORTBbits. RB5#define RB6
PORTBbits. RB6#define RB7 PORTBbits. RB7int a; a= 0; void edge(void);
//edge functionvoid main( void) //delay function
```

```
{
```

```
TRISA = 0b00011; //initialize port a two pins as input and other as
outputTRISB = 0x00; //initialize portb as outputTMR0= 0; //set timer0 value
to zeroINTCONbits. T0IE= 1; //enable intocon register timer0 overflow
interruptINTCONbits. GIE= 1; //enable global interruptOPTION_REG=
0b11001000; //set RBPU Interrupt Edge Select bit to rising edge and not use
prescalarPORTB= 0; //set initial value of portb to zerowhile(1) //loop will run
endlessly
```

```

{
RA3= 0 ; //run fan in one directionRA4= 1; edge(); //edge function which will
run to three edgesTMR0= 254; //interrupt of timer 0 overflowRA3= 0; //stop
motorRA4= 0; RA3= 1; run motor in other directionRA4= 0; edge(); //edge
functionTMR0= 254; RA3= 0; RA4= 0; while(1);}} // end of mainvoid
edge(void) //edge function{int a; for(a= 0; a <3; a++)

{
while(PORTAbits. RA1== 1); while(PORTAbits. RA1== 0);

}

}

void interrupt ISR() //interrupt function which is use to turn on off leds in
sequence

{
INTCONbits. GIE= 0; //set global interrupt to zerowhile(a <7) //this loop will
run 7 times

{
if( INTCONbits. T0IF == 1&&a== 0) //if statement when timer0 flag is high
and int a is zero then go into if body

{
RB0=! RB0; //toggle led on RB0

}

if( INTCONbits. T0IF == 1&&a== 1) //second if to toggle other led

```

```
{
RB0=! RB0; RB1=! RB1;}if( INTCONbits. T0IF == 1&&a== 2)

{
RB1=! RB1; RB2=! RB2;

}
if( INTCONbits. T0IF == 1&&a== 3)

{
RB2=! RB2; RB3=! RB3;}if( INTCONbits. T0IF == 1&&a== 4)

{
RB3=! RB3; RB4=! RB4;

}
if( INTCONbits. T0IF == 1&&a== 5)

{
RB4=! RB4RB5=! RB5;

}
if( INTCONbits. T0IF == 1&&a== 6)

{
RB5=! RB5; RB6=! RB6;

}
if( INTCONbits. T0IF == 1&&a== 7)
```

```

{
RB6=! RB6; RB7=! RB7;

}
break;

}
if(a== 7) //when loop complete 7 cycle then go to if body and set portb to
zero

{
PORTB= 0;

}
INTCONbits. T0IF = 0; // reset INTO interrupt flagINTCONbits. GIE= 1; //enable
global interrupt

}

```

Task three of lab2 multiple interrupts:

In this lab we use multiple interrupts to use blinking of led's in one direction then to reverse direction

```

#include
#define RA3 PORTAbits. RA3 //define porta pins
#define RA4 PORTAbits. RA4//define RB0 PORTBbits. RB0 //define portb pins
#define RB1PORTBbits. RB1//define RB2 PORTBbits. RB2//define RB3 PORTBbits. RB3//define RB4 PORTBbits. RB4//define RB5 PORTBbits. RB5//define RB6 PORTBbits. RB6//define RB7 PORTBbits. RB7void
delay(void); //delay functionint dir= 0; //int directionvoid edge(void); //edge
funcitonvoid main( void)

```

```

{
TRISA = 0b00011; //initialize porta pins
TRISB = 0b00000001; //initialize portb
pins as output except RB0 which is used as external interrupt
PORTA = 0b000000 ; /* set Port A register to 0 */
TMR0= 0; //initial value of timer0 is set to 0
INTCON= 0b11111000; //set intcon register enable these interrupt
Global Interrupt Enable bit EE Write Complete Interrupt Enable bit
TMR0 Overflow Interrupt Enable bit RB0/INT External Interrupt Enable bit
and RB Port Change Interrupt Enable bit
OPTION_REG= 0b11001000; PORTB= 0; //set portb to zero
while(1) //loop will run endlessly

```

```

{
RA3= 0 ;//MOVE FORWARD
RA4= 1; edge();//THREE RISING EDGE
TMR0= 254;
RA3= 0;//STOP
RA4= 0; RA3= 1;//BACKWARD
RA4= 0; edge(); TMR0= 254;
//timer0 interrupt
RA3= 0; //stop
RA4= 0; while(1);} // end of main
void edge(void) //edge function

```

```

{
int a; for(a= 0; a <3; a++)

```

```

{
while(PORTAbits. RA1== 1); while(PORTAbits. RA1== 0);

```

```

}}
void delay (void) //delay function

```

```

{
int n; for (n= 0; n < 5; n++)

```

```
}  
void interrupt ISR() //interrupt  
  
{  
  INTCONbits.GIE= 0;//disable all other interrupts  
  if(dir== 0) //if statement  
  when int dir is 0 then turn on leds on one direction  
  
  {  
    RB1= 1; //turn on led connected to RB1  
    delay(); //delay function  
    RB1= 0;  
    RB2= 1; //turn on other led  
    delay(); RB2= 0; RB3= 1; delay(); RB3= 0; RB4=  
    1; delay(); RB4= 0; RB5= 1; delay(); RB5= 0; RB6= 1; delay(); RB6= 0; RB7=  
    1; delay();  
  
  }  
  else //it is used to turn on led's in reverse direction  
  
  {  
    RB7= 1; delay(); RB7= 0; RB6= 1; delay(); RB6= 0; RB5= 1; delay(); RB5= 0;  
    RB4= 1; delay(); RB4= 0; RB3= 1; delay(); RB3= 0; RB2= 1; delay(); RB2= 0;  
    RB1= 1; delay();  
  
  }  
  dir=! dir; //reverse the direction of leds  
  INTCON= 0b11111000;}
```