

```

1  #include
    <main.h>

                                   //CCS
2
3  #define HS
    PIN_B3

                                   //CCS
4
5  unsigned int Read_Memory(unsigned int);

    //Function for Reading Memeory
6  unsigned int counter=0;

                                   //Function for Counter
7  void Write_Memory(unsigned int,unsigned int);

    //Function for Writing to Memory
8
9  #INT_TIMER2

                                   //CCS
10
11 void TIMER2_isr(void)

                                   //Function for Timer
12 {
13     counter++;

                                   //Interupt
14     if(counter==40)

                                   //When Counter
15     {
16         output_toggle(pin_c0);

    //Switch Ouput of C0
17     counter=0;

                                   //Set Counter to Zero
18     }
19 }
20
21 void main()

                                   //Main Program Below
22 {
23
24     setup_timer_2(T2_DIV_BY_16,249,10);

    //Timer Parameters
25     enable_interrupts(INT_TIMER2);

                                   //Enables Timer 2 Interupts
26     enable_interrupts(GLOBAL);

                                   //Enable ALL Interupts
27
28     unsigned int Data,Old_Data;

                                   //Function for Data and Old Data
29     port_b_pullups(0x08);

                                   //Set Handshaking on Port B with Pull Up
30
31     while(TRUE)

                                   //Repeat the Code Below
32     {
33         delay_ms(10);

                                   //Delay for PIC

```

```

34         Data=Read_Memory(0x00);

        //Read from Memory
35
36         if(Data!=Old_Data)

            //When Data is Different
37         {
38             output_d(Data);

            //Write onto Output D
39             Write_Memory(0x05,Data);

        //Write to Memory Location
40     }
41
42     if(Data & 0x20)

        //Function for
Hazards
        //If
43     {
44         if(!input(PIN_A0))

        //Check Status
45     {
46         output_high(PIN_D1);                                //Set Ouput
On
47         if(Data!=Old_Data)                                    //When
the Data is Different
48         Write_Memory(0x09,Data);                                //Write Memmory
49     }
50     else

        //If Not
51     {
52         output_low(PIN_D1);                                //Set
Output Off
53         if(Data!=Old_Data)                                    //When
the Data is Diffrent
54         Write_Memory(0x09,0xff);                                //Write
Memmory
55     }
56     }
57     Old_Data=Data;

        //End of Function
59
60     }
61 }
62
63 unsigned int Read_Memory(unsigned int address)                                //Function
for Reading Memeory
64 {
65     unsigned int val;

        //Start
66     while(!input(HS));

        //Is Handshaking Line High ?
67     output_low(HS);

```

```

68         //Set Handshaking Line Low
            i2c_start();

        //I2C Start Pulse
69         i2c_write(0xA0);

        //Set PIC Type and Number
70         i2c_write(address);

    //Set Address
71         delay_ms(10);

        //Delay for PIC
72         i2c_start();

        //I2C Start Pulse
73         delay_ms(10);

        //Delay for PIC
74         i2c_write(0xA1);

        //Set PIC Type and Number
75         val=i2c_read();

        //Read from PIC
76         delay_ms(10);

        //Delay for PIC
77         i2c_stop();

        //I2C Stop Pulse
78         delay_ms(10);

        //Delay for PIC
79         input(HS);

        //Set Handshaking Line High
80         return(val);

    //End
81     }
82
83 void Write_Memory(unsigned int address,unsigned int data)
84                                     //Function for
    Writing to Memeory
85     {
86         while(!input(HS));

        //Is Handshaking Line High ?
86         output_low(HS);

        //Set Handshaking Line Low
87         i2c_start();

        //I2C Start Pulse
88         i2c_write(0xA0);

        //Set PIC Type and Number
89         i2c_write(address);

    //Set Address
90         i2c_write(data);

        //Write Data to Address
91         delay_ms(10);

        //Delay for PIC
92         i2c_stop();

        //I2C Stop Pulse

```

```
93         delay_ms(10);

          //Delay for PIC
94         input(HS);

          //Set Handshaking Line High
95     }
96
```