```
#include <xc.h>     //include the neccassary  header file
#define _XTAL_FREQ 6000000  //intialize the clock speed
void init(void);    //function declaration
void lcddata(unsigned char i);  //function declaration
void lcdcmd(unsigned char i);   //function declaration
void lcdoutput(int i);  //function declaration
void keyscan(void); //function declaration
void lcddec(int i); //function declaration
void indicator(void);   //function declaration
unsigned char array[30]={"BATT VOLT:    V"},
        bat[12]={"BATT LOW    "},bat1[12]={"BATT NORMAL"},bat2[12]={"BATT HIGH   "};
                                            //declare the array is compile time array
unsigned char a,x,value,dot='.';    //declare the char variable
unsigned int d2,d1,j;   //declare the int variable
float v,w;  //declare the float variable
int de=0;   //declare the int variable
void main()
{
    init();     //call the init function
    while(1)    //infinait while loop
    {
        keyscan();  //key scan for check the wich key pressed



    }
}
void init() //init function
{
TRISC=0x00;  //TRISC set as output
TRISD=0x00; //TRISD set as output
```

```
TRISB=0xF0; //TRISB set the first nibble as input
OPTION_REG&=0x7F; //option register to enable the internal pull up resisitor
lcdcmd(0x30); //function set command
__delay_ms(5);  //delay
lcdcmd(0x30); //function set command
__delay_ms(10);
lcdcmd(0x30);   //function set command
lcdcmd(0x38);   //display will set two line and font will set as default
lcdcmd(0x0C);   //curser will set as off and display on
}
void keyscan()  //keyscan function
{
   value=PORTB&=0xF0;  //port input data will be store in the value
   switch(value)   //switch case parameter of value
   {
      case 0xE0:  //case1
         j=20;   //set the intially as 15
         lcdcmd(0x80);
         lcdcmd(0x01);//set the location of display
         for(x=0;x<10;x++)
         {
            lcddata(array[x]);  //print in the display
         }
         lcdcmd(0x8E);   //set the location of the display
         for(x=14;x<15;x++)
         {
            lcddata(array[x]);  //set the final v
         }
         lcdoutput(j); //j will be passed to lcdoutput
         lcddata(dot); //print the dot
         lcddec(de);  //print the decimal value named as de
         indicator(); //indicator will be called for set the secondline
         __delay_ms(50); //delay
         break; //break
      case 0xD0:  //case 2
         if(j==22&&de==5) //checjk the j equal to 22 and de equal to 5
         {
            j=22;   //j will be set as 22
            de=5;   //de will be set as 5
            lcdoutput(j); //j will be set as lcdoutput
            lcddata(dot); //print the dot in the display
            lcddec(de); //de will be sent to the lcddec
            indicator(); //indicator will be called for second line printing
         }
         else
         {
            de++; //post increment of de
            if(de>=10)  //check the de smaller then or equalto 10
            {
               j++;    //post increment of j
               de=0;   //de will be set as 0
               lcdoutput(j); //j will be given to the lcdoutput
               lcddata(dot); //print the dot in the display
               lcddec(de);  //de will be sent to the lcddec
               indicator();    //indicator call
            }
            else
            {
               lcdoutput(j); //j will be sent to the lcd output
               lcddata(dot);   //dot will be print in the display
               lcddec(de); //de will be given to the lcdec
               indicator();
            }
          }
          __delay_ms(50);
          break;
        case 0xB0:
          if(j==15&&de==0) //ceck the j equal to 15 and de=0
          {
            j=15;   //j will set as 15
            de=0;   //de will be set as 0
            lcdoutput(j);   //sent the j data to print
            lcddata(dot);    //lcddata function
            lcddec(de);     //lcddec is used for print the dicmal function
            indicator();    //indicator is used for the print the low or high
          }
          else
          {
            de--; //post decrement of the de
            if(de<0) //check the de lesser then 0
            {
              j--; //post increment of j4
              de=9; //9 will be store in the de
              lcdoutput(j);   //j data will be given to the lcdoutput
              lcddata(dot);   //dot will be given to the lcddata
              lcddec(de);     //decimal will be given to the lcddec
              indicator();    //indicator is used for the print the low or high
            }
            else
            {
              lcdoutput(j);   //j data will be give to the lcdoutput
              lcddata(dot);  //dot will be given to the lcddata
              lcddec(de);     //decimal value will be given to the lcddec
```

```
          indicator();    //indicator is used for print the batt low or high
        }
      }
      __delay_ms(50);
      break;
    case 0x70:  //case 4
      j=17; //17 will be stored in the j
      de=7;
      lcdoutput(j);   //function call
      lcddec(de); //function call
      indicator();    //function call
      __delay_ms(50); //delay
      break;
  }
}
void lcdcmd(unsigned char i)    //lcdcmd function
{
  PORTC&=~(0x08); //set RS pin as 0
  PORTD=i;    //i will be sent to the display
  PORTC|=(0x01);  //enable pin set as 1
  __delay_ms(25); //delay
  PORTC&=~(0x01); //enable pin set as 0

}
void lcddata(unsigned char i) //lcddata function
{
  PORTC|=(0x08);  //set RS pin as 1
  PORTD=i;    //i will be sent to the display
  PORTC|=(0x01);  //enable pin set as 1
  __delay_ms(25); //delay
  PORTC&=~(0x01); ////enable pin set as 0
```

```
}
void lcdoutput(int i)
{
   d2=(unsigned char)((i%100)/10); //split the int value and store in the d2
   d1=(unsigned char)(i%10);   //split the int value and store in the d1
   lcdcmd(0x8A); //to set set the location of display
   lcddata(0x30+d2); //function call
   lcddata(0x30+d1);   //function call
}
void lcddec(int i) //lcddec set the decimal value
{
   lcdcmd(0x8D);  //location of the display
   lcddata(0x30+i); //int value convert to ascii
}
void indicator()    //indicator function
{
   w=(float)de/10;  //de/ by 10 used for convert into fraction
   v=(float)j+w;   //add the j and w and typecast and store in the v
   if (v>=15.0&&v<=17.5)
   {
      lcdcmd(0xC0); //location for the display
      for(a=0;a<=11;a++)  //loop
         lcddata(bat[a]);    //print the batt low
   }
   else if(v>17.5&&v<=20.5)
   {
      lcdcmd(0xC0);  //location for the display
      for(a=0;a<=11;a++) //loop
         lcddata(bat1[a]); //print batt normal
   }
```

```
  else
  {
    lcdcmd(0xC0);   //location for the display
    for(a=0;a<=12;a++) //loop
      lcddata(bat2[a]);   //batt high
  }
}
```