// LCD module connections

sbit LCD\_RS at RB0\_bit;

sbit LCD\_EN at RB1\_bit;

sbit LCD\_D4 at RB4\_bit;

sbit LCD\_D5 at RB5\_bit;

sbit LCD\_D6 at RB6\_bit;

sbit LCD\_D7 at RB7\_bit;

sbit LCD\_RS\_Direction at TRISB0\_bit;

sbit LCD\_EN\_Direction at TRISB1\_bit;

sbit LCD\_D4\_Direction at TRISB4\_bit;

sbit LCD\_D5\_Direction at TRISB5\_bit;

sbit LCD\_D6\_Direction at TRISB6\_bit;

sbit LCD\_D7\_Direction at TRISB7\_bit;

// End LCD module connections

char txt1[] = " BM for 48V ";

char Volttxt[]= " " ;

char Temptxt[]= " " ;

char Amptxt[] = " 00.0";

int Volt1, Volt2, Volt3, Volt4 ;

int Temp1, Temp2, Temp3, Temp4 ;

unsigned int Ampraw = 0 , chgAmp = 0 , DischgAmp = 0 ;

sbit PowerSW at RD0\_bit;

sbit DisChg1 at RD1\_bit;

sbit DisChg2 at RD2\_bit;

sbit DisChg3 at RD3\_bit;

sbit DisChg4 at RD4\_bit;

void initialize(void); // Initialization Function

void get\_Temp12(void) ; // Temperature Read Function

void get\_Temp34(void) ;

void get\_V\_T\_I\_Ctl(void); // Read Volt and CTL , Temp and Ctl , Current and Ctl

void get\_Volt12(void) ; // Voltage Read Function and Display

void get\_Volt34(void) ;

void get\_Amp(void); // Amp sensor

void main(){

initialize( );

while(1){

unsigned int tip = 0 ;

get\_Volt12( );

Lcd\_Out(1,1,Volttxt);

get\_Temp12( );

Lcd\_Out(2,1,Temptxt);

for(tip = 0 ; tip < 10 ; tip++){

get\_V\_T\_I\_Ctl();

Delay\_ms(100);

}

get\_Volt34( );

Lcd\_Out(1,1,Volttxt);

get\_Temp34( );

Lcd\_Out(2,1,Temptxt);

for(tip = 0 ; tip < 10 ; tip++){

get\_V\_T\_I\_Ctl();

Delay\_ms(100);

}

get\_Amp( );

Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

Lcd\_Out(1,1,Amptxt);

for(tip = 0 ; tip < 10 ; tip++){

get\_V\_T\_I\_Ctl();

Delay\_ms(100);

}

}

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

void get\_Temp12(void){

Temp1 = ADC\_Read(0)/ 2 ;

Temp1--;

Temptxt[0] = 'B';

Temptxt[1] = '1';

Temptxt[2] = '-';

Temptxt[3] = '>';

Temptxt[4] = (signed)(Temp1 / 10) + 48;

Temptxt[5] = (signed)(Temp1 % 10) + 48;

Temptxt[6] = 'C';

Temptxt[7] = ' ';

Temp2 = ADC\_Read(1)/ 2 ;

Temp2--;

Temptxt[8] = 'B';

Temptxt[9] = '2';

Temptxt[10] = '-';

Temptxt[11] = '>';

Temptxt[12] = (signed)(Temp2 / 10) + 48;

Temptxt[13] = (signed)(Temp2 % 10) + 48;

Temptxt[14] = 'C';

Temptxt[15] = ' ';

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

void get\_Temp34(void){

Temp3 = ADC\_Read(2)/ 2 ;

Temp3--;

Temptxt[0] = 'B';

Temptxt[1] = '3';

Temptxt[2] = '-';

Temptxt[3] = '>';

Temptxt[4] = (signed)(Temp3 / 10) + 48;

Temptxt[5] = (signed)(Temp3 % 10) + 48;

Temptxt[6] = 'C';

Temptxt[7] = ' ';

Temp4 = ADC\_Read(3)/ 2 ;

Temp4--;

Temptxt[8] = 'B';

Temptxt[9] = '4';

Temptxt[10] = '-';

Temptxt[11] = '>';

Temptxt[12] = (signed)(Temp4 / 10) + 48;

Temptxt[13] = (signed)(Temp4 % 10) + 48;

Temptxt[14] = 'C';

Temptxt[15] = ' ';

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

void get\_V\_T\_I\_Ctl(void){

Volt1 = ADC\_Read(4) ;

Volt2 = ADC\_Read(5) ;

Volt3 = ADC\_Read(6) ;

Volt4 = ADC\_Read(7) ;

Volt1 = Volt1 -Volt2 ; // 48 - 36

if (Volt1 > 145 ){

DisChg1 = 1 ;

}else

{

DisChg1 = 0 ;

}

Volt2 = VOlt2 -Volt3 ; // 36 - 24

if (Volt2 > 145 ){

DisChg2 = 1 ;

}else

{

DisChg2 = 0 ;

}

Volt3 = Volt3 -Volt4 ; // 24 -12

if (Volt3 > 145 ){

DisChg3 = 1 ;

}else

{

DisChg3 = 0 ;

}

if (Volt4 > 145 ){

DisChg4 = 1 ;

}else

{

DisChg4 = 0 ;

}

Temp1 = ADC\_Read(0)/ 2 ;

Temp2 = ADC\_Read(1)/ 2 ;

Temp3 = ADC\_Read(2)/ 2 ;

Temp4 = ADC\_Read(3)/ 2 ;

Ampraw = ADC\_Read(8);

if(Temp1 > 45 || Temp2 > 45 || Temp3 > 45 || Temp4 > 45 || Ampraw < 256 || Ampraw > 768 || Volt1 < 100 || Volt2 < 100 || Volt3 < 100 || Volt4 < 100 ){

PowerSW = 0 ;

}else{

PowerSW = 1 ;

}

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

void get\_Volt12(void){

Volt1 = ADC\_Read(4) ;

Volttxt[0] = 'B' ;

Volttxt[1] = '1' ;

Volttxt[2] = '=' ;

Volttxt[3] = (signed)(Volt1 / 100) + 48;

Volttxt[4] = (signed)((Volt1 % 100)/10) + 48;

Volttxt[5] = '.' ;

Volttxt[6] = (signed)((Volt1 % 100 )%10) + 48;

Volttxt[7] = 'V' ;

Volt2 = ADC\_Read(5);

Volttxt[8] = 'B' ;

Volttxt[9] = '2' ;

Volttxt[10] = '=' ;

Volttxt[11] = (signed)(Volt2 / 100) + 48;

Volttxt[12] = (signed)((Volt2 % 100)/10) + 48;

Volttxt[13] = '.' ;

Volttxt[14] = (signed)((Volt2 % 100 )%10) + 48;

Volttxt[15] = 'V' ;

//InttoStr(Volt1,Volttxt);

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

void get\_Volt34(void){

Volt3 = ADC\_Read(6) ;

Volttxt[0] = 'B' ;

Volttxt[1] = '3' ;

Volttxt[2] = '=' ;

Volttxt[3] = (signed)(Volt3 / 100) + 48;

Volttxt[4] = (signed)((Volt3 % 100)/10) + 48;

Volttxt[5] = '.' ;

Volttxt[6] = (signed)((Volt3 % 100 )%10) + 48;

Volttxt[7] = 'V' ;

Volt4 = ADC\_Read(7);

Volttxt[8] = 'B' ;

Volttxt[9] = '4' ;

Volttxt[10] = '=' ;

Volttxt[11] = (signed)(Volt4 / 100) + 48;

Volttxt[12] = (signed)((Volt4 % 100)/10) + 48;

Volttxt[13] = '.' ;

Volttxt[14] = (signed)((Volt4 % 100 )%10) + 48;

Volttxt[15] = 'V' ;

//InttoStr(Volt1,Volttxt);

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

void get\_Amp(void){

Ampraw = ADC\_Read(8);

// 512 is offset remove 2.5V

if(Ampraw > 515 ){

chgAmp = ((Ampraw - 512)\*100)/256 ;

Amptxt[0] = '+';

Amptxt[1] = (signed)(chgAmp/100)+48 ;

Amptxt[2] = (signed)((chgAmp%100)/10)+48 ;

Amptxt[3] = '.' ;

Amptxt[4] = (signed)((chgAmp%100)%10)+48 ;

}else if(Ampraw < 510){

dischgAmp = ((512 - Ampraw)\*100)/256;

Amptxt[0] = '-';

Amptxt[1] = (signed)(dischgAmp/100)+48 ;

Amptxt[2] = (signed)((dischgAmp%100)/10)+48 ;

Amptxt[3] = '.' ;

Amptxt[4] = (signed)((dischgAmp%100)%10)+48 ;

}else{

dischgAmp = 0 ;

chgAmp = 0 ;

}

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

void initialize(void ){

ANSEL = 0xFF; // Temperature and Volt input ADC

ANSELH = 0x04; // Current sensor input ADC

C1ON\_bit = 0; // Disable comparators

C2ON\_bit = 0; // Disable comparators not use

TRISA4\_bit = 1 ; // V1

TRISE0\_bit = 1 ; // V2

TRISE1\_bit = 1 ; // V3

TRISE2\_bit = 1 ; // V4

TRISA0\_bit = 1 ; // T1

TRISA1\_bit = 1 ; // T2

TRISA2\_bit = 1 ; // T3

TRISA3\_bit = 1 ; // T4

TRISB2\_bit = 1 ; // Isen

TRISD = 0; // PORTD are outputs

Lcd\_Init(); // Initialize LCD

Lcd\_Cmd(\_LCD\_CLEAR); // Clear display

Lcd\_Cmd(\_LCD\_CURSOR\_OFF); // Cursor off

Lcd\_Out(1,1,txt1); // Write text in first row

}

//++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++