

#### Timer\_Interrupt

Timer0\_Init  
Enable\_Timer0\_Interrupt  
extern void Disable\_Timer0\_Interrupt

#### Periph\_Inti

DIO\_Inti  
HeaterON  
Cooler\_ON  
Cooler\_Heater\_OFF

#### SevenSeg

Display7s  
SEGMENT\_Display\_2Digit\_without\_port  
Display\_OFF

#### Temp\_sensor

adc\_init  
adc\_Read  
adc\_Start\_Conv  
Temp\_calc

#### Set\_Temp

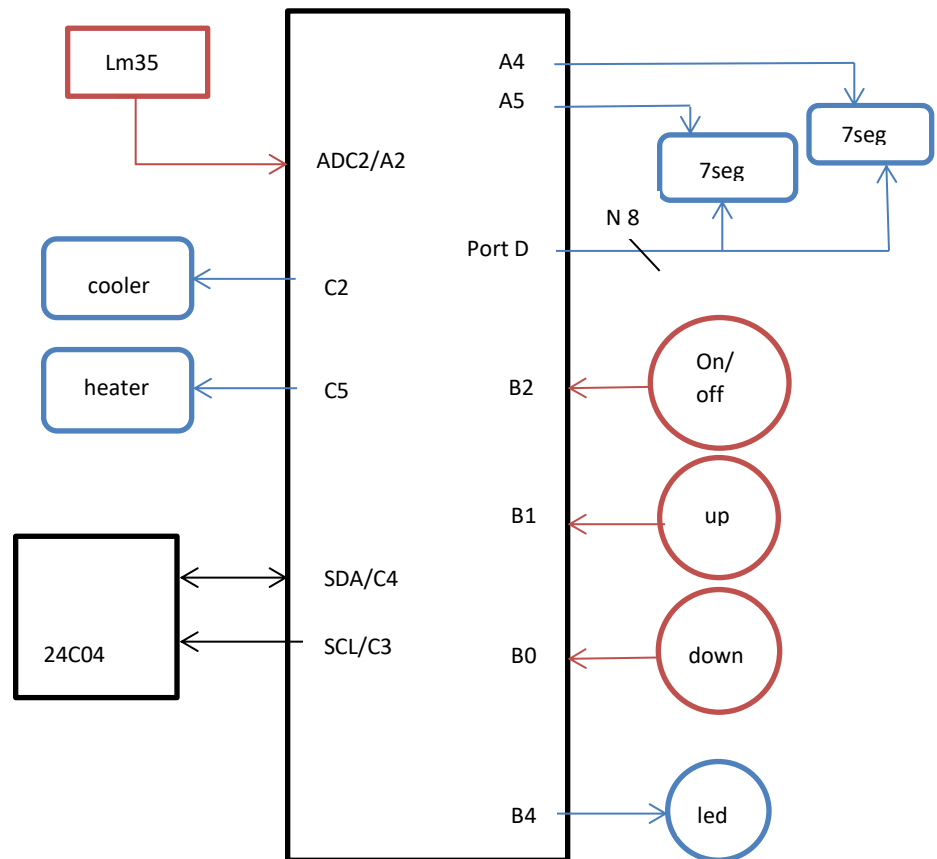
Temp\_Inc  
Temp\_Dec  
Temp\_push  
Calculate\_Temp\_Avg

#### E2PROM

E2PROM\_Read  
E2PROM\_Write

## Hardware diagram

PIC 16F877A



### 1) PORTA

**A2** : input for ADC2 to read the temperature sensor " Lm35" .

**A4** : output Enable for seven segment 1 .

**A5** : output Enable for seven segment 2 .

### 2) PORTB

**B0** : input to read the down button.

**B1** : input to read the up button.

**B2** : input to read the on/off button.

**B4** : output heater led .

### 3) PORTC

**C2** : output to enable or disable the cooler system .

**C5** : output to enable or disable the heater system .

**C3** : Shares the clock signal .

**C4** : Sends the data to and from between the pic and eeprom " 24c04" .

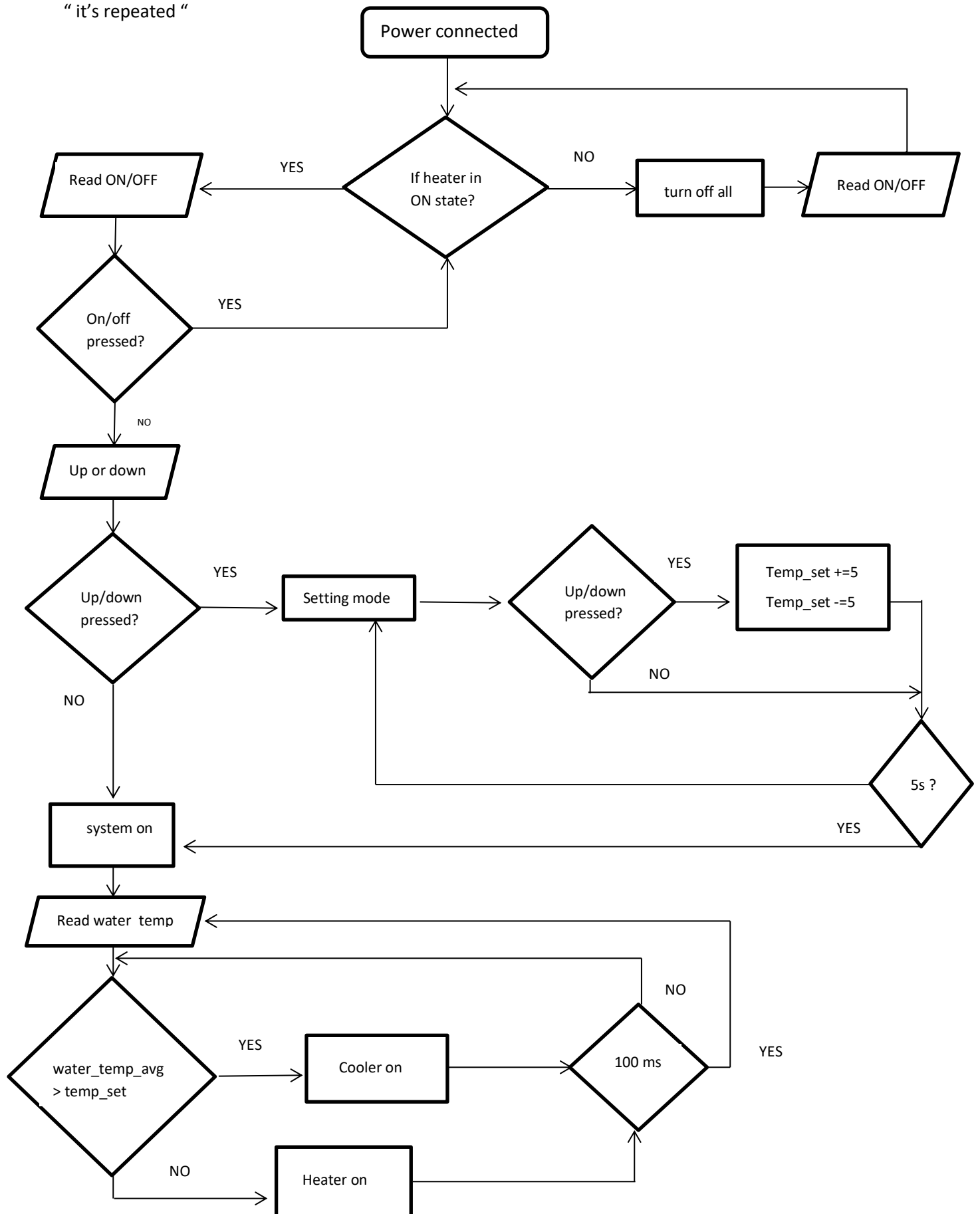
### 4) PORTD

All pins of port D is output shared between the 2 seven segment to write the data.

### 5) Timer0.

### Simple Flow chart for code :

“ it's repeated “



## Main Function

- The super loop contain 2 main loops polling on state of the Heater.

```
while(1) {  
  
    while( heater_state == HEATER_OFF ) {  
        ...  
    }  
  
    while( heater_state == HEATER_ON ) {  
        ...  
    }  
}
```

- “HEATER\_OFF” state loop :
  - Turn off all “heater – cooler – led- seven seg ).
  - Check if the “ON/OFF” button is pressed or not.

“flag\_Heater\_OFF” : turn them off just for the first time inside

The loop , after that it just check “on/off” button .

```
while( heater_state == HEATER_OFF) {  
  
    if( !Read_Bit(PORTB,ON_OFF_BUTTON )){  
        heater_state = HEATER_ON;  
        while( !Read_Bit(PORTB,ON_OFF_BUTTON ));  
    }  
  
    if(flag_Heater_OFF==1){  
        Display_OFF();  
        Cooler_Heater_OFF ();  
        flag_Heater_OFF=0;    }  
}
```

```
Main(){  
  
    ... ..  
  
    While(1){  
        While(heater_state==HEATER_OFF)  
            { ... .. }  
  
        While(heater_state==HEATER_ON) {  
            If( ... ) { ... }  
  
            else{  
                if( ... ){ ... }  
  
                while((Temp_Setting==Temp_Setting_ON) && ... .. )  
                    { ... ..  
                        ... .. }  
  
                while(( TempSet_Flag==1 ) && ... .. )  
                    { ... ..  
                        ... .. }  
            }  
        }  
    }  
}
```

2. "HEATER\_ON " state loop :

- Take a shot from the current water temperature.
- Set the " flag\_Heater\_OFF ".
- Check on IF " on/off " button.

```
if( !Read_Bit(PORTB,ON_OFF_BUTTON )) {  
  
    heater_state = HEATER_OFF;  
  
    while( !Read_Bit(PORTB,ON_OFF_BUTTON ));  
}
```

```
while( heater_state == HEATER_ON ){  
  
    ///// inti value water temp  
  
    if (flag_Heater_OFF==0){  
  
        adc_Start_Conv();  
  
        Water_Temp =Temp_calc();  
  
        flag_Heater_OFF=1;  
    }  
}
```

○ Else :

Check IF the " up or down " buttons .

- 1) Enter the setting mode by setting its flag " Temp\_setting " .
- 2) Enable timer interrupt to start count the 5 seconds .
- 3) Clear all other flags .

```
else {  
  
    if ( (!Read_Bit(PORTB,UP_Button)) || (!Read_Bit(PORTB,DOWN_Button))) && (Temp_Setting==Temp_Setting_OFF) )  
    {  
  
        /////////////// START COUNT THE 5 SECONDS FOR THE SETTING MODE////////////////////  
        TMR0 = 155;  
        cnt_5s_Flag=0;  
        Enable_Timer0_Interrupt();  
  
        Temp_Setting=Temp_Setting_ON;  
        TempSet_Flag=0;  
        cnt_Temp_Read_Flag=0;  
        flag_heat_element_on=0;  
        Cooler_Heater_OFF ();  
        while((!Read_Bit(PORTB,UP_Button)) || (!Read_Bit(PORTB,DOWN_Button)));  
    }  
}
```

While “ **Temp\_Setting\_ON** ”

- 1) If **UP** button is pressed :  
Set 155 to the 8 bit register TMR0 of timer0 and Restart the 5 seconds by clearing the counter flag .

Then call **Temp\_Inc** function to increase the setting temperature by 5 degrees .

- 2) If **DOWN** button is pressed :  
Restart the 5 seconds counter.

Then call **Temp\_Dec** function to decrease the setting temperature by 5 degrees .

```
else {
    ...
    ...
    while((Temp_Setting==Temp_Setting_ON)&& Read_Bit(PORTB,ON_OFF_BUTTON )){

        if(!Read_Bit(PORTB,UP_Button))
        { // up button pressed
            TMR0 = 155;
            cnt_5s_Flag=0;

            Temp_Inc();

            while(!Read_Bit(PORTB,UP_Button));

        }

        else if((! Read_Bit(PORTB,DOWN_Button)) )
        { // DOWN button pressed
            TMR0 = 155;
            cnt_5s_Flag=0;

            Temp_Dec();

            while(!Read_Bit(PORTB,DOWN_Button));

        }

        if(Display_delay_Flag==1)
        { SEGMENT_Display_2Digit_without_port(temperature_set) ; }

        else {Display_OFF();}

    }
}
```

- 3) **Display\_delay\_Flag** is toggled by the interrupt every 1 second to flash the seven segment .

If the 5 seconds passed without press any up or down button the flag “ **Temp\_setting** ” set to “**Temp\_Setting\_OFF** ”, break the loop and set “ **TempSet\_Flag** ” to 1 by the interrupt .

the timer interrupt do not effect on “ **cnt\_5s\_flag** ”and “**Display\_delay\_Flag** ” anymore out of this loop so I will use it in reading the temperature sensor every 100ms and toggle heater led every 1s .

```
void interrupt() {
    TMR0 = 155;

    if(Temp_Setting==Temp_Setting_ON){
        cnt_5s_Flag++;
    }

    ....
    ....
    ....

    if((cnt_5s_Flag%1000)==0){
        Display_delay_Flag^=1;
    }

    if(cnt_5s_Flag==5000){
        Temp_Setting=Temp_Setting_OFF;
        TempSet_Flag=1;
    }

    ....
    ....

    INTCON = 0x20;
}
```