ENGINEERING TEAM PROJECT

University Technology of PETRONAS

AUTOMATIC RETRACTABLE WIPER SYSTEM [ARWIS]

ELECTRICAL & ELECTRONIC PART

Prepared by: Luqman Radzuan

www.electroniccircuitbook.com

INTRODUCTION

In this project we divide the electrical & electronic discussion into 2 parts.

1. Software

2. Hardware

SOFTWARE

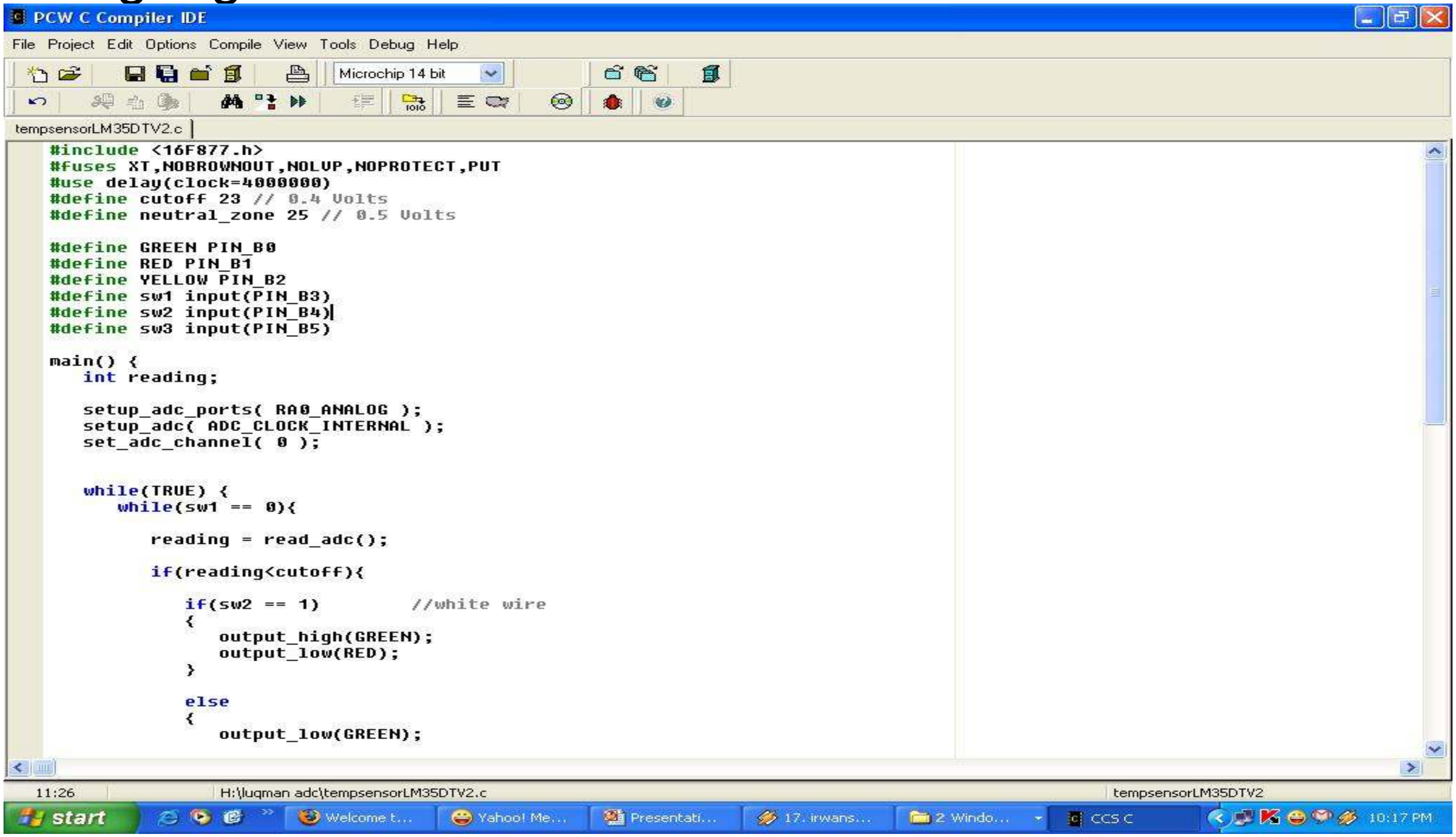
- C language
- PCWH compiler
- UP00A programmer
- Flow Chart & Source Code
- Program description
- Calculation

C LANGUAGE

- It’s a common programming language
- Easier to understand
- Require less time to program.

PCWH COMPILER

- A program that compile C language to machine language.

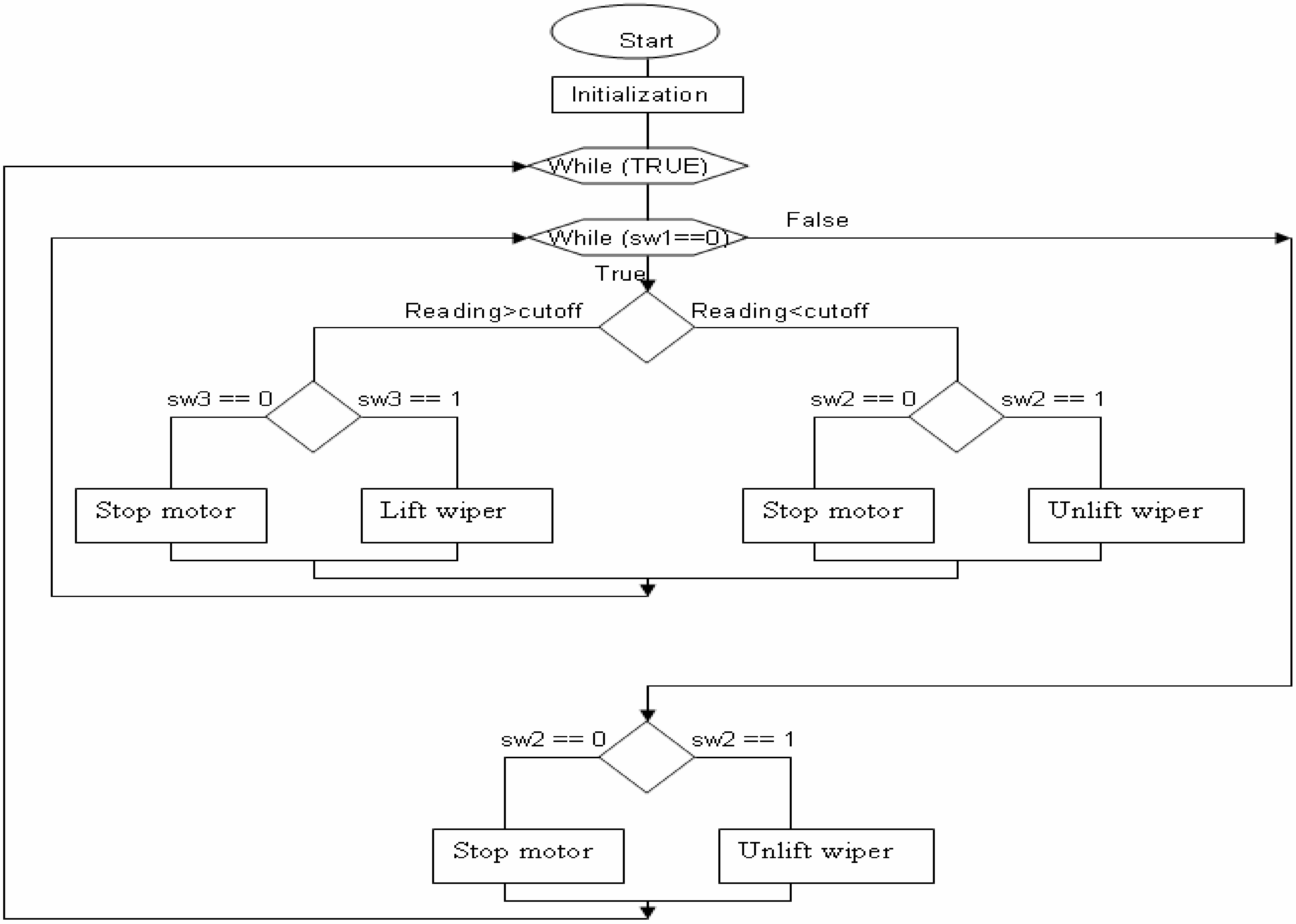


```
#include <16F877.h>
#fuses XT,NOBROWNOUT,NOLVP,NOPROTECT,PUT
#use delay(clock=4000000)
#define cutoff 23 // 0.4 Volts
#define neutral_zone 25 // 0.5 Volts

#define GREEN PIN_B0
#define RED PIN_B1
#define YELLOW PIN_B2
#define sw1 input(PIN_B3)
#define sw2 input(PIN_B4)
#define sw3 input(PIN_B5)

main() {
    int reading;

    setup_adc_ports( RA0_ANALOG );
    setup_adc( ADC_CLOCK_INTERNAL );
    set_adc_channel( 0 );


    while(TRUE) {
        while(sw1 == 0){

            reading = read_adc();

            if(reading<cutoff){

                if(sw2 == 1)          //white wire
                {
                    output_high(GREEN);
                    output_low(RED);
                }

                else
                {
                    output_low(GREEN);
```

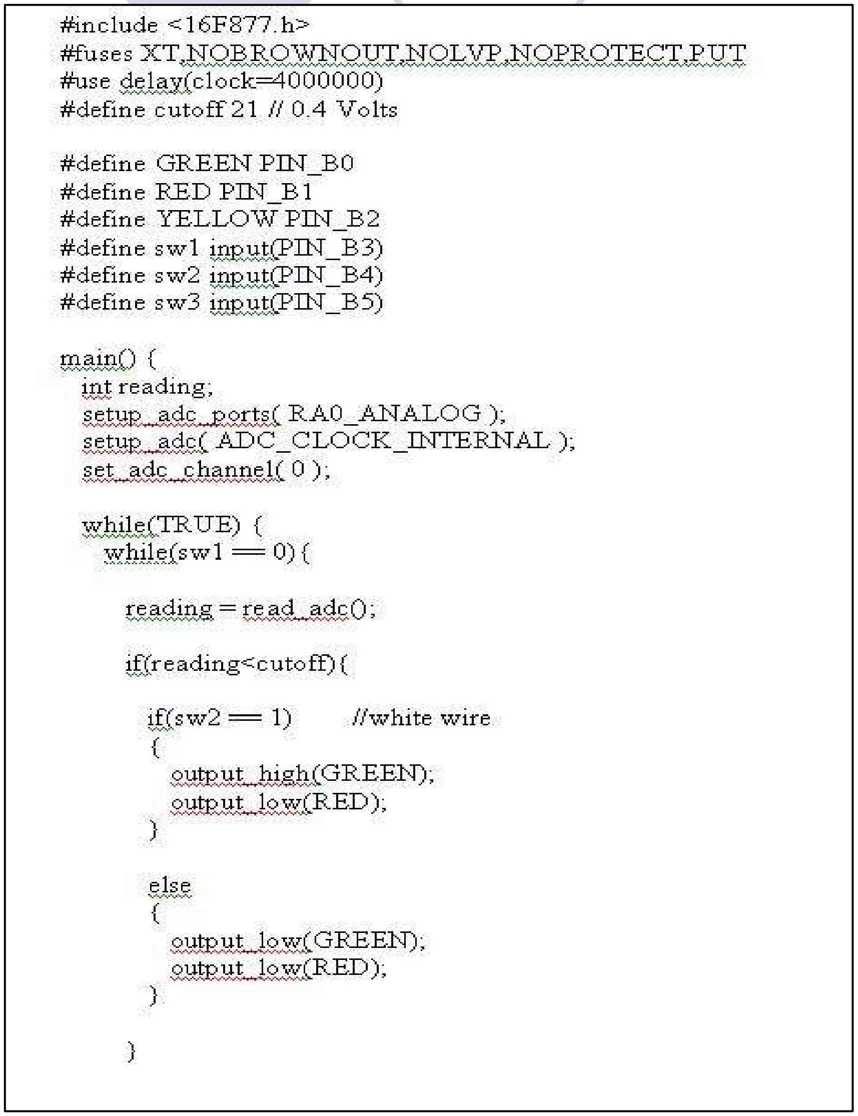
UP00A PROGRAMMER

- The programmer that is used to download the program onto the PIC16F877

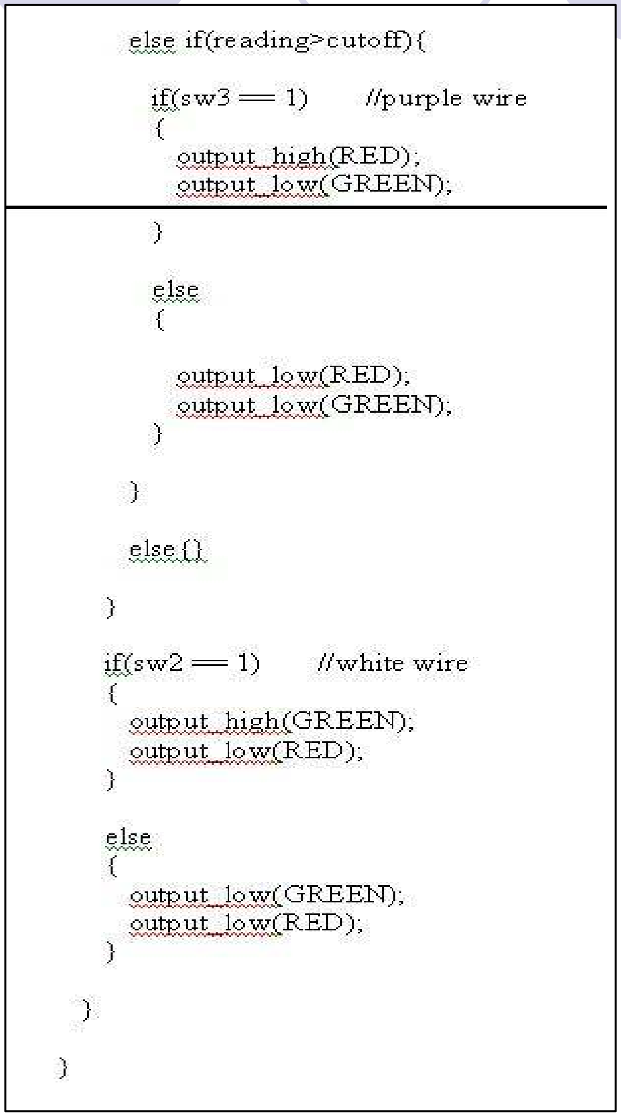
FLOW CHART


Start
Initialization
While (TRUE)
While (sw1==0)
False
True
Reading>cutoff
Reading<cutoff
sw3 == 0
sw3 == 1
sw2 == 0
sw2 == 1
Stop motor
Lift wiper
Stop motor
Unlift wiper
sw2 == 0
sw2 == 1
Stop motor
Unlift wiper

SOURCE CODE



```
#include <16F877.h>
#fuses XT,NOBROWNOUT,NOLVP,NOPROTECT,PUT
#use delay(clock=4000000)
#define cutoff 21 // 0.4 Volts

#define GREEN PIN_B0
#define RED PIN_B1
#define YELLOW PIN_B2
#define sw1 input(PIN_B3)
#define sw2 input(PIN_B4)
#define sw3 input(PIN_B5)

main() {
  int reading;
  setup_adc_ports( RA0_ANALOG );
  setup_adc( ADC_CLOCK_INTERNAL );
  set_adc_channel( 0 );

  while(TRUE) {
    while(sw1 == 0) {

      reading = read_adc();

      if(reading<cutoff){

        if(sw2 == 1)      //white wire
        {
          output_high(GREEN);
          output_low(RED);
        }

        else
        {
          output_low(GREEN);
          output_low(RED);
        }

      }
```



```
      else if(reading>cutoff){

        if(sw3 == 1)      //purple wire
        {
          output_high(RED);
          output_low(GREEN);
        }

        else
        {

          output_low(RED);
          output_low(GREEN);
        }

      }

      else {}

    }

    if(sw2 == 1)      //white wire
    {
      output_high(GREEN);
      output_low(RED);
    }

    else
    {
      output_low(GREEN);
      output_low(RED);
    }

  }

}
```

CALCULATION

- By default, the analog to digital converter is 8 bits. Thus, a range of 0 to 5 volts analog is represented by the numbers 0-255. The A/D reading can be converted to volts by the formula:

  Volts = Reading*(5.0/255)

  Reading = Volts*(255/5)

  Voltage produce by temperature sensor at 40°C,

  Vts = 10m*40

  = 400mV

  = 0.4V

  Reading = 0.4*(255/5)

  = 20.4

  = 21

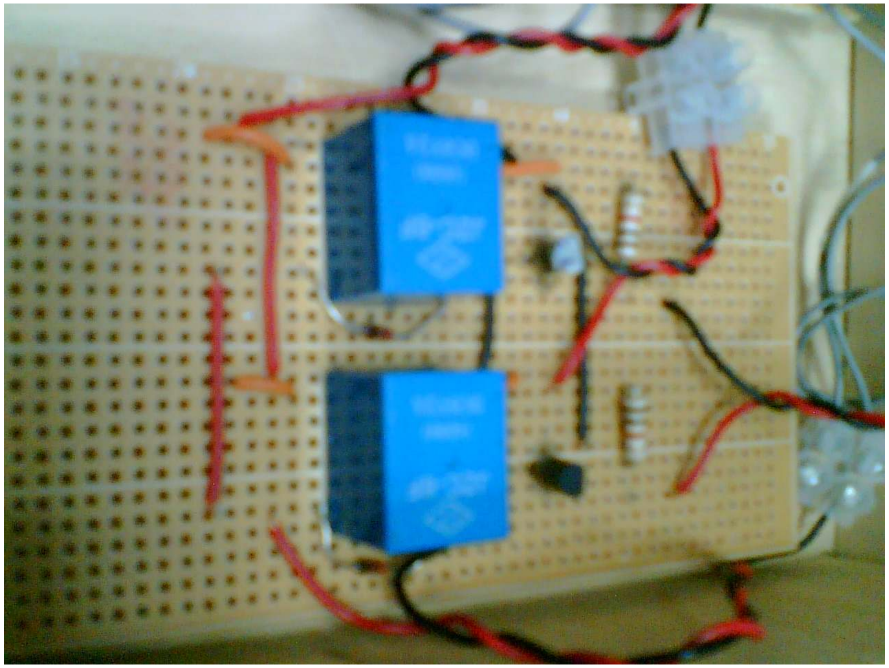
- So in the programming we set 21 as the limit where the temperature sensor exceeds 40°C.

HARDWARE

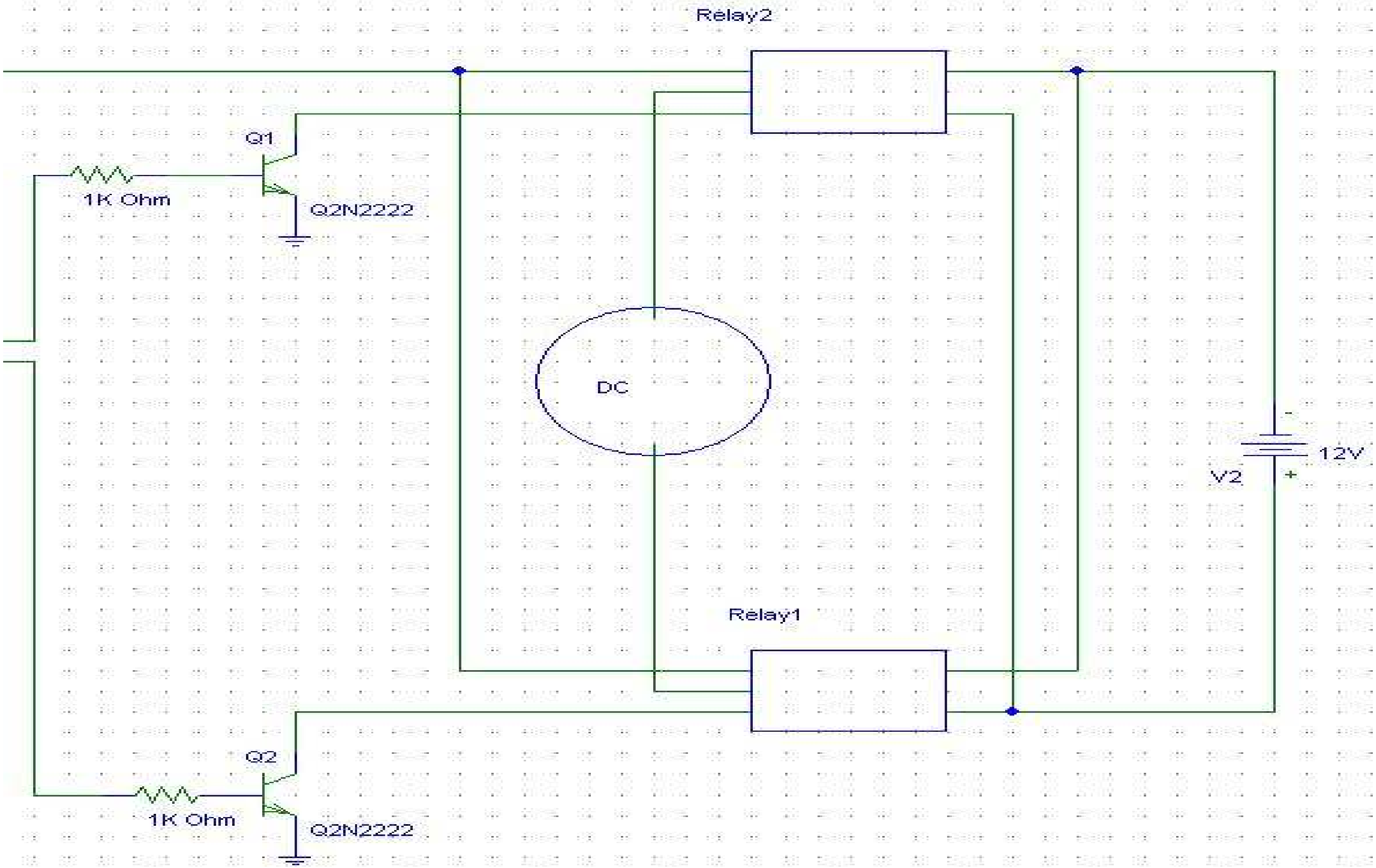
- Circuit Implementation
  - Analog Circuit
  - Digital Circuit

ANALOG CIRCUIT

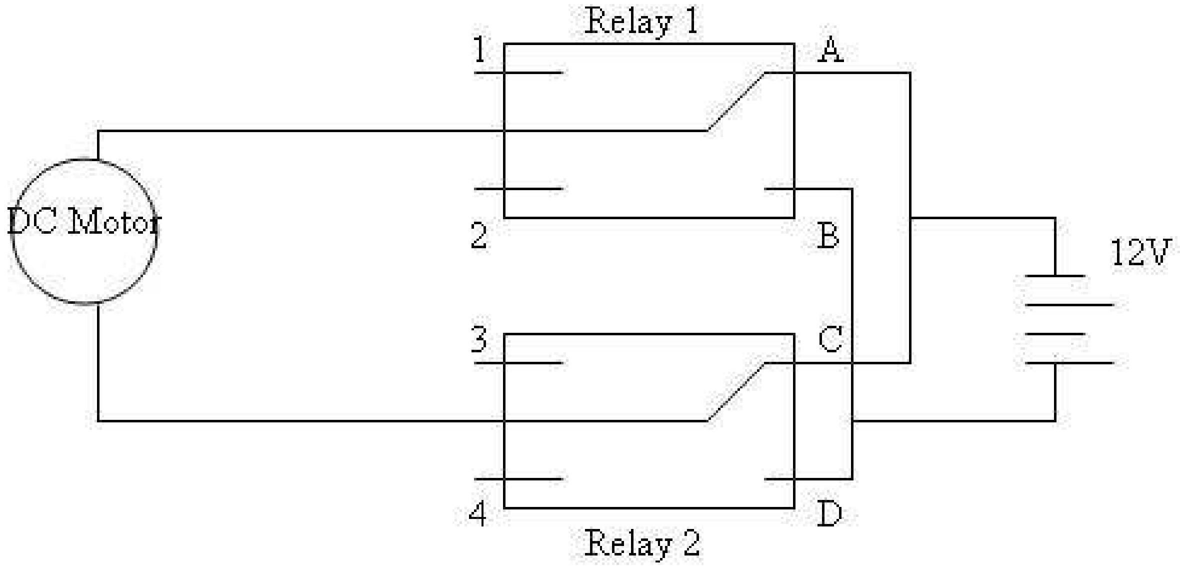
- The function of this circuit is to interchange the terminal of the DC motor so that it can turn clockwise and counter-clockwise.
- We achieve the goal by using two relays.



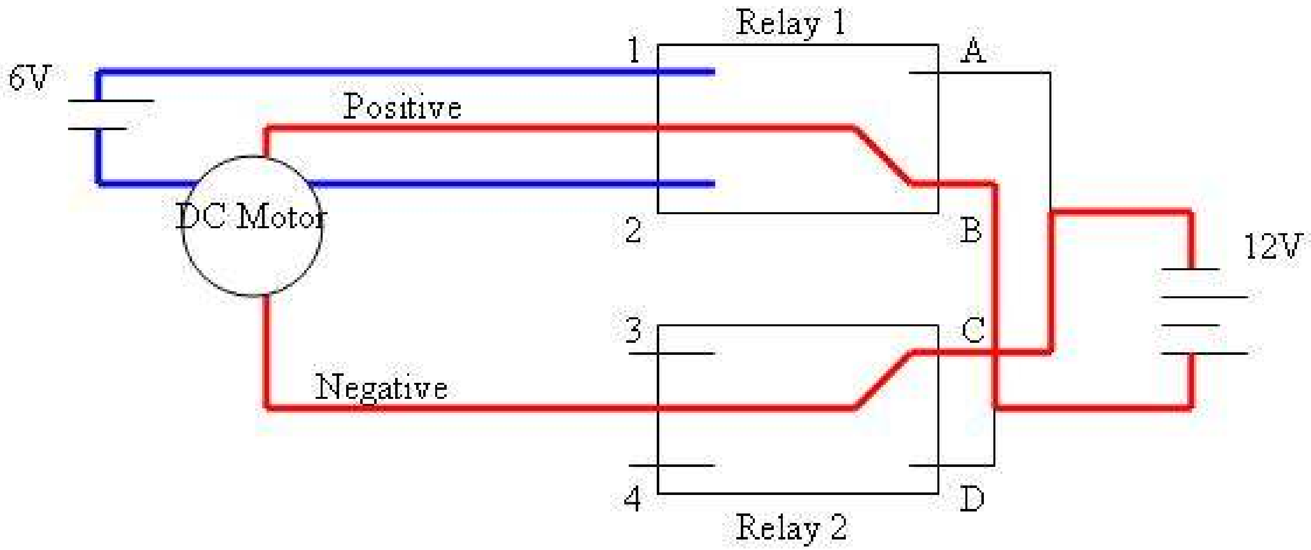
ANALOG CIRCUIT SCHEMATIC



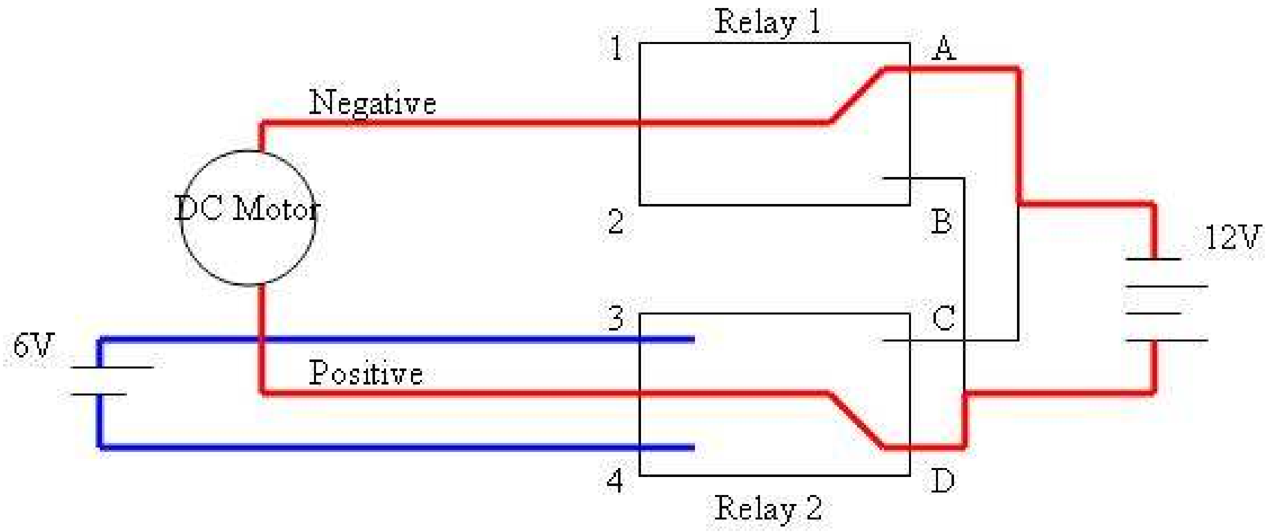
- Initially, both of the DC motor terminal are connected to the negative terminal of the 12V battery. Thus the motor will not turn.



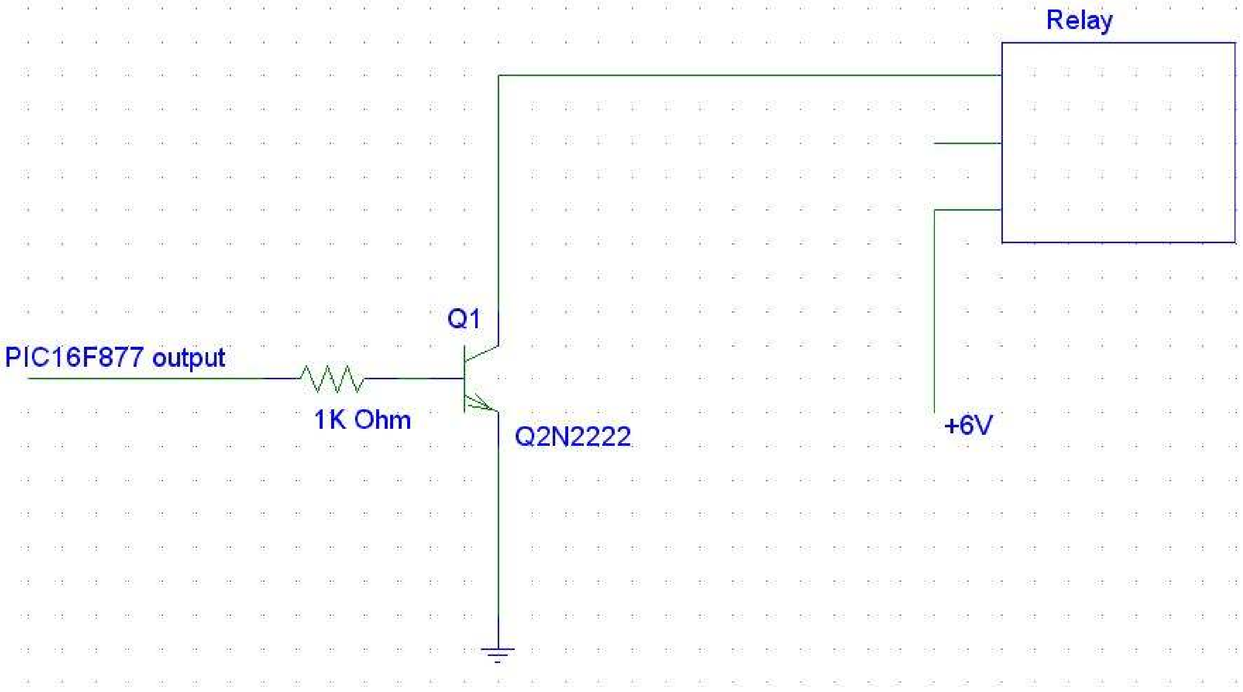
- When relay1 is activated, the inner connection of relay1 change from A to B. This will complete the circuit.



- When relay2 is activated, the inner connection of relay2 change from C to D. This will complete the circuit.



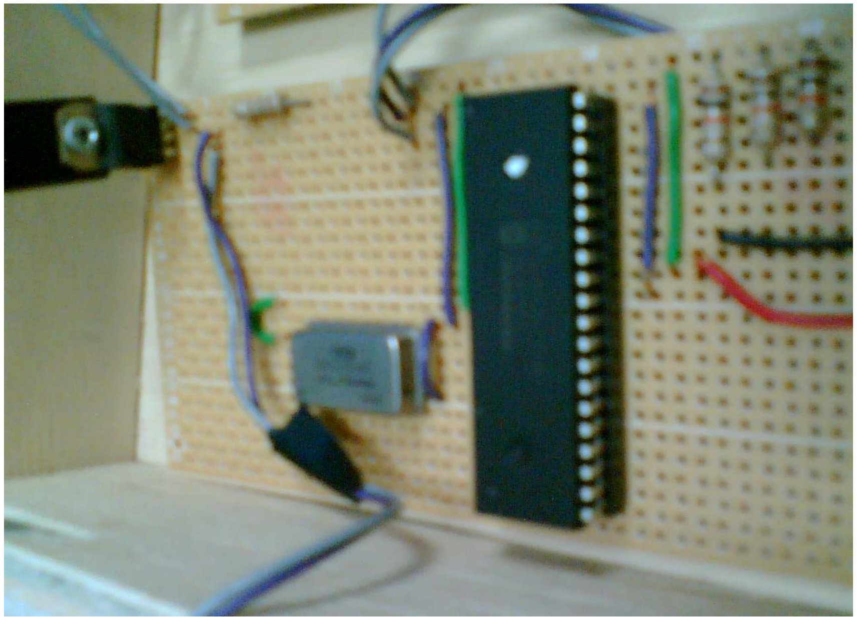
To activate the relay, we need to have at least 6V of supply. The output of the PIC16F877 only generates 4.6V. This is insufficient to trigger the relay. We solve the problem by using the 2N2222 BJT. As we all know, BJT can act as a switch.

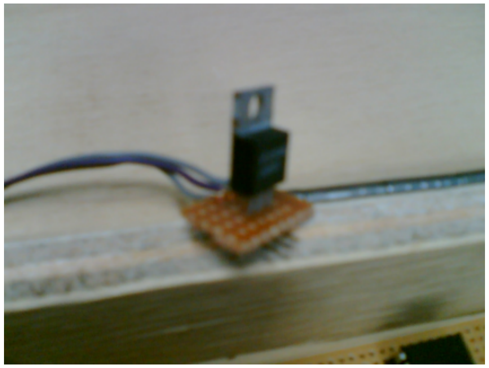


From the diagram above, the current from collector(C) will be allowed to pass to emitter(E) only if there is current flowing in from base(B) which is from the PIC16F877. When the PIC16F877 output is HIGH, the current will flow from collector to emitter and complete the circuit. Thus the relay will be activated.

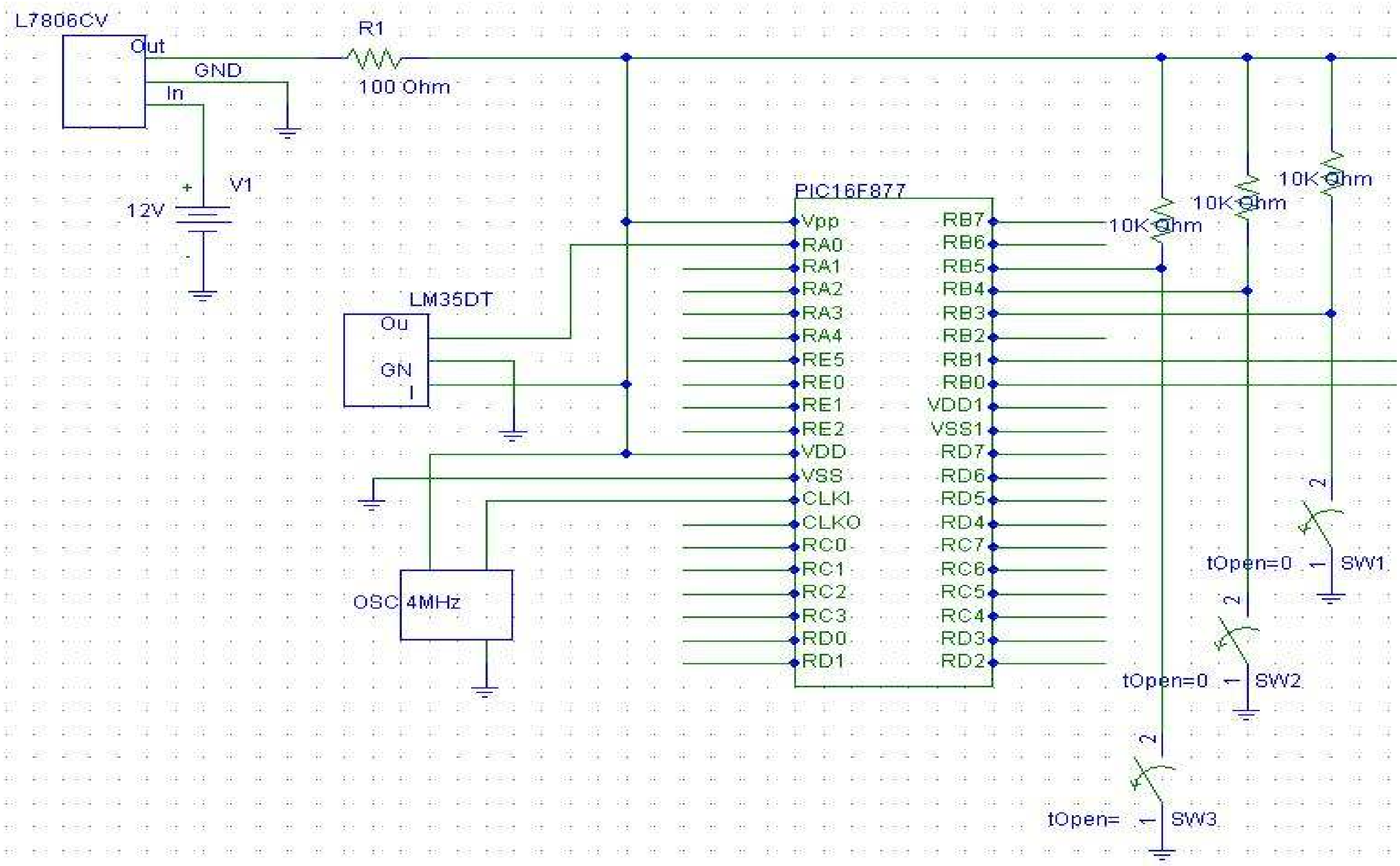
DIGITAL CIRCUIT

- The function of this circuit is to control the DC motor based on the temperature changes.
- It consist of the PIC16F877,LM35DT(temperature sensor),4MHz oscillator,L7806CV(voltage regulator) and three 10K Ohm resistor.

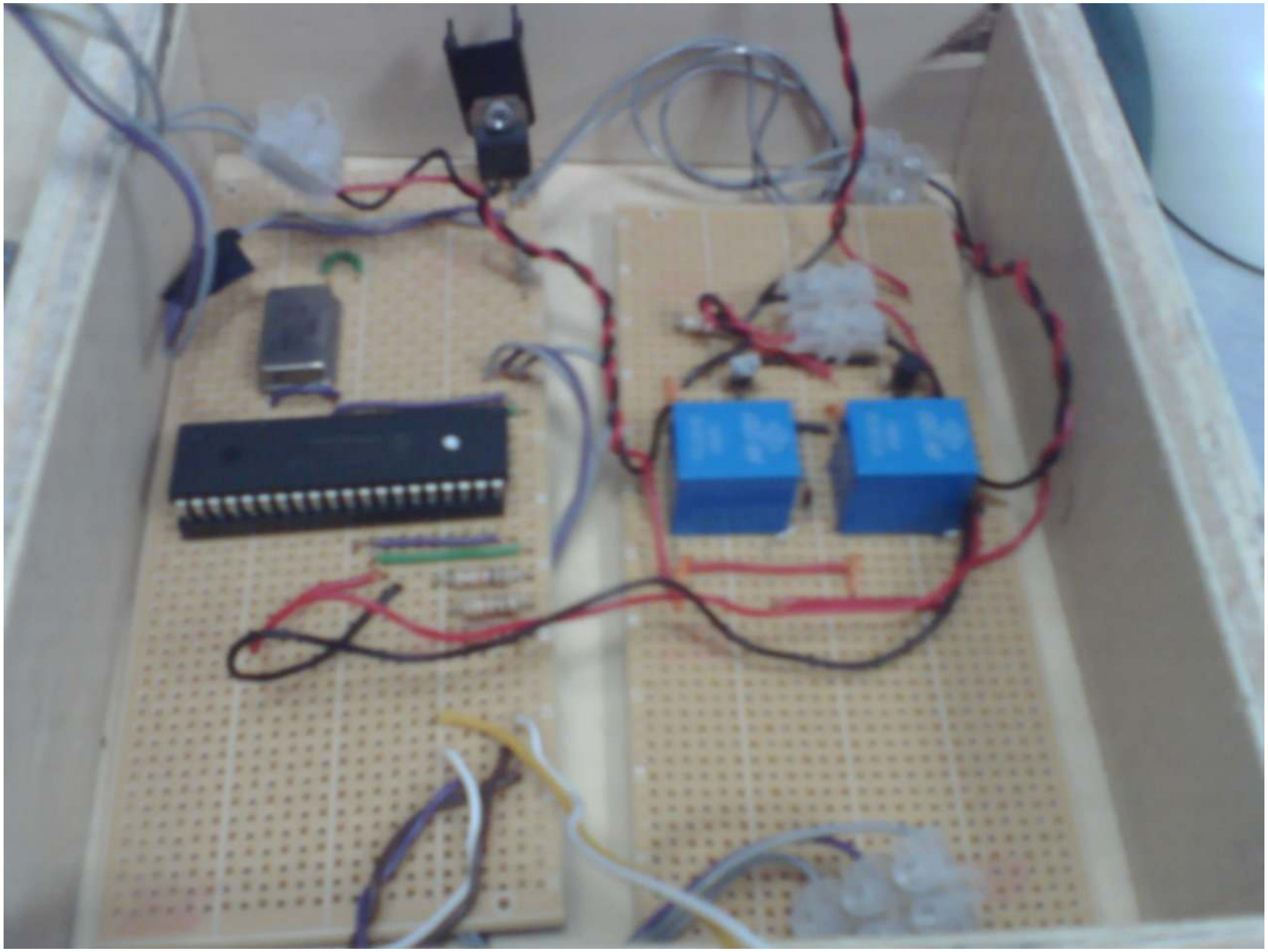




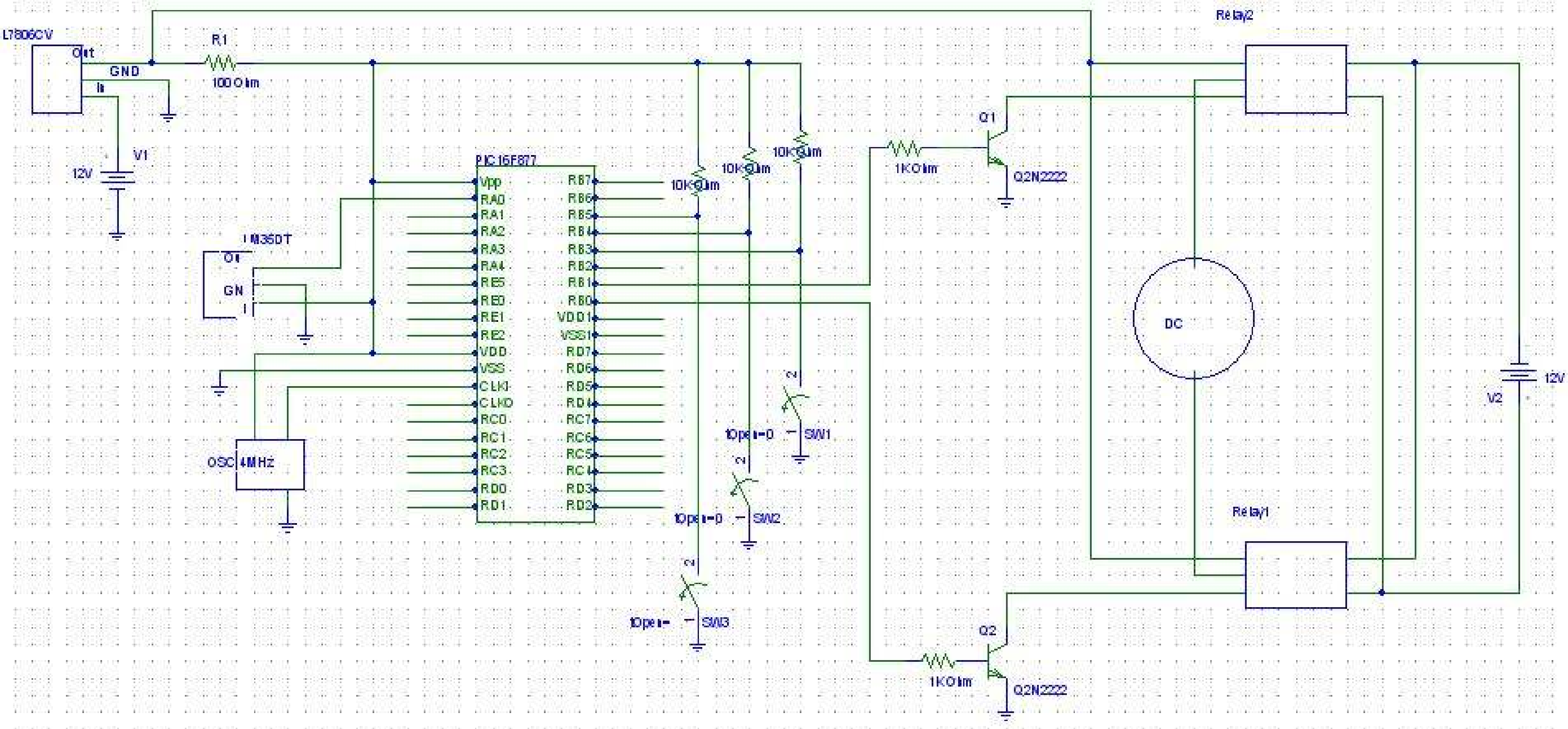
DIGITAL CIRCUIT SCHEMATIC



MAIN CIRCUIT



MAIN CIRCUIT SCHEMATIC



L7806CV
Out
GND
In
R1
100 Ohm
V1
12V
PIC16F877
Vpp
RA0
RA1
RA2
RA3
RA4
RE5
RE0
RE1
RE2
VDD
VSS
CLKI
CLKO
RC0
RC1
RC2
RC3
RD0
RD1
RB7
RB6
RB5
RB4
RB3
RB2
RB1
RB0
VDD1
VSS1
RD7
RD6
RD5
RD4
RC7
RC6
RC5
RC4
RD3
RD2
LM35DT
Out
GN
I
OSC 4MHz
10K Ohm
10K Ohm
10K Ohm
SW1
SW2
SW3
1Open=0
1Open=0
1Open=
1KOhm
1KOhm
Q1
Q2
Q2N2222
Q2N2222
Relay2
Relay1
DC
V2
12V