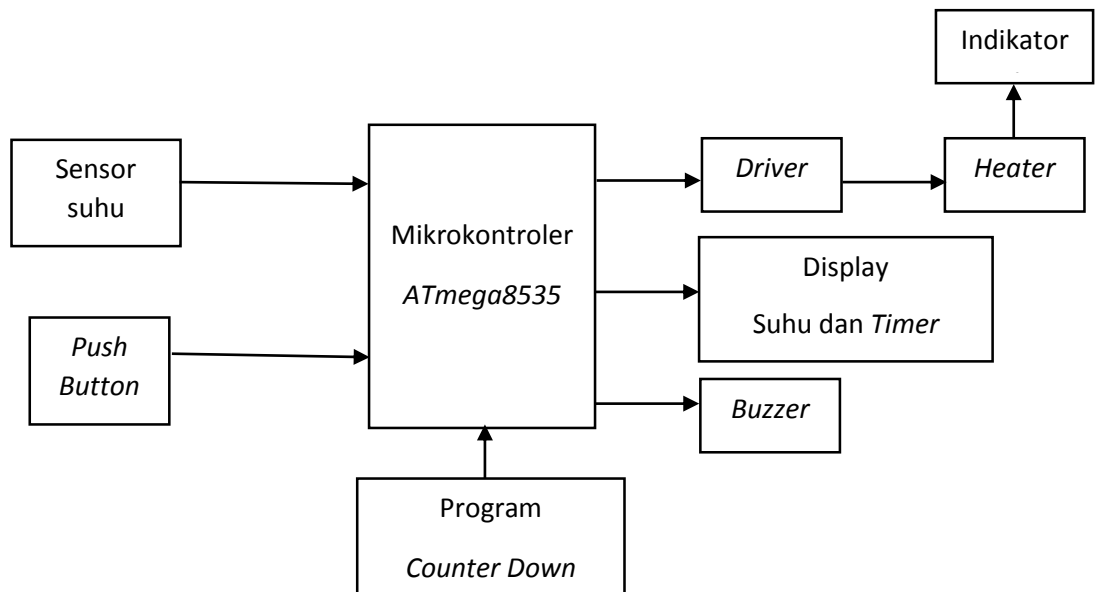


BAB III

METODE PENELITIAN

3.1. Blok Diagram

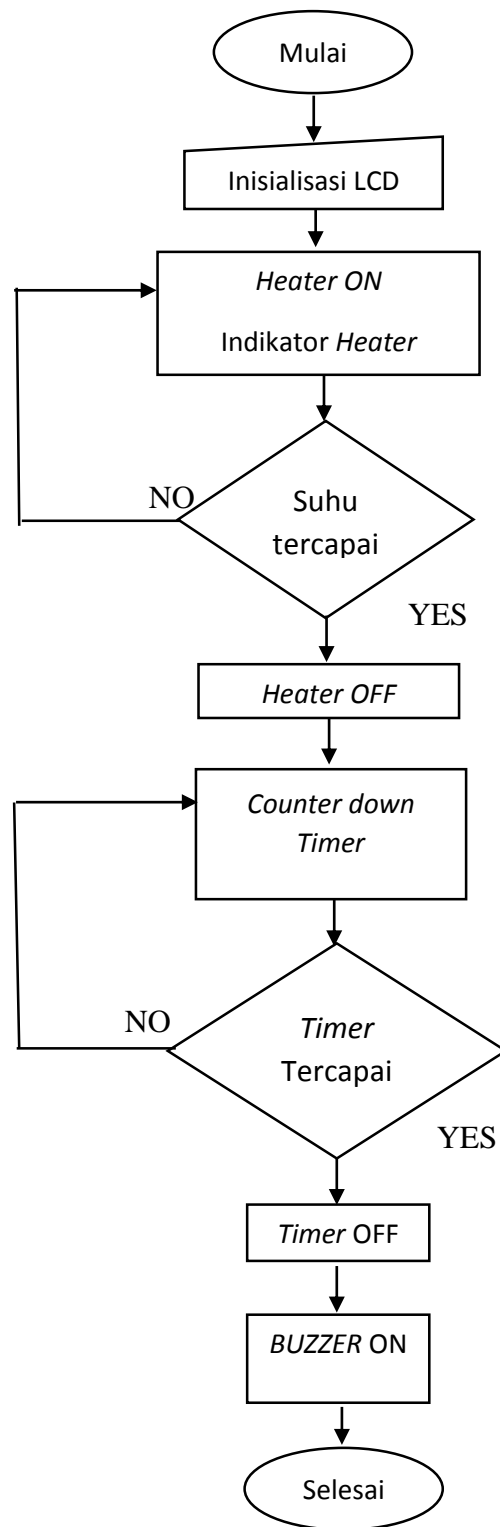


Gambar 3.1. Blok Diagram

Cara kerja Blok Diagram

Ketika *push button* ditekan maka mikrokontroler *ATmega8535* akan memberikan logika yang akan mengaktifkan *driver heater* sehingga *heater* akan bekerja, suhu dari *heater* akan dibaca oleh sensor LM35 yang kemudian ditampilkan didisplay 16x2. Setelah suhu tercapai maka *heater off* dan program *timer* yang ada pada IC mikrokontroler akan menghitung mundur. Setelah *timer* selesai maka *buzzer* akan bekerja/bunyi untuk menandakan bahwa penyeterilan sudah selesai.

3.2 Diagram Alir



Gambar 3.2. Diagram Alir

Penjelasan Diagram Alir

1. Mulai

Untuk memulai program.

2. Inisialisasi LCD

Sebelum mengerjakan program, Mikrokontroler melakukan persiapan ke LCD.

3. Start

Ketika start ditekan *heater* mulai bekerja. Setelah suhu tercapai (100°C) maka *heater off* dan program *couter down* bekerja.

4. Couter Down Timer

Terjadi proses perhitungan mundur waktu yang telah disetting pada mikroontroler. *Buzzer* akan berbunyi apabila *Timer* habis dan menandakan proses sterilisasi selesai.

5. Suhu Setting < Suhu Sensor

Apabila suhu pada settingan lebih kecil dari pada suhu sensor, maka *heater* akan nyala secara otomatis.

6. Suhu Sensor > 100°C

Pada alat ini dibatasi suhu maksimum adalah 100°C . Suhu akan naik terus mencapai suhu setting, apabila suhu melebihi settingan maka *heater* akan mati.

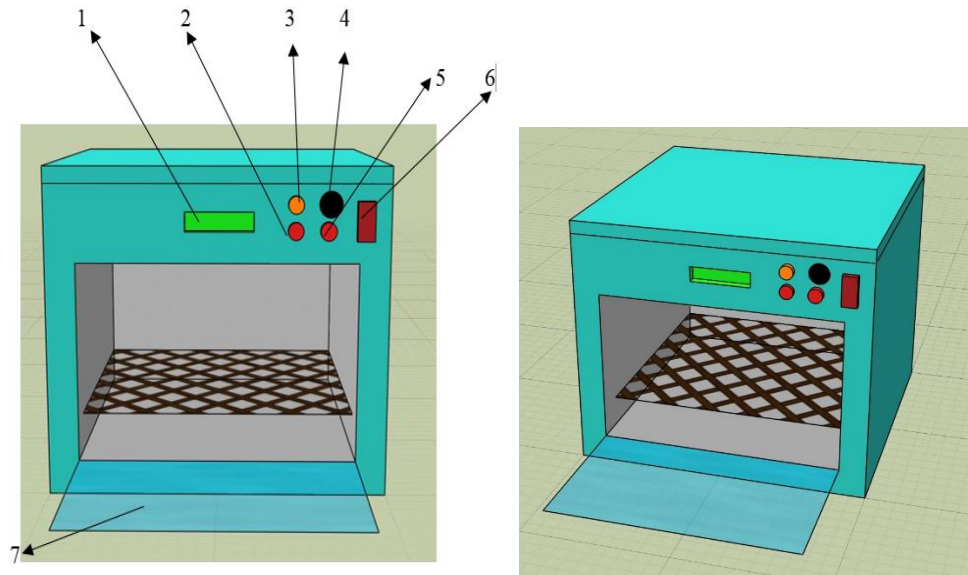
7. Selesai

Menandakan bahwa sterilisasi selesai.

Cara Kerja alat

Cara kerja modul sterilisator botol susu bayi berbasis mikrokontroler ini, yaitu ketika *power ON/OFF* dalam posisi *ON* maka seluruh rangkaian akan mendapatkan tegangan dari *power supply* sebesar +5V DC. Kemudian, Inisialisasi LCD dan masuk ke menu *setpoint*. Ketika tombol start ditekan maka mikrokontroler akan memberikan logika ke PORTD.5=0 untuk menghidupkan SSR dan sensor suhu, SSR disini berfungsi sebagai saklar untuk menghidupkan *heater* kaca yang sumber nya dari 220 Volt PLN. *Outputan* sensor LM35 masuk kerangkaian *buffer* untuk menguatkan tegangan yang masuk ke ADC mikrokontroler *ATmega8535*, sensor suhu akan mendeteksi suhu pada ruangan sterilisasi yang dipanasi oleh *heater*, lalu data analog yang didapat akan diolah oleh program ADC yang ada di mikrokontroler *ATmega8535* menjadi data digital kemudian suhu yang terbaca akan ditampilkan di display LCD 16x2, rangkaian LCD 16x2 diletakkan di PORT.C pada ic mikrokontroler *ATmega8535*. Apabila suhu pada ruangan sterilisasi tercapai maka mikrokontroler akan memberikan logika pada PORTD.5=0 sehingga *heater* akan mati dan *timer* bekerja selama 5 menit. Jika suhu kurang dari 100°C maka mikrokontroler akan memberikan logika lagi untuk menghidupkan SSR dan *heater* kaca. Kemudian bila waktu habis maka mikrokontroler akan merberikan logika ke PORTD.1=1 untuk mengaktifkan *buzzer*. Rangkaian *buzzer* difungsikan sebagai penanda atau isyarat bahwa proses dari sterilisasi selesai.

3.3. Diagram Mekanis



Gambar 3.3. Desain alat tampak dari depan

Keterangan :

1. LCD 16 X 2
2. Tombol START
3. Indikatar *Heater*
4. *Buzzer*
5. *Reset*
6. *Power*
7. Pintu

3.4. Keseluruhan Alat dan Bahan

1. Elemen pemanas(*Heater Kaca*)
2. Resistor
3. Kapasitor

4. LM35
5. *ATmega8535*
6. *LCD character 16x2*
7. *SSR (Solid State Relay)*
8. *Buzzer*

3.5. Jenis Penelitian

Jenis penelitian yang dipakai untuk melakukan penelitian ini menggunakan metode eksperimen yaitu alat sterilisator botol susu bayi berbasis mikrokontroler.

3.6. Variabel Penelitian

3.6.1 Variabel Bebas

Variabel bebasnya adalah bakteri.

3.6.2 Variabel Tergantung

Sebagai variabel tergantung adalah sensor suhu (LM35) bekerja tergantung dari heater menyala.

3.6.3 Variabel Terkendali

Sebagai variabel terkendali adalah monitoring timer dan LCD, menggunakan IC ATmega8535.

3.7. Definisi Oprasional

Dalam kegiatan operasionalnya, *varaiabel-variabel* yang digunakan dalam perencanaan pembuatan modul, baik *variabel* terkendali, tergantung dan bebas memiliki fungsi-fungsi antara lain:

- 3.7.1. Sensor LM35 digunakan untuk pengontrolan suhu.
- 3.7.2. *Heater* kaca digunakan sebagai penyeteril botol susu bayi.
- 3.7.3. Atmega8535 sebagai pengatur dari *heater* dan *timer*.

3.8. Teknik Analisa Data

Pengukuran tegangan pada beberapa titik test point dilakukan beberapa kali dalam percobaan. Kemudian hasil pengukuran tersebut dibandingkan dengan angka standart dan berapa nilai rata-rata, standart deviasi(SD), ketidakpastian dan error dengan rumus seperti dibawah ini:

3.8.1. Rata-rata

Rata-rata dalam perkataan sehari-hari, orang sudah menafsirkan dengan rata-rata hitung. Dan arti sebenarnya adalah bilangan yang di dapat dari hasil pembagian jumlah nilai data ooleh banyaknya data dalam kumpulan pengukuran tersebut. Dinyatakan dengan rumus :

$$\bar{x} = \frac{\sum x_i}{n} \dots\dots\dots (1)$$

Keterangan:

$\sum x_i$: Jumlah X sebanyak i

n : Banyak data

\bar{x} : Rata-rata

3.8.2. Simpangan (*error*)

Merupakan selisih dari rata-rata nilai terhadap masing-masing nilai yang di ukur. Dinyatakan dengan rumus :

$$\text{Simpangan} = x - \bar{x} \quad \dots\dots\dots (2)$$

Keterangan :

X : data x

\bar{x} : Rata-rata

3.8.3. *Error* (%)

Merupakan nilai persen dari simpangan(error) terhadap nilai yang di kehendaki. Dinyatakan dengan rumus :

$$\text{Error}(\%) = \frac{x - \bar{x}}{x} \times 100 \quad \dots\dots\dots (3)$$

Keterangan :

Error : besaran simpangan/nilai error dalam%

X : data x

\bar{x} : rata-rata

3.8.4. Standart deviasi

Standar deviasi adalah suatu nilai yang menunjukkan tingkat(derajat) variasi kelompok data atau ukuran standart penyimpangan dari

menanya/rata-ratanya. Jika standart deviasi semakin kecil maka data tersebut semakin presisi. Dinyatakan dengan rumus :

$$SD = \sqrt{\frac{\sum_{i=1}^n (x - \bar{x})^2}{n-1}} \dots\dots\dots (4)$$

Keterangan :

SD : standart devisiasi

X : data x

\bar{x} : rata-rata

n : banyak data

3.8.5. Ketidakpastian (UA)

Merupakan perkiraan mengenai hasil pengukuran yang di dalamnya terdapat harga yang benar. Dinyatakan dengan rumus :

$$U_a = \frac{SD}{\sqrt{n}} \dots\dots\dots (5)$$

Keterangan :

U_a : ketidakpastian

SD : Standar Devisiasi

n : banyak data

3.9. Perakitan Rangkaian Minimum Sistem & Power Supply 5 Volt

3.9.1. Alat

1. Papan PCB
2. Solder

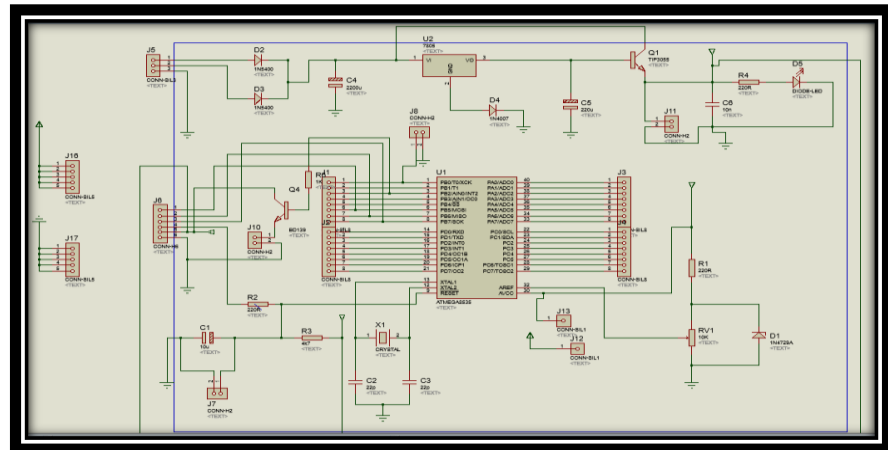
3. Penyedot Timah (*Atractor*)
4. Timah
5. Gergaji

3.9.2. Bahan

1. Dioda bridge 2 A
2. Software Proteus di Laptop
3. Travo 1 A
4. Kapasitor 2200 μ f 1 buah
5. Kapasitor non polar 104 (4)
6. *IC* regulator 7805
7. Dioda
8. Soket *IC* Mikrokontroler
9. *ATmega8535*
10. *Crystal 12.000000*
11. Multitune
12. BD139 3 buah
13. Resistor
14. Led 3ml
15. Transistor TIP 3055
16. Socket AC

3.9.3. Langkah Perakitan

1. Membuat rangkaian skematik minimum sistem dan *power supply* 5 Volt di aplikasi proteus. Gambar 3.4. menunjukkan rangkaian skematik minimum sistem dan *power supply* 5 Volt.

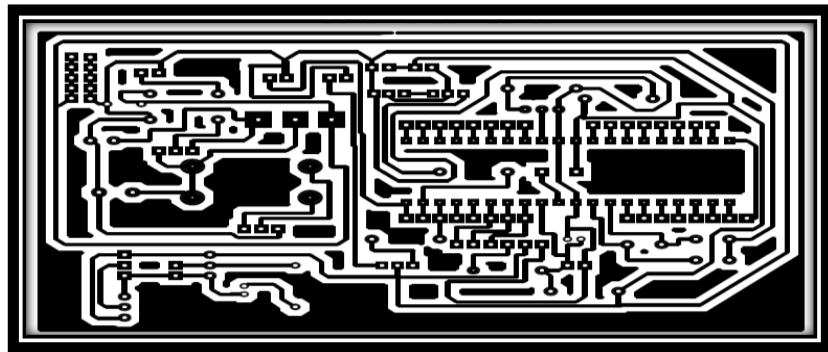


Gambar 3.4. Skematik minimum sistem dan *power supply* 5 Volt.

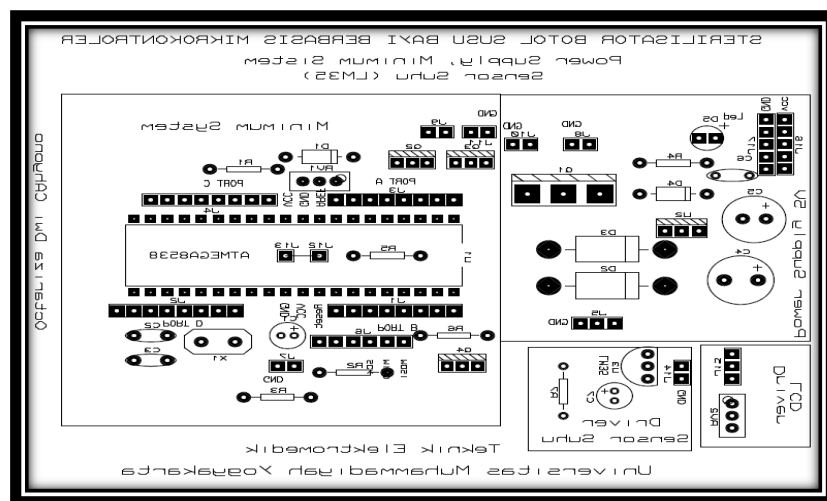
Rangkaian *power supply* pada modul ini berfungsi sebagai *supply* tegangan ke semua rangkaian yang menggunakan tegangan DC yang besarnya 5 Volt. Fungsi komponen transistor TIP305 di *power supply* yaitu sebagai penguat arus, R220 ohm sebelum led berfungsi untuk menghambat arus agar arus di led nya tidak *over heat* (putus). Prinsip kerja *power supply* adalah mengubah tegangan yang semula AC menjadi tegangan DC dengan menggunakan transformator sebagai penurun tegangan dan dioda sebagai komponen yang berfungsi sebagai penyearah tegangan. *Power supply* yang penulis buat akan mengubah tegangan AC menjadi DC sebesar 5 VDC dengan menggunakan IC regulator 7805. Adapun tegangan 5 VDC digunakan untuk menyuplay rangkaian minimum sistem dan SSR

(*solid state relay*). Sedangkan rangkaian minimum sistem pada modul ini berfungsi sebagai kontrol kerja modul secara keseluruhan dengan memanfaatkan IC *ATmega8535* yang akan memberikan logika yang akan mengaktifkan *SSR (solid state relay)* sehingga *heater* akan bekerja.

2. Setelah sistematis rangkaian jadi, tahap selanjutnya membuat *lay out* nya dan disablon ke papan *pcb*. Untuk gambar *lay out* minimum sistem dan *power supply 5 Volt* pada papan *pcb* dapat dilihat pada gambar di bawah ini:



Gambar 3.5. *Lay out* minimum sistem dan *power supply 5 Volt*

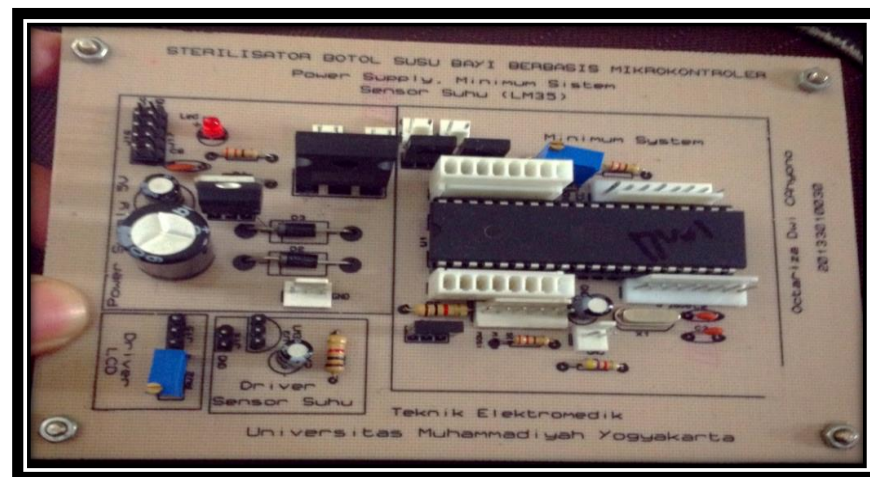


Gambar 3.6. *Lay out* tata letak komponen *minimum sistem* dan *power supply 5 volt*

3. Kemudian rakit komponen yang dibutuhkan dengan menggunakan solder dan timah.

3.9.4. Hasil dari perakitan

Gambar 3.7. dibawah ini adalah hasil perakitan *minimum sistem* dan *power supply 5 volt*.



Gambar 3.7. Hasil perakitan *minimum sistem* dan *power supply 5 volt*.

3.10. Membuat Rangkaian Sensor LM35 & LCD

3.10.1. Alat

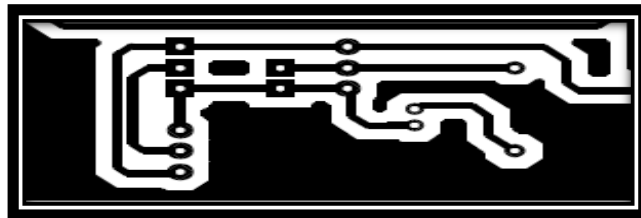
1. Papan PCB
2. Solder
3. Penyedot Timah
4. Timah

3.10.2. Bahan

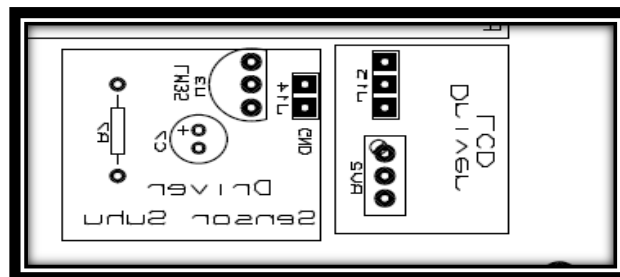
1. LCD 16 x 2
2. Mutitune

rangkaian LCD 16x2 diletakan dibagian *PORT.C* rangkaian *minimum system ATmega8535*.

2. Setelah selesai membuat skematik langkah selajutnya membuat *lay out* di *isis* (proteus) dan disablun/diCetak ke papan pcb yang telah dipotong sesuai ukuran dari gambar *lay out*. Gambar 3.9. merupakan hasil dari *lay out*nya.



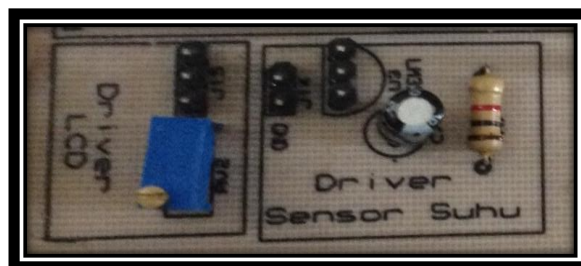
Gambar 3.9. *Lay out* belakang



Gambar 3.10. *Lay out* depan

3.10.4. Hasil dari perakitan

Gambar 3.11. dibawah ini adalah hasil perakitan rangkaian LM35 dan *driver* LCD.



Gambar 3.11. Hasil perakitan rangkaian LM35 dan *driver* LCD.

3.11. Membuat Rangkaian *Buzzer*

3.11.1. Alat

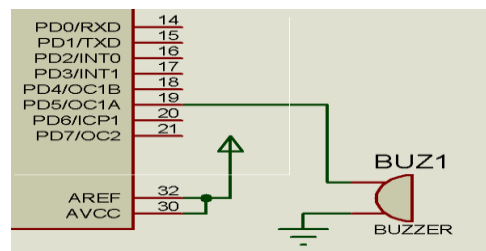
1. Solder
2. Timah
3. Penyedot timah

3.11.2. Komponen

1. *Buzzer*
2. Kabel pelangi *female* dan *male*

3.11.3. Langkah perakitan

1. Pasang kabel pelangi ke kaki *Buzzer*.
2. Hubungkan kaki *positif Buzzer* ke port D pin 5 dan kaki *negatif Buzzer* pasang ke *ground*.
3. Sesuai sistematik di bawah ini :



Gambar 3.12. Rangkaian *Buzzer*

Rangkaian *buzzer* ini di gunakan untuk memberikan peringatan apabila semua sistem yang sudah berjalan selesai sesuai *timer* yang di tentukan, *Buzzer* ini di sambungkan langsung dengan mikrokontroler dengan kaki positifnya ke *port D pin 5* dan kaki negatifnya di sambungkan ke *ground* sudah dapat bekerja dengan baik.

3.12. Membuat Rangkaian Driver Heater

3.12.1. Alat

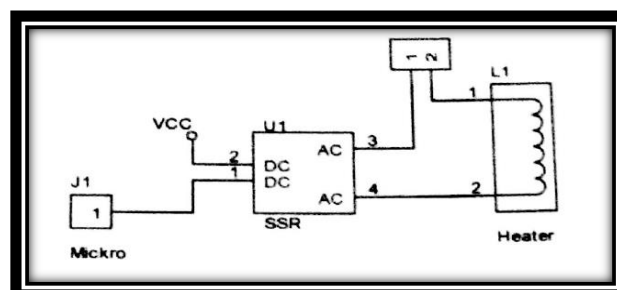
1. Obeng +
2. Tang Potong
3. Solder

3.12.2. Bahan

1. SSR (*Solid State Relay*)
2. Kabel Serabut
3. Kabel Pelangi

3.12.3. Langkah Perakitan

1. Hubungkan inputan SSR (*Solid State Relay*) kabel stecker 220 Volt, dan inputan satu nya ke *heater*.
2. Hubungkan ouput dari kaki positif SSR (*Solid State Relay*) ke ADC mikrokontroler yaitu PORT.A pin 1.
3. Hubungkan ouput dari kaki negatif SSR (*Solid State Relay*) mikrokontroler *ground*.
4. Sesuai gambar skematik dibawah ini :



Gambar 3.13. Rangkaian *Driver Heater*

Perhitungan SSR (*Solid State Relay*):

$$I_c = (V_c - 1) / 1000$$

$$I_c = (5-1) / 1000$$

$$I_c = 4 / 1000 = 0,004 \text{ A} = 4 \text{ mA}$$

Tegangan input DC pada SSR mempengaruhi arus yang masuk dari mikro. Pada rangkaian diatas, SSR (*Solid State Relay*) digunakan sebagai driver ke *heater* dengan spesifikasi tegangan input 3-32 VDC, output sebesar 220 volt, dengan konsumsi arus maksimal 10A. SSR akan bekerja ketika kaki no 2 diberikan tegangan vcc sebesar 5 volt dan kaki no 1 mendapatkan logika low/0 dari mikrokontroler. Jadi ketika VCC 5 volt bertemu dengan logika 0, maka SSR (*Solid State Relay*) akan bekerja dikarenakan tegangan 5 volt berada dirange atau spesifikasi SSR (*Solid State Relay*) yang membutuhkan inputan DC antara 3-32 VDC. SSR (*Solid State Relay*) bekerja, maka akan mengalirkan arus tegangan dari PLN 220 volt ke *heater*, sehingga *heater* dapat bekerja dengan merubah energi listrik menjadi energi panas.

3.13. Pembuatan Program Modul

Untuk pembuatan program pada modul ini menggunakan aplikasi CV AVR dengan bahasa C. Program yang digunakan ialah program *ADC* sebagai pengendali sensor LM35 dan *timer* sebagai pengontrol waktunya.

Pembuatan *list program* dengan cara manual untuk menambahkan perintah dan logika pada program ini agar berjalan sesuai yang diinginkan penulis, Berikut ini program dari modul saya :

Isi program modul

```

Chip type           : ATmega8535
Program type       : Application
AVR Core Clock frequency : 12,000000 MHz
Memory model      : Small
Data Stack size   : 128
*****/

#include <mega8535.h>
#include <stdlib.h>
#include <delay.h>
#include <alcd.h>
} Header dari program

unsigned char detik, a=0,temp[6], temp2[6], temp3[3], menit=5;
float data, suhu;
bit timer_aktif=0, b=0, c=0;
interrupt [TIM1_OVF] void timer1_ovf_isr(void)
{
TCNT1H=0x48E5 >> 8;
TCNT1L=0x48E5 & 0xff;
if(detik==0)
{menit--;detik=59;}
else{detik--;}
//}
}

```

Listing 3.1. Listing Header Program

Program yang penulis buat menggunakan mikrokontroler *ATmega8535*.

`#include <mega8535.h>` = merupakan library dari mikrokontroller yang kita gunakan ,disini penulis menggunakan *ATmega8535*.

`#include <delay.h>` = merupakan *library* fungsi *delay*.

```
void mulai_timer()
{
  if(timer_aktif==1&& suhu>=100)
  {
    TCCR1B=0x04;a=1;
  } else if(timer_aktif==0){TCCR1B=0x00;}}

void stop_timer()
{
  if(menit==0&&detik==0)
  {
    lcd_clear();
  while(1)
  {
    TCCR1B=0x00;
    timer_aktif=0;
    PORTD.1=0;
    PORTD.5=0;
    lcd_clear();
    lcd_gotoxy(6,0);
    lcd_putsf("SELESAI");
  }
}
}
```

Listing 3.2. Listing Program Timer

Listing program timer ini digunakan sebagai pengaturan timer, penulis menggunakan timer 1 (TCCR1B) timer aktif apabila mendapatkan logika 0 dari program. Settingan program timer yaitu selama 5 menit dengan metode *counter down*.

```

// pembacaan sensor suhu
void baca_suhu()
{
    lcd_gotoxy(0,0);
    lcd_putsf("SUHU:");
    data=read_ADC(0);
    suhu=(data*3.05)/1024;
    suhu=suhu*100;
    if (suhu<100)
}
void driver_set()
{
    if(timer_aktif==1){
        if(suhu>=100)
        {
            PORTD.5=0;
        }else{PORTD.5=1;}}
}

```

} Port ini digunakan untuk mengaktifka buzzer

Listing 3.3. Listing Program Sensor Suhu dan Buzzer

Listing program sensor suhu ini digunakan sebagai pengaturan pembacaan sensor suhu LM35 saat sistem bekerja ada beberapa yang diatur dalam listing program ini antara lain rumus pembacaan dan batas sensor suhu pembacaan untuk mengontrol driver *heater*. PORT.C digunakan untuk menghidupkan buzzer, buzzer akan menyala apabila diberikan logika 0.

```

// RS - PORTC Bit 0
// RD - PORTC Bit 1
// EN - PORTC Bit 2
// D4 - PORTC Bit 4
// D5 - PORTC Bit 5
// D6 - PORTC Bit 6
// D7 - PORTC Bit 7
// Characters/line: 16

lcd_init(16);

lcd_clear(); // intruksi untuk membersihkan huruf atau angka di LCD.

lcd_gotoxy(0,0); // untuk peletakkan huruf di LCD kolom (x) atau baris(y).

lcd_putsf(" STERILISASI"); // intruksi data untuk ditampilkan di LCD.

lcd_gotoxy(1,1); // lama waktu huruf/angka ditampilkan

lcd_putsf("BOTOL SUSU BAYI"); // intruksi data untuk ditampilkan di LCD.

delay_ms(2000);

lcd_clear(); // intruksi untuk membersihkan huruf atau angka di LCD.

```

Pemasangan LCD 16X2 di mikrokontroler
ATmega8535

Listing 3.4. Listing Program LCD 16x2

Listing proram diatas adalah program LCD 16x2 ke mikrokontroler *ATmega8535*.

`lcd_clear();` // intruksi untuk membersihkan huruf atau angka di LCD.

`lcd_gotoxy(0,0);` // untuk peletakkan huruf di LCD kolom (x) atau baris(y).

`lcd_putsf(" STERILISASI");` // intruksi data untuk ditampilkan di LCD.

