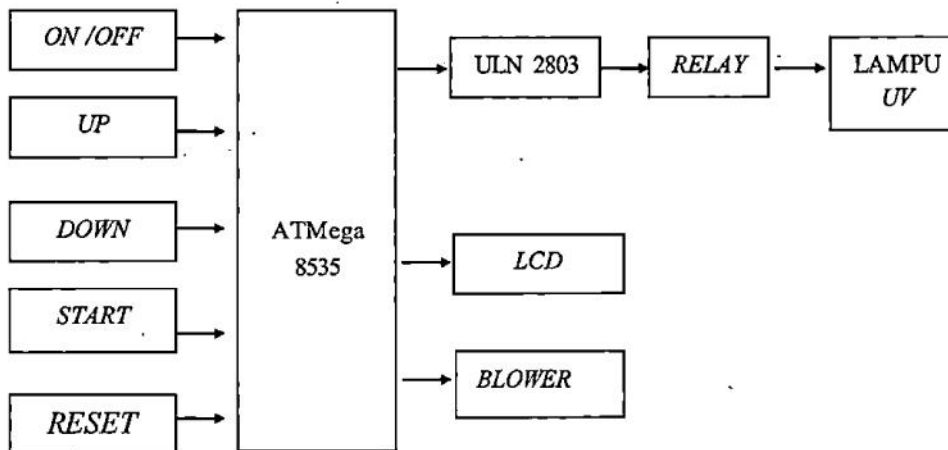


BAB III

METODE PENELITIAN

3.1 Perencanaan Blok Diagram

Pada perencanaan blok diagram pesawat *UV* Sterilisator penulis merencanakan penunjukan secara digital. Dengan penunjukan pada *display* maka diharapkan akan memudahkan dalam pembacaan beberapa waktu yang dipilih. Pada pemilihan waktu *setting* penulis juga merencanakan dengan sistem *digital*, sehingga pada saat pemilihan waktu menjadi mudah. Sebagai penjelasan dari rancang bangun rangkaian pesawat *UV* Sterilisator, maka penulis merencanakan diagram blok dari rangkaian-rangkaian yang direncanakan seperti pada gambar 3.1 :



Gambar 3.1. Blok Diagram *UV* Sterilisator

Gambar diatas menggambarkan cara kerja rangkaian secara diagram blok. Pada blok rangkaian *setting*, terdapat pengaturan waktu yang digunakan untuk memilih waktu yang akan digunakan.

Jika pengaturan waktu telah diatur maka untuk menjalankan pesawat *UV Sterilisator* ini adalah dengan menekan tombol *START*, maka *LCD* akan menyala.

Pada rangkaian *display* penulis menggunakan *LCD* sebagai tampilan. Pada rangkaian *display* ini terdapat 1 buah *LCD*

Rangkaian penggerak elemen berfungsi untuk memutus hubungan *supply* untuk *elemen* yang dikendalikan oleh *output mikrokontroler* yang akan membias *ULN2803* sebagai sakelar. Pengontrolan dari semua rangkaian keseluruhan dilakukan oleh *mikrokontroler ATmega8535* sesuai dengan *software* yang telah diprogram di dalam *mikrokontroler*.

Fungsi masing-masing blok tersebut adalah :

1. *Blok Up dan Down*

UP dan *DOWN* digunakan untuk mengatur waktu yang diperlukan. *UP* untuk menaikkan waktu dan *DOWN* untuk menurunkan waktu sesuai yang diinginkan.

2. *Blok Reset*

Reset berfungsi untuk mereset atau mengonolkan *display*.

3. *Blok START*

START berfungsi untuk memulai atau menjalankan program yang bertujuan menyalahkan lampu *UV Sterilisator*.

4. Blok Mikrokontroler AVR ATmega8535

Pada blok ini terdapat *IC* yang berfungsi sebagai otak atau pusat pengendali utama dari rangkaian keseluruhan. Mikrokontroler AVR ATmega8535 akan diisi program yang telah diinginkan untuk menjalankan pesawat UV Sterilisator. Masukan yang didapat oleh Mikrokontroler AVR ATmega8535 berasal dari tombol *up* dan *down* yang berfungsi sebagai pengatur penyinaran, yaitu penyinaran dengan menggunakan waktu. Keluaran dari mikrokontroler AVR ATmega8535 akan ditampilkan pada *display* sesuai dengan perintah *input*. Selain itu keluaran dari mikrokontroler AVR ATmega8535 berfungsi juga sebagai pengendali indikator *alarm* jika waktu penyinaran habis.

5. Blok Lampu

Lampu yang digunakan pada rangkaian ini adalah lampu UV Sterilisator 20 *Watt*. Lampu ini dapat langsung digunakan dengan menggunakan sumber 220 *VAC*.

6. Blok IC ULN 2803

Sebagai *ic driver* untuk mengaktifkan *relay*.

7. Blok LCD

Berfungsi sebagai *display* untuk menampilkan *settingan timer* pada UV.

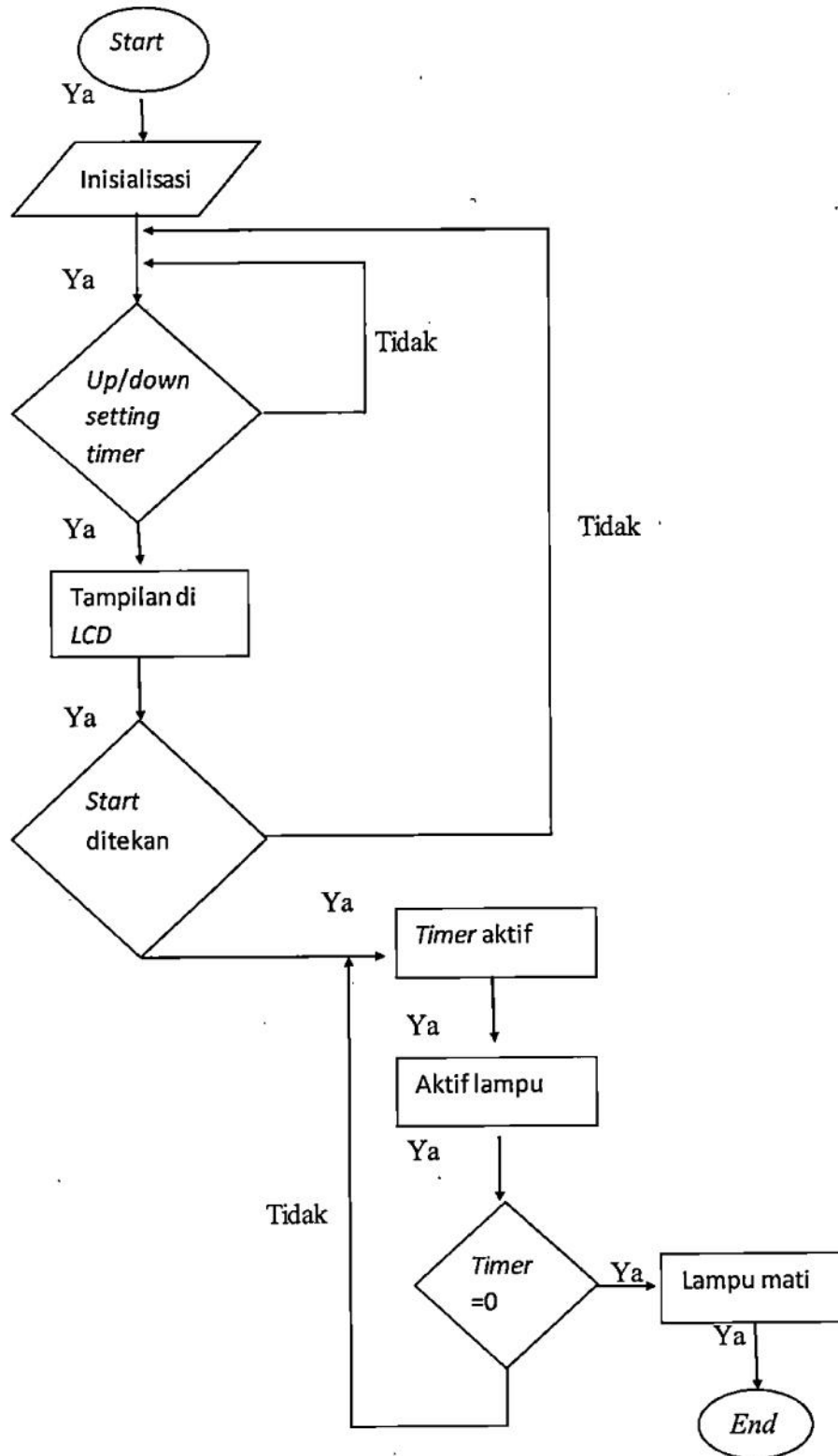
3.2. Flow Chart

Perangkat lunak yang dirancang, dibuat dengan menggunakan bahasa *bascom avr*. Algoritma program utama dari perancangan perangkat lunak dapat dilihat dalam *flow chart*, yang mana dengan kata lain disebut sebagai diagram yang menyajikan prosedur untuk menjalankan secara berurutan sesuai dengan yang kita harapkan.

Berikut adalah penjelasan cara kerja dari *flow chart*:

Pada saat menyalakan alat, pastikan alat telah mendapat input tegangan dari PLN. Tekan tombol *ON/OFF* untuk menyalakan alat, dan pastikan *LCD* telah menyala sebagai indikator. Setelah alat menyala, maka *user* dapat melakukan pengaturan waktu. Jika tombol *setting* waktu ditekan, maka setting waktu akan sesuai dengan *setting* waktu yang dipilih dan ditampilkan di *display*.

Setelah *setting* waktu diatur, *user* dapat langsung menyalakan alat dengan menekan tombol *START*. Pada saat lampu menyala, maka tampilan di *display* akan menunjukkan pengaturan waktu. Ketika waktu telah mencapai angka yang sesuai *setting* waktu, maka lampu akan padam. (mengisi halaman kosng)

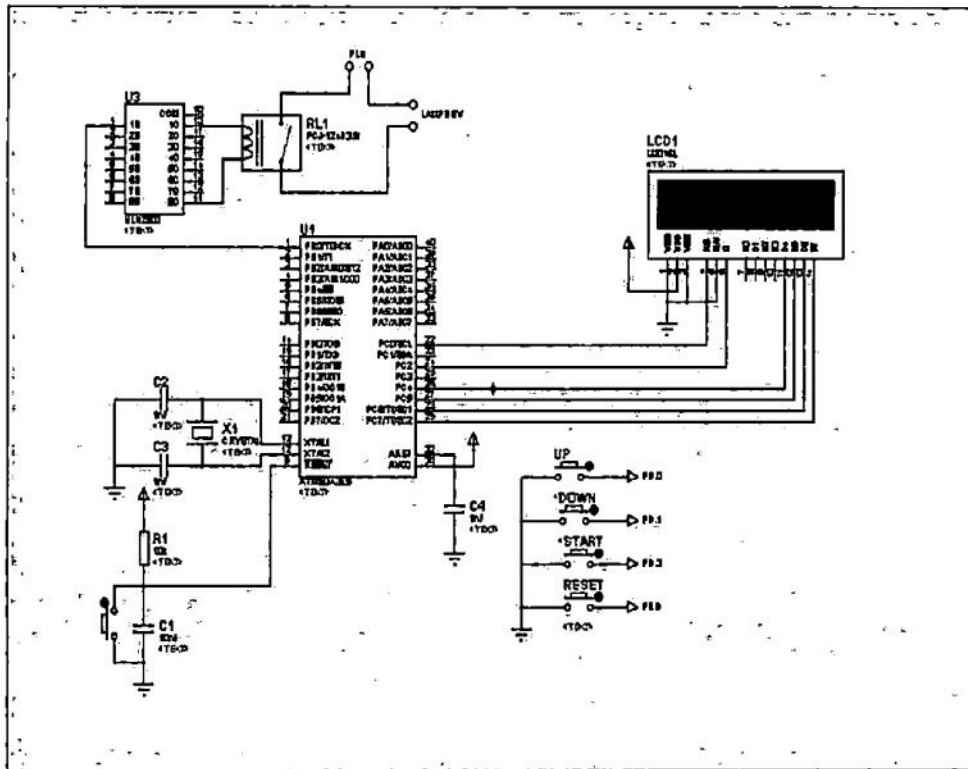


Gambar 3.2 Flow Chart UV Sterilisator

3.3. Perancangan Perangkat Keras

Perancangan perangkat keras ini dilakukan sebagai tata cara untuk menentukan program yang akan dimasukkan kedalam mikrokontroler yang berfungsi sebagai pengontrol perangkat keras ini.

Perancangan perangkat keras secara keseluruhan dapat digambarkan pada gambar 3.3. Rangkaian keseluruhan terdiri dari rangkaian catu daya, rangkaian sistem minimum *mikrokontroler*, rangkaian tombol, dan rangkaian *driver lcd*.



Gambar 3.3. Rangkaian Keseluruhan

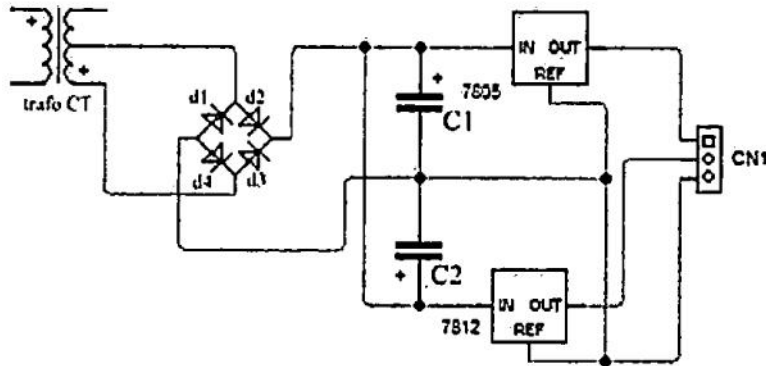
3.3.1. Rangkaian Catu Daya

Power Supply merupakan sumber tenaga listrik bagi semua komponen di dalam sebuah alat, semakin besar daya *power supply*,

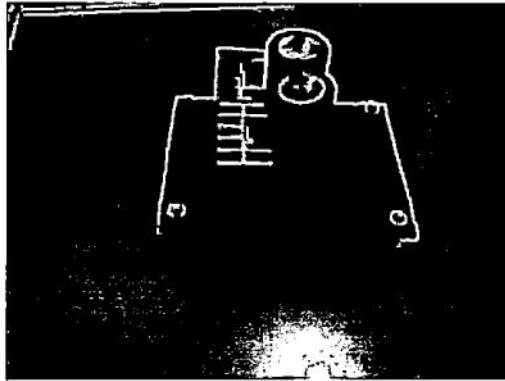
maka kestabilan alat akan semakin terjamin. Besarnya tegangan yang disuplai oleh *power supply* antara lain 3,3 V, 5 V, 12 V, -5 V, dan -12 V. Tegangan 3,3 V mempunyai arus maksimum 12 A sedangkan tegangan 5 V mempunyai arus maksimum 20 A. [5]

Pada rangkaian catu daya kali ini menggunakan trafo CT. Trafo dapat digunakan untuk transformasi impedansi. Impedansi masukan tidak sama dengan impedansi keluaran jika jumlah lilitan primer berbeda dengan jumlah lilitan sekunder. Transformator adalah komponen pasif yang dapat bekerja tanpa memerlukan daya listrik dari luar. Ini berarti daya keluaran tak akan lebih besar daripada daya masukan.[6]

Catu daya berfungsi untuk memberikan *suplay* tegangan, khususnya ke IC mikrokontroler ATmega8535. Karena menggunakan trafo CT, dimana *output* tegangan adalah 12 volt, maka untuk merubah tegangan menjadi 5 volt digunakan IC LM7805, dan untuk menstabilkan tegangan di 12 volt menggunakan IC LM7812. Pada rangkaian catu daya, dioda *bridge* digunakan untuk menyearahkan gelombang penuh dari AC ke DC. Sedangkan kapasitor berfungsi sebagai *filter* tegangan yang dihasilkan oleh dioda *bridge*. Skema rangkaian catu daya diperlihatkan pada gambar 3.4.



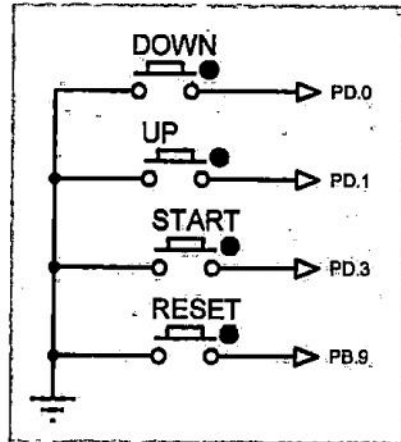
Gambar 3.4. Rangkaian Catu Daya



Gambar 3.5. Perangkat Keras Catu Daya

3.3.2. Rangkaian Tombol

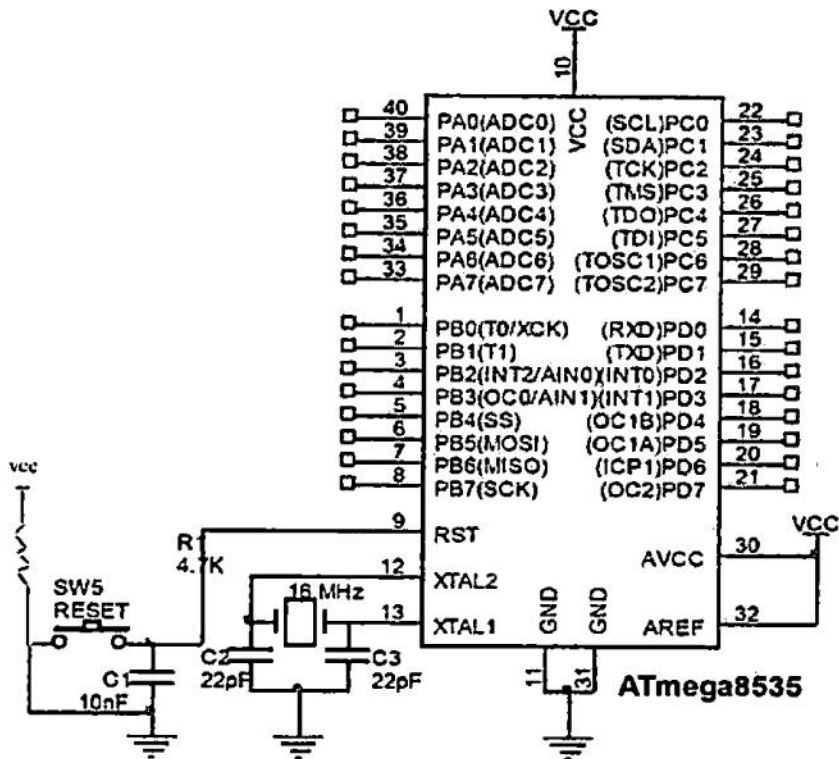
Rangkaian tombol merupakan sarana *input* bagi mikrokontroler. Sarana *input* tersebut berupa sinyal rendah untuk pengisian maupun pemilihan data. Seperti pada gambar 3.5, adanya masukan sinyal rendah ke kaki mikrokontroler ketika salah satu tombol ditekan sehingga terhubung ke *ground*. Jika salah satu tombol tidak ditekan, maka kaki mikrokontroler dalam keadaan sinyal tinggi karena memiliki *internal pull up resistance*.



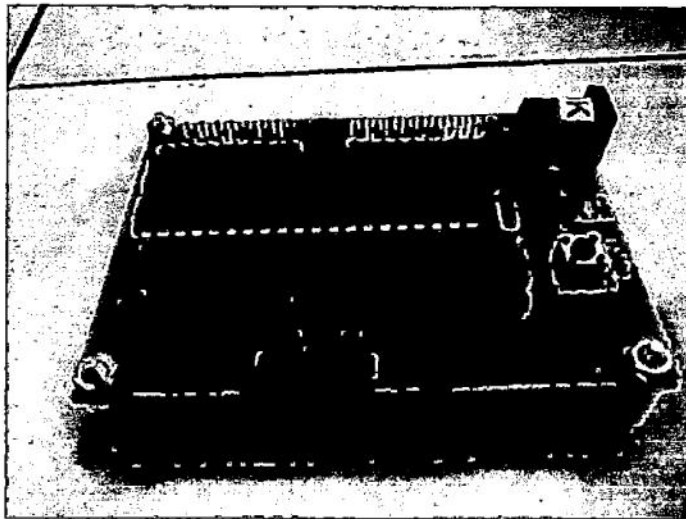
Gambar 3.6. Rangkaian Tombol

3.3.3. Rangkaian Sistem Minimum Mikrokontroler

Rangkaian sistem minimum mikrokontroler ini adalah rangkaian utama yang digunakan untuk memfungsikan mikrokontroler, dimana merupakan pengontrol utama dalam *UV* Sterilisator. Rangkaian ini difungsikan juga sebagai salah satu media untuk melakukan *programming*. Dalam rangkaian ini, menggunakan kristal 16MHz sebagai pembangkit sinyal *eksternal*. Skema rangkaian sistem minimum mikrokontroler dapat dilihat pada gambar 3.6.



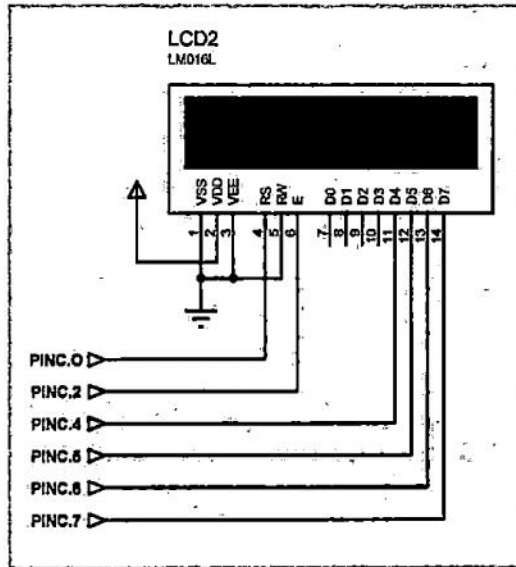
Gambar 3.7. Sistem Minimum Mikrokontroler



Gambar 3.8. Perangkat Keras Minimum Sistem

3.3.4. Rangkaian *Driver LCD*

Rangkaian *driver LCD* ini digunakan sebagai media penampil dari mikrokontroler ke *LCD*. Dalam rangkaian ini menggunakan jalur data 4 bit sehingga D0-D3 tidak terhubung. Gambar rangkaian *driver LCD* diperlihatkan pada gambar 3.7.



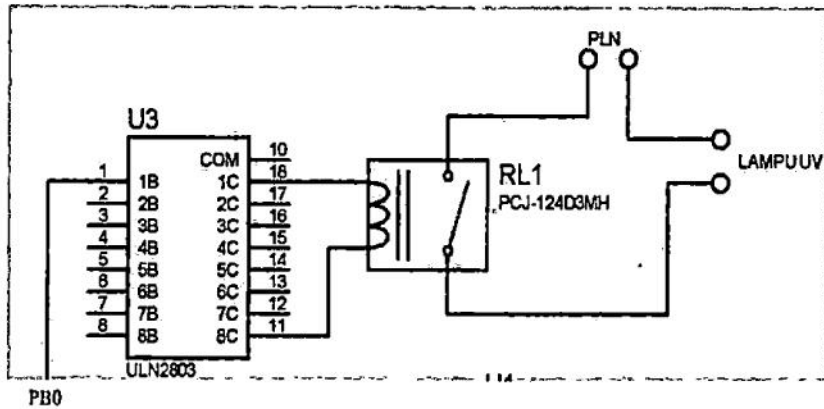
Gambar 3.9. Rangkaian *LCD*

Tabel 3.1. Fungsi Pin pada *LCD* Karakter 16x2

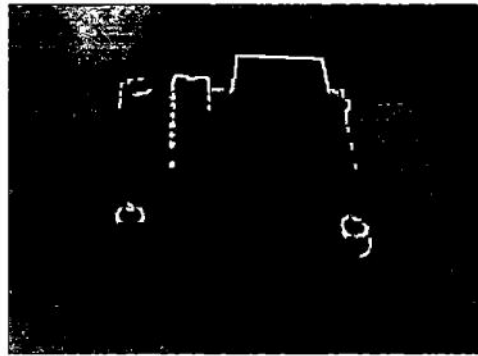
NO PIN	NAMA	KETERANGAN
1	<i>GND</i>	<i>Ground</i>
2	<i>VCC</i>	+ 5v
3	<i>VEE</i>	Kontras <i>LCD</i>
4	<i>RS</i>	<i>Control</i>
5	<i>RW</i>	<i>Control</i>
6	<i>E</i>	<i>Control</i>
7-14	DB0-DB7	Data bit 1-7
15	A	Anoda (<i>back light</i>)
16	K	Katoda (<i>back light</i>)

3.3.5. Rangkaian Kendali Lampu

Rangkaian kendali lampu berfungsi sebagai penghubung sinyal dari mikrokontroler dengan arus AC. Dalam rangkaian ini terdapat IC ULN 2803 dan relay 12 volt DC sebagai sakelar.



Gambar 3.10. Rangkaian Kendali Lampu



Gambar 3.11. Perangkat Keras Rangkaian Kendali Lampu

3.4. Perancangan Perangkat Lunak

Perangkat lunak merupakan suatu bagian yang tidak dapat dipisahkan dari suatu mikrokontroler. Perangkat keras yang sudah tertata dengan benar, tidak akan berfungsi dengan baik jika terdapat kesalahan pada perangkat

lunak. Oleh karena itu, dibutuhkan ketelitian dan ketepatan dalam penyusunan perangkat lunak yang akan diprogram ke mikrokontroler.

3.4.1. Program Pendukung

Pemrograman perangkat lunak pendukung ATmega8535 dilakukan dengan menulis *source code* program pada aplikasi *bascom avr*. *Source code* program yang sudah ditulis lalu disimpan dan di *compile* sehingga ber-*ekstensi hex*. Kemudian di *download* ke dalam mikrokontroler menggunakan program *progisp*.

3.4.2. Listing Program

Berikut adalah *list program* yang digunakan pada alat *UV Sterilisator*

:

```
'setting mikrokontroler
$regfile = "m8535.dat"
$crystal = 16000000
'inisialisasi lcd
ConfigLcdpin = Pin , Rs = Portc.0 , E = Portc.2 , Db4 = Portc.4
ConfigLcdpin = Pin , Db5 = Portc.5 , Db6 = Portc.6 , Db7 = Portc.7
ConfigLcd = 16 * 2
Cursor On
Cls
Locate 1 , 2
Lcd "UV STERILISATOR"
'inisialisasi data
Dim Waktu As Word , Menit As Byte , Detik As Byte , Jam As Byte
```

```
Ddrd.1 = 0
Portd.1 = 1
Ddrd.2 = 0
Portd.2 = 1
Ddrd.3 = 0
Portd.3 = 1
Config Portb = Output
Portb = 0
Config Int0 = Low Level
Config Timer1 = Timer, Prescale = 1024
On Ovf1 Count
Enable Interrupts
Enable Int1
Timer1 = 49910
Do
  'naik
  If Pind.1 = 0 Then
    Menit = Menit + 15
    Waitms 200
  End If
  If Menit = 240 Then
    Menit = Menit - 1
  End If
  'turun
  If Pind.3 = 0 Then
```

```
Menit= Menit- 15
Waitms 200
End If
If Menit = 241 Then
Menit= Menit * 0
End If
Waitms 200
'mulai
If Pind.2= 0 Then
Enable Ovf1
Timer1= 49910
Start Timer1
End If
'Penempatan variable waktu
Locate 2 , 1
Lcd Menit ; "mnt :"; Detik; "sc "
Loop
End
Count:
Portb.1 = 1
Timer1= 49910
Start Timer1
If Detik= 0 Then
If Menit = 0 Then
Else
```

```
Decr Menit
Detik = 59
End If
Else
Decr Detik
End If
If Menit = 0 And Detik = 0 Then
Disable Ovf1
Timer1= 49910
Stop Timer1
Portb.2 = 1
Portb.1 = 0
End If
Timer1= 49910
Start Timer1
Return
```