

BAB 3

PERANCANGAN DAN PEMBUATAN SISTEM

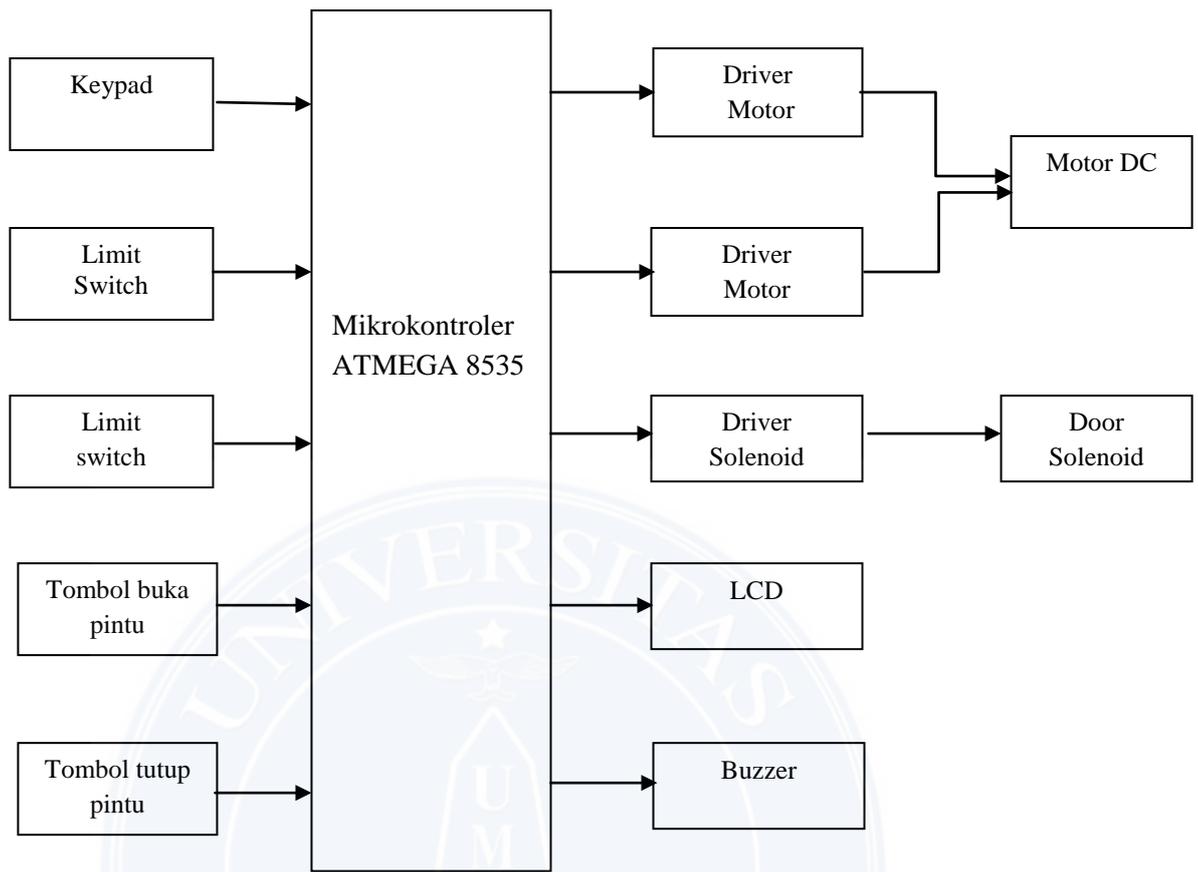
3.1. Spesifikasi Sistem

Sebelum merancang blok diagram dan rangkaian terlebih dahulu membuat spesifikasi awal rangkaian untuk mempermudah proses pembacaan, spesifikasi sistem dari rangkaian ini adalah sebagai berikut :

- Sumber Tegangan : 220 Volt AC (*Alternating Current*)
- Power Supply : 12 Volt DC (*Direct Current*)
- Software : Bascom AVR
- Mikrokontroler : ATMEGA8535
- Input : - Keypad 3x4
- 2 Buah Limit Switch
- 2 Buah Push Button NO
- Output : - Motor DC
- Door Solenoid
- LCD
- Buzzer
- Dimensi Miniatur Pintu : P : 30 cm
L : 25 cm
T : 26 cm

3.2. Blok Diagram Sistem

Sebelum merancang suatu sistem, terlebih dahulu membuat blok diagram. Blok diagram merupakan salah satu cara sederhana untuk menjelaskan cara kerja dari suatu sistem. Dengan adanya blok diagram, kita dapat menganalisa cara kerja rangkaian dan dapat merancang *hardware* yang akan dibuat. Blok diagram sistem ditunjukkan pada Gambar 3.1.



Gambar 3.1. Blok Diagram Sistem

3.3. Fungsi Masing-Masing Blok

Fungsi dari masing – masing blok adalah sebagai berikut:

- **Blok Keypad**

Blok ini berfungsi sebagai media input mikrokontroler untuk *security password* untuk membuka dan menutup pintu.

- **Blok Limit Switch**

Terdapat dua buah blok limit switch. 1 buah limit switch berfungsi untuk memberikan informasi kepada mikrokontroler bahwa pintu sudah terbuka, dan limit switch yang lainnya berfungsi untuk memberikan informasi kepada mikrokontroler bahwa pintu sudah tertutup.

- **Blok Tombol Buka Pintu (*Push Button NO*)**

Blok ini berfungsi apabila tombol di tekan maka akan memberikan masukan ke mikrokontroler yang kemudian mikrokontroler akan membuka pintu.
- **Blok Tombol Tutup Pintu (*Push Button NO*)**

Blok ini berfungsi apabila tombol di tekan maka akan memberikan masukan ke mikrokontroler yang kemudian mikrokontroler akan menutup pintu.
- **Blok Mikrokontroler ATMEGA8535**

Blok ini berfungsi sebagai pengontrol atau pengendali semua kerja rangkaian sehingga sistem ini dapat menerima *password*, membunyikan buzzer hingga proses pembukaan dan penutupan pintu.
- **Blok Driver Motor**

Blok Driver motor DC yang dipakai ada 2, karena driver motor DC pertama berfungsi untuk membuka pintu dan motor DC yang lain untuk menutup pintu.
- **Blok Motor DC**

Blok ini berfungsi sebagai penggerak pintu pada saat pintu terbuka dan tertutup secara otomatis yang menggunakan 2 driver motor DC (relay).
- **Blok Driver Solenoid**

Blok ini berfungsi untuk mengendalikan saklar relay sehingga menghidupkan dan mematikan door solenoid.
- **Blok Door Solenoid**

Blok ini berfungsi sebagai pengunci pada pintu. Door solenoid ini akan aktif jika diberikan tegangan.
- **Blok LCD**

Blok ini berfungsi untuk menampilkan data yang telah di proses oleh mikrokontroler.
- **Blok Buzzer**

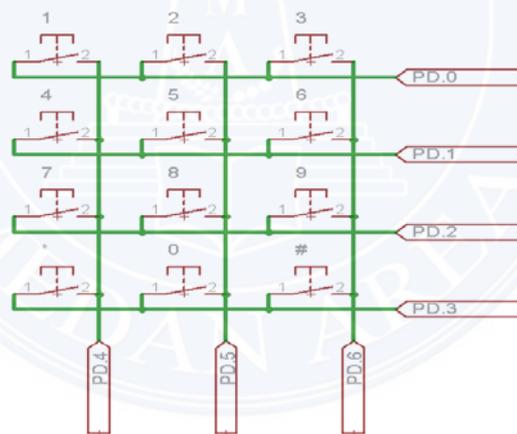
Blok ini berfungsi memberi sinyal suara apabila terjadi kesalahan dalam memasukkan *password*.

3.4. Perancangan Hardware

Bagian *hardware* terdiri atas beberapa bagian, yaitu rangkaian sistem Mikrokontroler atmega8535, rangkaian keypad, rangkaian limit switch, rangkaian tombol buka dan tutup pintu (*Push Button NO*), rangkaian driver motor, rangkaian motor DC, rangkaian driver solenoid, rangkaian door solenoid, rangkaian LCD dan rangkaian *buzzer*.

3.4.1. Rangkaian Keypad

Rangkaian keypad merupakan bagian yang digunakan untuk mengakses pembukaan dan penutupan pintu dengan cara memasukkan *password* yang ditentukan. Rangkaian keypad ini adalah keypad 3x4 artinya bahwa keypad tersebut terdiri dari 3 kolom dan 4 baris. Dalam perancangan keypad ini komponen yang digunakan adalah *switch push button*. Keypad ini merupakan rangkaian *push on*.

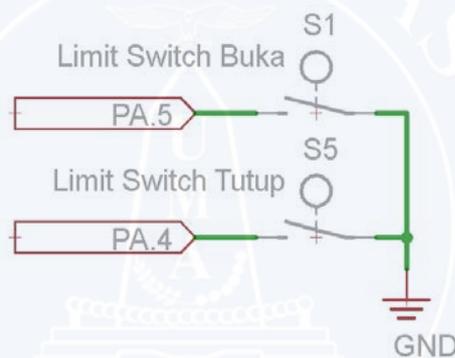


Gambar 3.2 Rangkaian Keypad 3x4

Pada rangkaian keypad 3x4 dirancang dengan menggunakan cara *scanning*, dimana setiap kolom pada salah satu pin masing-masing *switch* akan dihubungkan. Sedangkan pada bagian baris pada pin yang satunya juga akan dihubungkan. Adapun keuntungan dari konsep *scanning* adalah akan menghemat jumlah pin yang dibutuhkan untuk dihubungkan ke *port* mikrokontroler.

3.4.2. Rangkaian Limit Switch

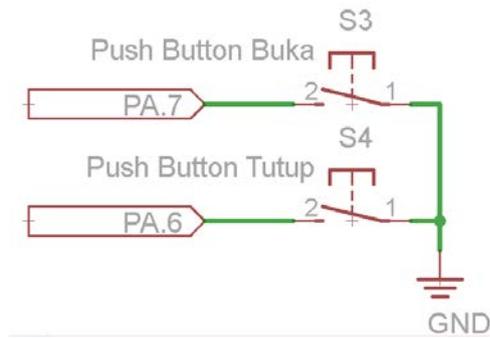
Saklar batas atau *limit switch* adalah saklar yang dapat dioperasikan baik secara otomatis maupun non otomatis. *Limit switch* yang bekerja secara otomatis adalah limit switch yang mempertahankan kontak, sedangkan *limit switch* yang bekerja non otomatis adalah *limit switch* yang tidak mempertahankan kontak. Kontak pada *limit switch* sama seperti kontak-kontak yang terdapat pada tombol tekan, yaitu mempunyai kontak *Normally Open* (NO) dan kontak *Normally Close* (NC). Skematik rangkaian dan bentuk fisik *limit switch* dapat dilihat pada Gambar 3.3.



Gambar 3.3 Rangkaian Limit Switch

3.4.3. Rangkaian Push Button

Pada penggunaannya di dalam rangkaian kontrol, tombol tekan yang digunakan adalah tombol tekan NO (*Normal Open*). Tombol tekan NO merupakan peralatan listrik yang digunakan untuk menghubungkan sesaat, yaitu pada saat tombolnya ditekan.

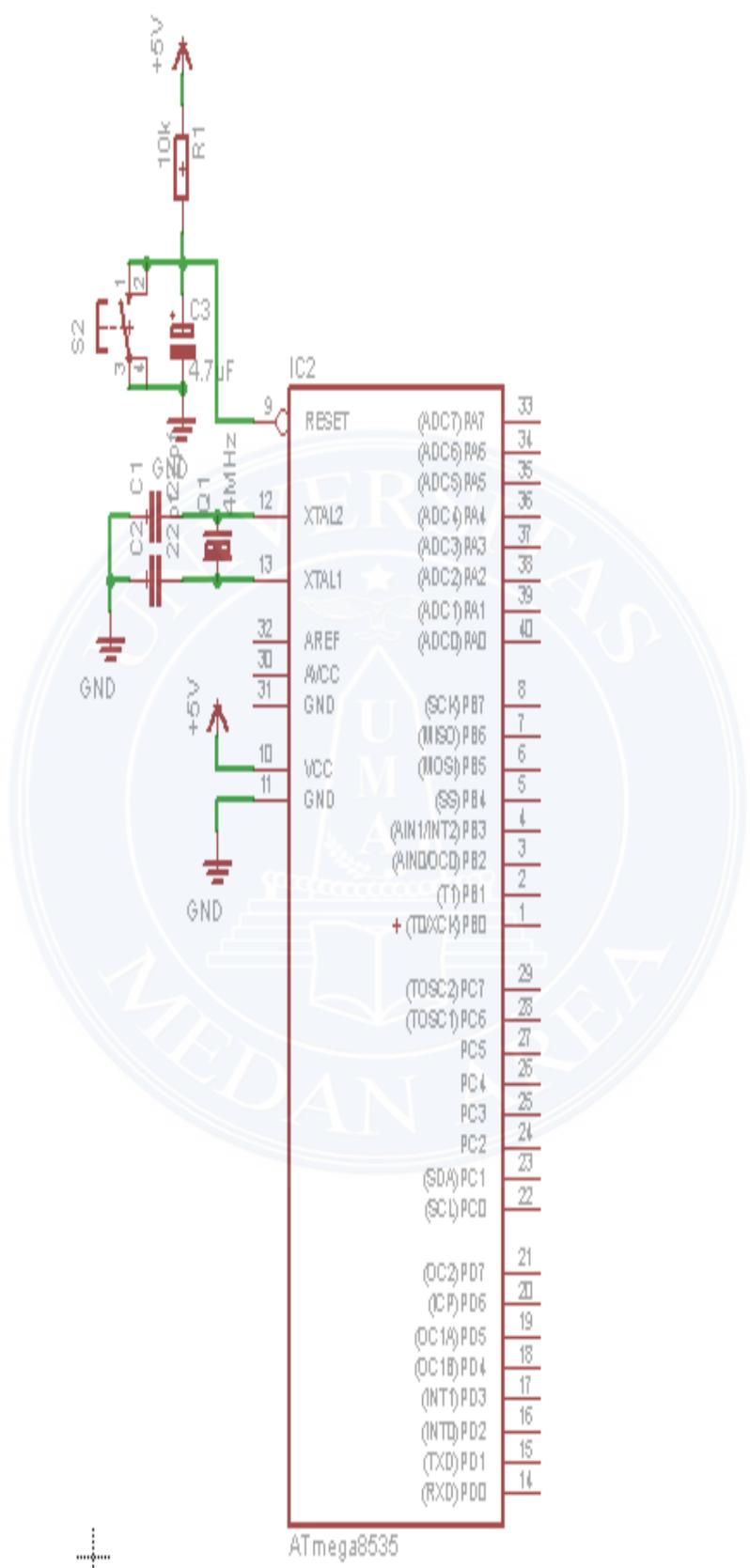


Gambar 3.4 Rangkaian Push Button

3.4.4. Rangkaian Sistem Mikrokontroler ATMEGA8535

Rangkaian ini berfungsi sebagai pengendali utama keseluruhan sistem yang ada. Komponen utama pada rangkaian ini adalah mikrokontroler atmega8535. Mikrokontroler merupakan bagian yang berfungsi untuk mendeteksi sensor, untuk menerima perintah dari *user*, serta mengendalikan beban. Mikrokontroler yang digunakan adalah jenis AVR dengan tipe atmega8535. Pada IC mikrokontroler ini telah diisi program untuk menjalankan sistem.

Mikrokontroler ini memiliki beberapa port *input-output*. Perangkat ini diprogram untuk mengendalikan beban berdasarkan perintah yang diberikan melalui komputer. Pada proses pengendalian sistem dilengkapi *feedback* dari masing-masing komponen yang dikendalikan. Dengan demikian mikrokontroler mendeteksi masukan dari sensor dan mengirimkan hasil deteksinya tersebut ke komputer. Skematik rangkaian mikrokontroler Atmega8535 dengan hubungan pin-pinnya untuk pintu dapat dilihat pada Gambar 3.5.

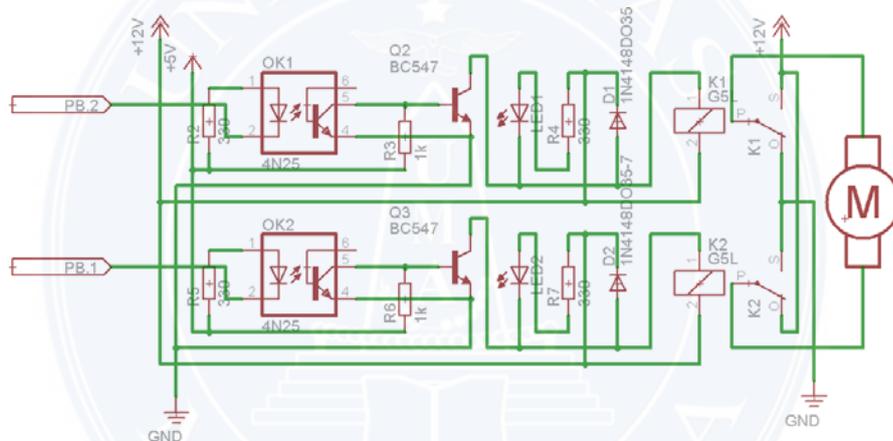


Gambar 3.5 Rangkaian Mikrokontroler Atmega8535

3.4.5. Rangkaian Driver Motor

Untuk mengendalikan arah putaran motor dc membuka dan menutup, digunakan dua buah rangkaian driver motor DC. Rangkaian driver motor DC ini terdiri dari optocoupler 4N25, transistor BC547, diode 1N4148, resistor dan relay. Pemilihan komponen ini berdasarkan konsumsi arus pada motor DC yang digunakan, karena relay masih sanggup untuk mengendalikan motor DC.

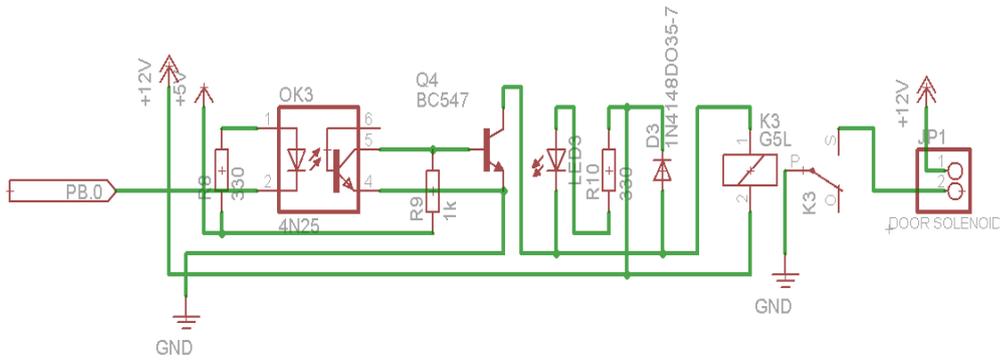
Seluruh pengontrolan pada rangkaian driver motor DC dikendalikan oleh mikrokontroler. Untuk membuka pintu, maka mikrokontroler harus memberikan logika *high* pada PB.2 dan PB.1, sedangkan untuk menutup pintu diberikan logika *low* pada PB.2 dan PB.1. Hubungan antara kendali *input* dengan *output* dapat dilihat pada Gambar 3.6.



Gambar 3.6 Rangkaian Driver Motor

3.4.6. Rangkaian Driver Solenoid

Untuk menghidupkan dan mematikan solenoid, digunakan 1 buah rangkaian relay. Rangkaian ini terdiri dari beberapa resistor, 1 buah optocoupler 4N25, 1 buah transistor BC547, dioda 1N4148 dan 1buah relay. Untuk menghidupkan rangkaian ini maka mikrokontroler memberikan logika *high* pada PB.0 dan untuk mematikannya diberikan logika *low* pada PB.0. Gambar rangkaian driver solenoid dapat dilihat pada Gambar 3.7

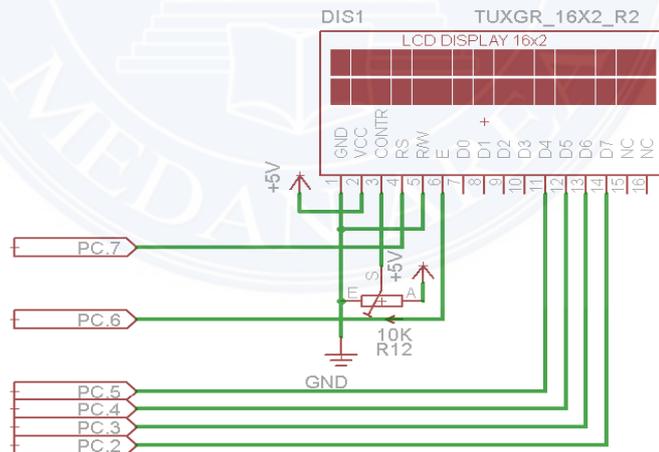


Gambar 3.7 Rangkaian Driver Solenoid

3.4.7. Rangkaian LCD

LCD merupakan penampil karakter elektronik, kapasitas karakter yang bisa ditampung oleh LCD tergantung pada spesifikasinya. Pada sistem LCD ini yang digunakan adalah LCD 2x16. LCD ini terdiri atas dua bagian. Pertama merupakan panel LCD sebagai media penampil informasi dalam bentuk huruf atau angka dua baris, masing-masing baris bisa menampung 16 huruf atau angka. Bagian yang kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroler yang ditempelkan dibagian belakang panel LCD, berfungsi mengatur tampilan formasi serta berfungsi mengatur komunikasi dengan mikrokontroler.

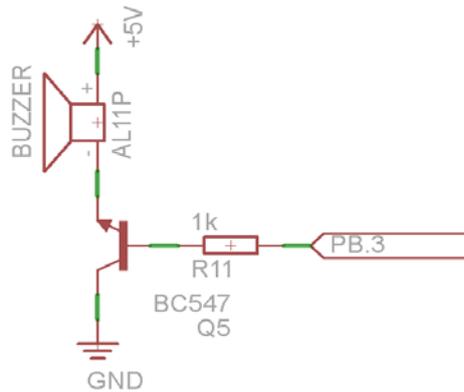
Skematik rangkaian LCD dapat dilihat pada Gambar 3.8.



Gambar 3.8 Rangkaian LCD

3.4.8. Rangkaian Buzzer

Rangkaian Driver Buzzer seperti pada Gambar 3.9 berfungsi untuk mengaktifkan suara buzzer atau bila *password* yang ditekan tidak sesuai.



Gambar 3.9 Rangkaian Buzzer

3.5 Software Pendukung

Untuk merancang program dan menulis data *hex* pada memori *flash* mikrokontroler digunakan tiga *software* utama, yaitu BASCOM AVR, BASCOM 8051 IDE dan *Universal Serial Bus In System Programing Downloader*.

3.5.1. Bascom AVR

BASCOM AVR merupakan program *compiler* berbasis *Windows* yang digunakan untuk mikrokontroler keluarga AVR, seperti ATmega8535, ATmega16, ATmega8, dll. Bahasa pemograman ini tidak jauh berbeda dengan bahasa pemograman *Visual Basic* yang sering diaplikasikan pada PC (*Personal Computer*). Jendela BASCOM AVR dapat dilihat pada gambar 3.10



Gambar. 3.10 Tampilan Jendela BASCOM AVR

3.5.2 USB Downloader

Universal Serial Bus In-System Programming (USB ISP) Downloader merupakan program *downloader* yang dapat digunakan untuk men-*download* data *hex* pada memori *flash* mikrokontroler. Mikrokontroler yang didukung cukup banyak, seperti keluarga AVR dan MCS-51.

Untuk menulis data *hex* pada memori *flash* mikrokontroler digunakan sebuah perangkat keras sebagai media lanjutan dari perangkat lunak USB ISP *Downloader*. Perangkat keras dan perangkat lunak USB ISP *Downloader* ini dapat dilihat pada Gambar 3.11.



Gambar 3.11 Hardware USB ISP Downloader

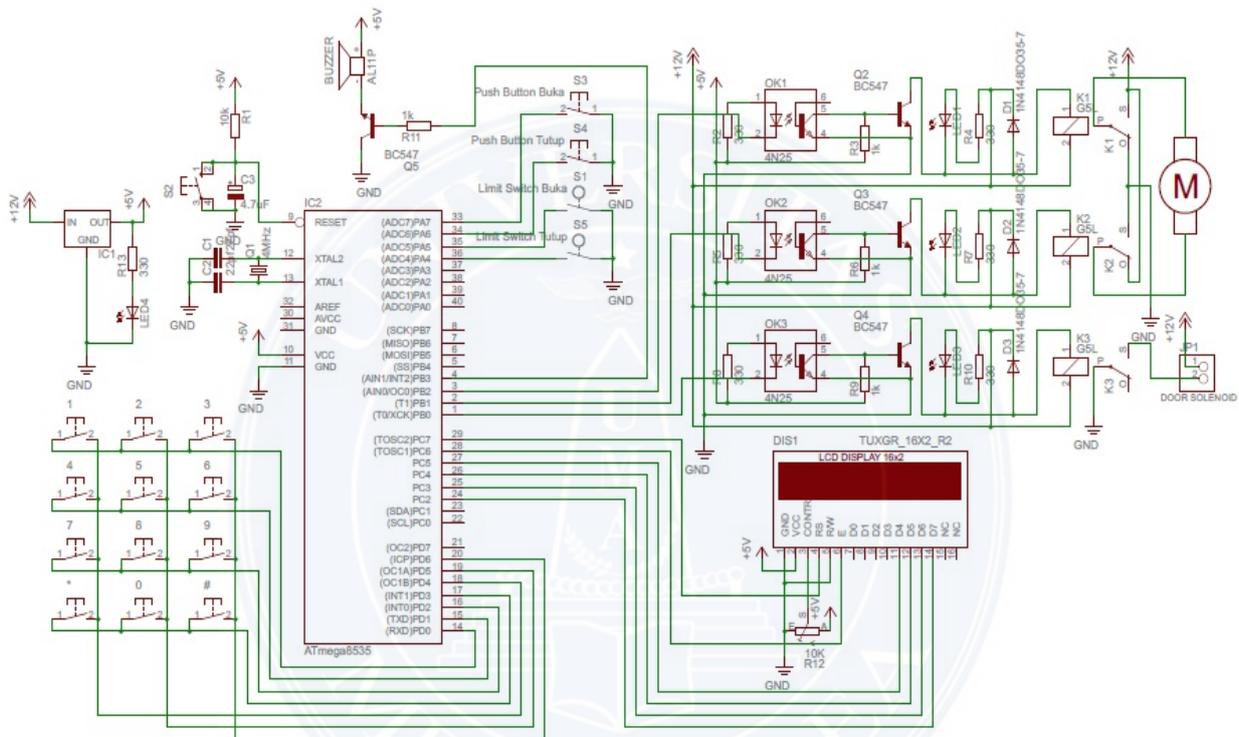
3.6 Perancangan PCB

Perancangan PCB (*Printed Circuit Board*) dilakukan bersama dengan perancangan tata letak komponen. Proses ini sangat erat kaitannya dengan pola PCB. Perancangan PCB menggunakan *software* EAGLE 5.11.0.

Software ini merupakan *software* berbasis *windows* yang difungsikan untuk merancang PCB dan menggambar skematik rangkaian. EAGLE 5.11.0 mendukung pembuatan PCB *double-layer* maupun *single-layer*. Dengan menggunakan EAGLE 5.11.0, *user* diberi kemudahan dalam membentuk pola PCB dan mengatur tata letak komponen, karena EAGLE5.11.0 sudah memberikan *data base* cukup lengkap pada *library*-nya. Dalam merancang tata letak komponen dan pembuatan jalur, perlu diperhatikan hal-hal sebagai berikut :

- Letakkan komponen yang rapi dan simetris sehingga pengawatan menjadi pendek dan ukuran PCB semakin kecil.
- Menghindari sudut atau belokan yang tajam, agar jalur tidak mudah mengelupas.

Proses pertama dalam membuat sebuah PCB adalah menggambar skema rangkaian. Proses ini membutuhkan ketelitian dalam menghubungkan kaki-kaki komponen.



Gambar 3.12 Penggambaran Skema Rangkaian

3.6.1 Pembuatan Layout

Dalam membuat *layout* PCB hal yang perlu diperhatikan adalah kerapian dan keseragaman jalur-jalur. Tata letak komponen juga merupakan pertimbangan dalam membuat *layout* PCB.

3.6.2 Pembuatan Sistem

Pada pembuatan *hardware* dilakukan beberapa tahap yaitu mulai dari pembuatan *layout* PCB sampai dengan pemasangan komponen dari penyolderan.

Pembuatan harus dilakukan secepat mungkin guna menghindari kegagalan yang ditimbulkannya.

3.6.3 Pembuatan PCB

Pembuatan PCB dapat dilakukan dengan dua cara dasar, yaitu dengan *Direct Etching* dan *Indirect Etching* (teknik penyablonan). Dengan *Direct Etching* pola *layout* digambar langsung pada PCB dengan menggunakan spidol *permanent* dan selanjutnya dilarutkan dengan FeCl₃. Pada *Indirect Etching* terdapat dua cara, yaitu dengan teknik penyablonan dan penggosokan. Dalam pembuatan PCB untuk menghindari kerumitan penggambaran *layout* langsung pada PCB. Pada pembuatan alat ini direncanakan menggunakan *Indirect Etching* (teknik penyablonan).

3.6.4 Film Layout PCB

Film *layout* PCB memegang peranan penting karena akan dijadikan film yang akan dicetak di PCB. Perancangan *layout* PCB didasarkan pada beberapa pertimbangan yang menyangkut keamanan dan efisiensi PCB yang digunakan. Untuk *layout* PCB dari rangkaian model mikrokontroler ini beberapa rangkaian akan dirancang terpisah untuk menghindari kerusakan dan mempermudah menganalisa kerusakannya. *Layout* PCB yang telah siap untuk dicetak ke PCB dapat dilihat pada Gambar 3.14, 3.15, 3.16, dan 3.17

Gambar 3.14 layout Mikrokontroler

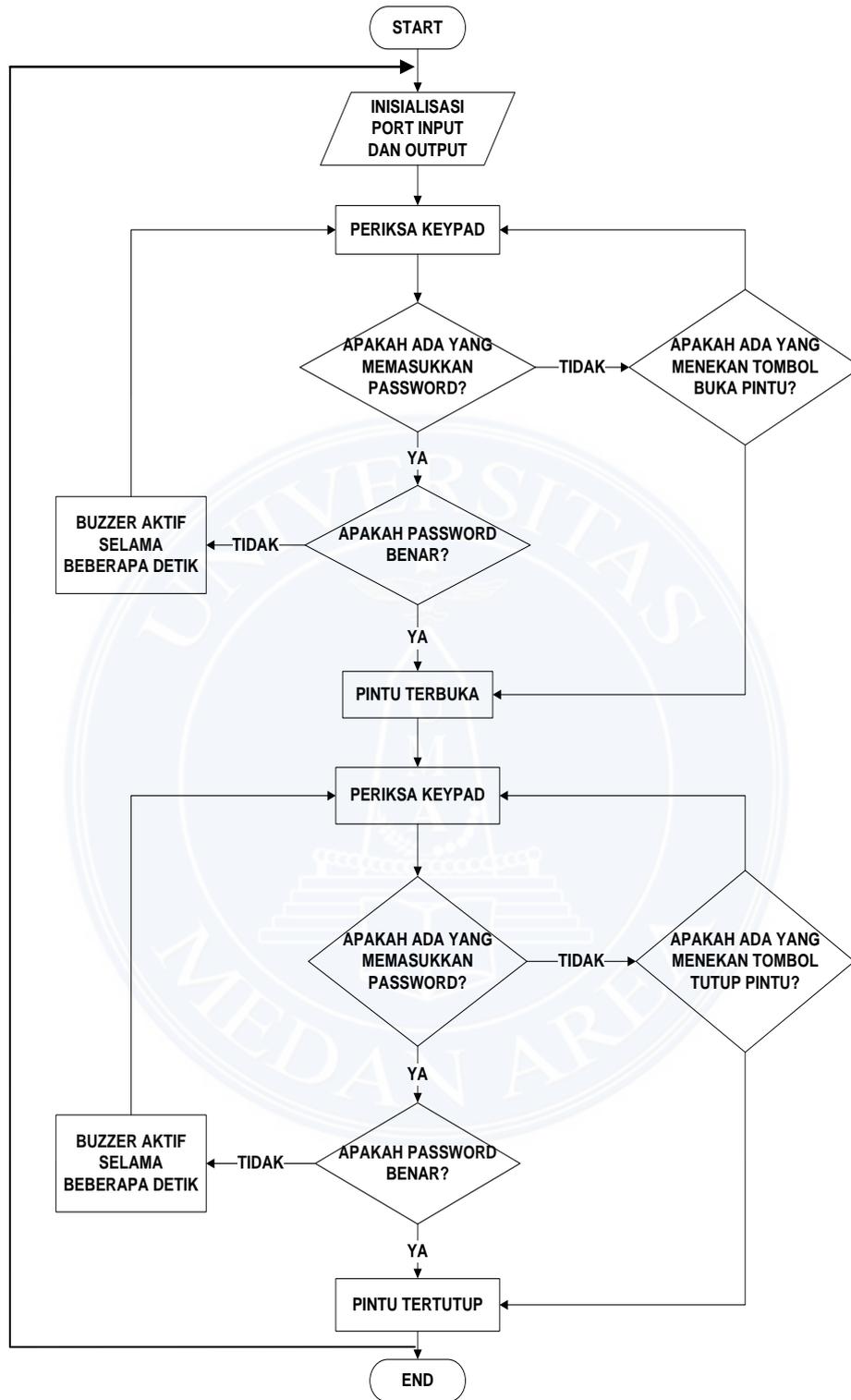
Gambar 3.15 layout Motor DC

Gambar 3.16 layout rangkaian relay

Gambar 3.17 layout Buzzer

3.6.5 Perancangan Alogaritma Program

Sebelum merancang *software* program, terlebih dahulu dirancang flowchart. Dengan adanya flowchart akan memudahkan kita untuk merancang bahasa programnya. Flowchart merupakan gambaran tentang proses-proses yang terjadi pada program. Flowchart dari sistem ini ditunjukkan pada Gambar 3.18.



Gambar 3.18 Flowchart Pemrograman Sistem