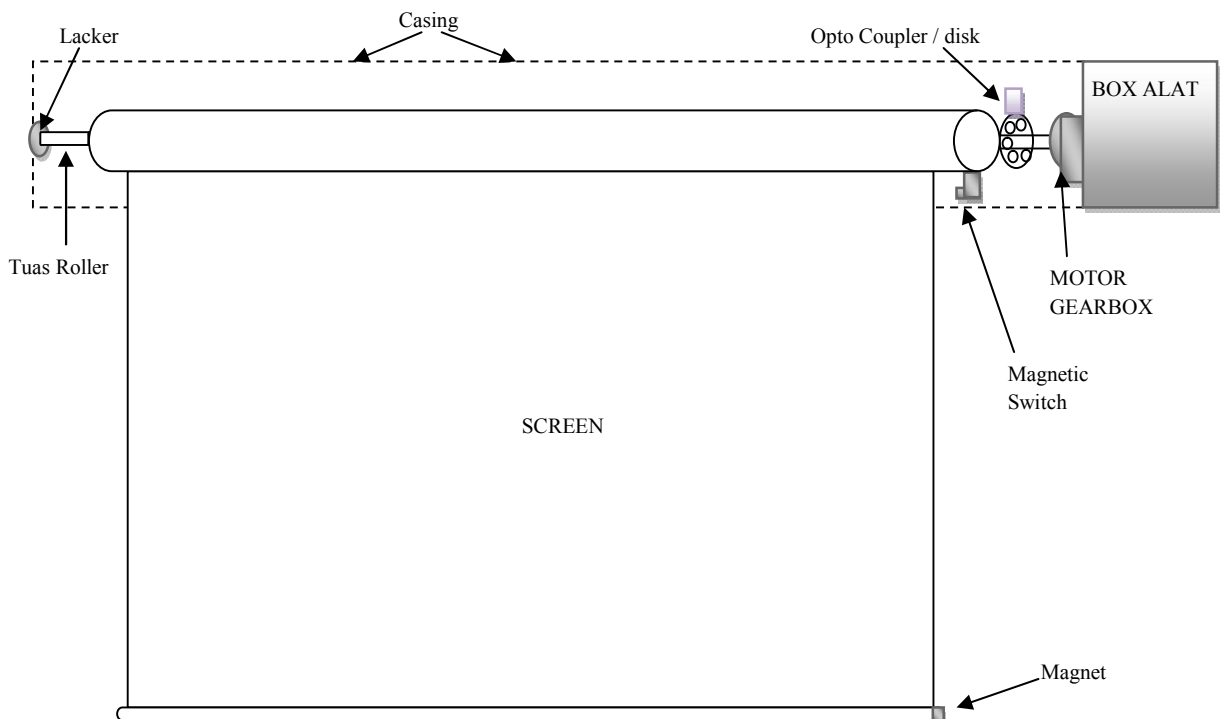




### 3.1.1 Perancangan Konstruksi Layar Proyektor

Pada blok diagram sebagaimana gambar 3.1, posisi motor DC, *magnetic door switch* untuk layar tertutup dan optocoupler untuk layar terbuka ditempatkan pada konstruksi mekanik dari layar yang dikendalikan. *Magnetic door switch* layar difungsikan sebagai detektor agar mikrokontroler AT-mega32 mengetahui apakah layar tersebut telah tertutup rapatsaat motor digerakkan untuk menggulung layar, sementara untuk posisi terbuka penuh digunakan optocoupler sebagai *rotari encoder* untuk mengetahui seberapa besar layar terbuka sehingga dalam keadaan tersebut mikrokontroler AT-Mega32 dapat menghentikan putaran Motor DC untuk menghindari kerusakan pada Motor akibat *overload* atau *overcurrent*. Adapun perancangan konstruksi mekanik ditunjukkan pada gambar 3.2:



Gambar 3.2 Perancangan Konstruksi Mekanik

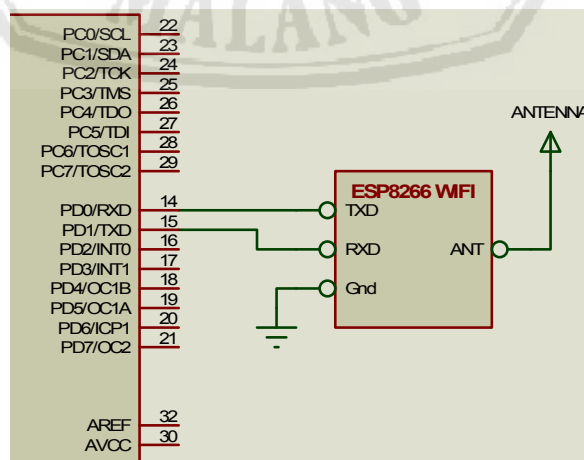
### 3.1.2 Perencanaan Konstruksi Mekanik Layar Proyektor

Berdasarkan block diagram sebagaimana Gambar 3.1 dan perancangan Mekanik pada gambar 3.2, maka prinsip kerja dari sistem adalah sebagai berikut: Saat pertama kali power supply dinyalakan, maka mikrokontroler AT-mega32 melakukan inisialisasi terhadap port input-output, seperti halnya komunikasi serial asynchronus untuk keperluan komunikasi dari *smartphone* ke

modul wifi, inialisai port LCD, port input *magnetic door swich*, keypad dan port pengendali driver. Saat inialisai dilakukan, maka selanjutnya MCU mikrokontroller AT-mega32 menampilkan status layar pada LCD, pada kondisi ini MCU siap menerima data masukan dari modul wifi, apabila ada perintah yang dikirimkan dari *smartphone* android berupa kode pengaktifan pengontrolan layar melalui jaringan Wifi. Jika perintah yang dikirim sesuai, maka selanjutnya akan di proses oleh mikrokontroller untuk di eksekusi pada motor agar dapat membuka atau menutup layar sesuai perintah yang dipilih pada menu android. Sementara itu *keypad* digunakan untuk memasukan *password user* dan SSID (*service set identifiere*) yang nantinya akan di proses oleh mikrokontroller serta pengoperasian layar secara manual melalui tombol.

### 3.1.3 Perancangan Modul Wifi Menggunakan Modul ESP8266

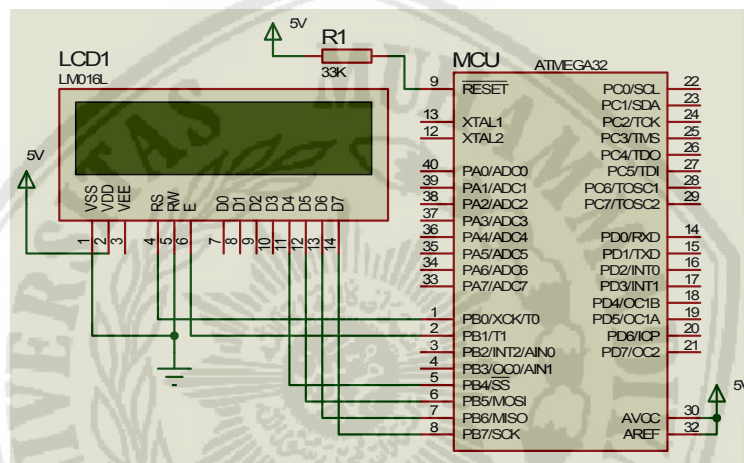
Untuk dapat berkomunikasi secara *wireless* antara sistem dengan *smartphone* android, maka pada perancangan ini digunakan perangkat *Wifi to serial* menggunakan modul ESP8266 yang dihubungkan ke rangkaian kontroller. ESP8266 merupakan piranti elektronik yang bekerja pada protokol *wireless* IEEE802.11.4/b/g/n atau yang umum dikenal dengan Wifi. Modul ini bekerja dengan komunikasi serial UART dengan variable baudrate 9600, 19200, 57600, 115200 dan diakses melalui protokol ATcommand sehingga dapat dihubungkan langsung ke serial mikrokontroller. Adapun hubungan pin AT-Mega32 dengan ESP8266 ditunjukkan pada gambar 3.3:



Gambar 3.3 Rangkaian modul Wifi ESP8266

### 3.1.4 Perancangan Rangkaian LCD

Bagian penampil informasi pada perancangan block diagram adalah menggunakan LCD 16X2, agar sistem yang dikendalikan oleh mikrocontroller AT-Mega32 dapat diketahui dan dimengerti oleh audiens, maka diperlukan sebuah penampil informasi LCD. LCD digunakan untuk menampilkan menu-menu registrasi pada sistem dan perintah-perintah yang dipantau audiens dalam melakukan pengontrolan terhadap layar yang dikontrol. LCD yang digunakan adalah LCD M1632 yang mampu menampilkan karakter 2 baris 16 Karakter. Adapun pin koneksi rangkaian LCD terhadap controller (MCU) ditunjukkan pada Gambar 3.4:



Gambar 3.4 Rangkaian LCD

Nilai  $R_1$  ditentukan 33K hal ini mengacu pada ketentuan datasheet AT-mega32, dimana resistor pullup untuk reset adalah 30K hingga 60K yang disarankan.

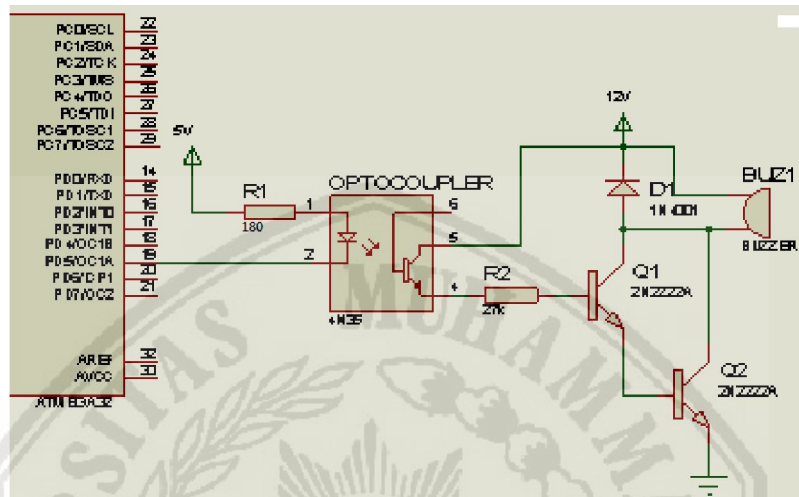
### 3.1.5 Perancangan rangkaian Driver Buzzer

Pada perancangan ini Buzzer yang digunakan mempunyai tegangan suplay sebesar 12Volt DC, sementara itu tegangan sistem AT-mega32 yang bertindak sebagai pengendali Buzzer adalah 5V, maka untuk mengontrol Buzzer diperlukan rangkaian Driver yang pada rancangan ini menggunakan perantara Optocoupler sebagai pemisah beda tegangan antara rangkaian driver Buzzer dan minimum sistem AT-mega32. Pada perencanaan Driver Buzzer dari alat yang akan dirancang, Buzzer yang digunakan mempunyai resistansi coil sekitar 50 $\Omega$  dan

membutuhkan catu daya sebesar 12Volt DC, dengan demikian, maka arus yang diperlukan buzzer adalah :

$$I = \frac{VCC}{R_{coil\_Buzzer}} = \frac{12}{50} = 0,24 \text{ Amper.}$$

Sementara itu perancangan rangkaian driver buzzer ditunjukkan dalam Gambar 3.5



Gambar 3.5 Rangkaian Driver Buzzer

Alasan penggunaan IC optocoupler ini adalah untuk pengisolasian rangkaian kontrol dengan beban supaya tidak terjadi gangguan pada saat sistem bekerja. Dalam rangkaian ini juga terdapat dioda yang berfungsi untuk melewatkan arus balik Coil Buzzer pada saat perubahan kondisi dari keadaan ON ke OFF supaya tidak merusak transistor 2N222A. Dioda yang digunakan sebaiknya yang mempunyai kemampuan melewatkan arus lebih besar dari besarnya arus balik Coil Buzzer yang akan melewatinya.

Transistor yang digunakan sebagai driver dipilih dengan pertimbangan arus collector maksimum yang besarnya 3 kali. Dengan kebutuhan arus tersebut maka dipilih transistor type 2N222A sebagai driver Buzzer. Dari datasheet diketahui bahwa transistor tipe 2N222A memiliki arus kolektor sebesar 800mA. Dalam perencanaannya kedua transistor 2N222A didarlington untuk penguatan yang besar yakni perkalian dari penguatan kedua transistor tersebut .

Ditetapkan drop tegangan pada  $V_{ce\ opto} = 0,3V$  (*saturation*).

Maka  $R_2$  atau  $R_b$  dapat dicari dengan rumus:

$$R_2 = \frac{V_{cc} - V_{ce(opto)} - 2V_{be}}{I_b}$$

Dimana  $I_b = I_c \text{ opto}$

untuk  $I_c$  dapat dicari dengan menetapkan berapa  $I_b / \beta$  dari tiap- tiap transistor.

$$I_c = I_b \cdot (\beta_1 \cdot \beta_2)$$

Sedangkan  $I_{in}$  dapat dicari dengan :

$$I_{in} = \frac{V_{cc}}{R_{in}}$$

Untuk semua perencanaannya dapat dilihat sebagai berikut:

Dalam pengukuran diketahui  $R_{in} = 50 \Omega$

Maka:

$$I_{in} = \frac{12V}{50 \Omega} = 0,24 A = 240mA$$

Maka kita harus mencari transistor yang mempunyai  $I_c$  lebih besar dari  $I_{in}$ .

Untuk itu dipilih transistor 2N222 yang memiliki  $I_c = 800mA$ .

Karena  $I_b = I_c \text{ TR}_1$ , Maka:

$$I_c \text{ TR}_1 = \frac{I_c \text{ TR}_2}{\beta}$$

$$= \frac{800mA}{50} = 0,016A$$

$$I_b \text{ TR}_1 = \frac{I_c \text{ TR}_1}{\beta}$$

$$= \frac{0,016 A}{50} = 0,00032A$$

Dengan demikian nilai  $R_b$  dapat ditentukan dengan rumus:

$$R_b = \frac{V_{cc} - V_{ce(opto)} - 2V_{be}}{I_b \text{ TR}_1}$$

$$= \frac{(12 - 0,3 - 2 * 1,2)V}{0,00032A}$$

$$= \frac{9,3V}{0,00032A} = 29,062 \Omega \approx 27K\Omega \text{ (harga R dipasaran)}$$

Untuk nilai  $R_1$  dapat dicari dengan rumus:

$$R_1 = \frac{V_{OH} - V_{LED}}{I_{LED}}$$

Dimana nilai  $V_{LED} = 1,5 \text{ V}$ ,

$V_{OH} = 4,2 \text{ V}$  (untuk kondisi  $I_{OH} = 20 \text{ mA}$  pada tegangan  $5 \text{ V}$ )

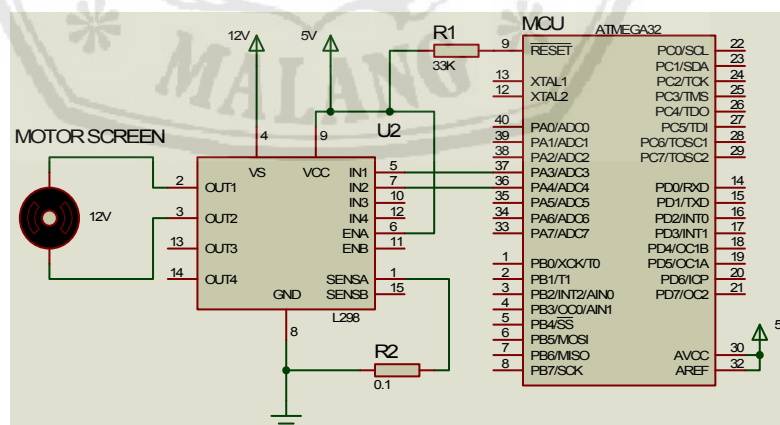
dan  $I_{LED}$  sebesar  $16 \text{ mA}$

$$\text{Maka } R_1 = \frac{(4,2 - 1,5) \text{ V}}{16 \text{ mA}} = 168,75 \Omega \approx 180 \Omega$$

sehingga didapat nilai  $R_1 = 180 \Omega$  (dipasaran).

### 3.1.6 Perancangan Driver Pengontrol Motor

Untuk dapat mengendalikan arah putaran motor dalam membuka dan menutup layar proyektor, maka diperlukan rangkaian *driver* motor DC. Dengan demikian maka *driver* motor yang digunakan adalah *driver* 2 arah yang mana fungsinya dapat membalik polaritas tegangan motor DC penggerak mekanik layar. Pada perancangan ini *driver* motor yang digunakan adalah IC driver L298N yang dapat difungsikan sebagai *driver* motor 2 arah dengan kemampuan maksimum  $4 \text{ Amper}$  serta dapat diparalel hingga  $8 \text{ Amper}$  pada tiap *driver*nya. Adapun perancangan rangkaian *driver* 2 arah ditunjukkan sebagaimana gambar 3.6:



Gambar 3.6 Rangkaian driver motor L298

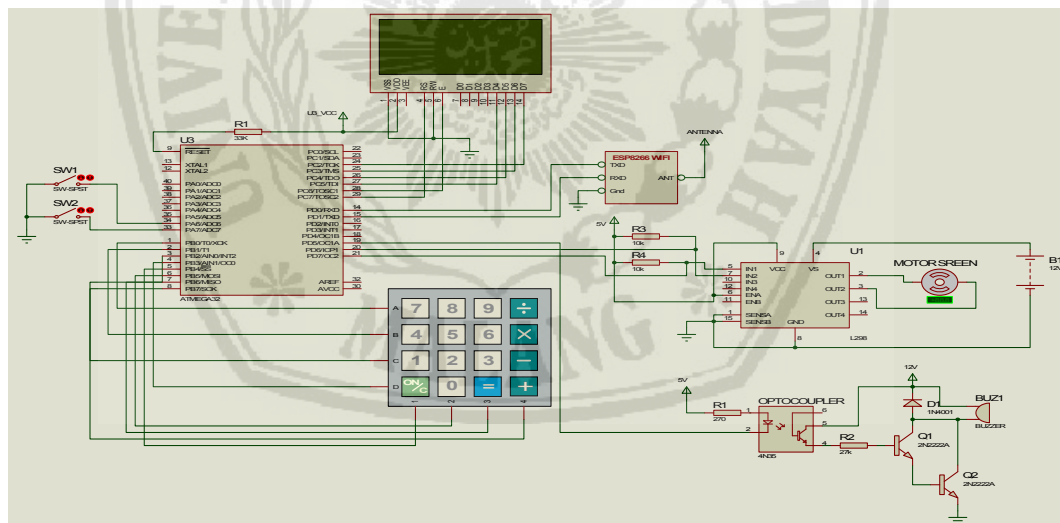
Dengan mengacu pada prinsip kerja sebagaimana datasheet, maka untuk mengakses *driver* L298N ditunjukkan sebagaimana tabel 3.1:

Tabel 3.1 Tabel kebenaran *Driver Relay*

No.	Input Driver IN1	Input Driver IN2	Kondisi Motor
1.	0	0	Stop
2.	0	1	Putar Kanan
3.	1	0	Putar Kiri
4.	1	1	Stop

### 3.1.7 Perancangan Mikrokontroler ATMEGA32

Pada perancangan alat ini, controller yang digunakan adalah AT-Mega32. mikrokontroler AT-mega32 merupakan pengontrol utama dari sistem pengontrol dan pengaman layar menggunakan *Magnetic door switch*, dimana semua proses pembacaan data serial dari optocoupler, modul wifi, *magnetic door swith*, pengontrolan ke driver, pembacaan *keypad* serta penampil informasi pada LCD dilakukan sepenuhnya oleh AT-Mega32. Adapun konfigurasi pin-pin yang dipakai dan rangkaian microcontroller AT-Mega32 ditunjukkan pada Gambar 3.7



Gambar 3.7 Rangkaian Kontrol AT-Mega32

Agar sistem dapat beroperasi dengan benar, maka diperlukan *software* AT-Mega32 yang didownloadkan pada memori flash didalamnya.

### 3.2 Perancangan Software

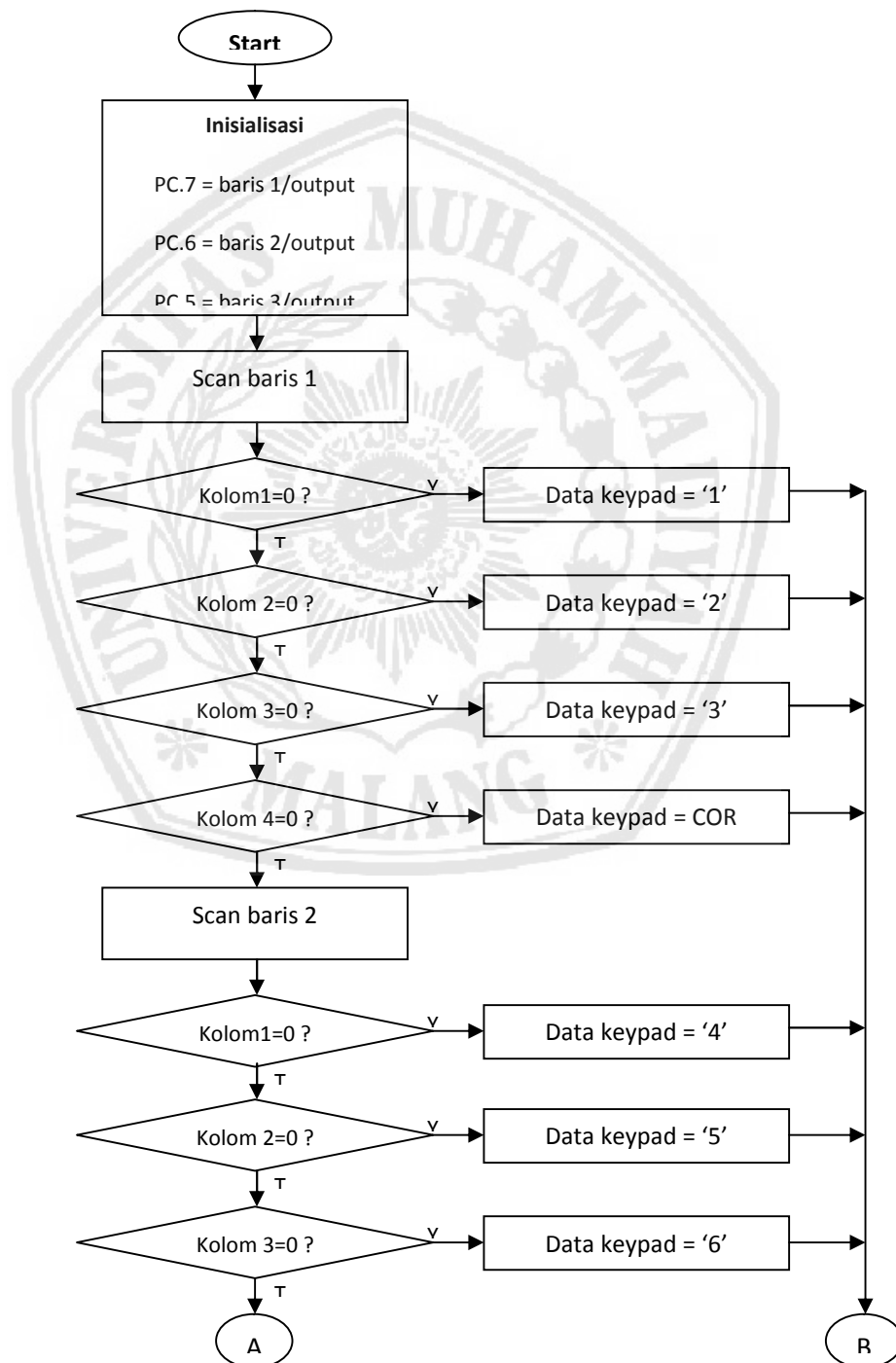
Agar seluruh *hardware* yang dirancang dapat dikontrol dan dimonitor oleh mikrokontroler, maka diperlukan *software* pengendali yang didownloadkan pada

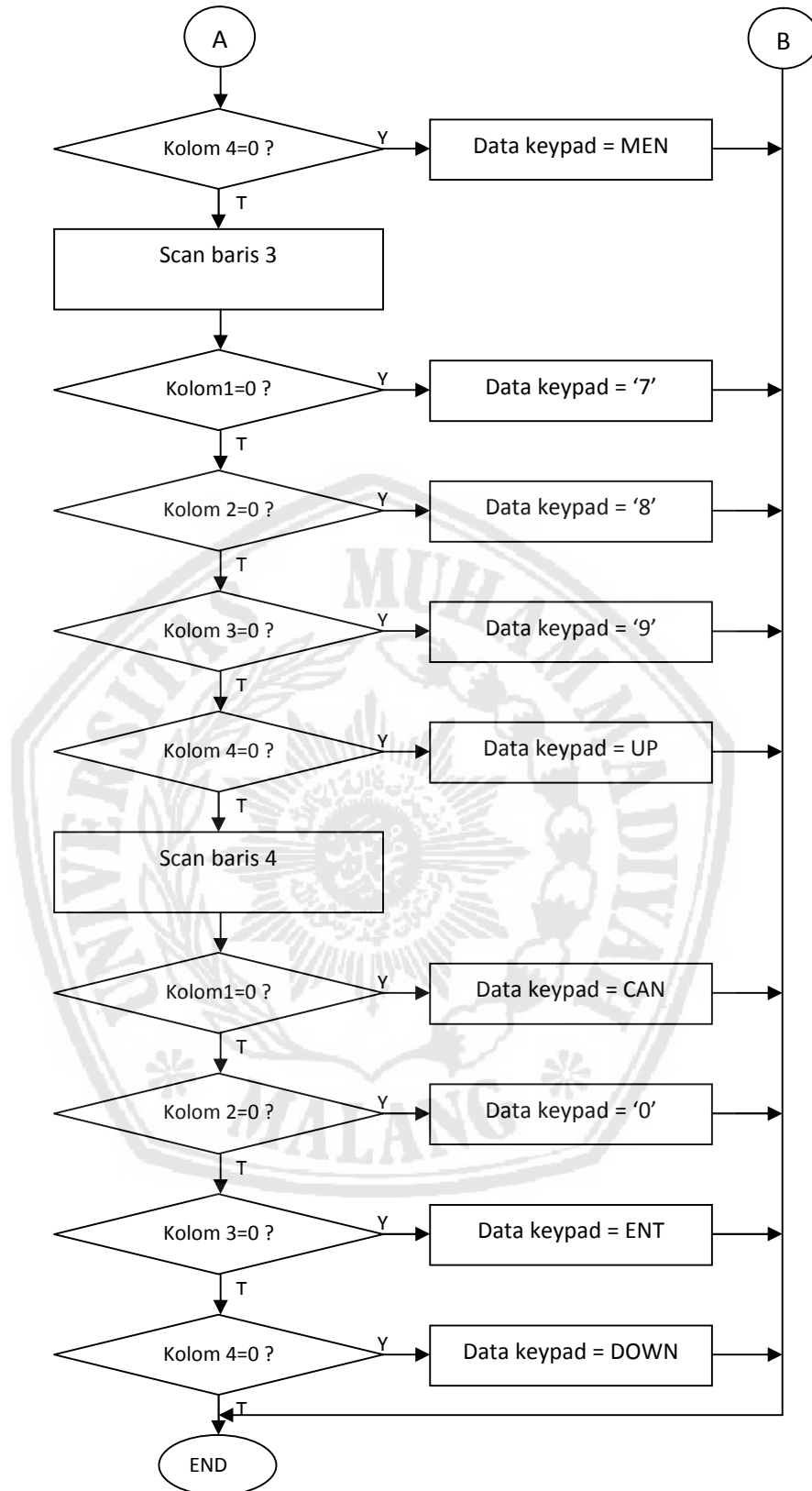


mikrokontroler tersebut. Perancangan *software* pada sistem pengaman dan pengontrol layar dirancang menggunakan *software* BASCOM-AVR. Sedangkan proses dari kerja masing-masing *software* mengacu pada algoritma kerja sistem sebagaimana pembahasan berikut:

### 3.2.1 Algoritma Pembacaan Matrix Keypad

Pembacaan *keypad* matrix 4X4 dalam perancangan ini menggunakan metode *scan*, adapun prosesnya ditunjukkan sebagaimana *flowchart* pada Gambar 3.8:



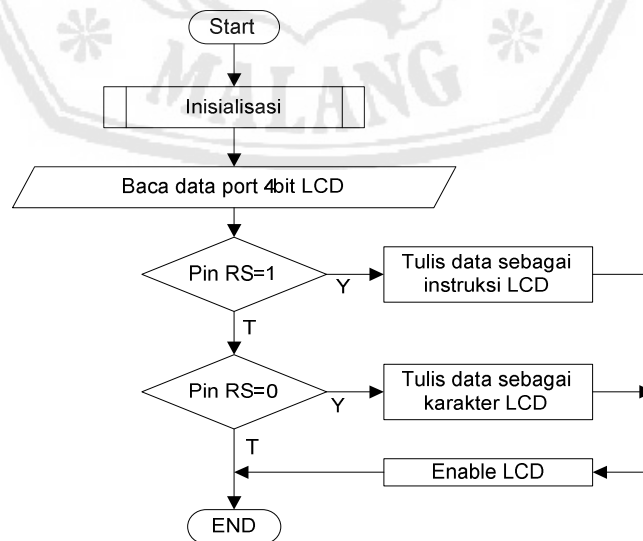


Gambar 3.8 Algoritma pembacaan Keypad dengan sistem matrik

Proses scanning keypad matrix 4x4 pada perancangan sebagaimana ditunjukkan pada flowchat gambar 3.8 merupakan proses pencarian data penekanan tombol yang diakses secara terus menerus, dimana proses yang dilakukan adalah dengan memberikan data baris yang hanya memebrikan logika nol pada salah satu baris sementara baris lainnya 1 (high), kemudian jika ada penekanan tombol pada area kolom 1 hingga kolom 4, maka nilai input pada kolom tersebut akan ikut berlogika nol, hasil input tersebut selanjutnya dihubungkan dengan nomor matrix dari keypad untuk mendapatkan data dari masukan keypad, contoh untuk scan baris 1, data yang diberikan pada PORTB=01111111B, dimana baris 1 ditempati port B.7 yang berlogika 0, jika pada saat tersebut terdapat penekanan tombol, maka salah satu dari input kolom akan berlogika 0 akibat short ke 0 dari port B.7, seuatu kolom 1 ditekan (tombol 1 atau baris 1 kolom 1), maka port B.3 yang ditempati kolom 1 akan berlogika 0 dana data port B menjadi 01110111B yaitu port B.3 short dengan port B.7, hasil ini selanjutnya digunakan sebagai acuan data penekanan tombol dari baris 1 kolom 1 dan memberikan hasil data keypad 1 sebagai tanda angka 1 ditekan. Begitu seterusnya untuk scan baris 2 dan selanjutnya.

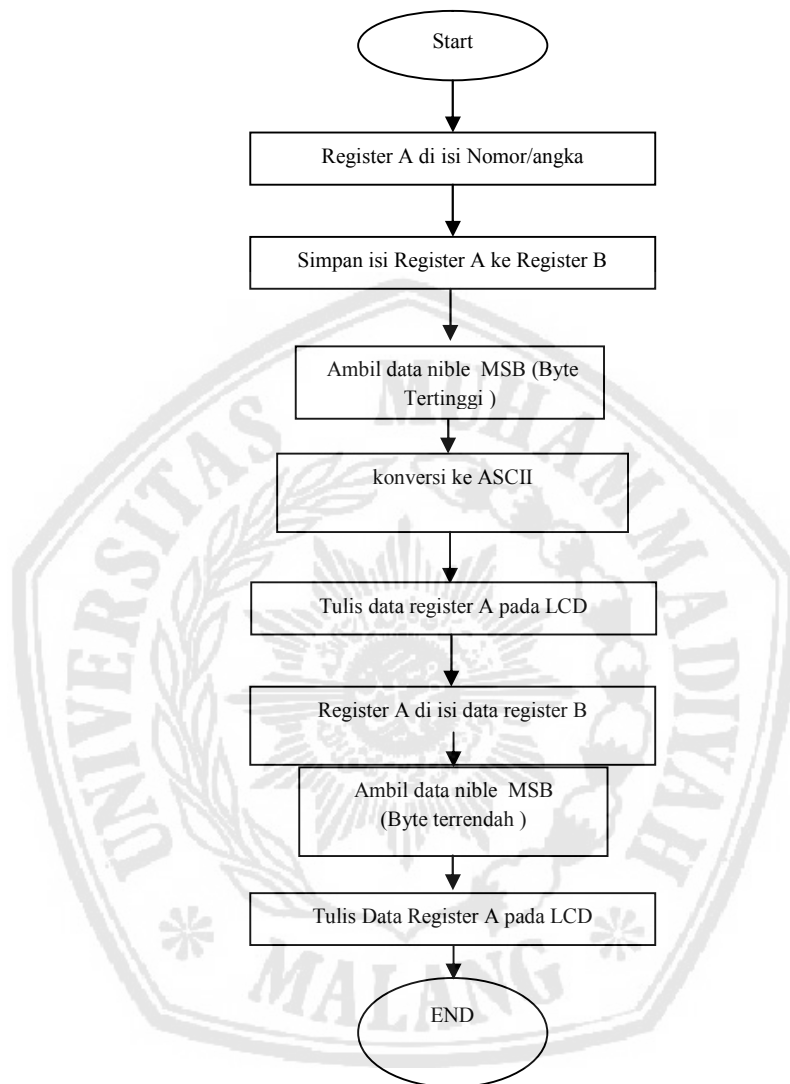
### 3.2.2 Algoritma Tulis Instruksi LCD

Prinsip kerja dari proses penampil LCD ditunjukkan pada *flowchart* dalam Gambar 3.9



Gambar 3.9 Algoritma penulisan LCD

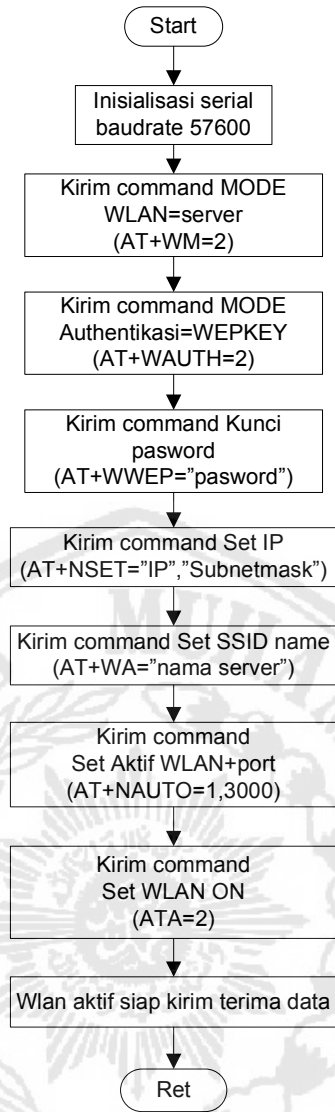
Sementara itu proses penampil angka pada LCD saat menampilkan hasil daya, arus, tegangan maupun *setting* memerlukan beberapa tahapan proses sebagaimana ditampilkan pada Gambar 3.10



Gambar 3.10 Algoritma tulis data numeric pada LCD

### 3.2.3 Algoritma Seting Komunikasi WLAN Pada Modul Wifi

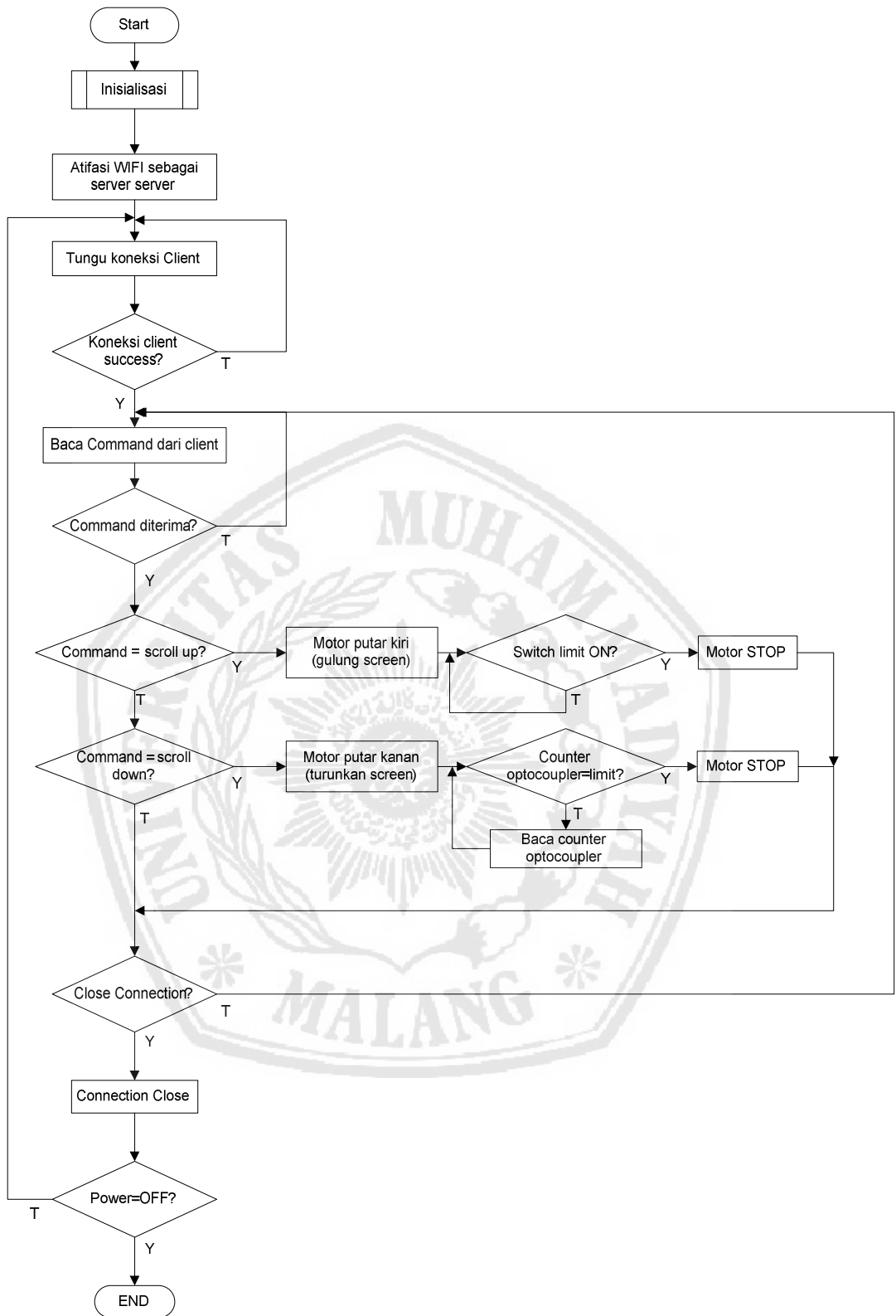
Agar modul wifi ESP8266 dapat bekerja pada jaringan LAN, maka perlu dilakukan inisialisasi berupa seting IP, mode WLAN, SSID dan lain sebagainya dengan mangacu pada datasheet. Adapun perancangan seting komunikasi pada ESP8266 ditunjukkan pada gambar 3.11



Gambar 3.11 Algoritma seting wifi pada modul ESP8266

### 3.2.4 Algoritma Perangkat Lunak Keseluruhan

Adapun perancangan perangkat lunak secara keseluruhan mengacu pada flowchart sistem sebagaimana gambar 3.12



Gambar 3.12 Algoritma keseluruhan