

BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

Perancangan dan pembuatan alat dalam bab ini secara garis besar terdapat dua bagian perancangan dalam pembuatan alat yaitu :

- Perancangan perangkat keras (*Hardware*)
- Perancangan perangkat lunak (*Software*)

Pada perancangan perangkat keras akan meliputi rangkaian-rangkaian dan sistem antarmuka pada mikrokontroler yang digunakan pada alat ini. Sedangkan pada perancangan perangkat lunak meliputi diagram alir dan *software* secara umum.

4.1 Spesifikasi Alat

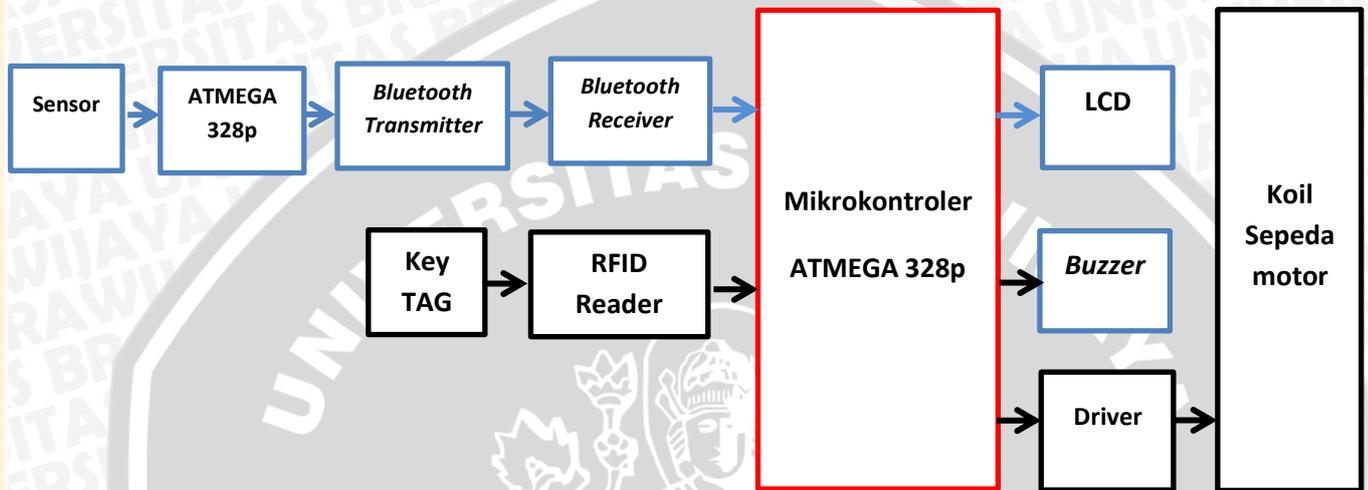
Spesifikasi alat secara global ditetapkan terlebih dahulu sebagai acuan dalam perancangan selanjutnya. Spesifikasi alat yang dirancang yaitu :

- Alat akan bekerja ketika kunci pada tali helm 'klik' atau 'on' dan kontak juga pada posisi 'on' atau keduanya dalam keadaan 'high'.
- Modul *Bluetooth* yang dipakai adalah HC-05 yang sudah dilengkapi *PCB Extendednya*.
- Mikrokontroler yang dipakai adalah AVR ATmega328p. ATmega ini terdapat pada helm dan kontak. ATmega ini berfungsi untuk mengaktifkan *Bluetooth master* dan mengolah keluaran *bluetooth slave* agar dapat diproses lebih lanjut.
- *Buzzer* digunakan sebagai indikator jika terjadi kesalahan pada langkah "starter" sepeda motor.
- LCD digunakan untuk menampilkan karakter yang diinginkan.
- Modul sensor dan Mikrokontroler menggunakan *supply* tegangan 5 V_{DC}.

4.2 Diagram Blok Sistem

Keseluruhan bagian *hardware* atau biasa disebut dengan diagram blok *hardware*. Pada gambar 4.1 merupakan gambar blok diagram dari *hardware*.

Penelitian ini dikerjakan oleh dua peneliti. Adapun yang akan dikerjakan pada penelitian ini adalah blok yang berwarna biru, sedangkan yang berwarna merah dikerjakan secara bersama-sama, dan blok berwarna hitam dikerjakan oleh peneliti lainnya.



Gambar 4.1. Diagram blok sistem dari *hardware*

Penjelasan masing-masing blok diagram:

a. *Sensor*

Merupakan skalar yang melekat pada tali helm, sehingga ketika tali helm "klik", maka saklar akan 'on' dan arus akan mengalir untuk mengaktifkan catu pada mikrokontroler dan *Bluetooth*.

b. *ATMega 328p* (blok biru)

Merupakan modul yang digunakan untuk mengolah data yang akan dikirimkan oleh *Bluetooth transmitter/master*.

c. *Bluetooth Transmitter/master*

Merupakan modul yang berfungsi sebagai pengirim data dari mikrokontroler.

d. *Bluetooth Receiver/salve*

Merupakan modul yang berfungsi sebagai penerima data dari *Bluetooth Transmitter/master*.

e. ATmega 328p (blok merah)

Merupakan mikrokontroler yang berfungsi sebagai pemroses dari masukan modul *bluetooth receiver/slave* serta menghasilkan *output* untuk mengaktifkan *buzzer* dan membuat tampilan pada LCD.

4.3. Prinsip Kerja Alat

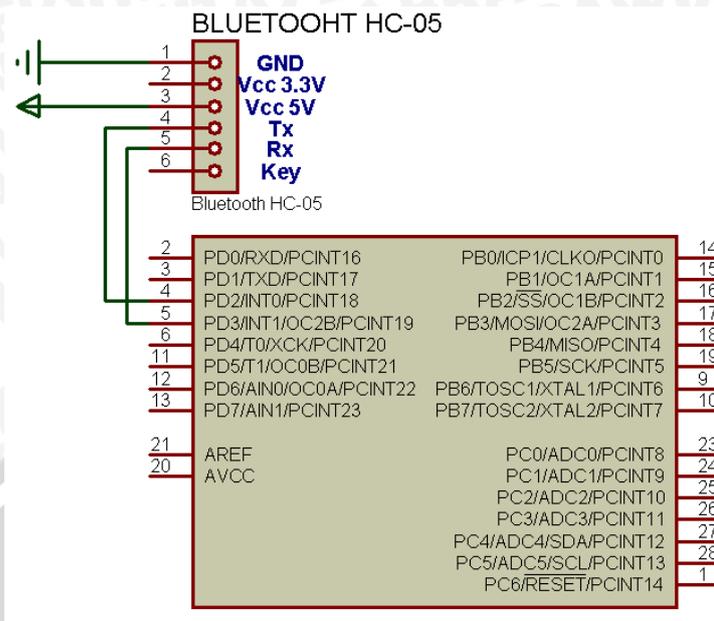
Prinsip kerja alat berdasarkan blok diagram pada Gambar 4.1. Pada saat sensor atau saklar helm "klik", Vcc mencatu mikrokontroler *master*, data pada mikrokontroler *master* akan dikirimkan melalui media *bluetooth master*. *Bluetooth slave* akan menerima data dari *bluetooth master* kemudian data diolah oleh mikrokontroler *master*. LCD akan menampilkan karakter jika proses pengiriman data tersambung atau tidak. *Buzzer* akan aktif jika helm tidak dalam keadaan tersambung atau "klik".

4.4. Perancangan Perangkat Keras (*Hardware*)

4.4.1. Perancangan Rangkaian Antarmuka *Bluetooth*

Modul *bluetooth* ini digunakan untuk mengirim dan menerima data mikrokontroler yang diletakkan pada helm dan kendaraan. Agar modul ini dapat bekerja maka *supply* modul diberi tegangan 5 V. Rangkaian koneksi antara mikrokontroler dengan *bluetooth* ditunjukkan pada Gambar 4.2. Pin-pin yang harus dihubungkan dengan mikrokontroler dalam perancangan ini antara lain:

- Pin 1 dihubungkan dengan GND
- Pin 2 dapat dihubungkan dengan Vcc 3.3 V
- Pin 3 dapat dihubungkan dengan Vcc 5 V
- Pin 4 dihubungkan dengan pin PD2/pin 2 digital mikrokontroler
- Pin 5 dihubungkan dengan pin PD3/pin 3 digital mikrokontroler



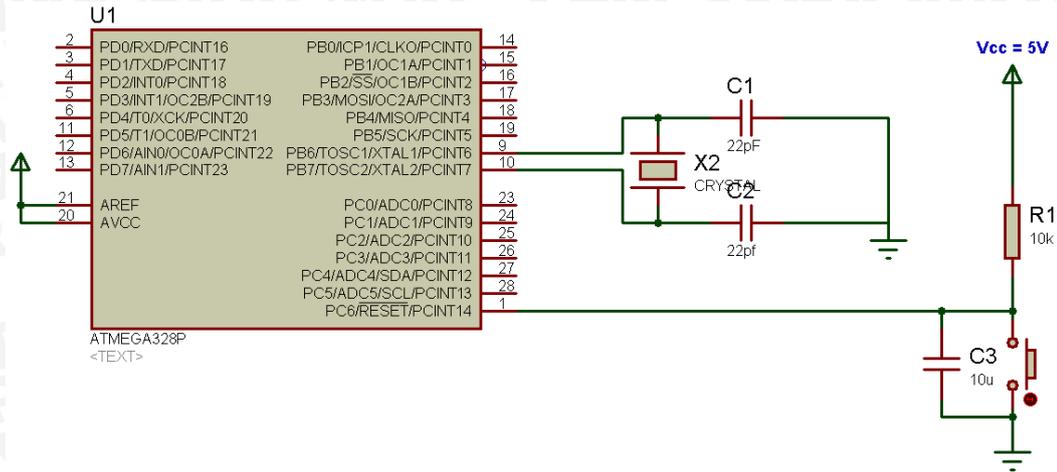
Gambar 4.2. Rangkaian Antarmuka *Bluetooth* dengan Mikrokontroler ATmega 328p

4.4.2. Perancangan Rangkaian Minimum Sistem Mikrokontroler ATmega 328p

Mikrokontroler yang digunakan pada rangkaian ini adalah ATmega 328p yang termasuk dalam seri AVR. IC mikrokontroler dapat bekerja dengan baik jika ditambahkan sebuah rangkaian *oscillator* atau pembangkit *clock* dan rangkaian reset. Rangkaian *oscillator* terdiri dari dua kapasitor dan xtal (Kristal). Sedangkan untuk rangkaian reset terdiri dari *push button*, resistor dan kapasitor.

Pada mikrokontroler ATmega 328p nilai XTAL yang diizinkan berkisar antara 0,4 MHz sampai 16 MHz. Perancangan ini menggunakan kristal sebesar 16 MHz. Besar nilai C1 dan C2 disesuaikan dengan apa yang tertera dalam *datasheet*, yaitu antara 12 pF sampai 22 pF. Perancangan minimum sistem yang digunakan dalam alat ini menggunakan kapasitor sebesar 22 pF. Rangkaian minimum sistem mikrokontroler ATmega328p ditunjukkan dalam Gambar 4.3.





Gambar 4.3. Rangkaian Minimum Sistem Mikrokontroler ATmega328p

Fungsi *reset* dalam rangkaian mikrokontroler adalah mengulang merestart program apabila terjadi *error* pada saat mikrokontroler sedang bekerja. Minimum *time reset* pada *datasheet* mikrokontroler ATmega328p tertera 2.5 μ s. Jadi dalam penentuan komponen penghitungan waktu untuk *time reset* harus di atas dari waktu minimum. Komponen yang digunakan dalam perancangan ini adalah kapasitor C3=10 μ F dan resistor R1= 10 k Ω . Penghitungan waktu untuk *time reset* apabila mengacu pada tegangan minimal kapasitor yang diketahui dalam *datasheet* ATmega 328p agar dapat melakukan *reset* adalah 0.2 V_{CC} , $V_{CC} = 5$ V sehingga didapatkan *time reset*:

$$\begin{aligned} V_{C(min)} &= 0.2 \times V_{CC} \\ &= 0.2 \times 5 \text{ V} \\ &= 1 \text{ V} \end{aligned}$$

$$V_C = V_s \cdot e^{-t/R.C(t)}$$

$$1 = 5 \cdot e^{-t/10K\Omega \cdot 10\mu F}$$

$$\frac{1}{5} = e^{-t/0.1}$$

$$\ln 0.2 = -t/0.1$$

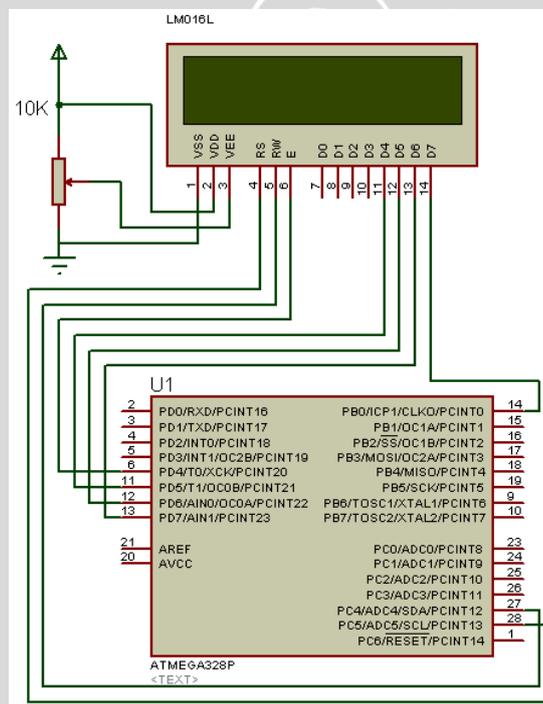
$$-1.6 = -t/0.1$$

$$t = 0.16 \text{ s}$$

Jadi nilai t yang didapat sebesar 0.16 s, sehingga pemilihan komponen kapasitor $C3=10\mu\text{F}$ dan resistor $R1=10\text{ K}\Omega$ memiliki *time reset* melebihi batasan minimal yang ditentukan oleh *datasheet* ATmega 328p.

4.4.3. Perancangan Rangkaian Antarmuka LCD

Pada perancangan rangkaian antarmuka LCD ini *port* yang digunakan pada mikrokontroler adalah PORT-BCD. Terdapat beberapa pin yang digunakan untuk antarmuka dengan mikrokontroler pada LCD di antaranya 3 pin yang digunakan sebagai pin kontrol yaitu pin 4 sebagai RS (*register select*), pin 5 sebagai R/W (*read/write*) dan pin 6 sebagai E (*enable*). Sedangkan untuk jalur data terdapat pada pin 11–14. Rancangan antarmuka LCD ditunjukkan dalam Gambar 4.4.

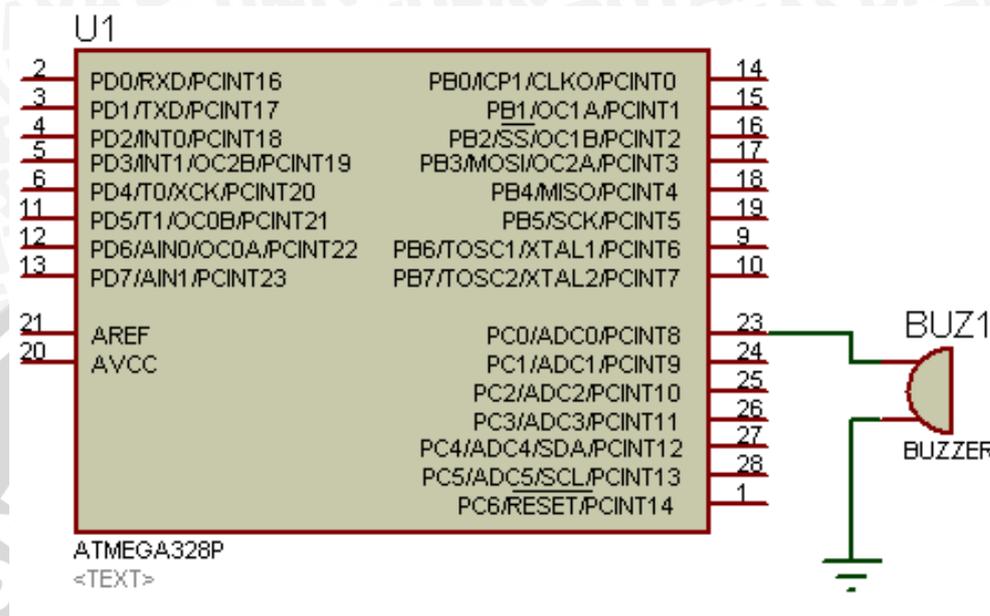


Gambar 4.4. Rangkaian Antarmuka LCD dengan Mikrokontroler ATmega328p

4.4.4. Perancangan Rangkaian Antarmuka Buzzer

Perancangan antarmuka *buzzer* ini, logika yang digunakan untuk mengaktifkan *buzzer* adalah aktif *high*. Sehingga, *buzzer* akan aktif ketika *output* dari mikrokontroler berlogika 1. Berdasarkan *datasheet*, I_{in-max} yang dapat diterima *buzzer* adalah 40mA maka *buzzer* tidak membutuhkan resistor untuk mengurangi arus yang memasuki *buzzer* karena *output* mikrokontroler hanya

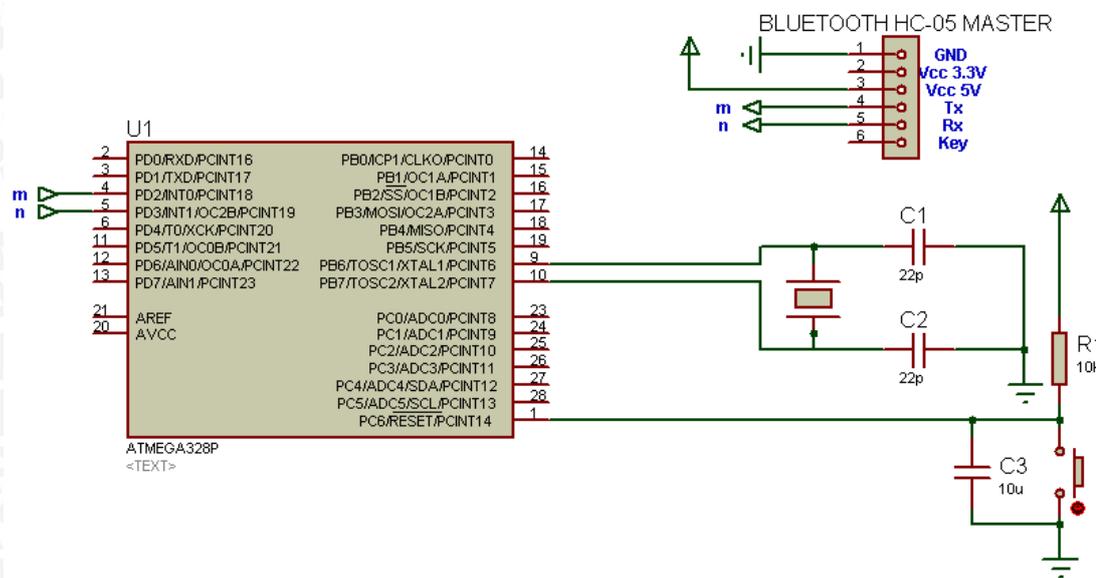
berkisar $\pm 20\text{mA}$ Rangkaian antarmuka *buzzer* dengan mikrokontroler ditunjukkan dalam Gambar 4.5.

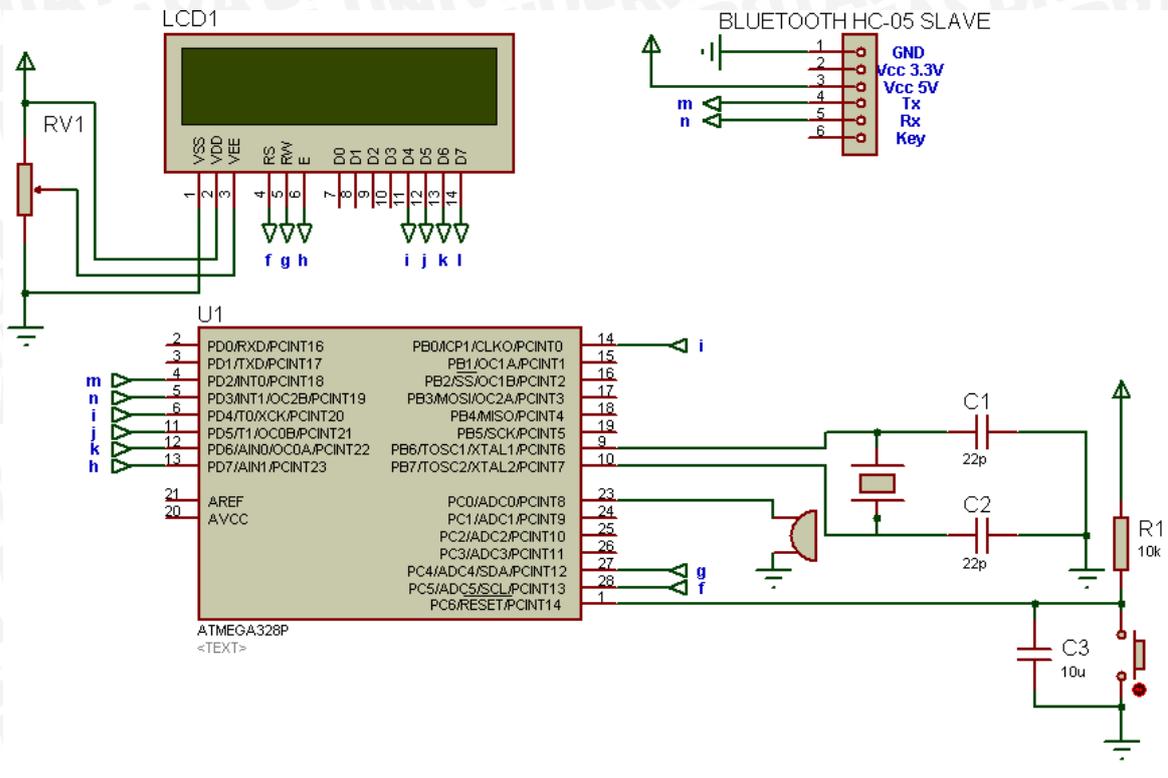


Gambar 4.5. Antarmuka *buzzer* dengan mikrokontroler

4.4.5. Perancangan Keseluruhan Sistem

Perancangan ini dilakukan dengan menggabungkan semua perancangan yang telah dilakukan dari masing-masing blok diagram yang telah ditentukan. Sistem keseluruhan dari alat ini ditunjukkan dalam Gambar 4.6.

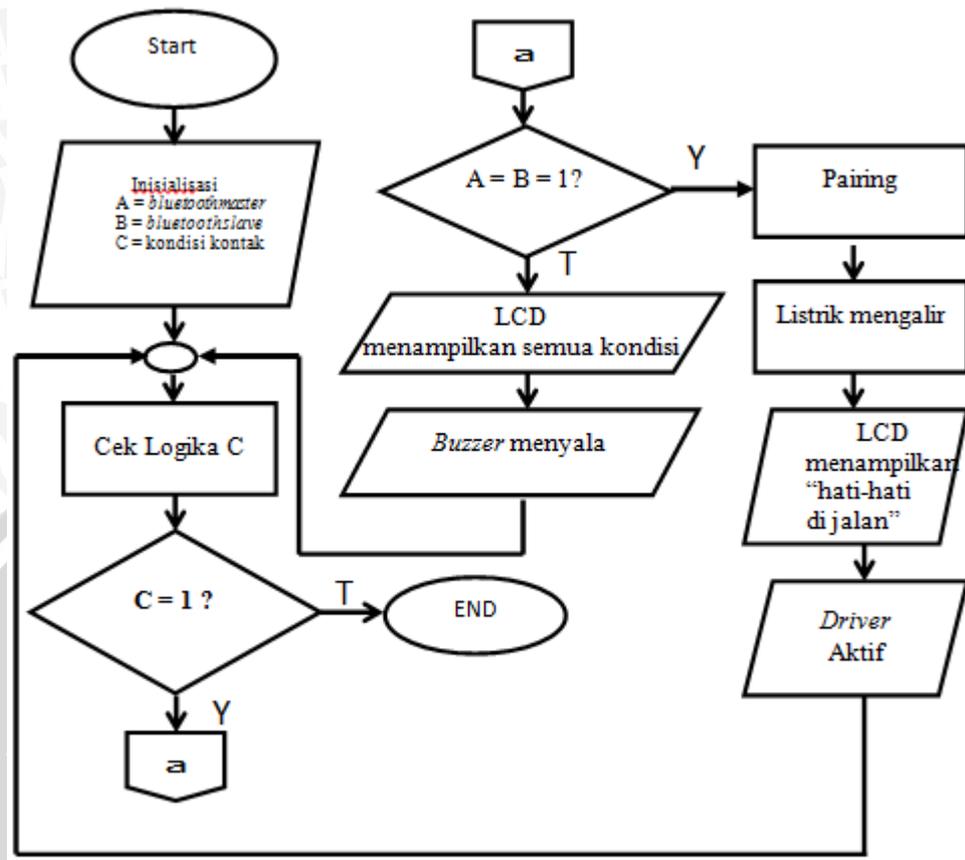




Gambar 4.6. Rangkaian keseluruhan sistem

4.5. Perancangan Perangkat Lunak (Software)

Perancangan perangkat lunak pada mikrokontroler berupa diagram alir atau *flowchart*. *Flowchart* ini berfungsi sebagai alur kerja dari rangkain *hardware* yang telah dirancang sebelumnya. Bahasa yang digunakan dalam membuat program untuk alat ini adalah bahasa C arduino, sedangkan *compiler* yang digunakan yaitu *Arduino*. *Flowchart* dari alat ini ditunjukkan dalam Gambar 4.7.



Gambar 4.7. Flowchart Prototype Alat

Alur kerja dari perancangan *software* di atas adalah pertama mikrokontroler akan melakukan inisialisasi di antaranya inisialisasi *bluetooth master*, *bluetooth slave* dan kontak. Kemudian melakukan pengecekan apakah kontak motor dalam keadaan hidup atau tidak. Apabila kontak motor hidup, *bluetooth slave* akan aktif. Apabila *bluetooth slave* dan *bluetooth master* keduanya dalam kondisi aktif, maka keduanya akan melakukan *pairing*. Dengan adanya *pairing*, maka listrik pada *driver* akan mengalir kemudian LCD akan menampilkan karakter "hati-hati di jalan".