

## SIMULATOR PENGATUR LAMPU LALU LINTAS BERDASARKAN WAKTU DAN KEPADATAN KENDARAAN BERBASIS MIKROKONTROLER AT89S52

**IGAP. Raka Agung**

Staff Pengajar Teknik Elektro, Fakultas Teknik, Universitas Udayana  
Kampus Bukit Jimbaran, Bali, 80361, tlp 0361 703315  
Email. puturaka@ee.unud.ac.id

### Abstrak

Sistem kontrol lampu lalu lintas sangat diperlukan untuk mengatur lalu lintas di persimpangan jalan. Agar sistem kontrol bekerja dengan baik untuk mengatasi kemacetan salah satu yang bisa dilakukan adalah dengan mengatur waktu dan lama nyala lampu lalu lintas yang disesuaikan dengan waktu dan hari saat kondisi lalu lintas normal, padat maupun lengang. Dari data kepadatan lalu lintas yang didapat di lapangan dilakukan perhitungan untuk mendapatkan waktu dan lama nyala lampu lalu lintas yang disimulasikan dengan peralatan yang dikendalikan oleh mikrokontroler AT89S52.

Mikrokontroler yang difungsikan sebagai sistem kontrol lampu lalu lintas dilengkapi dengan *real time clock* (RTC) DS1307, dimana RTC ini akan digunakan sebagai acuan waktu yang akan disesuaikan dengan kondisi di lapangan. Data waktu yang didapat dari perhitungan dimasukkan ke EEPROM eksternal AT24C04 melalui tombol edit waktu dan tanggal yang nilainya bisa diubah-ubah sesuai dengan kondisi di lapangan dan juga ditampilkan pada display LCD 16x2. Display lampu lalu lintas berupa 12 led dan dilengkapi juga dengan tombol pengaturan darurat yang diatur secara manual. Dari data volume lalu lintas yang diperoleh dapat diklasifikasikan waktu dan kondisinya yaitu kondisi padat 1 (pukul 07.00-09.00), padat 2 (pukul 11.00-14.00), padat 3 (pukul 16.00-19.00), lengang (pukul 00.00-05.00) dan normal (selain waktu padat dan lengang). Hasil perhitungan nilai nyala lampu lalu lintas yang diperoleh dari persimpangan Jl Waribang-WR.Supratman Denpasar yaitu nyala lampu hijau masing-masing phase dan siklus optimum.

Hasil perhitungan nilai nyala lampu lalu lintas yang diperoleh dari persimpangan Jl Waribang-WR.Supratman Denpasar yaitu siklus optimum dan nyala lampu hijau masing-masing phase. Untuk Hari Senin – Kamis untuk kondisi padat 1 didapat 45 detik, 15 detik dan 20 detik, kondisi padat 2 didapat 38 detik, 19 detik dan 9 detik, kondisi padat 3 didapat 39 detik, 10 detik, 19 detik dan untuk kondisi normal didapat 30 detik, 7 detik dan 13 detik. Untuk Hari Jumat – Minggu untuk kondisi padat 1 didapat 37 detik, 13 detik dan 14 detik, kondisi padat 2 didapat 39 detik, 9 detik dan 20 detik, kondisi padat 3 didapat 38 detik, 8 detik, 20 detik dan untuk kondisi normal didapat 28 detik, 8 detik dan 10 detik. Realisasi simulator sistem kontrol lampu lalu lintas berbasis mikrokontroler AT89S52 ini, diharapkan dapat digunakan sesuai dengan kondisi di lapangan sehingga dapat mengurangi kemacetan.

**Kata kunci** : lalu lintas, mikrokontroler, RTC, port, siklus optimum

### 1. PENDAHULUAN

Perkembangan teknologi saat ini semakin pesat dalam kehidupan manusia. Banyaknya aktifitas manusia menyebabkan banyaknya sarana yang digunakan untuk mempermudah kegiatan manusia. Salah satunya adalah sarana transportasi darat, seperti mobil, sepeda motor, serta angkutan darat lainnya. Kondisi ini kadang menyebabkan kemacetan terutama di daerah perkotaan yang banyak aktifitasnya. Lampu lalu lintas sangat berperan mencegah terjadinya kemacetan, karena pemakai jalan menjadi tertib untuk mengikuti hidup-matinya lampu tersebut.

Sistem lampu lalu lintas sangat dibutuhkan untuk dapat mengatasi kondisi tersebut. Kebanyakan lampu lalu lintas yang dioperasikan tidak mengikuti kondisi kesibukan lalu lintas baik saat kondisi

normal, padat, maupun lengang. Untuk mengatasi kondisi tersebut, maka dibuatlah suatu sistem lampu lalu lintas yang diharapkan dapat mengatasi kemacetan berdasarkan waktu sibuk dan kepadatan kendaraan. Sistem kontrolnya terdiri dari mikrokontroler, RTC, EEPROM, tombol-tombol dan display. RTC berisikan data berupa jam, menit, detik, tanggal, bulan dan tahun, yang nantinya digunakan sebagai masukan ke mikrokontroler untuk mengontrol lampu lalu lintas sesuai dengan kondisi.

Pada penelitian ini diharapkan dapat merealisasikan sebuah simulator sistem kontrol lampu lalu lintas berdasarkan waktu dan kepadatan kendaraan dengan menggunakan mikrokontroler, yang nantinya dapat digunakan untuk mengatasi kemacetan sesuai dengan waktu dan kondisi kepadatan lalu lintas

**2 METODE**

**2.1 Lampu Lalu Lintas**

Lampu lalu lintas merupakan alat yang mengatur pergerakan lalu lintas di persimpangan jalan, melalui pemisah waktu berbagai arah pergerakan yang saling berpotongan. Lampu lalu lintas dapat memberikan keuntungan bagi peningkatan keamanan lalu lintas, mengurangi kemacetan dan memberikan keamanan bagi pengguna penyeberang jalan. Adapun urutan dan maksud nyala lampu sebagai berikut:

1. Nyala merah : waktu yang diperuntukkan bagi pergerakan kendaraan untuk berhenti dan diperuntukkan bagi pejalan kaki untuk menyeberang jalan.
2. Nyala kuning: waktu yang diperlukan untuk memberikan peringatan kepada pengemudi bahwa nyala lampu akan berganti.
3. Nyala hijau: waktu yang di peruntukkan bagi pengendara untuk melewati persimpangan. [1]

Beberapa fungsi operasi lampu lalu lintas dapat dibedakan menjadi tiga jenis yaitu

1. *Traffic control signal*
2. *Pedestrian signal*
3. *Special traffic signal*

Tujuan diterapkannya pengaturan lampu lalu lintas adalah

1. Menciptakan pergerakan-pergerakan dan hak berjalan secara bergantian dan teratur, sehingga meningkatkan daya dukung pertemuan jalan dalam melayani arus lalu lintas.
2. Hirarki rute bisa dilaksanakan : rute utama diusahakan untuk mengalami kelambatan (*delay*) minimal.
3. Mengurangi terjadinya kecelakaan dan kelambatan lalu lintas.
4. Memberikan mekanisme pengaturan lalu lintas yang lebih efektif dan murah dibandingkan pengaturan manual.[1]

**2.2 Perhitungan Waktu Sinyal Lampu Lalu Lintas**

Tahapan perhitungan waktu sinyal lampu lalu lintas sebagai berikut :

**a. Trough Car Unit (TCU) ke Satuan Mobil Penumpang (SMP)**

Nilai equivalensi untuk berbagai jenis kendaraan bermotor dan pergerakan pada persimpangan terhadap mobil penumpang dihitung dalam satuan TCU. Untuk mengetahui jumlah tingkat kedatangan pada phase I, dapat dihitung dengan rumus :

$$Q = (MC \times 0.4) + (LV) + (HV \times 1.3) + (UM)$$

Dengan merupakan volume lalu lintas dengan satuan smp/jam.

**b. Saturation Flow (S)**

*Saturation flow* adalah volume lalu lintas maksimal (SMP) yang dapat melewati satu

pendekatan persimpangan pada nyala hijau, dimana iring-iringan kendaraan berjalan terus. *Saturation flow* dapat dipengaruhi berbagai faktor diantaranya adalah lebar jalan kelandaiannya serta kondisi lingkungan.

Harga *saturation flow* dapat dihitung dengan rumus: [1]

$$S = \frac{525}{W}$$

**c. Penentuan Fase dan Waktu Sinyal**

Pada analisis yang dilakukan bagi keperluan perancangan, waktu antara hijau berikut (kuning + merah semua) dapat dianggap sebagai nilai normal adalah seperti terlihat pada tabel 1.

**Tabel 1.** Nilai normal waktu antar hijau.

Ukuran Simpangan	Lebar jalan rata-rata	Nilai normal waktu antar hijau
Kecil	6 – 9 m	4 detik/fase
Sedang	10 – 14 m	5 detik/fase
Besar	≥ 15 m	≥ 6 detik/fase

**d. Rasio Arus / Rasio Arus Jenuh (Yi)**

Rasio arus jenuh adalah perbandingan antara tingkat kedatangan arus lalu lintas pada phase I, dengan arus jenuh pada phase I. Untuk menghitung harga Yi, digunakan persamaan :

$$Y_i = q_i / S_i$$

Dengan.

Yi = Rasio antar tingkat kedatangan dengan arus jenuh pada phase I.

qi = Tingkat kedatangan kendaraan pada phase I dalam SMP/jam.

Si = Arus jenuh pada phase I dalam SMP/jam.

**e. Kehilangan Waktu (*lost time*)**

Kehilangan waktu selama persimpangan tidak digunakan secara efektif untuk pergerakan lalu lintas atau waktu hilang dalam waktu siklus.

$$L = n (amber + r)$$

Dimana: L = *Lost time*

n = Jumlah phase

r = *All red*

**f. Waktu Siklus Optimum (*cycle time*)**

Waktu siklus optimum merupakan jumlah waktu sinyal untuk mencapai satu siklus, ditetapkan dalam detik.

$$C_o = \frac{(1.5 \times L) + 5}{(1 - Y)}$$

Dengan : Co = *Cycle time* (detik)

L = *Lost time* (detik)

Y = Yi maksimum (total)

**g. Waktu Phase Hijau (*green time*)**

Waktu phase hijau merupakan waktu yang diberikan dalam satu phase sinyal menunjukkan

warna hijau, dinyatakan dalam detik. Waktu hijau (*green time*) dapat dihitung dengan rumus:

$$g_i = \frac{Y_i}{Y} (C_o - L)$$

Dengan  $g_i$  = waktu hijau di phase i.  
 $Y_i$  = Harga Y terbesar pada phase i.  
 $C_o$  = Satu kali perputaran waktu  
 $Y$  =  $Y_i$  maksimum  
 $L$  = *Lost time* (dt)

#### h. Waktu Phase Kuning (*yellow time*)

Bertujuan untuk memberikan peringatan kepada pengemudi kendaraan, bahwa nyala lampu akan berubah menjadi merah, sehingga pengemudi dapat berhenti.

$$Y = t + \frac{V}{2a} + \frac{W + L}{V}$$

Dengan  
 $Y$  = Waktu phase kuning (dt)  
 $t$  = Waktu persepsi = 1 dt.  
 $a$  = Perlambatan kendaraan ( $15 \text{ ft/dt}^2 = 4.6 \text{ m/dt}^2$ )  
 $V$  = Kecepatan yang melewati persimpangan  
 $W$  = Lebar jalan  
 $L$  = Panjang kendaraan ( $20 \text{ ft} = 6.1 \text{ m}$ )

Sedangkan dalam Manual Kapasitas Jalan Indonesia 1997, panjang waktu kuning pada sinyal lalu lintas perkotaan Indonesia adalah minimal 3.0 detik.

### 2.3 Mikrokontroler AT89S52

Mikrokontroler AT89S52 merupakan salah satu jenis mikrokontroler keluarga MCS-51 yang memiliki flash memori. Perusahaan ATMEL memiliki lisensi dari Intel untuk mengembangkan mikrokontroler MCS-51. Salah satu type yang diperkenalkan adalah AT 89S52 yang kompetible dengan instruksi MSC-51.

Arsitektur AT89S52 secara umum diperlihatkan pada gambar 2.1, dimana chip ini sudah memiliki beberapa komponen yang pada masa lalu merupakan chip tersendiri, sub komponen tersebut adalah ;

1. CPU, singkatan dari *Central Prosessing Unit*, adalah otak atau unit pemroses pada suatu MCU
2. *Internal Oscilator* yang memungkinkan pemakai hanya menambahkan sebuah *quartz crystal* yang biasanya nilainya berkisar antara 6 Mhz hingga 24 Mhz. Meskipun demikian dari data sheet, MCU AT89S52 dapat bekerja dari frekuensi 0 hingga 24 Mhz.
3. *Interup Control*, suatu bagian yang mengatur prioritas interupsi dari luar atau dari dalam chip MCU.
4. Timer 1 dan Timer 2 adalah bagian yang dapat berfungsi sebagai pencacah pulsa masuk atau menentukan waktu (*Counter* atau *Timer*).
5. RAM adalah memori yang digunakan untuk menyimpan data sementara (data hilang bila catudaya padam)

6. Flash adalah jenis memori yang digunakan untuk menyimpan program dan data yang tetap, data pada memori ini tidak akan hilang bila catu daya padam.
7. Bus control digunakan untuk sinyal kendali akses data keluar system MCU.
8. I/O Port digunakan untuk menjembatani antara sistem MCU dengan dunia luarnya dengan cara paralel. I/O bisa sebagai output, misalnya menggerakkan relay, menyalakan lampu, dapat juga sebagai input misalnya mendeteksi saklar, menerima data dari ADC
9. Serial Port adalah penghubung ke dunia luar MCU dengan cara serial, dengan port serial, MCU dapat berkomunikasi dengan PC yang juga harus melalui serial port. [2],[6],[7].

### 2.4 RTC (*Real Time Clock*) DS 1307

Serial *Real Time Clock* (RTC) DS1307 merupakan *low-power* dan *full binary-coded-decimal* (BCD). Data dan alamat ditransfer berurutan secara serial melalui dua kabel dan *bidirectional bus*. Clock/Calendar menyediakan detik, menit, jam, hari, tanggal, bulan, dan informasi tahun. Akhir dari tanggal bulan secara otomatis disesuaikan selama sebulan paling sedikit 31 hari, mencakup koreksi untuk tahun kabisat. Jam beroperasi dalam format 12-jam atau 24-jam dengan AM/PM indikator. DS1307 mempunyai suatu pendeteksi gangguan daya dan secara otomatis memnyuplai tegangan dari baterai apabila  $V_{CC}$  lebih kecil dari  $V_{BAT}$ . [3]

### 2.5 EEPROM AT24C04

AT24C04 berfungsi untuk menyimpan data, yang memiliki memory sebesar  $512 \times 8$  (4K). Dimana AT24C04 melakukan komunikasi secara I2C dan data transfer protokol dua arah. Kelebihan dari EEPROM ini adalah memiliki pin *Write Protect* yang digunakan untuk melindungi data pada hardware.

Pengoperasian dari AT24C04 sebagai berikut :

1. **Clock dan Data Transmision** : pin SDA secara normal *pull high* dengan *device* eksternal. Data pada pin SDA bisa dirubah hanya selama SCL periode waktu *low* . Data berubah setelah SCL periode *high* akan menandai adanya suatu kondisi start atau stop.
2. **Kondisi Start** : Saat transisi *high* ke *low* pada SDA dengan SCL *high* adalah suatu kondisi start yang harus mendahului suatu perintah.
3. **Kondisi Stop** : Saat transisi *low* ke *high* pada SDA dengan SCL *high* adalah suatu kondisi stop. Setelah suatu urutan dibaca, perintah stop akan membuat EEPROM pada mode standby.
4. **Acknowledge** : Semua alamat dan data word dikirimkan secara serial ke EEPROM dalam 8-bit word. EEPROM mengirimkan bit 0 *Acknowledge* untuk penerimaan masing-masing word setelah bit ke 9 clock cycle.

5. **Standby Mode** : AT24C04 memiliki low-power, standby mode pada saat : *power-up* dan setelah penerimaan bit stop dan penyelesaian operasi internal .
6. **Memory Reset** : Setelah interupsi protokol, power loss dan sistem reset, ada dua jalur yang direset sesuai dengan langkah-langkah berikut ini :
  1. Clock lebih dari 9 siklus.
  2. Melihat SDA *high* pada setiap siklus saat SCL *high*.
  3. Membuat kondisi start.[4]

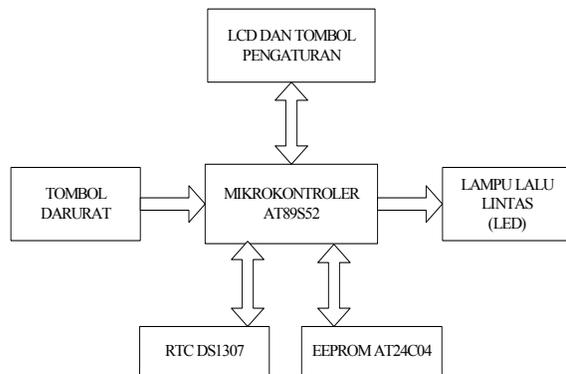
**2.6 LCD Karakter 16x2 (HD44780)**

Modul LCD karakter dapat dengan mudah dihubungkan dengan mikrokontroler seperti AT89S52. LCD ini mempunyai lebar display 2 baris 16 kolom atau biasa disebut sebagai LCD Character 2x16.[8]

**2.7 Metode Perancangan**

Sebelum melakukan penelitian, terlebih dahulu dilakukan kajian-kajian yang berkaitan dengan perancangan perangkat keras untuk kontrol lampu lalu lintas.

Berikut ini merupakan diagram blok keseluruhan simulator sistem kontrol lampu lalu lintas. Diagram blok ini terdiri dari 6 unit seperti gambar 1.



**Gambar 1.** Diagram blok simulator sistem kontrol lampu lalu lintas

Dari diagram blok gambar 1 dapat dijelaskan fungsi masing-masing blok sebagai berikut :

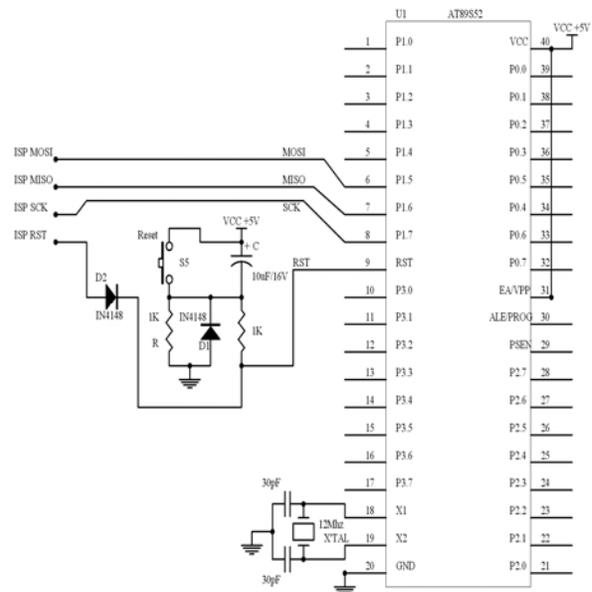
1. Tombol Darurat terdiri dari 4 buah *toggle switch*, dimana masing-masing *switch* digunakan untuk mengatur lampu lalu lintas secara manual.
2. Mikrokontroler AT89S52 digunakan untuk menyimpan program yang dibuat dan mengeluarkannya sebagai out put.
3. RTC DS1307 dan EEPROM AT24C04 digunakan sebagai penyedia waktu dan penyimpan data.

4. LCD dan tombol pengaturan digunakan untuk menampilkan dan merubah data waktu yang akan diubah sesuai dengan yang diinginkan.
5. Lampu lalu lintas yang digunakan berupa LED (merah, kuning dan hijau) yang digunakan sebagai simulasi tampilan kontrol lampu lalu lintas.

Dalam perancangan rangkaian lampu lalu lintas digunakan asumsi sebuah persimpangan empat (4) untuk dibuat sistem dan rangkaian kontrolnya. Masing-masing blok atau unit dirancang, direalisasikan dan diuji untuk mendapatkan hasil sesuai dengan yang dirancang atau diinginkan. Jika hasilnya tidak sesuai diadakan perubahan dan pengujian lagi sampai hasil pengujian ini sesuai dengan yang diinginkan. Setelah itu baru diuji secara keseluruhan sampai didapat hasil yang diinginkan.

**2.7.1 Mikrokontroler AT89S52**

Mikrokontroler yang digunakan tipe AT89S52, dalam perancangan rangkaian dan nilai komponen pendukungnya disesuaikan dengan data *sheet* minimum sistem DT-51. Tegangan  $V_{CC}$  yang digunakan sebesar 5 V, kristal yang digunakan 12Mhz sebagai pembangkit frekuensi osilator dan kapasitor C1 dan C2 yang digunakan 30pF. Untuk komunikasi ke ISP programming, RST dihubungkan resistor 1K, dioda IN4148 , switch dan kapasitor 10µF/16V ke  $V_{CC}$  untuk memberi logika "1" yang digunakan untuk mereset mikrokontroler. Serta sebagai pembatas tegangan antara RST pada mikrokontroler dengan RST pada ISP programming. Untuk gambar rangkaiannya seperti gambar 2.

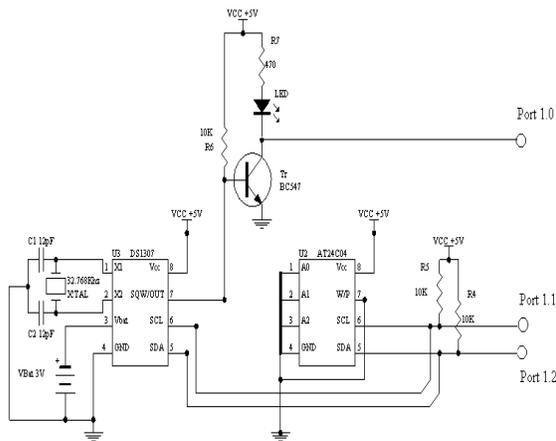


**Gambar 2.** Mikrokontroler AT89S52

**2.7.2 Rangkaian RTC DS1307 dan EEPROM AT24C04**

RTC DS1307 adalah dalam bentuk IC 8 pin dimana pada DS1307 menggunakan kristal 32.768Khz sebagai pembangkit frekuensi osilator dan kapasitansi beban sebesar 12.5pF sesuai dengan data sheetnya. Untuk kapasitansi bebannya, dalam perancangan ini digunakan 12pF. Tegangan  $V_{CC}$  yang digunakan 5V dan  $V_{BAT}$  nya sebesar 3V DC. RTC dihubungkan secara serial dengan mikrokontroler melalui Port1.1 untuk SCL, Port1.2 untuk SDA dan pin SQW/OUT dihubungkan ke Port1.0. Serial SDA, SCL dan SQW/OUT memerlukan *pull-up* eksternal resistor, dalam rancangan ini digunakan sebesar 10K sebagai pembatas arusnya. Karena output dari SQW/OUT sebesar 0.4 V untuk memberi masukan "1" atau *high* pada mikrokontroler antara tegangan 2 sampai dengan 5 V. Maka outputnya harus dikuatkan untuk memperoleh logika "1" sebagai masukan ke mikrokontroler. Transistor BC547 tipe NPN digunakan sebagai penguatnya dimana untuk tahanan basis digunakan sebesar 10K sesuai dengan *pull-up* eksternal resistornya. Untuk tahanan kolektor yang digunakan sebesar 470 Ohm sekaligus sebagai pembatas arus untuk LED.

Sedangkan untuk EEPROM AT24C04 pin A0, A1, A2, GND dan W/P dihubungkan ke ground. Sama dengan RTC komunikasi serial SCL dan SDA dihubungkan pada Port1.1 dan Port 1.2 mikrokontroler serta  $V_{CC}$  yang digunakan sebesar 5 V. Gambar 3 berikut adalah rangkaian RTC dan EEPROM :



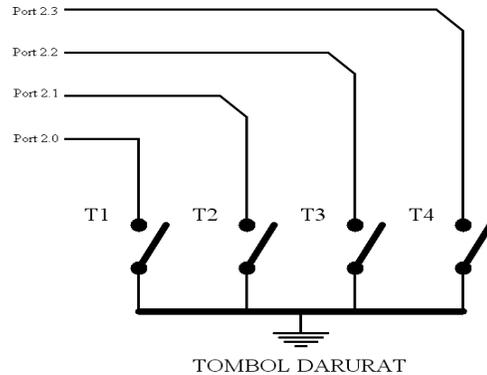
Gambar 3. Rangkaian RTC dan EEPROM

**2.7.3 Tombol Darurat**

Tombol darurat ini akan digunakan untuk mengatur lampu lalu lintas secara manual sesuai dengan arah persimpangan yang akan diatur nyala lampunya. Masing-masing *toggle switch* dihubungkan ke Port 2.0, 2.1, 2.2, 2.3 pada

mikrokontroler dan ke ground. Tombol darurat ini terdiri dari 4 yaitu :

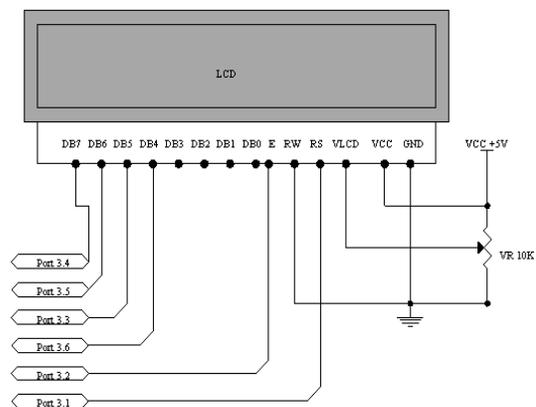
1. Tombol 1: untuk menyalakan lampu arah utara hijau, yang lainnya merah
2. Tombol 2 : untuk menyalakan lampu arah timur hijau, yang lainnya merah
3. Tombol 3: untuk menyalakan lampu arah selatan hijau, yang lainnya merah
4. Tombol 4 : untuk menyalakan lampu arah barat hijau, yang lainnya merah



Gambar 4. Persimpangan 4 dan tombol darurat

**2.7.4 LCD dan Tombol Pengaturan**

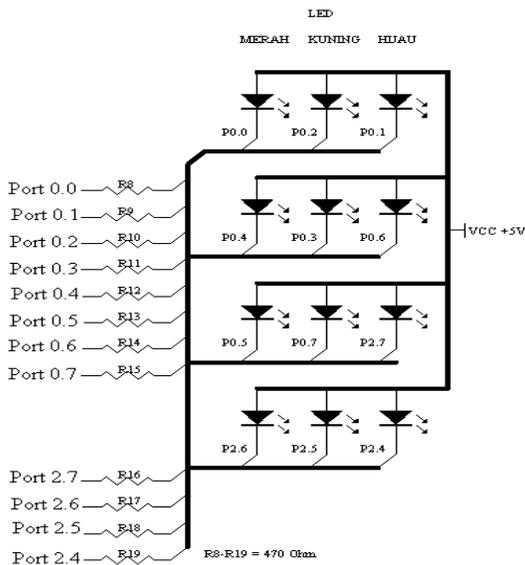
LCD yang digunakan dalam hal ini adalah LCD 16x2 (HD44780) dan tombol pengaturan menggunakan 4 buah *switch* yang dihubungkan pada Port P1.3, P1.4, P1.5 dan P1.6 difungsikan sesuai dengan perintah pada program yang dibuat. Pada LCD pin-pin yang digunakan adalah DB7, DB6, DB5, DB4, E, RS, VLCD, GND dan VCC yang akan dihubungkan ke port mikrokontroler yang telah ditentukan sesuai dengan gambar 3.6. Variabel resistor digunakan 10K untuk mengatur kontras pada LCD. Gambar rangkaian dapat dilihat pada gambar 5.[8]



Gambar 5. LCD karakter 16x2

**2.7.5 Lampu lalu lintas**

Untuk lampu lalu lintas digunakan LED untuk menampilkan simulasi nyala lampu lalu lintas, dimana warna LED yang digunakan adalah merah, kuning dan hijau, sebanyak 12 LED. Port mikrokontroler yang digunakan adalah Port 0.0 sampai Port 0.7 dan Port 2.4 sampai Port 2.7 dan digunakan resistor 470 ohm sebagai pembatas arusnya. Tegangan yang digunakan sebesar 5 V. Jadi arusnya sebesar 10.64 mA sehingga cukup untuk menghidupkan LED (sekitar 10mA). Untuk lampu (LED) penyeberang jalan diparalelkan dengan LED merah dan hijau pada lampu lalu lintas. Gambar rangkaiannya bisa dilihat pada gambar 6.



Gambar 6. Rangkaian LED

**2.8 Perancangan Program**

Program yang dibuat menggunakan bahasa basic yaitu menggunakan BASCOM-8051. Pada perangkat lunak ini sudah disediakan rutin-rutin untuk komunikasi antara mikrokontroler dengan RTC DS1307, EEPROM AT24C04 dan LCD sehingga pemrograman menjadi lebih singkat dan mudah. Setelah program dibuat dan dikompilasi, dihasilkan file dengan ekstension HEX yang selanjutnya dengan program IspPgm bisa didownload ke memori dari mikrokontroler untuk mengendalikan perangkat keras dari sistem yang dibuat.

Program utama dari simulator kontrol lampu lalu lintas yang dibuat seperti program berikut.

```

*****
'Program utama
*****
For N = 1 To 44
  Readeeprom Value , N
  'baca semua nilai pada EEPROM
  Value = Makedec(value)
  Rrr(n) = Value

```

```

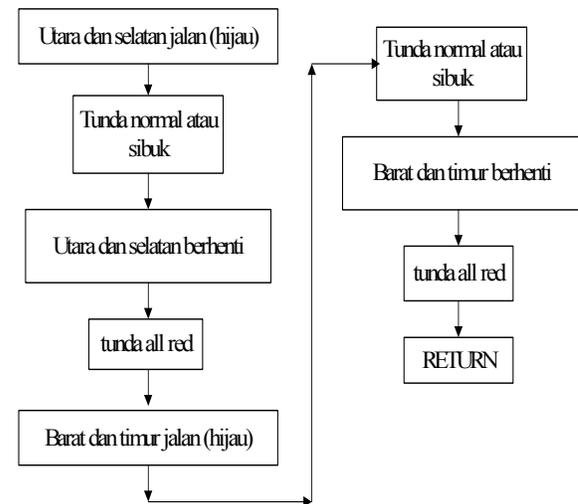
Next
Dlay = Rrr(1)
' baca nilai Dlay (all red) pada EEPROM
_mode = Rrr(3)
' baca nilai mode pada EEPROM

Cls
Main:
Do
Gosub Cek
If Weekday => 5 Then
' ceking hari

Comp_time_eep1 = Rrr(5) * 10000
' baca nilai jam pada EEPROM
Comp_time_eep2 = Rrr(6) * 10000
If Comp_time_eep1 < Comp_time_rtc And 'bandingkan
  nilai pada eeprom dengan RTC
Comp_time_eep2 > Comp_time_rtc Then
' jika sama , berlanjut ke sub program malam
Gosub Night
Else
' jika tidak sama, lanjut ke nilai tunda normal
Ta = Rrr(11) 'Ta = nilai tunda normal arah barat
Tb = Rrr(12) 'Tb = nilai tunda normal arah timur
Tc = Rrr(13) 'Tc = nilai tunda normal arah utara
Td = Rrr(14) 'Td = nilai tunda normal arah selatan

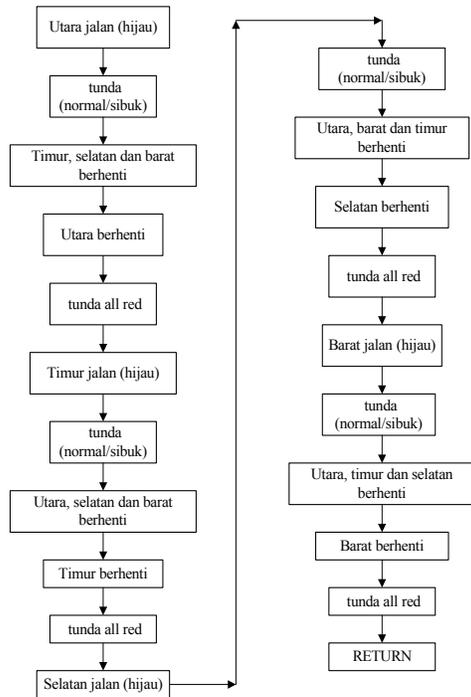
```

Untuk Sub program nyala lampu terdapat dua mode yaitu mode 1 untuk dua arah jalan, mode 2 untuk jalan satu-satu dan mode 3 untuk simpang tiga (3). Alur programnya untuk mode 1 dapat dilihat seperti gambar 7.



Gambar 7. Alur program dari sub program nyala lampu model

Untuk mode 2 , alur programnya dapat dilihat pada gambar 8.

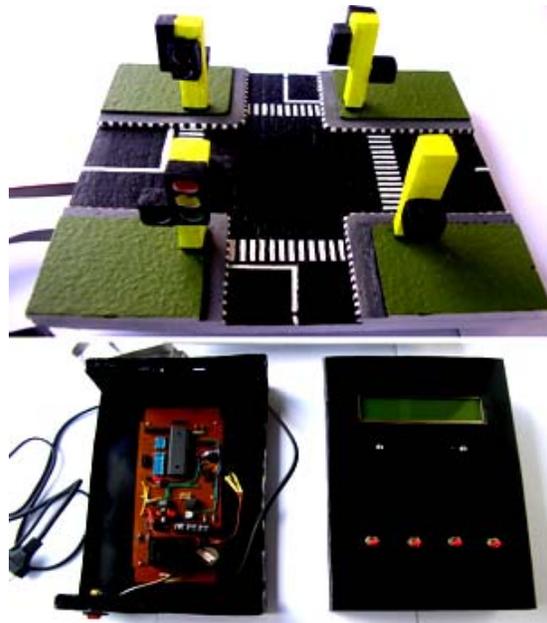


Gambar 8. Alur program dari sub program nyala lampu mode 2

### 3 HASIL DAN PEMBAHASAN

#### 3.1 Realisasi Peralatan

Adapun tampilan perangkat keras simulator pengatur lampu lalu lintas berbasis mikrokontroler AT89S52 tampak seperti gambar 9.



Gambar 9. Realisasi simulator Sistem Kontrol Lampu Lalu Lintas Berbasis Mikrokontroler

Dari gambar 8, dapat dijelaskan fungsi-fungsi dari sistem kontrol yang dibuat sebagai berikut:

1. Tombol darurat digunakan untuk mengatur lampu lalu lintas secara manual sesuai dengan yang diinginkan, adapun fungsi dari masing-masing tombol tersebut adalah :
  - a. Tombol 1 untuk membuat lampu arah utara hijau dan lainnya merah.
  - b. Tombol 2 untuk membuat lampu arah timur hijau dan lainnya merah.
  - c. Tombol 3 untuk membuat lampu arah selatan hijau dan lainnya merah.
  - d. Tombol 4 untuk membuat lampu arah barat hijau dan lainnya merah.
2. LCD digunakan untuk menampilkan data hari, tanggal, bulan, tahun, detik, menit, jam dan waktu tunda nyala lampu lalu lintas (*delay*).
3. Tombol pengaturan digunakan untuk merubah atau mengisi data waktu tunda nyala lampu lalu lintas, serta mengedit tanggal dan jam.
4. Mikrokontroler AT89S52 digunakan sebagai penyimpan program yang dibuat dan mengeluarkannya sebagai output.
5. RTC DS1307 digunakan sebagai penyedia waktu yang berisikan data detik, menit, jam, tanggal, hari, bulan dan tahun.
6. EEPROM AT24C04 digunakan sebagai penyimpan data yang akan diisi atau diubah, berupa data waktu tunda nyala lampu lalu lintas, jam serta tanggal.
7. Lampu lalu lintas yang digunakan adalah LED sebagai simulasi tampilan

#### 3.2 Hasil Perhitungan Keseluruhan

Dari perhitungan waktu nyala lampu lalu lintas untuk masing-masing phase dan kondisi yang dilakukan, maka dapat dilihat hasil perhitungan secara keseluruhan nyala lampu lalu lintas pada tabel 2.

Tabel 2a. Hasil perhitungan keseluruhan nyala lampu lalu lintas (senin-kamis)

Kondisi	Cycle Time (Siklus Optimum)	Waktu Nyala Lampu		
		Hijau		Kuning
		Phase I	Phase II	
Padat 1	45 dtk	15 dtk	20 dtk	3 detik untuk semua pahse dan kondisi
Padat 2	38 dtk	9 dtk	19 dtk	
Padat 3	39 dtk	10 dtk	19 dtk	
Normal	30 dtk	7 dtk	13 dtk	

**Tabel 2b.** Hasil perhitungan keseluruhan nyala lampu lalu lintas (jumat-minggu)

Kondisi	Cycle Time (Siklus Optimum)	Waktu Nyala Lampu		
		Hijau		Kuning
		Phase I	Phase II	
Padat 1	37 dtk	13 dtk	14 dtk	3 detik untuk semua pahse dan kondisi
Padat 2	39 dtk	9 dtk	20 dtk	
Padat 3	38 dtk	8 dtk	20 dtk	
Normal	28 dtk	8 dtk	10	

Dari hasil perhitungan dan pengujian pada perangkat simulator pengatur lampu lalu lintas berdasarkan waktu dan kepadatan kendaraan berbasis mikrokontroler AT89S52 didapatkan bahwa simulator sudah bekerja sesuai dengan hasil rancangan.

**4 KESIMPULAN**

Dari hasil pengujian yang dilakukan dapat disimpulkan:

1. Simulator sistem lampu lalu lintas dapat dikendalikan sesuai dengan kondisi normal, padat (sibuk) maupun lengang, menggunakan mikrokontroler AT89S52.
2. Pengaturan lampu lalu lintas dapat dilakukan secara periodis berdasarkan jam dan hari yang sebenarnya dengan input waktu dari RTC (*Real Time Clock*) DS 1307.
3. Hasil perhitungan nyala lampu lalu lintas untuk hari senin-kamis yang diperoleh dari persimpangan Jalan Waribang-WR.Supratman Denpasar adalah
  - a. Kondisi padat 1 : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 45 detik dan besar nilai nyala lampu hijaunya yaitu 15 detik untuk phase I dan 20 detik untuk phase II.
  - b. Kondisi padat 2 : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 38 detik dan besar nilai nyala lampu hijaunya yaitu 9 detik untuk phase I dan 19 detik untuk phase II.
  - c. Kondisi padat 3 : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 39 detik dan besar nilai nyala lampu hijaunya yaitu 10 detik untuk phase I dan 19 detik untuk phase II.
  - d. Kondisi normal : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 30 detik dan besar nilai nyala lampu hijaunya yaitu 7 detik untuk phase I dan 13 detik untuk phase II.
4. Hasil perhitungan nyala lampu lalu lintas untuk hari jumat-minggu yang diperoleh dari

persimpangan Jalan Waribang-WR.Supratman Denpasar adalah sebagai berikut:

- a. Kondisi padat 1 : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 37 detik dan besar nilai nyala lampu hijaunya yaitu 13 detik untuk phase I dan 14 detik untuk phase II.
- b. Kondisi padat 2 : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 39 detik dan besar nilai nyala lampu hijaunya yaitu 9 detik untuk phase I dan 20 detik untuk phase II.
- c. Kondisi padat 3 : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 38 detik dan besar nilai nyala lampu hijaunya yaitu 8 detik untuk phase I dan 20 detik untuk phase II.
- d. Kondisi normal : siklus optimum atau *cycle time* yang diperoleh pada phase I dan II sebesar 28 detik dan besar nilai nyala lampu hijaunya yaitu 8 detik untuk phase I dan 10 detik untuk phase II.

**5 DAFTAR PUSTAKA**

- [1] Malkhamah.S, 1996, *Survey Lampu Lalu Lintas dan Pengantar Manajemen Lalu Lintas*, Yogyakarta : KMTS FT UGM.
- [2] Wahyudi.D, 2007, *Belajar Mudah Mikrokontroler AT89S52 dengan Bahasa Basic Menggunakan BASCOM-8051*, Yogyakarta, ANDI.
- [3] *DS1307 64x8 Serial Real Time Clock*, <http://maxim-ic.com>. 14/2/2008.
- [4] *Real-Time-Clock Selection and Optimization*<http://maxim-ic.com>.
- [5] <http://www.atmel.com>
- [6] Putra,A.E., 2002, *Belajar Mikrokontroler AT89C51/AT98C52 (Teori dan Aplikasi)*. Yogyakarta : Gava Media.
- [7] Sudjadi, 2005, *Teori dan Aplikasi Mikrokontroler*, Yogyakarta : Graha Ilmu.
- [8] *Tutorial Microntroller MSC-51 Atmel ISP*, <http://myTutorialCafe.com>.
- [9] <http://www.datasheetcatalog.com>