

**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA**



SKRIPSI

**PERANCANGAN DAN PEMBUATAN
PEMINDAH JALUR REL KERETA API SEMI OTOMATIS
DI STASIUN MENGGUNAKAN MIKROKONTROLER
R5F21134FP**

**Disusun Oleh:
TJIA SINDARTO SETIADI
NIM : 02.17.059**

MARET 2007

LEMBAR PERSETUJUAN

**PERANCANGAN DAN PEMBUATAN
PEMINDAH JALUR REL KERETA API SEMI OTOMATIS
DI STASIUN MENGGUNAKAN MIKROKONTROLER**

R5F21134FP

SKRIPSI

Diajukan Guna Memenuhi Salah Satu Syarat Untuk Memperoleh Gelar Sarjana
Teknik Pada Jurusan Teknik Elektro S-1 Konsentrasi Elektronika

Disusun Oleh :

TJIA SINDARTO SETIADI

NIM : 02.17.059



Mengetahui

Ketua Jurusan Teknik Elektro S-1

(Ir. F. Yudi Limpraptono, MT)
NIP. 1039500274

Diperiksa dan Disetujui

Dosen Pembimbing

(Ir. Sidik Noertjahjono, MT)
NIP. 1028700167

**KONSENTRASI ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO S - 1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG**



**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA**

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : Tjia Sindarto Setiadi
NIM : 0217059
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : PERANCANGAN DAN PEMBUATAN PEMINDAH
JALUR REL KERETA API SEMI OTOMATIS DI
STASIUN MENGGUNAKAN MIKROKONTROLER
R5F21134FP

Dipertahankan dihadapan Tim Penguji Skripsi Jenjang Strata Satu (S-1) pada:

Hari : Sabtu
Tanggal : 17 Maret 2007
Dengan Nilai : 89 (A) *fy*



Panitia Ujian Skripsi

Ketua

(Ir. Mochtar Asroni, MSME)
NIP.Y. 1018100036

Sekretaris

(Ir.F.Yudi Limpraptono, MT)
NIP.Y. 1039500274

Penguji Pertama

Anggota Penguji

Penguji Kedua

(Dr. Cahyo Crysdian, MSc.)

(I Komang Somawirata, ST., MT.)

ABSTRAKSI
PERANCANGAN DAN PEMBUATAN PEMINDAH
JALUR REL KERETA API SEMI OTOMATIS DI STASIUN
MENGGUNAKAN MIKROKONTROLER R5F21134FP

(Tjia Sindarto Setiadi, 0217059, Teknik Elektro/Elektronika S-1)

(Dosen Pembimbing : Ir. Sidik Noertjahjono, MT)

Kata Kunci : Jalur Rel, Kereta Api, Semi Otomatis

Pada skripsi ini dirancang sebuah sistem pemindah jalur rel kereta api dengan menitikberatkan permasalahan yang timbul pada pendeteksi kedatangan kereta api, menginformasikan kedatangan kereta api kepada petugas dan calon penumpang, perpindahan jalur rel kereta api serta mendeteksi jalur kereta api yang sedang digunakan dan jalur kereta api yang kosong.

Kerja dari sistem pemindah jalur rel kereta api ini adalah mendeteksi kedatangan kereta api kemudian menginformasikan kedatangan kepada petugas serta calon penumpang. Setelah kereta api berhenti pada jalur yang telah disiapkan petugas maka indikator menampilkan jalur mana yang sedang digunakan oleh kereta api. Kereta api yang meninggalkan stasiun akan terdeteksi kemudian indikator akan menampilkan jalur kereta api yang kosong.

Sistem pemindah jalur rel kereta api ini dapat bekerja sesuai dengan permasalahan yang timbul pada informasi kedatangan kereta api, pemindahan jalur rel kereta api serta jalur kereta api yang digunakan di stasiun.

KATA PENGANTAR

Segala puji syukur hanya milik Tuhan YME, karena atas bimbingan dan rahmat-Nya, sehingga penyusunan dapat menyelesaikan Laporan Skripsi yang berjudul “Perancangan Dan Pembuatan Pemindah Jalur Rel Kereta Api Semi Otomatis Di Stasiun Menggunakan Mikrokontroler R5F21134FP” ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika Institut Teknologi Nasional Malang dan untuk mencapai gelar Sarjana Teknik.

Keberhasilan penyelesaian Laporan Skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Bapak Prof. DR. Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1.
3. Bapak Ir. Sidik Noertjahjono, MT selaku Dosen Pembimbing.
4. Seluruh Dosen Institut Teknologi Nasional Malang, yang telah membantu.
5. Ayah dan Ibu serta saudara-saudara kami yang telah memberikan do'a restu, dorongan dan semangat.
6. Rekan-rekan Elka 2 serta Instruktur di Laboratorium.

7. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Kami menyadari bahwa dalam penyusunan Laporan ini masih ada lebih dan kurangnya, untuk itu saran dan kritik yang akan Kami terima dengan senang hati dan penghargaan yang sebesar-besarnya, demi membangun hasil laporan yang lebih baik.

Malang, Maret 2007

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	i
ABSTRAKSI.....	ii
KATA PENGANTAR.....	iii
DAFTAR ISI.....	v
DAFTAR GAMBAR	x
DAFTAR TABEL.....	xiv
BAB I PENDAHULUAN.....	1
1.1.Latar Belakang	1
1.2.Rumusan Masalah	2
1.3.Batasan Masalah	2
1.4.Tujuan	3
1.5.Sistematika Penulisan.....	3
BAB II TEORI DASAR.....	5
2.1. Pendahuluan.....	5
2.2. Keypad.....	6
2.3. Limit Switch	6
2.4. LED Infra Merah.....	7
2.5. Photodiode.....	8
2.6. Transistor Bipolar.....	10
2.6.1. Arus Bias.....	10

2.6.2. Arus Emiter	11
2.6.3. Alpha (α).....	11
2.6.4. Beta (β).....	12
2.6.5. Common Emiter (CE)	12
2.6.6. Kurva Base	13
2.6.7. Kurva Kolektor	14
2.6.8. Daerah Aktif.....	15
2.6.9. Daerah Saturasi.....	16
2.6.10. Daerah Cut-Off.....	16
2.6.11. Daerah Breakdown.....	17
2.7. Operational <i>Amplifier</i> (Op-Amp).....	17
2.8. <i>Schmitt Trigger</i>	19
2.9. <i>Buzzer</i>	21
2.10. <i>Relay</i>	21
2.11. Motor DC.....	23
2.12. IC <i>Information Storage Device</i> (ISD) 2560.....	27
2.13. Mikrokontroler AT89S51	32
2.13.1. Perangkat Keras Mikrokontroler AT89S51	33
2.13.2. Arsitektur AT89S51	34
2.13.3. Konfigurasi Pin-pin Mikrokontroler	35
2.13.4. Organisasi Memori.....	38
2.13.5. SFR (<i>Special Function Register</i>)	39
2.13.6. Metode Pengalamatan	44

2.14. Mikrokontroler Renesas R8C/13 Tiny (R5F21134FP)	44
2.14.1. Spesifikasi R5F21134FP	45
2.14.2. Kelebihan Kunci R8C/Tiny	46
2.14.3. Konfigurasi Pin R8C R5F21134FP	47
2.14.4. <i>Peripheral</i> R8C R5F21134FP	50
2.15. Komunikasi RS-232	53
2.15.1. IC MAX 232	53
2.16. LCD (<i>Liquid Crystal Display</i>) M1632	54
2.16.1. Sinyal <i>Interface</i> M1632	55
2.16.2. <i>Interface</i> Ke MCS-51	57
2.16.3. Mengatur Tampilan M1632	60
BAB III PERANCANGAN DAN PEMBUATAN ALAT	6
3.1. Blok Diagram Sistem	67
3.2. Perancangan Perangkat Keras	71
3.2.1. Detektor Jalur Kereta Api	71
3.2.2. Detektor Kereta Api dan Motor Limiter	73
3.2.3. <i>Keypad</i>	74
3.2.4. Rangkaian <i>Encoder Keypad</i>	74
3.2.5. Rangkaian Indikator	75
3.2.5.1. Rangkaian Indikator Jalur Rel	75
3.2.5.2. Rangkaian Indikator Posisi Wessel	76
3.2.6. Rangkaian <i>Driver Buzzer</i>	77
3.2.7. <i>Driver</i> Motor	79

3.2.8. Rangkaian IC MAX 232	81
3.2.9. Rangkaian <i>Information Storage Device</i> (ISD2560).....	82
3.2.10. Rangkaian LCD	85
3.2.11. Rangkaian <i>Reset</i> AT89S51.....	86
3.2.12. Rangkaian <i>Clock</i> AT89S51	87
3.2.13. <i>Minimum System</i> Mikrokontroler Master Renesas R5F21134FP.....	88
3.2.14. <i>Minimum System</i> Mikrokontroler <i>Slave 1</i> AT89S51	90
3.2.15. <i>Minimum System</i> Mikrokontroler <i>Slave 2</i> AT89S51	91
3.2.16. Komunikasi Paralel	92
3.2.16.1. Mikrokontroler Renesas dan <i>Slave 1</i> AT89S51.....	92
3.2.16.1. Mikrokontroler <i>Slave 1</i> AT89S51 dan <i>Slave 2</i> AT89S51.....	93
3.3. Perancangan Perangkat Lunak (<i>Software</i>).....	94
3.3.1. <i>Flow Chart</i> Keseluruhan Sistem.....	95
3.3.2. <i>Flow Chart Software</i> Pada Mikrokontroler.....	97
BAB IV PENGUJIAN ALAT	101
4.1. Rangkaian Detektor Kereta Api.....	101
4.2. Rangkaian Detektor Jalur Kereta Api	103
4.3. Rangkaian <i>Encoder Keypad</i>	104
4.4. Rangkaian <i>Driver</i> Motor	107
4.5. Rangkaian IC MAX 232.....	108

BAB V PENUTUP	111
5.1 Kesimpulan	111
5.2 Saran	111

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

2.1. Keypad 2×4.....	6
2.2. Limit Switch.....	6
2.3. Simbol LED Infra Merah	7
2.4. Simbol <i>Photodiode</i>	8
2.5. Arus Emitor	11
2.6. Rangkaian CE.....	12
2.7. Kurva I_R-V_{BE}	13
2.8. Kurva Kolektor	15
2.9. Rangkaian <i>Driver</i> LED	16
2.10. Penguat Tak Membalik (<i>Non Inverting</i>)	18
2.11. Konfigurasi Pin LM324	18
2.12. Rangkaian <i>Voltage Follower</i> Atau <i>Buffer</i>	19
2.13. Rangkaian Dasar <i>Schmitt Trigger</i>	20
2.14. Simbol <i>Buzzer</i>	21
2.15. Cara Kerja <i>Relay</i>	22
2.16. <i>Relay SPST</i>	22
2.17. <i>Relay SPDT</i>	22
2.18. <i>Relay DPDT</i>	23
2.19. Garis-Garis Medan Magnet Disekitar Arus Listrik Pada Kawat Lurus	23
2.20. Kaidah Tangan Kanan.....	24
2.21. Dasar Konstruksi Motor DC.....	24
2.22. Sebuah Motor DC	25

2.23. Arah Putaran Motor DC	26
2.24. Pin-pin IC ISD 2560	28
2.25. Blok Diagram Mikrokontroler AT89S51	33
2.26. Konfigurasi Pin-pin AT89S51	35
2.27. Rangkaian <i>Reset</i>	37
2.28. Rangkaian <i>Clock</i>	38
2.29. Memori Program.....	39
2.30. Register PSW Dalam Mikrokontroler AT89S51	42
2.31. Blok Diagram R8C/13 dan Peta <i>Peripheral</i> -nya	46
2.32. Konfigurasi Pin R8C R5F21134FP	47
2.33. Diagram Blok ADC	50
2.34. Konfigurasi Pin IC MAX 232	53
2.35. Rangkaian Operasi MAX 232	54
2.36. Mengirim/Mengambil Data Ke/Dari M1632.....	56
2.37. Hubungan M1632 ke MCS-51	57
2.38. Rangkaian LCD M1632	62
3.1. Diagram Blok Pemindah Jalur Rel Kereta Api	68
3.2. Rangkaian Detektor Jalur Kereta Api.....	71
3.3. Rangkaian Detektor Kereta Api	73
3.4. Rangkaian <i>Keypad</i> 2×4	74
3.5. Rangkaian <i>Encoder Keypad</i>	75
3.6. Rangkaian Indikator Jalur Rel.....	76
3.7. Rangkaian Indikator Posisi Wessel.....	76

3.8. Rangkaian <i>Driver Buzzer</i>	78
3.9. Rangkaian <i>Driver Motor</i>	80
3.10. Rangkaian IC MAX 232	82
3.11. Rangkaian ISD 2560	83
3.12. Rangkaian <i>Liquid Crystal Display (LCD)</i>	85
3.13. Rangkaian <i>Reset</i>	86
3.14. Rangkaian <i>Clock</i>	88
3.15. <i>Minimum System</i> Mikrokontroler Renesas R8C Tiny R5F21134FP	88
3.16. <i>Minimum System</i> Mikrokontroler <i>Slave 1</i> AT89S51	90
3.17. <i>Minimum System</i> Mikrokontroler <i>Slave 2</i> AT89S51	91
3.18. Komunikasi Mikrokontroler Renesas Dengan <i>Slave 1</i>	93
3.19. Komunikasi Mikrokontroler <i>Slave 1</i> Dengan <i>Slave 2</i>	94
3.20. <i>Flow Chart</i> Kedatangan Kereta Api	95
3.21. <i>Flow Chart</i> Keberangkatan Kereta Api	96
3.22. <i>Flowchart</i> Mikrokontroler Master Renesas R8C/13	97
3.23. <i>Flowchart</i> Mikrokontroler <i>Slave 1</i> AT89S51.....	99
3.24. <i>Flowchart</i> Mikrokontroler <i>Slave 2</i> AT89S51.....	100
4.1. Pengujian Detektor Kereta Api.....	101
4.2. Foto Pengujian <i>Limit Switch</i>	102
4.3. Pengujian Rangkaian Detektor Jalur Kereta Api.....	103
4.4. Foto Pengujian Rangkaian Detektor Jalur Kereta Api.....	104
4.5. Diagram Blok Pengujian <i>Encoder Keypad</i>	104
4.6. Rangkaian Pengujian <i>Encoder Keypad</i>	105

4.7. Foto Pengujian Rangkaian <i>Encoder Keypad</i>	106
4.8. Rangkaian Pengujian <i>Driver Motor</i>	107
4.9. Foto Pengujian Rangkaian <i>Driver Motor</i>	108
4.10. Rangkaian Pengujian IC MAX 232.....	109
4.11. Foto Pengujian Rangkaian IC MAX 232.....	110

DAFTAR TABEL

2.1. Fungsi Khusus <i>Port 1</i>	36
2.2. Fungsi Khusus <i>Port 3</i>	36
2.3. <i>Special Function Register</i>	40
2.4. Konfigurasi Pin-pin dari R8C R5F21134FP	48
2.5. Mode-mode <i>Timer</i>	51
2.6. Fungsi Pin-pin LCD.....	63
3.1. Data Hasil Perekaman dan Alamatnya Pada <i>ISD 2560</i>	85
4.1. Hasil Pengujian <i>Limit Switch</i>	102
4.2. Hasil Pengujian Detektor Jalur Kereta Api.....	103
4.3. Hasil Pengujian <i>Encoder Keypad</i>	106
4.4. Hasil Pengujian Tegangan Rangkaian <i>Driver Motor</i>	107
4.5. Hasil Pengujian Arus Rangkaian <i>Driver Motor</i>	108
4.6. Hasil Pengujian Rangkaian IC <i>MAX 232</i>	109



BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan ilmu pengetahuan dan teknologi yang semakin cepat khususnya dalam bidang elektronika menyebabkan manusia tidak akan lepas dari penggunaan berbagai macam peralatan elektronika yang ada, baik itu penggunaan peralatan yang menggunakan perangkat keras maupun perangkat lunak elektronika. Penggunaan teknologi yang semakin modern dan canggih diharapkan semakin mempermudah dan meringankan manusia dalam menjalankan pekerjaannya serta mengurangi kesalahan yang diakibatkan kelalaian manusia.

Salah satu bidang teknologi yang mengalami perkembangan pesat adalah teknologi kontrol otomatis yang berkembang pesat seiring perkembangan jaman. Telah banyak berbagai kontrol otomatis yang dibuat dan diaplikasikan untuk keperluan industri, layanan umum maupun untuk penelitian.

Pada saat ini kedatangan serta pemindahan jalur rel kereta api di stasiun-stasiun dilakukan secara manual oleh petugas penjaga sehingga petugas harus selalu dalam keadaan siaga dalam menjalankan pekerjaannya. Kelalaian petugas dalam menjalankan pekerjaannya dapat mengakibatkan kecelakaan antar kereta api. Untuk menghindari hal tersebut maka dirancang alat yang mampu mendeteksi kedatangan kereta api dan mengontrol pemindahan jalur rel kereta api.

1.2. Rumusan Masalah

Dalam perancangan dan pembuatan pemindah jalur rel kereta api semi otomatis dapat dirumuskan beberapa masalah yang akan dibahas, yaitu :

- 1 Bagaimana merancang dan membuat pendeteksi kedatangan kereta api dan menginformasikan dalam bentuk suara terprogram dan indikator kedatangan.
- 2 Bagaimana merancang dan membuat alat kontrol otomatis untuk memindah jalur rel kereta api.
- 3 Bagaimana merancang dan membuat perangkat lunak atau *software* pada mikrokontroler yang mengendalikan semua kerja sistem ini agar dapat bekerja sesuai yang telah direncanakan.

1.3. Batasan Masalah

Agar pembahasan dari perancangan dan pembuatan pemindah jalur rel kereta api semi otomatis tidak meluas maka perlu adanya pembatasan permasalahan yang meliputi :

1. Perancangan dan pembuatan alat ini, hanya berupa prototipe.
 2. Perancangan dan pembuatan simulasi dibatasi untuk 4 jalur rel kereta api saja.
 3. Membahas detektor kereta api dan detektor jalur kereta api.
 4. Motor yang digunakan hanya untuk simulasi konstruksi alat yang dibuat saja.
 5. Tidak membahas mengenai kereta api (lokomotif) dan gerbongnya.
 6. Tidak membahas masalah mekanik.
-

7. Tidak membahas catu daya.

1.4. Tujuan

Tujuan dari perancangan dan pembuatan pemindah jalur rel kereta api semi otomatis ini adalah merancang dan membuat pemindah jalur rel kereta api, pendeteksi kedatangan kereta api serta menginformasikan kedatangan kereta api yang dikontrol oleh Mikrokontroler.

1.5. Sistematika Penulisan

Sistematika pembahasan dari skripsi ini terdiri dari pokok pembahasan yang saling berkaitan antara satu dengan yang lain, yaitu :

BAB I Pendahuluan

Pada bab ini dibahas tentang latar belakang permasalahan, rumusan masalah, batasan masalah, sistematika penulisan dari alat yang direncanakan.

BAB II Landasan Teori

Pada bab ini dibahas tentang teori-teori yang mendukung dalam perancangan dan pembuatan alat ini yang meliputi detektor, input tombol, pengkondisi sinyal, rangkaian *driver*, motor, LCD M1632, ISD 2560, dan Mikrokontroler.

BAB III Perancangan Dan Pembuatan Alat

Pada bab ini dibahas tentang perancangan dan pembuatan keseluruhan sistem perangkat keras (*hardware*) dan perangkat lunak (*software*).

BAB IV Pengujian Alat

Pada bab ini dibahas tentang proses serta hasil dari pengujian alat, yang didasarkan oleh pengukuran-pengukuran yang diperlukan.

BAB V Penutup

Pada bab ini akan disampaikan kesimpulan dari perancangan dan pembuatan sistem ini.



BAB II TEORI DASAR

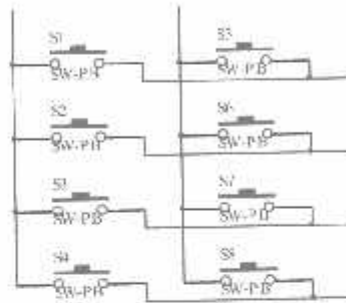
2.1. Pendahuluan

Pada bab ini akan dibahas mengenai teori penunjang dari peralatan yang direncanakan. Teori penunjang ini akan membahas tentang komponen dan peralatan pendukung pada alat yang dibuat. Pokok pembahasan pada bab ini adalah :

1. *Keypad*
2. *Limit Switch*
3. LED Infra Merah
4. *Photodiode*
5. Transistor Bipolar
6. *Operasional Amplifier (OP-AMP)*
7. *Schmitt Trigger*
8. *Buzzer*
9. *Relay*
10. Motor DC
11. *Information Storage Device (ISD) 2560*
12. Mikrokontroler AT89S51
13. Mikrokontroler Renesas R8C Tiny R5F21134FP
14. Komunikasi RS-232
15. *Liquid Crystal Display (LCD) M1632*

2.2. Keypad

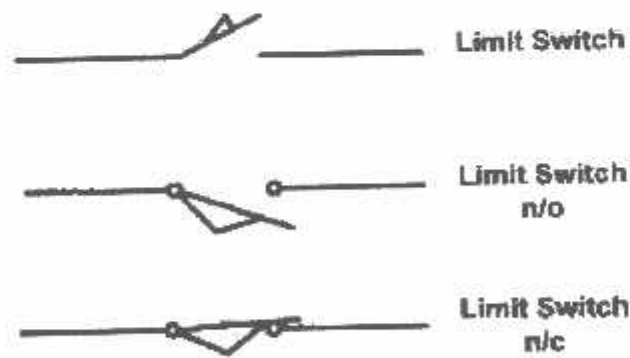
Keypad ini terdiri dari beberapa *switch* yang disusun secara matrik dimana jumlah dari *switch* tersebut adalah perkalian antara jumlah baris dan kolom. Rangkaian susunan *keypad* 2×4 dapat dilihat pada gambar dibawah ini :



Gambar 2.1. Keypad 2×4

2.3. Limit Switch

Limit switch merupakan salah satu komponen yang sangat banyak kegunaannya, antara lain sebagai sensor tekan untuk mendeteksi kereta api dan saklar pengaman dari putaran mekanik motor dc.

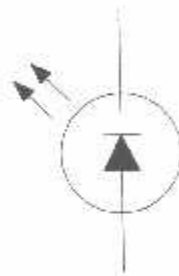


Gambar 2.2. Limit Switch

2.4. LED Infra Merah

LED infra merah digunakan untuk menghasilkan sinar infra merah. Prinsip kerja dari infra merah adalah pada waktu LED infra merah dibias *forward*, elektron dari pita konduksi melewati *junction* jatuh ke dalam *hole* pita valensi, sehingga elektron tersebut memancarkan energi. Pada dioda penyearah biasa, energi ini dipancarkan sebagai energi panas, sedangkan pada LED infra merah energi ini dipancarkan sebagai cahaya.

Simbol LED infra merah yang sering digunakan adalah :



Gambar 2.3. Simbol LED Infra Merah

LED infra merah merupakan *pin function* yang memancarkan radiasi infra merah yang tidak kelihatan oleh mata kita. Apabila pada anoda diberi tegangan dan katoda ke ground maka LED menjadi ON dan arus akan mengalir dari anoda ke katoda. Pada reaksi semikonduktor, suatu dioda akan terjadi perpindahan elektron dari tipe N ke tipe P. Proses rekombinasi antara elektron dan *hole* menghasilkan pelepasan energi berupa pancaran cahaya.

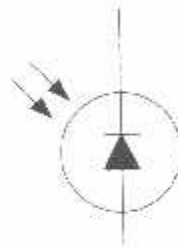
Efisiensi pancaran cahaya akan berkurang seiring dengan berkurangnya arus input dan kenaikan suhu. Pada LED infra merah, cahaya yang dipancarkan mempunyai panjang gelombang 0,1 mm – 1 μm sehingga pancaran gelombang tersebut tidak tertangkap oleh mata manusia.

2.5. Photodioda

Photodioda merupakan dioda yang peka terhadap cahaya. Suatu sumber cahaya menghasilkan energi panas begitu pula dengan spektrum infra merah. Karena spektrum infra merah mempunyai energi panas yang lebih besar dari cahaya tampak, maka *photodioda* lebih peka menangkap radiasi dari infra merah.

Komponen ini akan mengubah energi cahaya, dalam hal ini energi cahaya infra merah menjadi sinyal listrik. Komponen ini harus mampu mengumpulkan sinyal cahaya sebanyak mungkin sehingga pulsa-pulsa sinyal listrik yang dihasilkan kualitasnya cukup baik. Semakin besar intensitas cahaya yang diterima maka sinyal pulsa listrik yang dihasilkan akan baik jika sinyal cahaya diterima intensitasnya lemah maka penerima tersebut harus mempunyai pengumpul cahaya (*light collector*) yang cukup baik dan sinyal pulsa yang dihasilkan oleh sensor cahaya ini harus dikuatkan.

Simbol dari *photodioda* adalah :



Gambar 2.4. Simbol *Photodioda*

Pada *photodioda* ini terdapat suatu jendela kecil yang memungkinkan cahaya luar dapat masuk mengenai *pin function*. Pada keadaan normal *photodioda* berlaku sebagai dioda biasa yang dapat menghantarkan listrik dari anoda ke katoda, namun mempunyai tahanan balik yang besar. Bila cahaya luar mengenai *pin function photodioda*, maka tahanan balik akan mengecil dan menimbulkan

arus balik, sehingga photodiode berlaku sebagai diode yang dibalik atau dibias *reverse*.

Semakin besar intensitas cahaya yang diterima maka semakin besar pula arus balik yang ditimbulkannya. Bila energi foton diserap dalam suatu semikonduktor maka akan dihasilkan pasangan *electron hole* pada lapisan yang telah dibangkitkan oleh foton yang saling memisahkan diri karena pengaruh medan listrik, dimana elektron-elektron akan menuju ke sisi N dan hole menuju ke sisi P, sehingga dihasilkan arus dari katoda menuju anoda. Karena pengaruh suhu *junction* yang lebih tinggi, menciptakan lebih banyak pasangan *electron hole*, sehingga mengakibatkan arus balik yang melewati *junction* bertambah.

Sebuah *photodiode* biasanya dikemas dengan plastik transparan yang juga berfungsi sebagai lensa fresnel. Lensa ini merupakan lensa cembung yang mempunyai sifat mengumpulkan cahaya. Walaupun demikian cahaya yang nampakpun masih bisa mengganggu kerja dari *photodiode* karena tidak semua cahaya nampak bisa difilter dengan baik. Oleh karena itu sebuah penerima laser harus mempunyai filter kedua yaitu rangkaian filter yang berfungsi untuk memfilter sinyal *carrier* yang terbawa oleh cahaya laser tersebut.

Faktor lain yang juga berpengaruh pada kemampuan penerima infra merah adalah '*active area*' dan '*respond time*'. Semakin besar area penerimaan suatu *photodiode* maka semakin besar pula intensitas cahaya yang dikumpulkannya sehingga arus bocor yang diharapkan pada teknik '*reversed bias*' semakin besar. Selain itu semakin besar area penerimaan maka sudut penerimaannya juga semakin besar. Kelemahan area penerimaan yang semakin besar ini adalah *noise* yang dihasilkan juga semakin besar pula. Begitu juga dengan respon terhadap

frekuensi, semakin besar area penerimaannya maka respon frekuensinya turun dan sebaliknya jika area penerimaannya kecil maka respon terhadap sinyal frekuensi tinggi cukup baik.

Respond time dari suatu *photodiode* (penerima) mempunyai waktu respon yang biasanya dalam satuan nano detik. *Respond time* ini mendefinisikan lama agar *photodiode* merespon cahaya infra merah yang datang pada area penerima. Sebuah *photodiode* yang baik paling tidak mempunyai *respond time* sebesar 500 nano detik atau kurang. Jika *respond time* terlalu besar maka *photodiode* ini tidak dapat merespon sinyal cahaya yang dimodulasi dengan sinyal *carrier* frekuensi tinggi dengan baik. Hal ini akan mengakibatkan adanya *data loss*.

2.6. Transistor Bipolar

Prinsip kerja transistor adalah arus bias base-emiter yang kecil mengatur besar arus kolektor-emiter. Bagian penting berikutnya adalah bagaimana caranya memberi arus bias yang tepat sehingga transistor dapat bekerja optimal.

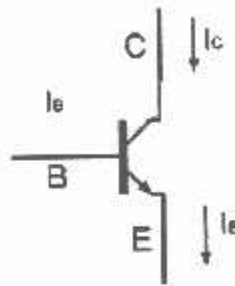
2.6.1. Arus Bias

Ada tiga cara yang umum untuk memberi arus bias pada transistor, yaitu rangkaian *CE (Common Emitter)*, *CC (Common Collector)* dan *CB (Common Base)*. Namun dalam hal ini akan lebih detail dijelaskan bias transistor rangkaian *CE*. Dengan menganalisa rangkaian *CE* akan dapat diketahui beberapa parameter penting dan berguna terutama untuk memilih transistor yang tepat untuk berbagai aplikasi.

2.6.2. Arus Emiter

Dari hukum Kirchoff diketahui bahwa jumlah arus yang masuk ke satu titik akan sama jumlahnya dengan arus yang keluar. Jika teorema tersebut diaplikasikan pada transistor, maka hukum itu menjelaskan hubungan :

$$I_E = I_C + I_B$$



Gambar 2.5. Arus Emitor

Persamaan diatas mengatakan arus *emiter* I_E adalah jumlah dari arus kolektor I_C dengan arus base I_B . Karena arus I_B sangat kecil sekali atau disebutkan $I_B \ll I_C$, maka dapat dinyatakan :

$$I_E = I_C$$

2.6.3. Alpha (α)

Pada tabel data transistor (*databook*) sering dijumpai spesifikasi a_{dc} (*alpha dc*) yang tidak lain adalah :

$$a_{dc} = I_C/I_E$$

Definisinya adalah perbandingan arus kolektor terhadap arus emitor. Karena besar arus kolektor umumnya hampir sama dengan besar arus emiter maka idealnya besar a_{dc} adalah = 1 (satu). Namun umumnya transistor yang ada memiliki a_{dc} kurang lebih antara 0.95 sampai 0.99.

2.6.4. Beta (b)

Beta didefinisikan sebagai besar perbandingan antara arus kolektor dengan arus *base*.

$$b = I_C / I_B$$

Dengan kata lain, *b* adalah parameter yang menunjukkan kemampuan penguatan arus (*current gain*) dari suatu transistor. Parameter ini ada tertera di *data book* transistor dan sangat membantu para perancang rangkaian elektronika dalam merencanakan rangkaiannya.

Misalnya jika suatu transistor diketahui besar $b=250$ dan diinginkan arus kolektor sebesar 10 mA, maka berapakah arus bias base yang diperlukan. Maka :

$$I_B = I_C / b = 10 \text{ mA} / 250 = 40 \text{ uA}$$

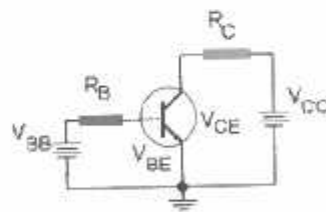
Arus yang terjadi pada kolektor transistor yang memiliki $b = 200$ jika diberi arus bias base sebesar 0.1 mA adalah :

$$I_C = b I_B = 200 \times 0.1 \text{ mA} = 20 \text{ mA}$$

Dari rumusan ini lebih terlihat definisi penguatan arus transistor, yaitu sekali lagi, arus *base* yang kecil menjadi arus kolektor yang lebih besar.

2.6.5. Common Emitter (CE)

Rangkaian CE adalah rangkain yang paling sering digunakan untuk berbagai aplikasi yang menggunakan transistor. Dinamakan rangkaian CE, sebab titik *ground* atau titik tegangan 0 volt dihubungkan pada titik *emiter*.



Gambar 2.6. Rangkaian CE

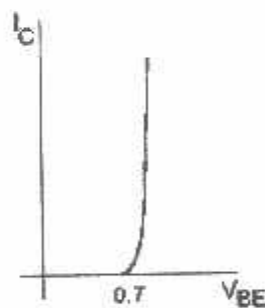
Sekilas tentang notasi, ada beberapa notasi yang sering digunakan untuk menunjukkan besar tegangan pada suatu titik maupun antar titik. Notasi dengan 1 *subscript* adalah untuk menunjukkan besar tegangan pada satu titik, misalnya V_C = tegangan kolektor, V_B = tegangan base dan V_E = tegangan emiter.

Ada juga notasi dengan 2 *subscript* yang dipakai untuk menunjukkan besar tegangan antar 2 titik, yang disebut juga dengan tegangan jepit. Diantaranya adalah :

- ❖ V_{CE} = tegangan jepit kolektor-emitor.
- ❖ V_{BE} = tegangan jepit base-emitor.
- ❖ V_{CB} = tegangan jepit kolektor-base.

Notasi seperti V_{BB} , V_{CC} , V_{EE} berturut-turut adalah besar sumber tegangan yang masuk ke titik base, kolektor dan emitor.

2.6.6. Kurva Base



Grafik 2.7. Kurva $I_B - V_{BE}$

Hubungan antara I_B dan V_{BE} tentu saja akan berupa kurva dioda. Karena memang telah diketahui bahwa junction *base-emitor* tidak lain adalah sebuah dioda. Jika hukum Ohm diterapkan pada loop base diketahui adalah :

$$I_B = (V_{BB} - V_{BE}) / R_B$$

V_{BE} adalah tegangan jepit dioda *junction base-emitor*. Arus hanya akan mengalir jika tegangan antara base-emitor lebih besar dari V_{BE} . Sehingga arus I_B mulai aktif mengalir pada saat nilai V_{BE} tertentu. Besar V_{BE} umumnya tercantum di dalam *databook*. Tetapi untuk penyederhanaan umumnya diketahui $V_{BE} = 0.7$ volt untuk transistor silikon dan $V_{BE} = 0.3$ volt untuk transistor germanium.

Sampai disini akan sangat mudah mengetahui arus I_B dan arus I_C dari rangkaian berikut ini, jika diketahui besar $b = 200$. Katakanlah yang digunakan adalah transistor yang dibuat dari bahan silikon.

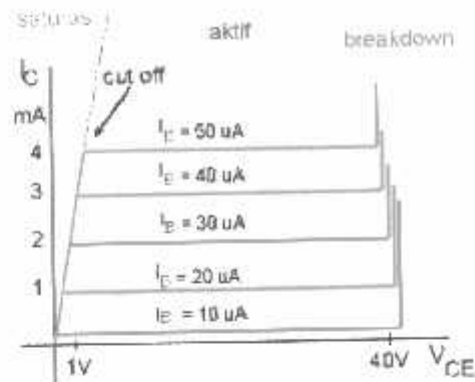
$$\begin{aligned} I_B &= (V_{BB} - V_{BE}) / R_B \\ &= (2V - 0.7V) / 100 K \\ &= 13 \mu A \end{aligned}$$

Dengan $b = 200$, maka arus kolektor adalah :

$$I_C = b I_B = 200 \times 13 \mu A = 2.6 \text{ mA}$$

2.6.7. Kurva Kolektor

Sekarang sudah diketahui konsep arus base dan arus kolektor. Satu hal lain yang menarik adalah bagaimana hubungan antara arus base I_B , arus kolektor I_C dan tegangan kolektor-emitor V_{CE} . Pada grafik berikut telah diplot beberapa kurva kolektor arus I_C terhadap V_{CE} dimana arus I_B dibuat konstan.



Grafik 2.8. Kurva Kolektor

Dari kurva ini terlihat ada beberapa *region* yang menunjukkan daerah kerja transistor. Pertama adalah daerah *saturasi*, lalu daerah *cut-off*, kemudian daerah aktif dan seterusnya daerah *breakdown*.

2.6.8. Daerah Aktif

Daerah kerja transistor yang normal adalah pada daerah aktif, dimana arus I_C konstan terhadap berapapun nilai V_{CE} . Dari kurva ini diperlihatkan bahwa arus I_C hanya tergantung dari besar arus I_B . Daerah kerja ini biasa juga disebut daerah linier (*linear region*).

Jika hukum Kirchhoff mengenai tegangan dan arus diterapkan pada loop kolektor (rangkaian CE), maka dapat diperoleh hubungan :

$$V_{CE} = V_{CC} - I_C R_C$$

Dapat dihitung disipasi daya transistor adalah :

$$P_D = V_{CE} I_C$$

Rumus ini mengatakan jumlah disipasi daya transistor adalah tegangan kolektor-emitor dikali jumlah arus yang melewatinya. Disipasi daya ini berupa panas yang menyebabkan naiknya temperatur transistor. Umumnya untuk transistor power sangat perlu untuk mengetahui spesifikasi P_{Dmax} . Spesifikasi ini

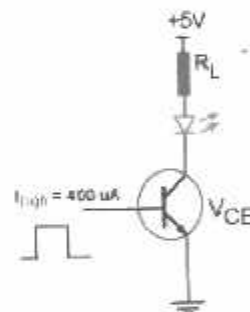
menunjukkan temperatur kerja maksimum yang diperbolehkan agar transistor masih bekerja normal. Sebab jika transistor bekerja melebihi kapasitas daya P_{Dmax} , maka transistor dapat rusak atau terbakar.

2.6.9. Daerah Saturasi

Daerah saturasi adalah mulai dari $V_{CE} = 0$ volt sampai kira-kira 0.7 volt (transistor silikon), yaitu akibat dari efek dioda kolektor-base yang mana tegangan V_{CE} belum mencukupi untuk dapat menyebabkan aliran elektron.

2.6.10. Daerah *Cut-Off*

Jika kemudian tegangan V_{CC} dinaikkan perlahan-lahan, sampai tegangan V_{CE} tertentu tiba-tiba arus IC mulai konstan. Pada saat perubahan ini, daerah kerja transistor berada pada daerah *cut-off* yaitu dari keadaan saturasi (OFF) lalu menjadi aktif (ON). Perubahan ini dipakai pada system digital yang hanya mengenal angka biner 1 dan 0 yang tidak lain dapat direpresentasikan oleh status transistor OFF dan ON.



Gambar 2.9. Rangkaian *Driver* LED

Misalkan pada rangkaian driver LED di atas, transistor yang digunakan adalah transistor dengan $b = 50$. Penyalaan LED diatur oleh sebuah gerbang logika (*logic gate*) dengan arus *output high* = 400 uA dan diketahui tegangan

forward LED, $V_{LED} = 2.4$ volt. Lalu pertanyaannya adalah, berapakah seharusnya resistansi R_L yang dipakai.

$$I_C = \beta I_B = 50 \times 400 \mu A = 20 \text{ mA}$$

Arus sebesar ini cukup untuk menyalakan LED pada saat transistor *cut-off*. Tegangan V_{CE} pada saat *cut-off* idealnya = 0, dan aproksimasi ini sudah cukup untuk rangkaian ini.

$$\begin{aligned} R_L &= (V_{CC} - V_{LED} - V_{CE}) / I_C \\ &= (5 - 2.4 - 0)V / 20 \text{ mA} \\ &= 2.6V / 20 \text{ mA} \\ &= 130 \text{ Ohm} \end{aligned}$$

2.6.11. Daerah *Breakdown*

Dari kurva kolektor, terlihat jika tegangan V_{CE} lebih dari 40V, arus I_C menanjak naik dengan cepat. Transistor pada daerah ini disebut berada pada daerah *breakdown*. Seharusnya transistor tidak boleh bekerja pada daerah ini, karena akan dapat merusak transistor tersebut. Untuk berbagai jenis transistor nilai tegangan V_{CEmax} yang diperbolehkan sebelum *breakdown* bervariasi. V_{CEmax} pada databook transistor selalu dicantumkan juga.

2.7. Operational Amplifier (*Op-Amp*)

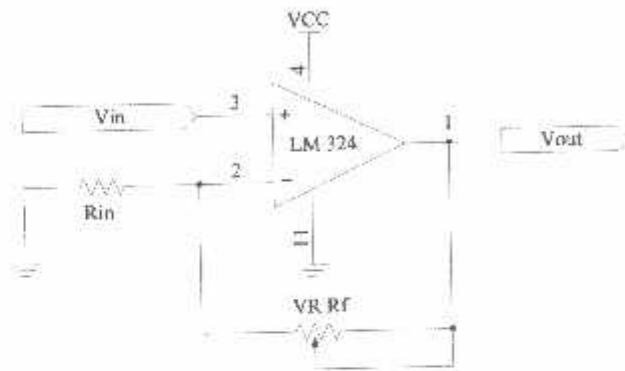
Penguat operasional (*Operational Amplifier*) dapat dipakai sebagai penguat tak membalik sebagaimana terlihat pada gambar 2.10. Penguat adalah suatu rangkaian yang menerima sebuah isyarat di masukannya dan mengeluarkan sebarang isyarat tak berubah di keluarannya. Tegangan keluaran (V_{out}) akan mempunyai polaritas yang sama seperti tegangan masukan (V_{in}).

Tegangan keluaran pada penguat tak membalik dapat dihitung dengan persamaan sebagai berikut:

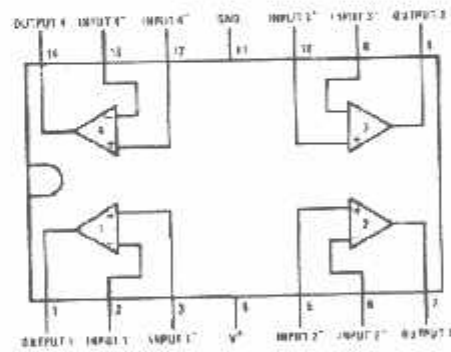
$$V_{out} = \left(1 + \frac{R_f}{R_{in}}\right) \cdot V_{in}$$

Dan besarnya penguatan tegangan dapat diperoleh dengan rumus:

$$A = \left(1 + \frac{R_f}{R_{in}}\right)$$

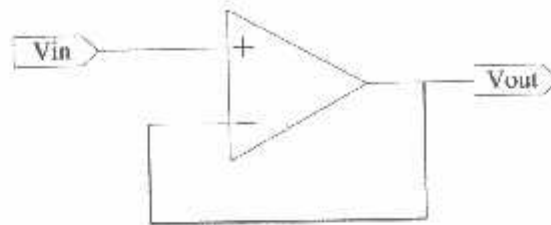


Gambar 2.10. Penguat Tak Membalik (Non Inverting)



Gambar 2.11. Konfigurasi Pin LM324

Jika sebuah penguat memiliki penguatan sebesar 1 X, dikatakan bahwa penguat tersebut sebagai *voltage follower* atau *buffer*. Rangkaian *voltage follower* atau *buffer* seperti pada gambar berikut ini:



Gambar 2.12. Rangkaian *Voltage Follower* Atau *Buffer*

2.8. Schmitt Trigger

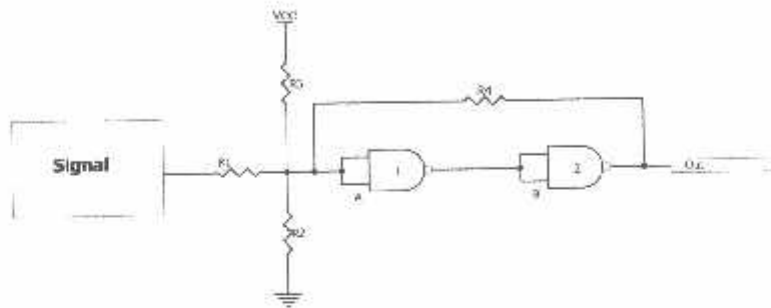
Schmitt Trigger, serupa dengan bistabil, astabil, dan monostabil, merupakan rangkaian *regenerative* dua gate dengan *feedback* diantara input dan outputnya. Suatu IC schmitt trigger seperti yang diperlihatkan dalam gambar 2.9, merupakan suatu *voltage level detector*. Bila tegangan inputnya lebih rendah dari suatu batas tegangan tertentu maka disebut sebagai *upper trip level (UTL)*, tegangan outputnya tepat pada batas logik 0 dan bilamana tegangan *input* tersebut diatas UTL-nya maka tegangan outputnya tepat pada batas logik 1.

Pada mulanya tegangan input pada suatu batas *high* maka rangkaian akan berubah menjadi output dengan level 0 pada suatu tegangan input yang disebut *Lower Trip Level (LTL)*. LTL ini lebih rendah dibandingkan dengan UTL dan perbedaan tegangan merupakan sesuatu yang *hysteresis*, diukur dalam volts :

$$\text{Hysteresis} = \text{UTL} - \text{LTL}$$

Rangkaiannya adalah *regenerative*, dan sebagai level trips yang mencapai transisi yang terjadi dengan kecepatan besar. Disamping itu digunakan untuk

mengubah suatu *slowly changing signal* menjadi sesuatu dengan transisi yang cepat. Ini mempunyai manfaat yang besar dalam *pulses couple circuitry* dimana kopling kapasitornya dapat diperkecil. Dalam gambar 2.9, tahanan R2 dapat digunakan untuk mengatur *trips point*-nya.



Gambar 2.13. Rangkaian Dasar *Schmitt Trigger*

Rangkaian gambar diatas dioperasikan dalam cara berikut ini. Jika tegangan signal dan lebih rendah dari tegangan pada titik A, input pada gate satu akan lebih rendah/dibawah tegangan *threshold*-nya.

Hal ini dibantu oleh level 0 dari gate 2 melalui R4. Bila *signal input*-nya pada suatu *high* yang hampir mengenai tegangan (UTL) menjadikan tegangan pada A sama dengan tegangan *threshold* dari *gate*-nya, *gate 1* mengubah level output-nya dan saat itu pula menjadikan output gate 2 menjadi level 1. Ini membantu menjaga/memelihara titik A berada pada tegangan transisinya. (Pekerjaan ini merupakan *regenerative* dan mengambil posisi secara cepat).

Sekarang jika kita memulai memperkecil tegangan inputnya, kita harus memperkecilnya sehingga tegangan pada titik A mencapai pada suatu tegangan *threshold* lagi. Ini harus lebih kecil daripada UTL sebab R4 sekarang dihubungkan pada suatu level yang lebih tinggi. Pada tegangan ini, LTL, circuit *regenerative*-nya segera menuju output lagi.

2.9. Buzzer

Pada perancangan alat ini digunakan beberapa rangkaian indikator yang akan digunakan sebagai tanda informasi adanya kereta api yang akan masuk ke stasiun.

Buzzer digunakan sebagai indikator dimana bila sensor kedatangan kereta api mendeteksi adanya kereta api yang akan memasuki stasiun maka *buzzer* akan berbunyi. *Buzzer* memerlukan sumber tegangan 5 volt dan dihubungkan dengan pin P2.1 pada mikrokontroler AT89S51 slave 1.



Gambar 2.14. Simbol *Buzzer*

2.10. Relay

Relay adalah komponen elektronika yang terdiri dari sebuah lilitan kawat (kumparan/koil) yang terlilit pada sebuah besi lunak. Pada gambar 2.15, Jika kumparan dialiri arus listrik maka inti besi akan menjadi magnet dan menarik pegas sehingga kotak AB terhubung dan BC terputus. *Relay* merupakan suatu alat untuk menghubungkan atau memerlukan kontak antara komponen yang satu dengan yang lain.

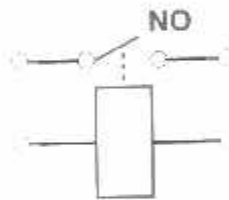


Gambar 2.15. Cara Kerja *Relay*

Dalam memutus atau menghubungkan kontak digerakkan oleh fluksi yang ditimbulkan dari adanya medan magnet listrik yang dihasilkan oleh kumparan yang melilit pada besi lunak. Ada beberapa macam *relay*, antara lain:

1. **SPST** (*Single Pin Single Terminal*)

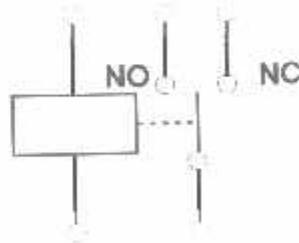
Simbol *Relay SPST*



Gambar 2.16. *Relay SPST*

2. **SPDT** (*Single Pin Dual Terminal*)

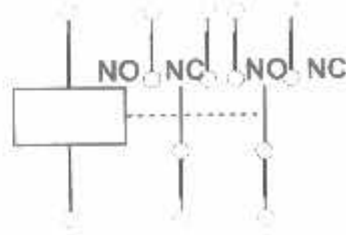
Simbol *Relay SPDT*



Gambar 2.17. *Relay SPDT*

3. DPDT (*Dual Pin Dual Terminal*)

Simbol *Relay DPDT*



Gambar 2.18. Relay DPDT

2.11. Motor DC



Gambar 2.19. Garis-Garis Medan Magnet disekitar Arus Listrik
Pada Kawat Lurus

Setiap arus yang mengalir melalui sebuah konduktor akan menimbulkan medan magnet. Arah medan magnet dapat ditentukan dengan kaidah tangan kanan. Ibu jari tangan menunjukkan arah aliran arus listrik sedangkan jari-jari yang lain menunjukkan arah medan magnet yang timbul, seperti yang ditunjukkan oleh gambar 2.19.

Kaidah tangan kanan untuk motor menunjukkan arah arus yang mengalir didalam sebuah konduktor yang berada dalam medan magnet. Jari tengah menunjukkan arah arus yang mengalir pada konduktor, jari telunjuk menunjukkan arah medan magnet dan ibu jari menunjukkan arah gaya putar.

Adapun besarnya gaya yang bekerja pada konduktor tersebut dapat dirumuskan dengan :

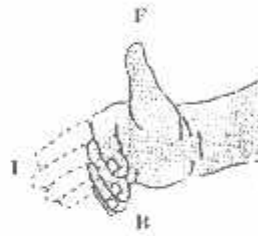
$$F = B \cdot I \cdot L \cdot \sin \theta \quad (\text{Newton})$$

Dimana : B = kerapatan fluks magnet (weber)

L = panjang konduktor (meter)

I = arus listrik (ampere)

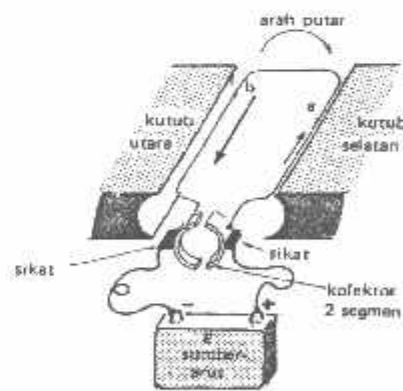
$\sin \theta$ = sudut antara antara arus dengan garis-garis medan



Gambar 2.20. Kaidah Tangan Kanan

Cara Kerja Motor DC

Adapun cara kerja motor DC dapat dilihat pada gambar dibawah ini:

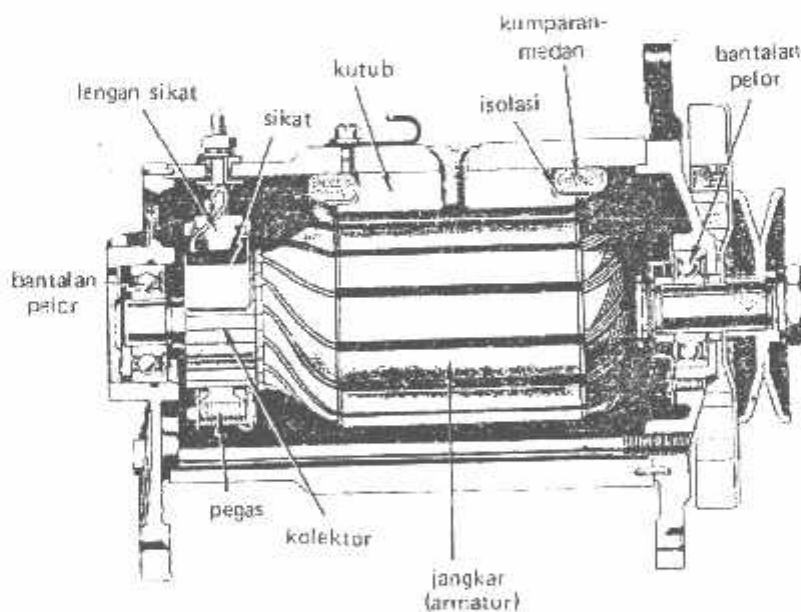


Gambar 2.21. Dasar Konstruksi Motor DC

Ada satu lilit kawat a – b berada di dalam medan magnet. Lilitan ini dapat berputar dengan bebas, lilitan ini biasa disebut dengan jangkar (*armour*).

Pada jangkar dimasukkan arus yang berasal dari sumber (baterai) E. Koneksi baterai dengan jangkar melalui sikat-sikat. Sikat-sikat ini terpasang pada sebuah cincin yang terbelah dua, yang disebut kolektor. Adapun tujuan dari konstruksi ini adalah agar lilitan kawat dapat berputar apabila ada arus listrik yang melewatinya.

Pada kawat yang berada di kanan arus mengalir dari depan ke belakang dalam kawat yang di kiri, arus mengalir dari belakang ke depan kawat a dan b secara berganti-gantian berada di kiri dan kanan. Karena itu arah arus di a dan arah arus di b selalu membolak balik. Pembalikan arah arus itu terjadi pada saat lilitan kawat melintasi posisi vertikal.

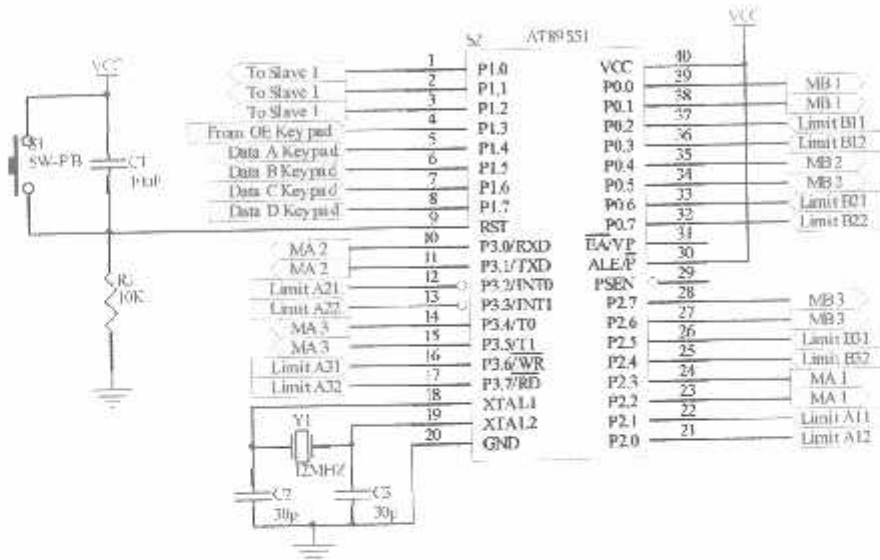


Gambar 2.22. Sebuah Motor DC

- ❖ P3.0 memberikan pulsa ke RS pada LCD.
- ❖ P3.1 memberikan pulsa ke E (enable) pada LCD.
- ❖ P3.4 s.d. P3.7 sebagai output indikator LED.

3.2.15. Minimum System Mikrokontroler Slave 2 AT89S51

Mikrokontroler ini berfungsi sebagai pengendali motor dc Gambar berikut adalah konfigurasi pin-pin yang terpakai :



Gambar 3.17. Minimum System Mikrokontroler Slave 2 AT89S51

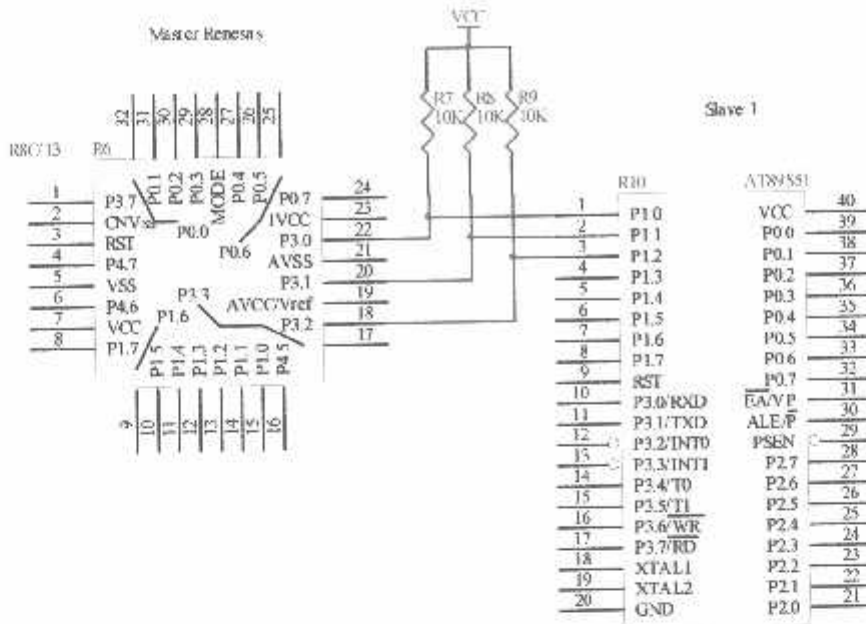
- ❖ P0.0 dan P0.1 digunakan untuk memberikan masukan driver motor B1.
- ❖ P0.2 dan P0.3 sebagai masukan dari sensor limit motor B1.
- ❖ P0.4 dan P0.5 digunakan untuk memberikan masukan driver motor B2.
- ❖ P0.6 dan P0.7 sebagai masukan dari sensor limit motor B2.
- ❖ P2.0 dan P2.1 sebagai masukan dari sensor limit motor A1.
- ❖ P2.2 dan P2.3 digunakan untuk memberikan masukan driver motor A1.
- ❖ P2.4 dan P2.5 sebagai masukan dari sensor limit motor B3.
- ❖ P2.6 dan P2.7 digunakan untuk memberikan masukan driver motor B3.

- ❖ P3.0 dan P3.1 digunakan untuk memberikan masukan driver motor A2.
- ❖ P3.2 dan P3.3 sebagai masukan dari sensor limit motor A2.
- ❖ P3.4 dan P3.5 digunakan untuk memberikan masukan driver motor A3.
- ❖ P3.6 dan P3.7 sebagai masukan dari sensor limit motor A3.
- ❖ P1.0 berfungsi untuk mengirim pulsa ke mikrokontroler *slave 1* untuk membaca data.
- ❖ P1.1 dan P1.2 digunakan untuk mengirim data ke mikrokontroler *slave 1*.
- ❖ P1.3 sebagai masukan dari OE driver keypad.
- ❖ P1.4 s.d. P1.7 digunakan sebagai masukan data dari driver keypad.

3.2.16. Komunikasi Paralel

3.2.16.1. Mikrokontroler Renesas dan *Slave 1* AT89S51

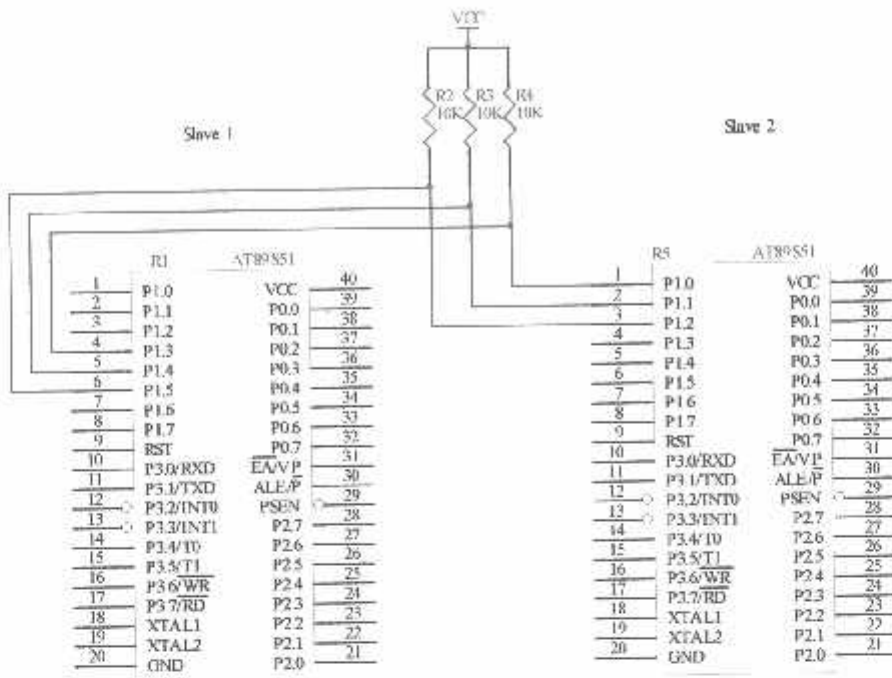
Mikrokontroler renesas berkomunikasi secara paralel dengan mikrokontroler *slave 1*. Menggunakan 3 pin yang dihubungkan secara paralel antara mikrokontroler renesas dengan *slave 1*. *Port 3.0* Renesas terhubung dengan *port 1.0 slave 1* digunakan sebagai *clock enable* dari renesas untuk memberitahu kepada *slave 1* bahwa terdapat data baru. *Port 3.1* dan *3.2* mikrokontroler renesas dihubungkan dengan *port 1.1* dan *1.2 slave 1* digunakan sebagai transfer data antar mikrokontroler.



Gambar 3.18. Komunikasi Mikrokontroler Renesas Dengan Slave 1

3.2.16.2. Mikrokontroler Slave 1 AT89S51 dan Slave 2 AT89S51

Mikrokontroler *slave* 1 juga berkomunikasi secara paralel dengan mikrokontroler *slave* 2. Menggunakan 3 pin yang dihubungkan secara paralel antara mikrokontroler *slave* 1 dengan *slave* 2. *Port* 1.0 *slave* 2 terhubung dengan *port* 1.3 *slave* 1 digunakan sebagai *clock enable* dari *slave* 2 untuk memberitahu kepada *slave* 1 bahwa terdapat data baru. *Port* 1.1 dan 1.2 mikrokontroler *slave* 2 dihubungkan dengan *port* 1.4 dan 1.5 *slave* 1 digunakan sebagai transfer data antar mikrokontroler.



Gambar 3.19. Komunikasi Mikrokontroler *Slave 1* Dengan *Slave 2*

3.3. Perancangan Perangkat Lunak (*Software*).

Pada perancangan perangkat lunak ini dipaparkan dalam diagram alir software dari tiap-tiap mikrokontroler. Bahasa pemrograman yang digunakan adalah bahasa C.

1. Mikrokontroler *Master* Renesas R8C/Tiny R5F21134FP menggunakan pemrograman bahasa C dengan *Compiler* yang dipaket bersama pada suatu IDE yaitu HEW (*High-performance Embedded Workshop*). Fasilitas lainnya yang dibawakan Renesas adalah *software emulator* KD30 dengan menggunakan fasilitas *On-Chip Debugger* R8C yang mempunyai kehandalan mengeliminasi kebutuhan akan *simulator software* dan dapat melakukan *debug* langsung pada *hardware*. *Development Tool Software*

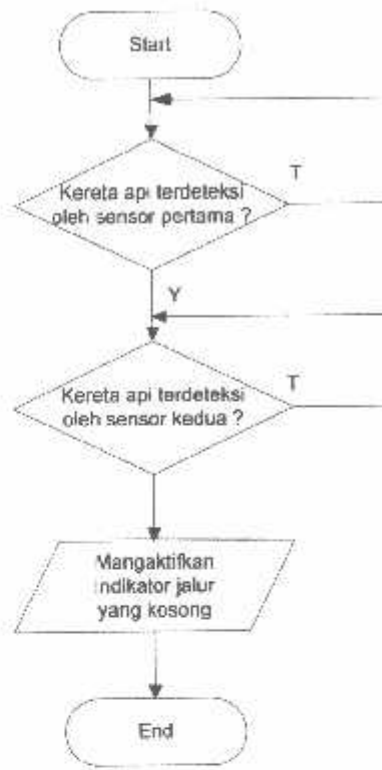
ini disediakan *freeware*-nya beserta *application-application note*-nya pada situs Renesas.

2. Mikrokontroler *Slave 1* dan *Slave 2* AT89S51 menggunakan pemrograman bahasa C. Untuk IDE menggunakan *M-IDE Studio*, *compiler* SDCC (*Small Device C Compiler*) dan untuk *software downloading programmer*-nya menggunakan *ISP Flash Programmer – 3.0a*, kedua software ini dapat *download* di internet secara *freeware*

3.3.1. Flow Chart Keseluruhan Sistem

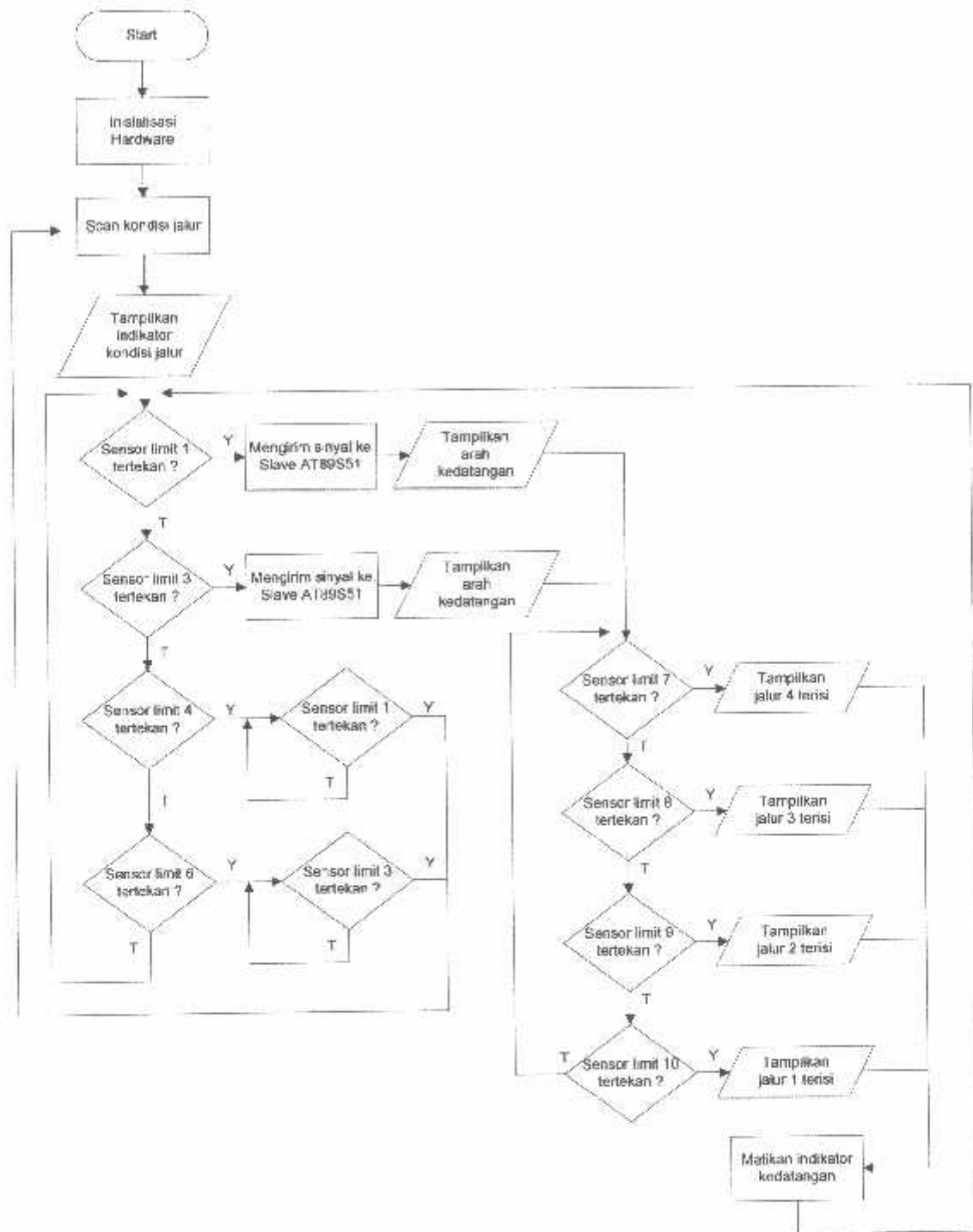


Gambar 3.20. Flow Chart Kedatangan Kereta Api

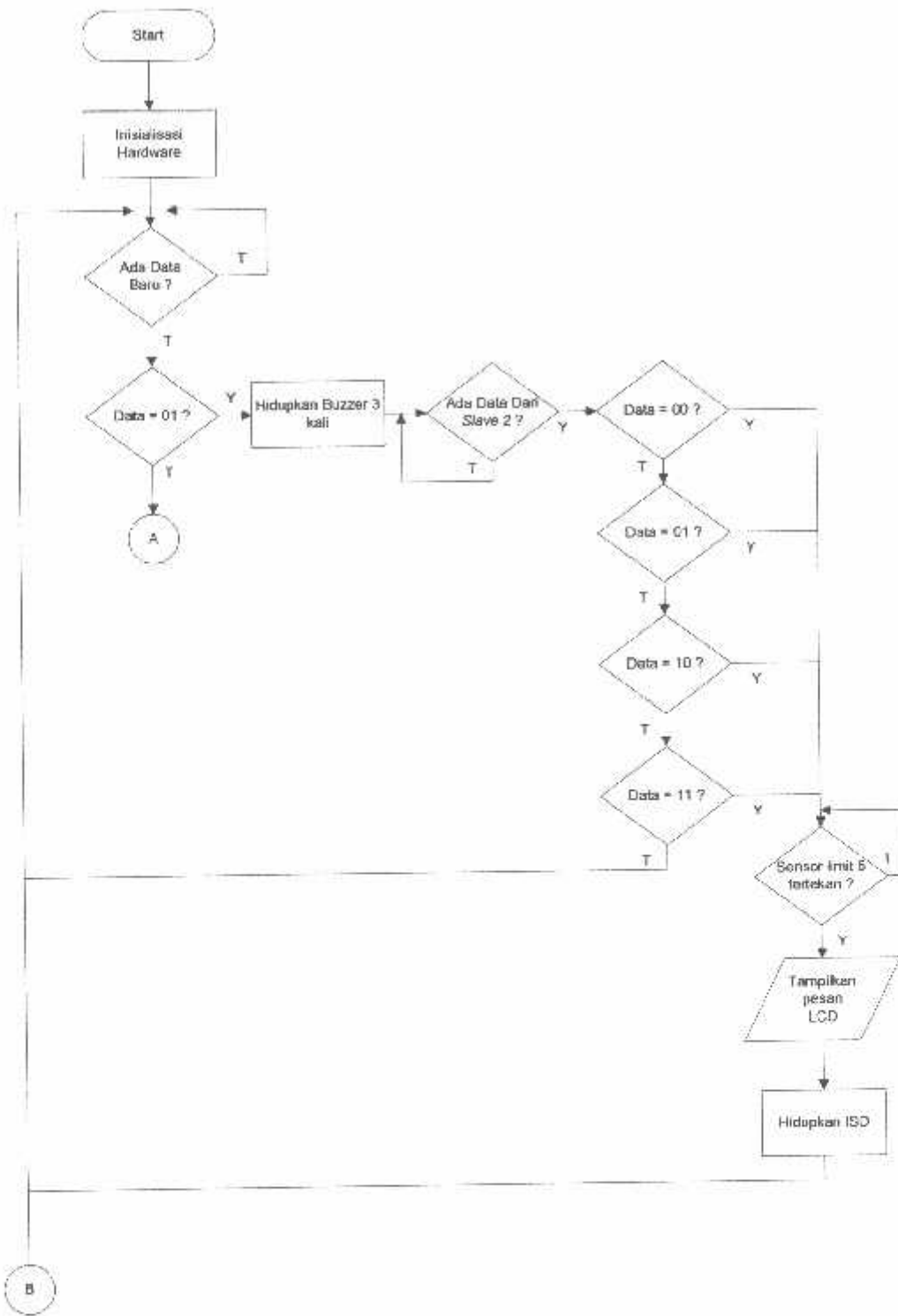


Gambar 3.21. *Flow Chart* Keberangkatan Kereta Api

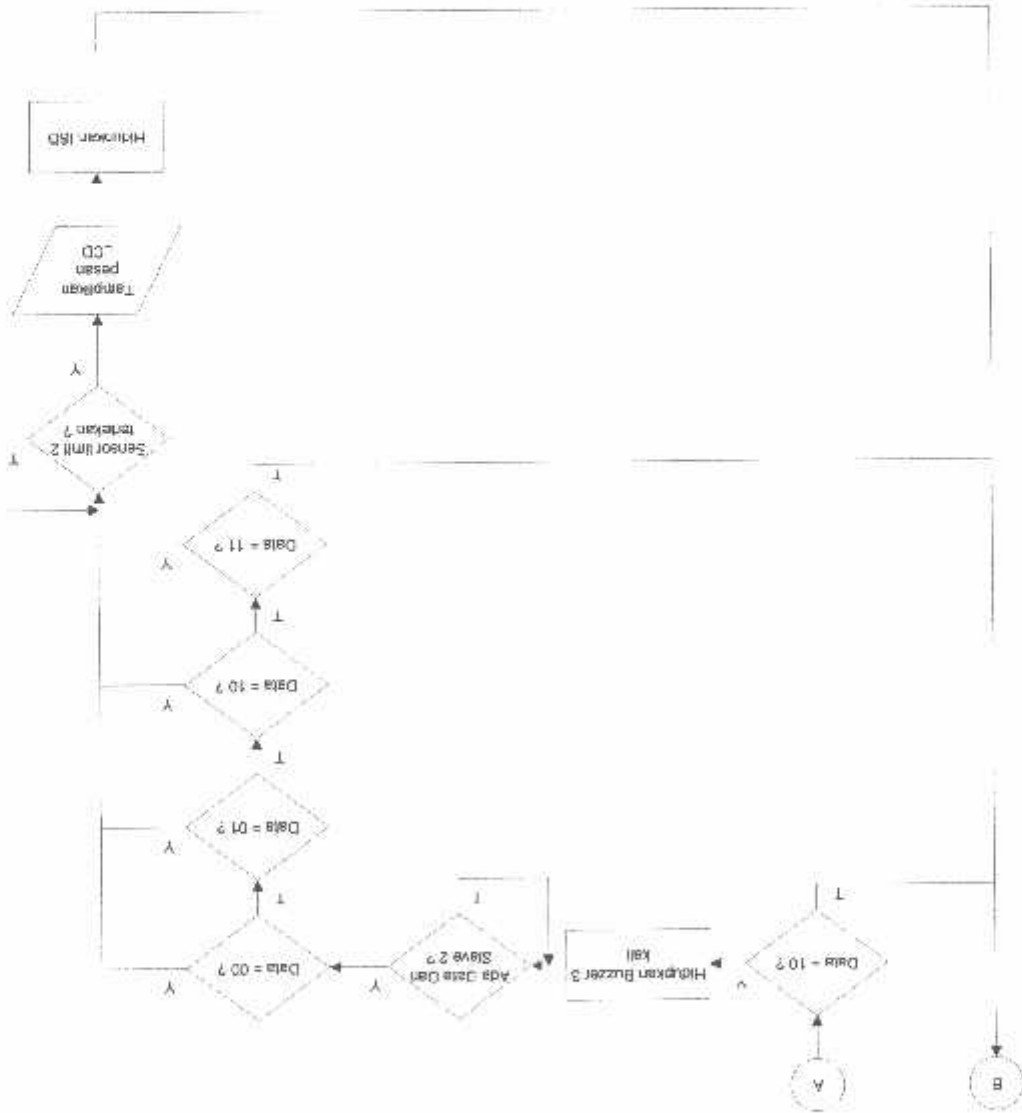
3.3.2. Flow Chart Software Pada Mikrokontroler



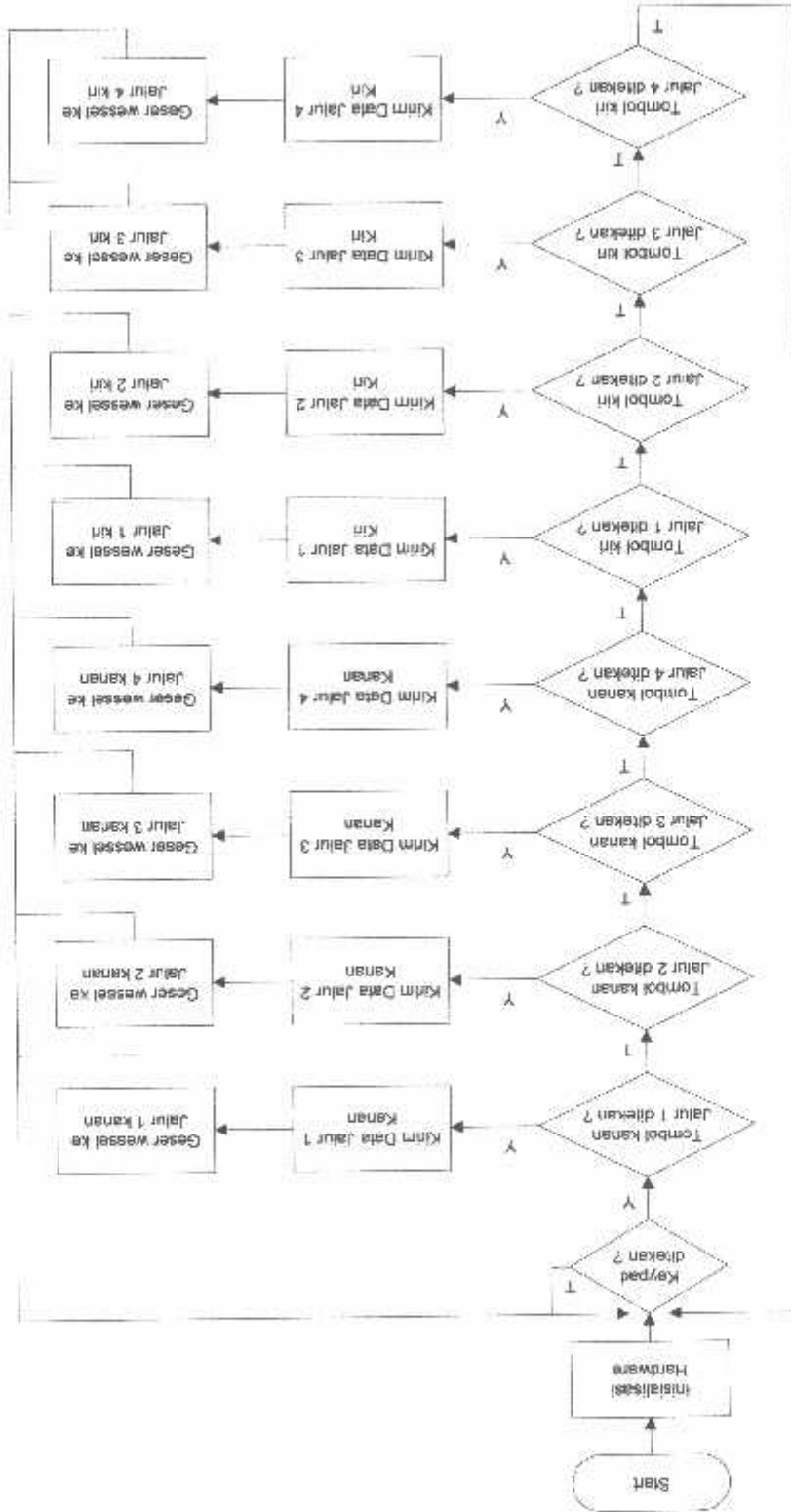
Gambar 3.22. Flowchart Mikrokontroler Master Renesas R8C/13



Gambar 3.23. Flowchart Mikrokontroler Slave 1 AT89S51



Gambar 3.24. Flowchart Mikrokontroler Slave 2 AT89S51



BAB IV

PENGUJIAN ALAT

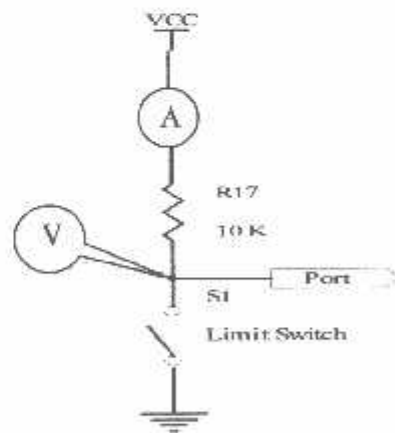
Untuk mendapatkan hasil yang maksimal setelah melaksanakan perancangan dan pembuatan alat, maka perlu dilakukan suatu pengujian terhadap alat yang telah dibuat. Pengujian ini bertujuan untuk mengetahui apakah alat yang telah dibuat dapat bekerja sesuai yang dengan perencanaan.

Bagian-bagian yang diuji dari peralatan ini adalah :

1. Rangkaian detektor kereta api.
2. Rangkaian detektor jalur kereta api.
3. Rangkaian *encoder keypad*.
4. Rangkaian *driver motor*.
5. Rangkaian IC MAX 232

4.1. Rangkaian Detektor Kereta Api

Tujuan pengujian rangkaian detektor kereta api adalah untuk mengetahui nilai tegangan dan arus yang mengalir pada rangkaian *limit switch*.

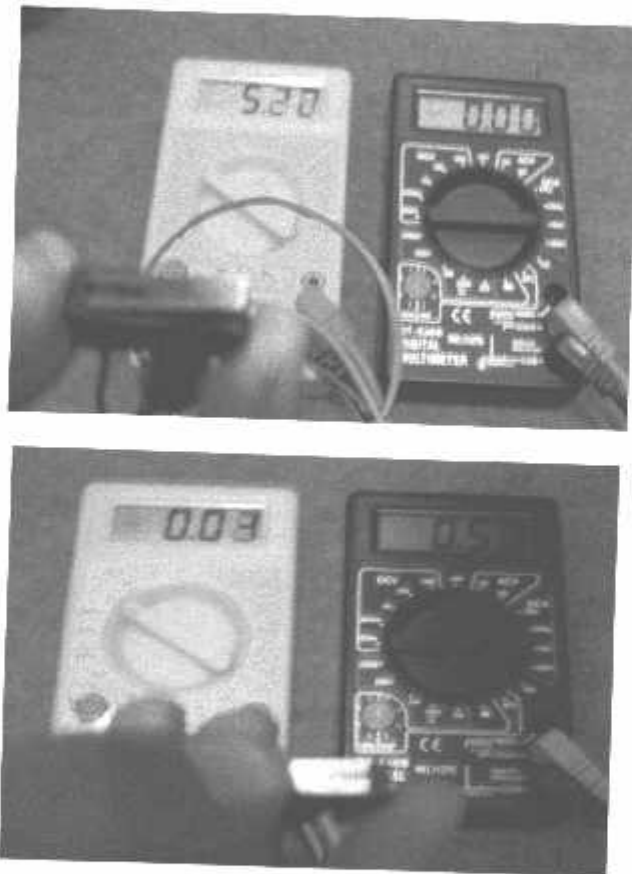


Gambar 4.1. Pengujian Detektor Kereta Api

Rangkaian *limit switch* dihubungkan dengan voltmeter dan amperemeter digital seperti pada gambar 4.1. Kemudian rangkaian diberi catu daya 5 volt. Pada saat *limit switch* tidak tertekan maka tegangan yang terukur 5,22 V dan arusnya 0 A. Ketika *limit switch* tertekan maka tegangan yang terukur menjadi 0,03 V dan arus 0,51 mA.

Tabel 4.1. Hasil Pengujian *Limit Switch*

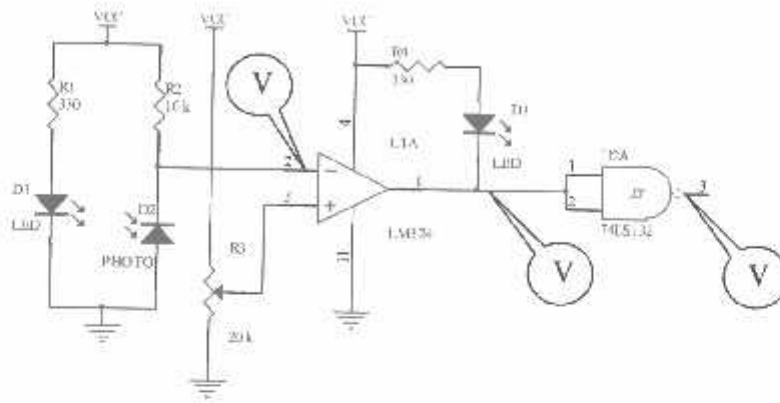
Tegangan (Volt)	Arus (mA)	Kondisi Sensor
5,22	0	Tidak Tertekan
0,03	0,51	Tertekan



Gambar 4.2. Foto Pengujian *Limit Switch*

4.2. Rangkaian Detektor Jalur Kereta Api

Tujuan pengujian rangkaian detektor jalur kereta api adalah untuk mengetahui tegangan keluaran dari komparator dan tegangan keluaran dari *schmitt trigger*.

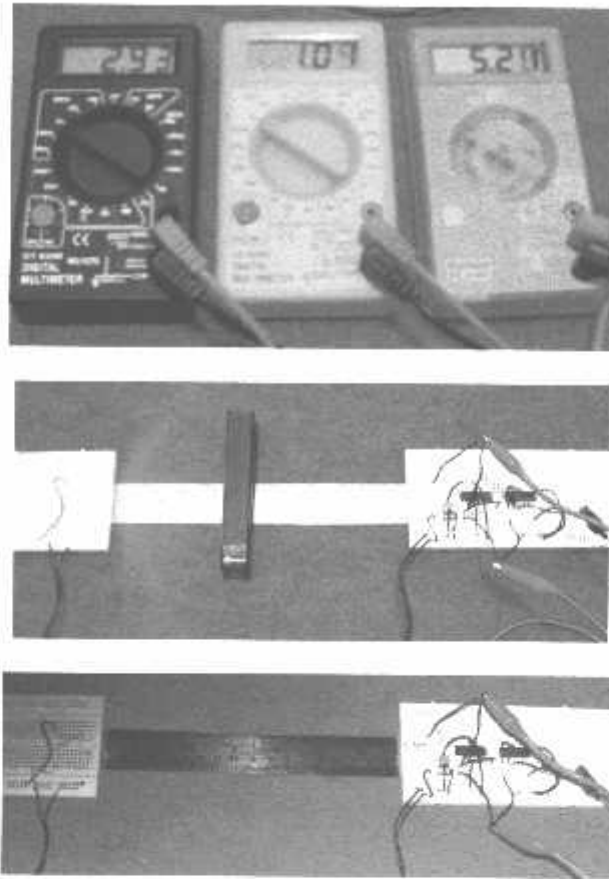


Gambar 4.3. Pengujian Rangkaian Detektor Jalur Kereta Api

Rangkaian detektor jalur kereta api dihubungkan dengan 3 voltmeter digital, masing-masing terhubung dengan keluaran dari *photodiode*, komparator dan *schmitt trigger*. Pengujian dilakukan pada jarak tertentu kemudian mencatat tegangan keluaran yang ditunjukkan voltmeter.

Tabel 4.2. Hasil Pengujian Detektor Jalur Kereta Api

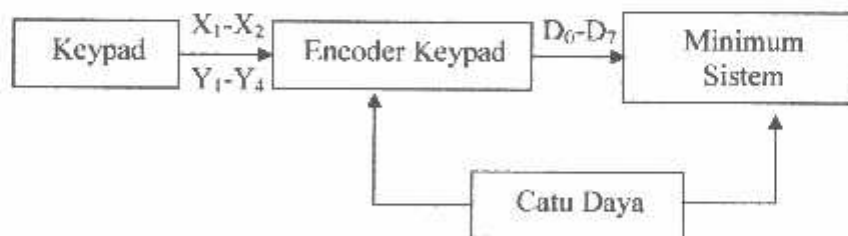
Jarak (cm)	Tegangan Output (Volt)					
	Tanpa Halangan			Terdapat Halangan		
	Output Photodiode	Output Komparator	Output Schmit Trigger	Output Photodiode	Output Komparator	Output Schmit Trigger
2	0.12	1.09	5.2	5.12	4.02	0.02
4	0.15	1.06	5.2	5.12	3.99	0.02
8	0.62	1.05	5.2	5.12	3.97	0.02
12	1.2	1.08	5.2	5.12	3.98	0.02
16	2.38	1.08	5.2	5.12	3.97	0.02
18	2.9	1.05	5.2	5.12	4.01	0.02
Rata-rata	1.228	1.068	5.2	5.12	3.99	0.02



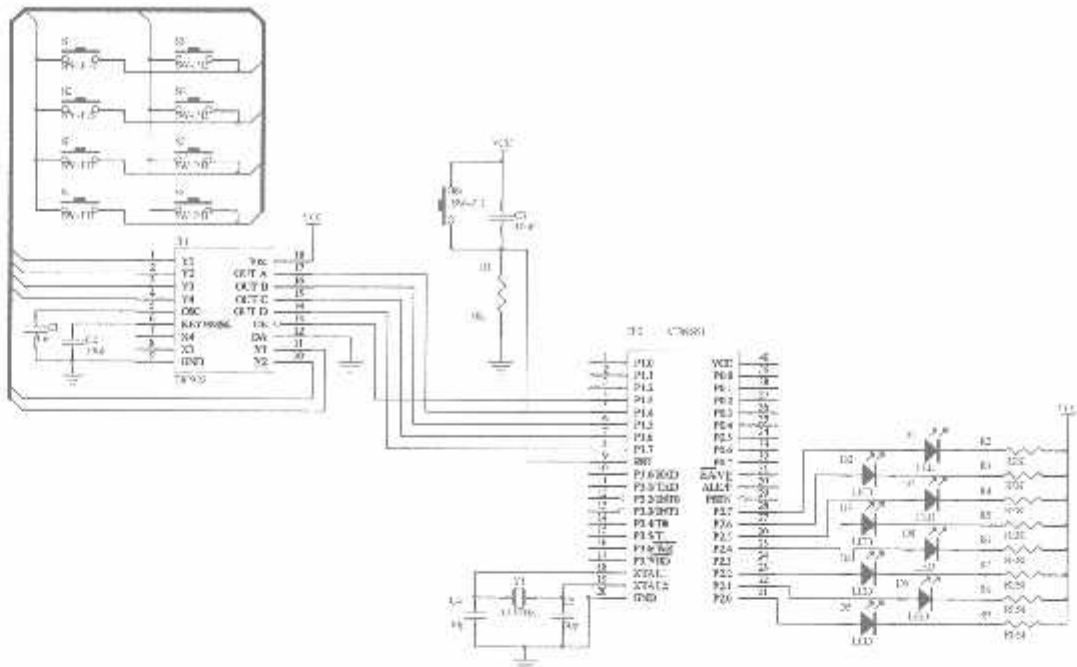
Gambar 4.4. Foto Pengujian Rangkaian Detektor Jalur Kereta Api

4.3. Rangkaian *Encoder Keypad*

Tujuan pengujian rangkaian *encoder keypad* adalah untuk mengetahui data keluaran dari *encoder* IC 74C922 sama dengan data yang ada di *datasheet*.



Gambar 4.5. Diagram Blok Pengujian *Encoder Keypad*



Gambar 4.6. Rangkaian Pengujian *Encoder Keypad*

Dari perancangan *encoder keypad* diketahui bahwa tombol jalur kanan (tombol merah) terhubung dengan kolom X1 dan tombol jalur kiri (tombol hijau) terhubung dengan kolom X2. Data *encoder* terhubung dengan *port* 1.4 – 1.7, sedangkan *port* 1.3 terhubung dengan OE dari *encoder*. Mikrokontroler akan membaca *port* 1.3 berulang ulang. Ketika terdapat penekanan tombol maka OE akan memberi pulsa 1 sesaat yang akan terbaca oleh mikrokontroler. Kemudian mikrokontroler akan membaca data pada *port* 1.4 – 1.7. Data tersebut kemudian akan ditampilkan pada display LED *port* 2. Berikut ini merupakan program mikrokontroler untuk pengujian *encoder keypad*.

```
#include    <AT89x51.h>

void main()
{
    P2 = 0xff;
    while(1)
    {
        if(P1_3 == 1)
```

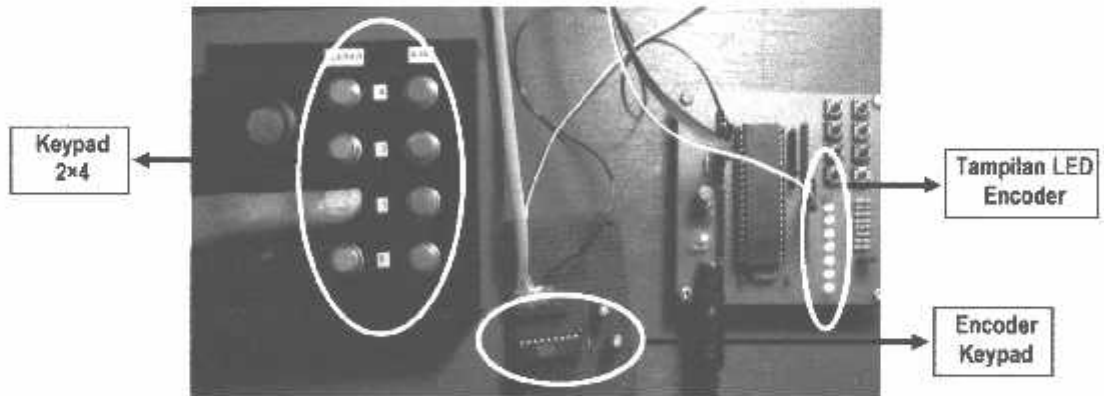
```

    {
        unsigned char a;
        a = P1;
        a &= 0xf0;
        P2 = a;
    }
}

```

Tabel 4.3. Hasil Pengujian *Encoder Keypad*

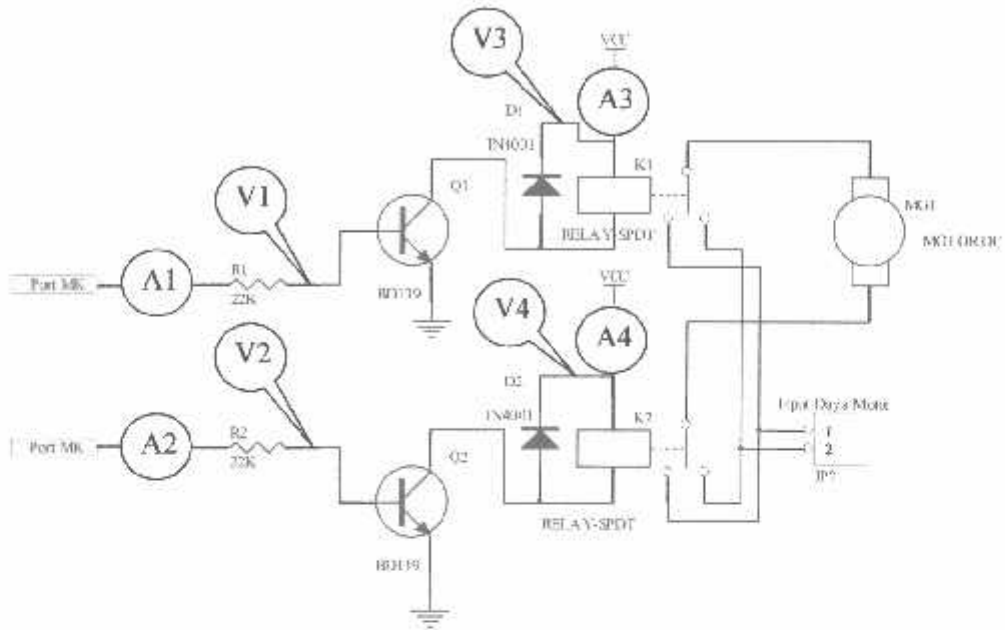
Jalur	Tombol Merah				Tombol Hijau			
	A	B	C	D	A	B	C	D
4	0	0	0	0	1	0	0	0
3	0	0	1	0	1	0	1	0
2	0	0	0	1	1	0	0	1
1	0	0	1	1	1	0	1	1



Gambar 4.7. Foto Pengujian Rangkaian *Encoder Keypad*

4.4. Rangkaian *Driver* Motor

Tujuan pengujian rangkaian *driver* motor adalah untuk mengetahui tegangan serta arus pada rangkaian *driver* motor.



Gambar 4.8. Rangkaian Pengujian *Driver* Motor

Pengujian rangkaian *driver* motor dilakukan dengan mengukur arus dan tegangan pada basis transistor. Pengukuran juga dilakukan pada arus dan tegangan kolektor transistor.

Tabel 4.4. Hasil Pengujian Tegangan Rangkaian *Driver* Motor

V1 (Volt)	V2 (Volt)	V3 (Volt)	V4 (Volt)	Kondisi Motor
0,69	0,01	11,76	0,3	Berputar Ke Kiri
0,02	0,68	0,2	11,75	Berputar Ke Kanan

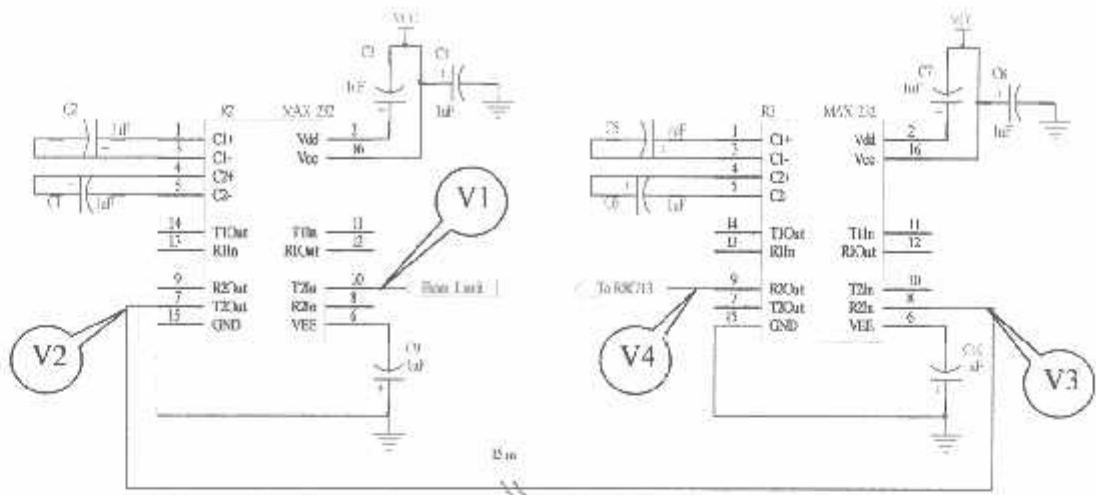
Tabel 4.5. Hasil Pengujian Arus Rangkaian *Driver* Motor

A1 (mA)	A2 (mA)	A3 (mA)	A4 (mA)
0,67	0	27,2	0
0	0,67	0	26,7

**Gambar 4.9.** Foto Pengujian Rangkaian *Driver* Motor

4.5. Rangkaian IC MAX 232

Tujuan pengujian rangkaian *driver* motor adalah untuk mengetahui tegangan serta arus pada rangkaian *driver* motor.

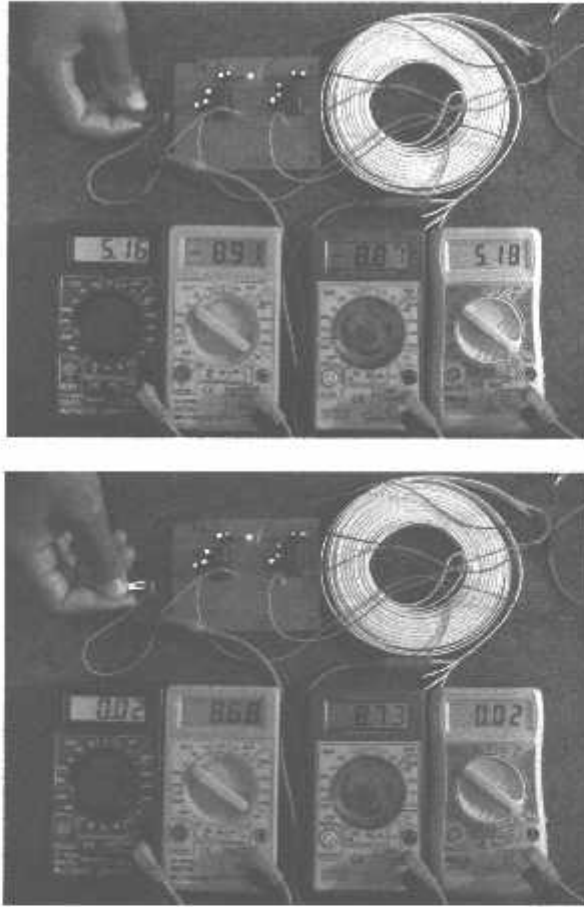


Gambar 4.10. Rangkaian Pengujian IC MAX 232

Pengujian rangkaian IC MAX 232 dilakukan dengan mengukur tegangan dari masukan detektor kereta api (V1), tegangan keluaran dari IC MAX 232 1 (V2), tegangan masukan IC MAX 232 (V3) dan tegangan keluaran IC MAX 232 2 (V4).

Tabel 4.6. Hasil Pengujian Rangkaian IC MAX 232

Kondisi Limit	V1 (Volt)	V2 (Volt)	V3 (Volt)	V4 (Volt)
Tidak Tertekan	5,16	- 8,91	- 8,87	5,18
Tertekan	0,02	8,68	8,73	0,02



Gambar 4.11. Foto Pengujian Rangkaian IC MAX 232

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil pengujian dan analisa dari alat yang telah dibuat maka dapat disimpulkan :

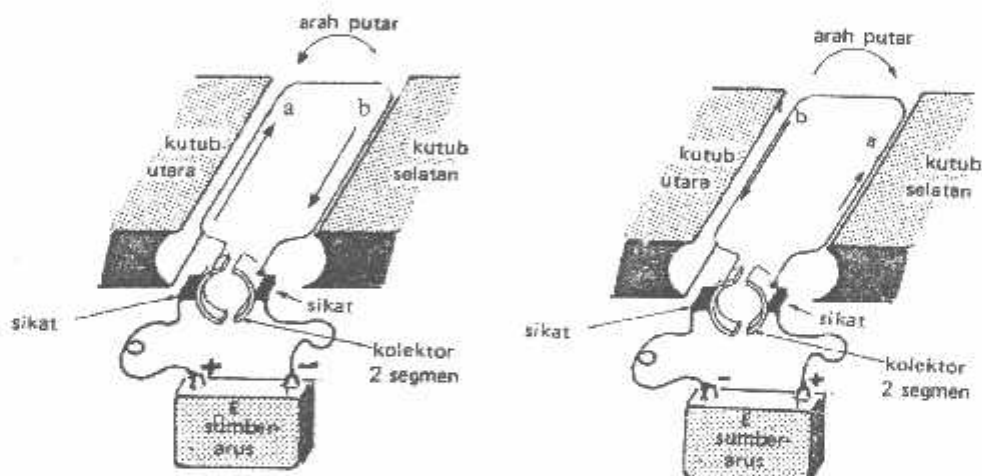
1. Jarak maksimal antara LED *infra red* dan *photodiode* agar sensor dapat bekerja dengan baik adalah ± 18 cm.
2. Tegangan keluaran rata-rata komparator ketika tidak ada halangan (tidak ada kereta api yang berhenti) sebesar 1,068 V dan tegangan keluaran *schmit trigger* 5,2 V. Apabila ada halangan (ada kereta api yang berhenti), maka tegangan keluaran rata-rata dari komparator sebesar 3,99 V dan tegangan keluaran dari *schmit trigger* 0,02 V.
3. Dari hasil percobaan sensor *limit*, peringatan kepada petugas operator bahwa ada kereta api yang akan memasuki stasiun berupa bunyi *buzzer* 3 kali dan indikator LED untuk mengetahui arah kedatangan kereta api masuk dari kanan atau kiri operator.

5.2 Saran

1. Untuk pengembangan lebih lanjut dapat ditambahkan penjadwalan kedatangan kereta api serta database sebagai data perjalanan kereta api.
2. Sensor yang digunakan dapat diganti dengan sensor yang lebih berkualitas.

Disini kolektor berfungsi bagaikan penyearah mekanik. *Flux* magnet yang ditimbulkan magnet permanen disebut medan magnetnya motor. Dalam gambar arah fluk magnetik adalah dari kiri ke kanan. Adapun gaya yang bekerja pada penghantar b adalah ke atas, sementara gaya yang bekerja pada penghantar a adalah ke bawah. Gaya-gaya yang bekerja sama kuatnya, jadi ada kopel yang bekerja pada kawat sehingga lilitan pun dapat berputar. Setelah berputar 90° arah arus berbalik, pada saat itu penghantar a dan penghantar b bertukar tempat. Akibatnya arah gerak putaran tidak berubah.

Pengendalian Arah Putaran Motor DC



Gambar 2.23. Arah Putaran Motor DC

Dari gambar 2.23. diatas, agar arah putaran motor dc berubah, maka polaritas tegangan pada baterai harus dibalik.

2.12. IC *Information Storage Device* (ISD) 2560

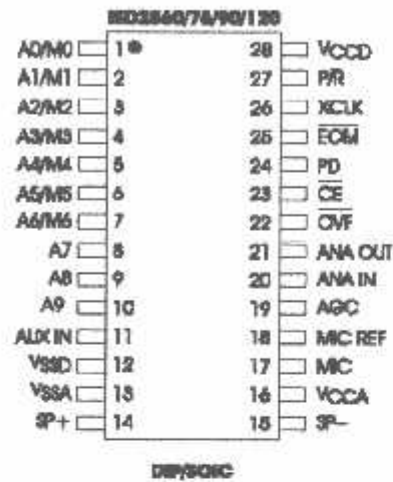
IC penyimpanan suara yang digunakan merupakan jenis EEPROM (*Electrically Erasable Programmable Read Only Memory*) yaitu ROM yang dapat diprogram, dihapus dan diprogram ulang secara elektrik dengan arus listrik, bukan sinar *ultraviolet*. IC ISD (*Information Storage Device*), yang dipakai yaitu ISD 2560. IC ini dapat merekam pesan maksimal 60 detik dan dapat dikaskade sehingga pesan yang disimpan dapat diperpanjang sesuai dengan keinginan kita dengan 160 alamat yang berbeda.

Didalam ISD 2560 dilengkapi dengan internal *amplifier*, *internal automatic gain control (AGC)*, *filter antialiasing* (perata) dan *speaker amplifier* (penguat speaker). Secara keseluruhan seri ISD 2560 dapat melakukan sebuah perekaman atau pemutaran ulang pesan dengan komponen sederhana seperti mikropon, *speaker*, beberapa komponen penunjang, dua buah saklar dan sumber tegangan.

Rekaman akan disimpan dalam sel memori yang tidak mudah hilang (*non volatile*), memberikan tempat penyimpanan yang masih kosong. Cara unik ini yang membuat ISD disebut *Direct Analog Storage Technology (DAST)* atau teknik penyimpanan analog langsung, dengan jalan sinyal suara (*voice*) dan bunyi disimpan secara langsung dalam bentuk analog, kedalam memori EPROM. Penyimpanan analog langsung memungkinkan reproduksi suara secara alami dalam satu chip tunggal.

Susunan ISD 2560 DAST adalah dikelompokkan dalam 160 segmen dari alamat A0 sampai A7 yang menunjukkan akses tiap segmen dalam kesatuan untuk

alamat pesan. Kemampuan pemberian atau penyediaan alamat yang berupa pesan yang disimpan dalam bentuk kalimat dan suara.



Gambar 2.24. Pin-pin IC ISD 2560

Berikut adalah konfigurasi pin-pin ISD 2560 :

❖ **Address Input (A0-A9) Pin 1-10**

Input alamat ini mempunyai dua fungsi, tergantung dari level dari dua *Most Significant Bits* (MSB) dari alamat. Jika dua MSB ini keduanya low, maka semua input digunakan sebagai bit pengalamatan (*Address Bits*) dan digunakan sebagai alamat untuk memulai (*Start Address*) dari perekaman atau pemutaran ulang (*Play Back*). Kaki-kaki dari pengalamat hanya merupakan masukan dan bukan merupakan informasi keluaran pengalamatan internal (*Output Internal Address Information*). Ketika proses operasional sedang berjalan dan pada saat kedua MSB ini high, maka sinyal input pengalamatan digunakan sebagai bits mode (*Mode Bit*) yang membuat mode operasi normal dan pengalamatan secara tidak langsung (*Simultaneously*).

❖ **V_{ssd} dan V_{ssa} (Ground) Pin 12 dan 13**

Sama seperti V_{ccd} dan V_{cca}, analog input dan digital sirkuit di dalam ISD 2560 menggunakan *bus ground* yang terpisah untuk meminimalisasi *noise*. Pin ini harus dihubungkan sedekat mungkin dengan *ground*.

❖ **Speaker Output (SP +, SP -) Pin 14 dan 15**

Pin SP + dan SP - digunakan untuk mengeluarkan suara yang telah direkam ke *speaker* atau ke *device* lainnya. Output ini mempunyai impedansi sebesar 16 Ohm.

❖ **Microphone Input (MIC) Pin 17**

Kaki mikropon ini terhubung dengan V_{cc} melalui beberapa kapasitor yang terhubung secara seri, bersamaan dengan resistor 10 K Ω yang berada didalam chip (*internal*). Harga dari kapasitor dari dalam perancangan ini menggunakan harga kapasitor sesuai dengan yang tertera dalam rangkaian data sheet ISD 2560.

❖ **Microphone Reference (MIC REF) Pin 18**

Ketika MIC REF menghubungkan antara V_{cc} dengan mikropon *ground*, maka tingkat *noise* selama perekaman dapat dikurangi. *Noise* itu disebabkan oleh *pre-amplifier* yang terdapat didalam *chip*. Bila pin ini tidak digunakan, maka tidak boleh dihubungkan dengan sinyal atau dengan tegangan apapun, harus dalam keadaan terbuka.

❖ **Automatik Gain Control (AGC) Pin 19**

Kegunaan dari AGC adalah untuk menambah atau mengurangi secara otomatis penguatan (*Gain*) dari *pre-amplifier* yang juga meluaskan

batas dari sinyal input yang dapat digunakan oleh mikropon tanpa terjadi distorsi. AGC ini dapat secara dinamis meluaskan batas dari suara yang terekam baik itu suara bisikan sampai suara yang keras. Untuk menggunakan fasilitas AGC ini, resistor dan kapasitor luar (eksternal) harus dihubungkan secara *parallel* antara pin AGC dengan *ground*. Harga yang direkomendasikan adalah $R = 470\text{ K}\Omega$ dan $C = 4,7\ \mu\text{F}$ (Dalam perancangan ini juga dipakai harga seperti diatas sama dengan data sheet ISD).

❖ **Analog Input (ANA IN) Pin 20**

Kapasitor eksternal (luar) menghubungkan antara ANA IN ke ANA OUT pin harga-harga dari kapasitor luar bersama dengan $3\text{ K}\Omega$ input impedansi di ANA IN dapat dipilih sendiri untuk memberikan keadaan *cut off* (terputus) pada frekuensi rendah sampai pada *pass band* suara. ANA IN juga dapat digunakan pada input sumber *alternative* dari sinyal analog pada sinyal mikropon terus ke kapasitor kopling.

❖ **Analog Output (ANA OUT) Pin 21**

Sinyal dari mikropon dikuatkan dan dikeluarkan melalui ANA OUT pin. Penguatan tegangan dari *pre-amp* tergantung dari tingkat tegangan AGC (*Automatik Gain Control*) pin. *Pre-amplifier* ini mempunyai penguatan maksimum sekitar 24 dB untuk tingkat masukan kecil.

❖ **Overflow (OVF) Pin 22**

Ketika ruang memory sudah habis maka OVF akan memberi sinyal *low*.

❖ **Chip Enable (CE) Pin 23**

Ketika sinyal ini berpindah dari *high* ke *low*, maka CE akan aktif. *Playback* akan berjalan sampai input ini tertekan *high*, tanda akhir dari pesan tercapai atau ruang memori sudah habis. ISD akan kembali ke mode *standby* setelah *playback* ini berhenti.

❖ **Power Down (PD) Pin 24**

Ketika tidak ada proses perekaman atau *playback*, PD dapat diberi input *high* sehingga ISD berada dalam kondisi *standby*. Ketika *overflow* (OVF), PD harus diberi pulsa *high* untuk *me-reset* alamat kembali ke awal. Pada mode *push button* maka PD dapat digunakan sebagai tombol *stop / reset*.

❖ **Record LED Output (RECLEL) Pin 25**

Selama proses perekaman output RECLEL akan *low*. Maka output ini bisa digunakan untuk menjalankan sebuah led yang berfungsi untuk mengetahui bahwa terjadi proses perekaman. Ketika tanda akhir dari pesan tercapai pada saat *playback*, maka RECLEL akan *low* sebentar.

❖ **Optimal External Clock (XCLK) Pin 26**

Digunakan untuk penambahan kristal clock bila dibutuhkan pewaktuan yang lebih besar dan presisi. Bila input ini tidak digunakan, harus dihubungkan dengan ground.

❖ **Record (REC) Pin 27**

Input sinyal REC akan aktif dalam kondisi *low*. ISD 2560 akan merekam bila REC dalam keadaan *low*, dan sinyal ini harus terus dalam keadaan *low* bila ingin terus merekam. Jika input REC ini

tertekan low dalam keadaan masih memutar ulang pesan (*playback*), maka *playback* akan berhenti dan ISD akan merckam.

❖ VCCA Dan VCCD Pin 16 Dan 28

Analog dan digital sirkuit yang terdapat didalam chip ISD 2560 menggunakan bus power yang terpisah untuk meminimalisasi noise. Pin power ini harus dihubungkan sedekat mungkin dengan sumber tegangan.

2.13. Mikrokontroller AT89S51

Perbedaan mendasar antara mikrokontroller dan mikroprosesor adalah mikrokontroller selain memiliki CPU juga dilengkapi dengan memori *I/O* yang merupakan kelengkapan sebagai suatu *minimum system* mikrokomputer sehingga mikrokontroller dapat dikatakan sebagai mikrokomputer dalam keping tunggal (*single chip microcomputer*) yang dapat berdiri sendiri.

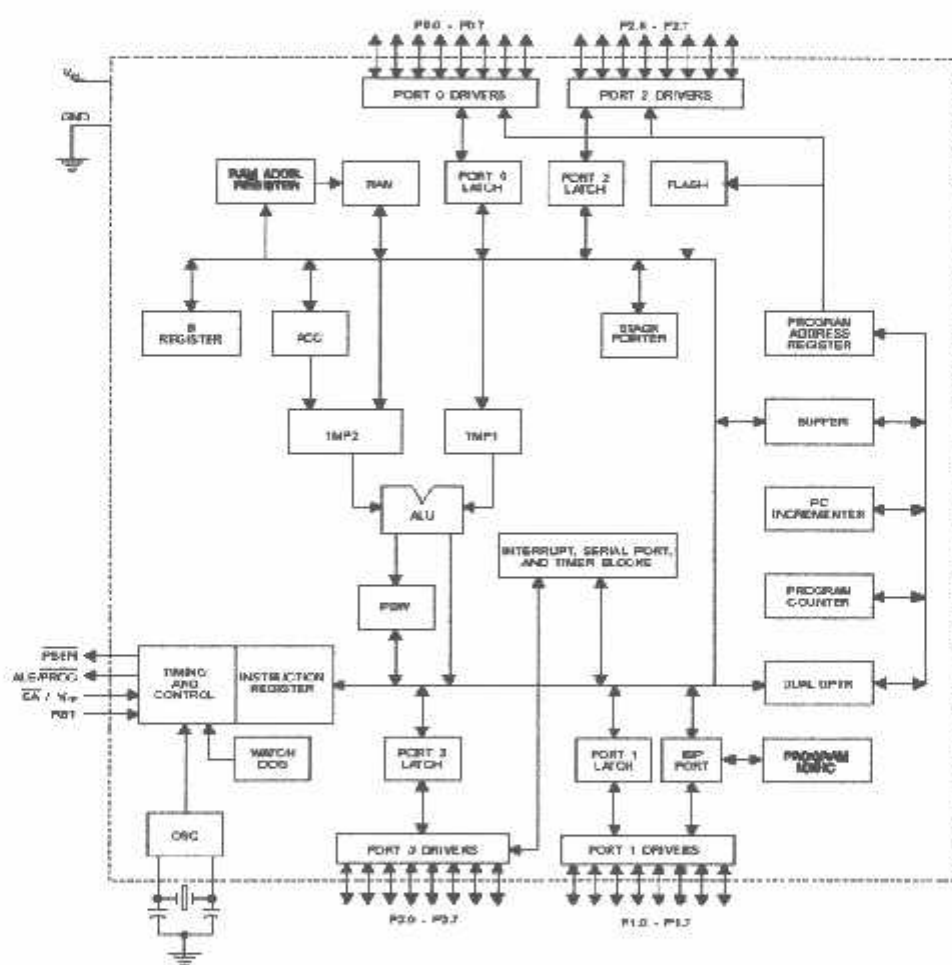
Mikrokontroller AT89S51 adalah mikrokontroller ATMEL *kompatibel* penuh dengan mikrokontroller keluarga MCS-51, dengan *supply* daya rendah, memiliki *performance* yang tinggi, dan merupakan mikrokontroller 8 bit yang dilengkapi 4 Kbyte EPROM (*Enable and programmable read Only Memory*) dan 128 Byte RAM *internal*. Program memori dapat diprogram ulang dalam sistem atau dengan menggunakan *Programmer Nonolatelly Memory Konvensional*.

Dalam sistem mikrokontroller terdapat dua hal yang mendasar, yaitu: perangkat keras dan perangkat lunak yang keduanya saling terkait dan mendukung.

2.13.1. Perangkat keras Mikrokontroler AT89S51

Mikrokontroler AT89S51 secara umum memiliki:

- CPU 8 bit
- *Memory*
- *Port I/O*
- *Timer dan Counter*
- *Sumber Interrupt*
- *Program Serial yang dapat diprogram*
- *Osilator dan Clock*



Gambar 2.25. Blok Diagram Mikrokontroler AT89S51

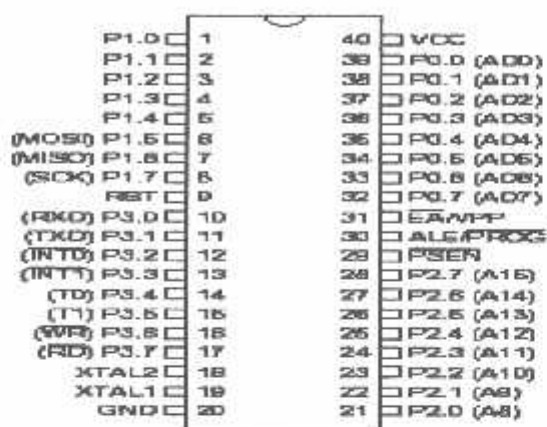
2.13.2. Arsitektur AT89S51

Arsitektur mikrokontroler AT89S51 adalah sebagai berikut :

1. CPU (*Central Processing Unit*) 8-bit dengan *register A (accumulator)* dan B
 2. 16-Bit *Program Counter (PC)* dan *Data Pointer (DPTR)*.
 3. 8- Bit *Program Status Word (PSW)*.
 4. 4-Bit *Stack Pointer (SP)*.
 5. 4 Kbyte *internal EPROM*.
 6. 128 byte *internal RAM*.
 - 4 *bank register*, masing-masing berisi 8 byte
 - 16 byte alamat serbaguna yang dapat diakses sebagai byte atau bit, tergantung *software* yang digunakan.
 - 80 byte *general purpose memory data*.
 7. 32 pin *input-output* tersusun atas P0-P3, masing-masing 8-bit.
 8. 2 buah 16-bit *Timer/Counter*.
 9. 2 buah *port serial full duplex*
 10. *Control Register*, antara lain : TCON, SCON, PCON, IP, dan IE
 11. 5 buah sumber interupsi (2 buah sumber interup eksternal dan 3 buah sumber *internal*).
 12. Osilator dan *Clock Internal*.
 13. *Watch Dog Programmable Timer*.
 14. *ISP Port* .
-

2.13.3. Konfigurasi Pin-pin Mikrokontroller

Konfigurasi kaki-kaki Mikrokontroller terdiri dari 40 pena (pin), seperti pada gambar dibawah ini :



Gambar 2.26. Konfigurasi Pin-pin AT89S51

Fungsi dari tiap-tiap pena adalah sebagai berikut :

1. VCC (suplay tegangan).
2. GND (*ground*).
3. Port 0.

Merupakan *port input* dua arah dan dikonfigurasikan sebagai *multipleks* dua bus alamat rendah (A0-A7) dan data selama pengaksesan program memori dan data internal.

4. Port 1.

Merupakan *port input* dua arah dengan pull-up dan juga menerima *Low-order address byte* selama memprogram dan verifikasi dari *flash*. Pada mikrokontroller AT89S51 port 1 memiliki 3 pin dengan fungsi khusus.

Tabel 2.1. Fungsi Khusus Port 1

Port Pin	Alternative Functions
P1.5	MOSI (<i>used for In-system Programming</i>)
P1.6	MOSI (<i>used for In-system Programming</i>)
P1.7	SCK (<i>used for In-Programming</i>)

5. Port 2

Merupakan port I/O dengan *internal pull-up*. Mengeluarkan *address* tinggi selama pengambilan (*fetching*) program memori eksternal. Selama pengaksesan ke eksternal data memori, port 2 mengeluarkan isi SFR (*Special Function Register*). Menerima *address* dan beberapa sinyal kontrol selama pemrograman.

6. Port 3

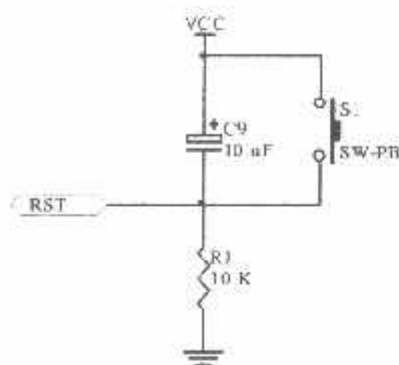
Merupakan port I/O dengan *internal pull-up*. Port 3 juga memiliki fungsi khusus, yaitu :

Tabel 2.2. Fungsi Khusus Port 3

Port Pin	Alternative Functions
P3.0	RXD (<i>serial input port</i>)
P3.1	TXD (<i>serial output port</i>)
P3.2	$\overline{INT0}$ (<i>external interrupt 0</i>)
P3.3	$\overline{INT1}$ (<i>external interrupt 1</i>)
P3.4	T0 (<i>timer 0 external input</i>)
P3.5	T1 (<i>timer 1 external input</i>)
P3.6	\overline{WR} (<i>internal data memory write strobe</i>)
P3.7	\overline{RD} (<i>external memory read strobe</i>)

7. Reset

Perubahan taraf tegangan dari rendah ke tinggi akan mereset AL 89S51.



Gambar 2.27. Rangkaian *Reset*

8. ALE/PROG

Pulsa output ALE digunakan untuk proses-proses *'latching' byte address* rendah (A0-A7) selama pengaksesan ke memori eksternal. Pin ini juga digunakan untuk memasukkan pulsa program (*prog*) selama pemrograman.

9. PSEN

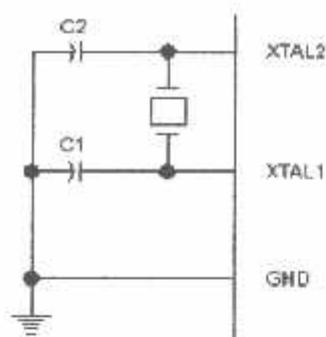
Merupakan *strobe* baca ke program memori eksternal.

10. EA/VPP

External Address Enable (EA) digroundkan jika mengakses memori eksternal, akan dihubungkan ke VCC jika digunakan untuk mengakses memori internal.

11. X-TALL 1 dan X-TALL 2

Kaki ini dihubungkan dengan kristal bila menggunakan osilator internal. XTALL 1 merupakan *input inverting osilator amplifier* sedangkan X-TALL 2 merupakan *output inverting osilator amplifier*.



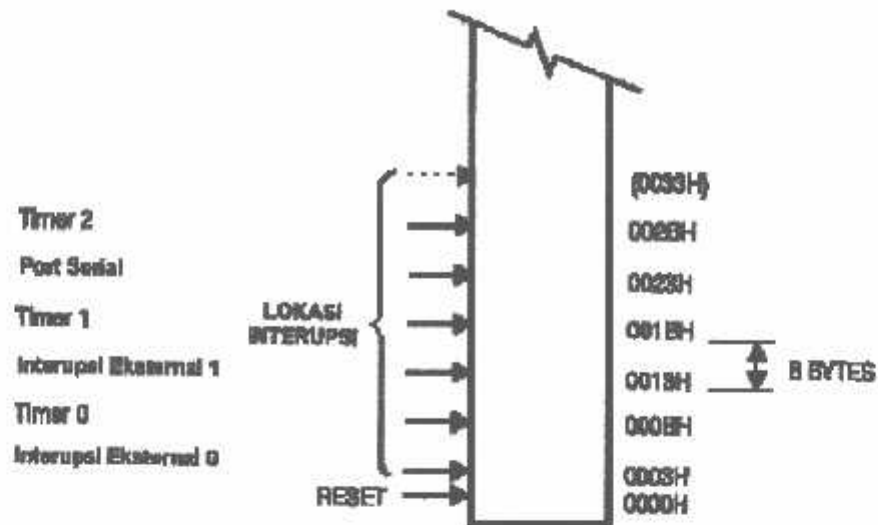
Gambar 2.28. Rangkaian *Clock*

2.13.4. Organisasi Memori

Organisasi memori pada mikrokontroler AT89S51 dapat dibagi menjadi dua bagian besar yaitu memori program dan memori data. Pembagian tersebut didasarkan atas fungsi dari penyimpanan data maupun program. Memori program digunakan untuk menyimpan instruksi-instruksi yang akan diajarkan oleh mikrokontroler, sedangkan memori data digunakan sebagai tempat menyimpan instruksi-instruksi yang sedang diolah mikrokontroler.

Program mikrokontroler disimpan dalam memori program berupa ROM. Mikrokontroler AT89S51 dilengkapi dengan ROM *internal*, sehingga untuk menyimpan program tidak digunakan ROM eksternal yang terpisah dari mikrokontroler. Agar tidak menggunakan memori program eksternal, EA (*Eksternal Address enable*) dihubungkan dengan Vcc.

Memori program mikrokontroler menggunakan alamat 16 bit mulai 0000_H-0FFF_H , sehingga kapasitas penyimpanan program maksimal adalah 4Kbyte. Sinyal / PSEN (*Program Strobe Enable*) tidak digunakan jika menggunakan memori *internal*.



Gambar 2.29. Memori Program

Selain program mikrokontroller AT 89S51 juga memiliki data internal 128 byte dan mampu mengakses memori data eksternal sebesar 64Kb. Semua memori data internal dapat dialamati dengan data langsung atau tidak langsung. Sebagai memori tersebut dapat dialamati dengan memori satu bit. Untuk membaca data digunakan sinyal / RD sedangkan untuk menulis data digunakan sinyal / WR.

2.13.5. SFR (*Special Function Register*)

Register Fungsi Khusus (*Special Function Register*) terletak pada 128 byte bagian atas memori data internal dan berisi *register-register* untuk pelayanan *latch port*, *timer*, *program status words*, *control peripheral* dan sebagainya. Alamat register fungsi khusus ditunjukkan pada tabel 2.3 dibawah ini:

Tabel 2.3. *Special Function Register*

Simbol	Nama Register	Alamat
ACC	<i>Accumulator</i>	E0 _H
B	<i>Register B</i>	F0 _H
PSW	<i>Program Status Word</i>	D0 _H
SP	<i>Stack Pointer</i>	81 _H
DPTR	<i>Data Pointer 2 Byte</i>	
DPL	Bit Rendah	82 _H
DPH	Bit Tinggi	83 _H
P0	<i>Port 0</i>	80 _H
P1	<i>Port 1</i>	90 _H
P2	<i>Port 2</i>	A0 _H
P3	<i>Port 3</i>	B0 _H
IP	<i>Interupt Priority Control</i>	D8 _H
IE	<i>Interupt Enable Control</i>	A8 _H
TMOD	<i>Timer/Counter Mode Control</i>	89 _H
TCON	<i>Timer/Counter Control</i>	88 _H
TH0	<i>Timer/Counter 0 High byte</i>	8C _H
TL0	<i>Timer/Counter 0 Low byte</i>	8A _H
TH1	<i>Timer/Counter 1 High byte</i>	8D _H
TL1	<i>Timer/Counter 1 Low</i>	8B _H
SCON	<i>Serial Control</i>	98 _H
SBUF	<i>Serial Data Buffer</i>	99 _H
PCON	<i>Power Control</i>	87 _H

Beberapa macam *register* fungsi khusus yang sering digunakan adalah sebagai berikut :

- *Accumulator* (ACC) merupakan *register* untuk penambahan dan pengurangan. Perintah *mnemonic* untuk mengakses akumulator disederhanakan sebagai A.
 - *Register B* merupakan *register* khusus yang berfungsi melayani operasi perkalian dan pembagian.
 - *Register R* merupakan delapan set *register* yang dinamakan R0, R1, R2, R3, R4, R5, R6 dan R7, fungsi dari *register-register* ini adalah sebagai *register* yang membantu penyimpanan data yang menggunakan banyak operasi. *Register-register* ini yang membantu akumulator dalam melakukan operasi antara dua operan.
 - *Stack Pointer* (SP) merupakan *register* 8 bit yang dapat diletakkan dialamat manapun pada RAM *internal*.
 - *Data Pointer* (DPTR) terdiri dari dua *register*, yaitu *register* untuk byte tinggi (*Data Pointer High*, DPH) dan *register* untuk byte rendah (*Data Pointer Low*, DPL) yang berfungsi untuk mengunci alamat 16 bit. DPTR berfungsi untuk menunjuk suatu lokasi data, namun pada beberapa perintah DPTR digunakan untuk mengakses memori eksternal.
 - *PC* (*Program Counter*) merupakan alamat 16 bit yang menginstruksikan AT89S51 alamat instruksi yang selanjutnya akan dilaksanakan. Saat inisialisasi AT89S51, PC terisi dengan 0000h dan akan bertambah satu setiap kali instruksi telah dilaksanakan. Harga PC tidak dapat langsung dirubah
-

dengan menggunakan perintah MOV PC,2340h, namun dengan perintah LJMP 2340 yang akan mengisi PC dengan 2340h..

- *Program Status Word (PSW)* berisi bit-bit status yang berkaitan dengan kondisi CPU saat itu. PSW terletak pada alamat D0H.

PSW		PSW.7	PSW.6	PSW.5	PSW.4	PSW.3	PSW.2	PSW.1	PSW.0
D0H		CY	AC	F0	RS1	RS0	OV	-	P

Gambar 2.30. Register PSW dalam Mikrokontroler AT89S51

1. CY (*Flag Carry*)

Flag carry, yang terletak pada alamat D7H, berfungsi sebagai pendeteksi terjadinya kelebihan pada operasi penjumlahan, atau terjadinya peminjaman (*borrow*) pada operasi pengurangan. Misalnya, jika data pada akumulator adalah FFH dan dijumlahkan dengan bilangan satu atau lebih, maka akan terjadi kelebihan sehingga akan membuat *carry* menjadi *set*. Demikian juga apabila data pada akumulator adalah 00H dan dikurangkan dengan bilangan satu atau lebih, akan terjadi peminjaman sehingga membuat *carry* juga menjadi *set*.

2. AC (*Flag Auxiliary Carry*)

Flag auxiliary carry akan selalu dalam kondisi *set* apabila pada saat proses penjumlahan terjadi *carry* dari bit ketiga hingga bit keempat.

3. Flag 0

Flag 0 dapat digunakan untuk tujuan umum tergantung pada kebutuhan pemakai.

4. RS (*Register Select*)

Bit Pemilih *Bank Register (Register Bank Select Bits)* RS0 dan RS1 digunakan untuk menentukan lokasi dari *bank register (R0-R7)* pada memori. RS0 dan RS1 selalu bernilai 0 setiap kali sistem di reset sehingga lokasi dari *register R0* hingga *R7* akan berada pada alamat 00H hingga 07H.

5. OV (*Flag Overflow*)

Flag overflow akan berada pada kondisi set jika pada operasi aritmatik menghasilkan bilangan yang lebih besar daripada 128 atau lebih kecil dari -128.

6. P (bit paritas)

Bit paritas akan berada pada kondisi set jika jumlah bit 1 dalam akumulator adalah ganjil dan akan berada pada kondisi *clear* jika jumlah bit 1 dalam akumulator adalah genap. Misalnya, data yang tersimpan pada akumulator adalah 10101110b atau AEH maka *parity bit* akan berada pada kondisi *set*. Data AEH mempunyai lima bit yang berkondisi 1 atau dapat disebut mempunyai bit 1 dalam jumlah yang ganjil.

- *Port 0* sampai *Port 3* merupakan *register* yang berfungsi untuk membaca dan mengeluarkan data pada port 0,1,2 dan 3. Masing-masing *register* ini dapat dialamati per byte maupun per bit.
- *Control Register* terdiri dari register yang mempunyai fungsi kontrol. Untuk mengontrol sistem interupsi, terdapat dua *register* khusus yaitu *register IP (Interrupt Priority)* dan *register IE (Interrupt Enable)*. Untuk mengontrol *timer/counter* terdapat dua *register* khusus yaitu register TCON (*Timewr*

Counter Control) serta *port serial* menggunakan *register SCON (Serial Port Control)*.

2.13.6. Metode Pengalamatan

1). Pengalamatan Langsung

Pengalamatan langsung dilakukan dengan memberikan nilai ke suatu *register* secara langsung. Untuk melaksanakan pengalamatan langsung digunakan tanda #.

Contoh : MOV A, #0A

2). Pengalamatan Tak Langsung

Operand pengalamatan tak langsung menunjuk ke sebuah *register* yang berisi lokasi alamat memori yang akan digunakan dalam operasi. Lokasi yang nyata tergantung pada isi *register* saat instruksi dijalankan. Untuk melaksanakan pengalamatan tak langsung digunakan simbol @.

Contoh : ADD A, @RO

2.14. Mikrokontroler Renesas R8C/13 Tiny (R5F21134FP)

Renesas Technology adalah produsen semikonduktor tingkat internasional. Renesas terbangun dari gabungan dua produsen semikonduktor, yaitu Mitsubishi dan Hitachi. Sebagai produsen semikonduktor, renesas juga mengeluarkan berbagai jenis keluarga mikrokontroler (MK).

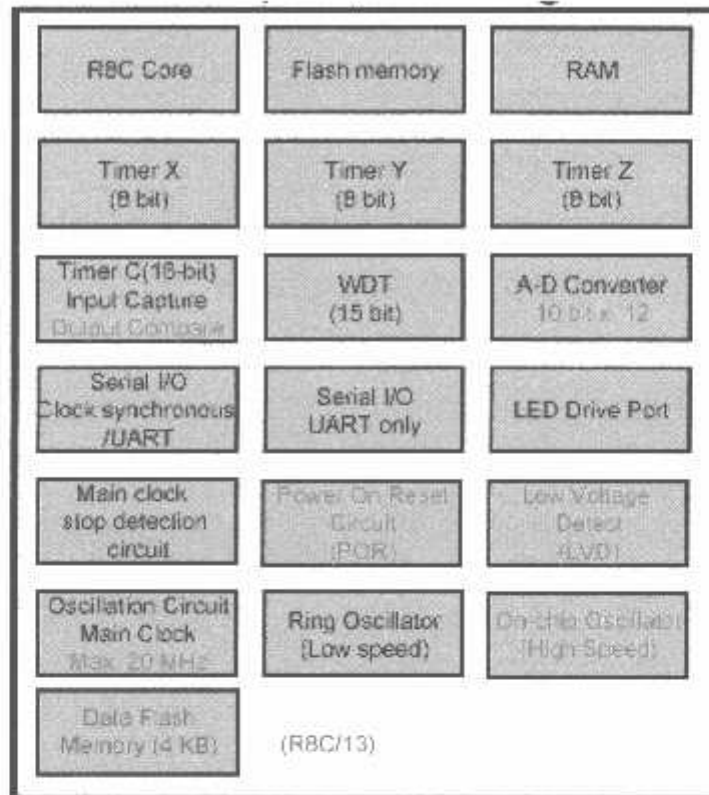
Renesas R8C adalah salah satu jenis seri dalam keluarga MK M16C. CPU R8C sama dengan CPU CISC 16-bit M16C, hanya saja lebar jalur data R8C adalah 8-bit. Karena menggunakan CPU yang sama maka R8C memiliki *instruction set* hampir sama dengan M16C. Perbedaannya hanya terletak pada 2

instruksi, yaitu R8C tidak memiliki instruksi JMPS (*Jump Special Page*) dan JSRS (*Jump Subroutine Special Page*). R8C/13 adalah salah satu tipe MK dalam seri R8C. MK ini memiliki kemasan 32-pin LQFP. Dalam perancangan pada skripsi ini menggunakan menggunakan MK seri R5F21134, yaitu R8C/13 yang memiliki Flash ROM 16 KB (1000 E/W *cycles*) dan RAM sebesar 1 KB.

2.14.1. Spesifikasi R5F21134FP

Berikut ini adalah spesifikasi *R5F21134FP* dengan peta peripheral dan memori-memorinya.

- ❖ Mempunyai *CPU Core* (16-bit) 1 – 20 MHz, 3.0 – 5.5 Volt dan 1 – 10MHz 2.7 – 5.5 Volt.
 - ❖ Rangkaian Clock, kecepatan *Low/High On-Chip Oscillator*, Clock utama dengan *Xin/Xout*.
 - ❖ Memory (ROM/SRAM) 16 Kbytes / 1 Kbytes, 2 x 2 K Bytes Data Flash pada R8C/12, 13.
 - ❖ Kemasan 32 pin LQFP (7mm x 7mm)
-



Gambar 2.31. Blok Diagram R8C/13 dan Peta *Peripheral*-nya

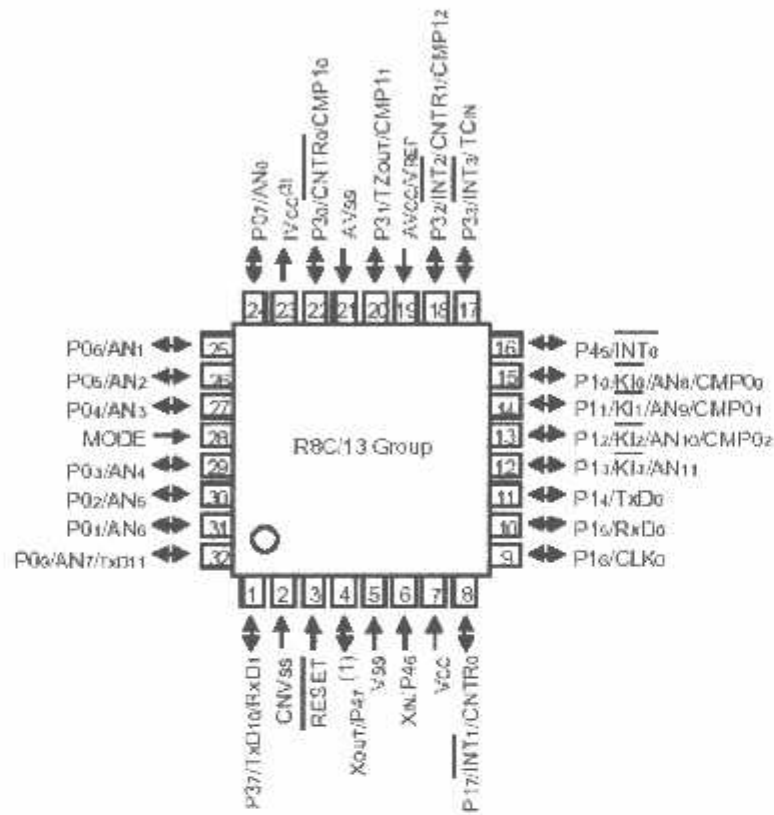
2.14.2. Kelebihan Kunci R8C/Tiny

Banyak kelebihan-kelebihan yang dimiliki R8C/Tiny diantaranya adalah :

- ❖ Kompatibel dengan M16C yaitu kompatibel dalam instruksi dan kode.
- ❖ *Peripheral* lebih terintegrasi jadi lebih hemat.
- ❖ *Electromagnetic Compatibility* (EMC) mempunyai EMI rendah, EMS tinggi.
- ❖ *Development Tool* (*Compiler* dan *Debugger*) didapat dengan murah dan difasilitasi *On-Chip Debugger*
- ❖ Mempunyai fitur *fail-safe* yaitu pengamanan terhadap kegagalan sistem.
- ❖ Konsumsi daya rendah.
- ❖ 16-bit CISC CPU dengan kecepatan maksimal 20 MHz (1:1).

- ❖ 89 instruksi CISC lebih hemat ROM kira-kira 20 %, RAM sampai 1 KB.
- ❖ Waktu konversi ADC hanya 3 μ S.

2.14.3. Konfigurasi Pin R8C R5F21134FP



Gambar 2.32. Konfigurasi Pin R8C R5F21134FP

Gambar diatas adalah konfigurasi pin-pin dari R8C R5F21134FP untuk lebih jelasnya dapat diamati pada tabel dekripsi pin-pin berikut ini :

Tabel 2.4. Kofigurasi Pin-pin dari R8C R5F21134FP

Nama Sinyal	Nama Pin	Type I/O	Fungsi
Masukan Catu Daya	Vcc, Vss	I	Tegangan 2.7 V – 5.5 V pada pin Vcc. Tegangan 0 V pada Vss pin
I Vcc	Ivcc	O	Pin ini untuk men-stabilkan catu daya <i>internal</i> . pin ini dihubungkan pada Vss melalui kapasitor 100nF. Jangan dihubungkan pada Vcc.
Input Catu Daya Analog	Avcc, Avss	I	Ini adalah untuk catu daya pada ADC. Avcc dihubungkan pada Vcc, A Vss dihubungkan ke Vss. Dianjurkan untuk menghubungkan kapasitor diantara pin A Vcc dan A Vss.
Input Reset	RESET	I	"1." untuk masukan ini mereset MCU
CNVss	CNVss	I	Pin ini dihubungkan pada Vss melalui sebuah resistor.
MODE	MODE	I	Pin ini dihubungkan pada Vcc melalui sebuah resistor.
Input Clock Utama	Xin	I	Pin-pin ini disediakan untuk membangkitkan rangkaian I/O Clock Utama. Dihubungkan dengan sebuah keramik resonator atau kristal diantara pin Xin dan Xout. Jika digunakan clock internal maka pin Xin dan Xout dalam keadaan terbuka.
Output Clock Utama	Xout	O	
Input Interupsi	INT0 –INT3	I	Pin ini sebagai masukan interupsi.

Input Kunci Interupsi	KI0 – KI3	I	Pin ini sebagai masukan kunci interupsi.
Timer X	CNTR 0	I/O	Pin I/O ini adalah untuk Timer X .
	CNTR 0	O	Pin Ouput untuk Timer X.
Timer Y	CNTR 1	I/O	Pin I/O untuk Timer Y.
Timer Z	TZout	O	Pin Ouput untuk Timer Z.
Timer C	TC in	I	Pin Input untuk Timer C.
	CMP00 – CMP03, CMP10 CMP13	O	Pin Output untuk Timer C.
Serial Interface	CLK 0	I/O	Pin I/O untuk memindahkan Clock.
	RXD0, RXD1	I	Pin input untuk data Serial.
	TXD0, TXD10, TXD11	O	Pin output untuk data Serial.
Input Tegangan Referensi	Vref	I	Tegangan referensi input ini untuk ADC. Vref pin dihubungkan ke Vcc.
ADC, pengubah dari analog ke digital	AN0–AN11	I	Pin analog input pada ADC.
Port I/O	P00-P07, P10-P17, P30-P33, P37, P45	I/O	Merupakan port I/O CMOS 8-bit . Setiap port mempunyai pilihan register pengarah sebagai input atau output. Tiap Port dapat dialamati per bit. Dapat di-set menggunakan pull up resistor dengan program. P10 – P17 mempunyai driver

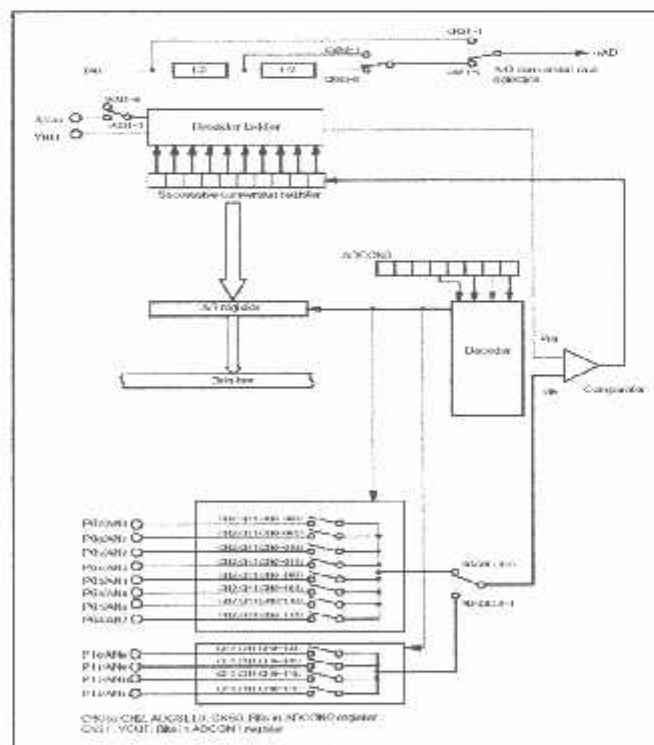
			transistor.
Port Input	P46, P47	I	Pin ini hanya bisa digunakan sebagai input.

2.14.4. *Peripheral R8C R5F21134FP*

Mikrokontroler *R8C R5F21134FP* mempunyai beberapa *peripheral-peripheral* yang banyak digunakan pada beberapa aplikasi-aplikasi penting, diantaranya adalah sebagai berikut :

❖ *Analog To Digital Converter (ADC)*

Dengan 12 SAR ADC S/H yang mempunyai resolusi 8-bit atau 10-bit. Mode Operasinya menggunakan *One-Shot* dan *Repeat* dengan waktu konversi 2.8 μ S (pada clock 10 MHz). Berikut gambar diagram blok ADC *built in* pada mikrokontroler ini :



Gambar 2.33. Diagram Blok ADC

❖ *Timer Mode*

Mempunyai timer sebanyak 4 yaitu timer X, Y, Z, C. Berikut adalah mode-mode timernya :

Tabel 2.5. Mode-mode *Timer*

Item		Timer X	Timer Y	Timer Z	Timer C
Configuration		8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	16-bit timer
Count		Down	Down	Down	Up
Count source		f1 f2 f8 f32	f1 f8 fRING Input from CNTR1 pin	f1 f2 f8 Timer Y underflow	f1 f8 f32
Function	Timer mode	provided	provided	provided	not provided
	Pulse output mode	provided	not provided	not provided	not provided
	Event counter mode	provided	provided ¹	not provided	not provided
	Pulse width measurement mode	provided	not provided	not provided	not provided
	Pulse period measurement mode	provided	not provided	not provided	not provided
	Programmable waveform generation mode	not provided	provided	provided	not provided
	Programmable one-shot generation mode	not provided	not provided	provided	not provided
	Programmable wait one-shot generation mode	not provided	not provided	provided	not provided
	Capture	not provided	not provided	not provided	provided
Input pin		CNTR0	CNTR1	INT0	TCN
Output pin		CNTR0 CNTR0	CNTR1	TZOUT	not provided
Related interrupt		Timer X int INT1 int	Timer Y int INT2 int	Timer Z int INT0 int	Timer C int INT3 int
Timer stop		provided	provided	provided	provided

❖ *Low Voltage Detect (LVD)*

LVD adalah untuk mendeteksi Vcc krang dari 3.8 V (± 0.5 V)

❖ *Watchdog Timer*

Watchdog berfungsi untuk mendeteksi ketika program diluar kontrol.

❖ *On Chip Debugger*

Fasilitas ini mempunyai fungsi untuk dapat di-*debug* pada waktu mikro sedang berjalan. Antara PC dan MK dapat berkomunikasi, PC akan mengetahui aktivitas MK saat itu. Syarat-syarat *On Chip Debugger* adalah:

- ❑ Vektor *Address Match interrupt* harus dihindari.
- ❑ *Single step interrupt* tidak dapat digunakan bersamaan interrupt lain.
- ❑ *UART1* tidak boleh dipakai.
- ❑ Instruksi BRK tidak boleh dipakai.
- ❑ Flash Address C000H – C7FFH.
- ❑ PD 3.7 harus "0".
- ❑ B5 FMR 0 harus "1"
- ❑ Menyiapkan 8 Byte untuk Stack.
- ❑ *On Chip Debugger* berpengaruh pada *timing run*.

❖ **Rangkaian Osilator**

Pada osilator utama menggunakan kristal luar sampai dengan 20 MHz, dengan memiliki fitur *Clock Stop Detect*. Kemudian untuk *On Chip Osilator* disediakan kecepatan *Low* 125 KHz dan *High* 8 MHz. Saat setelah reset, default clock adalah kecepatan rendah *On Chip osilator* 125 KHz.

2.15. Komunikasi RS-232

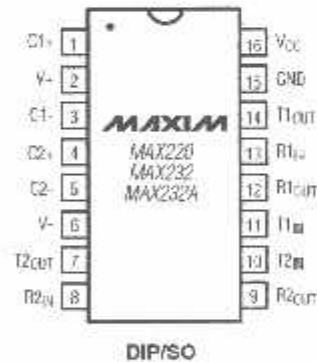
RS 232 merupakan kependekan dari *Recommended Standart number 232* yang merupakan rekomendasi dari EIA (*Electronic Industry Assosiation*), untuk interface antara peralatan terminal data dan peralatan komunikasi data, dengan menggunakan data biner serial sebagai data yang ditransmisikan.

2.15.1. IC MAX 232

IC Max 232 merupakan konverter tegangan dari level TTL CMOS ke level RS 232. IC Max 232 ini mempunyai empat buah bagian konverter yaitu dua buah driver receiver dan dua buah driver transmitter.

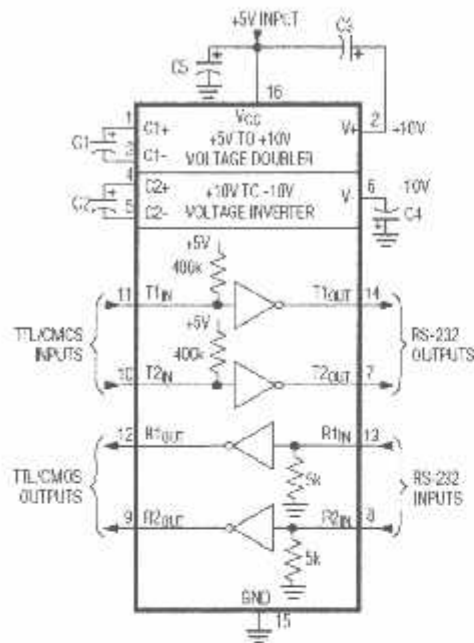
Level sinyal yang digunakan dimana *logic 0 (low)* dinyatakan sebagai tegangan antara +3 Volt sampai +10 Volt dan *logic 1 (high)* dinyatakan sebagai tegangan antara -3 Volt sampai -10 Volt.

Dalam standar TTL *logic 0 (low)* dinyatakan sebagai tegangan antara 0 Volt sampai 0.8 Volt, dan *logic 1 (high)* dinyatakan sebagai tegangan antara 3.5 Volt sampai 5 Volt.



Gambar 2.34. Konfigurasi Pin IC MAX232

Rangkaian dasar dari MAX 232 dapat dilihat pada gambar berikut ini:



Gambar 2.35. Rangkaian Operasi MAX 232

2.16. LCD (Liquid Crystal Display) M1632

LCD *Display Module* M1632 buatan Seiko Instrument Inc. terdiri dari dua bagian, yang pertama merupakan panel LCD sebagai media penampil informasi dalam bentuk huruf/angka dua baris, masing-masing baris bisa menampung 16 huruf/angka.

Bagian kedua merupakan sebuah sistem yang dibentuk dengan mikrokontroler yang ditempelkan dibalik pada panel LCD, berfungsi mengatur tampilan informasi serta berfungsi mengatur komunikasi M1632 dengan mikrokontroler yang memakai tampilan LCD itu. Dengan demikian pemakaian M1632 menjadi sederhana, sistem lain yang M1632 cukup mengirimkan kode-

kode ASCII dari informasi yang ditampilkan seperti layaknya memakai sebuah printer.

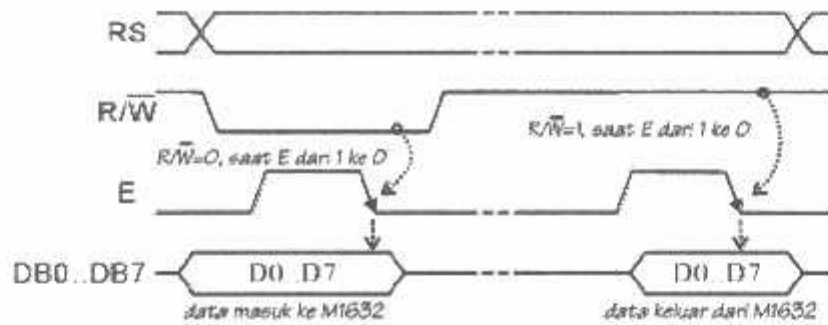
2.16.1. Sinyal interface M1632

Untuk berhubungan dengan mikrokontroler pemakai, M1632 dilengkapi dengan 8 jalur data (**DB0..DB7**) yang dipakai untuk menyalurkan kode ASCII maupun perintah pengatur kerjanya M1632. Selain itu dilengkapi pula dengan **E**, **R/W** dan **RS** seperti layaknya komponen yang kompatibel dengan mikroprosesor. Kombinasi lainya **E** dan **R/W** merupakan sinyal standar pada komponen buatan Motorola. Sebaliknya sinyal-sinyal dari MCS51 merupakan sinyal khas Intel dengan kombinasi sinyal **WR** dan **RD**.

RS, singkatan dari *Register Select*, dipakai untuk membedakan jenis data yang dikirim ke M1632. kalau **RS=0** data yang dikirim adalah perintah untuk mengatur kerja M1632, sebaliknya kalau **RS=1** data yang dikirim adalah kode ASCII yang ditampilkan.

Demikian pula saat pengambilan data, saat **RS=0** data yang diambil dari M1632 merupakan data status yang mewakili aktivitas M1632, dan saat **RS=1** maka data yang diambil merupakan kode ASCII dari data yang ditampilkan.

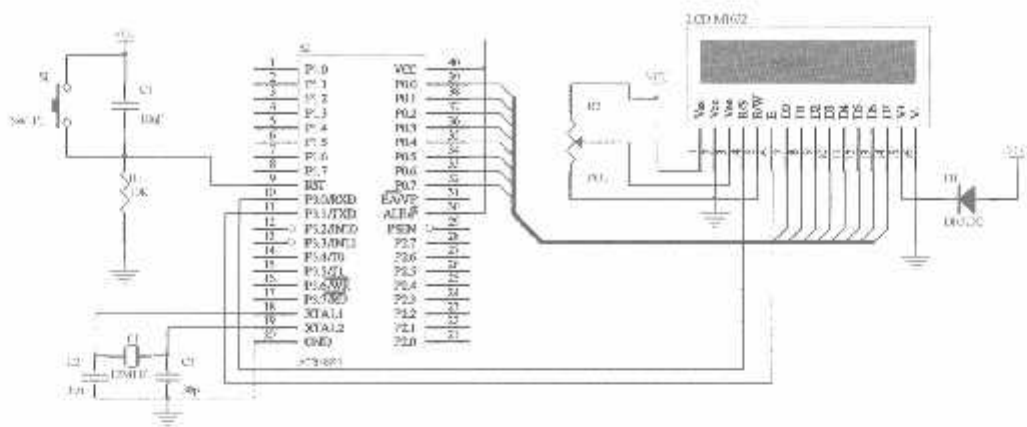
Proses mengirim/mengambil data ke/dari M1632 digambarkan dalam gambar 2.34. bisa dijabarkan sebagai berikut :



Gambar 2.36. Mengirim/Mengambil Data Ke/Dari M1632

1. RS harus dipersiapkan dulu, untuk menentukan jenis data seperti yang telah dibicarakan di atas.
2. R/W di-nol-kan untuk menandakan akan diadakan pengiriman data ke M1632. Data yang akan dikirim disiapkan di DB0..DB7, sesaat kemudian sinyal E di-satu-kan dan di-nol-kan kembali. Sinyal E merupakan sinyal sinkronisasi, saat E berubah dari 1 menjadi 0 data di DB0 .. DB7 diterima oleh M1632.
3. Untuk mengambil data dari M1632 sinyal R/W di-satu-kan, menyusul sinyal E di-satu-kan. Pada saat E menjadi 1, M1632 akan meletakkan datanya di DB0 .. DB7, data ini harus diambil sebelum sinyal E di-nol-kan kembali.

2.16.2. Interface Ke MCS-51



Gambar 2.37. Hubungan M1632 ke MCS-51

Sinyal-sinyal M1632 yang mengikuti standar teknik interface Motorola, tidak sesuai dengan sinyal dari MC-S51, dengan demikian sinyal-sinyal itu disimulasikan melalui port MCS51.

Sebagai contoh gambar 2.37 memperlihatkan hubungan AT89S51 dengan M1632, dalam gambar tersebut P3.0 dipakai untuk mensimulasikan sinyal RS, dan P3.1 sebagai E. Lewat program dibangkitkan sinyal-sinyal pada ketiga kaki Port 0 ini, sesuai dengan pesyaratan yang dikehendaki M1632.

Potongan Program 1 merupakan sub-rutin untuk mengendalikan M1632 yang dihubungkan ke AT89S51 seperti terlihat di gambar 2.37, sebelum sub-rutin ini dipakai, tepatnya pada saat setelah reset harus dikirimkan perintah CLR E.

Potongan program 1 terdiri dari dua bagian, yakni bagian mengirim data ke M1632 yang terdiri dari sub-rutin **KirimPerintah** dan sub-rutin **KirimASCII**, sedangkan bagian mengambil data dari M1632 terdiri dari sub-rutin **AmbilStatus** dan sub-rutin **AmbilASCII**.

Sebelum pengiriman data, Akumulator A sudah terlebih dulu diisi dengan data yang akan dikirim. Data yang dikirim dengan sub-rutin **KirimPerintah** akan diterima M1632 sebagai perintah untuk mengatur kerja M1632, dan data yang dikirim dengan sub-rutin **KirimASCII** akan ditampilkan di panel LCD.

Setelah M1632 menerima data, M1632 memerlukan waktu antara 40 sampai 1640 mikro-detik untuk mengolahnya, selama waktu itu M1632 untuk sementara tidak bisa menerima data, hal ini ditandai dengan bit 7 dari Register Status – '1'.

Proses pengiriman data ke M1632 dijelaskan sebagai berikut:

1. Perbedaan sub-rutin **KirimPerintah** dan **KirimASCII** terletak pada nilai **RS** pada saat sub-rutin itu bekerja. Sub-rutin **KirimPerintah** bekerja dengan **RS='0'** (baris 6), data yang dikirim AT89C2051 diterima M1632 sebagai perintah untuk mengatur kerja M1632. Sub-rutin **KirimASCII** bekerja dengan **RS='1'** (baris 10), data yang dikirim AT89C2051 diterima M1632 sebagai kode ASCII yang akan ditampilkan
2. Sinyal **RW** di-nol-kan agar M1632 siap menerima data (baris 12), setelah itu data di akumulator A diletakkan di **D0..D7** (Port 1 dari AT89C2051) di baris 13, baris 14 dan 15 membangkitkan sinyal sinkronisasi **E** dengan cara membuat **P3.4** menjadi '1' dan kemudian kembali menjadi '0'. Saat sinyal **E** kembali menjadi '0' data di Port 1 akan diterima oleh M1632.
3. Selesai mengirim data, program harus menunggu sampai M1632 siap menerima data lagi. Hal ini dilakukan dengan cara mengambil Status M1632 (baris 18), dan memeriksa bit 7-nya (baris 19), selama bit 7

bernilai '1' berarti M1632 masih sibuk mengurus diri, dan program menunggunya di **TungguDulu**.

Proses pengambilan data dari M1632 dijelaskan sebagai berikut :

4. Seperti bahasan di atas, **RS** dipakai untuk memilih Register Perintah/Status, sub-rutin **AmbilStatus** bekerja dengan **RS=0** dan sub-rutin **AmbilASCII** bekerja dengan **RS='1'**.
 6. Sinyal **RW** di-satu-kan agar M1632 siap memberi data (baris 29), setelah sinyal **E** menjadi '1' (baris 30) M1632 akan meletakkan data di **D0 .. D7**, setelah data ini diambil (baris 31) sinyal **E** dikembalikan menjadi '0'

Berikut adalah Potongan Program 1 AT89S51 dengan M1632 :

```

01:  E    bit  P3.1      ; sinyal E di P3.1
02:  RS   bit  P3.0      ; sinyal RS di P3.0
03:  ;Comment In Here
04:  KirimPerintah:
05:  CLR  RS              ; RS=0 : register perintah
06:  SJMP OutByte
07:  ;
08:  KirimASCII:
09:  SETB RS              ; RS=1 : Display Data RAM
10:  OutByte:
11:  CLR  RW              ; RW = '0', kirim data
12:  MOV  P1,A           ; siapkan data di D0..D7

```

```

13: SETB E           ; buat pulsa positif
14: CLR E           ; sesaat
15: ;
16: TungguDulu:
17: ACALL AmbilStatus
18: JB A.7,TungguDulu
19: RET
20: ;
21: AmbilStatus:
22: CLR RS           ; RS=0 : register status
23: SJMP InByte
24: ;
25: AmbilASCII:
26: SETB RS          ; RS=1 : Display Data RAM
27: InByte:
28: SETB RW          ; RW = '1', ambil data
29: SETB E           ; minta data pada M1632
30: MOV A,P1         ; ambil data
31: CLR E           ; kembalikan E ke '0'
32: RET

```

2.16.3. Mengatur tampilan M1632

M1632 mempunyai seperangkat perintah untuk mengatur tata kerjanya, perangkat perintah tersebut meliputi perintah untuk menghapus tampilan, meletakkan kembali *cursor* pada barishuruf pertama baris pertama,

menghidup/matikan tampilan dan lain sebagainya, semua itu dibahas secara terperinci dalam Lembar Data M1632.

Setelah diberi sumber daya, ada beberapa langkah persiapan yang harus dikerjakan dulu agar M1632 bisa dipakai, langkah-langkah tersebut antara lain adalah:

1. Tunggu dulu selama 15 mili-detik atau lebih.
2. Kirimkan perintah 30h, artinya transfer data antar M1632 dan mikrokontroler dilakukan dengan mode 8 bit
3. Tunggu selama 4.1 mili-detik
4. Kirimkan sekali lagi perintah 30h
5. Tunggu lagi selama 100 mikro-detik

Setelah langkah-langkah tersebut di atas M1632 barulah bisa menerima data dan menampilkannya dengan baik. Pada awalnya tampilan akan nampak kacau, dengan demikian perlu segera dikirim perintah menghapus tampilan dan lain sebagainya, sesuai dengan petunjuk yang ada di Lembar Data.

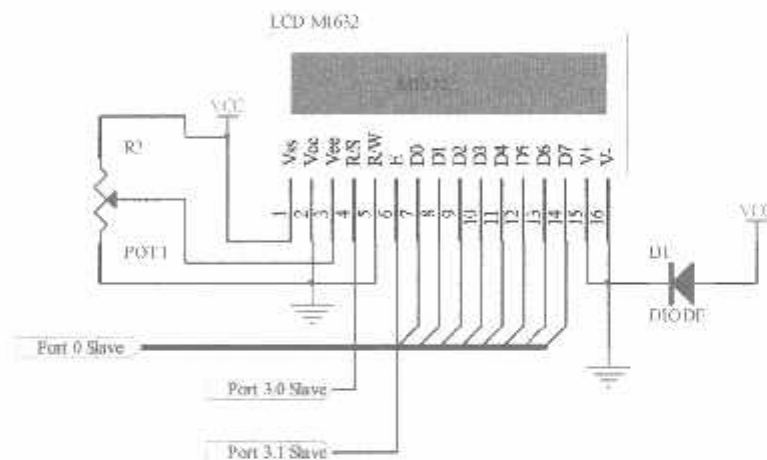
Di atas dipakai AT89C2051 sebagai contoh, meskipun demikian semua yang dibahas di atas sepenuhnya bisa dipakai pada mikrokontroler MCS 51. Dalam pemakaiannya karena berbagai macam alasan, bisa saja sinyal **E** dan **RS** tidak disimulasikan di **P3.1** dan **P3.0**. Hal ini bisa diselesaikan dengan melakukan beberapa penyesuaian, yakni tentukan dulu perubahan rangkaian sesuai dengan keadaan yang ada, dan perubahan rangkaian itu harus di sesuaikan di baris 1 sampai 3 pada potongan program di atas.

M1632 mempunyai 8 jalur data dan memerlukan 3 jalur kontrol, dalam suatu rangkaian yang memakai banyak port dari MC-S51, bisa terjadi kekurangan

port untuk menghubungkan MCS51 ke M1632. Jika sampai terjadi hal semacam ini bisa ditempuh hal hal berikut :

1. M1632 dipakai dalam mode data 4 bit, yakni hanya memakai jalur data **D0..D3**
2. Dengan sedikit tambahan rangkaian sinyal **WR** dan **RD** diubah menjadi sinyal **E** dan **R/W** gaya Motorola, sehingga tidak perlu menyediakan port untuk men-simulasikan sinyal-sinyal tersebut.

Berikut adalah gambar rangkaian LCD dengan komponen-komponen pendukung dengan pin-pin yang akan dihubungkan pada mikrokontroller MCS 51



Gambar 2.38. Rangkaian LCD M1632

LCD M1632 mempunyai spesifikasi sebagai berikut :

1. Memiliki 16 karakter dan dua baris tampilan yang terdiri dari 5 x 7 dot matrik ditambah dengan kursor.
2. Pembangkit karakter ROM untuk 192 jenis karakter.
3. Pembangkit karakter RAM untuk 8 jenis karakter.
4. 80 x 8 display data RAM (max 80 karakter).

5. Isolator didalam modul.
6. Memerlukan catu daya \pm volt.
7. Otomatis reset saat catu daya dinyalakan.

LCD modul M1632 mempunyai 16 pin dengan fungsi sebagai berikut :

Tabel 2.6. Fungsi Pin-pin LCD

No. PIN	Nama PIN	Fungsi
1	Vss	Terminal Ground
2	Vcc	Tegangan Catu + 5 volt
3	Vee	Mengendalikan kecerahan LCD
4	RS	Sinyal pemilihan register. 0 = Tulis 1 = Baca
5	R/W	Sinyal seleksi tulis atau baca 0 = Tulis 1 = Baca
6	E	Sinyal operasi awal yang mengaktifkan data tulis atau baca
7 - 14	DB0 - DB7	Merupakan saluran data berisi perintah data yang akan ditampilkan
15	V + BL	Back Light Supply 5 Volt (Volt)
16	V - BL	Back Ligth Supply 0 (Ground)

Pada LCD juga terdapat instruksi – instruksi sebagai berikut :

- ❖ *Display clear* : membersihkan tampilan yang ada pada LCD serta menyimpan, sedangkan kursor kembali ke posisi semula.
- ❖ *Cursor home* : hanya membersihkan tampilan dan kursor kembali ke semula.

❖ *Empty mode Set* : layar beraksi sebagai tampilan tulis.

S : 1/0 = menggeser layar.

I/0 : 1 = kursor bergerak ke kanan dan layar bergerak ke kiri.

I/0 : 0 = kursor bergerak ke kiri dan layar bergerak ke kanan

❖ *Display On/Off kontrol.*

D : 1 = layar on

D : 0 = layar off

C : 1 = kursor on

C : 0 = kursor off

B : 1 = kursor berkedip-kedip

B : 0 = kursor tidak berkedip – kedip

❖ *Cursor Display Shift*

S/C : 1 = LCD diidentifikasi sebagai layar

S/C : 0 = LCD diidentifikasi sebagai kursor

R/L : 1 = menggeser satu spasi ke kanan

R/L : 0 = menggeser satu spasi ke kiri

❖ *Fuction Set*

DL : 1 = panjang data LCD pada 8 bit

DL : 0 = panjang data LCD pada 4 bit

Bit upper ditransfer terlebih dahulu kemudian diikuti dengan 4 bit lower.

N : 1/0 = LCD menggunakan 2 atau 1 baris karakter

P : 1/0 = LCD menggunakan 5 x 10 dot matrik

- ❖ *CG RAM address set* : menulis alamat RAM ke karakter
 - ❖ *DD RAM address set* : menulis alamat RAM ke tampilan
 - ❖ *BF/address set* : BF = 1/0, LCD dalam keadaan sibuk atau tidak sibuk.
 - ❖ *Data write to CG RAM or DD RAM* : membaca byte dari alamat terakhir RAM yang dipilih.
-

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

Pendahuluan

Dalam bab ini akan dibahas perancangan dan pembuatan alat. Pembahasan akan dilakukan pada setiap blok rangkaian, cara kerja masing-masing blok rangkaian, perhitungan dan fungsi masing-masing blok rangkaian tersebut. Secara garis besar terdapat dua bagian perangkat yang ada yaitu :

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

Pada perancangan perangkat keras akan meliputi seluruh *peripheral* yang digunakan pada sistem ini. Pada perancangan perangkat lunak akan meliputi diagram alir dan *software* secara umum. Akan tetapi kedua perangkat ini dalam kerjanya akan saling menunjang satu sama lain.

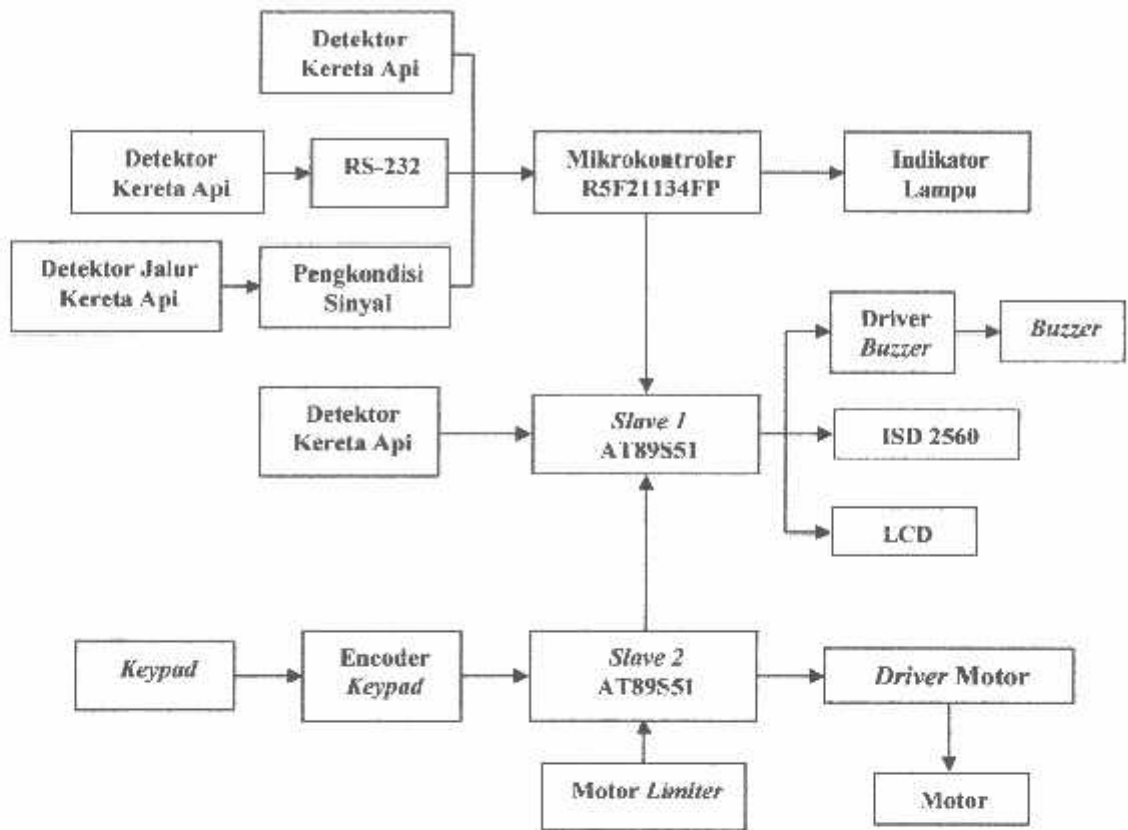
Secara umum sistem kerja dari keseluruhan sistem ini adalah apabila detektor kereta api mendeteksi kedatangan kereta api maka detektor akan mengirim sinyal indikator dan peringatan kepada petugas. Kemudian petugas menentukan jalur mana yang akan digunakan oleh kereta api dari input keypad yang kemudian dieksekusi oleh mikrokontroler *slave 2* AT89S51. Ketika detektor kereta kedua mendeteksi kereta api maka mikrokontroler *slave 1* AT89S51 mengaktifkan informasi suara dan tampilan kedatangan kereta api. Kemudian mikrokontroler renesas akan menunggu hingga detektor kereta tengah mendeteksi kereta api pada jalur stasiun dan mengaktifkan indikator jalur yang sedang digunakan oleh kereta api. Kereta api yang keluar dari stasiun akan terdeteksi oleh

detektor kereta api yang kemudian akan menunggu hingga kereta api keluar dari stasiun kemudian mikrokontroler akan membaca detektor jalur kereta api yang kemudian mengaktifkan indikator mana jalur yang sedang digunakan dan jalur yang kosong.

3.1. Blok Diagram Sistem

Gambar 3.1. merupakan diagram blok sistem yang secara umum terdiri dari masukan-masukan dan keluaran-keluaran yang diproses oleh mikrokontroler. Mikrokontroler dalam diagram blok terdapat 3 bagian yaitu mikrokontroler *Master* Renesas R8C Tiny R5F21134FP, mikrokontroler *Slave 1* AT89S51 dan mikrokontroler *Slave 2* AT89S51.

Mikrokontroler dalam diagram blok terdapat 3 bagian yaitu mikrokontroler *Master* Renesas R8C Tiny R5F21134FP, mikrokontroler *Slave 1* AT89S51 dan mikrokontroler *Slave 2* AT89S51 yang bekerja saling berhubungan. Mikrokontroler *Master* disini bertugas sebagai pengendali utama dari mikrokontroler *Slave*. Berikut penjelasan secara umum masing-masing diagram blok :



Gambar 3.1. Diagram Blok Pemindah Jalur Rel Kereta Api

- **Detektor Kereta Api**

Menggunakan 10 (sepuluh) *limit switch* yang letaknya terpisah sesuai dengan fungsinya. Dua detektor untuk mendeteksi kedatangan kereta api dari kanan dan kiri stasiun. Dua detektor untuk mengirim sinyal informasi bahwa kereta api telah memasuki stasiun. Dua detektor untuk mendeteksi kereta api meninggalkan stasiun dan empat detektor untuk mendeteksi jalur di stasiun yang digunakan kereta api.

- **Detektor Jalur Kereta Api**

Menggunakan 4 (empat) *photodiode* yang digunakan untuk mendeteksi keberadaan kereta api pada jalur rel kereta api di stasiun. Pada saat

detektor kereta api mendeteksi kereta api meninggalkan stasiun maka mikrokontroler akan membaca masukan dari *photodiode* untuk menentukan jalur kereta api yang kosong di stasiun.

- **RS-232**

Sebagai media komunikasi jarak jauh antara detektor kereta api yang digunakan untuk mendeteksi kedatangan kereta api dengan mikrokontroler renesas. Menggunakan standart komunikasi RS-232, jarak maksimal yang disimulasikan ± 15 meter.

- **Pengkondisi Sinyal.**

Untuk mengubah level sinyal analog dari masing-masing detektor menjadi level sinyal digital agar dapat diproses ke mikrokontroler.

- **Mikrokontroler Renesas R8C Tiny R5F21134FP.**

Mikrokontroler ini merupakan mikrokontroler pengendali utama yaitu menerima masukan-masukan dari detektor kereta api dan detektor jalur kereta api, kemudian memprosesnya pada tampilan indikator lampu serta memberi perintah kepada mikrokontroler slave AT89S51 untuk mengaktifkan LCD, *buzzer* serta ISD.

- **LCD M1632.**

Menampilkan informasi kedatangan kereta api melalui media tulisan.

- **Information Storage Device (ISD) 2560.**

Sebagai memori suara terprogram yang dapat menyimpan suara dalam bentuk data-data digital dan dapat dipanggil dan mengubahnya kembali menjadi data suara audio. *Peripheral* ISD 2560 ini mempunyai waktu

bicara sepanjang 60 detik. Fungsi dari bagian ini adalah menginformasikan kedatangan kereta api melalui media suara.

- **Buzzer**

Sebagai indikator peringatan kepada petugas bahwa terdapat kereta api yang akan memasuki stasiun.

- **Keypad**

Sebagai input untuk memilih jalur rel kereta api yang akan digunakan.

- **Encoder Keypad**

Encoder keypad melakukan scan kolom dan scan baris pada *keypad*. Bila terdapat tombol yang tertekan maka *encoder* akan mendeteksi tombol yang ditekan tersebut kemudian mengirimkan sinyal terdapat data baru pada output *encoder*. Data yang dikirim oleh *encoder keypad* tidak sama antara penekanan tombol satu dengan yang lain.

- **Motor Limiter**

Menggunakan *limit switch*, sebagai pembatas pergerakan motor dc.

- **Mikrokontroler AT89S51**

Sebagai pengendali utama pada penggerak pemindah jalur rel kereta api. Input dari *keypad* diolah oleh mikrokontroler yang kemudian menggerakkan motor untuk memindah jalur rel.

- **Driver Motor**

Merupakan *driver relay* yang menerima data-data digital dari mikrokontroler untuk mengendalikan arah putaran dari motor dc yang digunakan sebagai pemindah jalur rel kereta api.

- **Motor**

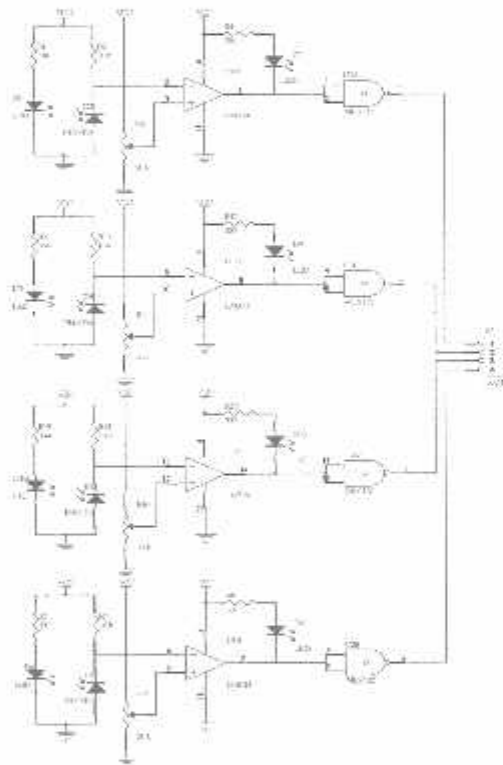
Untuk menggerakkan perpindahan jalur rel kereta api.

3.2. Perancangan Perangkat Keras (*Hardware*)

Perancangan perangkat keras terdiri dari beberapa bagian secara umum dapat dibagi menjadi 3 bagian utama yaitu :

- ❖ Bagian masukan yang terdiri dari detektor-detektor.
- ❖ Bagian pemroses data yaitu mikrokontroler.
- ❖ Bagian keluaran yang terdiri dari *driver-driver*, buzzer, indikator LED, motor dc, LCD dan ISD.

3.2.1. Detektor Jalur Kereta Api



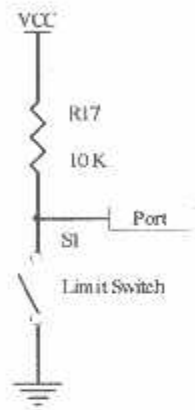
Gambar 3.2. Rangkaian Detektor Jalur Kereta Api

Dalam perancangan detektor ini menggunakan pasangan *infra red* sebagai pemancarnya dan *photodiode* sebagai penerimanya. *Photodiode* ini digunakan sebagai pendeteksi keberadaan kereta api pada jalur rel kereta api di stasiun. Pada gambar 3.2. adalah rangkaian *infra red* dan *photodiode* beserta pengkondisi sinyalnya.

Komponen utama dari rangkaian ini adalah *infra red* dan *photodiode*. *Infra red* bertugas memancarkan cahaya kemudian ditangkap oleh *photodiode*. Sifat dari *photodiode* bilamana terkena cahaya dari *infra red* maka hambatan pada dirinya menjadi kecil begitu juga sebaliknya. Hal ini dimanfaatkan untuk mengolah tegangan menjadi aktif *high* atau aktif *low* dengan membangun rangkaian pembagi tegangan. Variabel resistor pada rangkaian pembagi tegangan berfungsi untuk memberikan tegangan referensi pada op-amp. Perubahan tegangan ini kemudian dibandingkan dengan tegangan referensi pada op-amp. Bila perubahan tegangan dari pembagi tegangan sama dengan tegangan referensi maka pada keluaran op-amp akan memberikan level tegangan yang tinggi, begitu juga sebaliknya. Level tegangan dari keluaran op-amp ini masih berupa sinyal analog, untuk menjadikan sinyal digital agar mikrokontroler dapat menerima sinyal ini maka diperlukan pengkondisi sinyal yang dibangun dengan gerbang *schmitt trigger* 74HC132. Prinsipnya adalah level tegangan analog diregulasi menjadi sinyal digital. Detektor jalur kereta api ini dikoneksikan ke *port* 1.4 s.d. *port* 1.7 pada mikrokontroler renesas sebagai masukan data.

3.2.2. Detektor Kereta Api dan Motor *Limit*

Limit switch digunakan untuk mendeteksi kereta api yang melewati jalur tertentu serta digunakan untuk membatasi pergerakan dari motor dc. Menggunakan *limit switch* biasa diaktifkan secara aktif *low* dengan memberikan resistor *pull-up*.



Gambar 3.3. Rangkaian *Limit Switch*

Dalam gambar 3.3, rangkaian *limit switch* dipasang tahanan *pull up* yang dihubungkan ke masukan *port* pada mikrokontroler untuk menjamin agar masukan *port* berada dalam logika tinggi ketika tombol terbuka. Gambar 3.3, memperlihatkan persamaan tahanan *pull up*. Dengan melihat lembar data pada IC mikrokontroler Renesas R8C Tiny R5F21134FP, maka tahanan *pull up* minimum dapat dihitung :

$$R_{\text{pull up}} = \frac{V_{\text{CC}} - V_{\text{OL}}}{I_{\text{OL}}}$$

$$R_{\text{pull up}} = \frac{5 - 0,45}{1,6 \cdot 10^{-3}} = 2843,75 \Omega$$

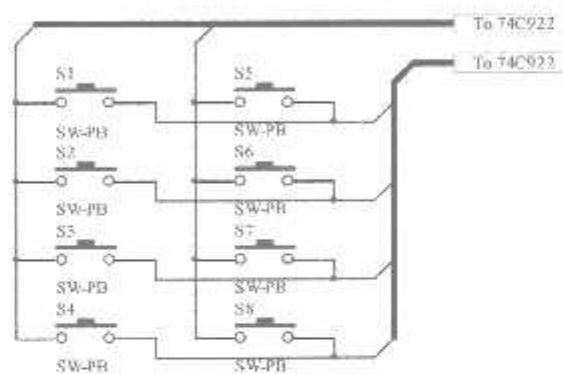
Dalam perancangan $R_{\text{pull up}} = 10 \text{ k}\Omega$ sehingga arus yang mengalir ketika logika rendah dapat diketahui, yaitu :

$$I_{\text{OL}} = \frac{5 - 0,45}{10000} = 0,455 \text{ mA}$$

Jadi dengan nilai $R_{pull\ up} = 10\text{ k}\Omega$, maka arus yang terserap ketika keluaran rendah tidak lebih besar dari batas arus I_{OL} -nya (umumnya sebesar 1,6 mA).

3.2.3. Keypad

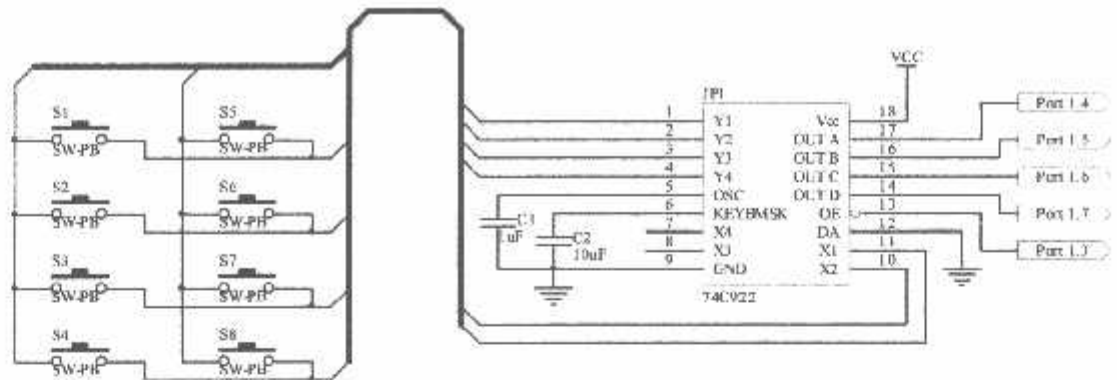
Rangkaian *keypad* dalam perencanaan ini menggunakan *keypad matrik* push button 2×4 . Teknik pembacaan dari *keypad* ini, yaitu model *scanning* pada jalur baris dan jalur kolom. Bila baris dan kolom disilangkan maka akan terbentuk titik-titik potong antara keduanya seperti pada gambar 3.4. di bawah ini:



Gambar 3.4 Rangkaian Keypad 2×4

3.2.4. Rangkaian Encoder Keypad

Encoder keypad melakukan scan kolom dan scan baris pada *keypad*. Bila terdapat tombol yang tertekan maka *encoder* akan mendeteksi tombol yang ditekan tersebut kemudian mengirimkan sinyal terdapat data baru pada output *encoder*. Data yang dikirim oleh *encoder keypad* tidak sama antara penekanan tombol satu dengan yang lain.



Gambar 3.5 Rangkaian *Encoder Keypad*

Pada perancangan keypad matrik 2×4, terdapat 4 baris *keypad* dan 2 kolom *keypad*. Baris pada *keypad* dihubungkan dengan pin Y₁, Y₂, Y₃, Y₄ dari *encoder keypad*. Kolom pada *keypad* dihubungkan dengan X₁, X₂ dari *encoder keypad*. Pin OE dari *encoder keypad* terhubung dengan port 1.3 mikrokontroler *slave 2* AT89S51. Pin OE akan memberikan logika *high* sesaat ketika terdapat data baru atau ketika tombol ditekan. Data output *encoder* terhubung dengan *port 1.4* s.d *1.7* mikrokontroler *slave 2* AT89S51.

3.2.5. Rangkaian Indikator

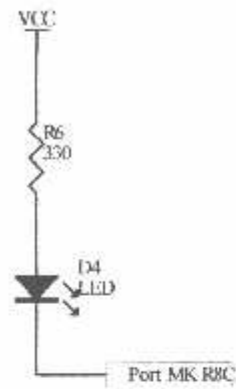
3.2.5.1. Rangkaian Indikator Jalur Rel

Dalam rangkaian indikator menggunakan LED yang diseri dengan sebuah resistor yang berfungsi untuk membatasi arus yang melalui LED sebesar 10 mA, sehingga R_S dapat diketahui :

$$R_S = \frac{V_{CC} - V_{LED} - V_{OL}}{I_{LED}} = \frac{(5 - 1,2 - 0,45) V}{10 mA} = 335 \Omega$$

Dalam perencanaan R_S yang digunakan sebesar 330Ω sehingga dapat dihitung kembali :

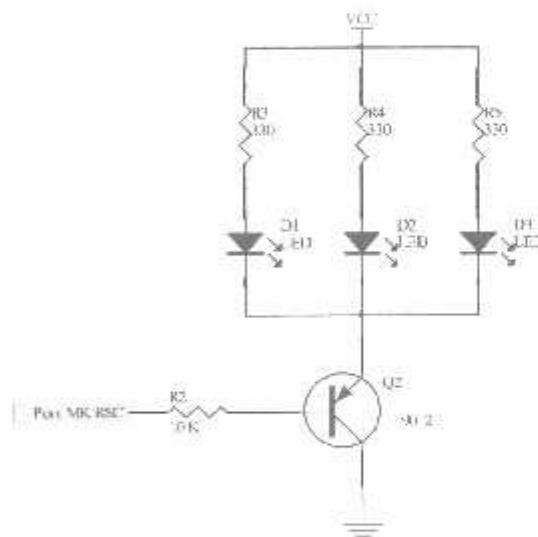
$$I_{LED} = \frac{(5 - 1,2 - 0,45) V}{330 \Omega} = 10,15 mA$$



Gambar 3.6. Rangkaian Indikator Jalur Rel

Gambar 3.6. memperlihatkan rangkaian indikator yang dihubungkan dengan mikrokontroler.

3.2.5.2. Rangkaian Indikator Posisi Wessel



Gambar 3.7. Rangkaian Indikator Posisi Wessel

Untuk mencari harga dari resistansi basis transistor 9012, sehingga akan bekerja sebagai transistor saklar, maka dengan data-data pengukuran H_{fe} Transistor didapat :

$$H_{fe} \text{ Transistor 9012} = \beta_{dc} = 170$$

$$R_c = R_{Led} = R_{paralel} = 110 \Omega$$

$$I_c = \frac{V_{cc}}{R_c}$$

$$I_c = \frac{5}{110}$$

$$= 45.45 \text{ mA}$$

$$I_b = \frac{I_c}{H_{fe}}$$

$$= \frac{0.04545}{170}$$

$$= 0.267 \text{ mA}$$

$$R_b = \frac{V_{in} - V_{bc} - V_{led} - V_{ol}}{I_b}$$

$$= \frac{5 - 0.7 - 1.2 - 0.45}{0.267 \times 10^{-3}}$$

$$= 9.92 \text{ K}\Omega$$

Karena dipasaran tidak ada resistor dengan harga 9.92 K Ω maka dipakai resistor 10 K Ω .

3.2.6. Rangkaian *Driver Buzzer*

Buzzer akan berbunyi jika sensor mendeteksi adanya kereta api yang akan memasuki satasiun. Jika *buzzer* berbunyi maka merupakan peringatan bagi

1. Overview

This MCU is built using the high-performance silicon gate CMOS process using a R8C/Tiny Series CPU core and is packaged in a 32-pin plastic molded LQFP. This MCU operates using sophisticated instructions featuring a high level of instruction efficiency. With 1M bytes of address space, it is capable of executing instructions at high speed.

The data flash ROM (2 KB X 2 blocks) is embedded.

1.1 Applications

Electric household appliance, office equipment, housing equipment (sensor, security), general industrial equipment, audio, etc.

1.2 Performance Outline

Table 1.1. lists the performance outline of this MCU.

Table 1.1 Performance outline

Item	Performance	
CPU	Number of basic instructions	89 instructions
	Shortest instruction execution time	50 ns ($f(XIN) = 20$ MHz, $V_{CC} = 3.0$ to 5.5 V) 100 ns ($f(XIN) = 10$ MHz, $V_{CC} = 2.7$ to 5.5 V)
	Operating mode	Single-chip
	Address space	1M bytes
	Memory capacity	See Table 1.2.
Peripheral function	Interrupt	Internal: 11 factors, External: 5 factors, Software: 4 factors, Priority level: 7 levels
	Watchdog timer	15 bits x 1 (with prescaler) Reset start function selectable
	Timer	Timer X: 8 bits x 1 channel, Timer Y: 8 bits x 1 channel, Timer Z: 8 bits x 1 channel (Each timer equipped with 8-bit prescaler) Timer C: 16 bits x 1 channel Circuits of input capture and output compare.
	Serial interface	•1 channel Clock synchronous, UART •1 channel UART
	A/D converter	10-bit A/D converter: 1 circuit, 12 channels
	Clock generation circuit	2 circuits •Main clock generation circuit (Equipped with a built-in feedback resistor) •On-chip oscillator (high-speed, low-speed) On high-speed on-chip oscillator the frequency adjustment function is usable.
	Oscillation stop detection function	Stop detection of main clock oscillation
	Voltage detection circuit	Included
	Power on reset circuit	Included
	Port	Input/Output: 22 (including LED drive port), Input: 2 (LED drive I/O port: 8)
Electrical characteristics	Power supply voltage	$V_{CC} = 3.0$ to 5.5V ($f(XIN) = 20$ MHz) $V_{CC} = 2.7$ to 5.5V ($f(XIN) = 10$ MHz)
	Power consumption	Typ.9 mA ($V_{CC} = 5.0$ V, ($f(XIN) = 20$ MHz, High-speed mode) Typ.5 mA ($V_{CC} = 3.0$ V, ($f(XIN) = 10$ MHz, High-speed mode) Typ.35 μ A ($V_{CC} = 3.0$ V, Wait mode, Peripheral clock stops) Typ.0.7 μ A ($V_{CC} = 3.0$ V, Stop mode)
Flash memory	Program/erase voltage	$V_{CC} = 2.7$ to 5.5 V
	Number of program/erase	10,000 times (Data area) 1,000 times (Program area)
Operating ambient temperature	-20 to 85°C -40 to 85°C (D-version)	
Package	32-pin plastic mold LQFP	

1.3 Block Diagram

Figure 1.1 shows this MCU block diagram.

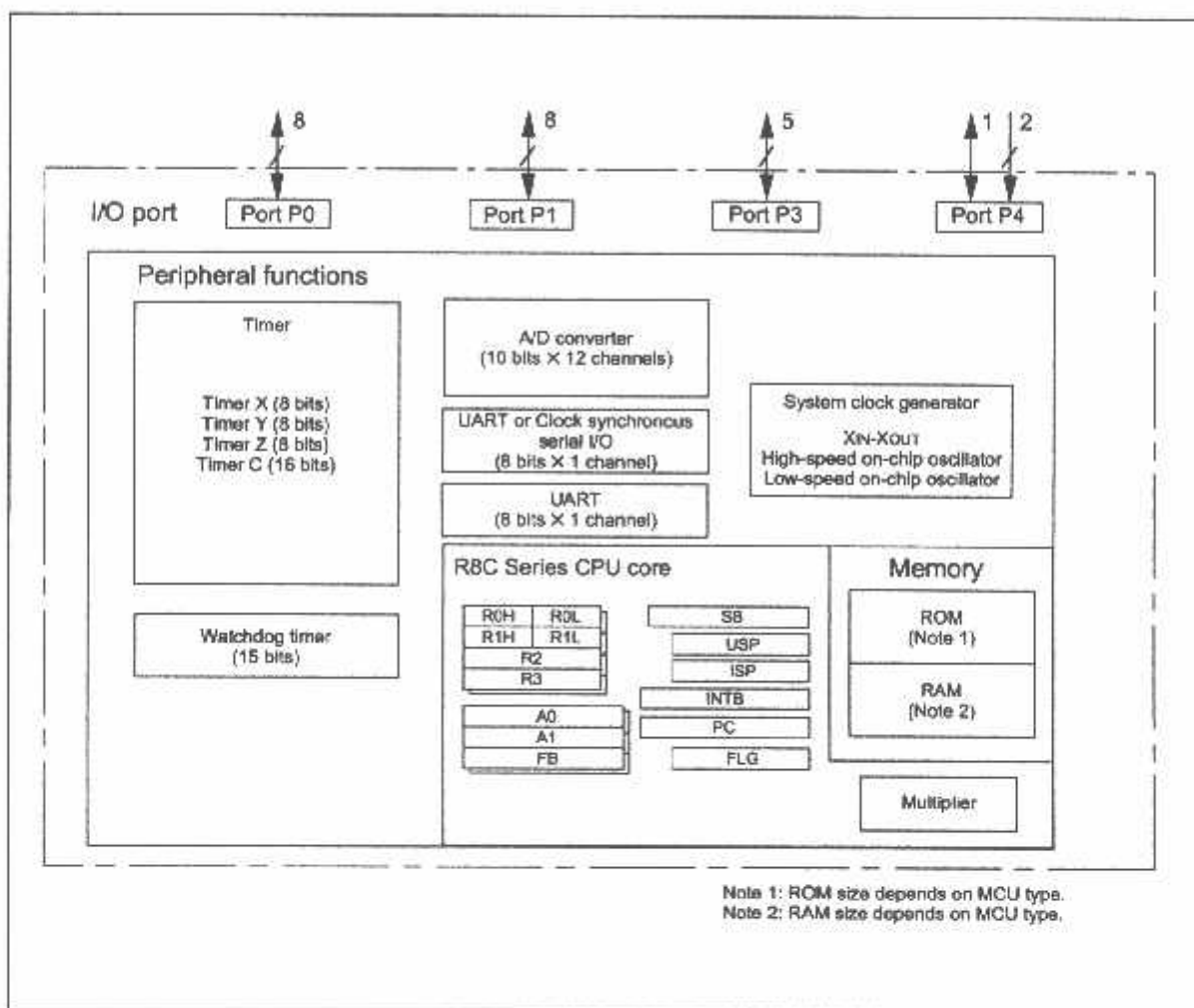


Figure 1.1 Block Diagram

1.4 Product Information

Table 1.2 lists the products.

Table 1.2 Product List

As of April 2005

Type No.	ROM capacity		RAM capacity	Package type	Remarks
	Program area	Data area			
R5F21-32FP	8K bytes	2K bytes x 2	512 bytes	PLQP0032GB-A	Flash memory version
R5F21-33FP	12K bytes	2K bytes x 2	768 bytes	PLQP0032GB-A	
R5F21-34FP	16K bytes	2K bytes x 2	1K bytes	PLQP0032GB-A	
R5F21132DFP	8K bytes	2K bytes x 2	512 bytes	PLQP0032GB-A	D version
R5F21133DFP	12K bytes	2K bytes x 2	768 bytes	PLQP0032GB-A	
R5F21134DFP	16K bytes	2K bytes x 2	1K bytes	PLQP0032GB-A	

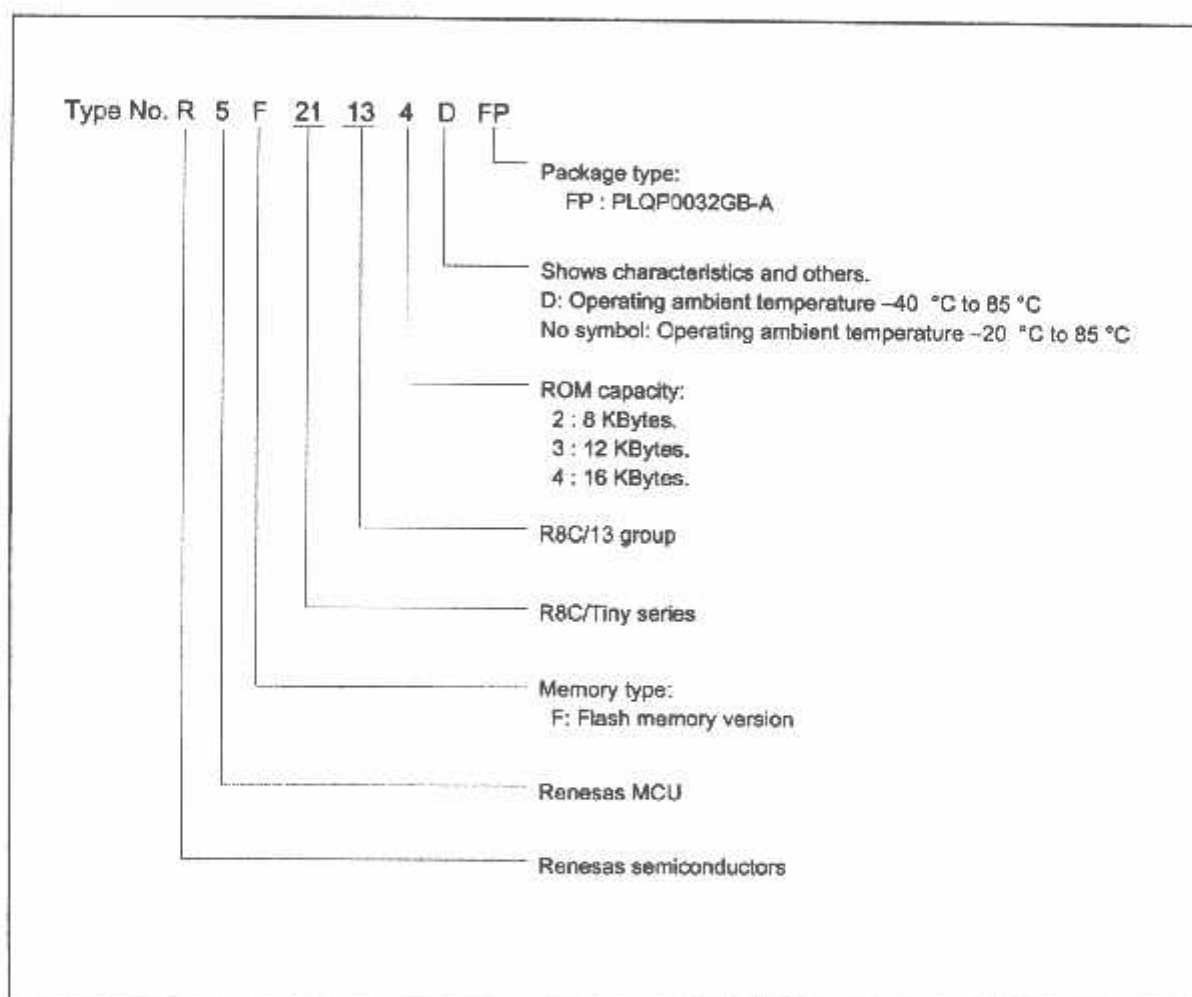
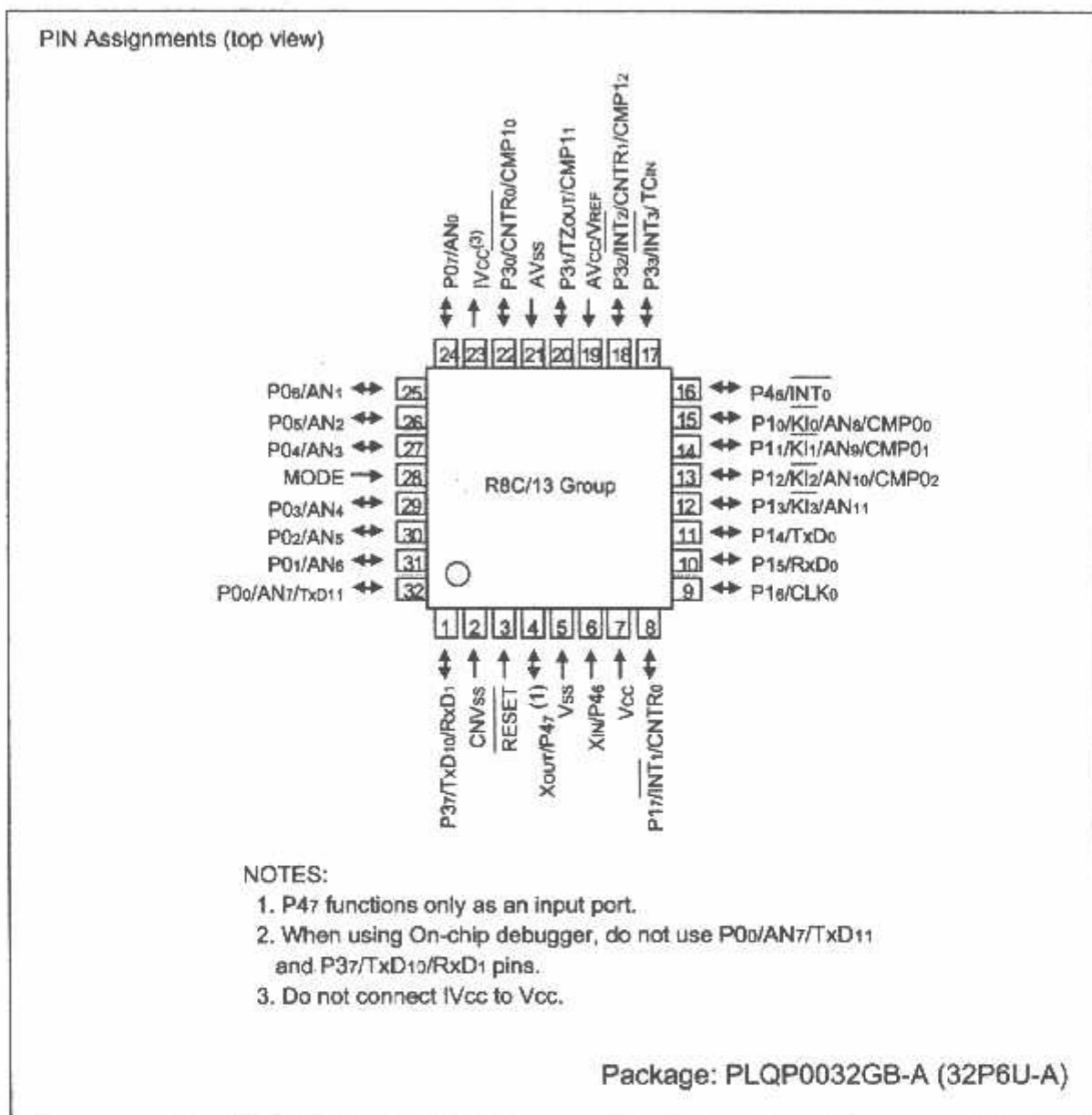


Figure 1.2 Type No., Memory Size, and Package

1.5 Pin Assignments

Figure 1.3 shows the pin configuration (top view).



NOTES:

1. P47 functions only as an input port.
2. When using On-chip debugger, do not use P00/AN7/TxD11 and P37/TxD10/RxD1 pins.
3. Do not connect IVcc to Vcc.

Figure 1.3 Pin Assignments (Top View)

1.6 Pin Description

Table 1.3 shows the pin description

Table 1.3 Pin description

Signal name	Pin name	I/O type	Function
Power supply input	Vcc, Vss	I	Apply 2.7 V to 5.5 V to the Vcc pin. Apply 0 V to the Vss pin.
IVcc	IVcc	O	This pin is to stabilize internal power supply Connect this pin to Vss via a capacitor (0.1 μ F) Do not connect to Vcc
Analog power supply input	AVcc, AVss	I	These are power supply input pins for A/D converter. Connect the AVcc pin to Vcc. Connect the AVss pin to Vss. Connect a capacitor between pins AVcc and AVss.
Reset input	RESET	I	"L" on this input resets the MCU.
CNVss	CNVss	I	Connect this pin to Vss via a resistor ⁽¹⁾
MODE	MODE	I	Connect this pin to Vcc via a resistor
Main clock input	XIN	I	These pins are provided for the main clock generating circuit I/O. Connect a ceramic resonator or a crystal oscillator between the XIN and XOUT pins. To use an externally derived clock, input it to the XIN pin and leave the XOUT pin open.
Main clock output	XOUT	O	
INT interrupt input	INT0 to INT3	I	These are INT interrupt input pins.
Key input interrupt input	KI0 to KI3	I	These are key input interrupt pins.
Timer X	CNTR0	I/O	This is the timer X I/O pin.
	CNTR0	O	This is the timer X output pin.
Timer Y	CNTR1	I/O	This is the timer Y I/O pin.
Timer Z	TZOUT	O	This is the timer Z output pin.
Timer C	TCIN	I	This is the timer C input pin.
	CMP00 to CMP03, CMP10 to CMP13	O	These are the timer C output pins.
Serial interface	CLK0	I/O	This is a transfer clock I/O pin.
	RxD0, RxD1	I	These are serial data input pins.
	TxD0, TxD10, TxD11	O	These are serial data output pins.
Reference voltage input	VREF	I	This is a reference voltage input pin for A/D converter. Connect the VREF pin to Vcc.
A/D converter	AN0 to AN11	I	These are analog input pins for A/D converter.
I/O port	P00 to P07, P10 to P17, P30 to P33, P37, P45	I/O	These are 8-bit CMOS I/O ports. Each port has an I/O select direction register, allowing each pin in that port to be directed for input or output individually. Any port set to input can select whether to use a pull-up resistor or not by program. P10 to P17 also function as LED drive ports.
Input port	P46, P47	I	These are input only pins.

2. Central Processing Unit (CPU)

Figure 2.1 shows the CPU registers. The CPU has 13 registers. Of these, R0, R1, R2, R3, A0, A1 and FB comprise a register bank. There are two register banks.

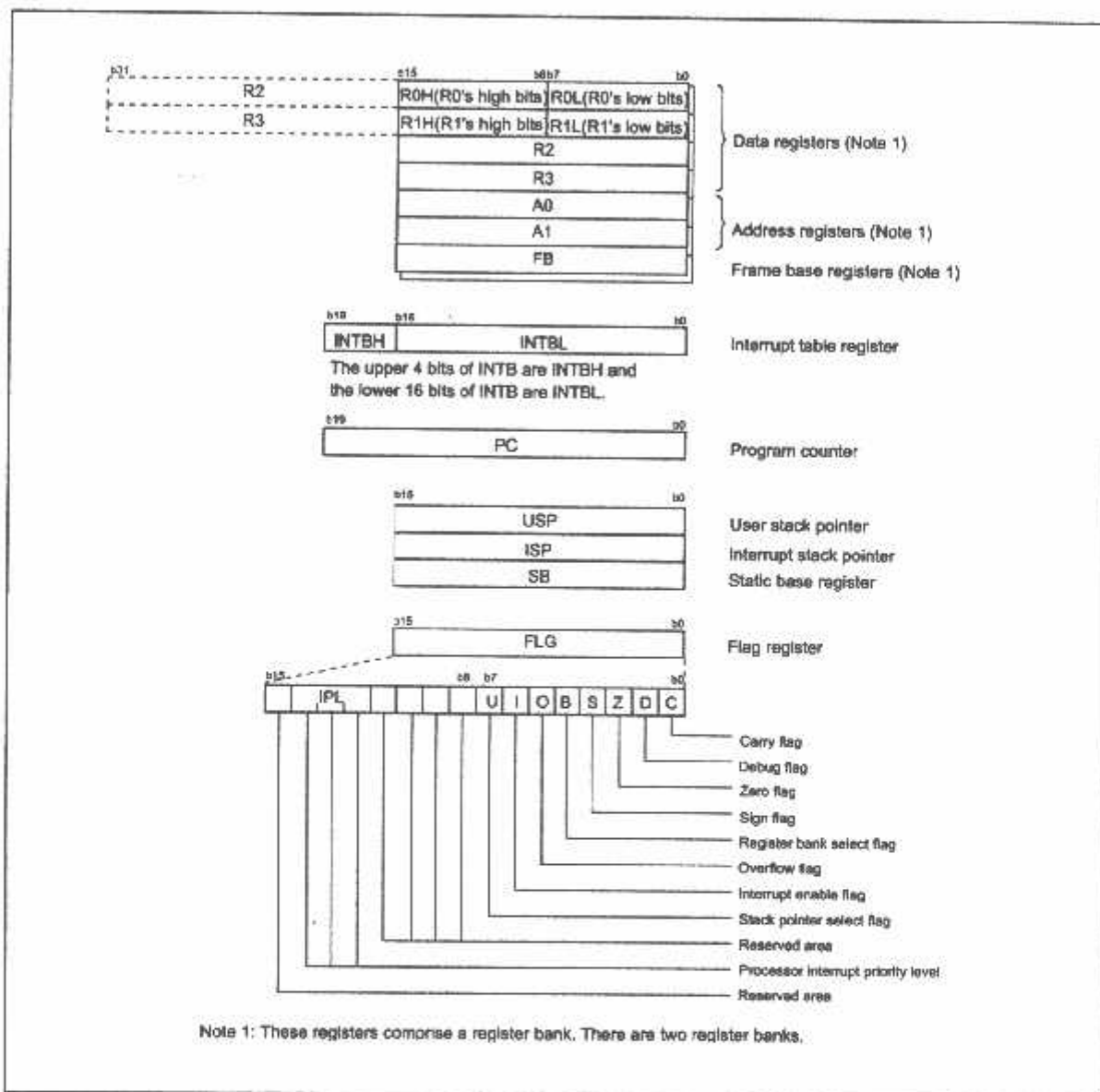


Figure 2.1 Central Processing Unit Register

2.1 Data Registers (R0, R1, R2 and R3)

The R0 register consists of 16 bits, and is used mainly for transfers and arithmetic/logic operations. R1 to R3 are the same as R0.

The R0 register can be separated between high (R0H) and low (R0L) for use as two 8-bit data registers. R1H and R1L are the same as R0H and R0L. Conversely, R2 and R0 can be combined for use as a 32-bit data register (R2R0). R3R1 is the same as R2R0.

2.2 Address Registers (A0 and A1)

The register A0 consists of 16 bits, and is used for address register indirect addressing and address register relative addressing. They also are used for transfers and logic/logic operations. A1 is the same as A0.

In some instructions, registers A1 and A0 can be combined for use as a 32-bit address register (A1A0).

2.3 Frame Base Register (FB)

FB is configured with 16 bits, and is used for FB relative addressing.

2.4 Interrupt Table Register (INTB)

INTB is configured with 20 bits, indicating the start address of an interrupt vector table.

2.5 Program Counter (PC)

PC is configured with 20 bits, indicating the address of an instruction to be executed.

2.6 User Stack Pointer (USP) and Interrupt Stack Pointer (ISP)

Stack pointer (SP) comes in two types: USP and ISP, each configured with 16 bits.

Your desired type of stack pointer (USP or ISP) can be selected by the U flag of FLG.

2.7 Static Base Register (SB)

SB is configured with 16 bits, and is used for SB relative addressing.

2.8 Flag Register (FLG)

FLG consists of 11 bits, indicating the CPU status.

2.8.1 Carry Flag (C Flag)

This flag retains a carry, borrow, or shift-out bit that has occurred in the arithmetic/logic unit.

2.8.2 Debug Flag (D Flag)

The D flag is used exclusively for debugging purpose. During normal use, it must be set to "0".

2.8.3 Zero Flag (Z Flag)

This flag is set to "1" when an arithmetic operation resulted in 0; otherwise, it is "0".

2.8.4 Sign Flag (S Flag)

This flag is set to "1" when an arithmetic operation resulted in a negative value; otherwise, it is "0".

2.8.5 Register Bank Select Flag (B Flag)

Register bank 0 is selected when this flag is "0"; register bank 1 is selected when this flag is "1".

2.8.6 Overflow Flag (O Flag)

This flag is set to "1" when the operation resulted in an overflow; otherwise, it is "0".

2.8.7 Interrupt Enable Flag (I Flag)

This flag enables a maskable interrupt.

Maskable interrupts are disabled when the I flag is "0", and are enabled when the I flag is "1". The I flag is cleared to "0" when the interrupt request is accepted.

2.8.8 Stack Pointer Select Flag (U Flag)

ISP is selected when the U flag is "0"; USP is selected when the U flag is "1".

The U flag is cleared to "0" when a hardware interrupt request is accepted or an INT instruction for software interrupt Nos. 0 to 31 is executed.

2.8.9 Processor Interrupt Priority Level (IPL)

IPL is configured with three bits, for specification of up to eight processor interrupt priority levels from level 0 to level 7.

If a requested interrupt has priority greater than IPL, the interrupt is enabled.

2.8.10 Reserved Area

When write to this bit, write "0". When read, its content is indeterminate.

3. Memory

Figure 3.1 is a memory map of this MCU. The address space extends the 1M bytes from address 00000_{16} to $FFFFFF_{16}$.

The internal ROM (program area) is allocated in a lower address direction beginning with address $0FFFF_{16}$. For example, a 16-Kbyte internal ROM is allocated to the addresses from $0C000_{16}$ to $0FFFF_{16}$.

The fixed interrupt vector table is allocated to the addresses from $0FFDC_{16}$ to $0FFFF_{16}$. Therefore, store the start address of each interrupt routine here.

The internal ROM (data area) is allocated to the addresses from 02000_{16} to $02FFF_{16}$.

The internal RAM is allocated in an upper address direction beginning with address 00400_{16} . For example, a 1-Kbyte internal RAM is allocated to the addresses from 00400_{16} to $007FF_{16}$. In addition to storing data, the internal RAM also stores the stack used when calling subroutines and when interrupts are generated.

Special function registers (SFR) are allocated to the addresses from 00000_{16} to $002FF_{16}$. Peripheral function control registers are located here. Of the SFR, any space which has no functions allocated is reserved for future use and cannot be used by users.

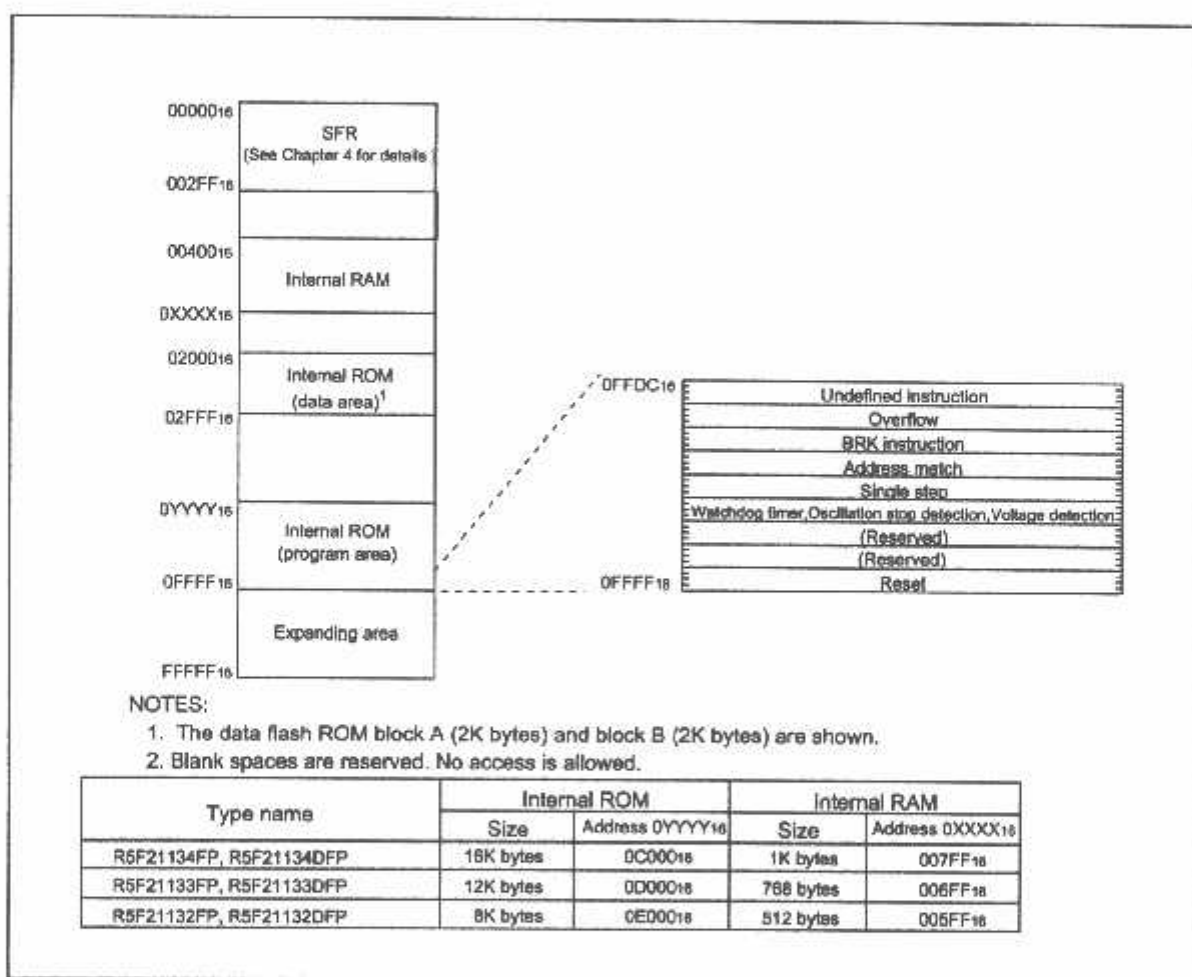


Figure 3.1 Memory Map

4. Special Function Register (SFR)

SFR(Special Function Register) is the control register of peripheral functions. Tables 4.1 to 4.4 list the SFR information

Table 4.1 SFR Information(1)(1)

Address	Register	Symbol	After reset
0000 ₁₆			
0001 ₁₆			
0002 ₁₆			
0003 ₁₆			
0004 ₁₆	Processor mode register 0 ¹	PM0	00 ₁₆
0005 ₁₆	Processor mode register 1	PM1	00 ₁₆
0006 ₁₆	System clock control register 0	CM0	01101000 ₂
0007 ₁₆	System clock control register 1	CM1	00100000 ₂
0008 ₁₆	High-speed on-chip oscillator control register 0	HR0	00 ₁₆
0009 ₁₆	Address match interrupt enable register	AIER	XXXXXX00 ₂
000A ₁₆	Protect register	PRCR	00XX000 ₂
000B ₁₆	High-speed on-chip oscillator control register 1	HRT	40 ₁₆
000C ₁₆	Oscillation stop detection register	OSD	00000100 ₂
000D ₁₆	Watchdog timer reset register	WDTR	XX ₁₆
000E ₁₆	Watchdog timer start register	WDT8	XX ₁₆
000F ₁₆	Watchdog timer control register	WDC	00011111 ₂
0010 ₁₆	Address match interrupt register 0	RMAD0	00 ₁₆
0011 ₁₆			00 ₁₆
0012 ₁₆			X0 ₁₆
0013 ₁₆			
0014 ₁₆	Address match interrupt register 1	RMAD1	00 ₁₆
0015 ₁₆			00 ₁₆
0018 ₁₆			X0 ₁₆
0017 ₁₆			
0018 ₁₆			
0019 ₁₆	Voltage detection register 1 ²	VCR1	00001000 ₂
001A ₁₆	Voltage detection register 2 ²	VCR2	00 ₁₆ ³ 10000000 ₂ ⁴
001B ₁₆			
001C ₁₆			
001D ₁₆			
001E ₁₆	INT0 input filter select register	INT0F	XXXXX000 ₂
001F ₁₆	Voltage detection interrupt register ²	D4INT	00 ₁₆ ³ 01000001 ₂ ⁴
0020 ₁₆			
0021 ₁₆			
0022 ₁₆			
0023 ₁₆			
0024 ₁₆			
0025 ₁₆			
0026 ₁₆			
0027 ₁₆			
0028 ₁₆			
0029 ₁₆			
002A ₁₆			
002B ₁₆			
002C ₁₆			
002D ₁₆			
002E ₁₆			
002F ₁₆			
0030 ₁₆			
0031 ₁₆			
0032 ₁₆			
0033 ₁₆			
0034 ₁₆			
0035 ₁₆			
0036 ₁₆			
0037 ₁₆			
0038 ₁₆			
0039 ₁₆			
003A ₁₆			
003B ₁₆			
003C ₁₆			
003D ₁₆			
003E ₁₆			
003F ₁₆			

X : Undefined

NOTES:

- Blank columns are all reserved space. No access is allowed.
- Software reset or the watchdog timer reset does not affect this register.
- Owing to Reset input.
- In the case of RESET pin = H retaining.

Table 4.2 SFR Information(2)⁽¹⁾

Address	Register	Symbol	After reset
0040 ₁₆			
0041 ₁₆			
0042 ₁₆			
0043 ₁₆			
0044 ₁₆			
0045 ₁₆			
0046 ₁₆			
0047 ₁₆			
0048 ₁₆			
0049 ₁₆			
004A ₁₆			
004B ₁₆			
004C ₁₆			
004D ₁₆	Key input interrupt control register	KUPIC	XXXXX0002
004E ₁₆	AD conversion interrupt control register	ADIC	XXXXX0002
004F ₁₆			
0050 ₁₆	Compare 1 interrupt control register	CMP1IC	XXXXX0002
0051 ₁₆	UART0 transmit interrupt control register	S0TIC	XXXXX0002
0052 ₁₆	UART0 receive interrupt control register	S0RIC	XXXXX0002
0053 ₁₆	UART1 transmit interrupt control register	S1TIC	XXXXX0002
0054 ₁₆	UART1 receive interrupt control register	S1RIC	XXXXX0002
0055 ₁₆	INT2 interrupt control register	INT2IC	XXXXX0002
0056 ₁₆	Timer X interrupt control register	TXIC	XXXXX0002
0057 ₁₆	Timer Y interrupt control register	TYIC	XXXXX0002
0058 ₁₆	Timer Z interrupt control register	TZIC	XXXXX0002
0059 ₁₆	INT1 interrupt control register	INT1IC	XXXXX0002
005A ₁₆	INT3 interrupt control register	INT3IC	XXXXX0002
005B ₁₆	Timer C interrupt control register	TCIC	XXXXX0002
005C ₁₆	Compare 0 interrupt control register	CMP0IC	XXXXX0002
005D ₁₆	INT0 interrupt control register	INT0IC	XXXXX0002
005E ₁₆			
005F ₁₆			
0060 ₁₆			
0061 ₁₆			
0062 ₁₆			
0063 ₁₆			
0064 ₁₆			
0065 ₁₆			
0066 ₁₆			
0067 ₁₆			
0068 ₁₆			
0069 ₁₆			
006A ₁₆			
006B ₁₆			
006C ₁₆			
006D ₁₆			
006E ₁₆			
006F ₁₆			
0070 ₁₆			
0071 ₁₆			
0072 ₁₆			
0073 ₁₆			
0074 ₁₆			
0075 ₁₆			
0076 ₁₆			
0077 ₁₆			
0078 ₁₆			
0079 ₁₆			
007A ₁₆			
007B ₁₆			
007C ₁₆			
007D ₁₆			
007E ₁₆			
007F ₁₆			

X : Undefined

NOTES:

1. Blank columns are all reserved space. No access is allowed.

Table 4.3 SFR Information(3)⁽¹⁾

Address	Register	Symbol	After reset
00E0 ₁₆	Timer Y, Z mode register	TYZMR	00 ₁₆
00E1 ₁₆	Prescaler Y	PREY	FF ₁₆
00E2 ₁₆	Timer Y secondary	TYSC	FF ₁₆
00E3 ₁₆	Timer Y primary	TYPR	FF ₁₆
00E4 ₁₆	Timer Y, Z waveform output control register	PUM	00 ₁₆
00E5 ₁₆	Prescaler Z	PREZ	FF ₁₆
00E6 ₁₆	Timer Z secondary	TZSC	FF ₁₆
00E7 ₁₆	Timer Z primary	TZPR	FF ₁₆
00E8 ₁₆			
00E9 ₁₆			
00EA ₁₆	Timer Y, Z output control register	TYZOC	00 ₁₆
00EB ₁₆	Timer X mode register	TXMR	00 ₁₆
00EC ₁₆	Prescaler X	PREX	FF ₁₆
00ED ₁₆	Timer X register	TX	FF ₁₆
00EE ₁₆	Count source set register	TCSS	00 ₁₆
00EF ₁₆			
00F0 ₁₆	Timer C register	TC	00 ₁₆
00F1 ₁₆			00 ₁₆
00F2 ₁₆			
00F3 ₁₆			
00F4 ₁₆			
00F5 ₁₆			
00F6 ₁₆	External input enable register	INTEN	00 ₁₆
00F7 ₁₆			
00F8 ₁₆	Key input enable register	KIEN	00 ₁₆
00F9 ₁₆			
00FA ₁₆	Timer C control register 0	TCC0	00 ₁₆
00FB ₁₆	Timer C control register 1	TCC1	00 ₁₆
00FC ₁₆	Capture, compare 0 register	TM0	00 ₁₆
00FD ₁₆			00 ₁₆ ²
00FE ₁₆	Compare 1 register	TM1	FF ₁₆
00FF ₁₆			FF ₁₆
00A0 ₁₆	UART0 transmit/receive mode register	U0MR	00 ₁₆
00A1 ₁₆	UART0 bit rate register	U0BRG	XX ₁₆
00A2 ₁₆	UART0 transmit buffer register	U0TB	XX ₁₆
00A3 ₁₆			XX ₁₆
00A4 ₁₆	UART0 transmit/receive control register 0	U0C0	000010002
00A5 ₁₆	UART0 transmit/receive control register 1	U0C1	000000102
00A6 ₁₆	UART0 receive buffer register	U0RB	XX ₁₆
00A7 ₁₆			XX ₁₆
00A8 ₁₆	UART1 transmit/receive mode register	U1MR	00 ₁₆
00A9 ₁₆	UART1 bit rate register	U1BRG	XX ₁₆
00AA ₁₆	UART1 transmit buffer register	U1TB	XX ₁₆
00AB ₁₆			XX ₁₆
00AC ₁₆	UART1 transmit/receive control register 0	U1C0	000010002
00AD ₁₆	UART1 transmit/receive control register 1	U1C1	000000102
00AE ₁₆	UART1 receive buffer register	U1RB	XX ₁₆
00AF ₁₆			XX ₁₆
00B0 ₁₆	UART transmit/receive control register 2	UCON	00 ₁₆
00B1 ₁₆			
00B2 ₁₆			
00B3 ₁₆			
00B4 ₁₆			
00B5 ₁₆			
00B6 ₁₆			
00B7 ₁₆			
00B8 ₁₆			
00B9 ₁₆			
00BA ₁₆			
00BB ₁₆			
00BC ₁₆			
00BD ₁₆			
00BE ₁₆			
00BF ₁₆			

X: Undefined

NOTES:

- Blank columns are all reserved space. No access is allowed.
- When the output compare mode is selected (the TCC13 bit in the TCC1 register = 1), the value is set to FFFF₁₆.

Table 4.4 SFR information(4)(1)

Address	Register	Symbol	After reset
00C0 ₁₆	AD register	AD	XX ₁₆
00C1 ₁₆			XX ₁₆
00C2 ₁₆			
00C3 ₁₆			
00C4 ₁₆			
00C5 ₁₆			
00C6 ₁₆			
00C7 ₁₆			
00C8 ₁₆			
00C9 ₁₆			
00CA ₁₆			
00CB ₁₆			
00CC ₁₆			
00CD ₁₆			
00CE ₁₆			
00CF ₁₆			
00D0 ₁₆			
00D1 ₁₆			
00D2 ₁₆			
00D3 ₁₆			
00D4 ₁₆	AD control register 2	ADCON2	00 ₁₆
00D5 ₁₆			
00D6 ₁₆	AD control register 0	ADCON0	00000XXX ₂
00D7 ₁₆	AD control register 1	ADCON1	00 ₁₆
00D8 ₁₆			
00D9 ₁₆			
00DA ₁₆			
00DB ₁₆			
00DC ₁₆			
00DD ₁₆			
00DE ₁₆			
00DF ₁₆			
00E0 ₁₆	Port P0 register	P0	XX ₁₆
00E1 ₁₆	Port P1 register	P1	XX ₁₆
00E2 ₁₆	Port P0 direction register	PD0	00 ₁₆
00E3 ₁₆	Port P1 direction register	PD1	00 ₁₆
00E4 ₁₆			
00E5 ₁₆	Port P3 register	P3	XX ₁₆
00E6 ₁₆			
00E7 ₁₆	Port P3 direction register	PD3	00 ₁₆
00E8 ₁₆	Port P4 register	P4	XX ₁₆
00E9 ₁₆			
00EA ₁₆	Port P4 direction register	PD4	00 ₁₆
00EB ₁₆			
00EC ₁₆			
00ED ₁₆			
00EE ₁₆			
00EF ₁₆			
00F0 ₁₆			
00F1 ₁₆			
00F2 ₁₆			
00F3 ₁₆			
00F4 ₁₆			
00F5 ₁₆			
00F6 ₁₆			
00F7 ₁₆			
00F8 ₁₆			
00F9 ₁₆			
03FA ₁₆			
00FB ₁₆			
00FC ₁₆	Pull-up control register 0	PUR0	00XX0000 ₂
00FD ₁₆	Pull-up control register 1	PUR1	XXXXXXXX ₂
00FE ₁₆	Port P1 drive capacity control register	DRR	00 ₁₆
00FF ₁₆	Timer C output control register	TCOUT	00 ₁₆
~ ~ ~			
01B3 ₁₆	Flash memory control register 4	FMR4	01000000 ₂
01B4 ₁₆			
01B5 ₁₆	Flash memory control register 1	FMR1	1000000X ₂
01B6 ₁₆			
01B7 ₁₆	Flash memory control register 0	FMR0	0000001 ₂
0FFF ₁₆	Option function select register ⁽²⁾	OFS	Note 2

X: Undefined

NOTES:

- The blank areas, 0100₁₆ to 01B2₁₆ and 01B8₁₆ to 02FF₁₆ are reserved and cannot be used by users.
- The watchdog timer control bit is assigned. Refer to "Figure 11.2 OFS, WDC, WDTR and WDTs registers" of Hardware Manual for details.

5. Electrical Characteristics

Table 5.1 Absolute Maximum Ratings

Symbol	Parameter	Condition	Rated value	Unit
V _{CC}	Supply voltage	V _{CC} =AV _{CC}	-0.3 to 6.5	V
AV _{CC}	Analog supply voltage	V _{CC} =AV _{CC}	-0.3 to 6.5	V
V _I	Input voltage		-0.3 to V _{CC} +0.3	V
V _O	Output voltage		-0.3 to V _{CC} +0.3	V
P _d	Power dissipation	T _{opr} =25 °C	300	mW
T _{opr}	Operating ambient temperature		-20 to 85 / -40 to 85 (D version)	°C
T _{stg}	Storage temperature		-65 to 150	°C

Table 5.2 Recommended Operating Conditions

Symbol	Parameter	Conditions	Standard			Unit
			Min.	Typ.	Max.	
V _{CC}	Supply voltage		2.7		5.5	V
AV _{CC}	Analog supply voltage			V _{CC} ³		V
V _{SS}	Supply voltage			0		V
AV _{SS}	Analog supply voltage			0		V
V _{IH}	"H" input voltage		0.8V _{CC}		V _{CC}	V
V _{IL}	"L" input voltage		0		0.2V _{CC}	V
I _{OH (sum)}	"H" peak all output currents	Sum of all pins' IOH (peak)			-60.0	mA
I _{OH (peak)}	"H" peak output current				-10.0	mA
I _{OH (avg)}	"H" average output current				-5.0	mA
I _{OL (sum)}	"L" peak all output currents	Sum of all pins' IOL (peak)			60	mA
I _{OL (peak)}	"L" peak output current	Except P10 to P17			10	mA
		P10 to P17	Drive ability HIGH		30	mA
I _{OL (avg)}	"L" average output current	Except P10 to P17			10	mA
		P10 to P17	Drive ability LOW		15	mA
f (XIN)	Main clock input oscillation frequency	3.0V ≤ V _{CC} ≤ 5.5V	0		5	MHz
		2.7V ≤ V _{CC} < 3.0V	0		10	MHz

Note

- 1: Referenced to V_{CC} = AV_{CC} = 2.7 to 5.5V at T_{opr} = -20 to 85 °C / -40 to 85 °C unless otherwise specified.
- 2: The mean output current is the mean value within 100ms.
- 3: Set V_{CC}=AV_{CC}

Table 5.3 A/D Conversion Characteristics

Symbol	Parameter		Measuring condition	Standard			Unit
				Min.	Typ.	Max.	
-	Resolution		$V_{ref} = V_{CC}$			10	Bit
-	Absolute accuracy	10 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{ref} = V_{CC} = 5.0\text{V}$			± 3	LSB
		8 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{ref} = V_{CC} = 5.0\text{V}$			± 2	LSB
		10 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{ref} = V_{CC} = 3.3\text{V}^3$			± 5	LSB
		8 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{ref} = V_{CC} = 3.3\text{V}^3$			± 2	LSB
R_{LADDER}	Ladder resistance		$V_{REF} = V_{CC}$	10		40	k Ω
t_{CONV}	Conversion time	10 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{ref} = V_{CC} = 5.0\text{V}$	3.3			μs
		8 bit mode	$f_{AD} = 10 \text{ MHz}$, $V_{ref} = V_{CC} = 5.0\text{V}$	2.8			μs
V_{REF}	Reference voltage				V_{CC}^4		V
V_A	Analog input voltage			0		V_{ref}	V
-	A/D operation clock frequency ²	Without sample & hold		0.25		10	MHz
		With sample & hold		1.0		10	MHz

Note

- 1: Referenced to $V_{CC} = AV_{CC} = 2.7$ to 5.5V at $T_{opr} = -20$ to 85°C / -40 to 85°C unless otherwise specified.
- 2: When f_{AD} is 10 MHz more, divide the f_{AD} and make A/D operation clock frequency (ϕ_{AD}) lower than 10 MHz.
- 3: When the AV_{CC} is less than 4.2V, divide the f_{AD} and make A/D operation clock frequency (ϕ_{AD}) lower than $f_{AD}/2$.
- 4: Set $V_{CC} = V_{ref}$

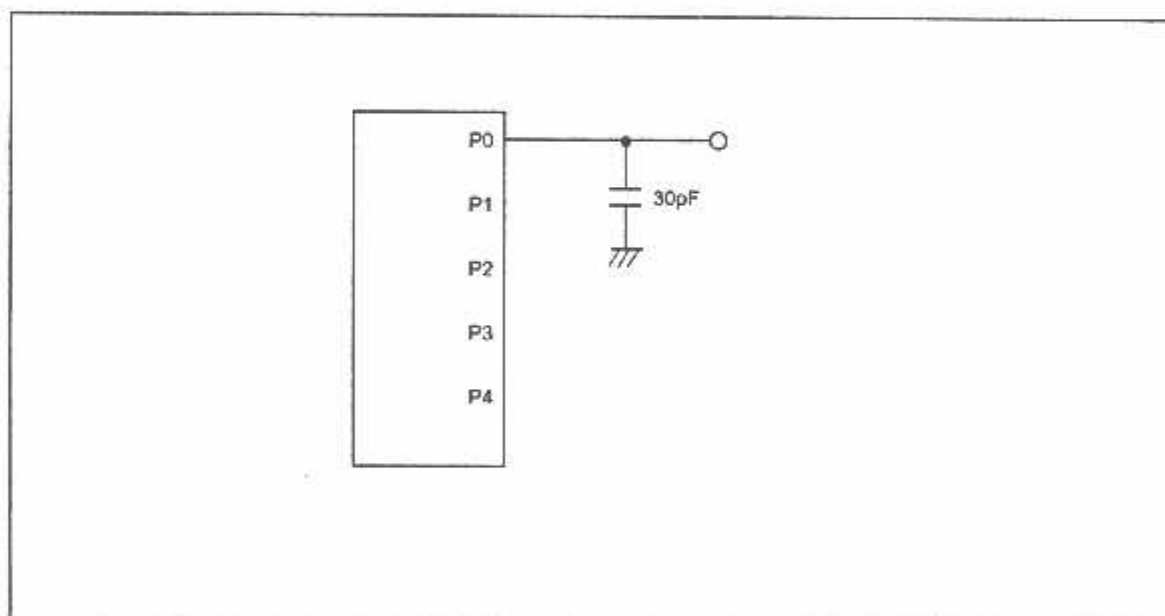


Figure 5.1 Port P0 to P4 measurement circuit

Table 5.4 Flash Memory (Program area) Electrical Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max	
—	Program/Erase cycle ²		1000 ³	—	—	cycle
—	Byte program time	V _{CC} = 5.0 V at Topr = 25 °C	—	60	—	μs
—	Block erase time	V _{CC} = 5.0 V at Topr = 25 °C	—	0.4	—	s
t _{d(SR-ES)}	Time delay from Suspend Request until Erase Suspend		—	—	8	ms
—	Erase Suspend Request Interval		10	—	—	ms
—	Program, Erase Voltage		2.7	—	5.5	V
—	Read Voltage		2.7	—	5.5	V
—	Program, Erase Temperature		0	—	60	°C
—	Data-retention duration	Topr = 55 °C	20	—	—	year

Table 5.5 Flash Memory (Data area Block A, Block B) Electrical Characteristics⁴

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max	
—	Program/Erase endurance ²		10000 ³	—	—	times
—	Byte program time(program/erase endurance ≤1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	50	400	μs
—	Byte program time(program/erase endurance >1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	65	—	μs
—	Block erase time(program/erase endurance ≤1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	0.2	9	s
—	Block erase time(program/erase endurance >1000 times)	V _{CC} = 5.0 V at Topr = 25 °C	—	0.3	—	s
t _{d(SR-ES)}	Time delay from Suspend Request until Erase Suspend		—	—	8	ms
—	Erase Suspend Request Interval		10	—	—	ms
—	Program, Erase Voltage		2.7	—	5.5	V
—	Read Voltage		2.7	—	5.5	V
—	Program/Erase Temperature		-20(-40) ⁸	—	85	°C
—	Data-retention duration	Topr = 55 °C	20	—	—	year

Note

1: Referenced to V_{CC}=AV_{CC}=2.7 to 5.5V at Topr = 0°C to 60°C unless otherwise specified.

2: Definition of Program/Erase

The cycle of Program/Erase shows a cycle for each block.

If the program/erase number is "n" (n = 1000, 10000), "n" times erase can be performed for each block.

For example, if performing one-byte write to the distinct addresses on Block A of 2K-byte block 2048 times and then erasing that block, the number of Program/Erase cycles is one time.

However, performing multiple writes to the same address before an erase operation is prohibited (overwriting prohibited).

3: Maximum numbers of Program/Erase cycles for which all electrical characteristics is guaranteed.

4: Table 16.5 applies for Block A or B when the Program/Erase cycles are more than 1000. The byte program time up to 1000 cycles are the same as that of the program area (see Table 5.4).

5: To reduce the number of Program/Erase cycles, a block erase should ideally be performed after writing in series as many distinct addresses (only one time each) as possible. If programming a set of 16 bytes, write up to 128 sets and then erase them one time. This will result in ideally reducing the number of Program/Erase cycles. Additionally, averaging the number of Program/Erase cycles for Block A and B will be more effective. It is important to track the total number of block erases and restrict the number.

6: If error occurs during block erase, attempt to execute the clear status register command, then the block erase command at least three times until the erase error disappears.

7: Customers desiring Program/Erase failure rate information should contact their Renesas technical support representative.

8: -4C °C for D version.

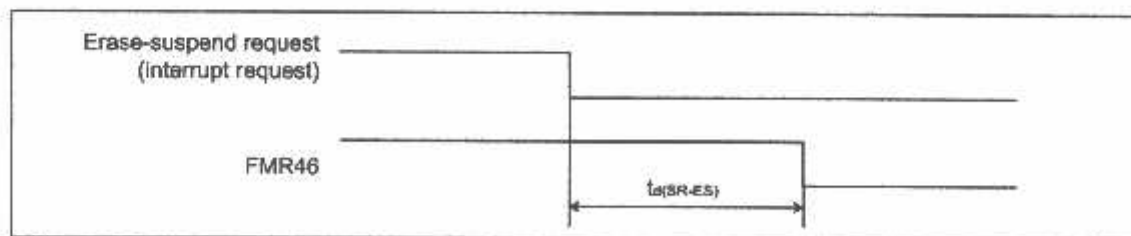


Figure 5.2 Time delay from Suspend Request until Erase Suspend

Table 5.6 Voltage Detection Circuit Electrical Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
Vdet	Voltage detection level		3.3	3.8	4.3	V
	Voltage detection interrupt request generating time ²			40		μs
	Voltage detection circuit self consumption current	VC27=1, VCC=5.0V		600		nA
td(E-A)	Waiting time until voltage detection circuit operation starts ³				20	μs
Vccmin	Microcomputer operation voltage minimum value		2.7			V

NOTES:

- The measuring condition is $V_{CC}=AV_{CC}=2.7V$ to $5.5V$ and $T_{opr}=-40^{\circ}C$ to $85^{\circ}C$.
- This shows the time until the voltage detection interrupt request is generated since the voltage passes Vdet.
- This shows the required time until the voltage detection circuit operates when setting to "1" again after setting the VC27 bit in the VCR2 register to "0".

Table 5.7 Reset Circuit Electrical Characteristics (When Using Hardware Reset 2^{1, 3})

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
Vpor2	Power-on reset valid voltage	$-20^{\circ}C \leq T_{opr} \leq 85^{\circ}C$	—	—	Vdet	V
tw(Vpor2-Vcc)	Supply voltage rising time when power-on reset is canceled ²	$-20^{\circ}C \leq T_{opr} < 85^{\circ}C$, $tw(por2) \geq 0s^1$	—	—	100	ms

NOTES:

- The voltage detection circuit which is embedded in a microcomputer is a factor to generate the hardware reset 2. Refer to 5.1.2 Hardware Reset 2.
- This condition is not applicable when using $V_{CC} \geq 1.0V$.
- When turning power on after the external power has been held below the valid voltage for greater than 10 seconds, refer to Table 16.8 Reset Circuit Electrical Characteristics (When Not Using Hardware Reset 2).
- $tw(por2)$ is time to hold the external power below effective voltage (Vpor2).

Table 5.8 Reset Circuit Electrical Characteristics (When Not Using Hardware Reset 2)

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
Vpor1	Power-on reset valid voltage	$-20^{\circ}C \leq T_{opr} < 85^{\circ}C$	—	—	0.1	V
tw(Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	$0^{\circ}C \leq T_{opr} \leq 85^{\circ}C$, $tw(por1) \geq 10s^1$	—	—	100	ms
tw(Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	$-20^{\circ}C \leq T_{opr} < 0^{\circ}C$, $tw(por1) \geq 30s^1$	—	—	100	ms
tw(Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	$-20^{\circ}C \leq T_{opr} < 0^{\circ}C$, $tw(por1) \geq 10s^1$	—	—	1	ms
tw(Vpor1-Vdet)	Supply voltage rising time when power-on reset is canceled	$0^{\circ}C \leq T_{opr} \leq 85^{\circ}C$, $tw(por1) \geq 1s^1$	—	—	0.5	ms

NOTES:

- When not using hardware reset 2, use with $V_{CC} \geq 2.7V$.
- $tw(por1)$ is time to hold the external power below effective voltage (Vpor1).

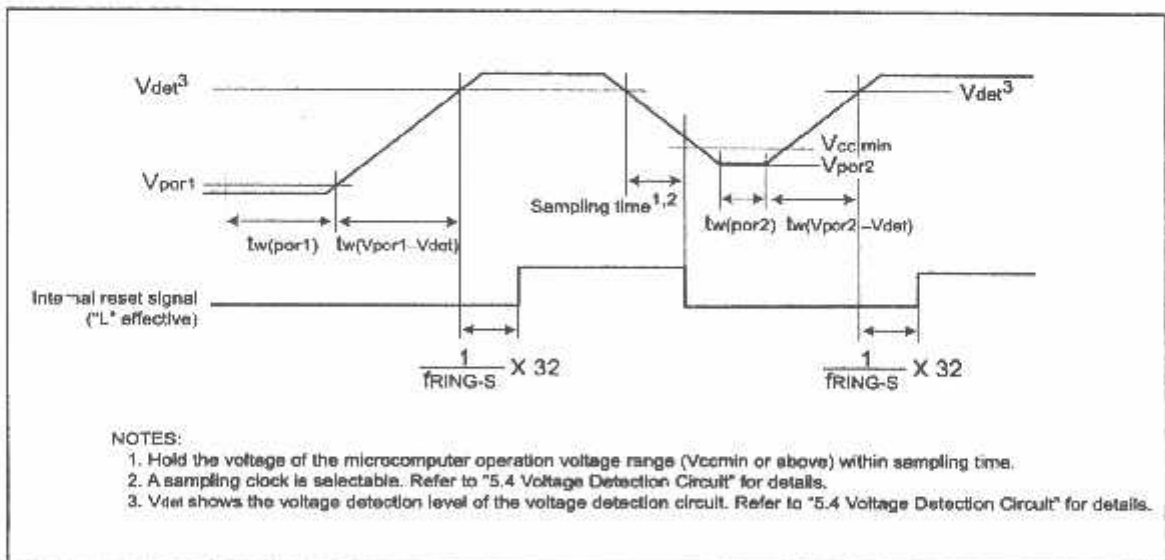


Figure 5.3 Reset Circuit Electrical Characteristics

Table 5.9 High-speed On-Chip Oscillator Circuit Electrical Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
—	High-speed on-chip oscillator frequency $f / (t0(HRoffst) + k1(HR))$ when the reset is released	VCC=5.0V, Topr=25 °C Set "401e" in the HR1 register	—	8	—	MHz
t0(HRoffst)	Settable high-speed on-chip oscillator minimum period	VCC=5.0V, Topr=25 °C Set "001e" in the HR1 register	—	51	—	ns
t0(HR)	High-speed on-chip oscillator period adjusted unit	Differences when setting "01-e" and "001e" in the HR register	—	1	—	ns
—	High-speed on-chip oscillator temperature dependence(1)	Frequency fluctuation in temperature range of -10 °C to 50 °C	—	±5	—	%
—	High-speed on-chip oscillator temperature dependence(2)	Frequency fluctuation in temperature range of -40 °C to 85 °C	—	±10	—	%

NOTES:

1. The measuring condition is Vcc=AVcc=5.0 V and Topr=25 °C.

Table 5.10 Power Circuit Timing Characteristics

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
t0(P-R)	Time for internal power supply stabilization during power-on ²		1		2000	μs
t0(R-S)	STCP release time ³				150	μs

Note

1: The measuring condition is Vcc=AVcc=2.7 to 5.5 V and Topr=25 °C.

2: This shows the wait time until the internal power supply generating circuit is stabilized during power-on.

3: This shows the time until BCLK starts from the interrupt acknowledgement to cancel stop mode.

Table 5.11 Electrical Characteristics (1) [Vcc=5V]

Symbol	Parameter		Measuring condition	Standard			Unit
				Min.	Typ.	Max.	
Voh	"H" output voltage	Except Xout	Ioh=-5mA Ioh=-200μA	Vcc-2.0	—	Vcc	V
		Xout	Drive capacity HIGH Ioh=-1 mA Drive capacity LOW Ioh=-500μA	Vcc-0.3	—	Vcc	V
Vol	"L" output voltage	P10 to P17 Except Xout ¹	Iol = 5 mA Iol = 200 μA	Vcc-2.0	—	Vcc	V
		P10 to P17	Drive capacity HIGH Iol = 15 mA	—	—	2.0	V
			Drive capacity LOW Iol = 5 mA	—	—	2.0	V
			Drive capacity LOW Iol = 200 μA	—	—	0.45	V
		XOUT	Drive capacity HIGH Iol = 1 mA	—	—	2.0	V
			Drive capacity LOW Iol = 500 μA	—	—	2.0	V
Vt-Vt	Hysteresis	INT0, INT1, INT2, INT3, K0, K1, K2, K3, CNTR0, CNTR1, TCN, RxD0, RxD1, P45 RESET		0.2	—	1.0	V
Ih	"H" input current		Vt=5V	—	—	5.0	μA
Il	"L" input current		Vt=0V	—	—	-6.0	μA
Rpullup	Pull-up resistance		Vt=0V	30	50	167	kΩ
R001	Feedback resistance	XIN		—	1.0	—	MΩ
fosc0-s	High-speed on-chip oscillator frequency			40	125	250	kHz
VRAM	RAM retention voltage		At stop mode	2.0	—	—	V

Note

1: Referenced to Vcc=AVcc=4.2 to 5.5V at Topr = -20 to 85 °C / -40 to 85 °C, f(XIN)=20MHz unless otherwise specified.

Table 5.12 Electrical Characteristics (2) [V_{CC}=5V]

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
I _{CC}	Power supply current (V _{CC} =3.3 to 5.5V) In single-chip mode, the output pins are open and other pins are V _{SS} .	High-speed mode Xin=20 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz No division		9	15	mA
		Xin=18 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz No division		8	14	mA
		Xin=10 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz No division		5		mA
		Medium-speed mode Xin=20 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8		4		mA
		Xin=18 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8		3		mA
		Xin=10 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8		2		mA
		High-speed on-chip oscillator mode Main clock off High-speed on-chip oscillator on=8 MHz Low-speed on-chip oscillator on=125 kHz No division		4	8	mA
		Main clock off High-speed on-chip oscillator on=8 MHz Low-speed on-chip oscillator on=125 kHz Division by 8		1.5		mA
Low-speed on-chip oscillator mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8		470	900	μA		
Wait mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz When a WAIT instruction is executed* Peripheral clock operation VC27="0"		40	60	μA		
Wait mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz When a WAIT instruction is executed* Peripheral clock off VC27="0"		38	76	μA		
Stop mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator off CM10="1" Peripheral clock off VC27="0"		0.8	3.0	μA		

NOTES

- 1: The power supply current measuring is executed using the measuring program on flash memory.
- 2: Timer Y is operated with timer mode.

Timing requirements (Unless otherwise noted: $V_{CC} = 5V$, $V_{SS} = 0V$ at $T_a = 25^\circ C$) [$V_{CC}=5V$]

Table 5.13 XIN input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_C(XIN)$	XIN input cycle time	50		ns
$t_{WH}(XIN)$	XIN input HIGH pulse width	25		ns
$t_{WL}(XIN)$	XIN input LOW pulse width	25		ns

Table 5.14 CNTR0 input, CNTR1 input, $\overline{INT2}$ input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_C(CNTR0)$	CNTR0 input cycle time	100		ns
$t_{WH}(CNTR0)$	CNTR0 input HIGH pulse width	40		ns
$t_{WL}(CNTR0)$	CNTR0 input LOW pulse width	40		ns

Table 5.15 TCIN input, $\overline{INT3}$ input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_C(TCIN)$	TCIN input cycle time	400 ¹		ns
$t_{WH}(TCIN)$	TCIN input HIGH pulse width	200 ²		ns
$t_{WL}(TCIN)$	TCIN input LOW pulse width	200 ²		ns

NOTES

- 1: When using the Timer C input capture mode, adjust the cycle time above ($1/\text{Timer C count source frequency} \times 3$).
- 2: When using the Timer C input capture mode, adjust the pulse width above ($1/\text{Timer C count source frequency} \times 1.5$).

Table 5.16 Serial Interface

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_C(C\leftarrow)$	CLKi input cycle time	200		ns
$t_W(CKH)$	CLKi input HIGH pulse width	100		ns
$t_W(CKL)$	CLKi input LOW pulse width	100		ns
$t_d(C\rightarrow Q)$	TxDi output delay time		80	ns
$t_h(C\rightarrow Q)$	TxDi hold time	0		ns
$t_{su}(D\rightarrow C)$	RxDi input setup time	35		ns
$t_h(C\rightarrow D)$	RxDi input hold time	90		ns

Table 5.17 External Interrupt $\overline{INT0}$ input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_W(INH)$	$\overline{INT0}$ input HIGH pulse width	250 ¹		ns
$t_W(INL)$	$\overline{INT0}$ input LOW pulse width	250 ²		ns

NOTES

- 1: When selecting the digital filter by the $\overline{INT0}$ input filter select bit, use the $\overline{INT0}$ input HIGH pulse width to the greater value, either ($1/\text{digital filter clock frequency} \times 3$) or the minimum value of standard.
- 2: When selecting the digital filter by the $\overline{INT0}$ input filter select bit, use the $\overline{INT0}$ input LOW pulse width to the greater value, either ($1/\text{digital filter clock frequency} \times 3$) or the minimum value of standard.

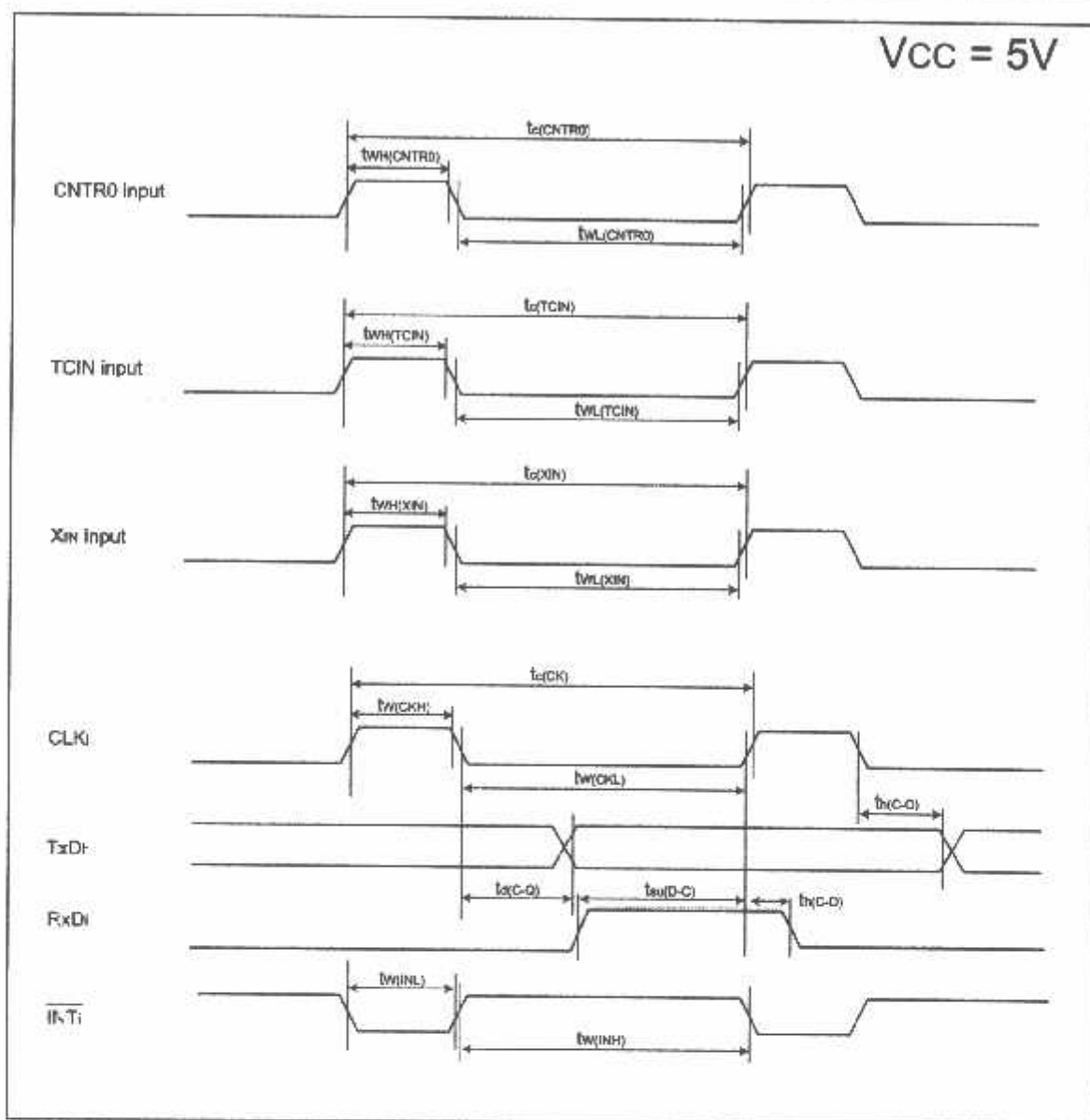


Figure 5.4 Vcc=5V timing diagram

Table 5.18 Electrical Characteristics (3) [Vcc=3V]

Symbol	Parameter		Measuring condition		Standard			Unit
					Min.	Typ.	Max.	
V _{OH}	*H* output voltage	Except X _{OUT}	I _{OH} =-1mA		V _{CC} -0.5	—	V _{CC}	V
		X _{OUT}	Drive capacity HIGH	I _{OH} =-0.1 mA	V _{CC} -0.5	—	V _{CC}	V
			Drive capacity LOW	I _{OH} =-50 μA	V _{CC} -0.5	—	V _{CC}	V
V _{OL}	*L* output voltage	P10 to P17 Except X _{OUT}	I _{OL} = 1 mA		—	—	0.5	V
			P10 to P17	Drive capacity HIGH	I _{OL} = 2 mA	—	—	0.5
		Drive capacity LOW		I _{OL} = 1 mA	—	—	0.5	V
		X _{OUT}	Drive capacity HIGH	I _{OL} = 0.1 mA	—	—	0.5	V
			Drive capacity LOW	I _{OL} =50 μA	—	—	0.5	V
V _{T+V_{T-}}	Hysteresis	INT ₀ , INT1, INT2, INT3, K16, K17, K18, K19, CNTR0, CNTR1, TCIN, RxD0, RxD1, P48			0.2	—	0.8	V
		RESET			0.2	—	1.8	V
I _H	*H* input current			V _H =3V	—	—	4.0	μA
I _L	*L* input current			V _L =0V	—	—	-4.0	μA
R _{PULLUP}	Pull-up resistance			V _L =0V	88	160	500	kΩ
R _{FB}	Feedback resistance	X _{IN}			—	3.0	—	MΩ
f _{OSC-S}	Low-speed on-chip oscillator frequency				40	125	250	kHz
V _{RAM}	RAM retention voltage			A: stop mode	2.0	—	—	V

Note

1: Referenced to V_{CC}=AV_{CC}=2.7 to 3.3V at Topr = -20 to 85 °C / -40 to 85 °C, f(X_{IN})=10MHz unless otherwise specified.

Table 5.19 Electrical Characteristics (4) [V_{CC}=3V]

Symbol	Parameter	Measuring condition	Standard			Unit
			Min.	Typ.	Max.	
I _{CC}	Power supply current (V _{CC} =2.7 to 3.3V) In single-chip mode, the output pins are open and other pins are V _{SS} .	High-speed mode Xc=20 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz No division Xc=16 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz No division Xc=10 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz No division		8	13	mA
				7	12	mA
				6		mA
		Medium-speed mode Xc=20 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8 Xc=16 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8 Xc=10 MHz (square wave) High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8		3		mA
				2.5		mA
				1.8		mA
		High-speed on-chip oscillator mode Main clock off High-speed on-chip oscillator on=6 MHz Low-speed on-chip oscillator on=125 kHz No division Main clock off High-speed on-chip oscillator on=3 MHz Low-speed on-chip oscillator on=125 kHz Division by 9		3.5	7.5	mA
				1.6		mA
		Low-speed on-chip oscillator mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz Division by 8		420	800	μA
		Wait mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz When a WAIT instruction is executed ¹ Peripheral clock operation VC27="0"		37	74	μA
		Wait mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator on=125 kHz When a WAIT instruction is executed ² Peripheral clock off VC27="0"		35	70	μA
		Stop mode Main clock off High-speed on-chip oscillator off Low-speed on-chip oscillator off CM10="1" Peripheral clock off VC27="0"		0.7	3.0	μA

NOTES

- 1: The power supply current measuring is executed using the measuring program on fresh memory.
- 2: Timer Y is operated with timer mode.

Timing requirements (Unless otherwise noted: $V_{CC} = 3V$, $V_{SS} = 0V$ at $T_a = 25\text{ }^\circ\text{C}$) [$V_{CC}=3V$]

Table 5.20 XIN input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(XIN)}$	XIN input cycle time	100		ns
$t_{WH(XIN)}$	XIN input HIGH pulse width	40		ns
$t_{WL(XIN)}$	XIN input LOW pulse width	40		ns

Table 5.21 CNTR0 input, CNTR1 input, INT2 input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CNTR0)}$	CNTR0 input cycle time	300		ns
$t_{WH(CNTR0)}$	CNTR0 input HIGH pulse width	120		ns
$t_{WL(CNTR0)}$	CNTR0 input LOW pulse width	120		ns

Table 5.22 TCIN input, INT3 input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(TCIN)}$	TCIN input cycle time	1200 ¹		ns
$t_{WH(TCIN)}$	TCIN input HIGH pulse width	600 ²		ns
$t_{WL(TCIN)}$	TCIN input LOW pulse width	600 ²		ns

NOTES

- 1 : When using the Timer C input capture mode, adjust the cycle time above (1/ Timer C count source frequency x 3).
- 2 : When using the Timer C input capture mode, adjust the pulse width above (1/ Timer C count source frequency x 1.5).

Table 5.23 Serial Interface

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{c(CLK)}$	CLKi input cycle time	300		ns
$t_{W(CKH)}$	CLKi input HIGH pulse width	150		ns
$t_{W(CKL)}$	CLKi input LOW pulse width	150		ns
$t_{d(C-Q)}$	TxDi output delay time		160	ns
$t_{h(C-Q)}$	TxDi hold time	0		ns
$t_{su(D-C)}$	RxDi input setup time	55		ns
$t_{h(C-D)}$	RxDi input hold time	90		ns

Table 5.24 External interrupt INT0 input

Symbol	Parameter	Standard		Unit
		Min.	Max.	
$t_{W(INH)}$	INT0 input HIGH pulse width	380 ¹		ns
$t_{W(INL)}$	INT0 input LOW pulse width	380 ²		ns

NOTES

- 1 : When selecting the digital filter by the INT0 input filter select bit, use the INT0 input HIGH pulse width to the greater value, either (1/ digital filter clock frequency x 3) or the minimum value of standard.
- 2 : When selecting the digital filter by the INT0 input filter select bit, use the INT0 input LOW pulse width to the greater value, either (1/ digital filter clock frequency x 3) or the minimum value of standard.

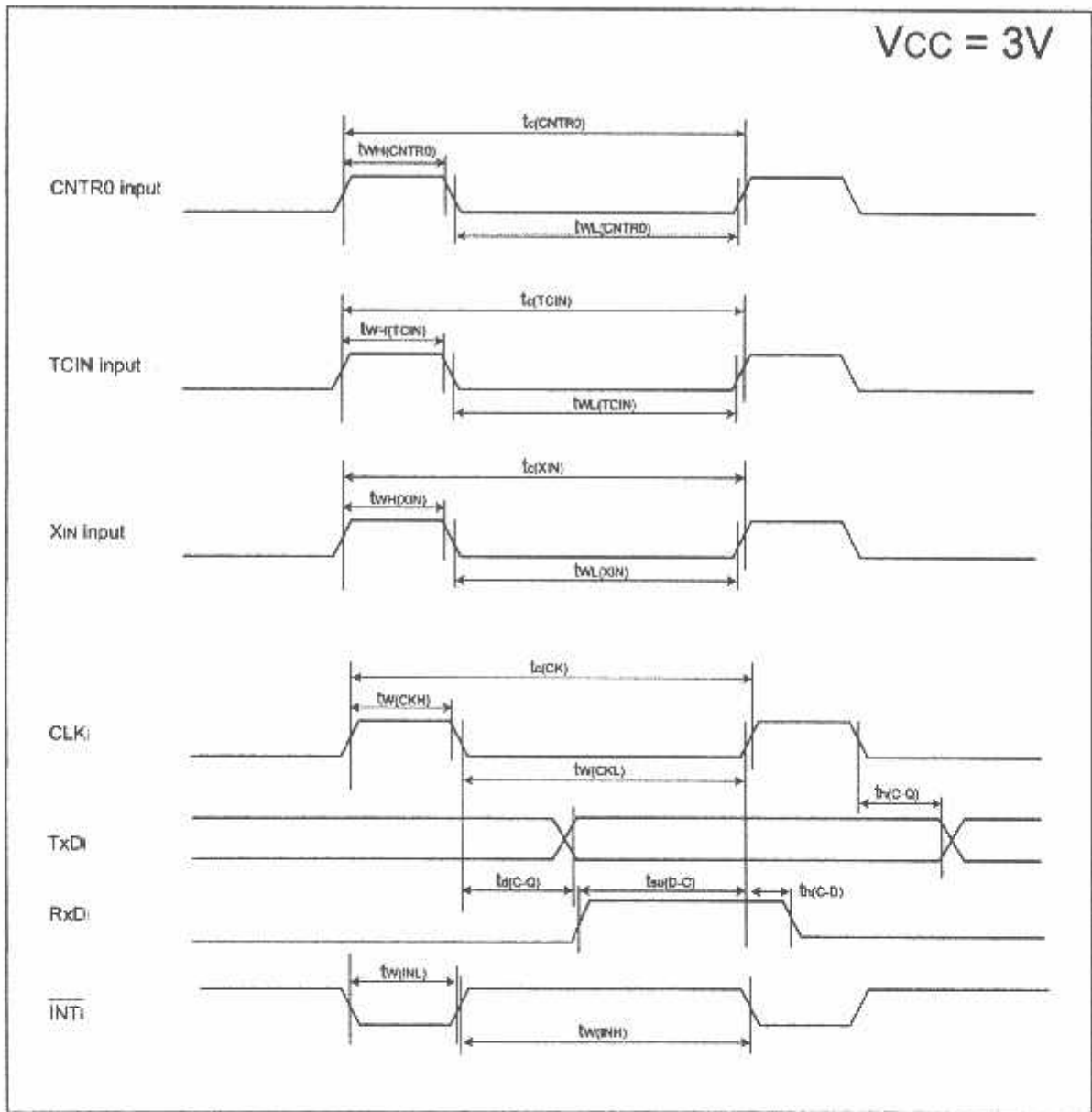
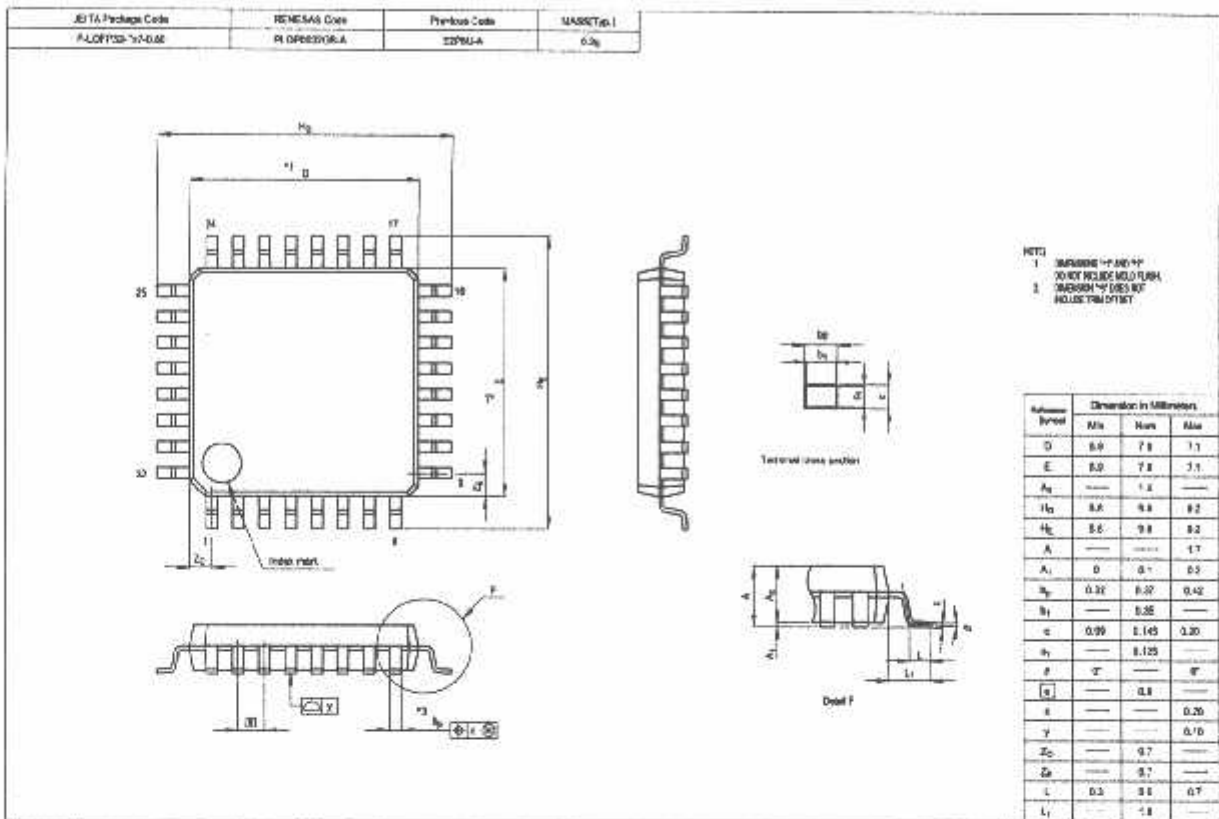
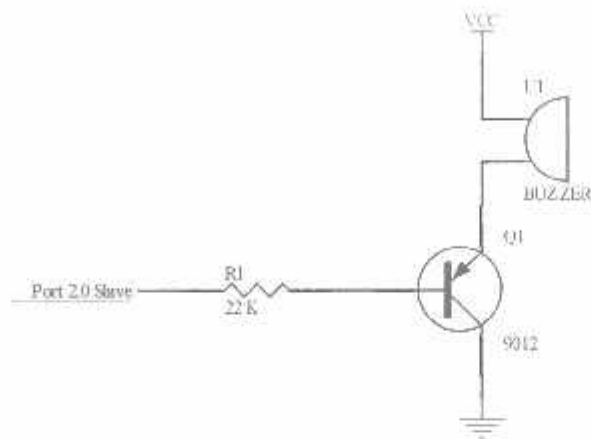


Figure 5.5 Vcc=3V timing diagram

Package Dimensions



petugas operator untuk mempersiapkan jalur di stasiun yang akan digunakan oleh kereta api. *Buzzer* dikendalikan oleh sebuah rangkaian *driver* yang ditunjukkan gambar dibawah ini :



Gambar 3.8. Rangkaian *Driver Buzzer*

Untuk mencari harga dari resistansi basis transistor 9012, sehingga akan bekerja sebagai transistor saklar, maka dengan data-data pengukuran Hfe Transistor didapat :

$$\text{Hfe Transistor 9012} = \beta_{dc} = 170$$

$$R_c = R \text{ Buzzer} = 160 \Omega$$

$$I_c = \frac{V_{cc}}{R_c}$$

$$I_c = \frac{5}{160}$$

$$= 31,25 \text{ mA}$$

$$I_b = \frac{I_c}{Hfe}$$

$$= \frac{0,03125}{170}$$

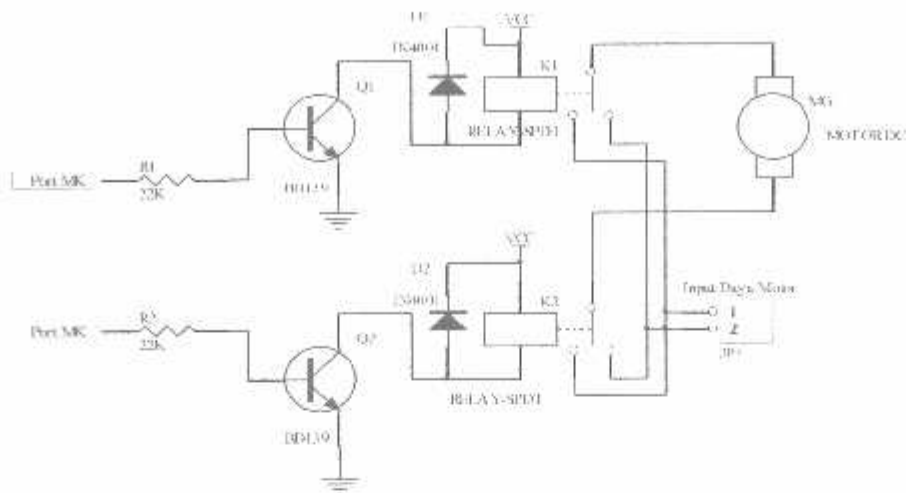
$$= 0,184 \text{ mA}$$

$$\begin{aligned}
 R_b &= \frac{V_{in} - V_{be}}{I_b} \\
 &= \frac{5 - 0,7}{0,184} \\
 &= 23,37 \text{ K}\Omega
 \end{aligned}$$

Karena dipasaran tidak ada resistor dengan harga 23,37 K Ω maka dipakai resistor 22K Ω .

3.2.7. Driver Motor

Di dalam perancangan *driver* motor dihubungkan dengan *Port* dari mikrokontroler *slave 2* AT89S51. Sistem kerjanya adalah apabila salah satu pin di-*ground*-kan pada motor dc dan pin yang satunya dihubungkan ke *Vcc* maka mengakibatkan berputarnya motor. Untuk mengubah arah putaran motor dapat dilakukan dengan cara mengubah data pada kedua pin tersebut (*input data*) secara bergantian. Hal ini berlaku juga untuk pin-pin lain yang terhubung ke masing-masing *driver* motor. Jadi setiap motor dc pada wessel memerlukan 2 *input data* dari mikrokontroler. Untuk menggerakkan motor dc ini diperlukan transistor driver, disini menggunakan transistor BD 139 untuk memicu *relay* untuk menggerakkan motor ke kiri dan ke kanan.



Gambar 3.9. Rangkaian *Driver* Motor

Gambar 3.9. adalah rangkaian *driver* motor dc untuk penggerak wessel. Cara kerja dari rangkaian ini adalah pada saat mikrokontroler memberikan logika *high* pada basis transistor pertama maka akan mensaturasikan transistor dan memberikan beda potensial pada relay yang pertama. *Relay* bekerja dan membuat posisi kontak pada arah *normally open* sehingga *pin* yang pertama motor dc mendapat tegangan positif, kemudian untuk transistor yang kedua pada saat logika *high* diberikan dari mikrokontroler maka akan mensaturasikan transistor dan memberikan beda potensial juga pada *relay* yang kedua. *Relay* bekerja dan membuat posisi kontak pada arah *normally close* sehingga *pin* kedua dari motor *gear box* mendapat tegangan negatif, hal ini menyebabkan motor *gear box* berputar ke kanan dan kiri, transistor akan *cut off* dan saturasi sesuai dengan logika yang diberikan dari mikrokontroler *slave 2* AT89S51 yang bergantian. Untuk mencari harga dari resistansi basis transistor BD 139, sehingga akan bekerja sebagai transistor saklar, maka dengan data-data pengukuran Hfe Transistor didapat :

$$H_{fe} \text{ Transistor BD 139} = \beta_{dc} = 160$$

$$R_c = R \text{ Relay} = 420 \Omega$$

$$I_c = \frac{V_{cc}}{R_c}$$

$$I_c = \frac{12}{420}$$

$$= 0,02857 \text{ A}$$

$$I_b = \frac{I_c}{H_{fe}}$$

$$= \frac{0,02857}{160}$$

$$= 0,00017856 \text{ A}$$

$$R_b = \frac{V_{in} - V_{be}}{I_b}$$

$$= \frac{5 - 0,7}{0,00017856}$$

$$= 24081 \Omega$$

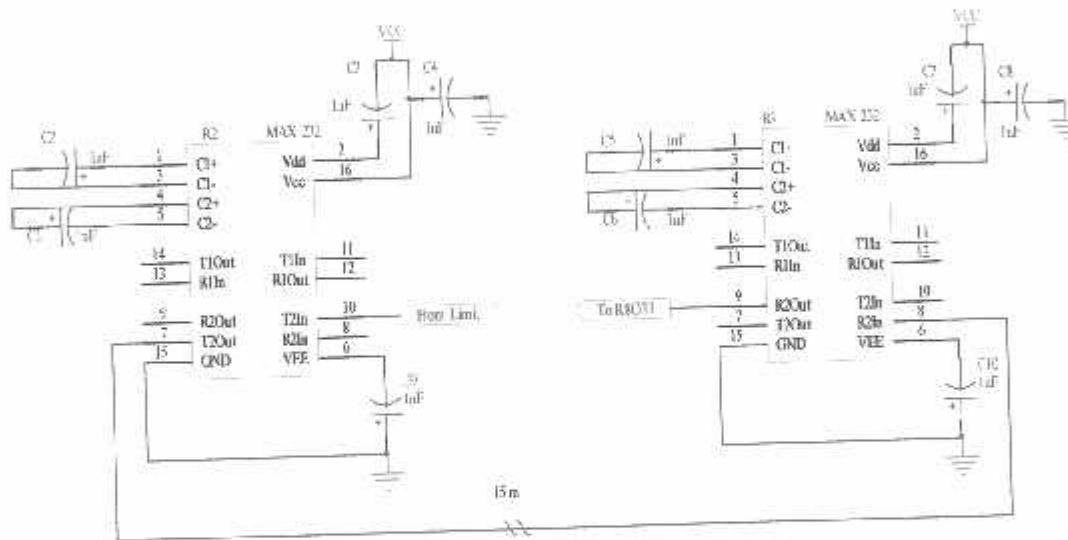
$$= 24 \text{ K}\Omega$$

Karena yang ada dipasaran yang mendekati 24 K Ω maka dipakai resistor 22K Ω .

3.2.8. Rangkaian IC MAX 232

Rangkaian *converter* MAX 232 digunakan untuk mengubah level tegangan TTL menjadi level tegangan RS 232 maupun sebaliknya. IC MAX 232 memiliki charge pump yang akan membangkitkan +10V dan -10V dari sumber +5V tunggal. MAX 232 memiliki sepasang terminal masukan dan sepasang

terminal keluaran yang bekerja pada level tegangan TTL yang berhubungan dengan sepasang terminal masukan dan keluaran yang bekerja pada level tegangan RS 232. Sebagai saluran penerima data dari detektor kereta api yaitu pin 10 (T2_{IN}) dari IC MAX 232 1 kemudian keluaran RS 232 pada pin 7 (T2_{OUT}) IC MAX 232 1 terhubung dengan pin 8 (R2_{IN}) pada IC MAX 232 2. Kemudian keluaran data pada pin 9 (R2_{OUT}) dari IC MAX 232 2 dihubungkan dengan mikrokontroler renesas. Detektor kereta api pada kanan operator terhubung dengan port 0.0 mikrokontroler renesas sedangkan detektor kereta api pada kiri operator terhubung dengan port 0.1 mikrokontroler renesas. Antara IC MAX 232 dihubungkan dengan kabel sepanjang ±15 meter.

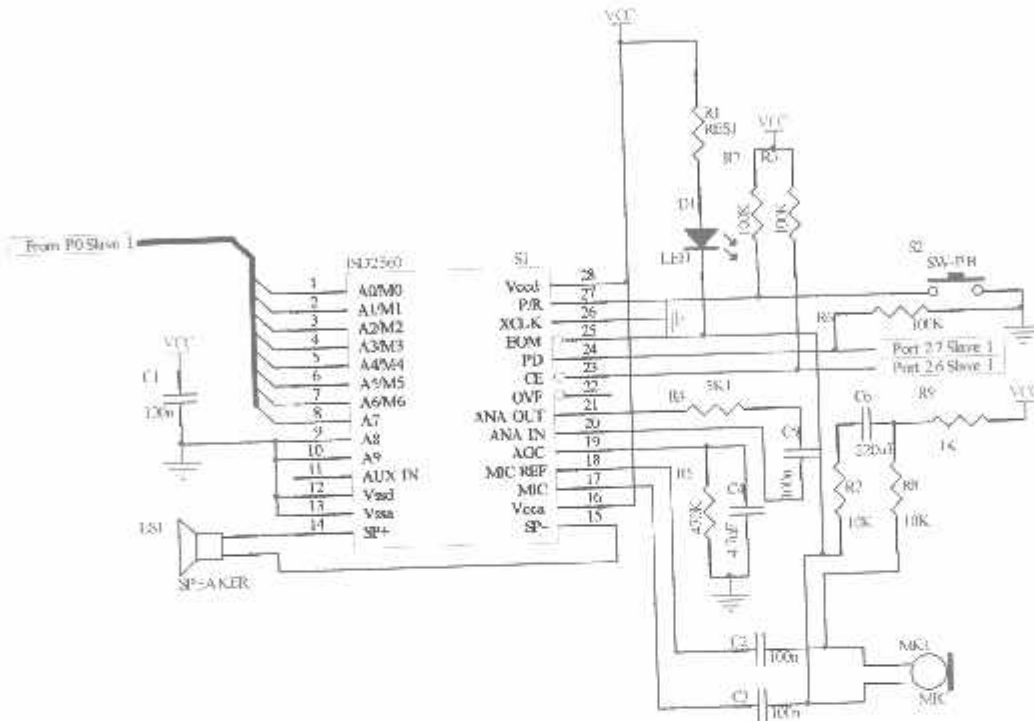


Gambar 3.10. Rangkaian IC Max 232

3.2.9. Rangkaian *Information Storage Device* (ISD 2560)

Pada rangkaian ini suara yang di rekam menggunakan *Information Storage Device* ISD 2560 yang disimpan dalam EEPROM. Berdasarkan data sheet, ISD 2560 ini mampu merekam suara dengan lama perekaman 60 detik dengan alamat

yang berbeda. Rangkaian ISD 2560 ditunjukkan pada gambar 3-10. Pin *address* data menerima masukan 8 bit dari mikrokontroler *Slave 1* pada Port 0. Alamat-alamat ini akan memilih data suara yang mana yang akan dipanggil. Pin *Chip Enable* (CE) berfungsi sebagai pengaktifan (*Enabled*), secara aktif *low* yang terkoneksi dengan *Port 2.6* mikrokontroler *slave 1* AT89S51.



Gambar 3.11. Rangkaian ISD 2560

Untuk menginformasikan kepada calon penumpang bahwa kereta api akan memasuki stasiun, menggunakan IC ISD 2560. Sinyal suara merupakan bentuk sinyal analog kemudian diubah menjadi bentuk digital untuk disimpan ke dalam *memory*. Data-data digital yang sudah tersimpan yang berasal dari data analog (suara) dapat dipanggil kembali dengan memanggil alamat penyimpanan datanya. Proses perekaman pada ISD 2560 adalah sebagai berikut :

1. Pin *Chip Enable* (CE) pin 23 mendapat logika *low*.

2. Memberikan alamat dengan mengatur dip *switch*.
3. Kemudian Pin *Playback/Record* pin 27 mendapat logika *low*.
4. Perekaman dimulai dengan memasukkan data suara pada mikrofon.
5. Pin *Playback/Record* mendapat logika *high* kembali.
6. Mencari alamat terakhir yang ditandai pada *End Of Message* (EOM) pin 25 terjadi pulsa *low* sesaat kemudian *high* kembali.
7. Demikian seterusnya, untuk melanjutkan perekaman alamat terakhir dari sebelumnya diberikan spasi dan memulai perekaman kembali sampai batas waktu dari kemampuan ISD-nya.

Kemudian untuk prosedur pemanggilan data-data suara yang telah direkam adalah sebagai berikut :

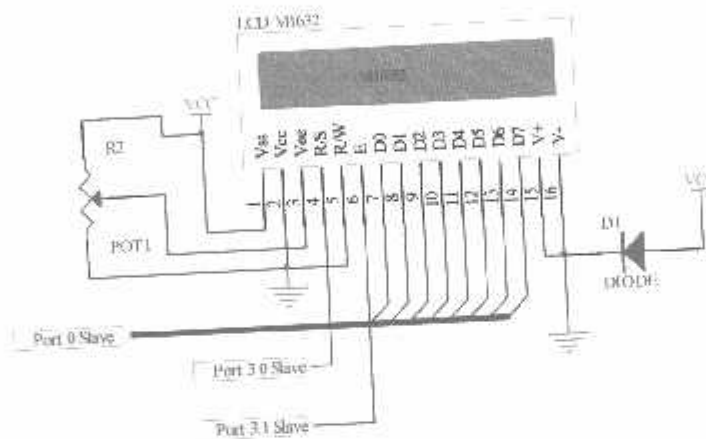
1. Pin *Chip Enable* (CE) pin 23 mendapat logika *low*.
2. Kemudian memberikan logika *low* pada *Playback/Record* pin 27, yang berarti pemanggilan suara dimulai.
3. Menunggu pada *End Of Message* (EOM) pin 25 terdapat pulsa logika *low* sesaat.
4. Bila sudah terdapat pulsa logika *low* sesaat pada EOM, segera dibrikan logika *high* pada pin *Chip Enable* CE, ini menandakan akhir dari data suara pada alamat itu.
5. Demikian seterusnya pada alamat-alamat selanjutnya, sesuai dengan data suara yang diinginkan.

Untuk memudahkan dalam mencari alamat-alamat pada waktu *Record* maupun *Playback* menggunakan tabel logika biner 8 bit. Berikut ini adalah data-data suara dan alamat-alamatnya yang telah direkam dengan lebar data 8 bit.

Tabel 3-1 Data Hasil Perekaman dan Alamatnya Pada ISD 2560

Data Suara yang Direkam	Alamat Data Suara Dalam Biner 8 Bit	Alamat Data Suara Dalam Hexadesimal
"Perhatian-perhatian, kereta api segera memasuki jalur"	0000 0000	00H
"Satu"	0011 1000	38H
"Dua"	0100 0111	47H
"Tiga"	0101 0110	56H
"Empat"	0110 0011	63H

3.2.10. Rangkaian LCD



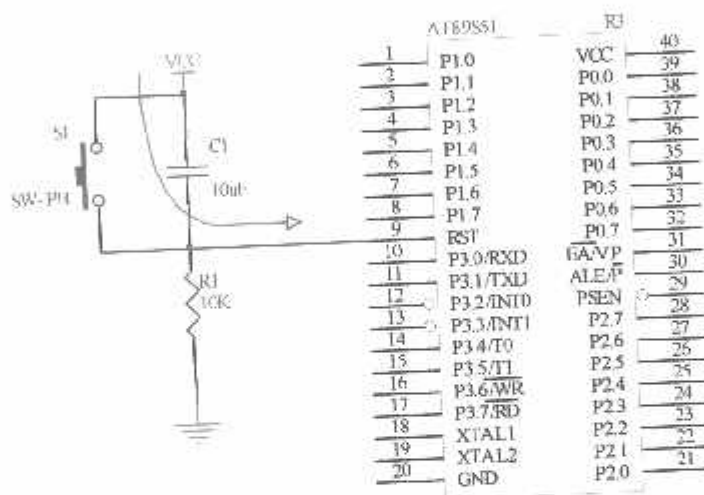
Gambar 3.12. Rangkaian *Liquid Crystal Display* (LCD)

Untuk tampilan dipergunakan LCD Dot Matrik 2 x 16 karakter. Sinyal-sinyal yang diperlukan oleh LCD adalah RS dan Enable, sinyal RS dan Enable dipergunakan sebagai input yang outputnya dipakai untuk mengaktifkan LCD. LCD akan aktif apabila mikrokontroller memberikan instruksi tulis pada LCD. Saat kondisi RS *don't care* dan Enable 0 maka LCD tetap pada kondisi semula, pengiriman data ke LCD dilakukan saat RS berlogika 0 dan enable berlogika 1. Instruksi dikirim pada LCD bila keadaan RS 1 dan Enable 1. Pin LCD ini untuk data terkoneksi pada Port 0 mikrokontoler *slave 1*. Kemudian untuk RS

dihubungkan pada *Port 3.0*, tulis/baca (*Read/Write*) diberikan logika *low* karena disini LCD bersifat menulis data, dan yang terakhir *Enable (E)* dikendalikan dengan *Port 3.1*. Gambar rangkaian LCD ditunjukkan pada gambar 3-11.

3.2.11. Rangkaian *Reset* AT89S51

Untuk mereset mikrokontroler AT89S51, maka pin RST diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal *reset* kapasitor dihubungkan dengan *Vcc* dan sebuah resistor yang dihubungkan ke *ground*. Rangkaian *reset* ditunjukkan dalam gambar 3-12 sebagai berikut :



Gambar 3.13. Rangkaian *Reset*

Karena kristal yang digunakan mempunyai frekuensi sebesar 12 MHz, maka satu periode membutuhkan waktu sebesar :

$$T = \frac{1}{f_{XTAL}} = \frac{1}{12 \text{ MHz}} \text{ s} = 8.33 \times 10^{-8} \text{ s}$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk me-*reset* mikrokontroler adalah :

$$\begin{aligned} \text{Reset (min)} &= T \times \text{periode yang dibutuhkan} \\ &= 8.33 \times 10^{-8} \times 24 = 1.9992 \mu\text{s} \end{aligned}$$

Jadi mikrokontroler membutuhkan waktu minimal 1.9992 μs untuk *m-ereset*. Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C. Dari persamaan diatas dengan menentukan nilai R = 10 k Ω maka nilai C dapat ditentukan:

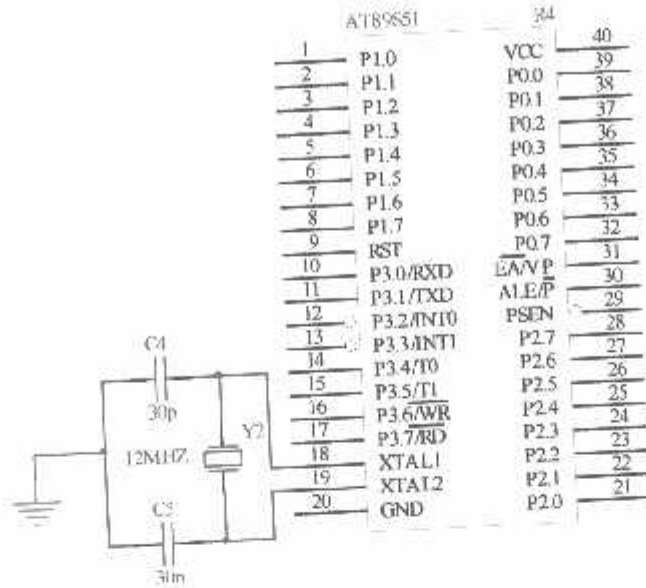
$$\begin{aligned} C &= \frac{T}{R} \\ &= \frac{1,999 \times 10^{-6}}{10 \times 10^3} \\ &= 199,9 \times 10^{-12} \text{ F} \end{aligned}$$

Kapasitor minimal yang dibutuhkan adalah 199,9 pF. Dengan menggunakan kapasitor sebesar 10 μF , maka akan menjamin waktu reset di atas nilai minimal waktu yang dibutuhkan untuk mereset mikrokontroler.

3.2.12. Rangkaian *Clock* AT89S51

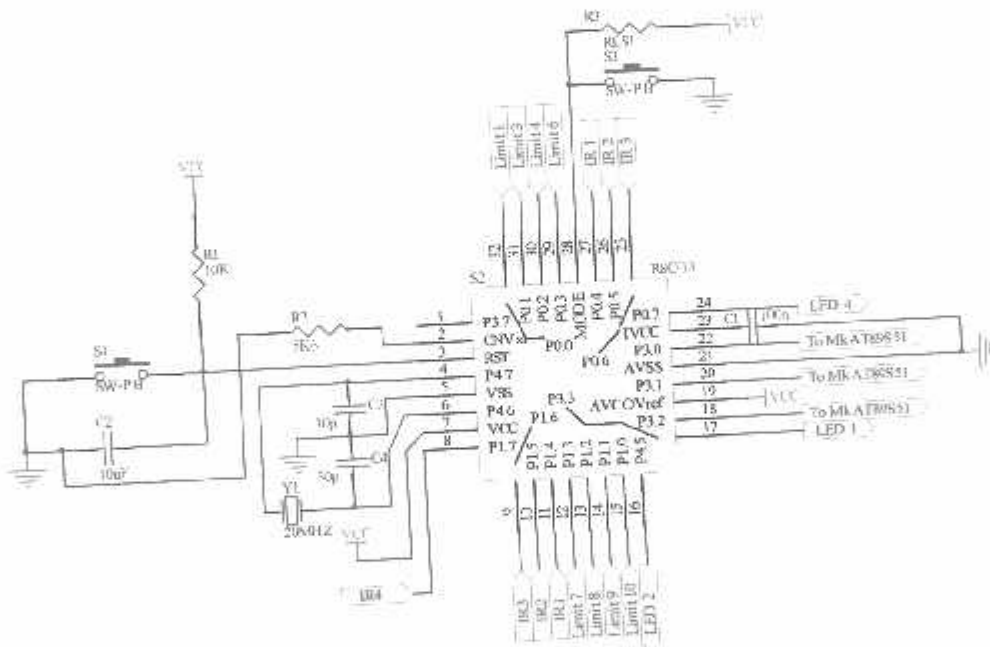
Kecepatan proses yang dilakukan oleh mikrokontroler ditentukan oleh sumber *clock* yang mengendalikan mikrokontroler tersebut. Sistem yang dirancang ini menggunakan osilator internal yang telah tersedia dalam *chip* AT89S51. Untuk menentukan frekuensi osilatonya cukup dengan menghubungkan kristal dalam pin 19 (X_1) dan pin 18 (X_2) serta dua buah kapasitor ke *ground*.

Besarnya kapasitansinya disesuaikan dengan spesifikasi dalam lembar data AT89S51 yaitu 30 pF. Kristal yang digunakan adalah 12 MHz. Gambar 3-13 memperlihatkan rangkaian *clock* yang dirancang.



Gambar 3.14. Rangkaian Clock

3.2.13. Minimum System Mikrokontroler Master Renesas R5F21134FP



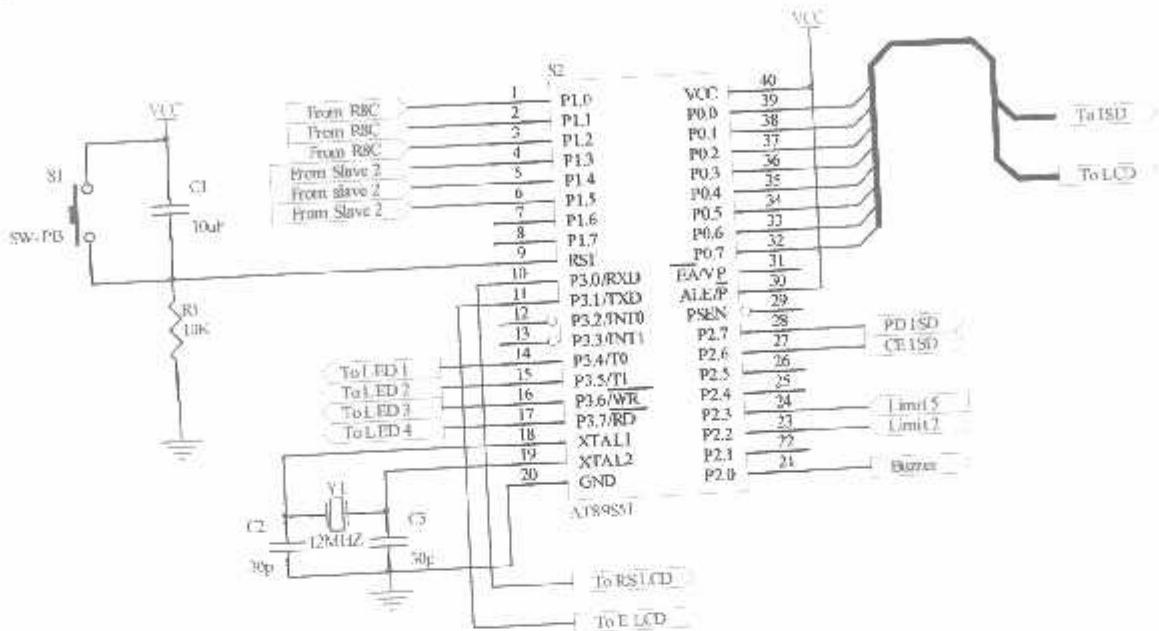
Gambar 3.15. Minimum System Mikrokontroler Renesas R8C Tiny R5F21134FP

Mikrokontroler *Master* merupakan mikrokontroler pengendali utama. Dalam hal ini menggunakan mikrokontroler Renesas R8C Tiny R5F21134FP karena mempertimbangkan keunggulan-keunggulannya dan fasilitas-fasilitas yang dimiliki IC ini. Pembahasan ini telah dijelaskan pada bab-bab sebelumnya. Mikrokontroler *master* ini mempunyai I/O Port yaitu P0.0 – P0.7, P1.0 – P1.7, P3.0 – P3.3, P3.7 dan P4.5 sedangkan P4.6, P4.7 hanya bisa digunakan sebagai *input* saja, bila konfigurasi kristal memakai kristal *internal*. Dalam hal ini yang digunakan *Port* I/O saja. Berikut adalah konfigurasi pin-pin mikrokontroler *master* :

- ❖ P0.0 s.d. P0.3 digunakan sebagai masukan sensor limit.
 - ❖ P0.4 s.d. P0.7 digunakan sebagai output indikator led penggunaan jalur di stasiun.
 - ❖ P1.0 s.d. P1.3 digunakan sebagai masukan sensor limit deteksi keberadaan kereta api pada jalur kereta api di stasiun.
 - ❖ P1.4 s.d. P1.7 digunakan sebagai masukan sensor photodiode.
 - ❖ P3.0 memberikan pulsa ke mikrokontroler *Slave* AT89S51 untuk membaca data perintah baru.
 - ❖ P3.1 dan P 3.2 digunakan sebagai input data ke mikrokontroler *Slave* AT89S51.
 - ❖ P3.3 dan P 4.5 digunakan sebagai output display arah kedatangan kereta api.
-

3.2.14. Minimum System Mikrokontroler Slave 1 AT89S51

Mikrokontroler *Slave 1* menggunakan mikrokontroler AT89S51. Mikrokontroler ini bertugas mengendalikan sistem suara pada *ISD*, buzzer, menerima masukan dari sensor limit serta tampilan *LCD*.



Gambar 3.16. Minimum System Mikrokontroler Slave 1 AT89S51

- ❖ Port 0, (D0 - D7) memberikan input data ke LCD dan ISD 2560.
- ❖ P1.0 menerima pulsa dari mikrokontroler Renesas R8C/13.
- ❖ P1.1 dan P1.2 menerima data perintah dari mikrokontroler Renesas R8C/13.
- ❖ P1.3 menerima pulsa dari mikrokontroler *slave 2* AT89S51.
- ❖ P1.4 dan P1.5 menerima data dari mikrokontroler *slave 2* AT89S51.
- ❖ P2.0 berfungsi untuk mengaktifkan buzzer.
- ❖ P2.2 dan p2.3 menerima data dari sensor limit.
- ❖ P2.6 berfungsi untuk mengaktifkan CE ISD
- ❖ P2.7 berfungsi untuk mengaktifkan PD ISD.

100

100

100

100

100



DAFTAR PUSTAKA

Agfianto Eko Putra, **Belajar Mikrokontroller AT 89C51/52/55**, Gava Media
Yogyakarta, 2002

Budioko, Totok, **Belajar Dengan Mudah dan Cepat Pemrograman Bahasa C
Dengan SDCC (Small Device C Compiler) Pada Mikrokontroler
AT89X051/AT89C51/52 Teori, Simulasi dan Aplikasi**, Gava Media,
Yogyakarta, 2005

Kernighan, Brian W, Dennis M. Ritchie, **The C Programming Language,
Second Edition**, Prentice Hall of India Private Limited, New Delhi, 1989

<http://www.renesas.com>

<http://www.winbond.com>

Wasito S, **Vandemekum Elektronika, Edisi Kedua**, PT. Gramedia Pustaka
Utama, Jakarta, 1990

LAMPIRAN



FORMULIR BIMBINGAN SKRIPSI

Nama : Tjia Sindarto Setiadi
 NIM : 02.17.059
 Masa Bimbingan : 5 Oktober 2006 - 5 April 2007
 Judul Skripsi : Perancangan dan Pembuatan Pemindah Jalur Rel Kereta Api Semi Otomatis Di Stasiun Menggunakan Mikrokontroler R5F21134FP

NO	Tanggal	Uraian	Paraf Pembimbing
1	24/9/06	Menerjimmahkan Permasalahannya, menentukan judul.	
2	12/10/06	Identifikasi & inventarisasi masalah melalui blok diagram.	
3	4/11/06	Dari PCB ops. di buat lembar pengurutan, letas di Bab IV.	
4	22/11/06	Dari Bab V, hasil uji coba tulis di Bab V secara kuantitatif.	
5	04/12/06	Bab III, sesuaikan dgn masalah otomatisasi sirkuit.	
6	12/12/06	Dari Bab III, tinjau & Amati kemampuan 2 jenis ke ops berbeda.	
7	20/12/06	Bab II, lengkapi teori tentang motor & penggerak (AC/DC).	
8	1/1/07	Bab I: tinjau ulang tgl permasalahan & penyelesaiannya.	
9	10/3/07	Ambil hasil pengujian (Bab V) di akhir abstrak.	
10	14/3/07	Siapkan ujian komprehensif.	

Malang,
 Dosen Pembimbing

Ir. Sidik Noertjahiono, MT
 NIP. 1028700163



FORMULIR PERBAIKAN SKRIPSI

Dari hasil ujian komprehensif jenjang Strata Satu (S-1) Jurusan Teknik Elektro, Konsentrasi Elektronika yang diselenggarakan pada:

Hari : Sabtu
Tanggal : 17 Maret 2007

Telah dilakukan perbaikan skripsi oleh :

Nama : Tjia Sindarto Setiadi
N.I.M : 0217059

Masa Bimbingan : 5 Oktober 2006 – 5 April 2007

Judul : Perancangan dan Pembuatan Pemindah Jalur Rel Kereta
Api Semi Otomatis Di Stasiun Menggunakan
Mikrokontroler R5F21134FP

Perbaikan Meliputi :

No	Materi Perbaikan	Paraf Dosen Penguji
1.	Tambahkan RS-232 dalam komunikasi mikrokontroler dengan sensor/peralatan yang letaknya jauh	
2.	Buat bahan khusus tentang komunikasi dan kerja mikrokontroler secara paralel (3 mikrokontroler)	

Disetujui Oleh:
Penguji Pertama

(Dr. Cahyo Crysdian, MSc.)

Mengetahui,
Dosen Pembimbing

Ir. Sidik Noertjahjono, MT
NIP. 1028700167



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi


Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : NIA SINARU S.
NIM : 0217059
Perbaikan meliputi

1) ~~Tan~~ bahan RS-232 di komunikasi ke
yg antar / yg pada ketanya jauh
peralatan

2) ~~Bat~~ bahan litrus Hz komunikasi
~~nya~~ ke secara paralel.
(3 ke)

Malang, 17
3 27.


(Dr) CAHYO KUSDIANJOSE



FORMULIR PERBAIKAN SKRIPSI

Dari hasil ujian komprehensif jenjang Strata Satu (S-1) Jurusan Teknik Elektro, Konsentrasi Elektronika yang diselenggarakan pada:

Hari : Sabtu
Tanggal : 17 Maret 2007
Telah dilakukan perbaikan skripsi oleh :
Nama : Tjia Sindarto Setiadi
N.I.M : 0217059
Masa Bimbingan : 5 Oktober 2006 – 5 April 2007
Judul : Perancangan dan Pembuatan Pemindah Jalur Rel Kereta
Api Semi Otomatis Di Stasiun Menggunakan
Mikrokontroler R5F21134FP

Perbaikan Meliputi :

No	Materi Perbaikan	Paraf Dosen Penguji
1.	Hal. 72. Istilah sensor diganti detektor	

Disetujui Oleh:
Penguji Kedua

(Ir. Komang Somawirata, ST., MT.)
NIP.1030100361

Mengetahui,
Dosen Pembimbing

Ir. Sidik Noertjahjono, MT
NIP. 1028700167



INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Janjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : DIA SINDHARTO

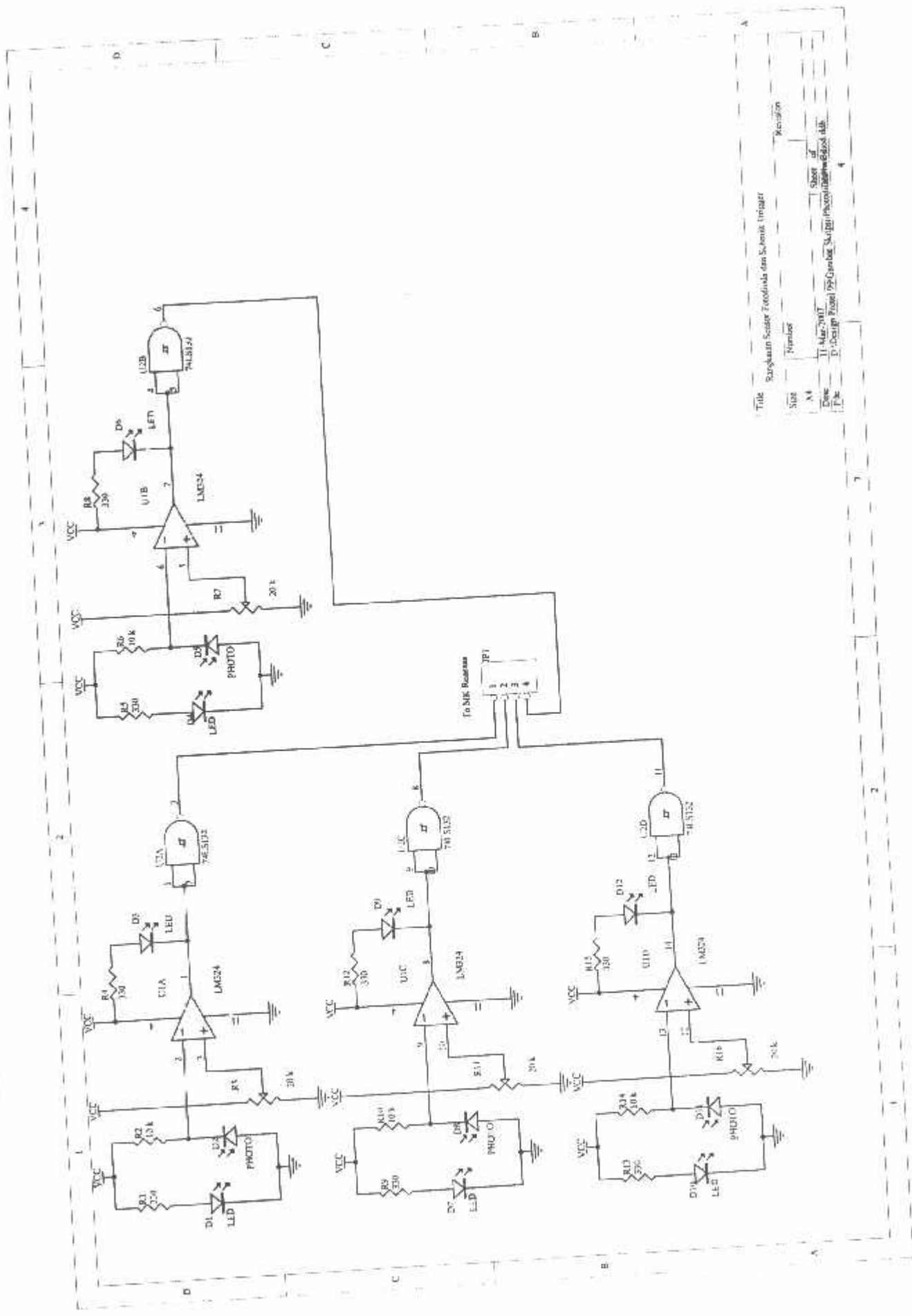
NIM : 0217019

Perbaikan meliputi

4 hal 72. / uraian sensor → pefektif

Malang,


(DIA SINDHARTO ST, 2019)



Tipe : Barisan Scade Tridimensi dan Seluruh Integrator

No	Number	Keterangan
1	1	Sheet of
2	1	Sheet of
3	1	Sheet of
4	1	Sheet of

Uraian : 11 Mar 2007
 Nama : P. Desai Prati 192 Garuda Sud Pr Micro Elektronika Lab


```

*****
/*****
*
*
*
*****

#include <stdio.h>
#include "sfr_r813.h"

#define aa p0_0 // Port 0.0
#define ta2 p0_1
#define ab p0_2
#define tb2 p0_3
#define let1 p0_4
#define let2 p0_5
#define let3 p0_6
#define let4 p0_7 // Port 0.7
#define lt1 p1_0 // Port 1.0
#define lt2 p1_1
#define lt3 p1_2
#define lt4 p1_3
#define ir1 p1_4
#define ir2 p1_5
#define ir3 p1_6 // Port 1.7
#define ir4 p1_7
#define buffer p3_0
#define danan p3_3
#define dari p4_5

void delay(char z)
{
    while (z-->0);
}

void delay1()
{
    char i,j; for(i=0;i<5;i++);
    for(j=0;j<500000;j++);
}

void checkawal()
{
    delay(500000);
    if (ir1 = 0)
        {let1 = 0;}
    if (ir2 = 0)
        {let2 = 0;}
    if (ir3 = 0)
        {let3 = 0;}
    if (ir4 = 0)
        {let4 = 0;}
    if (ir1 = 1)
        {let1 = 1;}
    if (ir2 = 1)
        {let2 = 1;}
    if (ir3 = 1)
        {let3 = 1;}
    if (ir4 = 1)
        {let4 = 1;}
}

void check1im()
{
    unsigned int add = 0;
    while ( add )
    {
        if (lt1 = 0)

```



```

                                Master
{
    delay(500000);
    let1 = 1;add++;
}
if (1t2 = 0)
{
    delay(500000);
    let2 = 1;add++;
}
if (1t3 = 0)
{
    delay(500000);
    let3 = 1;add++;
}
if (1t4 = 0)
{
    delay(500000);
    let4 = 1;add++;
}
}
danan = 1;
dari = 1;
buffer = 1;
delay(500000);
buffer = 0;
}

/*****
*      Function : main()
*      program section
*****/
void main()
{
    asm("FCLR I");
    prcr = 0;
    cm13 = 1;
    cm15 = 1;
    cm05 = 0;
    cm16 = 0;
    cm17 = 1;
    cm06 = 0;
    asm("nop");
    asm("nop");
    asm("nop");
    asm("nop");
    ocd2 = 1;
    prcr = 0;

    prc2 = 1;
    pd0 = 0xf0;
    prc2 = 0;
    pd1 = 0x00;
    pd3 = 0xff;
    p3 = 0x08;
    pd4_5 = 1;
    p4_5 = 1;
    checkwai();

    while(1)
    {
        if (aa = 0)
        {
            p3_1 = 0;
            p3_2 = 1;
            buffer = 1;
            danan = 0;
            delay(5000);
            buffer = 0;
        }
    }
}

```

```

        Master
        checklim();
    }
    if (ab = 0)
    {
        p3_1 = 1;
        p3_2 = 0;
        buffer = 1;
        dari = 0;
        delay(5000);
        buffer = 0;
        checklim();
    }
    if(ta2 = 0)
    {
        while(aa);
        checkawal();
        delay1();
    }
    if(tb2 = 0)
    {
        while(ab);
        checkawal();
        delay1();
    }
}
}
}

```

Slave LCD

```
//=====
// Tjia Sindarto Setiadi
// Institut Teknologi Nasional Malang
//=====
```

```
#include <AT89x51.h>
#include <LCD.h>
#include <Buzzer.h>
#include <ISD.h>

#define ta1 P2_2
#define tb1 P2_3
#define jalur1 P3_4
#define jalur2 P3_5
#define jalur3 P3_6
#define jalur4 P3_7
#define datach P1_5

unsigned int baru;

void tambah(char waktu)
{
    while(waktu-->0);
}

void delaylong()
{
    unsigned int i=0;
    while(i<8)
    {
        tambah(35000);
        i++;
    }
}

void datang()
{
    clear();
    TulisLCD(0x80," Selamat Datang ");
    TulisLCD(0xc0," Stasiun KA ");
}

void hati1()
{
    clear();
    TulisLCD(0x80," HATI-HATI ");
    TulisLCD(0xc0,"KA masuk jalur 1");
}

void hati2()
{
    clear();
    TulisLCD(0x80," HATI-HATI ");
    TulisLCD(0xc0,"KA masuk jalur 2");
}

void hati3()
{
    clear();
    TulisLCD(0x80," HATI-HATI ");
    TulisLCD(0xc0,"KA masuk jalur 3");
}
}
```

Slave LCD

```

void hati4()
{
    clear();
    TulisLCD(0x80," HATI-HATI ");
    TulisLCD(0xC0,"KA masuk jalur 4");
}

void mulaikan()
{
    unsigned char a = 0;
    while ( a )
    {
        if( datach = 1 )
        {
            tunda(2000);
            baru = P1;
            baru &= 0xC0;
            while(tal);
            if( baru = 0x00 ) //jalur1
            {
                jalur1 = 0;
                hati1();
                say1();
                a++;
            }
            if( baru = 0x80 ) //jalur2
            {
                jalur2 = 0;
                hati2();
                say2();
                a++;
            }
            if( baru = 0x40 ) //jalur3
            {
                jalur3 = 0;
                hati3();
                say3();
                a++;
            }
            if( baru = 0xC0 ) //jalur4
            {
                jalur4 = 0;
                hati4();
                say4();
                a++;
            }
        }
    }
}

void mulaikir()
{
    unsigned char a = 0;
    while ( a )
    {
        if( datach = 1 )
        {
            tunda(2000);
            baru = P1;
            baru &= 0xC0;
            while(tal);
            if( baru = 0x00 ) //jalur1
            {
                jalur1 = 0;
            }
        }
    }
}

```

```

        slave LCD
        hati1():
        say1();
        a++;
    }
    if( baru = 0x80 ) //jalur2
    {
        jalur2 = 0;
        hati2();
        say2();
        a++;
    }
    if( baru = 0x40 ) //jalur3
    {
        jalur3 = 0;
        hati3();
        say3();
        a++;
    }
    if( baru = 0xc0 ) //jalur4
    {
        jalur4 = 0;
        hati4();
        say4();
        a++;
    }
}

void kanan()
{
    buzzer(3);
    while(ta1);
    mulaikan();
    while(!P1_0);
    P3 = 0xf0;
    datang();
}

void kiri()
{
    buzzer(3);
    while(tb1);
    mulaikir();
    while(!P1_0);
    P3 = 0xf0;
    datang();
}

void main()
{
    P0 = 0x00;
    P1 = 0x00;
    P2 = 0xff;
    P3 = 0x00;
    initLCD();
    TulisLCD(0x80, "Tjia Sindarto S.");
    TulisLCD(0xc0, "02.17.059");
    delay2():delay2();
    datang();
    while(1)
    {
        if(P1_0 = 1)

```

```
Slave LCD
testdata = P1;
testdata &= 0x06;
if( testdata = 0x04 )
{
    kanan();
}
if( testdata = 0x02 )
{
    kiri();
}
```

```
}
```

```
}
```

```
}
```

```
=====
Tjia Sindarto Setiadi
*****
Institut Teknologi Nasional Malang
=====
```

```
#include <AT89x51.h>
#include <stdio.h>
#include <Tunggu.h>
```

```
#define LB11 P0_2
#define LB12 P0_3
#define LB21 P0_6
#define LB22 P0_7
#define LB31 P2_5
#define LB32 P2_4
#define LA11 P2_1
#define LA12 P2_0
#define LA21 P3_2
#define LA22 P3_3
#define LA31 P3_6
#define LA32 P3_7
#define data1 P1_0
#define data2 P1_1
#define data3 P1_2
```

```
void BelokMA1()
```

```
    P2_2 = 0;
    P2_3 = 1;
    while(LA12);
    P2_3 = 0;
```

```
void LurusMA1()
```

```
    P2_3 = 0;
    P2_2 = 1;
    while(LA11);
    P2_2 = 0;
```

```
void LurusMA2()
```

```
    P3_1 = 0;
    P3_0 = 1;
    while(LA21);
    P3_0 = 0;
```

```
void BelokMA2()
```

```
    P3_0 = 0;
    P3_1 = 1;
    while(LA22);
    P3_1 = 0;
```

```
void LurusMA3()
```

```
    P3_5 = 0;
    P3_4 = 1;
    while(LA32);
    P3_4 = 0;
```

```
void BelokMA3()
```

Motor1

```
P3_4 = 0;
P3_5 = 1;
while(LA31);
P3_5 = 0;
```

void BelokMB1()

```
P0_1 = 0;
P0_0 = 1;
while(LB12);
P0_0 = 0;
```

void LurusMB1()

```
P0_0 = 0;
P0_1 = 1;
while(LB11);
P0_1 = 0;
```

void BelokMB2()

```
P0_5 = 0;
P0_4 = 1;
while(LB22);
P0_4 = 0;
```

void LurusMB2()

```
P0_4 = 0;
P0_5 = 1;
while(LB21);
P0_5 = 0;
```

void LurusMB3()

```
P2_7 = 0;
P2_6 = 1;
while(LB32);
P2_6 = 0;
```

void BelokMB3()

```
P2_6 = 0;
P2_7 = 1;
while(LB31);
P2_7 = 0;
```

Program Utama
void main()

```
data1 = 0;
data2 = 0;
data3 = 0;
P0 = 0xCC;
P2 = 0x33;
P3 = 0xCC;
while(1)
```

//Untuk menonaktifkan semua

```
{
    if ( P1_3 == 1 ) // Tunggu input keypad
    {
        unsigned int a; // Test keypad
```



```

Motor1
a = P1;
a &= 0xf0;
if ( a == 0x00 ) //jalur 4A
{
    data2 = 1;
    data3 = 1;
    delay1s();
    data1 = 1;
    BelokMA1();
    LurusMA3();
    delay1m();
    data1 = 0;
}
if ( a == 0x40 ) //jalur 3A
{
    data2 = 1;
    data3 = 0;
    delay1s();
    data1 = 1;
    BelokMA1();
    BelokMA3();
    delay1m();
    data1 = 0;
}
if ( a == 0x80 ) //jalur 2A
{
    data2 = 0;
    data3 = 1;
    delay1s();
    data1 = 1;
    LurusMA1();
    LurusMA2();
    delay1m();
    data1 = 0;
}
if ( a == 0xc0 ) //jalur 1A
{
    data2 = 0;
    data3 = 0;
    delay1s();
    data1 = 1;
    LurusMA1();
    BelokMA2();
    delay1m();
    data1 = 0;
}
if ( a == 0x10 ) //jalur 4B
{
    data2 = 1;
    data3 = 1;
    delay1s();
    data1 = 1;
    BelokMB1();
    LurusMB3();
    delay1m();
    data1 = 0;
}
if ( a == 0x50 ) //jalur 3B
{
    data2 = 1;
    data3 = 0;
    delay1s();
    data1 = 1;
    BelokMB1();
    BelokMB3();
    delay1m();
    data1 = 0;
}
}

```

```

        Motor1
if ( a == 0x90 ) //jalur 2B
{
    data2 = 0;
    data3 = 1;
    delay1s();
    data1 = 1;
    LurusMB1();
    LurusMB2();
    delay1m();
    data1 = 0;
}
if ( a == 0xD0 ) //jalur 1B
{
    data2 = 0;
    data3 = 0;
    delay1s();
    data1 = 1;
    LurusMB1();
    BelokMB2();
    delay1m();
    data1 = 0;
}
}
}

```

20 V PNP general purpose transistors

PSS9012 series

FEATURES

- High power dissipation: 710 mW
- Low collector capacitance
- Low collector-emitter saturation voltage
- High current capability.

APPLICATIONS

- General purpose switching and amplification.

DESCRIPTION

PNP general purpose transistor in a SOT54 (TO-92) leaded plastic package. NPN complement: PSS9013 series.

MARKING

TYPE NUMBER	MARKING CODE
PSS9012G	S9012G
PSS9012H	S9012H

QUICK REFERENCE DATA

SYMBOL	PARAMETER	MAX.	UNIT
V_{CE0}	collector-emitter voltage	-20	V
I_C	collector current (DC)	-500	mA
I_{CM}	peak collector current	-1	A

PINNING

PIN	DESCRIPTION
1	collector
2	base
3	emitter

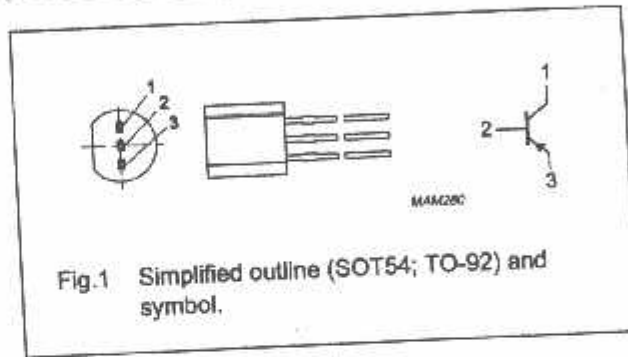


Fig.1 Simplified outline (SOT54; TO-92) and symbol.

LIMITING VALUES

In accordance with the Absolute Maximum System (IEC 60134).

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{CB0}	collector-base voltage	open emitter	-	-40	V
V_{CE0}	collector-emitter voltage	open base	-	-20	V
V_{EB0}	emitter-base voltage	open collector	-	-5	V
I_C	collector current (DC)		-	-500	mA
I_{CM}	peak collector current		-	-1	A
I_{BM}	peak base current		-	-100	mA
P_{tot}	total power dissipation	$T_{amb} \leq 25\text{ }^\circ\text{C}$; note 1	-	710	mW
T_{stg}	storage temperature		-65	+150	$^\circ\text{C}$
T_j	junction temperature		-	150	$^\circ\text{C}$
T_{amb}	operating ambient temperature		-65	+150	$^\circ\text{C}$

Note

1. Device mounted on a FR4 printed-circuit board, single-sided copper, tinplated and standard footprint.

20 V PNP general purpose transistors

PSS9012 series

THERMAL CHARACTERISTICS

SYMBOL	PARAMETER	CONDITIONS	VALUE	UNIT
$R_{th(j-a)}$	thermal resistance from junction to ambient	in free air; note 1	175	K/W

Note

1. Device mounted on a FR4 printed-circuit board, single-sided copper, tinplated and standard footprint.

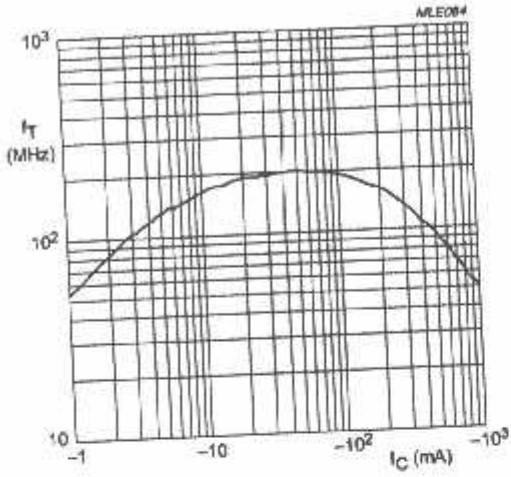
CHARACTERISTICS

$T_{amb} = 25\text{ }^{\circ}\text{C}$ unless otherwise specified.

SYMBOL	PARAMETER	CONDITIONS	MIN.	TYP.	MAX.	UNIT
I_{CBO}	collector-base cut-off current	$V_{CB} = -35\text{ V}; I_E = 0$	-	-	-100	nA
		$V_{CB} = -35\text{ V}; I_E = 0; T_J = 150\text{ }^{\circ}\text{C}$	-	-	-50	μA
I_{EBO}	emitter-base cut-off current	$V_{EB} = -5\text{ V}; I_C = 0$	-	-	-100	nA
h_{FE}	DC current gain	$V_{CE} = -1\text{ V}; I_C = -500\text{ mA}$	40	-	-	
h_{FE}	DC current gain PSS9012G PSS9012H	$V_{CE} = -1\text{ V}; I_C = -50\text{ mA}$	112	-	166	
			144	-	202	
V_{CEsat}	collector-emitter saturation voltage	$I_C = -100\text{ mA}; I_B = -10\text{ mA}$	-	-60	-250	mV
		$I_C = -500\text{ mA}; I_B = -50\text{ mA}$	-	-230	-600	mV
V_{BEsat}	base-emitter saturation voltage	$I_C = -500\text{ mA}; I_B = -50\text{ mA}$	-	-1	-1.2	V
V_{BEon}	base-emitter turn on voltage	$V_{CE} = -1\text{ V}; I_C = -100\text{ mA}$	-	-760	-1000	mV
C_c	collector capacitance	$V_{CB} = -6\text{ V}; I_E = I_e = 0;$ $f = 1\text{ MHz}$	-	6	-	pF

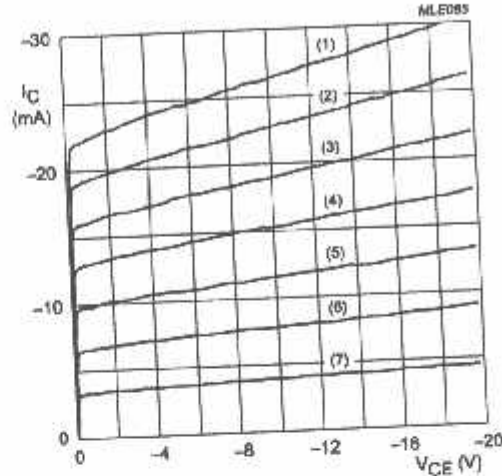
20 V PNP general purpose transistors

PSS9012 series



$V_{CE} = -6 V$.

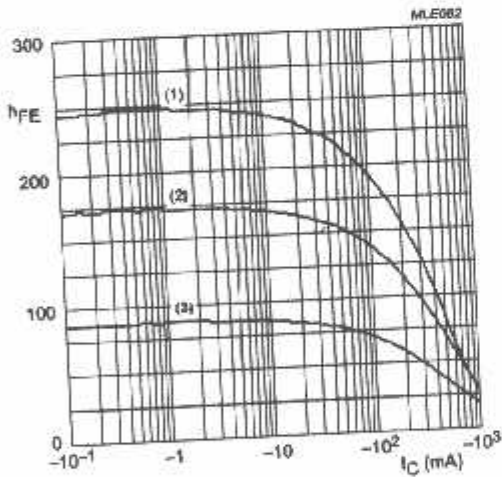
Fig.2 Transition frequency as a function of collector current; typical values.



$T_{amb} = 25 ^\circ C$.

- (1) $I_B = -140 \mu A$, (4) $I_B = -80 \mu A$, (7) $I_B = -20 \mu A$.
- (2) $I_B = -120 \mu A$, (5) $I_B = -60 \mu A$.
- (3) $I_B = -100 \mu A$, (6) $I_B = -40 \mu A$.

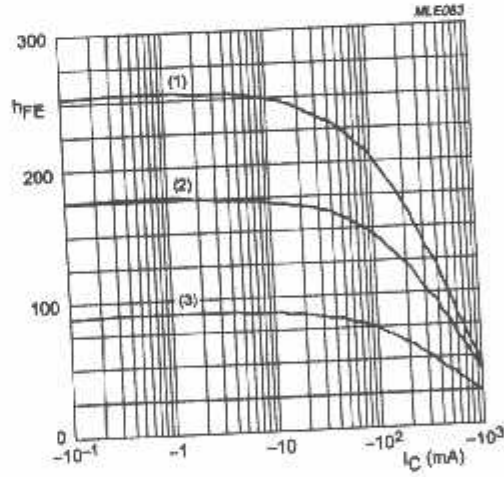
Fig.3 Collector current as a function of collector-emitter voltage; typical values.



$V_{CE} = -1 V$.

- (1) $T_{amb} = 100 ^\circ C$.
- (2) $T_{amb} = 25 ^\circ C$.
- (3) $T_{amb} = -55 ^\circ C$.

Fig.4 DC current gain as a function of collector current; typical values.



$V_{CE} = -2 V$.

- (1) $T_{amb} = 100 ^\circ C$.
- (2) $T_{amb} = 25 ^\circ C$.
- (3) $T_{amb} = -55 ^\circ C$.

Fig.5 DC current gain as a function of collector current; typical values.

20 V PNP general purpose transistors

PSS9012 series

