

TUGAS AKHIR

PERANCANGAN DAN PEMBUATAN KENDALI PERALATAN LISTRIK TANPA KABEL DENGAN MK AT89S51



Disusun Oleh :

Nama : Hidayat Surya Dilaga

NIM : 02.52. 011

**KONSENTRASI TEKNIK ENERGI LISTRIK
JURUSAN TEKNIK ELEKTRO D-III
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
Agustus 2007**

TIKAS AKHIL

MATAKIPER LIAGUEN MATAKIPER MADI MADMADAMATI
LESTERI TANPA LABEL BERGAM MKA AT30001



: 0000000000

Nama : Hidayat Saqqa Saqqa

NO. 00.00.011

KELOMPOK TEKNIK ELEKTRO
M-01 ORIENTASI KEHIMPUNAN
FASILITAS TEKNIK DAN KEMAHIRAN
DINAMIA JAWABAN MOCAMMOT TUTTUMU
TGGG: 0000000000

**LEMBAR PERSETUJUAN
TUGAS AKHIR**

**PERANCANGAN KENDALI PERALATAN LISTRIK TANPA
KABEL DENGAN MK AT89S51**

Disusun Oleh:

NAMA : Hidayat Surya.D

NIM : 0252011



**Mengetahui
Ketua Jurusan Teknik Elektro D III**

**(Ir. Choirul Saleh, MT)
NIP.Y : 1018800190**

**Diperiksa dan disetujui
Dosen Pembimbing**

**(Bambang Priyo H, ST.MT)
NIP.Y : 1028400082**

**KONSENTRASI ENERGI LISTRIK
JURUSAN TEKNIK ELEKTRO D-III
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
September 2007**

ABSTRAK

PERANCANGAN DAN PEMBUATAN KENDALI PERALATAN LISTRIK TANPA KABEL DENGAN MK AT89S51

(Hidayat Surya Dilaga, 02.52.011, Teknik Elektro D-III/Energi Listrik)
(Dosen Pembimbing : Bambang Prio H.ST,MT)

Kata kunci : AT89S51, LCD M1632, TRW 2,4G, Diver Relay

Penggunaan Mikrokontroler AT89S51 sebagai unit-unit kendali sudah sangat luas, hal ini dikarenakan peralatan-peralatan yang terkontrol dengan mikrokontroler memberikan kemudahan-kemudahan dalam penggunaannya. Seperti halnya alat kontrol instalasi rumah tangga dengan menggunakan remot control yang sudah ada, dan sekarang dikembangkan lagi dengan menggunakan TRW 2,4G yang dilakukan secara digital dengan tampilan pada Modul LCD M1632. Dengan pemakaian TRW 2,4G untuk mempermudah pemilik atau penghuni rumah untuk mengendalikan peralatan listrik secara otomatis pada beban listrik dari suatu tempat yang dikontrol, sehingga akan lebih mudah dan praktis.

Perencanaan dan pembahasan alat yang dilakukan pada tugas akhir ini adalah bersifat praktis yaitu perencanaan dan pembuatan alat yang mengarah pada pembuatan alat secara nyata. Awal pembahasan ini dimulai dengan membahas masing-masing rangkaian tersebut, kemudian diikuti dengan perencanaan dan pembuatan perangkat lunak maupun alat. Proses pengaturannya adalah pada TRW 2,4G mengirimkan data sesuai dengan tombol yang ditekan. Data yang dikirim adalah data serial dengan format tertentu dan dimodulasi dengan sinyal *carrier* 40 KHz. Data termodulasi ini kemudian di-filter oleh modul IR8510 sehingga didapatkan data asli. Data tersebut kemudian dibaca oleh Mikrokontroler AT89S51 dan dikonversikan menjadi data biner sebagai perintah untuk seting timer peralatan listrik. *driver relay* agar menyalakan maupun mematikan peralatan listrik yang dikontrol.

Jarak yang dapat dicapai maksimal alat ini adalah 11 meter dengan tidak ada rintangan, dan jarak yang dapat di capai jika ada rintangan adalah 9 meter. Alat hanya bisa bekerja dengan daya maksimal 450w. Alat ini dapat bekerja pada frekuensi 2,4GHz

KATA PENGANTAR

Puji syukur atas Karunia Tuhan Yang Maha Kuasa, sehingga saya dapat menyelesaikan Tugas Akhir yang berjudul “ **KENDALI PERALATAN LISTRIK TANPA KABEL DENGAN MK AT89S51** ” ini dengan lancar. Tugas Akhir ini dikerjakan sebagai salah satu syarat untuk menyelesaikan studi di Jurusan Teknik Teknik Elektro D-III/Energi Listrik Institut Teknologi Nasional Malang.

Keberhasilan penyelesaian laporan Tugas Akhir ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penulis menyampaikan ucapan terima kasih kepada :

1. Bapak Ir. H. Choirul Saleh, MT selaku Ketua Jurusan Teknik Elektro D-III.
2. Bapak Bambang Prio H.ST,MT selaku Dosen Pembimbing
3. Bapak Bambang Prio H.ST,MT selaku Sekretaris Jurusan Teknik Elektro D-III.
4. Buat semua Dosen – Dosen di jurusan Teknik Elektro D-III/Energi Listrik
Terima kasih atas ilmu yang diberikan kepada saya sehingga bisa menjadi manusia yang berguna.
5. Keluarga (ayah, mama dan adik) yang telah banyak memberikan do’a, dorongan semangat, dan biaya baik secara moril maupun materiil.
6. Teman – Teman semua, Teknik Elektro D-III/Energi Listrik terimakasih atas semua bantuanya.

7. Teman – Teman Kost semua di GREEN HOUSE, atas bantuan dan doanya semua.

Meskipun telah dikaji ulang dan dikerjakan dengan sungguh-sungguh, namun penulis menyadari bahwa Tugas Akhir ini masih jauh dari sempurna, karena keterbatasan pengetahuan dan keterampilan yang penulis miliki, sehingga segala kritik dan saran penulis terima untuk dijadikan pedoman didalam menyusun laporan berikutnya yang lebih sempurna.

Akhir kata semoga Tugas Akhir ini dapat memberikan manfaat bagi seluruh masyarakat ilmiah pada khususnya dan pembaca pada umumnya.

Malang, September 2007

Penyusun
(Hidayat Surya Dilaga)

DAFTAR ISI

Lembar Persetujuan	i
Abstraksi.....	ii
Kata pengantar.....	iii
Daftar isi	v
Daftar Gambar	viii
Daftar Tabel	ix
BAB I. PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	2
1.3. Tujuan	2
1.4. Batasan Masalah	2
1.5. Metodologi Penelitian.....	3
1.6. Sistematika Penulisan	3
BAB II. TEORI DASAR.....	5
2.1. Mikrokontroler AT89S51	5
2.1.1. Teori umum.....	5
2.1.2. Arsitektur AT89S51	6
2.1.3. Konfigurasi Pin pada Mikrokontroler AT89S51	7
2.1.4. Karakteristik <i>Oscillator Inverting</i>	10
2.1.5. Organisasi <i>Memory</i>	11
2.1.5.1. Program <i>Memory Internal</i>	11
2.1.5.2. Data Memory (RAM) <i>Internal</i>	11
2.1.5.3 SFR (<i>Special Function Register</i>)	13
2.2. Modul TRW 2,4 G	13
2.3. Dioda.....	15
2.3.1. Dioda Penyearah.....	16
2.4. Kapasitor.....	16
2.4.1. Cara Kerja Kapasitor.....	18

2.5. Resistor	19
2.6. LCD (<i>Liquid Crystal Display</i>) M1632.....	22
2.6.1. Sinyal <i>interface</i> M1632.....	23
2.6.2. <i>Interface</i> Ke MCS-51	24
2.6.3. Mengatur tampilan M1632	28
2.7. <i>Relay</i>	33
BAB III. PERENCANAAN DAN PEMBUATAN ALAT	36
3.1. Diagram Blok.....	36
3.2. Prinsip Kerja Alat	38
3.3. Mikrokontroler AT89S51	38
3.3.1. Pemetaan <i>Memori</i>	38
3.3.2. Rangkaian <i>Clock</i>	39
3.3.3. Rangkaian <i>Reset</i>	40
3.3.4. Hubungan <i>Pin</i> Pada AT89S51 (Rangkaian <i>Transmitter</i>).....	41
3.3.5. Hubungan <i>Pin</i> Pada AT89S51 (Rangkaian <i>Receiver</i>).....	42
3.4. Perancangan Rangkaian TRW 2,4G	43
3.5. Perancangan Rangkaian <i>LCD</i>	45
3.6. Perancangan Rangkaian <i>Driver Relay</i>	49
BAB IV. PENGUJIAN ALAT	53
4.1. Pengujian Terhadap Rangkaian <i>Power Supply</i>	53
4.2. Pengujian Terhadap Rangkaian TRW 2,4G.....	56
4.3. Pengujian Terhadap Rangkaian <i>Push Buttom</i>	57
4.4. Pengujian Terhadap Rangkaian <i>Driver Relay</i>	59
BAB V. PENUTUP.....	62
5.1. Kesimpulan	62
5.2. Saran	63

Daftar Pustaka	64
Lampiran	65

DAFTAR GAMBAR

Gambar 2-1. Diagram Blok AT89S51	5
Gambar 2-2. Konfigurasi Pin-Pin AT89S51	7
Gambar 2-3. Karakteristik Oscilator	10
Gambar 2-4. Organisasi RAM <i>Internal</i>	12
Gambar 2-5. Modul TRW 2,4G	15
Gambar 2-6. Simbol Dioda	16
Gambar 2-7. Simbol Kapasitor	17
Gambar 2-8. Rangkaian Pengisian Kapasitor	18
Gambar 2-9. Grafik Tegangan dan Arus Pengisian Kapasitor	19
Gambar 2-10. Mengirim / Mengambil Data Ke / Dari M1632	23
Gambar 2-11. Hubungan M1632 Ke MCS'51	24
Gambar 2-12. Rangkaian LCD M1632	30
Gambar 2-13. Gambar Konstruksi <i>Relay</i>	33
Gambar 2-14. Sebuah <i>Relay</i> Praktis dari <i>Siemens</i>	34
Gambar 2-15. Kemungkinan – Kemungkinan Kontak dari <i>Relay</i>	35
Gambar 3-1. Blok Diagram	36
Gambar 3-2. Rangkaian <i>Clock</i> AT89S51	40
Gambar 3-3. Rangkaian <i>Reset</i>	41
Gambar 3-4. Hubungan <i>Pin</i> Pada AT89S51 (Pada Rangkaian <i>Transmitter</i>)	42
Gambar 3-5. Hubungan <i>Pin</i> Pada AT89S51 (Pada Rangkaian <i>Receiver</i>)	43
Gambar 3-6. Rangkaian <i>Level Converter</i>	45
Gambar 3-7. Rangkaian <i>LCD</i> Pada Mikrokontroler	48
Gambar 3-8. Rangkaian <i>Driver Relay</i>	50
Gambar 4-1. Cara Pengukuran Tegangan Pada Catu Daya	54
Gambar 4-2. Diagram Blok Pengujian TRW 2,4G	57
Gambar 4-3. Cara Pengukuran Tegangan Pada Pengujian <i>Push Button</i>	58
Gambar 4-4. Pengujian Rangkaian <i>Driver Relay</i>	60
Gambar 4-5. Foto Alat Tampak Atas	61
Gambar 4-6. Foto Alat Tampak Dalam	61

DAFTAR TABEL

Tabel 2-1. Pengaturan RS0-RS1 Untuk Select Register Bank.....	12
Tabel 2-2. 128 Byte Special Function Register.....	13
Tabel 2-3. Gelang Warna Resistor	20
Tabel 2-4. Fungsi Pin – Pin LCD	31
Tabel 4-1. Hasil Pengukuran Rangkaian Power Supply	55
Tabel 4-2. Hasil Pengujian Pada Rangkaian Push Buttom.....	59
Tabel 4-3. Hasil Pengujian Pada Rangkaian Driver Relay	60

BAB I

PENDAHULUAN

1.1. Latar Belakang

Di zaman perkembangan teknologi seperti sekarang ini, dimungkinkan sekali untuk mendesain suatu alat yang dapat digunakan untuk mengendalikan peralatan listrik, baik yang terdapat di rumah, perkantoran, ruko, mal, maupun apartemen dengan sebuah mikrokontroler sebagai pusat pengendali.

Sebagai contoh adalah sistem operator peralatan listrik pada sebuah gedung bertingkat. Pada sebuah gedung bertingkat, agar kita tidak harus naik dan turun tangga hanya untuk mematikan sebuah lampu ataupun ingin mengaktifkan peralatan listrik yang lain, maka dibutuhkan suatu sistem operator peralatan listrik dalam gedung secara terpusat (Budianto, 2005).

Dengan menggunakan sistem operator peralatan listrik secara terpusat, maka kita akan mendapatkan kemudahan dalam mengontrol keseluruhan peralatan listrik dalam gedung tersebut secara terpusat dalam sebuah ruang kontrol. Di satu sisi sistem tersebut pada kenyataannya memang memberikan kemudahan-kemudahan, tetapi disatu sisi yang lain dapat dipastikan jumlah kabel yang dibutuhkan akan menjadi sangat banyak (panjang) sehingga kurang efisien dan dapat mengganggu pemandangan dalam gedung tersebut.

Dan alat ini bisa diaplikasikan di peralatan listrik untuk menghidupkan atau mematikan peralatan listrik seperti lampu, dan peralatan listrik rumah tangga dengan kapasitas maksimal daya 450w, karna disini kita menggunakan relay dengan kapasitas 2 Ampere. Dan alat berkerja memanfaatkan gelombang

frekuensi sebagai sarana penghubung antara trasmitter dengan receiver. Frekuensi yang digunakan adalah 2,4GHz.

1.2. Rumusan Masalah

Dari uraian tersebut diatas maka timbul beberapa permasalahan diantaranya adalah:

1. Berapa jarak yang dapat di tempuh rangkaian trasmitter kerangkain receiver dengan ada rintangan maupun tidak
2. Berapa daya maksimal yang dapat di hasilkan
3. Bagaimana proses komunikasi data tanpa kabel (*wireless*) ?

1.3. TUJUAN

merancang peralatan listrik secara terpusat menggunakan teknologi wireless.

1.4. BATASAN MASALAH

Agar permasalahan tidak meluas maka penulis membatasi permasalahan sebagai berikut:

1. Menggunakan Mikrokontroler sebagai pengendali utama..
2. Menggunakan komunikasi data melalui RF (Radio Frekuensi).
3. Menggunakan *Relay* sebagai penghubung peralatan listrik dengan tegangan jala-jala PLN AC220V.
4. Menggunakan LCD sebagai display kerja alat.
5. Alat ini bisa dipergunakan hanya untuk lima sumber tegangan. Dengan daya maksimal 450w.

1.5. Metodologi Penulisan

Metodologi yang dilakukan dalam penulisan tugas akhir Aplikasi Mikrokontroler Sebagai Pusat Kendali Peralatan Listrik Tanpa Kabel (*wireless*) adalah sebagai berikut:

1. Kajian pustaka, meliputi konsep dasar dan teori yang digunakan
2. Perancangan dan pembuatan aplikasi meliputi perancangan model, proses pembuatan disertai diagram alirnya
3. Pengujian tentang aplikasi yang sedang dibuat
4. Membuat kesimpulan serta saran dari hasil pengujian aplikasi

1.6. SISTEMATIKA PENULISAN

Untuk mempermudah dan memperjelas pembahasan dari laporan tugas akhir ini penulis menerapkan system penulisan seperti dibawah ini :

BAB I : PENDAHULUAN

Berisi latar belakang permasalahan, rumusan masalah, tujuan pembahasan, batasan masalah, metodologi penulisan dan sistematika penulisan.

BAB II : DASAR TEORI

Membahas tentang teori dasar rangkaian yang digunakan, mikrokontroler AT89S51, TRW-24G, Relay, dan teori dasar alat-alat pendukung lainnya.

BAB III : PERENCANAAN DAN PEMBUATAN ALAT

Membahas tentang Perencanaan dan Pembuatan Aplikasi Mikrokontroler Sebagai Pusat Kendali Peralatan Listrik Tanpa Kabel (*wireless*).

BAB IV : PENGUJIAN ALAT

Berisi tentang uji coba alat yang telah dibuat, pengoperasian dan spesifikasi alat.

BAB V : PENUTUP

Merupakan kesimpulan dari pembahasan pada bab-bab sebelumnya dan kemungkinan pengembangan alat.

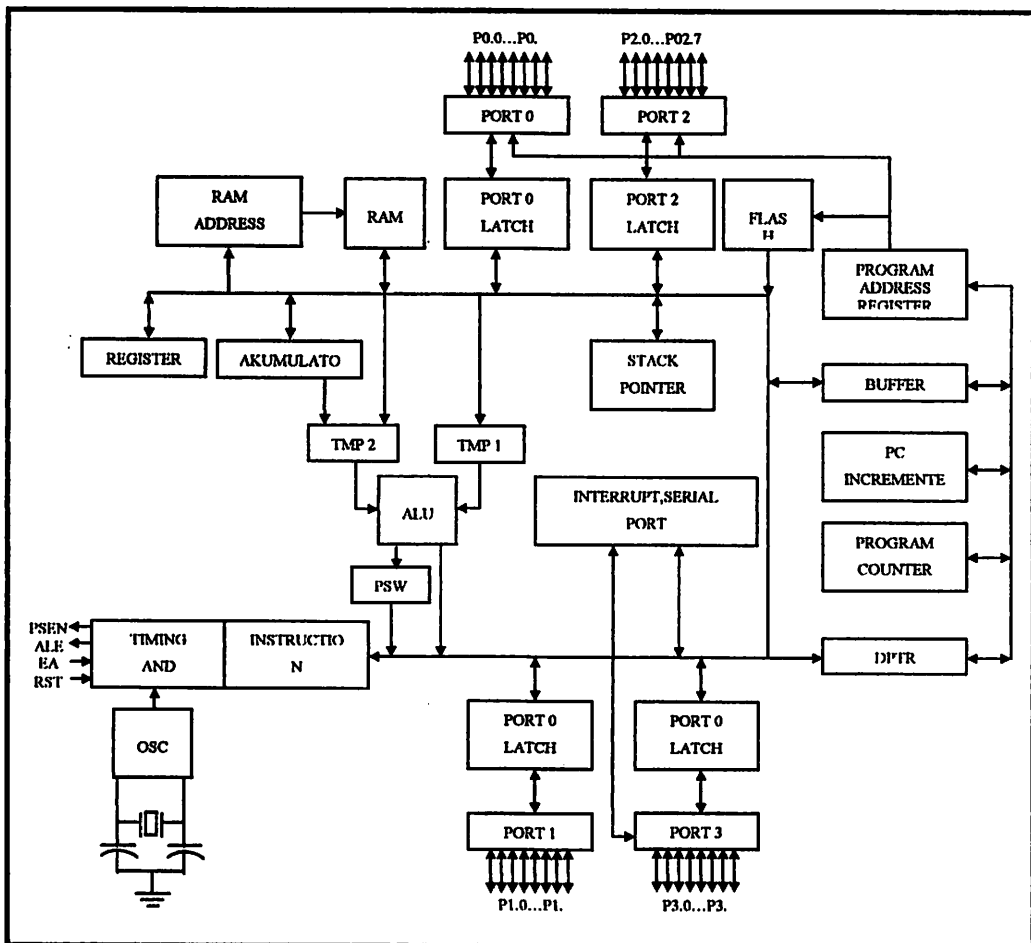
BAB II

TEORI DASAR

Bab ini akan menguraikan tentang dasar-dasar teori dasar yang menunjang dalam perancangan dan pembuatan alat. Uraian teori dalam bab ini meliputi teori IC AT89S51, LCD M1632, TRW 2.4 G dan perangkat-perangkat pendukung lainnya.

2.1. Mikrokontroler AT89S51

2.1.1. Teori umum.



Gambar 2.1. Diagram Blok AT89S51 ^[4]

AT89S51 merupakan sebuah mikrokontroler 8-bit CMOS, Low Power dengan 4Kb flash Programmable and Erasable Read Only Memory (PEROM). IC ini dibuat sesuai dengan standart industri konfigurasi pin dan intruction set dari MCS-51.

- 4Kb Flash Memory.
- 128 Byte Internal RAM.
- 32 I/O Lines.
- 2 Timer/Counter 16-Level.
- 1 Serial Port Full Duplex.
- On Chip Osilator.

AT89S51 mempunyai dua buah Power-Saving mode yang dapat diatur melalui software, yaitu: IDE Mode yang akan menghentikan CPU sebagai RAM, Timer/Counter, Serial Port dan Interrupsi system tetap berfungsi. Power Down Mode yang akan menyimpan ini di RAM, tetapi menahan Oscilator untuk tidak mengaktifkan chip yang lain sampai terjadi reset secara hardware.

2.1.2. Arsitektur AT89S51.

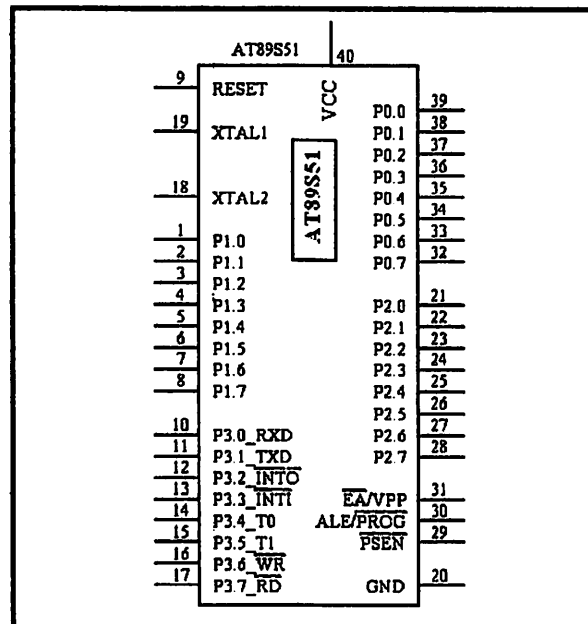
Arsitektur Mikrokontroler AT89S51 adalah sebagai berikut:

- CPU (Central Processing Unit) 8-bit dengan register A (Accumulator) dan B.
- 16-Bit Program Counter (PC) dan data pointer (DPTR).
- 8-Bit Program status word (PSW).
- 4-Bit Stack pointer (SP).
- 4 Kbyte internal EPROM
- 128 byte internal RAM.

- 4 bank register, masing-masing berisi 8 register.
- 16 Byte yang dapat dialamati pada bit level.
- 80 Byte general purpose memory data.
- 32 pin input-output tersusun atas P0-P3 masing-masing 8-bit.
 - 2 Buah 16-bit timer counter.
 - Receiver Register, yaitu : TCON, TMOP, SCON, IP, dan IE.
 - 5 Buah sumber interupt (2 buah sumber interupt eksternal dan 3 buah sumber interupt internal).
 - Oscilator dan Clock internal.

2.1.3. Konfigurasi Pin pada Mikrokontroler AT89S51.

Konfigurasi kaki-kaki mikrokontroler AT89S51 terdiri dari 40 pin, seperti terlihat pada gambar 2.2 sebagai berikut.



Gambar 2.2. Konfigurasi Pin-Pin AT89S51 ^[4]

Fungsi dari tiap-tiap pin adalah sebagai berikut:

1. VCC (Supply tegangan).
2. GND (Ground).
3. Port 0

Merupakan port input-output dua arah dan dikonfigurasi sebagai multiplex dua bus alamat rendah (A0 – A7) dan data selama pengaksesan program memory dan data memory internal.

4. Port 1

Merupakan port input-output dua arah dengan internal pull-up.

5. Port 2

Merupakan port input-output dua arah dengan internal pull-up. Mengeluarkan address tinggi selama pengambilan (fetch) program memory internal dan selama pengaksesan ke data memory port 2 mengeluarkan isi P2SFR (Special Function Register) menerima address tinggi dan beberapa sinyal kontrol selama pemrograman dan verifikasi.

6. Port 3.

Merupakan port input-output dua arah dengan internal pull-up. Port 3 juga memiliki fungsi khusus, yaitu:

- RXD (P3.0) : Port input serial.
- TXD (P3.1) : Port out-put serial.
- INTO (P3.2) : Interrup 0 external.
- INTI (P3.3) : Internal 1 external.
- TO (P3.4) : Input external timer 0.
- TXD (P3.1) : Port out-put serial.

- INTO (P3.2) : Interrupt 0 external.
- INTI (P3.3) : Interrupt 1 external.
- TO (P3.4) : Input external timer 0.
- T1 (P3.5) : Input external timer 1.
- WR (P3.6) : Strobe tulis data memory external.
- RD (P3.7) : Strobe baca data memory external.

7. RST

Input reset.

8. ALE\PROG

Pulsa output ALE digunakan untuk proses “latching” byte address rendah (A0 - A7) selama pengaksesan ke external memory. Pin ini juga untuk memasukkan pulsa program selama pemrograman.

Pada operasi normal ALE mengeluarkan rate konstant yaitu 16 frekuensi osilasi dan boleh digunakan untuk timing external.

9. PSEN

Merupakan strobe baca ke program memory external.

10. EA\VPP

External address enable EA digroundkan jika mengakses memory external.

Untuk mengakses memory internal maka dihubungkan ke VCC.

11. XTAL 1 dan XTAL 2.

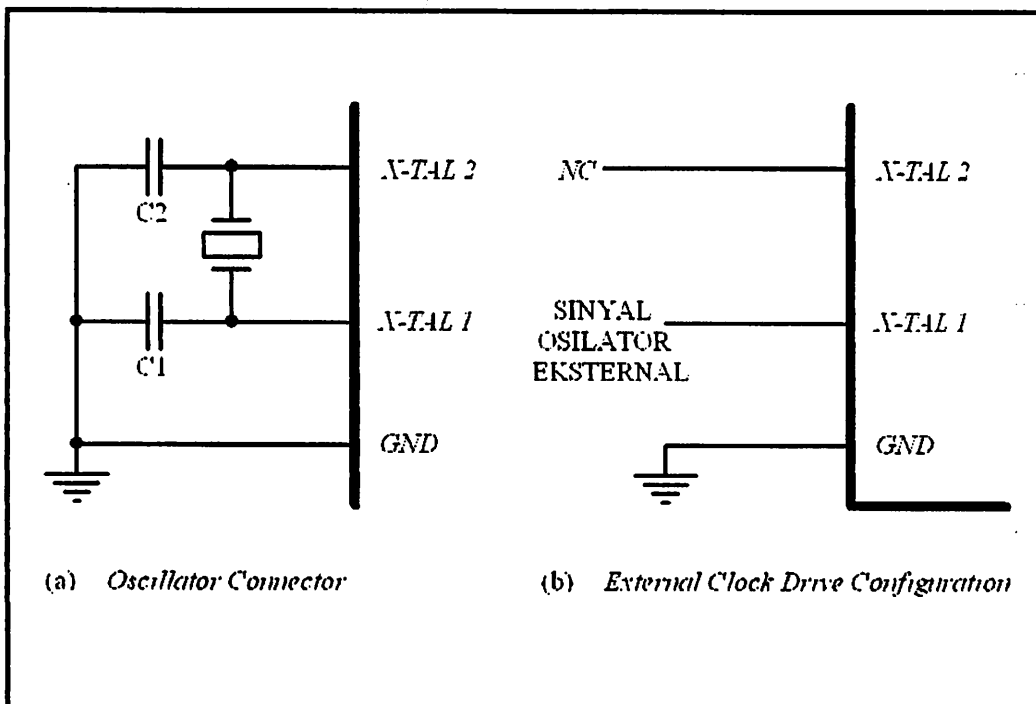
Kaki ini dihubungkan dengan kristal bila menggunakan osilator internal.

XTAL 1 merupakan input inverting osilator amplifier sedangkan XTAL 2 merupakan out-put inverting osilator amplifier.

2.1.4. Karakteristik Oscillator Inverting.

X-TAL 1 dan X-TAL 2 secara berurutan merupakan input dan output dari sebuah inverting amplifier yang dapat dikonfigurasi penggunaannya sebagai on chip oscillator seperti yang ditunjukkan pada gambar 2-3a. X-TAL 1 dan X-TAL 2 ini dapat menggunakan sebuah kristal quartz maupun resonator keramik.

Untuk memberikan AT89S51 dari sumber clock external. Maka pin X-TAL 2 dibiarkan tidak berhubungan dan X-TAL 1 dihubungkan dengan sumber clock external seperti pada gambar 2-3b. Rangkaian ini tidak melakukan duty cycle dari setiap sinyal clock internal, karena input bagi masukan rangkaian clock internal dihubungkan ke flip-flop pembagi dua, tetapi spesifikasi nilai tegangan pada saat tinggi dan rendah, maksimum dan minimumnya harus diberikan.



Gambar 2.3. Karakteristik Oscillator^[4]

2.1.5. Organisasi Memory.

Didalam AT89S51 memiliki ruangan alamat telah dibedakan untuk program memory dan data memory. Pemisahan memori program dan data tersebut membolehkan memori data diakses dengan alamat 8-bit, sehingga dapat dengan cepat dan mudah disimpan dan dimanipulasi oleh CPU 8-bit. Namun demikian, alamat memori data 16-bit bisa juga dihasilkan melalui register DPTR.

2.1.5.1. Program Memory Internal

AT89S51 memiliki program memory internal sebesar 4Kbyte dengan ruang alamat 0000H-0FFFH. Jika alamat-alamat program lebih tinggi daripada 0FFFH, yang melebihi kapasitas ROM\Flash memory internal menyebabkan AT89S51 secara otomatis mengambil Code Byte dari program memory external. Code Byte juga dapat diambil hanya dari external memory dengan alamat 0000H-FFFFH dengan cara menghubungkan Pin EA ke Ground.

2.1.5.2. Data Memory (RAM) Internal.

Ruang alamat bawah memory data (RAM) internal dengan kapasitas 128 byte yaitu 00H-7FH yang terbagi atas 3 daerah, yaitu :

- **4 Bank Register**

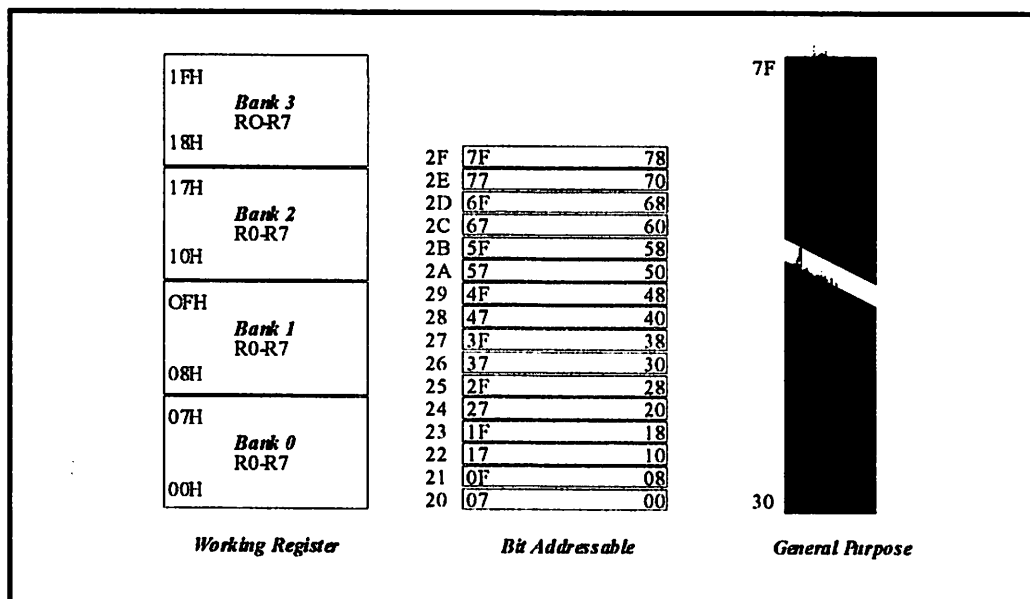
Setiap bank terdiri dari 8 register (R0-R7). Sehingga jumlah register untuk keempat bank register (bank 0 - bank 3) menjadi 32 buah register yang menempati ruang alamat 00H-1FH. Mengaktifkan salah satu bank register dapat dilakukan dengan mengatur RSO-RSI pada PSW (Program Status Word).

- Bit Addressable.

Terdiri dari 16 byte yang berada pada alamat 20H-2FH. Masing-masing 128 bit lokasi ini dapat di alamatkan secara langsung.

- General Purpose.

Terdiri dari 80 byte yang menempati alamat 30H-7FH. yang dapat dialamati secara langsung dan dapat digunakan untuk keperluan umum (General Purpose RAM). Misalkan digunakan untuk lokasi stack.



Gambar 2.4. Organisasi RAM Internal ^[4]

Tabel 2-1. Pengaturan RS0-RS1 Untuk Select Register Bank ^[4]

RS1	RS0	Select Register Bank
0	0	Bank 0
0	1	Bank 1
1	0	Bank 2
1	1	Bank 3

2.1.5.3. SFR (Special Function Register).

Untuk mengoperasikan AT89S51 yang tidak menggunakan alamat internal RAM (00-7FH) dilakukan oleh SFR yang beraddress 80H-FFH, tetapi tidak semua address tersebut digunakan sebagai SFR. Fungsi-fungsi SFR dijelaskan oleh tabel 2 - 2 sebagai berikut.

Tabel 2-2. 128 Byte Special Function Register ^[4]

SYMBOL	NAME	ADDRESS
ACC	ACCUMULATOR	0E0H
B	B REGISTER	0F0H
PSW	PROGRAM STATUS WORD	0D0H
SP	STACK POINTER	81H
DPTR	DATA POINTER 2 BYTE	
DPL	LOW BYTE	82H
DPH	HIGH BYTE	83H
P0	PORT 0	80H
P1	PORT 1	90H
P2	PORT 2	0A0H
P3	PORT 3	080H
IP	INTERUPT PERIORITY CONTROL	088H
IE	INTERUPT ENABLE CONTROL	0ABH
TMOD	TIMER / COUNTER MODE CONTROL	89H
TCON	TIMER/COUNTER CONTROL	88H
TH0	TIMER/COUNTER 0 HIGH CONTROL	8CH
TL0	TIMER/COUNTER 0 LOW CONTROL	8DH
TH1	TIMER/COUNTER 1 HIGH CONTROL	8DH
TL1	TIMER/COUNTER 1 LOW CONTROL	8CH
SCON	SERIAL CONTROL	98H
SBUF	SERIAL DATA BUFFER	99H
PCON	POWER CONTROL	87H

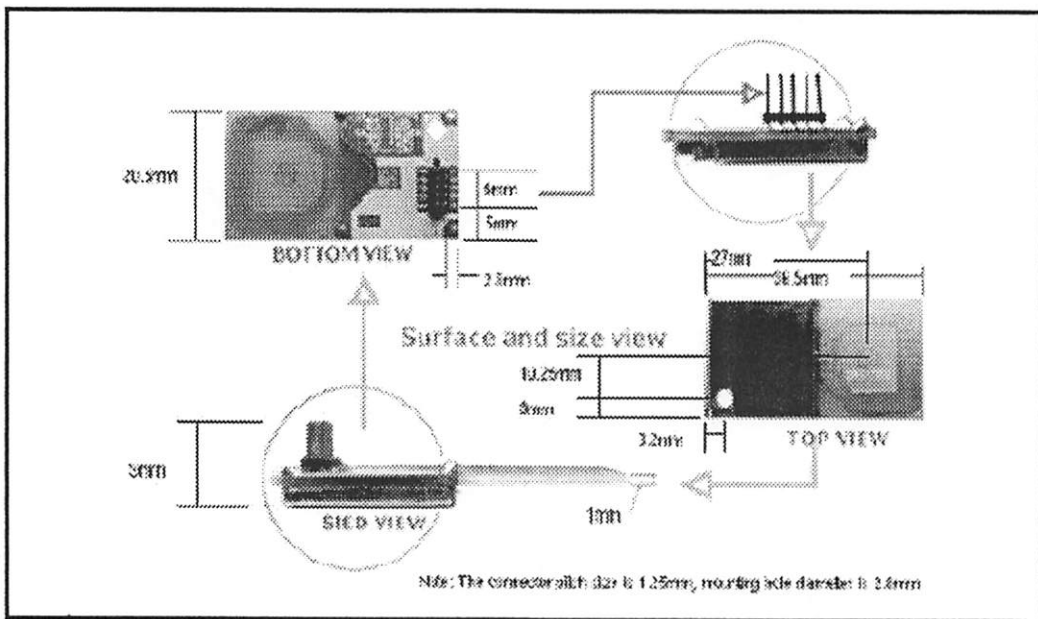
2.2. Modul TRW 2.4 G

Untuk dapat mengirimkan data serial melalui udara minimal diperlukan suatu device yang dapat melakukan proses penumpangan data serial digital ke

frekuensi pembawa dengan frekuensi yang lebih tinggi untuk kemudian dipancarkan ke udara. Salah satu contoh device yang dapat melakukan hal tersebut adalah modul TRW-2.4G Wireless Transceiver. Modul tersebut dapat mengirimkan dan menerima data serial melalui media udara, dengan kecepatan 1Mbps atau 250Kbps dengan range frekuensi 2.4-2.524 GHz ISM band. Penggunaan modul tersebut relatif cukup praktis karena dari segi ukuran cukup kecil dan sudah terdapat built-in FIFO buffer data, power amplifier dan antena. Selain ini juga terdapat built-in CRC, sehingga kesalahan data akibat gangguan ketika proses pengiriman dapat diminimalisasi. Modul tersebut bekerja dengan supply antara 1.9 sampai 3.6VDC. Dalam 1 modul terdapat sebuah pengirim dan dua buah penerima yang dapat digunakan secara simultan. Bentuk fisik dari modul TRW-2.4G adalah seperti pada gambar 2.5. Data serial yang akan dipancarkan melalui RF diumpankan ke modul TRW-2.4G oleh mikrokontroler secara synchronous serial. Begitu pula data yang diterima, akan diambil oleh mikrokontroler secara synchronous serial.

Pada modul tersebut terdapat dua macam mode proses pengiriman dan penerimaan data serial, yaitu mode Shockburst dan mode Direct. Mode Shockburst adalah proses pengiriman data dengan kecepatan tinggi (1Mbps) melalui jalur 2.4GHz band tanpa memerlukan mikrokontroler berkecepatan tinggi untuk proses pengiriman data. Sedangkan mode Direct adalah mode pengiriman data secara konvensional, sehingga modul TRW-2.4G bekerja seperti RF device seperti pada umumnya. Pada mode Direct, data serial yang akan dikirimkan melalui RF haruslah berkecepatan 1Mbps atau 250Kbps, agar penerima dapat mendeteksi sinyalnya, sehingga memerlukan mikrokontroler yang berkecepatan tinggi pula.

Agar dapat bekerja dengan mode yang diinginkan, modul TRW-2.4G perlu dikonfigurasi. Proses konfigurasi ini dapat dilakukan hanya dengan menggunakan 3 kabel interface, dengan kecepatan transfer datanya diatur oleh mikrocontroller, karena menggunakan transmisi data synchronous serial.



Gambar 2.5. Modul TRW 2.4G ^[13]

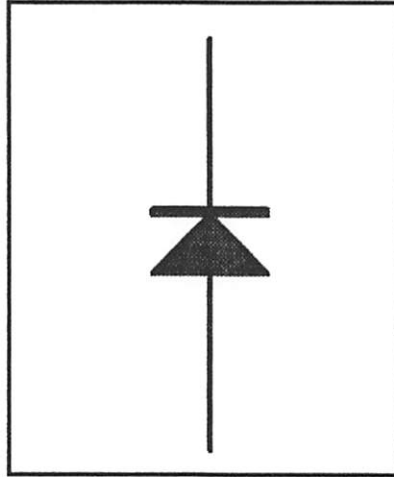
2.3. Dioda

Semua dioda prinsip kerjanya adalah sebagai penyearah. Tetapi karena proses pembuatan, bahan dan penerapannya yang berbeda-beda, maka namanya pun berbeda-beda serta memiliki karakteristik yang berbeda pula. Macam-macam dioda antara lain: Dioda Penyearah, Dioda Zener, Dioda Tunnel, Dioda Varaktor dan Dioda Photo.

2.3.1. Dioda Penyearah

Kebanyakan dioda penyearah dibuat dari bahan silicon, mengingat kemampuan dan keandalannya yang tinggi. Mampu dilewati arus yang besar dan

dapat bekerja pada suhu yang tinggi. Keuntungan ini dimungkinkan karena elektron tersebut baru bias terlepas dari intinya oleh kekuatan atau energi yang sangat besar. Penyearah disini berarti merubah arus bolak-balik (AC) menjadi arus searah (DC) yang berfrekuensi rendah. Gambar simbol dioda seperti dibawah ini:



Gambar 2.6. Simbol Dioda ^[7]

2.4. Kapasitor

Kapasitor adalah komponen elektronika yang terdiri dari plat konduktor yang satu disekat terhadap yang lainnya dengan isolator (dielektrik) dan berfungsi sebagai pelawan untuk arus bolak-balik, filter, tuning, kopel antar rangkaian, pembangkit gelombang sinus dan konduktor khusus. Untuk menentukan banyaknya muatan listrik yang dapat disimpan oleh kapasitor, dapat dirumuskan sebagai berikut:

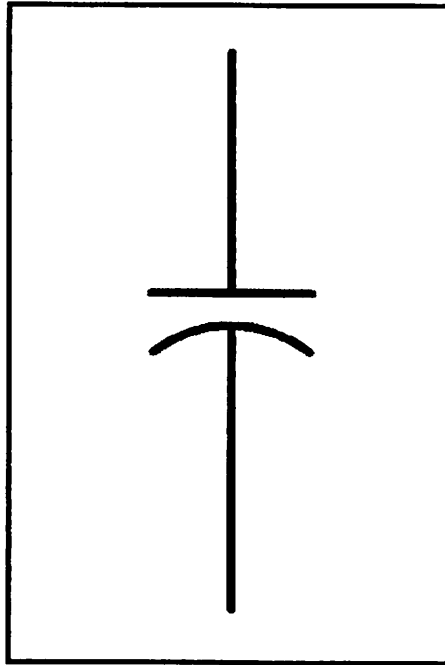
$$Q = C \times V \dots \dots \dots (2-1)$$

Dimana :

Q = Banyaknya muatan dalam Coulomb

C = Kapasitor dalam farad

V = Tegangan yang ada diantara keping keping dalam Volt



Gambar 2.7. Simbol Kapasitor ^[7]

Sebuah kapasitor dikatakan mempunyai kapasitas satu farad apabila muatan sebesar satu coulomb membuat tegangan naik sebesar satu volt, apabila farad ternyata terlampau besar sehingga digunakan mikrofarad dan satuan lainnya yaitu nano farad dan pikofarad. Kapasitor yang berfungsi sebagai penyimpan tenaga listrik dalam praktek dapat digunakan untuk pewaktu, untuk pewaktu kapasitor dengan resistor yang disebut sebagai rangkaian sebagai rangkaian RC.

Rumus pada rangkaian RC sebagai berikut:

$$RC = R \times C \dots\dots\dots(2-2)$$

Dimana:

R = Tahanan dalam kapasitor

C = Kapasitor dalam farad

RC = Konstanta waktu dalam detik

Dalam kapasitor terdapat reaktansi kapasitor, jika kapasitor tersebut diterapkan pada arus bolak balik sinus, tegangan akan tertinggal sejauh 90° .

Untuk menentukan besarnya nilai reaktansi adalah:

$$X_c = \frac{1}{3,14.FC} \dots\dots\dots(2-3)$$

Dimana:

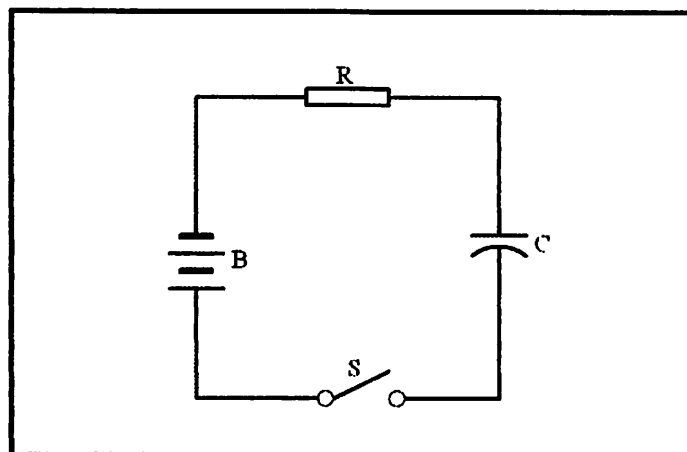
X_c = Reaktansi kapasitor

F = Frekuensi dalam ohm

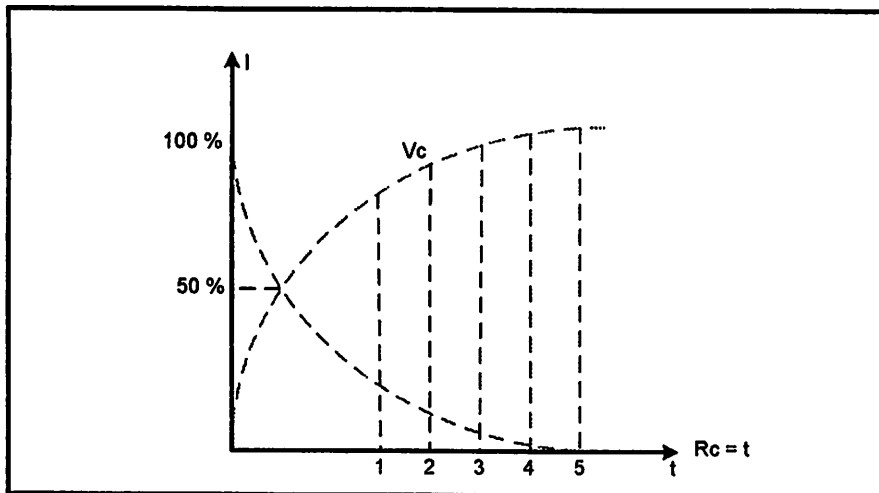
C = kapasitas dalam farad

2.4.1. Cara Kerja Kapasitor

Terjadinya perubahan tegangan pada kapasitor menyebabkan perubahan arus listrik, sedangkan perubahan tersebut berbanding lurus dengan perubahan tegangan, energi akan timbul kembali pada saat tegangan berkurang menjadi nol. Pada saat saklar (S) posisi ON, arus dari tegangan V_2 mengalir ke dalam kapasitor dan beberapa saat kapasitor dapat terisi penuh. Dalam kondisi ini berarti $V_s = V_c$ dan $I = 0$.



Gambar 2.8. Rangkaian Pengisian Kapasitor ^[7]



Gambar 2.9. Grafik Tegangan dan Arus Pengisian Kapasitor ^[7]

Berdasarkan grafik diatas, dapat kita ketahui bahwa setelah terjadi pengisian kapasitor secara penuh dan kapasitor akan membuang melalui R, maka C tidak seketika menjadi netral dan dalam menetralkan muatan tersebut diperlukan waktu tertentu.

2.5. Resistor

Pada dasarnya semua bahan memiliki sifat resistif namun beberapa bahan seperti tembaga, perak, emas dan bahan metal umumnya memiliki resistansi yang sangat kecil. Bahan-bahan tersebut menghantar arus listrik dengan baik, sehingga dinamakan konduktor. Kebalikan dari bahan yang konduktif, bahan material seperti karet, gelas, karbon memiliki resistansi yang lebih besar menahan aliran elektron dan disebut sebagai insulator.

Resistor adalah komponen dasar elektronika yang digunakan untuk membatasi jumlah arus yang mengalir dalam satu rangkaian. Sesuai dengan namanya resistor bersifat resistif dan umumnya terbuat dari bahan karbon. Dari

hukum Ohm diketahui, resistansi berbanding terbalik dengan jumlah arus yang mengalir melaluinya. Satuan resistansi dari suatu resistor disebut Ohm atau dilambangkan dengan simbol Ω (Omega).

Tipe resistor yang umum adalah berbentuk tabung dengan dua kaki tembaga di kiri dan kanan. Pada badannya terdapat lingkaran membentuk gelang kode warna untuk memudahkan pemakai mengenali besar resistansi tanpa mengukur besarnya dengan Ohm meter. Kode warna tersebut adalah standar manufaktur yang dikeluarkan oleh EIA (Electronic Industries Association) seperti yang ditunjukkan pada tabel 2-3.

Tabel 2-3. Tabel Gelang Warna Resistor ^[7]

Warna	Nilai	Faktor Pengali	Toleransi
Hitam	0	1	
Coklat	1	10	1%
Merah	2	100	2%
Jingga	3	1.000	
Kuning	4	10.000	
Hijau	5	100.000	
Biru	6	10^6	
Violet	7	10^7	
Abu-abu	8	10^8	
Putih	9	10^9	
Emas	-	0.1	5%
Perak	-	0.01	10%
Tanpa warna	-	-	20%

Resistansi dibaca dari warna gelang yang paling depan ke arah gelang toleransi berwarna coklat, merah, emas atau perak. Biasanya warna gelang toleransi ini berada pada badan resistor yang paling pojok atau juga dengan lebar yang lebih menonjol, sedangkan warna gelang yang pertama agak sedikit ke

dalam. Dengan demikian pemakai sudah langsung mengetahui berapa toleransi dari resistor tersebut. Kalau kita telah bisa menentukan mana gelang yang pertama selanjutnya adalah membaca nilai resistansinya.

Jumlah gelang yang melingkar pada resistor umumnya sesuai dengan besar toleransinya. Biasanya resistor dengan toleransi 5%, 10% atau 20% memiliki 3 gelang (tidak termasuk gelang toleransi). Tetapi resistor dengan toleransi 1% atau 2% (toleransi kecil) memiliki 4 gelang (tidak termasuk gelang toleransi). Gelang pertama dan seterusnya berturut-turut menunjukkan besar nilai satuan, dan gelang terakhir adalah faktor pengalinya.

Misalnya resistor dengan gelang kuning, violet, merah dan emas. Gelang berwarna emas adalah gelang toleransi. Dengan demikian urutan warna gelang resistor ini adalah, gelang pertama berwarna kuning, gelang kedua berwarna violet dan gelang ke tiga berwarna merah. Gelang ke empat tentu saja yang berwarna emas dan ini adalah gelang toleransi. Dari tabel 2-3 diketahui jika gelang toleransi berwarna emas, berarti resistor ini memiliki toleransi 5%. Nilai resistansinya dihitung sesuai dengan urutan warnanya. Pertama yang dilakukan adalah menentukan nilai satuan dari resistor ini. Karena resistor ini resistor 5% (yang biasanya memiliki tiga gelang selain gelang toleransi), maka nilai satuannya ditentukan oleh gelang pertama dan gelang kedua.

Masih dari tabel 2-3 diketahui gelang kuning nilainya = 4 dan gelang violet nilainya = 7. Jadi gelang pertama dan kedua atau kuning dan violet berurutan, nilai satuannya adalah 47. Gelang ketiga adalah faktor pengali, dan jika warna gelangnya merah berarti faktor pengalinya adalah 100. Sehingga dengan ini

diketahui nilai resistansi resistor tersebut adalah nilai satuan x faktor pengali atau $47 \times 100 = 4.7K \text{ Ohm}$ dan toleransinya adalah 5%.

Spesifikasi lain yang perlu diperhatikan dalam memilih resistor pada suatu rancangan selain besar resistansi adalah besar watt-nya. Karena resistor bekerja dengan dialiri arus listrik, maka akan terjadi disipasi daya berupa panas sebesar $W=I^2R$ watt. Semakin besar ukuran fisik suatu resistor bisa menunjukkan semakin besar kemampuan disipasi daya resistor tersebut.

Umumnya di pasar tersedia ukuran 1/8, 1/4, 1, 2, 5, 10 dan 20 watt. Resistor yang memiliki disipasi daya 5, 10 dan 20 watt umumnya berbentuk kubik memanjang persegi empat berwarna putih, namun ada juga yang berbentuk silinder. Tetapi biasanya untuk resistor ukuran jumbo ini nilai resistansi dicetak langsung dibadannya, misalnya $100 \Omega 5W$.

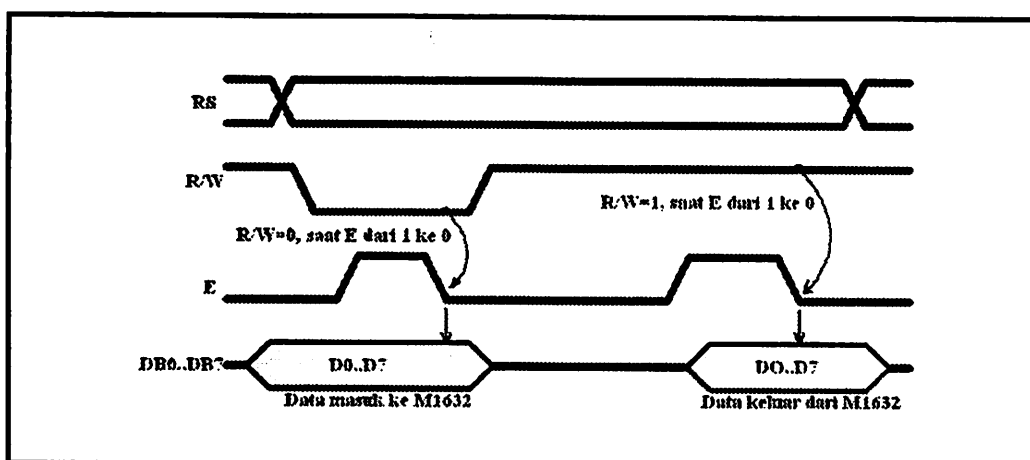
2.6. LCD (Liquid Crystal Display) M1632

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 menggunakan M1632 cukup mengirimkan kode-kode ASCII dari informasi yang ditampilkan seperti layaknya memakai sebuah printer.

2.6.1. 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 lainnya **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 2.10 bisa dijabarkan sebagai berikut :



Gambar 2.10. Mengirim/Mengambil Data Ke/Dari M1632 ^[12]

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 2.11 memperlihatkan hubungan AT89C2051 dengan M1632, dalam gambar tersebut **P3.7** dipakai untuk mensimulasikan sinyal **RS**, **P3.5** sebagai **R/W** dan **P3.4** sebagai **E**. Lewat program dibangkitkan sinyal-sinyal pada ketiga kaki Port 3 ini, sesuai dengan persyaratan yang dikehendaki M1632.

Potongan Program 1 merupakan sub-rutin untuk mengendalikan M1632 yang dihubungkan ke AT89C2051 seperti terlihat di gambar 2.11, 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 **Kirim Perintah** dan sub-rutin **Kirim ASCII**, sedangkan bagian mengambil data dari M1632 terdiri dari sub-rutin **Ambil Status** dan sub-rutin **Ambil ASCII**.

Sebelum pengiriman data, Akumulator A sudah terlebih dulu diisi dengan data yang akan dikirim. Data yang dikirim dengan sub-rutin **Kirim Perintah** akan diterima M1632 sebagai perintah untuk mengatur kerja M1632, dan data yang dikirim dengan sub-rutin **Kirim ASCII** 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 **Kirim Perintah** dan **Kirim ASCII** terletak pada nilai **RS** pada saat sub-rutin itu bekerja. Sub-rutin **Kirim Perintah** bekerja dengan **RS='0'** (baris 6), data yang dikirim AT89C2051 diterima M1632 sebagai perintah untuk mengatur kerja M1632. Sub-rutin **Kirim ASCII** 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 Tunggu Dulu.

Proses pengambilan data dari M1632 dijelaskan sebagai berikut :

1. Seperti bahasan di atas, **RS** dipakai untuk memilih Register Perintah/Status, sub-rutin **Ambil Status** bekerja dengan **RS=0** dan sub-rutin **Ambil ASCII** bekerja dengan **RS='1'**.
2. 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 AT89C2051 dengan M1632 :

```

01:      E      bit   P3.4      ; sinyal E   di P3.4
02:      RW     bit   P3.5      ; sinyal R/W di P3.5
03:      RS     bit   P3.7      ; sinyal RS  di P3.7
04:      ;Comment In Here
05:      KirimPerintah:
06:      CLR   RS                ; RS=0 : register perintah
07:      SJMP  OutByte
08:      ;
09:      KirimASCII:
10:      SETB  RS                ; RS=1 : Display Data RAM
11:      OutByte:
12:      CLR   RW                ; RW = '0', kirim data
13:      MOV   P1,A              ; siapkan data di D0..D7
14:      SETB  E                 ; buat pulsa positif
15:      CLR   E                 ; sesaat
16:      ;
17:      TungguDulu:
18:      ACALL AmbilStatus
19:      JB   A.7,TungguDulu
20:      RET
21:      AmbilStatus:
22:      CLR   RS                ; RS=0 : register status
24:      SJMP  InByte

```

```

26:      AmbilASCII:
27:      SETB RS                ; RS=1 : Display Data RAM
28:      InByte:
29:      SETB RW                ; RW = '1', ambil data
30:      SETB E                  ; minta data pada M1632
31:      MOV  A,P1              ; ambil data
32:      CLR  E                  ; kembalikan E ke '0'
33:      RET

```

2.6.3. Mengatur tampilan M1632

M1632 mempunyai seperangkat perintah untuk mengatur tata kerjanya, perangkat perintah tersebut meliputi perintah untuk menghapus tampilan, meletakkan kembali cursor pada baris huruf 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:

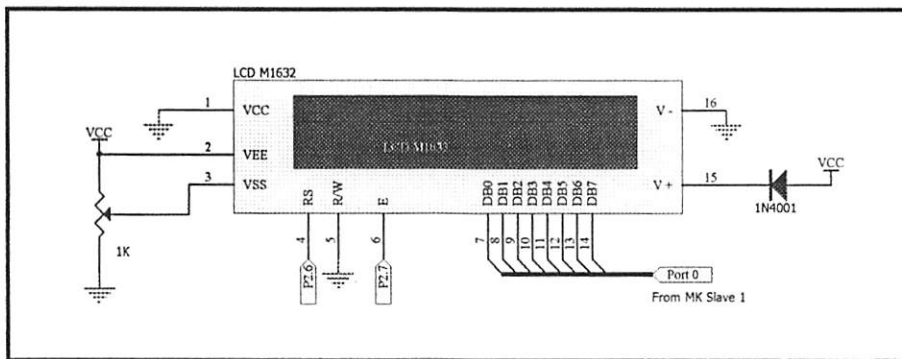
1. Tunggu dulu selama 15 mili-detik atau lebih.
2. Kirimkan perintah 30h, artinya transfer data antar M1632 dan mikrokontroler dilakukan dengan mode 8 bit
3. Tunggu selama 4.1 mili-detik
4. Kirimkan sekali lagi perintah 30h
5. Tunggu lagi selama 100 mikro-detik

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**, **RW** dan **RS** tidak disimulasikan di **P3.4**; **P3.5** dan **P3.7**. 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 mikrokontroler MCS 51 :



Gambar 2.12. Rangkaian LCD M1632 [6]

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 ± 5 volt.
7. Otomatis reset saat catu daya dinyalakan.

Tabel 2-4. Fungsi Pin – Pin LCD ^[12]

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.
- Cursor home : hanya membersihkan tampilan dan kursor kembali ke semula.
- Empty mode Set : layar beraksi sebagai tampilan tulis.

S : 1/0 = menggeser layar.

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

- 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

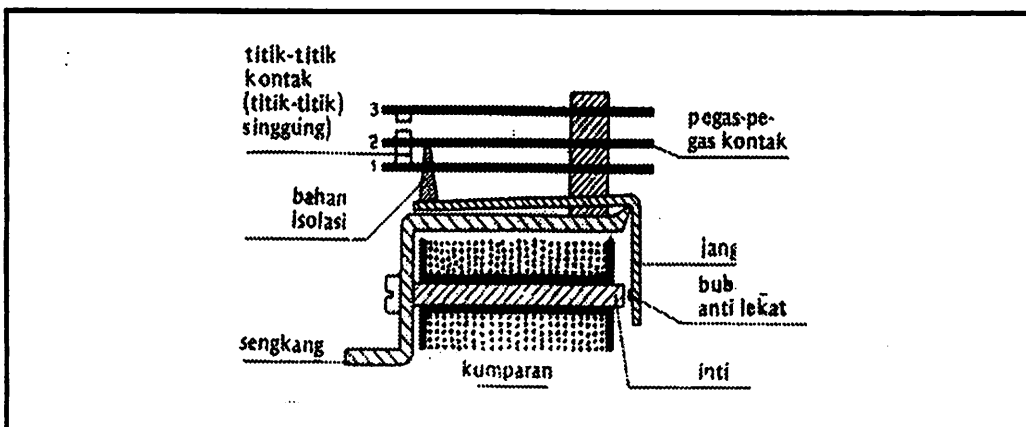
- **CG RAM address set** : menulis alamat RAM ke karakter
- **DD RAM address set** : menulis alamat RAM ke tampilan
- **BF/address set** : BF = 1/0, LCD dalam keadaan sibuk atau tidak sibuk.

- Data write to CG RAM or DD RAM : membaca byte dari alamat terakhir RAM yang dipilih.

2.7. Relay

Relay merupakan salah satu jenis saklar magnetik yang dapat memutuskan atau menghubungkan kontak-kontak dari jarak jauh dengan arus. Sebuah *relay* terdiri dari satu kumparan dan inti yang apabila dialiri arus kumparan tersebut akan menjadi magnet dan menutup atau membuka kontak-kontak. Keuntungan dari *relay* adalah dapat menghubungkan daya yang besar dengan memberi daya yang kecil pada kumparannya. *Relay* digolongkan berdasarkan arusnya menjadi dua yaitu:

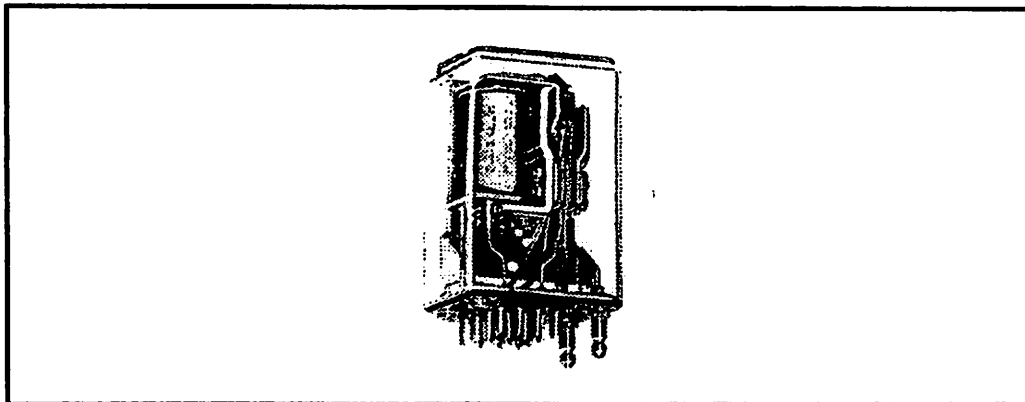
- *Relay* arus searah (*DC Relay*).
- *Relay* arus bolak-balik (*AC Relay*).



Gambar 2.13. Konstruksi *Relay*^[7]

Sebuah sengkang dipasang pada inti belitan kumparan, jika ada arus yang mengalir, inti akan menjadi magnet dan jangkar tertarik. Pada jangkar terpasang pegas spiral sehingga jangkar akan kembali jika arus diputuskan. Jadi pada dasarnya *relay* punya dua kontak, normal terbuka (*Normaly Open*) dan normal

tertutup (*Normaly Close*). Dalam praktek kontak-kontak *relay* tidak hanya dua atau empat, tetapi sampai 24 kontak yang merupakan kombinasi dari kontak NO (*Normaly Open*) dan NC (*Normaly Close*).



Gambar 2.14. Sebuah *relay* praktis dari Siemen ^[7]

Dari hasil perhitungan tahanan serta tegangan maksimum kita dapat menentukan arus maksimum yang dapat mengalir pada *relay* tersebut dengan menggunakan persamaan:

$$I_{relay} = \frac{V_{cc}}{R_{relay}} \quad \text{Ampere}$$

Dimana :

I_{relay} = Arus *relay* (*Ampere*)

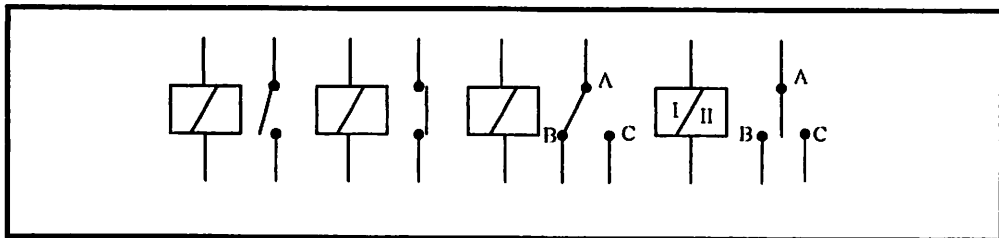
V_{cc} = Tegangan catu daya (*Volt*)

R_{relay} = Tahanan *relay* (*Ohm*)

Tetapi pada umumnya *relay-relay* yang ada sekarang ini telah ditentukan nilai tahanan serta tegangannya. Beberapa jenis *relay* menggunakan tegangan kerja untuk kumparannya dari 5 *Volt* sampai dengan 24 *Volt* dc dan membutuhkan arus kerja kurang dari 100 mA.

Berdasarkan jenis anak kontaknya *relay* dapat dibedakan sebagai berikut:

- **SPST (Single Pole Single Throw) Normally Open**, yaitu *relay* dengan satu kutub satu saklar dimana saklar ini akan tetap terbuka sampai kumparan *relay* mendapatkan tegangan yang memadai untuk menutup saklar.
- **SPST (Single Pole Single Throw) Normally Closed**, yaitu *relay* dengan satu kutub satu saklar dimana saklar ini akan tetap tertutup sampai kumparan *relay* mendapat tegangan.
- **SPDT (Single Pole Double Throw)**, yaitu *relay* dengan satu kutub dan dua posisi saklar.
- **DPDT (Double Pole Double Throw)**, yaitu *relay* dengan dua kutub dan dua posisi saklar.



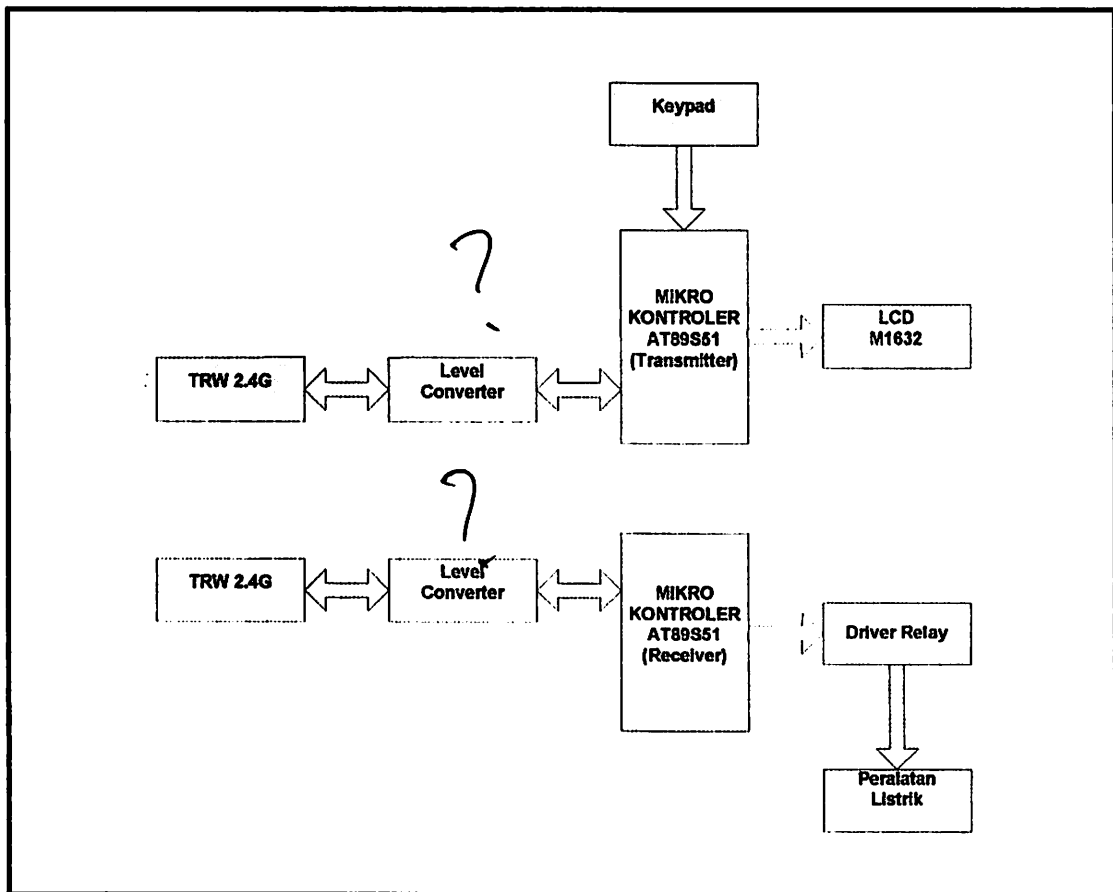
Gambar 2.15. Kemungkinan-kemungkinan Kontak dari *Relay*^[7]

BAB III

PERENCANAAN DAN PEMBUATAN ALAT

3.1. Diagram Blok

Aplikasi Mikrokontroler Sebagai Pusat Kendali Peralatan Listrik Tanpa Kabel (*wireless*) yang akan dibuat, digambarkan secara diagram blok sebagai berikut:



Gambar 3.1. Diagram Blok

Rangkaian Transmitter:

- Mikrokontroler AT89S51 berfungsi sebagai Pengendali
- Keypad, digunakan untuk seting peralatan listrik yang dikontrol.
- TRW 2-4G, digunakan sebagai transmitter data seting peralatan listrik melalui RF (Radio Frekuensi).
- Level Converter, digunakan agar TRW 2-4G dapat berkomunikasi dengan Mikrokontroler. TRW-2.4G menggunakan catu daya +3 VDC dan memiliki I/O yang bekerja pada level tegangan 3 V sedangkan mikrokontroler menggunakan catu daya +5 VDC dan memiliki I/O yang bekerja pada level tegangan 5. sehingga diperlukan rangkaian Level Converter.
- LCD M1632 digunakan sebagai display kerja alat..

Rangkaian Receiver:

- Mikrokontroler AT89S51 berfungsi sebagai Pengendali
- TRW 2-4G, digunakan sebagai Receiver data seting peralatan listrik melalui RF (Radio Frekuensi).
- Level Converter, digunakan agar TRW 2-4G dapat berkomunikasi dengan Mikrokontroler. TRW-2.4G menggunakan catu daya +3 VDC dan memiliki I/O yang bekerja pada level tegangan 3 V sedangkan mikrokontroler menggunakan catu daya +5 VDC dan memiliki I/O yang bekerja pada level tegangan 5. sehingga diperlukan rangkaian Level Converter.
- Driver Relay digunakan untuk mensaklar peralatan listrik.
- Peralatan Listrik sebagai objek yang dikontrol

3.2. Prinsip Kerja Alat

Rangkaian transmitter mengirimkan perintah melalui gelombang radio kepada rangkaian receiver, dimana perintah tersebut adalah perintah untuk mengaktifkan/mematikan peralatan listrik.

3.3. Mikrokontroler AT89S51

Mikrokontroler AT89S51 dirancang untuk dapat berdiri sendiri, karena sudah terdapat EPROM, RAM serta Port I/O *internal*. Untuk berhubungan dengan peralatan luar *chip* dibutuhkan 3 Bus yaitu :

➤ *Data Bus*

Yaitu jalur untuk input - output data yang lebarnya sesuai dengan data yang diolah oleh mikrokontroler, yaitu 8 bit.

➤ *Address Bus*

Yaitu jalur input - output atau dari *memori* yang dihubungi, sehingga pada suatu saat hanya ada satu *device* yang berhubungan dengan CPU. Lebar *address bus* mikrokontroler AT89S51 adalah 16 bit (A0 - A15).

➤ *Control Bus*

Berfungsi sebagai pengatur *sinkronisasi* hubungan antara CPU dengan *device* Luar.

3.3.1. Pemetaan Memori

Mikrokontroler AT89S51 memiliki 16 bit address (A0 - A15) dengan demikian kapasitas maksimumnya adalah $2^{16} = 65536 \text{ byte} = 64 \text{ Kbyte}$ dengan alamat 0000H-FFFFH. Jika alamat-alamat program lebih tinggi dari 0FFFH, yang melebihi kapasitas RAM *internal* menyebabkan mikrokontroler secara otomatis mengambil *code byte* dari

program *memori* eksternal. *Code byte* juga hanya diambil dari *memori* eksternal dengan alamat 0000H-0FFFH dengan menghubungkan *Pin* EA ke ground.

Dalam perancangan ini hanya menggunakan 4 *Kbyte*, karena program sudah mencukupi, sehingga EA dihubungkan ke VCC. Mikrokontroller AT89S51 memiliki 4*Kbyte memori internal* yang dapat diprogram dan dihapus sesuai dengan keinginan, dan bersifat *non volatile* (tidak hilang pada saat catu daya terputus).

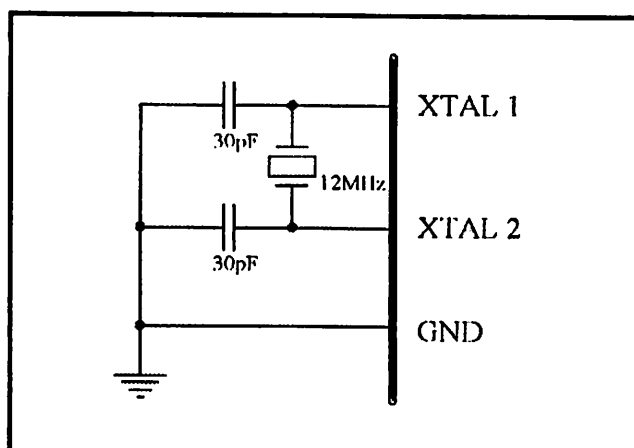
3.3.2. Rangkaian *Clock*

Mikrokontroller AT89S51 ini memiliki *internal clock*, yang berfungsi sebagai sumber *clock*, tapi masih diperlukan rangkaian tambahan untuk membangkitkan *clock* yang diperlukan.

Rangkaian ini terdiri dari 2 buah kapasitor dan sebuah kristal dengan ketentuan :

C1 dan C2 = 20 pF - 40 pF (digunakan untuk kristal)

= 30 pF - 50 pF (digunakan untuk keramik resonator)



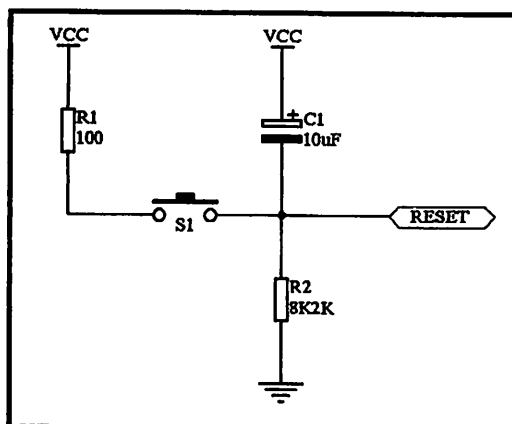
Gambar 3.2. Rangkaian *Clock* AT89S51

3.3.3. Rangkaian *Reset*

Rangkaian *reset* bertujuan agar mikrokontroler dapat menjalankan proses mulai dari awal. Rangkaian *reset* untuk mikrokontroler dirancang agar mempunyai kemampuan *power on reset* yaitu *reset* yang terjadi pada saat sistem dinyalakan untuk pertama kalinya. *Reset* juga dapat dilakukan secara manual dengan menyediakan tombol yang berupa switch. Jika saklar S1 ditekan, *reset* bekerja secara manual, aliran arus akan mengalir dari VCC melalui R1 menuju kaki RST. Tegangan di RST atau VR2 akan berubah menjadi :

$$\begin{aligned} VR2 &= \frac{R2 \times VCC}{R1 + R2} \\ &= \frac{8200 \times 5}{100 + 8200} \\ &= 4,94 \text{ volt} \end{aligned}$$

Pada tegangan 4,94 volt menyebabkan kaki RST berlogika 1 pada saat saklar ditekan. Saat saklar dilepas, aliran arus dari VCC melalui R1 akan berhenti dan tegangan pada kaki RST akan turun menuju ke nol sehingga logika berubah menjadi nol dan proses *reset* selesai. Rangkaian *reset* ini dapat dilihat seperti pada gambar berikut :

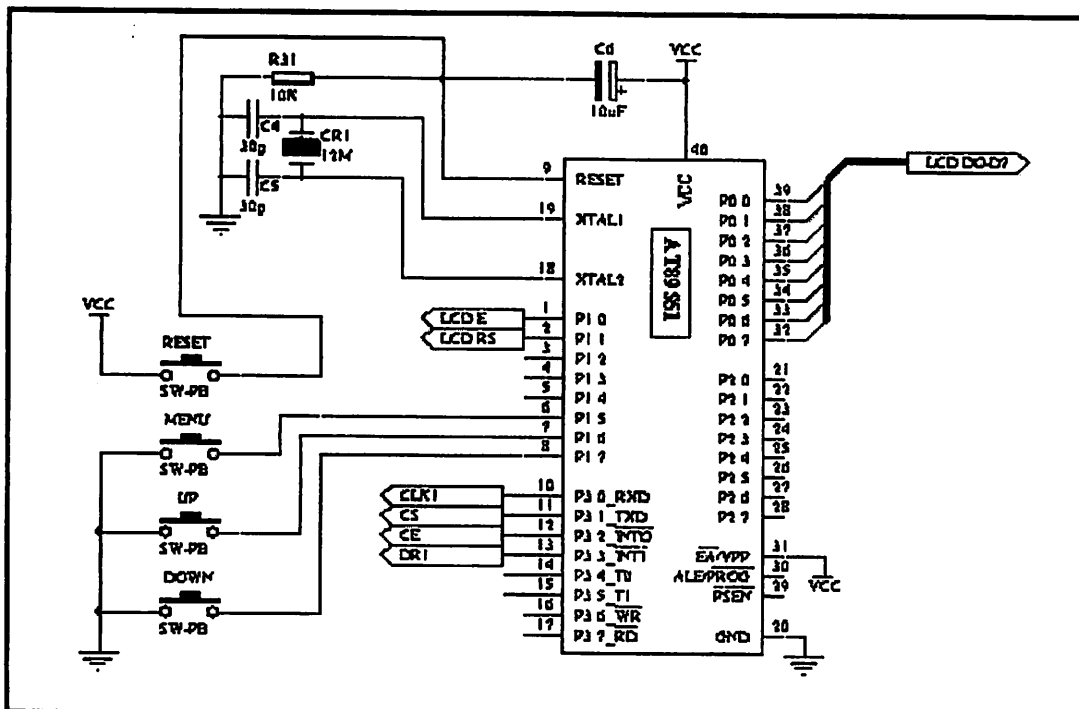


Gambar 3.3. Rangkaian *Reset*

3.3.4. Hubungan Pin Pada AT89S51 (Rangkaian Transmitter)

Diskripsi konfigurasi *Pin-Pin* AT89S51

- *Pin* 1 dan 2 : dihubungkan ke LCD (E dan RS)
- *Pin* 6 : dihubungkan ke Push-button Menu
- *Pin* 7 : dihubungkan ke Push-button Up
- *Pin* 8 : dihubungkan Push-button Down
- *Pin* 9 : dihubungkan Push-button Reset
- *Pin* 10-14 : dihubungkan TRW 24-G
- *Pin* 18-19 : dihubungkan X-tal 12M
- *Pin* 20 : dihubungkan ke ground
- *Pin* 32-39 : dihubungkan ke LCD
- *Pin* 31 dan 40 : dihubungkan ke VCC

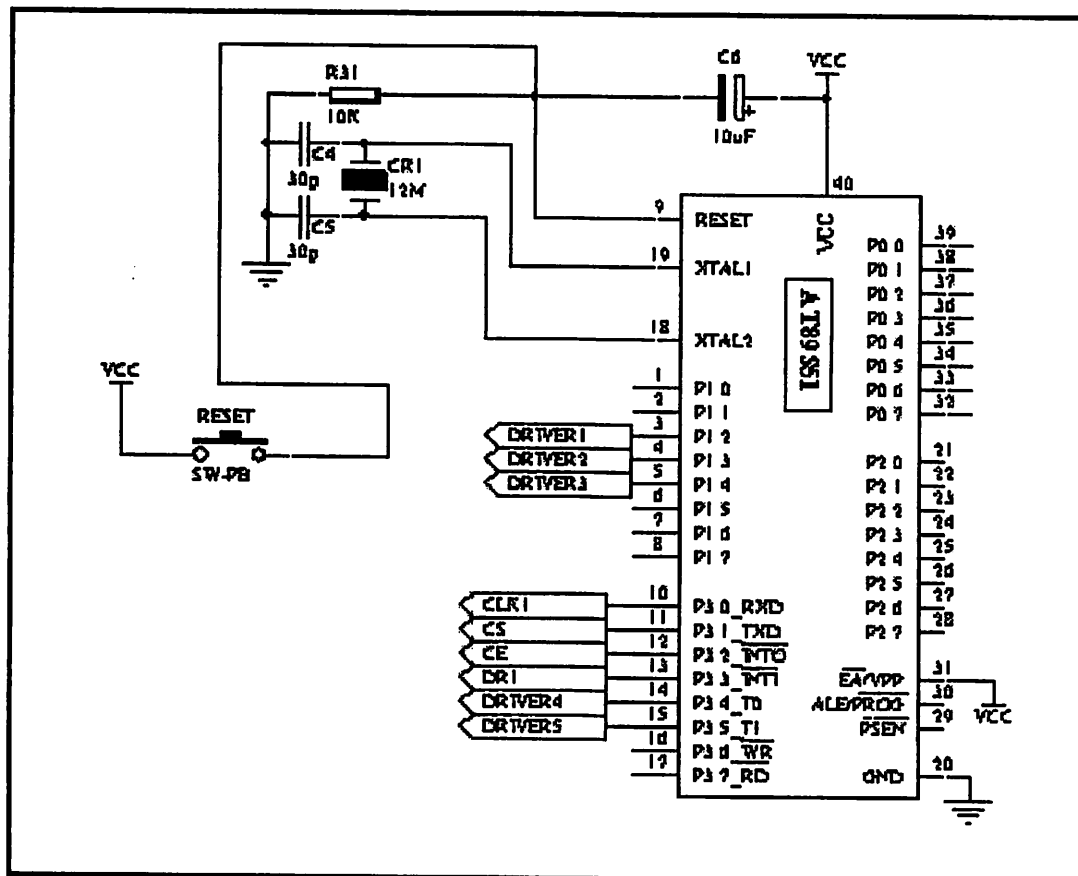


Gambar 3.4. Hubungan *Pin* Pada AT89S51 (pada rangkaian transmiter)

3.3.5. Hubungan *Pin* Pada AT89S51 (Rangkaian Receiver)

Diskripsi konfigurasi *Pin-Pin* AT89S51

- *Pin* 9 : dihubungkan Push-button Reset
- *Pin* 10-14 : dihubungkan TRW 24-G
- *Pin* 18-19 : dihubungkan X-tal 12M
- *Pin* 20 : dihubungkan ke ground
- *Pin* 32-39 : dihubungkan ke LCD
- *Pin* 31 dan 40 : dihubungkan ke VCC



Gambar 3.5. Hubungan *Pin* Pada AT89S51 (pada rangkaian receiver)

3.4. Perancangan Rangkaian TRW 24-G

TRW-2.4G merupakan modul RF dengan modulasi GFSK dan memiliki 2 mode transmisi data yaitu direct dan shock burst, Dalam aplikasi ini, TRW-2.4G digunakan dalam mode shock burst agar dapat menggunakan mikrokontroler yang murah yaitu AT89S51.

Aplikasi ini membutuhkan modul / komponen berikut:

- 2 bh AT89S51
- 2 bh TRW-2.4G,
- 2 bh 74HC125,
- 4 bh transistor 2N3904,
- 6 bh resistor 10K ohm ¼ W.

TRW-2.4G memiliki spesifikasi berikut:

- Frekuensi kerja : 2,4 – 2,524 GHz ISM band.
- Tipe modulasi : GFSK.
- Catu daya : 1,9 – 3,6 V.
- RF data rate : 1 Mbps atau 250 Kbps.
- Multi channel, hingga 125 channel.
- Dual receiver.
- Jangkauan : 280m @ 250Kbps, 150m @ 1Mbps.
- Built-in antenna.

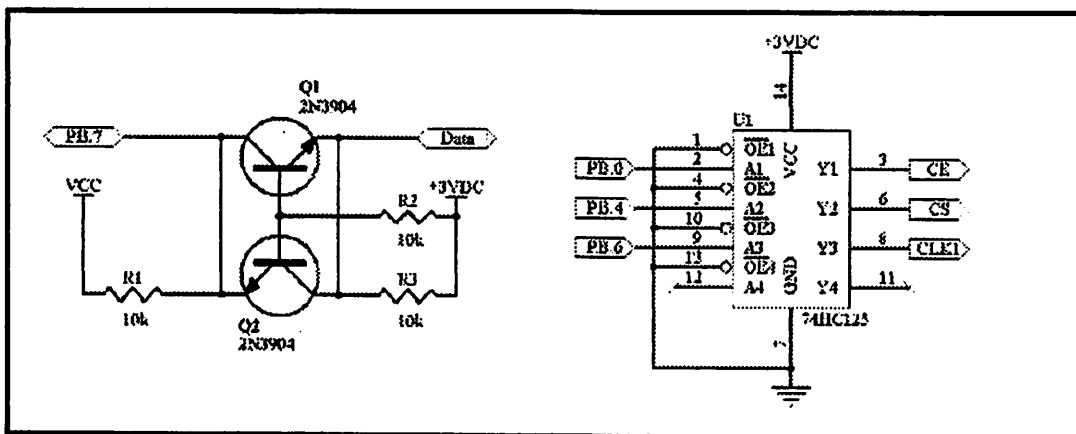
Hubungan antara modul-modul tersebut dengan mikrokontroler AT89S51 adalah sebagai berikut:

CLK1, dihubungkan P3.0 melalui rangkaian level converter

CS, dihubungkan P3.1 melalui rangkaian level converter

CE, dihubungkan P3.2 melalui rangkaian level converter

TRW-2.4G menggunakan catu daya +3 VDC dan memiliki I/O yang bekerja pada level tegangan 3 V sehingga beberapa hubungan diatas harus melalui rangkaian level converter seperti pada Gambar berikut.



Gambar 3.6. Rangkaian *Level Converter*

3.5. Perancangan Rangkaian LCD

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 L1632 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.

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 lainnya **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 bisa dijabarkan sebagai berikut :

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.

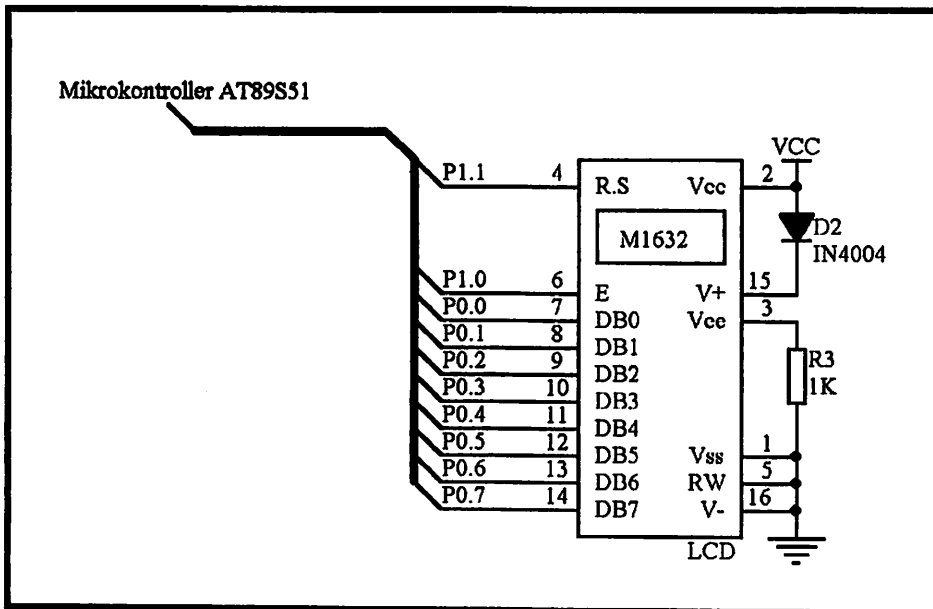
3. Untuk mengambil data dari M1632 sinyal **R/W** di-satu-kan, menyusul sinyal **E** di-satu-kan. Pada saat **E** menjadi 1, M1632 akan meletakkan datanya di **DB0 .. DB7**, data ini harus diambil sebelum sinyal **E** di-nol-kan kembali.

M1632 mempunyai seperangkat perintah untuk mengatur tata kerjanya, perangkat perintah tersebut meliputi perintah untuk menghapus tampilan, meletakkan kembali cursor pada baris huruf 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.



Gambar 3.7. Rangkaian LCD Pada Mikrokontroller

Diskripsi konfigurasi *Pin-Pin* pada LCD:

- *Pin* 1,3,5 dan 16 : dihubungkan dengan ground
- *Pin* 2 : dihubungkan dengan VCC
- *Pin* 4 : dihubungkan dengan P1.0 AT89S51
- *Pin* 6 : dihubungkan dengan P1.1 AT89S51
- *Pin* 7-14 : dihubungkan dengan AT89S51 (P0.0-P0.7)

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 AT89S51.

Kemudian untuk RS dihubungkan pada *Port 1.1*, tulis/baca (*Read/Write*) diberikan logika *low* karena disini LCD bersifat menulis data, dan yang terakhir *Enable (E)* dikendalikan dengan *Port 1.0*.

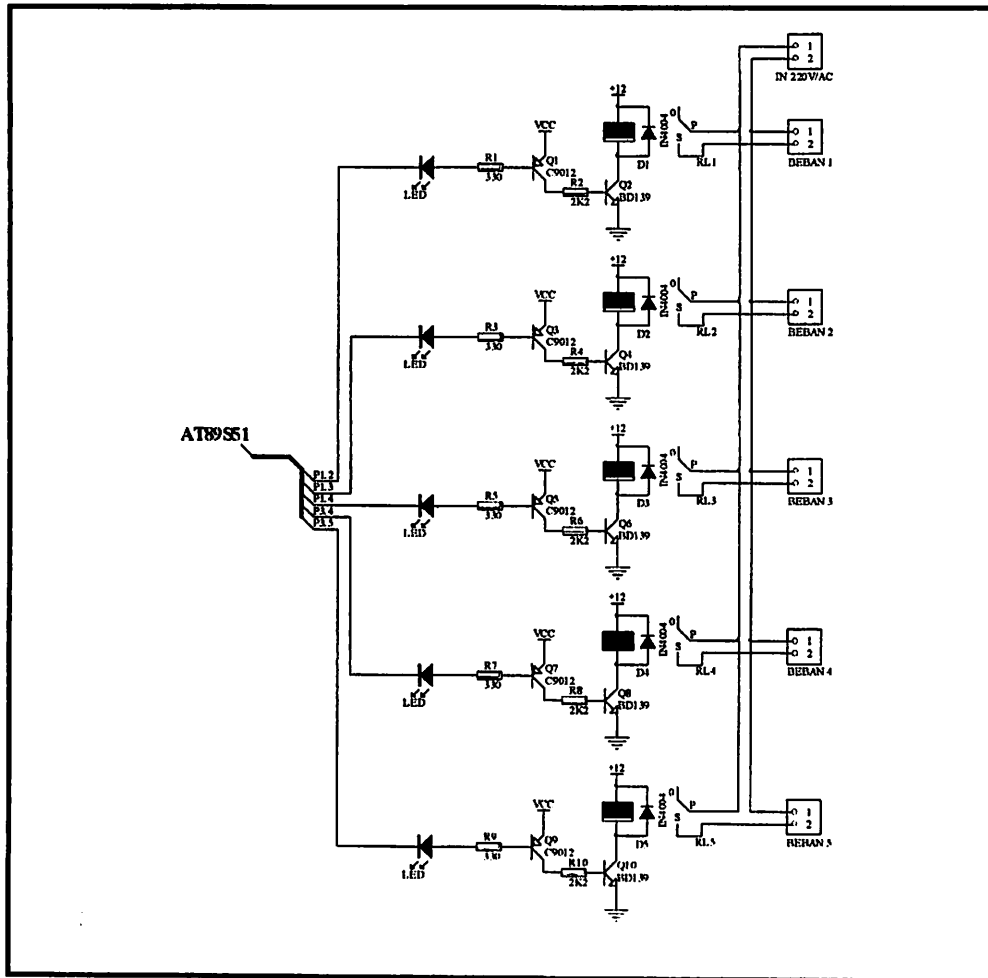
3.6. Perancangan Rangkaian *Driver Relay*

Relay merupakan salah satu komponen *output* yang paling sering digunakan baik pada industri, otomotif, ataupun peralatan elektronika lainnya. *Relay* berfungsi untuk menghubungkan atau memutus aliran arus listrik yang dikontrol dengan memberikan tegangan dan arus tertentu pada koilnya.

Ada 2 macam *relay* berdasarkan tegangan untuk menggerakkan koilnya yaitu AC dan DC. Pada perancangan ini digunakan *relay* DC dengan tegangan koil 12V DC, arus yang diperlukan sekitar 20-30mA. Karena itu kita tidak bisa langsung menghubungkan *output* dari mikrokontroler dengan *relay* karena arusnya tidak cukup besar.

Karena itu perlu digunakan *driver* untuk penguat arus yang biasanya berupa transistor, yang pada perancangan ini digunakan Transistor NPN BD139 sehingga mempunyai penguatan arus yang besar. Transistor PNP C9012 digunakan untuk mengontrol keaktifan BD139 melalui mikrokontroler AT89S51.

Tiap-tiap rangkaian driver pada alat ini dilengkapi dengan '*supression diode*'. Diode ini berfungsi untuk mencegah '*kickback*' yaitu *transient* yang terjadi pada koil *relay* (beban induktif) saat *relay* dimatikan .



Gambar 3.8. Rangkaian *Driver Relay*

Berikut adalah perhitungan perancangan pada rangkaian driver relay.

Diketahui :

$$R_{relay} = 400\Omega$$

$$V_{relay} = 12V$$

Q1 adalah transistor PNP C9012 dengan h_{FE} 125

Q2 adalah transistor NPN BD139 dengan h_{FE} 100

Perhitungan pada Q2:

$$\begin{aligned}
 I_{C_{sat}} &= \frac{V_{relay}}{R_{relay}} \\
 &= \frac{12}{400} \\
 &= 30mA
 \end{aligned}$$

$$\begin{aligned}
 I_{B_{sat}} &= \frac{I_{C_{sat}}}{h_{FE}} \\
 &= \frac{30mA}{100} \\
 &= 0,3 \text{ mA}
 \end{aligned}$$

$$\begin{aligned}
 R_b &= \frac{V_b - V_{be}}{I_b} \\
 &= \frac{5V - 0,7V}{0,3mA} \\
 &= 14K3
 \end{aligned}$$

Agar relai dapat bekerja dengan baik, perbedaan antara I_b dan $I_{B_{sat}}$ harus lebih besar dari 1 ($I_b - I_{B_{sat}} > 0,1$). Oleh karena itu pada perancangan ini besar R_b pada Q2 yang digunakan adalah sebesar 2K2. Sehingga besar arus I_b menjadi:

$$\begin{aligned}
 I_b &= \frac{V_b - V_{be}}{R_b} \\
 &= \frac{5 - 0,7}{2K2} \\
 &= 1,954 \text{ mA}
 \end{aligned}$$

Perhitungan pada Q1:

$$I_{c_{sat}} = I_{b_{Q2}}$$

$$= 1,954 \text{ mA}$$

$$I_{b_{sat}} = \frac{I_c}{h_{FE}}$$

$$= \frac{1,954}{125}$$

$$= 0,015 \text{ mA}$$

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

$$= \frac{5 - 0,7}{0,015}$$

$$= 286K6$$

Agar transistor Q1 dapat bekerja dengan baik dalam mensaklar transistor Q2, maka perbedaan antara I_b dan $I_{b_{sat}}$ harus lebih besar dari 1 ($I_b - I_{b_{sat}} > 0,1$), dan selain itu arus I_b pada Q1 supaya dapat digunakan untuk menyalakan LED sebagai lampu indikator bahwa rangkaian driver yang dirancang dapat bekerja dengan baik, maka pada perancangan ini besar R_b pada Q1 yang digunakan adalah sebesar 330Ω .

BAB IV

PENGUJIAN ALAT

Setelah melakukan perancangan atau pembuatan sistem kontrol ini, maka kita perlu melakukan suatu pengujian sistem. Yang mana pengujian sistem ini bertujuan antara lain :

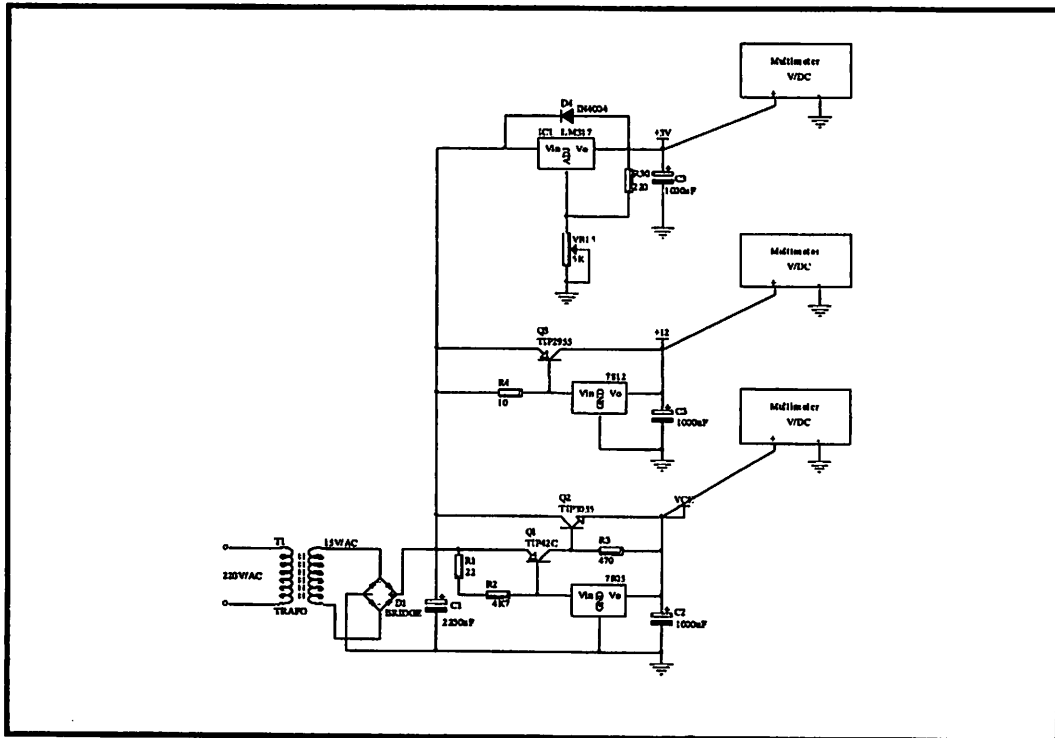
1. Mengetahui sejauh mana Aplikasi Mikrokontroler Sebagai Pusat Kendali Peralatan Listrik Tanpa Kabel (*wireless*) ini berfungsi sebagaimana yang kita harapkan.
2. Mencari dan menemukan berbagai kendala yang mungkin timbul pada saat Aplikasi Mikrokontroler Sebagai Pusat Kendali Peralatan Listrik Tanpa Kabel (*wireless*) beroperasi untuk kemudian diperbaiki sampai pada tingkat kesalahan sistem yang sekecil mungkin sehingga didapatkan hasil yang sebaik-baiknya.

4.1. Pengujian Rangkaian *Power Supply*

Pengujian ini bertujuan untuk mengetahui tegangan yang dikeluarkan oleh rangkaian catu daya yang telah dibuat. Dengan begitu dapat diketahui apakah terjadi kesalahan terhadap rangkaian atau tidak. Tegangan yang dibutuhkan untuk memberikan tegangan kerja keseluruhan rangkaian adalah +3V, +5V, +12V.

Untuk mengukur besarnya tegangan keluaran pada rangkaian *power supply* maka pada pengukuran ini menggunakan Multimeter Digital. Berikut cara pengukuran untuk pengambilan data besarnya tegangan pada rangkaian *power*

supply yang dilakukan satu persatu pada *output regulator* tegangan LM7805, LM7812 dan LM317.



Gambar 4.1. Cara pengukuran tegangan pada rangkaian catu daya

Untuk mengukur besarnya tegangan keluaran rangkaian *power supply* maka tegangan masukan pada rangkaian *power supply* juga harus tersambung dengan sumber tegangannya yaitu tegangan jala-jala dari PLN 220 V/AC. Multimeter digital di-seting pada *DC Volt* dengan batas maksimal pengukuran yaitu 20 *Volt DC*. Pengukuran tegangan dilakukan pada *output* keluaran dari *regulator* tegangan LM7805, LM7812 dan LM317.

. Dari hasil pengukuran pada rangkaian *power supply* diperoleh data seperti pada tabel berikut ini.

Tabel 4-1. Hasil Pengukuran Rangkaian *Power Supply*

<i>Regulator tegangan</i>	<i>Vout (V)</i>
LM7805	+4,9
LM7812	+11,9
LM317	3,3V

Dari hasil pengujian yang diperoleh dapat diketahui bahwa *power supply* ini dapat memberikan tegangan kerja untuk peralatan kontrol maupun peralatan *sensor* sesuai dengan tegangan kerja yang dibutuhkan. Hasil yang di peroleh dari pengukuran tersebut, dikarenakan oleh adanya beberapa faktor yang mempengaruhi, diantaranya kualitas dari tiap-tiap komponen yang digunakan nilainya tidak murni. Selain itu, tegangan jala-jala listrik yang digunakan tidak stabil.

4.2. Pengujian Rangkaian TRW 2.4G

RW-2.4G merupakan modul RF dengan modulasi GFSK dan memiliki 2 mode transmisi data yaitu direct dan shock burst, menurut vendor-nya daya jangkau modul ini mencapai 280 meter pada kondisi open area. Dalam aplikasi ini, TRW-2.4G digunakan dalam mode shock burst agar dapat menggunakan mikrokontroler yang murah yaitu mikrokontroler AT89S51.

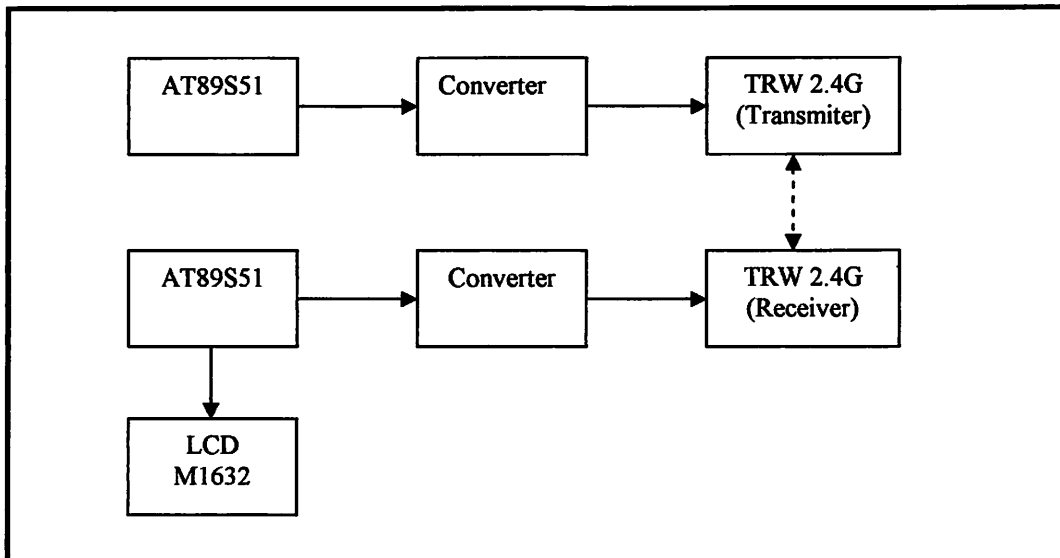
Pengujian ini membutuhkan modul / komponen berikut:

- 2 bh mikrokontroler AT89S51
- 2 bh TRW-2.4G,
- 2 bh 74HC125,
- 4 bh transistor 2N3904,
- 6 bh resistor 10K ohm ¼ W.
- 1 bh LCD M1632

TRW-2.4G memiliki spesifikasi berikut:

- Frekuensi kerja : 2,4 – 2,524 GHz ISM band.
- Tipe modulasi : GFSK.
- Catu daya : 1,9 – 3,6 V.
- RF data rate : 1 Mbps atau 250 Kbps.
- Multi channel, hingga 125 channel.
- Dual receiver.
- Jangkauan : 280m @ 250Kbps, 150m @ 1Mbps.
- Built-in antenna.

pertama kolom pertama. Pengujian ini dilakukan agar di ketahui berapa jarak terjauh yang dapat di capai dari trasmitter ke rangkaian receiver



Gambar 4.2. Diagram Blok Pengujian TRW 2.4G

Jarak	transmitter	receiver	Tidak ada rintangan	Ada rintangan
1 meter	mengirim	diterima	menyala	menyala
3 meter	mengirim	diterima	menyala	menyala
5 meter	mengirim	diterima	menyala	menyala
7 meter	mengirim	diterima	menyala	menyala
9 meter	mengirim	diterima	menyala	menyala
11 meter	mengirim	diterima	menyala	-
13 meter	mengirim	-	-	-

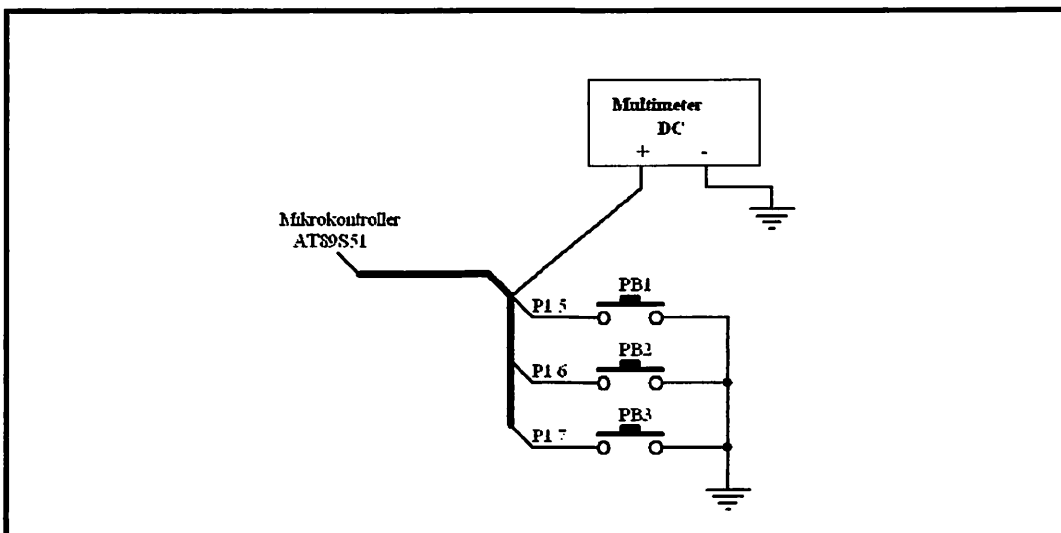
Tabel 4-2 Pengujian jarak rangkaian TRW2,4GHz

4.3. Pengujian Terhadap Rangkaian *Push-button*

Rangkaian *push-button* berfungsi sebagai inputan mikrokontroler pada rangkaian transmitter untuk men-setting ON/OFF peralatan listrik yang dikontrol

oleh rangkaian receiver. *Push-button* diberikan inputan tegangan sebesar 4,9 Volt dan dirangkai dengan menggunakan *common ground*. Sehingga apabila terjadi penekanan pada tombol *push-button*, maka akan terjadi arus hubung singkat yang menyebabkan kondisi tegangan pada pin *push-button* yang terhubung dengan mikrokontroler berubah dari kondisi “*high*” 4,9 Volt menjadi kondisi “*low*” 0 Volt.

Pengujian rangkaian *push-button* dilakukan dengan cara mengukur tegangan pada *pin* mikrokontroler yang terhubung dengan rangkaian *push-button* dengan menggunakan multimeter DC dengan batas 20V/DC. Kondisi tegangan pada pin mikrokontroler yang terhubung dengan rangkaian *push-button* sebelum adanya penekanan pada tombol *push-button* adalah 4,9V. Berikut cara pengukuran tegangan pada rangkaian *push-button*.



Gambar 4.3. Cara Pengukuran Tegangan Pada Pengujian Rangkaian *Push-Button*

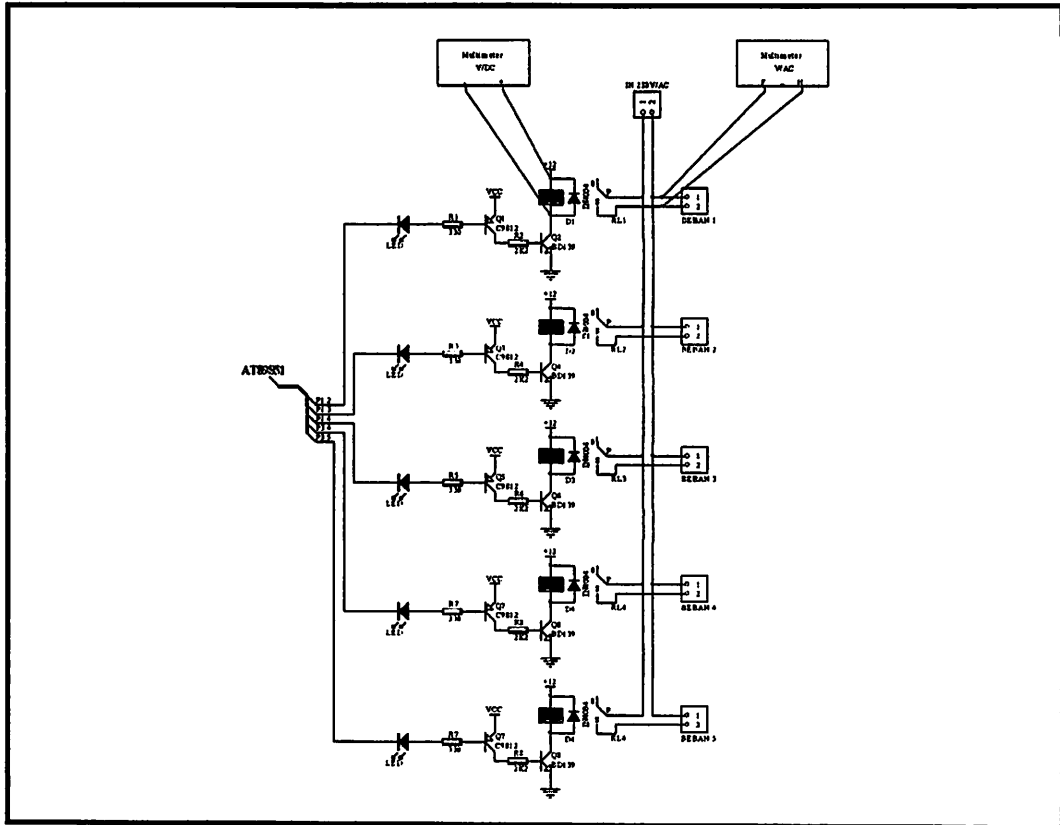
Tabel 4-2. Hasil Pengukuran Pada Rangkaian *Push-Button*

<i>Push-Button</i> (Tertekan)	P1.5 (V)	P1.6 (V)	P1.7 (V)
PB 1 (<i>Menu</i>)	0	4,9	4,9
PB 2 (<i>Up</i>)	4,9	0	4,9
PB 3 (<i>Down</i>)	4,9	4,9	0

4.4. Pengujian Rangkaian *Driver Relay*

Rangkaian *driver relay* berfungsi sebagai penguat arus yang dibutuhkan oleh *relay* agar dapat menghubungkan dan memutuskan aliran arus listrik dengan tegangan AC220V ke beban yang dikontrol. *Relay* yang digunakan adalah *relay DC* dengan tegangan koil 12V *DC*. Pengujian rangkaian *driver relay* secara *hardware* dilakukan dengan cara memberikan suatu kondisi 'low' (0 Volt) dan 'high' (5 Volt) pada *input driver relay*, kemudian mengukur tegangan pada tiap-tiap koil *relay* dan tegangan pada *output* rangkaian *driver relay* (pada tiap-tiap beban).

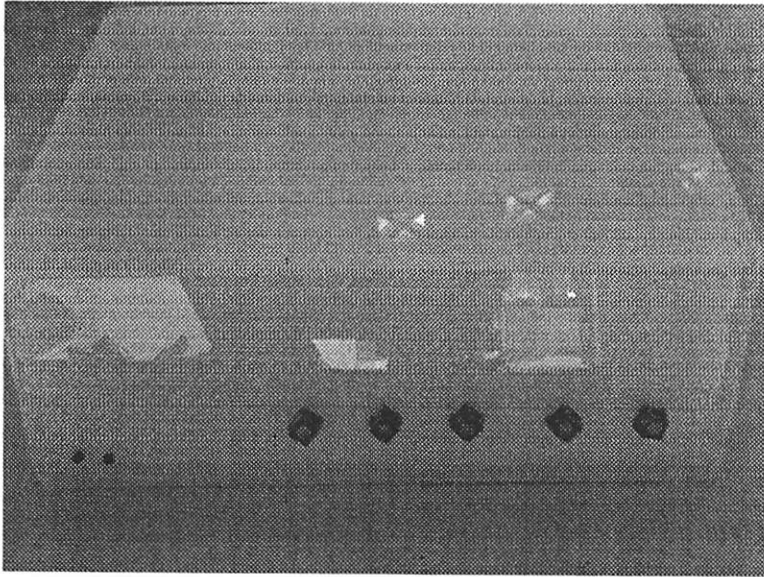
Dengan mengacu pada gambar diatas, pengujian rangkaian *driver relay* dilakukan dengan mengukur tegangan pada tiap-tiap koil *relay* dengan mengatur multimeter pada *DC* dengan batas 20 Volt dan kemudian mengukur tegangan pada tiap-tiap beban dengan mengatur multimeter pada *AC* dengan batas maksimal 250V.



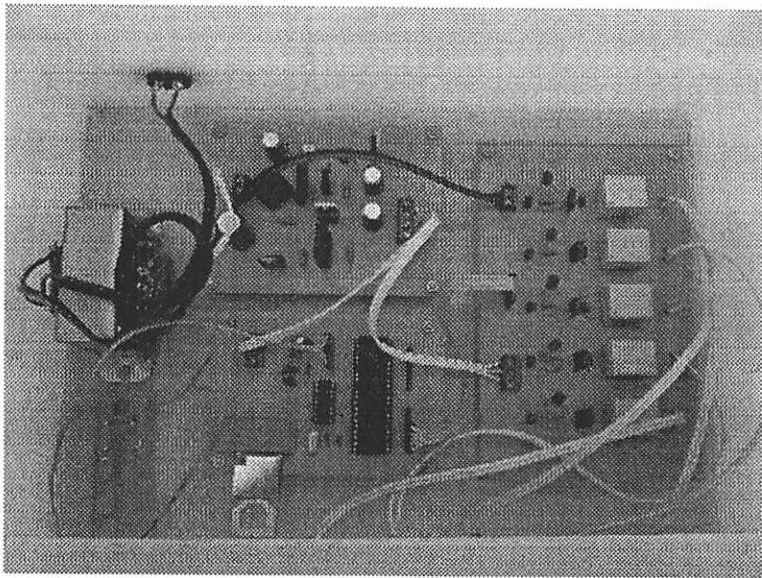
Gambar 4.4. Pengujian Rangkaian *Driver Relay*

Tabel 4-3. Hasil pengujian pada Rangkaian *Driver Relay*

Input Driver Relay (Volt/DC)					Tegangan pd Koil Relay (Volt/DC)					Tegangan pd Beban (Volt/AC)				
PI.2	PI.3	PI.4	P3.4	P3.5	RL1	RL2	RL3	RL4	RL5	B1	B2	B3	B4	B5
0	4,9	5,0	4,9	4,9	11,9	-	-	-	-	219	-	-	-	-
4,8	0	4,9	0	4,8	-	11,9	-	-	-	-	219	-	-	-
4,9	4,9	0	4,9	0	-	-	11,9	-	-	-	-	221	-	-
5,0	4,8	4,9	0	5,0	-	-	-	11,9	-	-	-	-	221	-
4,9	4,9	4,8	5,0	0	-	-	-	-	11,9	-	-	-	-	219



Gambar 4.5. Foto Alat Tampak Atas



Gambar 4.6. Foto Alat Tampak Dalam

BAB V

PENUTUP

5.1. Kesimpulan

Pada bab ini akan dibahas mengenai kesimpulan dari hasil laporan Tugas Akhir ini. Kesimpulan yang dibuat tentu saja berdasarkan dari hasil perencanaan dan pembuatan alat seperti yang dibahas pada bab-bab sebelumnya. Kesimpulan yang didapat adalah sebagai berikut :

1. Jarak yang dapat dicapai maksimal alat ini adalah 11 meter dengan tidak ada rintangan, dan jarak yang dapat di capai jika ada rintangan adalah 9 meter.
2. Alat hanya bisa bekerja dengan daya maksimal 450w
3. Alat ini dapat bekerja pada frekuensi 2,4GHz

5.2. Saran

1. Bila diinginkan untuk penggunaan beban yang lebih besar dari 2A, maka perlu *menggunakan relay* yang mempunyai spesifikasi kemampuan yang lebih besar pula.
2. Diharapkan alat ini dapat dikembangkan dengan menambahkan *output-an* yang lebih banyak

DAFTAR PUSTAKA

1. Andi Nalwan, Paulus. 2003. *Panduan Praktis Teknik Antarmuka dan Pemrograman Mikrokontroler 89C51*. Jakarta: PT Elex Media Komputindo.
2. Budiharto, Widodo. 2007. *12 Proyek Mikrokontroler untuk Pemula*. Jakarta: PT Elex Media Komputindo.
3. Eko Putra, Agfianto. 2004. *Belajar Mikrokontroler AT89C51/52/55 (Teori dan Aplikasi)*. Yogyakarta: Penerbit Gava Media.
4. www.atmel.com
5. www.beyondlogic.com
6. www.delta-electronic.com
7. www.electroniclab.com
8. www.fairchildsemi.com
9. www.joker-robotics.com
10. www.laipac.com
11. www.parallax.com
12. [www.seiko instruments inc.com](http://www.seiko-instruments-inc.com)
13. www.wenshing.com



INSTITUT TEKNOLOGI NASIONAL
Jl. Karanglo KM 2
Malang

LEMBAR ASISTENSI TUGAS AKHIR

Nama : Hidayat Surya Dilaga
N. I. M : 02.52.011
Program Studi : Teknik Elektro ST D-III
Konsentrasi : Energi Listrik
Judul : PERANCANGAN KENDALI PERALATAN
LISTRIK TANPA KABEL DENGAN MK AT89S51

NO	TANGGAL	BIMBINGAN	PARAF
1.	24-8-2007	BAB I revisi	b
2.	24-8-2007	BAB II	b
3.	10-9-2007	BAB III	b
4.	10-9-2007	BAB IV revisi dibetulkan	b
5.	10-9-2007	BAB V	b
6.	18-9-2007	Acc moju	b

Dosen Pembimbing

(Bambang Priyo H,ST.MT)
NIP.Y : 1028400082



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO D-III

BERITA ACARA UJIAN TUGAS AKHIR

Nama Mahasiswa : Hidayat Surya Dilaga
N . I . M : 02.52. 011
Jurusan : Teknik Elektro D-III
Konsentrasi : Energi Listrik
Judul Tugas Akhir : PERANCANGAN DAN PEMBUATAN KENDALI
PERALATAN LISTRIK TANPA KABEL DENGAN
MK AT89S51

Dipertahankan dihadapan tim penguji Tugas Akhir

Jenjang program Diploma III, pada :

Hari : Senin
Tanggal : 23 September 2007
Dengan Nilai : 75,45 (B+) *6*

Panitia Ujian Tugas Akhir



(Ir. Mochtar Asroni, MSME)

Sekretaris

(Ir. Choirul Saleh, MT)

Anggota Penguji

Penguji I

(Ir. Yusuf Ismail Nakhoda, MT)

Penguji II

(Ir. Taufik Hidayat, MT)



INSTITUT TEKNOLOGI NASIONAL
Jl. Karanglo KM 2
Malang

LEMBAR PERBAIKAN TUGAS AKHIR

Nama : Hidayat Surya Dilaga
N . I . M : 02.52. 011
Program Studi : Teknik Elektro ST D-III
Konsentrasi : Energi Listrik
Hari/tanggal : Senin 23 september 2007
Judul : PERANCANGAN DAN PEMBUATAN KENDALI PERALA

TAN LISTRIK TANPA KABEL DENGAN MK AT89S51

NO	MATERI PERBAIKAN	PARAF
1	Pengujian alat(jarak maksimum alat dapat dikendalikan)apakah sesuai dengan perancangan	
2	Kesimpulan diambil dari hasil pengujian alat	
3	Saran judul ditambah "pembuatan"	
4	Revisi kesimpulan,munculkan angka-angka dari hasil pengujian.	
5	Berapa jarak terjauh dari kemampuan alat baik ada rintangan maupun tidak.	
6	Berapa kapasitas maksimal memikul beban	
7	Ditambahkan dibatasan masalah hanya untuk lima sumber sumber saja.	
8	Alat ini bias digunakan apa saja	
9	Alat ini bekerja pada frekuensi berapa	

Disen penguji I

(Ir. Yusuf Ismail Nakhoda,MP)

Disen penguji II

(Ir. Taufik Hidayat,MT)

Dosen Pembimbing

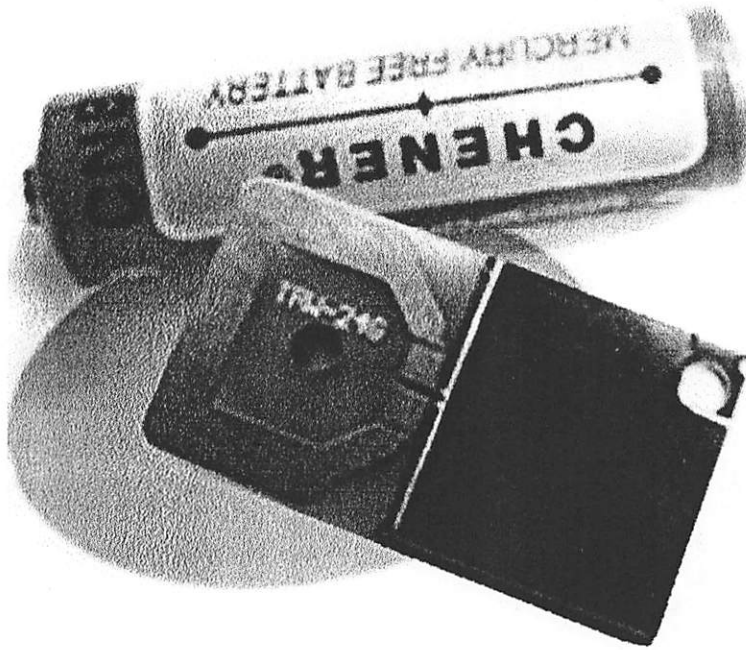
(Bambang Priyo H,ST.MT)

LAMPIRAN

WENSHING®

TRW-24G

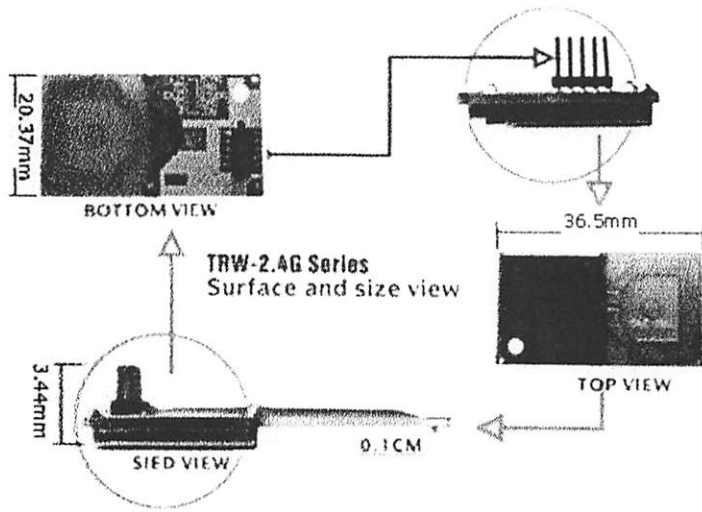
TRW-24G High Frequency Transceiver Module(GFSK)



◆Specification :

- * Frequency Range: 2.4~2.524 GHz
- * Modulate Mode: GFSK
- * Work Voltage: 3V
- * Channel: 83
- * Output Power: +4dBm
- * Data Rate: 1Mbps; 250Kbps
- * Operating Temperature: -40~+85 Centigrade
- * The longest range : 280m (250Kbps); 150m (1Mbps)
- * No dead spaces in reception.
- * Built in antenna.
- * Competitive price.
- * Apply for various type of products: Wireless Joysticks, Wireless Speaker, Wireless Earphone , Wireless Cell phone , Wireless Intercom , Wireless Mouse, Wireless Keyboard and Data Communication.....

◆Graph:



Conditions: VDD=+3V, VSS=0V, T_A=-40 centigrade to +85 centigrade

Symbol	Parameter(condition)	Notes	Min.	Typ.	Max.	Units
VDD	Supply voltage		1.9	3.0	3.6	V
TEMP	Operating temperature		-40	+27	+85	Centigrade
f _{op}	Operating frequency	1)	2400		2524	MHz
R _{GFSK}	Data rate direct mode	3)	250		1000	kbps
F _{CHANNEL}	Channel spacing			1		MHz
I _{VDD}	Supply current one channel 250kbps			18		mA
I _{VDD}	Supply current one channel 1000kbps			19		mA
I _{VDD}	Supply current two channels 250kbps			23		mA
I _{VDD}	Supply current two channels 1000kbps			25		mA
RX _{SENS}	Sensitivity at 0.1%BER(@250kbps)			-90		dB
RX _{SENS}	Sensitivity at 0.1%BER(@1000kbps)			-80		dB

Part No.	Description
TRW-24G	High frequency Transceiver module (GFSK) 2.4GHz



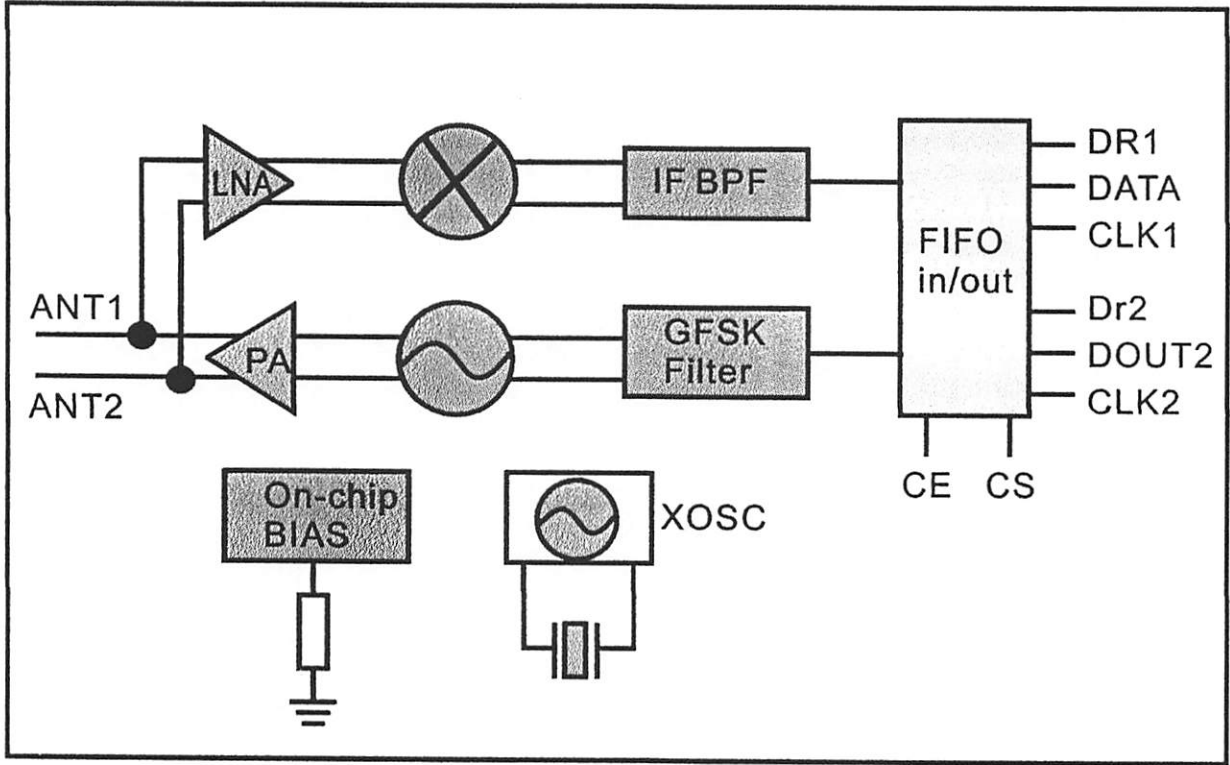
WENSHING

TRW -2.4GHz Radio Transceiver

Conditions: VDD = +3V, VSS = 0V, TA = -40°C to +85°C

Symbol	Parameter (condition)	Notes	Min.	Typ.	Max.	Units
Operating conditions						
VDD	Supply voltage		1.9	3.0	3.6	V
TEMP	Operating Temperature		-40	+27	+85	°C
Digital input pin						
V _{HI}	HIGH level input voltage		VDD- 0.3		VDD	V
V _{LI}	LOW level input voltage		V _{SS}		0.3	V
Digital output pin						
V _{OHI}	HIGH level output voltage (I _{OL} =-0.5mA)		VDD- 0.3		VDD	V
V _{OLI}	LOW level output voltage (I _{OL} =-0.5mA)		V _{SS}		0.3	V
General RF conditions						
f _{OP}	Operating frequency	1)	2400		2524	MHz
Δf	Frequency deviation			±156		kHz
R _{DATA}	Data rate ShockBurst™		>0		1000	kbps
F _{CHANNEL}	Channel spacing			1		MHz
Transmitter operation						
P _{RF}	Maximum Output Power	4)		0	+4	dBm
P _{RF C}	RF Power Control Range		16	20		dB
P _{RF CR}	RF Power Control Range Resolution				±3	dB
P _{BW}	20dB Bandwidth for Modulated Carrier				1000	kHz
P _{RF 2}	2 nd Adjacent Channel Transmit Power 2MHz				-20	dBm
P _{RF 3}	3 rd Adjacent Channel Transmit Power 3MHz				-40	dBm
I _{VDD}	Supply current @ 0dBm output power	5)		13		mA
I _{VDD}	Supply current @ -20dBm output power	5)		8.8		mA
I _{VDD}	Average Supply current @ -5dBm output power, Shock Burst™	6)		0.8		mA
I _{VDD}	Average Supply current in stand-by mode	7)		12		μA
I _{VDD}	Average Supply current in power down			1		μA
Receiver operation						
I _{VDD}	Supply current one channel 250kbps			18		mA
I _{VDD}	Supply current one channel 1000kbps			19		mA
I _{VDD}	Supply current two channels 250kbps			23		mA
I _{VDD}	Supply current two channels 1000kbps			25		mA
RX _{SENS}	Sensitivity at 0.1%BER (@250kbps)			-90		dBm
RX _{SENS}	Sensitivity at 0.1%BER (@1000kbps)			-80		dBm
C/I _{CO}	C/I Co-channel			6		dB
C/I _{1st}	1 st Adjacent Channel Selectivity C/I 1MHz			-1		dB
C/I _{2nd}	2 nd Adjacent Channel Selectivity C/I 2MHz			-16		dB
C/I _{3rd}	3 rd Adjacent Channel Selectivity C/I 3MHz			-26		dB
RX _U	Blocking Data Channel 2			-41		dB

Circuit Description



ShockBurst™

The ShockBurst™ technology uses on-chip FIFO to clock in data at a low data rate and transmit at a very high rate thus enabling extremely power reduction.

When operating the TRW-2.4G in ShockBurst™, you gain access to the high data rates (1 Mbps) offered by the 2.4 GHz band without the need of a costly, high-speed micro controller (MCU) for data processing.

By putting all high speed signal processing related to RF protocol on-chip, the TRW-2.4G offers the following benefits:

- Highly reduced current consumption
- Lower system cost (facilitates use of less expensive micro controller)
- Greatly reduced risk of 'on-air' collisions due to short transmission time

The TRW-2.4G can be programmed using a simple 3-wire interface where the data rate is decided by the speed of the micro controller.

By allowing the digital part of the application to run at low speed while maximizing the data rate on the RF link, the nRF ShockBurst™ mode reduces the average current consumption in applications considerably.

ShockBurst™ principle

When the TRW-2.4G is configured in ShockBurst™, TX or RX operation is conducted in the following way (10 kbps for the example only).

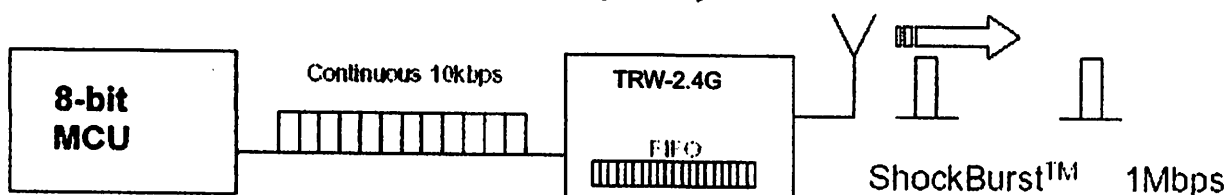


Figure 4 Clocking in data with MCU and sending with ShockBurst™ technology

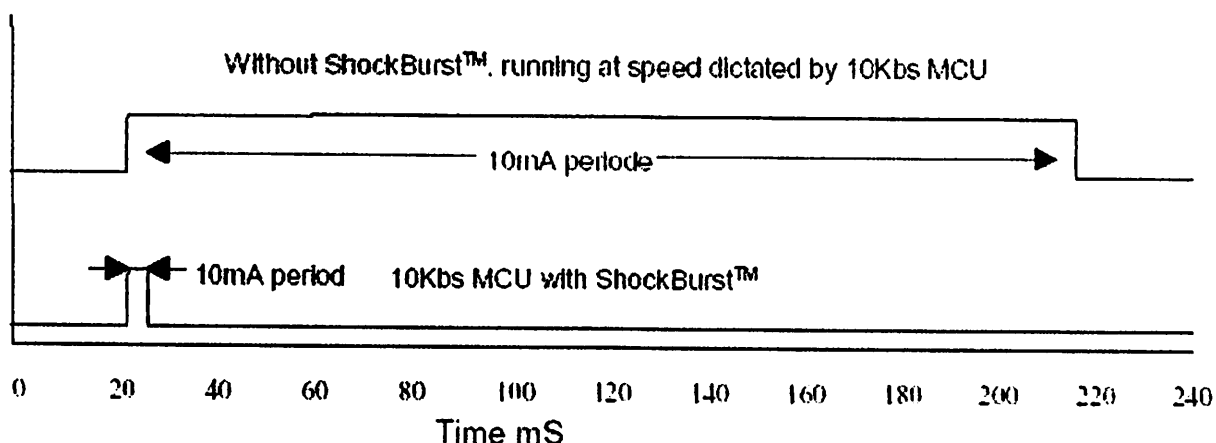


Figure 1 Current consumption with & without ShockBurst™ technology

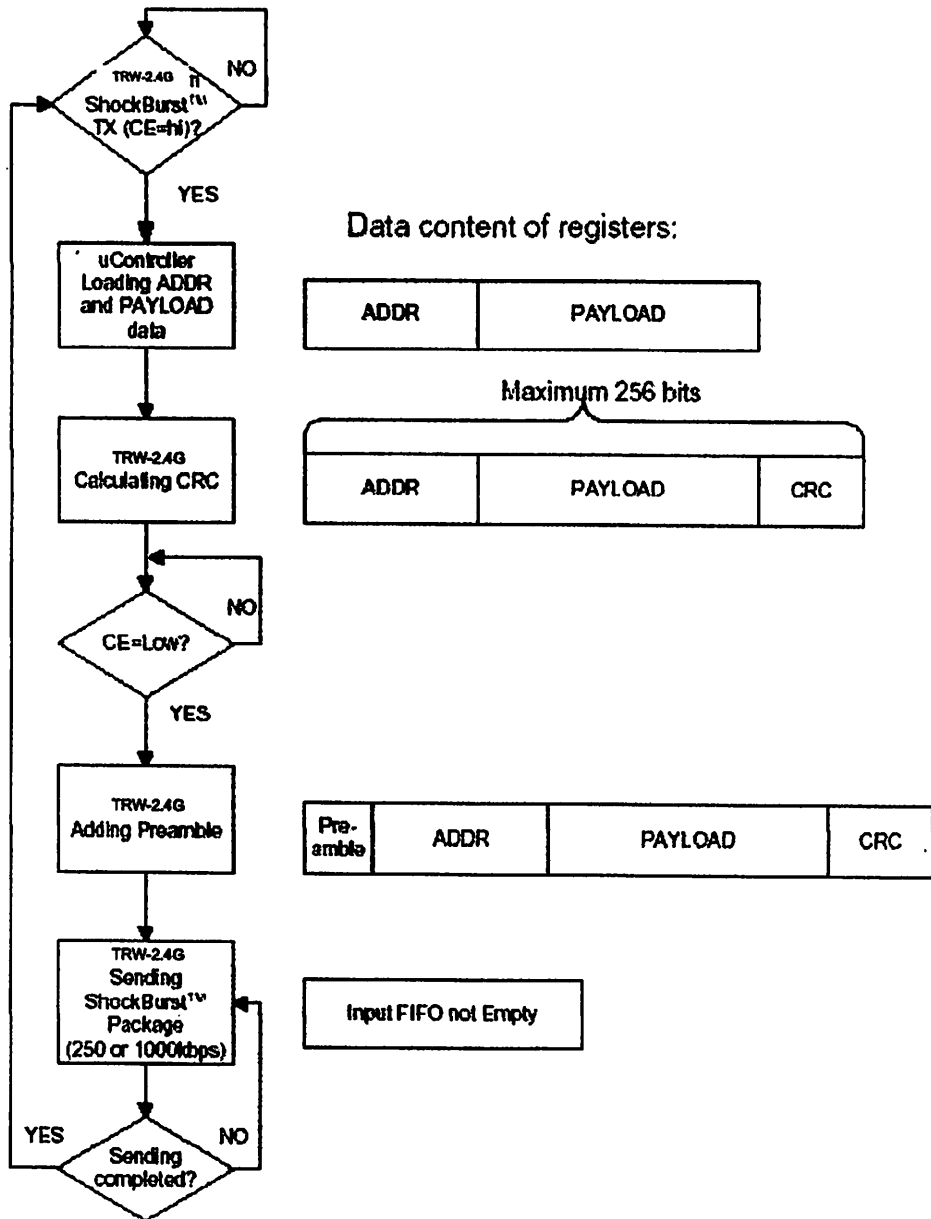


Figure 2 Flow Chart ShockBurst™ Transmit of TRW-2.4G

nRF2401 ShockBurst™ Transmit:

MCU interface pins: CE, CLK1, DATA

1. When the application MCU has data to send, set CE high. This activates TRW-2.4G on-board data processing.
2. The address of the receiving node (RX address) and payload data is clocked into the TRW-2.4G. The application protocol or MCU sets the speed <1Mbps (ex: 10kbps).
3. MCU sets CE low, this activates a TRW-2.4G ShockBurst™ transmission.
4. TRW-2.4G ShockBurst™:
 - RF front end is powered up
 - RF package is completed (preamble added, CRC calculated)
 - Data is transmitted at high speed (250 kbps or 1 Mbps configured by user).
 - TRW-2.4G return to stand-by when finished

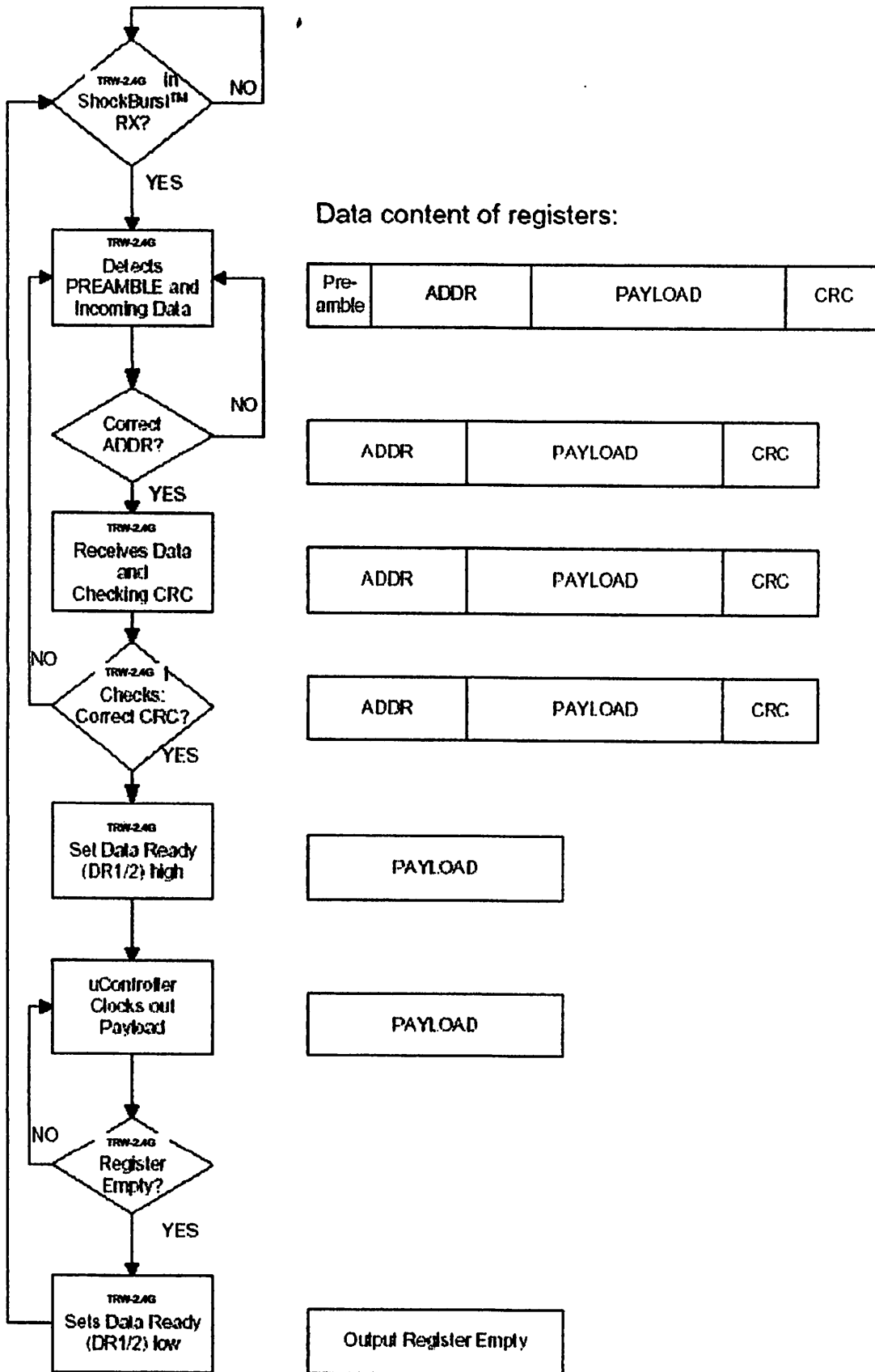


Figure 3 Flow Chart ShockBurst™ Receive of TRW-2.4G

TRW-2.4G ShockBurst™ Receive:

MCU interface pins: CE, DR1, CLK1 and DATA (one RX channel receive)

1. Correct address and size of payload of incoming RF packages are set when **TRW-2.4G** is configured to ShockBurst™ RX.
2. To activate RX, set CE high.
3. After 200 μ s settling, **TRW-2.4G** is monitoring the air for incoming communication.
4. When a valid package has been received (correct address and CRC found), **TRW-2.4G** removes the preamble, address and CRC bits.
5. **TRW-2.4G** then notifies (interrupts) the MCU by setting the DR1 pin high.
6. MCU may (or may not) set the CE low to disable the RF front end (low current mode).
7. The MCU will clock out just the payload data at a suitable rate (ex. 10 kbps).
8. When all payload data is retrieved **TRW-2.4G** sets DR1 low again, and is ready for new incoming data package if CE is kept high during data download. If the CE was set low, a new start up sequence can begin, see Figure 12

DuoCeiver™ Simultaneous Two Channel Receive Mode

In both ShockBurst™ modes the TRW-2.4G can facilitate simultaneous reception of two parallel independent frequency channels at the maximum data rate. This means:

- TRW-2.4G can receive data from two 1 Mbps transmitters (ex: TRW-2.4G or TRW-2.4G_) 8 MHz (8 frequency channels) apart through one antenna interface.
- The output from the two data channels is fed to two separate MCU interfaces.
 - Data channel 1: CLK1, DATA, and DR1
 - Data channel 2: CLK2, DOUT2, and DR2
 - DR1 and DR2 are available only in ShockBurst™.

The TRW-2.4G DuoCeiver™ technology provides 2 separate dedicated data channels for RX and replaces the need for two, stand alone receiver systems.

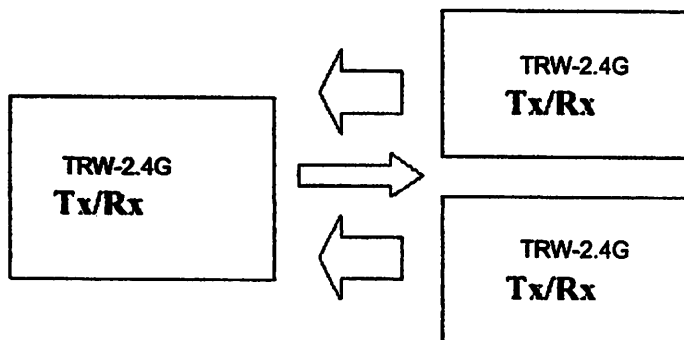


Figure 4 Simultaneous 2 channel receive on TRW-2.4G

There is one absolute requirement for using the second data channel. For the TRW-2.4G to be able to receive at the second data channel the frequency channel must be 8MHz higher than the frequency of data channel 1. The TRW-2.4G must be programmed to receive at the frequency of data channel 1. No time multiplexing is used in TRW-2.4G to fulfil this function. In direct mode the MCU must be able to handle two simultaneously incoming data packets if it is not multiplexing between the two data channels. In ShockBurst™ it is possible for the MCU to clock out one data channel at a time while data on the other data channel waits for MCU availability, without any lost data packets, and by doing so reduce the needed performance of the MCU.

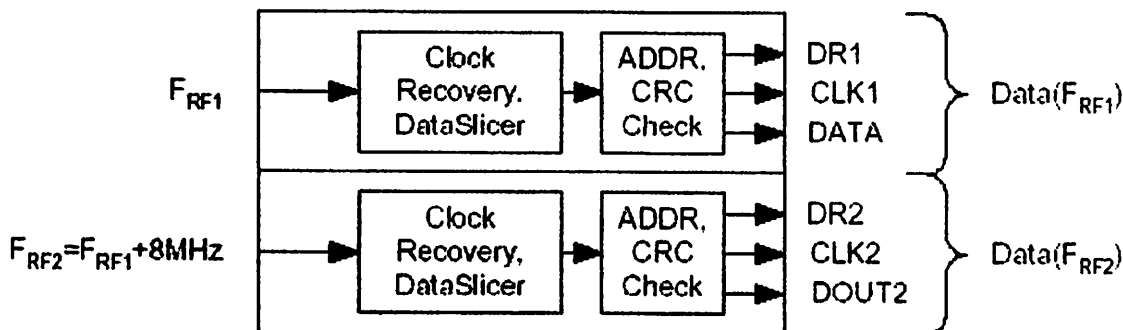


Figure 5 DuoCeiver™ with two simultaneously independent receive channels.

DEVICE CONFIGURATION

All configuration of the **TRW-2.4G** is done via a 3-wire interface to a single configuration register. The configuration word can be up to 15 bytes long for ShockBurst™

Configuration for ShockBurst™ operation

The configuration word in ShockBurst™ enables the **TRW-2.4G** to handle the RF protocol. Once the protocol is completed and loaded into **TRW-2.4G** only one byte, bit[7:0], needs to be updated during actual operation.

The configuration blocks dedicated to ShockBurst™ is as follows:

- **Payload section width:** Specifies the number of payload bits in a RF package. This enables the **TRW-2.4G** to distinguish between payload data and the CRC bytes in a received package.
- **Address width:** Sets the number of bits used for address in the RF package. This enables the **TRW-2.4G** to distinguish between address and payload data.
- **Address (RX Channel 1 and 2):** Destination address for received data.
- **CRC:** Enables nRF2401 on-chip CRC generation and de-coding.

NOTE:

These configuration blocks, with the exception of the CRC, are dedicated for the packages that a **TRW-2.4G** is to receive.

In TX mode, the MCU must generate an address and a payload section that fits the configuration of the **TRW-2.4G** that is to receive the data.

When using the **TRW-2.4G** on-chip CRC feature ensure that CRC is enabled and uses the same length for both the TX and RX devices.

PRE-AMBLE	ADDRESS	PAYLOAD	CRC
-----------	---------	---------	-----

Figure 10 Data packet set-up

Configuration Word overview

	Bit position	Number of bits	Name	Function
ShockBurst™ configuration	143:120	24	TEST	Reserved for testing
	119:112	8	DATA2_W	Length of data payload section RX channel 2
	111:104	8	DATA1_W	Length of data payload section RX channel 1
	103:64	40	ADDR2	Up to 5 byte address for RX channel 2
	63:24	40	ADDR1	Up to 5 byte address for RX channel 1
	23:18	6	ADDR_W	Number of address bits (both RX channels).
	17	1	CRC_L	8 or 16 bit CRC
	16	1	CRC_EN	Enable on-chip CRC generation/checking.
General device configuration	15	1	RX2_EN	Enable two channel receive mode
	14	1	CM	Communication mode (Direct or ShockBurst™)
	13	1	RFDR_SB	RF data rate (1Mbps requires 16MHz crystal)
	12:10	3	XO_F	Crystal frequency
	9:8	2	RF_PWR	RF output power
	7:1	7	RF_CH#	Frequency channel
	0	1	RXEN	RX or TX operation

Table 1 Table of configuration words.

The configuration word is shifted in MSB first on positive CLK1 edges. New configuration is enabled on the falling edge of CS.

NOTE.

On the falling edge of CS, the TRW-2.4G updates the number of bits actually shifted in during the last configuration.

Ex:

If the TRW-2.4G is to be configured for 2 channel RX in ShockBurst™, a total of 120 bits must be shifted in during the first configuration after VDD is applied.

Once the wanted protocol, modus and RF channel are set, only one bit (RXEN) is shifted in to switch between RX and TX.

WENSHING

TRW-2.4GHz Radio Transceiver

Configuration Word Detailed Description

The following describes the function of the 144 bits (bit 143 = MSB) that is used to configure the TRW-2.4G

General Device Configuration: bit[15:0]

ShockBurst™ Configuration: bit[119:0]

Test Configuration: bit[143:120]

MSB	TEST							
D143	D142	D141	D140	D139	D138	D137	D136	
Reserved for testing								
1	0	0	0	1	1	1	0	Default

MSB	TEST															
D135	D134	D133	D132	D131	D130	D129	D128	D127	D126	D125	D124	D123	D122	D121	D120	
Reserved for testing															Close PLL on TX	
0	0	0	0	1	0	0	0	0	0	0	1	1	1	0	0	Default

DATA2 W								
D119	D118	D117	D116	D115	D114	D113	D112	
Data width channel#2 in # of bits excluding addr/crc								
0	0	1	0	0	0	0	0	Default

DATA1 W								
D111	D110	D109	D108	D107	D106	D105	D104	
Data width channel#1 in # of bits excluding addr/crc								
0	0	1	0	0	0	0	0	Default

ADDR2												
D103	D102	D101	D71	D70	D69	D68	D67	D66	D65	D64	
Channel#2 Address RX (up to 40bit)												
0	0	0	...	1	1	1	0	0	1	1	1	Default

ADDR1												
D63	D62	D61	D31	D30	D29	D28	D27	D26	D25	D24	
Channel#1 Address RX (up to 40bit)												
0	0	0	...	1	1	1	0	0	1	1	1	Default

ADDR W						
D23	D22	D21	D20	D19	D18	
Address width in # of bits (both channels)						
0	0	1	0	0	0	Default

CRC		
D17	D16	
CRC Mode 1 - 16bit, 0 - 8bit	CRC 1 - enable, 0 - disable	
0	1	Default

RF-Programming															LSB	
D15	D14	D13	D12	D11	D10	D9	D8	D7	D6	D5	D4	D3	D2	D1	D0	
Two Ch.	BUF	OD	XO	RF Power			Channel selection								RXEN	
0	0	0	0	1	1	1	1	0	0	0	0	0	1	0	0	Default

Table 2 Configuration data word

The MSB bit should be loaded first into the configuration register.

Default configuration word: h8E08.1C20.2000.0000.00E7.0000.0000.E721.0F04.

ShockBurst™ configuration:

The section B[119:16] contains the segments of the configuration register dedicated to ShockBurst™ operational protocol. After VDD is turned on ShockBurst™ configuration is done once and remains set whilst VDD is present. During operation only the first byte for frequency channel and RX/TX switching need to be changed.

PLL_CTRL

PLL_CTRL		
D121	D120	PLL
0	0	Open TX/Closed RX
0	1	Open TX/Open RX
1	0	Closed TX/Closed RX
1	1	Closed TX/Open RX

Table 10 PLL setting.

Bit 121-120:

PLL_CTRL: Controls the setting of the PLL for test purposes. With closed PLL in TX no deviation will be present.

DATAx_W

DATA2_W							
119	118	117	116	115	114	113	112

DATA1_W							
111	110	109	108	107	106	105	104

Table 4 Number of bits in payload.

Bit 119 – 112:

DATA2_W: Length of RF package payload section for receive-channel 2.

Bit 111 – 104:

DATA1_W: Length of RF package payload section for receive-channel 1.

NOTE:

The total number of bits in a ShockBurst™ RF package may not exceed 256! Maximum length of payload section is hence given by:

$$DATAx_W(bits) = 256 - ADDR_W - CRC$$

Where:

ADDR_W: length of RX address set in configuration word B[23:18]

CRC: check sum, 8 or 16 bits set in configuration word B[17]

PRE: preamble, 4 or 8 bits are automatically included

Shorter address and CRC leaves more room for payload data in each package.

WENSHING

TRW -2.4GHz Radio Transceiver

ADDRx

ADDR2											
103	102	101	71	70	69	68	67	66	65	64

ADDR1											
63	62	61	31	30	29	28	27	26	25	24

Table 5 : Address of receiver #2 and receiver #1.

Bit 103 – 64:

ADDR2: Receiver address channel 2, up to 40 bit.

Bit 63 – 24: ADDR1

ADDR1: Receiver address channel 1, up to 40 bit.

NOTE!

Bits in ADDR_x exceeding the address width set in ADDR_W are redundant and can be set to logic 0.

ADDR_W & CRC

ADDR_W						CRC L	CRC EN
23	22	21	20	19	18	17	16

Table 6 Number of bits reserved for RX address + CRC setting.

Bit 23 – 18:

ADDR_W: Number of bits reserved for RX address in ShockBurst™ packages.

NOTE:

Maximum number of address bits is 40 (5 bytes). Values over 40 in ADDR_W are not valid.

Bit 17:

CRC_L: CRC length to be calculated by TRW-2.4G in ShockBurst™.
Logic 0: 8 bit CRC
Logic 1: 16 bit CRC

Bit: 16:

CRC_EN: Enables on-chip CRC generation (TX) and verification (RX).
Logic 0: On-chip CRC generation/checking disabled
Logic 1: On-chip CRC generation/checking enabled

NOTE:

An 8 bit CRC will increase the number of payload bits possible in each ShockBurst™ data packet, but will also reduce the system integrity.

General device configuration:

This section of the configuration word handles RF and device related parameters.

Modes:

RX2 EN	CM	RFDR SB	XO F			RF PWR	
15	14	13	12	11	10	9	8

Table 7 RF operational settings.

Bit 15:

RX2_EN:

Logic 0: One channel receive

Logic 1: Two channels receive

NOTE:

In two channels receive, the TRW-24G receives on two, separate frequency channels simultaneously. The frequency of receive channel 1 is set in the configuration word B[7-1], receive channel 2 is always 8 channels (8 MHz) above receive channel 1.

Bit 14:

Communication Mode:

Logic 1: nRF2401 operates in ShockBurst™ mode

Bit 13:

RF Data Rate:

Logic 0: 250 ^{TRW-24G} kbps

Logic 1: 1 Mbps

NOTE:

Utilizing 250 kbps instead of 1Mbps will improve the receiver sensitivity by 10 dB. 1Mbps requires 16MHz crystal.

Bit 12-10:

D12	D11	D10
0	1	1

Table 8

Bit 9-8:

RF_PWR: Sets TRW-2.4G RF output power in transmit mode:

RF OUTPUT POWER		
D9	D8	P [dBm]
0	0	-20
0	1	-10
1	0	-5
1	1	0

Table 9 · RF output power setting.

· RF channel & direction

RF CH#							RXEN
7	6	5	4	3	2	1	0

Table 10 Frequency channel + RX / TX setting.

Bit 7 – 1:

RF_CH#: Sets the frequency channel the nRF2401 operates on.

The channel frequency in *transmit* is given by:

$$Channel_{RF} = 2400 \text{ MHz} + RF_CH\# \cdot 1.0 \text{ MHz}$$

RF_CH #: between 2400MHz and 2527MHz may be set.

The channel frequency in *data channel 1* is given by:

$$Channel_{RF} = 2400 \text{ MHz} + RF_CH\# \cdot 1.0 \text{ MHz} \text{ (Receive at PIN\#8)}$$

RF_CH #: between 2400MHz and 2524MHz may be set.

NOTE:

The channels above 83 can only be utilized in certain territories (ex: Japan)

The channel frequency in *data channel 2* is given by:

$$Channel_{RF} = 2400 \text{ MHz} + RF_CH\# \cdot 1.0 \text{ MHz} + 8\text{MHz} \text{ (Receive at PIN\#4)}$$

RF_CH #: between 2408MHz and 2524MHz may be set.

Bit 0:

Set active mode:

Logic 0: transmit mode

Logic 1: receive mode

DATA PACKAGE DESCRIPTION

Figure 7 Data Package Diagram

The data packet for both ShockBurst™ mode and direct mode communication is divided into 4 sections. These are:

1. PREAMBLE	<ul style="list-style-type: none">· The preamble field is required in ShockBurst.
2. ADDRESS	<ul style="list-style-type: none">· The address field is required in ShockBurst. mode.· 8 to 40 bits length.· Address automatically removed from received packet in ShockBurst.mode
3. PAYLOAD	<ul style="list-style-type: none">· The data to be transmitted· In Shock-Burst mode payload size is 256 bits minus the following:(Address: 8 to 40 bits. + CRC 8 or 16 bits).
4. CRC	<ul style="list-style-type: none">· 8 or 16 bits length· The CRC is stripped from the received output data.

配置WS2401模組數據時從高位開始。

范例：在ShockBurth的發射模式，以一個通道在2410MHz下以1Mbps Rate傳送。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	1	1	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	0

范例：在ShockBurth的接收模式，以一個通道在2410MHz下以1Mbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	1	1	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

范例：在ShockBurth的发射模式，以一个通道在2410MHz下以250Kbps Rate发射。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	1	0	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	0

范例：在ShockBurth的接收模式，以一個通道在2410MHz下以250Kbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	1	0	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

范例：在ShockBurth的接收模式，以二個通道在2410MHz下以1Mbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
1	1	1	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

范例：在ShockBurth的接收模式，以二個通道在2410MHz下以250Kbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
1	1	0	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

配置WS2401模組數據時從高位開始。

范例：在直接發射模式，以一個通道在2410MHz頻道下以1Mbps Rate發射。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	0	1	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	0

范例：在直接接收模式，以一個通道在2410MHz頻道下以1Mbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	0	1	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

范例：在直接發射模式，以一個通道在2410MHz頻道下以250Kbps Rate發射。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	0	0	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	0

范例：在直接接收模式，以一個通道在2410MHz頻道下以250Kbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
0	0	0	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

范例：在直接接收模式，以二個通道在2410MHz頻道下以250Kbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
1	0	0	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

范例：在直接接收模式，以二個通道在2410MHz頻道下以1Mbps Rate接收。

Bit143	Bit142	Bit141	Bit140	Bit139	Bit138	Bit137	Bit136
1	0	0	0	1	1	1	0
Bit135	Bit134	Bit133	Bit132	Bit131	Bit130	Bit129	Bit128
0	0	0	0	1	0	0	0
Bit127	Bit126	Bit125	Bit124	Bit123	Bit122	Bit121	Bit120
0	0	0	1	1	1	0	0
Bit119	Bit118	Bit117	Bit116	Bit115	Bit114	Bit113	Bit112
1	1	0	0	1	0	0	0
Bit111	Bit110	Bit109	Bit108	Bit107	Bit106	Bit105	Bit104
1	1	0	0	1	0	0	0
Bit103	Bit102	Bit101	Bit100	Bit99	Bit98	Bit97	Bit96
1	1	0	0	0	0	0	0
Bit95	Bit94	Bit93	Bit92	Bit91	Bit90	Bit89	Bit88
1	0	1	0	1	0	1	0
Bit87	Bit86	Bit85	Bit84	Bit83	Bit82	Bit81	Bit80
0	1	0	1	0	1	0	1
Bit79	Bit78	Bit77	Bit76	Bit75	Bit74	Bit73	Bit72
1	0	1	0	1	0	1	0
Bit71	Bit70	Bit69	Bit68	Bit67	Bit66	Bit65	Bit64
0	1	0	1	0	1	0	1
Bit63	Bit62	Bit61	Bit60	Bit59	Bit58	Bit57	Bit56
1	0	1	0	1	0	1	0
Bit55	Bit54	Bit53	Bit52	Bit51	Bit50	Bit49	Bit48
0	1	0	1	0	1	0	1
Bit47	Bit46	Bit45	Bit44	Bit43	Bit42	Bit41	Bit40
1	0	1	0	1	0	1	0
Bit39	Bit38	Bit37	Bit36	Bit35	Bit34	Bit33	Bit32
0	1	0	1	0	1	0	1
Bit31	Bit30	Bit29	Bit28	Bit27	Bit26	Bit25	Bit24
1	0	1	0	1	0	1	0
Bit23	Bit22	Bit21	Bit20	Bit19	Bit18	Bit17	Bit16
1	0	1	0	0	0	1	1
Bit15	Bit14	Bit13	Bit12	Bit11	Bit10	Bit9	Bit8
1	0	1	0	1	1	1	1
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
0	0	0	1	0	1	0	1

IMPORTANT TIMING DATA

The following timing applies for operation of **nRF2401** TRW-2.4G

TRW-2.4G Timing Information

nRF2401 timing	Max.	Min.	Name
VDD OFF → ST_BY mode	3ms		Tpd2sby
VDD OFF → Active mode (RX/TX)	3ms		Tpd2a
ST_BY → TX ShockBurst™	195µs		Tsby2txSB
ST_BY → TX Direct Mode	202µs		Tsby2txDM
ST_BY → RX mode	202µs		Tsby2rx
Minimum delay from CS to data.		5µs	Tcs2data
Minimum delay from CE to data.		5µs	Tce2data
Minimum delay from DR1/2 to clk.		50ns	Tdr2clk
Maximum delay from clk to data.	50ns		Tclk2data
Delay between edges		50ns	Td
Setup time		500ns	Ts
Hold time		500ns	Th
Delay to finish internal GFSK data		1/data rate	Tfl
Minimum input clock high		500ns	Thmin
Set-up of data in Direct Mode	50ns		Tsdm
Minimum clock high in Direct Mode		300ns	Thdm
Minimum clock low in Direct Mode		230ns	Tldm

Table 11 Switching times for **nRF2401** TRW-2.4G

When **nRF2401** is in power down it must always settle in stand-by (Tpd2sby) before it can enter configuration or one of the active modes.

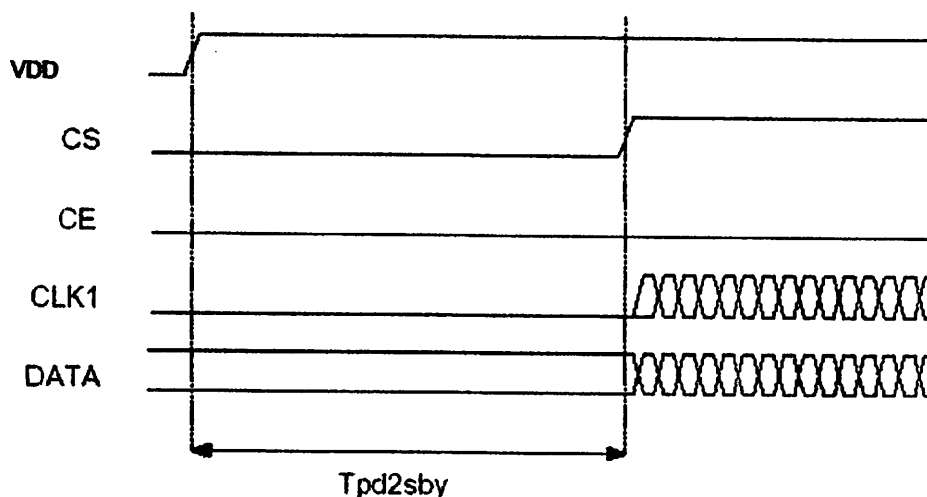


Figure 8 Timing diagram for **nRF2401** (or VDD off) to stand by mode for TRW-2.4G

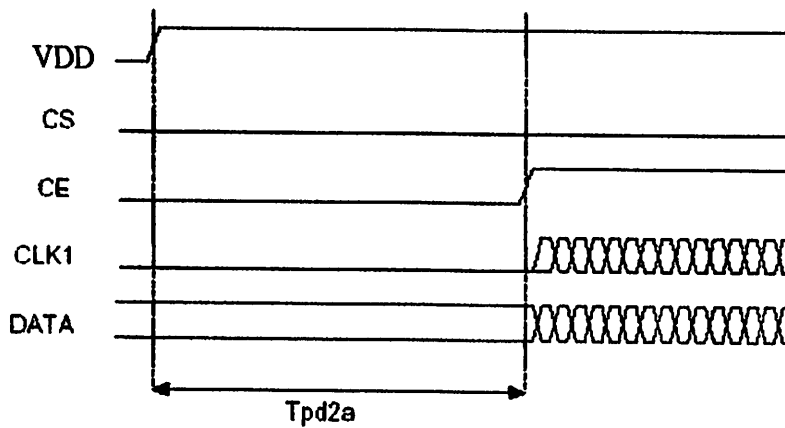


Figure 9

VDD off to active mode

Note that the configuration word will be lost when VDD is turned off and that the device then must be configured before going to one of the active modes. If the device is configured one can go directly from power down to the wanted active mode.

Note:

CE and CS may not be high at the same time. Setting one or the other decides whether configuration or active mode is entered.

Configuration mode timing

When one or more of the bits in the configuration word needs to be changed the following timing apply.

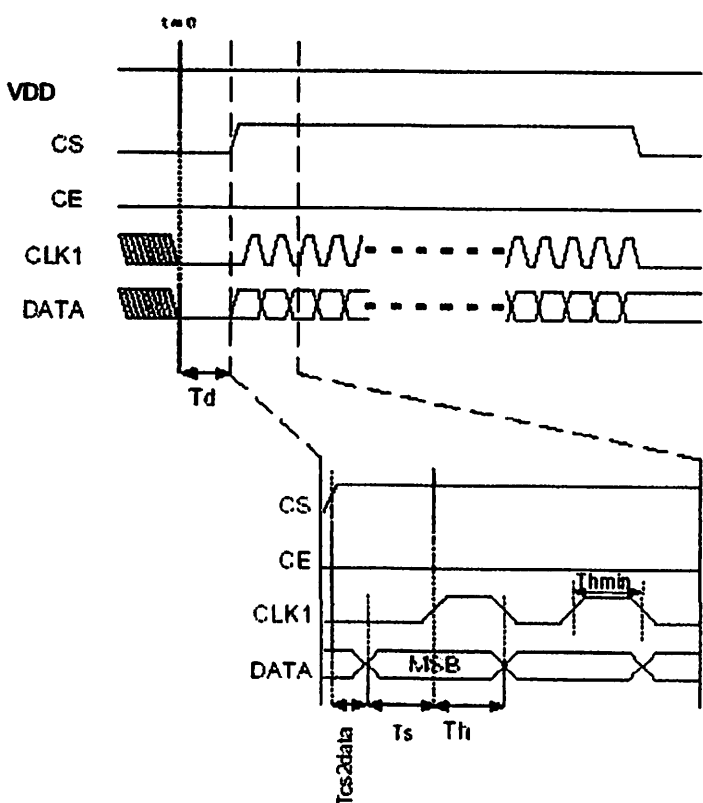


Figure 10 Timing diagram for configuration of TRW-2.4G

If configuration mode is entered from power down, CS can be set high after T_{pd2sby} as shown in Figure 8

ShockBurst™ Mode timing

ShockBurst™ TX:

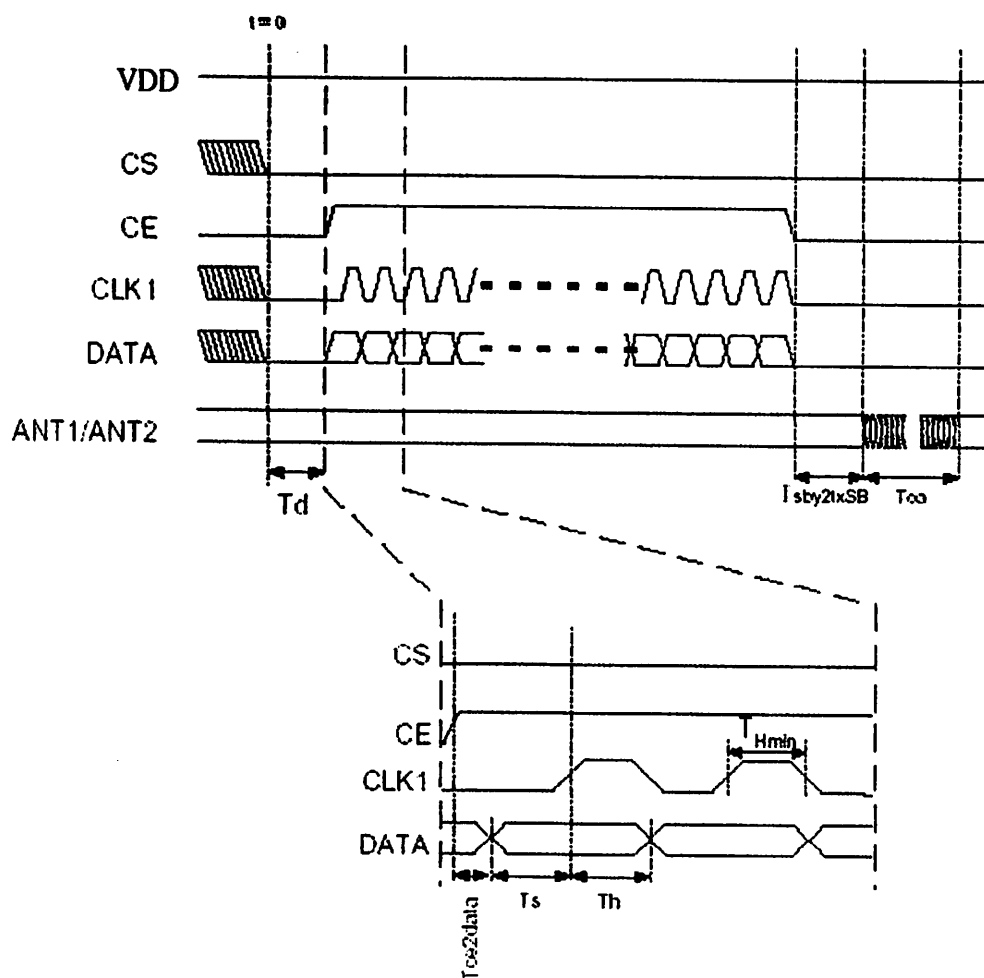


Figure 11 Timing of ShockBurst™ in TX

The package length and the data rate give the delay T_{toa} (time on air), as shown in the equation.

$$T_{toa} = \frac{L}{\text{data rate}} \cdot (\# \text{ databits} + 1)$$

ShockBurst™ RX:

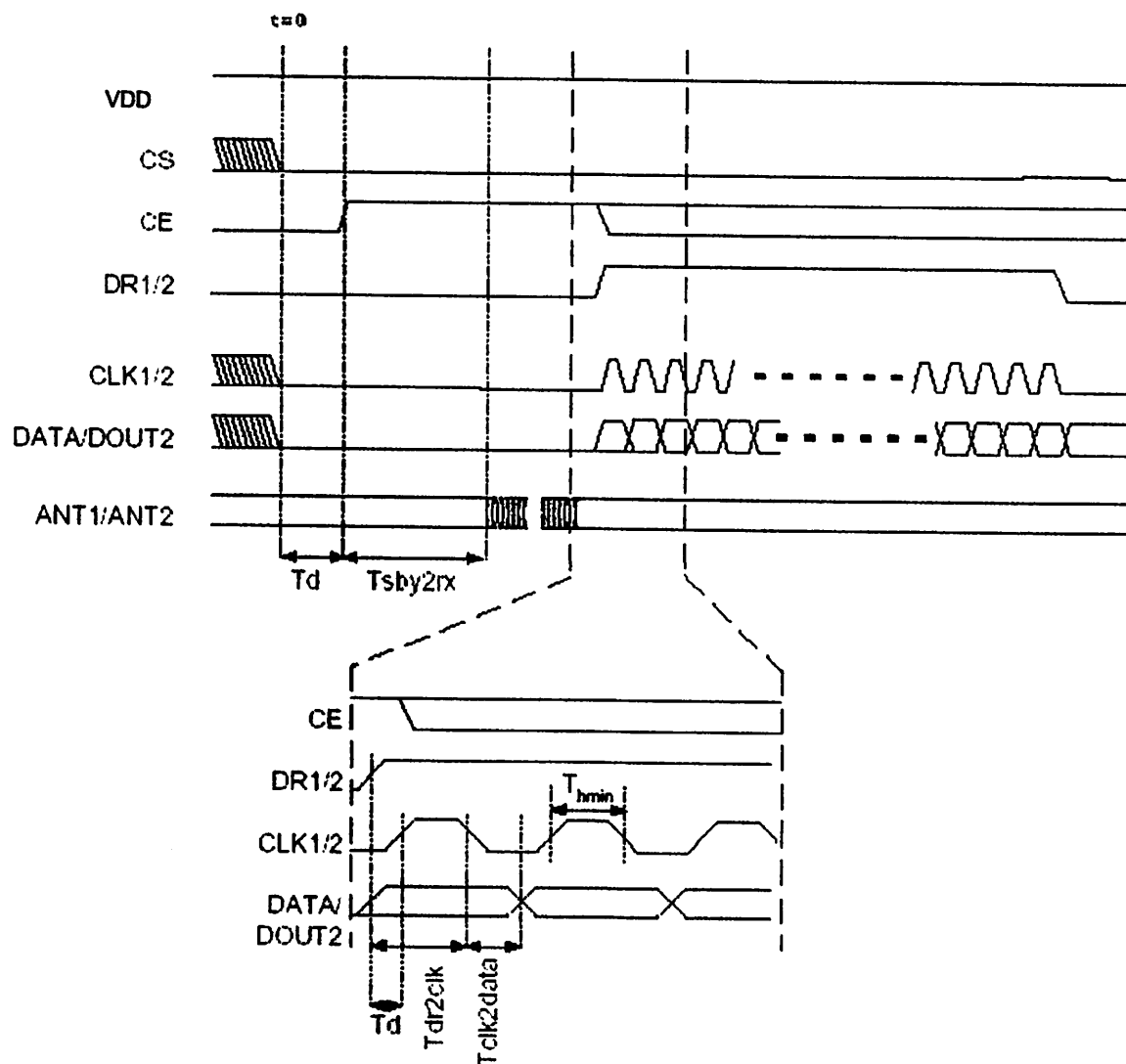


Figure 12 Timing of ShockBurst™ in RX

The CE may be kept high during downloading of data, but the cost is higher current consumption (18mA) and the benefit is no start-up time (200 μ s) after the DR1 goes low.

Output Power adjustment

Power setting bits of configuring word	RF output power	DC current consumption
11	0 dBm \pm 3dB	13.0 mA
10	-5 dBm \pm 3dB	10.5 mA
01	-10 dBm \pm 3dB	9.4 mA
00	-20 dBm \pm 3dB	8.8 mA

Conditions: VDD = 3.0V, VSS = 0V, T_A = 27°C, Load impedance = 400 Ω .


```
ORG 0X8
JMP INT_1
```

RF_CONFIG_TABLE:

```
MOV A,R0
ADD 0X2,A

RETL 0X8E ;TEST
RETL 0X08
RETL 0X1C

RETL 0X20 ;DATA2_W

RETL 0X0D0 ;DATA1_W

RETL 0X0BB ;ADDR2
RETL 0X0BB
RETL 0X0BB
RETL 0X0BB
RETL 0X0BB

RETL 0X12 ;ADDR1
RETL 0X34
RETL 0X56
RETL 0X78
RETL 0X9A

RETL 0X0A1 ;ADDR_W/CRC

RETL 0X53 ;RF-PROGRAMMING
RETL 0X02
```

```
-----
ADDRESS_TABLE: MOV A,R0
ADD 0X2,A
RETL 0X12 ;ADDR1
RETL 0X34
RETL 0X56
```

```
RETL 0X78
RETL 0X9A
```

```
;-----
RESET:
```

```
MOV A,@0X0
IOW 0XF
IOW 0XE
MOV A,@0X0FF
IOW 0XB
MOV A,@0X0F8
IOW 0XD
```

```
MOV A,@0X08 ;P5,0-LED P5,1-LED P5,2-PW
P5,3-CRYSTALL_SLECT
IOW 0X5
MOV A,@0X47 ;P6,0-KEY P6,1-KEY P6,2-KEY
P6,3-CE
IOW 0X6 ;P6,4-DATA P6,5-CLK1 P6,6-DR1 P6,7-CS
BC P6,3 ;CE
BC P6,7 ;CS
MOV A,@0X0D ;WDT
CONTW
MOV A,@0X80
IOW 0XE ;WATCHDOG ENABLE
```

```
MOV A,@0X2F
MOV 0X3F,A
MOV A,@0X10
MOV 0X4,A
CLEAR_REG: CLR0X0
INC 0X4
DJZ 0X3F
JMP CLEAR_REG
```

```
MOV A,@0X3
MOV 0X10,A
```

S_LED_TEST:

MOV A,@0X0FF

MOV 0X11,A

S_LED_TEST1:

CALL DELAY1MS

CALL DELAY1MS

WDTC

JBS 0X11,7

JMP S_LED_OFF

BS P5,0 ;LED

BS P5,1 ;LED

JMP S_SKIP

S_LED_OFF:

BC P5,0 ;LED

BC P5,1 ;LED

S_SKIP:

DJZ 0X11

JMP S_LED_TEST1

DJZ 0X10

JMP S_LED_TEST

; ENI

;

===== MAIN PROGRAM =====

;

MAIN_LOOP:

BS P5,2 ;PW

BC P5,0 ;R_LED

BC P5,1 ;T_LED

BC P6,3 ;CE

CALL DELAY1MS

MOV A,P6

AND A,@0X07

MOV KEY_REG,A

JBC P5,3 ;CRYSTALL SELECT<0-16M 1-20M>

BS KEY_REG,3

MOV A,KEY_REG

```
AND    A,@0X03
XOR    A,@0X0
JBC    0X3,2      ;Z
JMP    T_MODEL
```

```
MOV    A,KEY_REG
AND    A,@0X03
XOR    A,@0X1
JBC    0X3,2      ;Z
JMP    R_MODEL
```

```
MOV    A,KEY_REG
AND    A,@0X03
XOR    A,@0X03
JBC    0X3,2      ;Z
JMP    SLEEP_MODEL
```

```
JMP    MAIN_LOOP
```

T_MODEL:

```
BC     P5,0      ;R_LED
BC     P5,1      ;T_LED
CALL   DATA_PRO
MOV    A,@0X1C      ;TX_ON  RX_OFF
MOV    0X22,A
MOV    A,@0X4F      ;250K 0db
JBC    P6,2      ;<0-250K  1-1000K>
MOV    A,@0X6F
```

```
MOV    0X30,A
MOV    A,@0X14      ;2410
MOV    0X31,A
CALL   RF_CONFIG
CALL   DELAY200US
CALL   RF_CONFIG
CALL   DELAY200US
```

T_LOOP:

```

BS      P5,1      ;T_LED
CALL   RF_SEND
CALL   DELAY1MS
CALL   DELAY1MS
CALL   DELAY1MS
CALL   DELAY1MS
CALL   DELAY1MS
CALL   DELAY1MS
CALL   DELAY1MS
CALL   DELAY1MS
CALL   DELAY1MS
WDTC
MOV    A,P6
AND    A,@0X07
MOV    BYTE,A
JBC    P5,3      ;KEY
BS     BYTE,3
MOV    A,BYTE
XOR    A,KEY_REG
JBC    0X3,2      ;Z
JMP    T_LOOP
BC     P5,1      ;T_LED
JMP    MAIN_LOOP

```

R_MODEL:

```

BC     P5,0      ;R_LED
BC     P5,1      ;T_LED
CALL   DATA_PRO
MOV    A,@0X1F      ;TX_OFF  RX_ON
MOV    0X22,A
MOV    A,@0X4F      ;250K 0db
JBC    P6,2      ;<0-250K  1-1000K>
MOV    A,@0X6F
MOV    0X30,A
MOV    A,@0X15      ;2410
MOV    0X31,A
CALL   RF_CONFIG

```

```
MOV A,@0X10
MOV 0X13,A
BS P6,3 ;CE
CLR TCC
BC 0XF,0 ;TIMER FLAG
MOV A,@0X1
IOW 0XF ;TIMER INTERRUPT ENABLE
```

R_SP1:

```
BC P5,0 ;R_LED
```

R_LOOP1:

```
MOV A,@0X060
MOV TIMER_REG,A
BC FLAG,0 ;INTERRUPT FLAG
DJZ 0X13
JMP R_LOOP
JMP R_MODEL
```

R_LOOP:

```
WDTC
JBC FLAG,0 ;INTERRUPT FLAG
JMP R_SP1
MOV A,P6
AND A,@0X7
MOV BYTE,A
JBC P5,3 ;KEY
BS BYTE,3
MOV A,BYTE
XOR A,KEY_REG
JBS 0X3,2 ;Z
JMP R_RET
JBS P6,6 ;DR1
JMP R_LOOP
BS P5,0 ;R_LED
MOV A,@0X10
MOV 0X13,A
CALL RF_RECEIVE
CALL DELAY1MS
```

JMP R_LOOP1

R_RET:

BC P6,3 ;CE
BC P5,0
MOV A,@0X0
IOW 0XF
JMP MAIN_LOOP

SLEEP_MODEL:

CALL DATA_PRO
BC P5,0 ;R_LED
BC P5,1 ;T_LED
CALL DATA_PRO
MOV A,@0X1E ;TX_OFF RX_OFF
MOV 0X22,A
MOV A,@0X4C ;250K 0db
JBC P6,2 ;<0-250K 1-1000K>
MOV A,@0X6C
MOV 0X30,A
MOV A,@0X15 ;2410
MOV 0X31,A
CALL RF_CONFIG
BC P5,2 ;PW

S_LOOP:

CALL DELAY1MS
WDTC
MOV A,P6
AND A,@0X07
MOV BYTE,A
JBC P5,3 ;KEY
BS BYTE,3
MOV A,BYTE
XOR A,KEY_REG
JBC 0X3,2 ;Z
JMP S_LOOP

JMP MAIN_LOOP

DATA_PRO:

```
MOV A,@0X20
MOV 0X4,A
CLR R0
MOV A,@0X12 ;18
MOV 0X10,A
```

DATA_LOOP:

```
CALL RF_CONFIG_TABLE
MOV 0X0,A
INC R0
INC 0X4
DJZ 0X10
JMP DATA_LOOP
RET
```

RF CONFIG SUBROUTINE

RF_CONFIG:

```
BC P6,3 ;CE
BS P6,7 ;CS
CALL DELAY200US
MOV A,@0X12 ;18
MOV 0X10,A
MOV A,@0X20
MOV 0X4,A
```

RF_CONF_BYTE_LP:

```
MOV A,@0X8
MOV 0X11,A
MOV A,0X0
MOV BYTE,A
```

RF_CONF_BIT_LP:

```
BC P6,5 ;CLK1
JBS BYTE,7
BC P6,4 ;DATA
JBC BYTE,7
```



```

BS    P6,4    ;DATA
RLC   BYTE
BS    P6,5    ;CLK1
DJZ   0X11
JMP   RF_CONF_BIT_LP
INC   0X4
DJZ   0X10
JMP   RF_CONF_BYTE_LP
CALL  DELAY5US
BC    P6,5    ;CLK1
BC    P6,7    ;CS
RET

```

```

=====
;===== RF SEND SUBROUTINE =====
;=====

```

RF_SEND:

```

BC    P6,7    ;CS
BS    P6,3    ;CE
CALL  DELAY5US
CALL  DELAY5US
CLR   R0
MOV   A,@0X5    ;ADDRESS BIT 5*8=40 BIT
MOV   0X10,A

```

RF_S_ADD_BYTE:

```

CALL  ADDRESS_TABLE
MOV   BYTE,A
MOV   A,@0X8
MOV   0X11,A

```

RF_S_ADD_BIT:

```

BC    P6,5    ;CLK1
JBS   BYTE,7
BC    P6,4    ;DATA
JBC   BYTE,7
BS    P6,4    ;DATA
RLC   BYTE
BS    P6,5    ;CLK1
DJZ   0X11
JMP   RF_S_ADD_BIT

```

```
INC    R0
DJZ    0X10
JMP    RF_S_ADD_BYTE
```

```
MOV    A,@0X1A      ;208 BIT DATA
MOV    0X10,A
```

RF_S_D_BYTE:

```
MOV    A,@0X8
MOV    0X11,A
MOV    A,@0X037
MOV    BYTE,A
```

RF_S_D_BIT:

```
BC     P6,5      ;CLK1
JBS    BYTE,7
BC     P6,4      ;DATA
JBC    BYTE,7
BS     P6,4      ;DATA
RLC    BYTE
BS     P6,5      ;CLK1
DJZ    0X11
JMP    RF_S_D_BIT
DJZ    0X10
JMP    RF_S_D_BYTE
CALL   DELAY5US
BC     P6,5      ;CLK1
BC     P6,3      ;CE
RET
```

```
=====
;===== RF RECEIVE SUBROUTINE =====
;=====
```

RF_RECEIVE:

```
BC     P6,7      ;CS
BC     P6,3      ;CE
CALL   DELAY5US
CALL   DELAY5US
MOV    A,@0X20
MOV    0X4,A
MOV    A,@0X1A      ;208 bit
```

```

        MOV    0X10,A
        CLR    BYTE
RF_BYTE_LP:
        MOV    A,@0X8
        MOV    0X11,A
RF_BIT_LP:
        RLC    BYTE
        BS     P6,5      ;CLK1
        JBS    P6,4      ;DATA
        BC     BYTE,0
        JBC    P6,4      ;DATA
        BS     BYTE,0
        BC     P6,5      ;CLK1
        DJZ    0X11
        JMP    RF_BIT_LP
        INC    0X4
        DJZ    0X10
        JMP    RF_BYTE_LP
        CALL   DELAY5US
        CALL   DELAY5US
        BS     P6,3      ;CE
        RET

```

```

=====
===== DELAY1MS SUBROUTINE =====
=====

```

DELAY1MS:

```

        MOV    A,@0X0FF
        JMP    DLY_SKIP

```

DELAY200US:

```

        MOV    A,@0X32
        JMP    DLY_SKIP

```

DELAY5US:

```

        MOV    A,@0X1

```

DLY_SKIP:

```

        MOV    DLY_REG,A

```

```
DLY_LOOP:
    NOP
    NOP
    NOP
    DJZ    DLY_REG
    JMP    DLY_LOOP
    RET
```

```
=====
;===== INTERRUPT SUBROUTINE =====
;=====
```

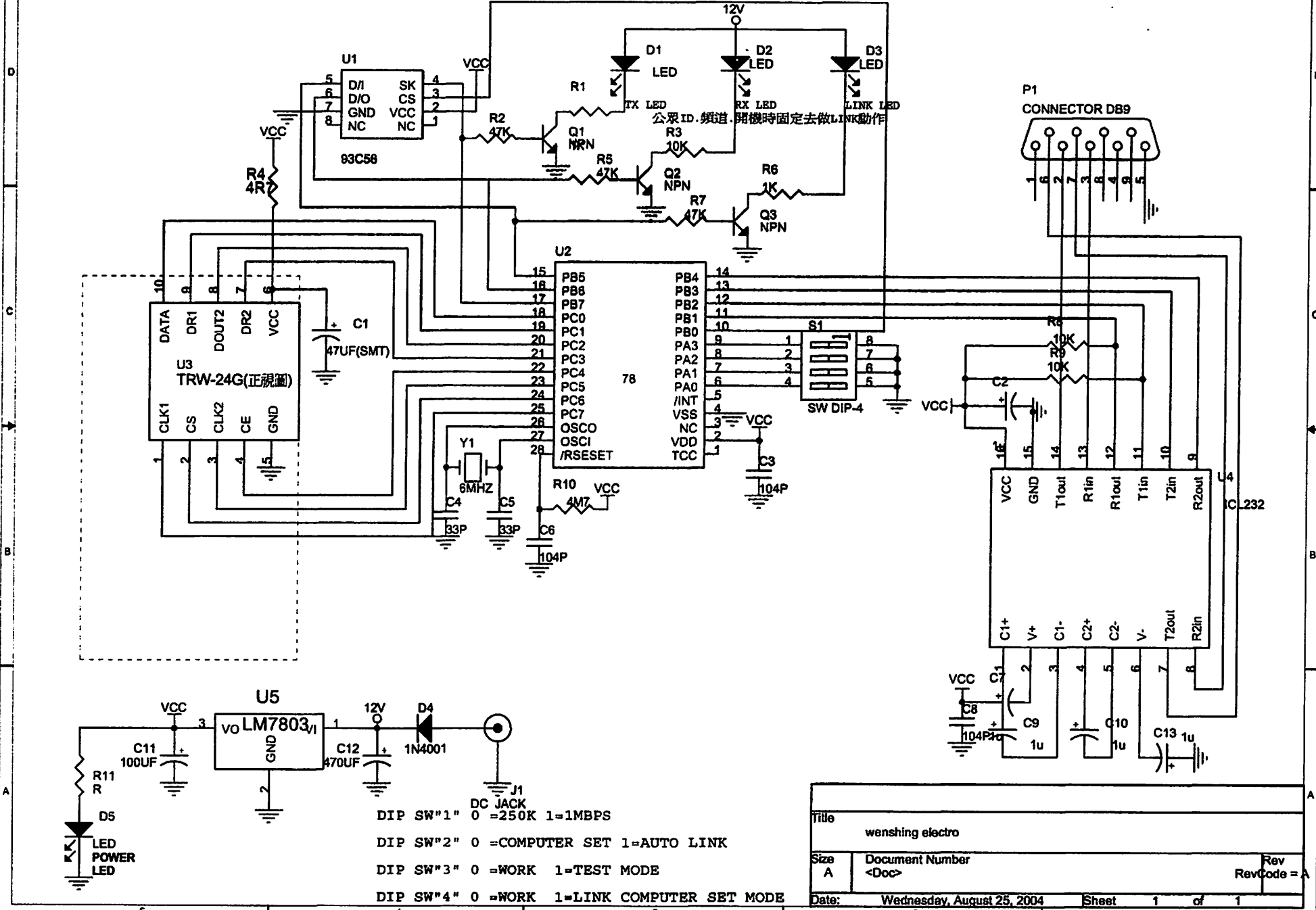
```
INT_0:
```

```
INT_1:
```

```
    BC    0XF,0    ;TIMER FLAG
    DJZ    TIMER_REG
    JMP    INT_RET
    BS    FLAG,0    ;INTERRUPT FLAG
```

```
INT_RET:
```

```
    RETI
```



- DIP SW"1" 0 =250K 1=1MBPS
- DIP SW"2" 0 =COMPUTER SET 1=AUTO LINK
- DIP SW"3" 0 =WORK 1=TEST MODE
- DIP SW"4" 0 =WORK 1=LINK COMPUTER SET MODE

Title		
wenshing electro		
Size	Document Number	Rev
A	<Doc>	Code = A
Date:	Wednesday, August 25, 2004	Sheet 1 of 1

```

/* -----
. . . . .
----- */
#include <C8051F330.H>
#include "Public_variable.h"
/* -----
. . . 10. . . . .
----- */
sbit TRW_24G_CLK = P1^1;
sbit TRW_24G_DATA = P1^6;
sbit TRW_24G_CS = P1^2;
sbit TRW_24G_CE = P1^3;
sbit TRW_24G_DR1 = P1^4;
/* -----
. . . . TRW-24N/G. . . . . BYTE. . . . . BYTE. . . . .
----- */
void Write_TRW_24N_BYTE(x)
{
    char i;
    for(i=0;i<8;i++)
    {
        TRW_24G_CLK = 0;
        if(x&0x80)
            TRW_24G_DATA= 1;
        else
            TRW_24G_DATA= 0;
        x<<=1;
        TRW_24G_CLK = 1;
        TRW_24G_CLK = 1;
        TRW_24G_CLK = 1;
    }
}
/* -----
. . . . TRW-24N/G. . . . . BYTE. . . . .
----- */
char Read_TRW_24N_BYTE(void)
{
    char i,x;
    for(i=0;i<8;i++)
    {
        TRW_24G_CLK = 0;
        TRW_24G_CLK = 0;
        TRW_24G_CLK = 0;
        TRW_24G_CLK = 1;
        x<<=1;
        if(TRW_24G_DATA)
            x|=0x01;
        else
            x|=0x00;
    }
    return(x);
}
/* -----
. . . . . TRW-24N/G. . . . .
----- */
void Config_TRW_24N(void)
{
    unsigned char i;
    P1MDOUT |= 0x4E;
    TRW_24G_CE = 0;
    TRW_24G_CS = 1;
    for(i=0;i<18;i++)
        Write_TRW_24N_BYTE(RF_Buffer[i]);
    TRW_24G_CS = 0;
    if(RF_Status[0]&&(RF_Status[6]==1))
    {

```

```

        P1MDOUT &= 0xBD;
        P1      |= 0x42;
        TRW_24G_CE = 1;
    }
}
/* -----
* . . . . . TRW_24N/S* . . . . .
* ----- */
void Send_TRW_24N(char x)
{
    unsigned char i;
    TRW_24G_CS = 0;
    TRW_24G_CE = 1;
    Write_TRW_24N_BYTE(0xF0);
    Write_TRW_24N_BYTE(0xF0);
    for(i=0;i<28;i++)
        Write_TRW_24N_BYTE(x);
    TRW_24G_CE = 0;
}
/* -----
* . . . . . TRW_24S* . . . . .
* ----- */
char Receive_TRW_24N(void)
{
    unsigned char i=0,RF_Data[30];
    TRW_24G_CE = 1;
    P1MDOUT &= 0xBF;
    P1      |= 0x40;
    TRW_24G_CLK = 0;
    if(!TRW_24G_DR1)
    {
        for(i=0;i<28;i++)
            RF_Data[i] = Read_TRW_24N_BYTE();
        i= RF_Data[4];
    }
    return(i);
}
/* -----
* . . . . . TRW_24S*Table* . . . . .
* ----- */
const unsigned char code TRW_24G_Table[18] =
{
    0x8E,0x08,0x1C,0xE0,0xE0,0x00,0x00,0x00,0xF0,
    0xF0,0x00,0x00,0x00,0xF0,0xF0,0x43,0x0F,0x00
};
/* -----
* . . . . . Kit_10* . . . . . TRW_24G/TRW_24N* . . . . .
* . . . . . RF_Status[7]* RF_Buffer[110]* . . . . .
* . . . . . RF_Status[7]* . . . . .
RF_Status[0] = 0 , . . . . .
                = 1 , . . . . .
RF_Status[1] = . . . . .
RF_Status[2][3] = . 16* . . . . .
                . . . . . = 0x0190, . . . . . 2400M* . . . . .
                . . . . . = 0x01B0, . . . . . 2342M* . . . . .
RF_Status[4] = 0 , . . . . . 1M* . . . . .
                = 1 , . . . . . 250K* . . . . .
RF_Status[5] = 0 , . . . . . 20dBm* . . . . .
                = 1 , . . . . . 10dBm* . . . . .
                = 1 , . . . . . 0dBm* . . . . .
RF_Status[6] = 0 , . . . . . direct* . . . . .
                = 1 , . . . . . ShockBurth* . . . . .
* ----- */
void Control_TRW_24N(void)
{
    unsigned char i,Send_Value = 0x00,Receive_Time = 0,Receive_Value=0;

```

```

unsigned int Freq_buffer;

Freq_buffer = RF_Status[2]<<8;
Freq_buffer += RF_Status[3];
if((RF_Status[6]==1)&&(RF_Status[0]==0))
{
    for(Freq_buffer=0;Freq_buffer<10000;Freq_buffer++)
        for(i=0;i<200;i++);
}
else if((Freq_buffer>527)|| (Freq_buffer<400))
    show_Freq_Error();
else
{
    for(i=0;i<18;i++)
        RF_Buffer[i]=TRW_24G_Table[i];
    if(RF_Status[0])
    {
        RF_Buffer[2] = 0x1F;
        RF_Buffer[17] |= 0x01;
    }
    else
    {
        RF_Buffer[2] = 0x1C;
        RF_Buffer[17] &= 0x00;
    }
    Freq_buffer -= 400;
    Freq_buffer <<=1;
    i=Freq_buffer&0xFE;
    RF_Buffer[17] |= i;
    if(RF_Status[6]==0)
        RF_Buffer[16] |= 0x40;
    if(RF_Status[4]==0)
        RF_Buffer[16] |= 0x20;
    switch(RF_Status[5])
    {
        case 0:
            RF_Buffer[16] &= 0xFC; break;
        case 1:
            RF_Buffer[16] |= 0x01; break;
        default:
            RF_Buffer[16] |= 0x03; break;
    }
    Config_TRW_24N();

    if(RF_Status[0]&&(RF_Status[6]==1))
    {
        LCD_write_String(0x02,0x11,0xB2,"Direct output RF",0);
        LCD_write_String(0x02,0x11,0xB3," DATA from TP5 3",0);
        while((Key_Value&0x08)==0x00)
            Scan_Key();
    }
    else
    {
        Freq_buffer >>= 1;
        Show_24G_Image(Freq_buffer);

        while((Key_Value&0x08)==0x00)
        {
            Scan_Key();
            if(RF_Status[0])
            {
                RF_Buffer[2] |= 0x1F;
                RF_Buffer[17] |= 0x01;
                Config_TRW_24N();
                for(Freq_buffer=0;Freq_buffer<500;Freq_buffer++);
                i =0;
            }
        }
    }
}

```


Features

Compatible with MCS-51[®] Products
4K Bytes of In-System Programmable (ISP) Flash Memory
– Endurance: 1000 Write/Erase Cycles
4.0V to 5.5V Operating Range
Fully Static Operation: 0 Hz to 33 MHz
Three-level Program Memory Lock
128 x 8-bit Internal RAM
32 Programmable I/O Lines
Two 16-bit Timer/Counters
Six Interrupt Sources
Full Duplex UART Serial Channel
Low-power Idle and Power-down Modes
Interrupt Recovery from Power-down Mode
Watchdog Timer
Dual Data Pointer
Power-off Flag
Fast Programming Time
Flexible ISP Programming (Byte and Page Mode)

Description

The AT89S51 is a low-power, high-performance CMOS 8-bit microcontroller with 4K bytes of in-system programmable Flash memory. The device is manufactured using Atmel's high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory programmer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S51 is a powerful microcontroller which provides a highly-flexible and cost-effective solution to many embedded control applications.

The AT89S51 provides the following standard features: 4K bytes of Flash, 128 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, two 16-bit timer/counters, a five-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and lock circuitry. In addition, the AT89S51 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM contents but freezes the oscillator, disabling all other chip functions until the next external interrupt or hardware reset.



**8-bit
Microcontroller
with 4K Bytes
In-System
Programmable
Flash**

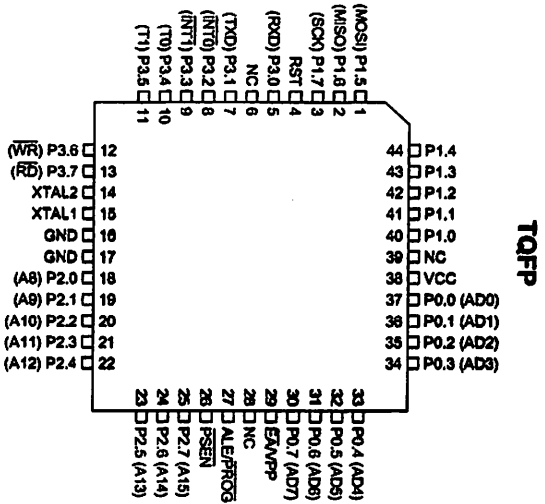
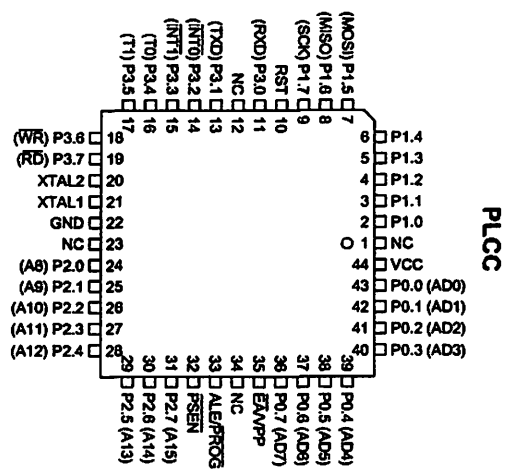
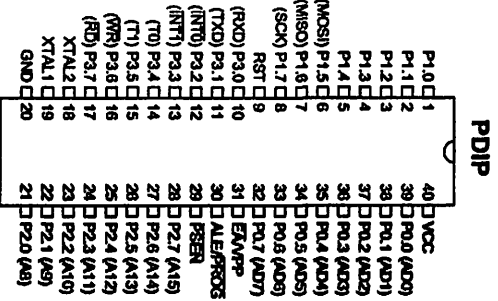
AT89S51

Rev. 2487A-10/01



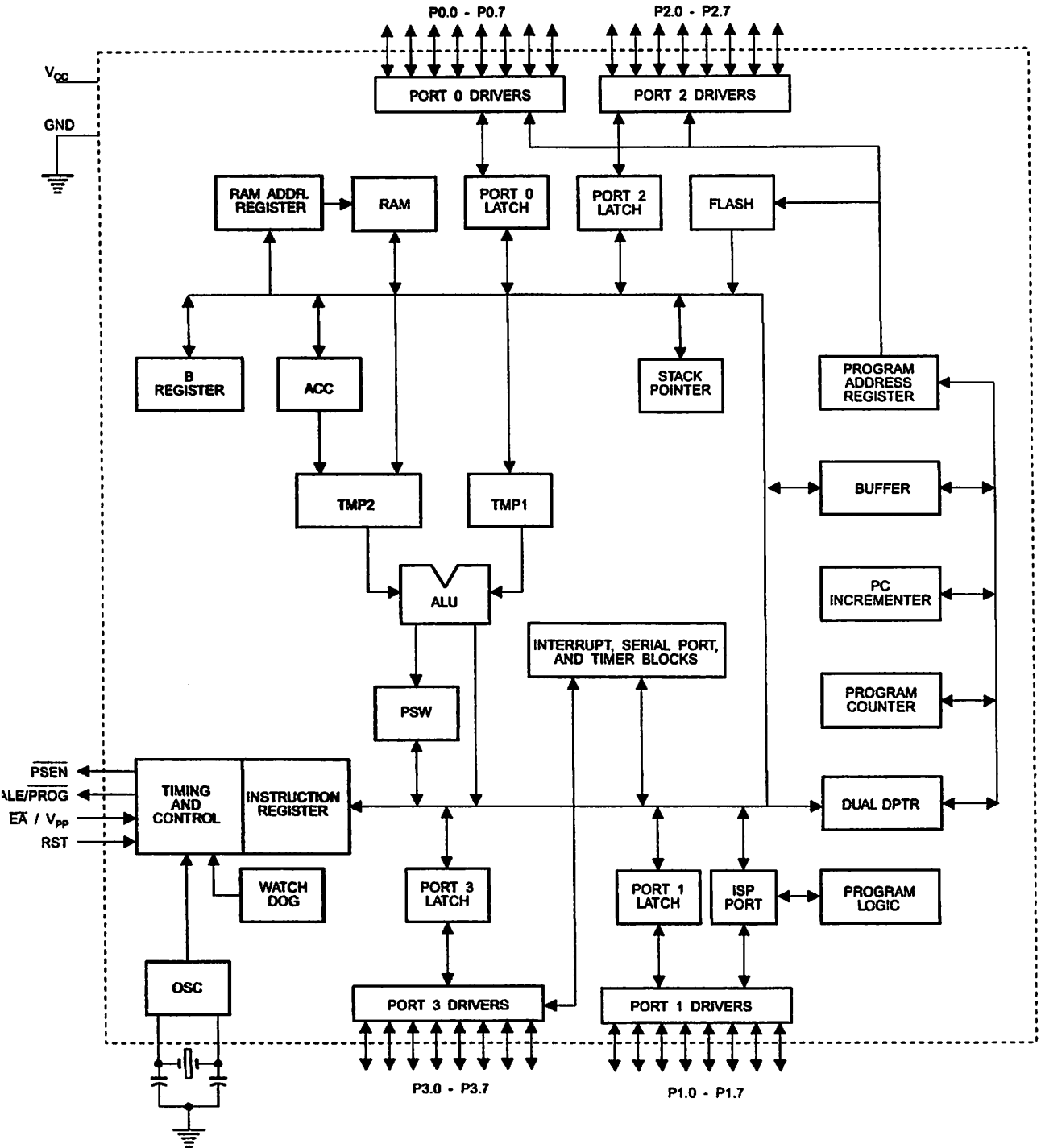


in Configurations



AT89S51

Block Diagram





Pin Description

VCC Supply voltage.

VND Ground.

Port 0 Port 0 is an 8-bit open drain bidirectional I/O port. As an output port, each pin can sink eight TTL inputs. When 1s are written to port 0 pins, the pins can be used as high-impedance inputs.

Port 0 can also be configured to be the multiplexed low-order address/data bus during accesses to external program and data memory. In this mode, P0 has internal pull-ups.

Port 0 also receives the code bytes during Flash programming and outputs the code bytes during program verification. External pull-ups are required during program verification.

Port 1 Port 1 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 1 output buffers can sink/source four TTL inputs. When 1s are written to Port 1 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 1 pins that are externally being pulled low will source current (I_{IL}) because of the internal pull-ups.

Port 1 also receives the low-order address bytes during Flash programming and verification.

Port Pin	Alternate Functions
P1.5	MOSI (used for In-System Programming)
P1.6	MISO (used for In-System Programming)
P1.7	SCK (used for In-System Programming)

Port 2 Port 2 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 2 output buffers can sink/source four TTL inputs. When 1s are written to Port 2 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 2 pins that are externally being pulled low will source current (I_{IL}) because of the internal pull-ups.

Port 2 emits the high-order address byte during fetches from external program memory and during accesses to external data memory that use 16-bit addresses (MOVX @ DPTR). In this application, Port 2 uses strong internal pull-ups when emitting 1s. During accesses to external data memory that use 8-bit addresses (MOVX @ RI), Port 2 emits the contents of the P2 Special Function Register.

Port 2 also receives the high-order address bits and some control signals during Flash programming and verification.

Port 3 Port 3 is an 8-bit bidirectional I/O port with internal pull-ups. The Port 3 output buffers can sink/source four TTL inputs. When 1s are written to Port 3 pins, they are pulled high by the internal pull-ups and can be used as inputs. As inputs, Port 3 pins that are externally being pulled low will source current (I_{IL}) because of the pull-ups.

Port 3 receives some control signals for Flash programming and verification.

Port 3 also serves the functions of various special features of the AT89S51, as shown in the following table.

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

Reset input. A high on this pin for two machine cycles while the oscillator is running resets the device. This pin drives High for 98 oscillator periods after the Watchdog times out. The DISRTO bit in SFR AUXR (address 8EH) can be used to disable this feature. In the default state of bit DISRTO, the RESET HIGH out feature is enabled.

Address Latch Enable (ALE) is an output pulse for latching the low byte of the address during accesses to external memory. This pin is also the program pulse input ($\overline{\text{PROG}}$) during Flash programming.

In normal operation, ALE is emitted at a constant rate of 1/6 the oscillator frequency and may be used for external timing or clocking purposes. Note, however, that one ALE pulse is skipped during each access to external data memory.

If desired, ALE operation can be disabled by setting bit 0 of SFR location 8EH. With the bit set, ALE is active only during a MOVX or MOVC instruction. Otherwise, the pin is weakly pulled high. Setting the ALE-disable bit has no effect if the microcontroller is in external execution mode.

Program Store Enable ($\overline{\text{PSEN}}$) is the read strobe to external program memory.

When the AT89S51 is executing code from external program memory, $\overline{\text{PSEN}}$ is activated twice each machine cycle, except that two $\overline{\text{PSEN}}$ activations are skipped during each access to external data memory.

External Access Enable. $\overline{\text{EA}}$ must be strapped to GND in order to enable the device to fetch code from external program memory locations starting at 0000H up to FFFFH. Note, however, that if lock bit 1 is programmed, $\overline{\text{EA}}$ will be internally latched on reset.

$\overline{\text{EA}}$ should be strapped to V_{CC} for internal program executions.

This pin also receives the 12-volt programming enable voltage (V_{PP}) during Flash programming.

Input to the inverting oscillator amplifier and input to the internal clock operating circuit.

Output from the inverting oscillator amplifier





Special Function Registers

A map of the on-chip memory area called the Special Function Register (SFR) space is shown in Table 1.

Note that not all of the addresses are occupied, and unoccupied addresses may not be implemented on the chip. Read accesses to these addresses will in general return random data, and write accesses will have an indeterminate effect.

Table 1. AT89S51 SFR Map and Reset Values

0F8H									0FFH
0F0H	B 00000000								0F7H
0E8H									0EFH
0E0H	ACC 00000000								0E7H
0D8H									0DFH
0D0H	PSW 00000000								0D7H
0C8H									0CFH
0C0H									0C7H
0B8H	IP XX000000								0BFH
0B0H	P3 11111111								0B7H
0A8H	IE 0X000000								0AFH
0A0H	P2 11111111		AUXR1 XXXXXX0				WDRST XXXXXXXX		0A7H
98H	SCON 00000000	SBUF XXXXXXXX							9FH
90H	P1 11111111								97H
88H	TCON 00000000	TMOD 00000000	TL0 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR XX00XX0		8FH
80H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000		PCON 0XXX0000	87H

User software should not write 1s to these unlisted locations, since they may be used in future products to invoke new features. In that case, the reset or inactive values of the new bits will always be 0.

Interrupt Registers: The individual interrupt enable bits are in the IE register. Two priorities can be set for each of the five interrupt sources in the IP register.

Table 2. AUXR: Auxiliary Register

AUXR		Address = 8EH					Reset Value = XXX00XX0B	
Not Bit Addressable								
Bit	7	6	5	4	3	2	1	0
–	–	–	WDIDLE	DISRTO	–	–	DISALE	
–	Reserved for future expansion							
DISALE	Disable/Enable ALE							
	DISALE							
	Operating Mode							
	0	ALE is emitted at a constant rate of 1/6 the oscillator frequency						
	1	ALE is active only during a MOVX or MOVC instruction						
DISRTO	Disable/Enable Reset out							
	DISRTO							
	0	Reset pin is driven High after WDT times out						
	1	Reset pin is input only						
WDIDLE	Disable/Enable WDT in IDLE mode							
	WDIDLE							
	0	WDT continues to count in IDLE mode						
	1	WDT halts counting in IDLE mode						

Dual Data Pointer Registers: To facilitate accessing both internal and external data memory, two banks of 16-bit Data Pointer Registers are provided: DP0 at SFR address locations 82H-83H and DP1 at 84H-85H. Bit DPS = 0 in SFR AUXR1 selects DP0 and DPS = 1 selects DP1. The user should always initialize the DPS bit to the appropriate value before accessing the respective Data Pointer Register.





Power Off Flag: The Power Off Flag (POF) is located at bit 4 (PCON.4) in the PCON SFR. POF is set to "1" during power up. It can be set and rest under software control and is not affected by reset.

Table 3. AUXR1: Auxiliary Register 1

AUXR1							
Address = A2H							
Reset Value = XXXXXX0B							
Not Bit Addressable							
Bit	-	-	-	-	-	-	DPS
	7	6	5	4	3	2	1
-	Reserved for future expansion						
DPS	Data Pointer Register Select						
	DPS						
	0	Selects DPTR Registers DP0L, DP0H					
	1	Selects DPTR Registers DP1L, DP1H					

Memory Organization

MCS-51 devices have a separate address space for Program and Data Memory. Up to 64K bytes each of external Program and Data Memory can be addressed.

Program Memory

If the \overline{EA} pin is connected to GND, all program fetches are directed to external memory.

On the AT89S51, if \overline{EA} is connected to V_{CC} , program fetches to addresses 0000H through FFFH are directed to internal memory and fetches to addresses 1000H through FFFFH are directed to external memory.

Data Memory

The AT89S51 implements 128 bytes of on-chip RAM. The 128 bytes are accessible via direct and indirect addressing modes. Stack operations are examples of indirect addressing, so the 128 bytes of data RAM are available as stack space.

Watchdog

Timer

One-time

Enabled with

(Reset-out)

The WDT is intended as a recovery method in situations where the CPU may be subjected to software upsets. The WDT consists of a 14-bit counter and the Watchdog Timer Reset (WDTRST) SFR. The WDT is defaulted to disable from exiting reset. To enable the WDT, a user must write 01EH and 0E1H in sequence to the WDTRST register (SFR location 0A6H). When the WDT is enabled, it will increment every machine cycle while the oscillator is running. The WDT timeout period is dependent on the external clock frequency. There is no way to disable the WDT except through reset (either hardware reset or WDT overflow reset). When WDT overflows, it will drive an output RESET HIGH pulse at the RST pin.

Using the WDT

To enable the WDT, a user must write 01EH and 0E1H in sequence to the WDTRST register (SFR location 0A6H). When the WDT is enabled, the user needs to service it by writing 01EH and 0E1H to WDTRST to avoid a WDT overflow. The 14-bit counter overflows when it reaches 16383 (3FFFH), and this will reset the device. When the WDT is enabled, it will increment every machine cycle while the oscillator is running. This means the user must reset the WDT at least every 16383 machine cycles. To reset the WDT the user must write 01EH and 0E1H to WDTRST. WDTRST is a write-only register. The WDT counter cannot be read or written. When WDT overflows, it will generate an output RESET pulse at the RST pin. The RESET pulse duration is $98 \times TOSC$, where $TOSC = 1/FOSC$. To make the best use of the WDT, it

should be serviced in those sections of code that will periodically be executed within the time required to prevent a WDT reset.

WDT During Power-down and Idle

In Power-down mode the oscillator stops, which means the WDT also stops. While in Power-down mode, the user does not need to service the WDT. There are two methods of exiting Power-down mode: by a hardware reset or via a level-activated external interrupt, which is enabled prior to entering Power-down mode. When Power-down is exited with hardware reset, servicing the WDT should occur as it normally does whenever the AT89S51 is reset. Exiting Power-down with an interrupt is significantly different. The interrupt is held low long enough for the oscillator to stabilize. When the interrupt is brought high, the interrupt is serviced. To prevent the WDT from resetting the device while the interrupt pin is held low, the WDT is not started until the interrupt is pulled high. It is suggested that the WDT be reset during the interrupt service for the interrupt used to exit Power-down mode.

To ensure that the WDT does not overflow within a few states of exiting Power-down, it is best to reset the WDT just before entering Power-down mode.

Before going into the IDLE mode, the WDIDLE bit in SFR AUXR is used to determine whether the WDT continues to count if enabled. The WDT keeps counting during IDLE (WDIDLE bit = 0) as the default state. To prevent the WDT from resetting the AT89S51 while in IDLE mode, the user should always set up a timer that will periodically exit IDLE, service the WDT, and reenter IDLE mode.

With WDIDLE bit enabled, the WDT will stop to count in IDLE mode and resumes the count upon exit from IDLE.

UART

The UART in the AT89S51 operates the same way as the UART in the AT89C51. For further information on the UART operation, refer to the ATMEL Web site (<http://www.atmel.com>). From the home page, select 'Products', then '8051-Architecture Flash Microcontroller', then 'Product Overview'.

Timer 0 and 1

Timer 0 and Timer 1 in the AT89S51 operate the same way as Timer 0 and Timer 1 in the AT89C51. For further information on the timers' operation, refer to the ATMEL Web site (<http://www.atmel.com>). From the home page, select 'Products', then '8051-Architecture Flash Microcontroller', then 'Product Overview'.

Interrupts

The AT89S51 has a total of five interrupt vectors: two external interrupts ($\overline{INT0}$ and $\overline{INT1}$), two timer interrupts (Timers 0 and 1), and the serial port interrupt. These interrupts are all shown in Figure 1.

Each of these interrupt sources can be individually enabled or disabled by setting or clearing a bit in Special Function Register IE. IE also contains a global disable bit, EA, which disables all interrupts at once.

Note that Table 4 shows that bit position IE.6 is unimplemented. In the AT89S51, bit position IE.5 is also unimplemented. User software should not write 1s to these bit positions, since they may be used in future AT89 products.

The Timer 0 and Timer 1 flags, TF0 and TF1, are set at S5P2 of the cycle in which the timers overflow. The values are then polled by the circuitry in the next cycle





Table 4. Interrupt Enable (IE) Register

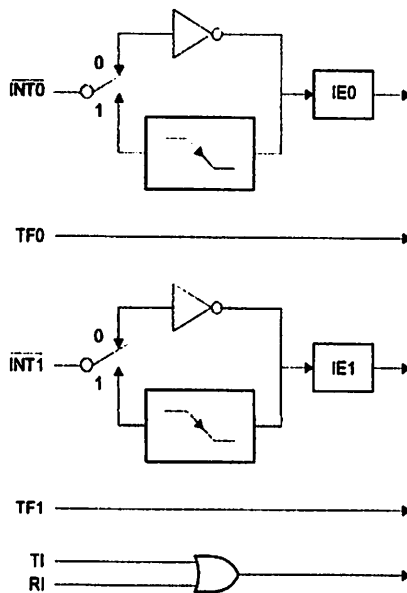
(MSB)				(LSB)			
EA	-	-	ES	ET1	EX1	ET0	EX0

Enable Bit = 1 enables the interrupt.
 Enable Bit = 0 disables the interrupt.

Symbol	Position	Function
EA	IE.7	Disables all interrupts. If EA = 0, no interrupt is acknowledged. If EA = 1, each interrupt source is individually enabled or disabled by setting or clearing its enable bit.
-	IE.6	Reserved
-	IE.5	Reserved
ES	IE.4	Serial Port interrupt enable bit
ET1	IE.3	Timer 1 interrupt enable bit
EX1	IE.2	External interrupt 1 enable bit
ET0	IE.1	Timer 0 interrupt enable bit
EX0	IE.0	External interrupt 0 enable bit

User software should never write 1s to reserved bits, because they may be used in future AT89 products.

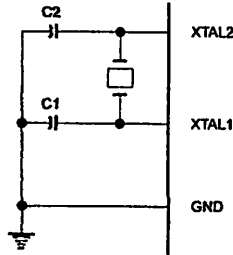
Figure 1. Interrupt Sources



Oscillator Characteristics

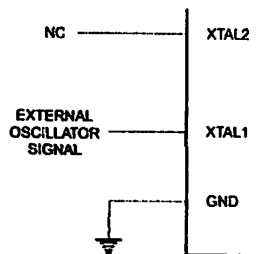
XTAL1 and XTAL2 are the input and output, respectively, of an inverting amplifier that can be configured for use as an on-chip oscillator, as shown in Figure 2. Either a quartz crystal or ceramic resonator may be used. To drive the device from an external clock source, XTAL2 should be left unconnected while XTAL1 is driven, as shown in Figure 3. There are no requirements on the duty cycle of the external clock signal, since the input to the internal clocking circuitry is through a divide-by-two flip-flop, but minimum and maximum voltage high and low time specifications must be observed.

Figure 2. Oscillator Connections



Note: C1, C2 = 30 pF ± 10 pF for Crystals = 40 pF ± 10 pF for Ceramic Resonators

Figure 3. External Clock Drive Configuration



Idle Mode

In idle mode, the CPU puts itself to sleep while all the on-chip peripherals remain active. The mode is invoked by software. The content of the on-chip RAM and all the special function registers remain unchanged during this mode. The idle mode can be terminated by any enabled interrupt or by a hardware reset.

Note that when idle mode is terminated by a hardware reset, the device normally resumes program execution from where it left off, up to two machine cycles before the internal reset algorithm takes control. On-chip hardware inhibits access to internal RAM in this event, but access to the port pins is not inhibited. To eliminate the possibility of an unexpected write to a port pin when idle mode is terminated by a reset, the instruction following the one that invokes idle mode should not write to a port pin or to external memory.

Power-down Mode

In the Power-down mode, the oscillator is stopped, and the instruction that invokes Power-down is the last instruction executed. The on-chip RAM and Special Function Registers retain their values until the Power-down mode is terminated. Exit from Power-down mode can be initiated either by a hardware reset or by activation of an enabled external interrupt into INT0 or INT1. Reset redefines the SFRs but does not change the on-chip RAM. The reset should not be activated before V_{CC} is restored to its normal operating level and must be held active long enough to allow the oscillator to restart and stabilize.





Table 5. Status of External Pins During Idle and Power-down Modes

Mode	Program Memory	ALE	PSEN	PORT0	PORT1	PORT2	PORT3
Idle	Internal	1	1	Data	Data	Data	Data
Idle	External	1	1	Float	Data	Address	Data
Power-down	Internal	0	0	Data	Data	Data	Data
Power-down	External	0	0	Float	Data	Data	Data

**Program
Memory Lock
bits**

The AT89S51 has three lock bits that can be left unprogrammed (U) or can be programmed (P) to obtain the additional features listed in the following table.

Table 6. Lock Bit Protection Modes

Program Lock Bits				Protection Type
	LB1	LB2	LB3	
1	U	U	U	No program lock features
2	P	U	U	MOVX instructions executed from external program memory are disabled from fetching code bytes from internal memory, \overline{EA} is sampled and latched on reset, and further programming of the Flash memory is disabled
3	P	P	U	Same as mode 2, but verify is also disabled
4	P	P	P	Same as mode 3, but external execution is also disabled

When lock bit 1 is programmed, the logic level at the \overline{EA} pin is sampled and latched during reset. If the device is powered up without a reset, the latch initializes to a random value and holds that value until reset is activated. The latched value of \overline{EA} must agree with the current logic level at that pin in order for the device to function properly.

**Programming
the Flash –
Parallel Mode**

The AT89S51 is shipped with the on-chip Flash memory array ready to be programmed. The programming interface needs a high-voltage (12-volt) program enable signal and is compatible with conventional third-party Flash or EPROM programmers.

The AT89S51 code memory array is programmed byte-by-byte.

Programming Algorithm: Before programming the AT89S51, the address, data, and control signals should be set up according to the Flash programming mode table and Figures 13 and 14. To program the AT89S51, take the following steps:

1. Input the desired memory location on the address lines.
2. Input the appropriate data byte on the data lines.
3. Activate the correct combination of control signals.
4. Raise \overline{EAV}_{pp} to 12V.
5. Pulse ALE/ \overline{PROG} once to program a byte in the Flash array or the lock bits. The byte-write cycle is self-timed and typically takes no more than 50 μ s. Repeat steps 1 through 5, changing the address and data for the entire array or until the end of the object file is reached.

Data Polling: The AT89S51 features \overline{Data} Polling to indicate the end of a byte write cycle. During a write cycle, an attempted read of the last byte written will result in the complement of the written data on P0.7. Once the write cycle has been completed, true data is valid on all outputs, and the next cycle may begin. \overline{Data} Polling may begin any time after a write cycle has been initiated.

Ready/Busy: The progress of byte programming can also be monitored by the RDY/ $\overline{\text{BSY}}$ output signal. P3.0 is pulled low after ALE goes high during programming to indicate $\overline{\text{BUSY}}$. P3.0 is pulled high again when programming is done to indicate READY.

Program Verify: If lock bits LB1 and LB2 have not been programmed, the programmed code data can be read back via the address and data lines for verification. The status of the individual lock bits can be verified directly by reading them back.

Reading the Signature Bytes: The signature bytes are read by the same procedure as a normal verification of locations 000H, 100H, and 200H, except that P3.6 and P3.7 must be pulled to a logic low. The values returned are as follows.

- (000H) = 1EH indicates manufactured by Atmel
- (100H) = 51H indicates 89S51
- (200H) = 06H

Chip Erase: In the parallel programming mode, a chip erase operation is initiated by using the proper combination of control signals and by pulsing ALE/PROG low for a duration of 200 ns - 500 ns.

In the serial programming mode, a chip erase operation is initiated by issuing the Chip Erase instruction. In this mode, chip erase is self-timed and takes about 500 ms.

During chip erase, a serial read from any address location will return 00H at the data output.

**rogramming
ne Flash –
erial Mode**

The Code memory array can be programmed using the serial ISP interface while RST is pulled to V_{CC} . The serial interface consists of pins SCK, MOSI (input) and MISO (output). After RST is set high, the Programming Enable instruction needs to be executed first before other operations can be executed. Before a reprogramming sequence can occur, a Chip Erase operation is required.

The Chip Erase operation turns the content of every memory location in the Code array into FFH.

Either an external system clock can be supplied at pin XTAL1 or a crystal needs to be connected across pins XTAL1 and XTAL2. The maximum serial clock (SCK) frequency should be less than 1/16 of the crystal frequency. With a 33 MHz oscillator clock, the maximum SCK frequency is 2 MHz.

**erial
rogramming
gorithm**

To program and verify the AT89S51 in the serial programming mode, the following sequence is recommended:

1. **Power-up sequence:**
Apply power between VCC and GND pins.
Set RST pin to "H".
If a crystal is not connected across pins XTAL1 and XTAL2, apply a 3 MHz to 33 MHz clock to XTAL1 pin and wait for at least 10 milliseconds.
2. Enable serial programming by sending the Programming Enable serial instruction to pin MOSI/P1.5. The frequency of the shift clock supplied at pin SCK/P1.7 needs to be less than the CPU clock at XTAL1 divided by 16.
3. The Code array is programmed one byte at a time in either the Byte or Page mode. The write cycle is self-timed and typically takes less than 0.5 ms at 5V.
4. Any memory location can be verified by using the Read instruction that returns the content at the selected address at serial output MISO/P1.6.
5. At the end of a programming session, RST can be set low to commence normal device operation.





Power-off sequence (if needed):

Set XTAL1 to "L" (if a crystal is not used).

Set RST to "L".

Turn V_{CC} power off.

Data Polling: The Data Polling feature is also available in the serial mode. In this mode, during a write cycle an attempted read of the last byte written will result in the complement of the MSB of the serial output byte on MISO.

Serial Programming Instruction Set






The Instruction Set for Serial Programming follows a 4-byte protocol and is shown in Table 8 on page 18.

Programming Interface – Parallel Mode

Every code byte in the Flash array can be programmed by using the appropriate combination of control signals. The write operation cycle is self-timed and once initiated, will automatically time itself to completion.

All major programming vendors offer worldwide support for the Atmel microcontroller series. Please contact your local programming vendor for the appropriate software revision.

Table 7. Flash Programming Modes

Mode	V_{CC}	RST	PSEN	ALE/ PROG	\overline{EA} / V_{PP}	P2.6	P2.7	P3.3	P3.6	P3.7	P0.7-0 Data	P2.3-0	P1.7-0
												Address	
Write Code Data	5V	H	L		12V	L	H	H	H	H	D_{IN}	A11-8	A7-0
Read Code Data	5V	H	L	H	H	L	L	L	H	H	D_{OUT}	A11-8	A7-0
Write Lock Bit 1	5V	H	L		12V	H	H	H	H	H	X	X	X
Write Lock Bit 2	5V	H	L		12V	H	H	H	L	L	X	X	X
Write Lock Bit 3	5V	H	L		12V	H	L	H	H	L	X	X	X
Read Lock Bits 1, 2, 3	5V	H	L	H	H	H	H	L	H	L	P0.2, P0.3, P0.4	X	X
Chip Erase	5V	H	L		12V	H	L	H	L	L	X	X	X
Read Atmel ID	5V	H	L	H	H	L	L	L	L	L	1EH	0000	00H
Read Device ID	5V	H	L	H	H	L	L	L	L	L	51H	0001	00H
Read Device ID	5V	H	L	H	H	L	L	L	L	L	06H	0010	00H

- Notes:
1. Each \overline{PROG} pulse is 200 ns - 500 ns for Chip Erase.
 2. Each \overline{PROG} pulse is 200 ns - 500 ns for Write Code Data.
 3. Each \overline{PROG} pulse is 200 ns - 500 ns for Write Lock Bits.
 4. $\overline{RDY/BSY}$ signal is output on P3.0 during programming.
 5. X = don't care.

Figure 4. Programming the Flash Memory (Parallel Mode)

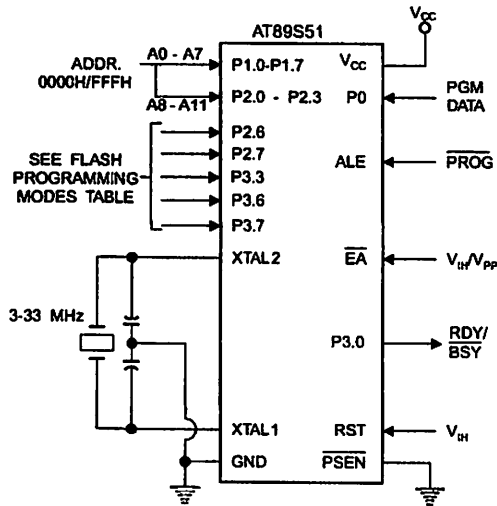
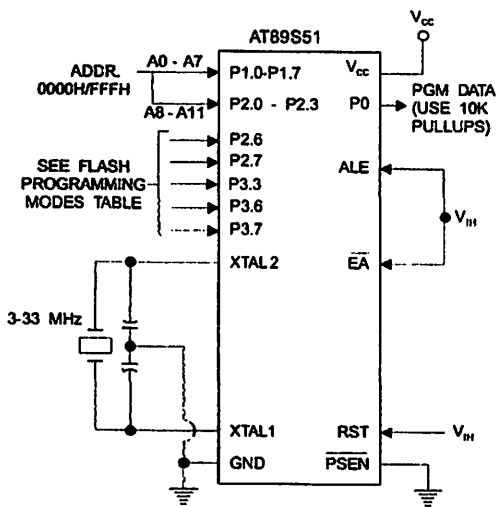


Figure 5. Verifying the Flash Memory (Parallel Mode)



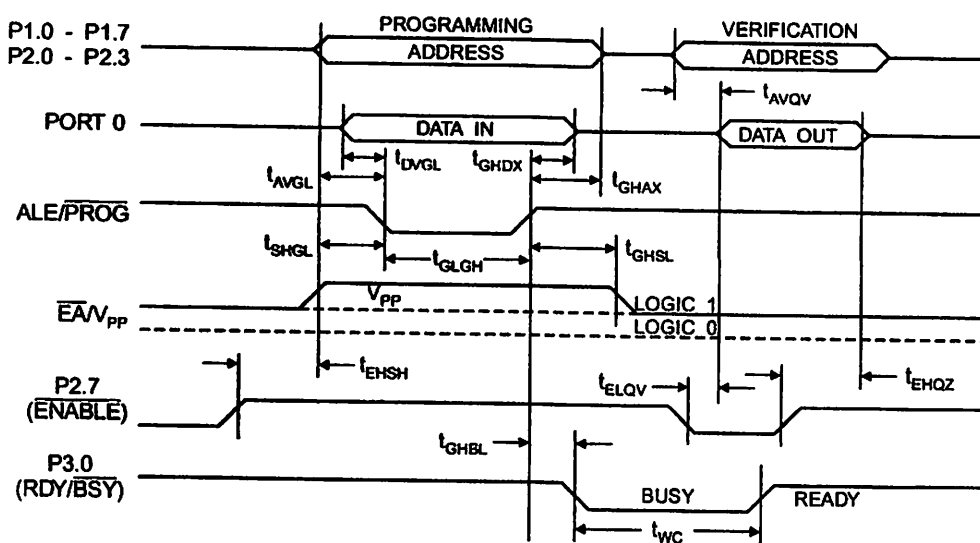


Flash Programming and Verification Characteristics (Parallel Mode)

$T = 20^{\circ}\text{C to } 30^{\circ}\text{C}, V_{CC} = 4.5 \text{ to } 5.5\text{V}$

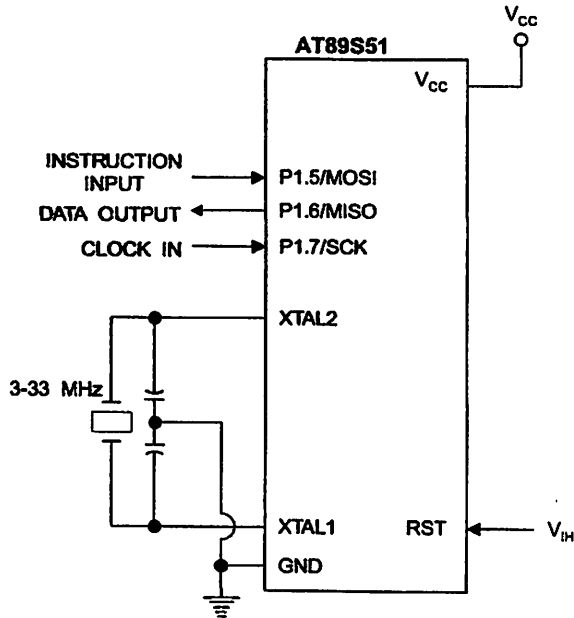
Symbol	Parameter	Min	Max	Units
V_{PP}	Programming Supply Voltage	11.5	12.5	V
I_{PP}	Programming Supply Current		10	mA
I_{CC}	V_{CC} Supply Current		30	mA
f_{CLCL}	Oscillator Frequency	3	33	MHz
t_{AVGL}	Address Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHAX}	Address Hold After $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{DVGL}	Data Setup to $\overline{\text{PROG}}$ Low	$48t_{CLCL}$		
t_{GHDX}	Data Hold After $\overline{\text{PROG}}$	$48t_{CLCL}$		
t_{HSH}	P2.7 (ENABLE) High to V_{PP}	$48t_{CLCL}$		
t_{SHGL}	V_{PP} Setup to $\overline{\text{PROG}}$ Low	10		μs
t_{HSL}	V_{PP} Hold After $\overline{\text{PROG}}$	10		μs
t_{GLGH}	$\overline{\text{PROG}}$ Width	0.2	1	μs
t_{AVQV}	Address to Data Valid		$48t_{CLCL}$	
t_{LQV}	$\overline{\text{ENABLE}}$ Low to Data Valid		$48t_{CLCL}$	
t_{EQZ}	Data Float After $\overline{\text{ENABLE}}$	0	$48t_{CLCL}$	
t_{HBL}	$\overline{\text{PROG}}$ High to $\overline{\text{BUSY}}$ Low		1.0	μs
t_{WC}	Byte Write Cycle Time		50	μs

Figure 6. Flash Programming and Verification Waveforms – Parallel Mode



AT89S51

Figure 7. Flash Memory Serial Downloading



Flash Programming and Verification Waveforms – Serial Mode

Figure 8. Serial Programming Waveforms

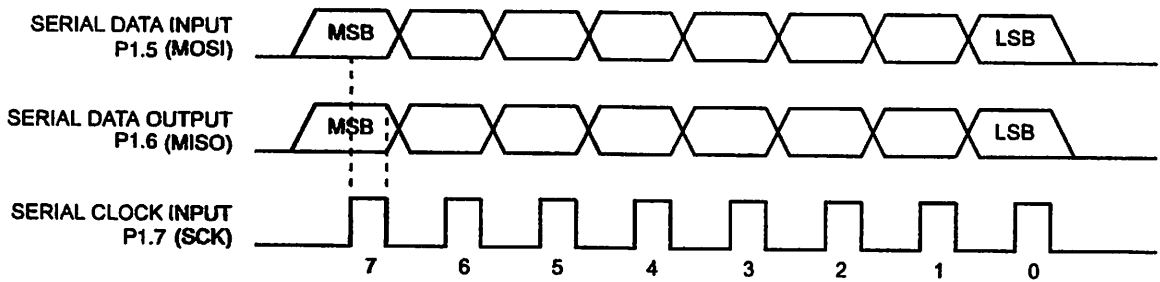




Table 8. Serial Programming Instruction Set

Instruction	Instruction Format				Operation
	Byte 1	Byte 2	Byte 3	Byte 4	
Programming Enable	1010 1100	0101 0011	xxxx xxxx	xxxx xxxx 0110 1001 (Output)	Enable Serial Programming while RST is high
Chip Erase	1010 1100	100x xxxx	xxxx xxxx	xxxx xxxx	Chip Erase Flash memory array
Read Program Memory (Byte Mode)	0010 0000	xxxx A11 A10 A9 A8	A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Read data from Program memory in the byte mode
Write Program Memory (Byte Mode)	0100 0000	xxxx A11 A10 A9 A8	A6 A5 A4 A3 A2 A1 A0	D7 D6 D5 D4 D3 D2 D1 D0	Write data to Program memory in the byte mode
Write Lock Bits ⁽²⁾	1010 1100	1110 00 B1 B2	xxxx xxxx	xxxx xxxx	Write Lock bits. See Note (2).
Read Lock Bits	0010 0100	xxxx xxxx	xxxx xxxx	xx B2 B1 xx	Read back current status of the lock bits (a programmed lock bit reads back as a "1")
Read Signature Bytes ⁽¹⁾	0010 1000	xxx A2 A3 A2 A1	A0 xxx xxxx	Signature Byte	Read Signature Byte
Read Program Memory (Page Mode)	0011 0000	xxxx A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Read data from Program memory in the Page Mode (256 bytes)
Write Program Memory (Page Mode)	0101 0000	xxxx A11 A10 A9 A8	Byte 0	Byte 1... Byte 255	Write data to Program memory in the Page Mode (256 bytes)

Notes: 1. The signature bytes are not readable in Lock Bit Modes 3 and 4.

- 2. B1 = 0, B2 = 0 → Mode 1, no lock protection
- B1 = 0, B2 = 1 → Mode 2, lock bit 1 activated
- B1 = 1, B2 = 0 → Mode 3, lock bit 2 activated
- B1 = 1, B2 = 1 → Mode 4, lock bit 3 activated



Each of the lock bits needs to be activated sequentially before Mode 4 can be executed.

After Reset signal is high, SCK should be low for at least 64 system clocks before it goes high to clock in the enable data bytes. No pulsing of Reset signal is necessary. SCK should be no faster than 1/16 of the system clock at XTAL1.

For Page Read/Write, the data always starts from byte 0 to 255. After the command byte and upper address byte are latched, each byte thereafter is treated as data until all 256 bytes are shifted in/out. Then the next instruction will be ready to be decoded.

Serial Programming Characteristics

Figure 9. Serial Programming Timing

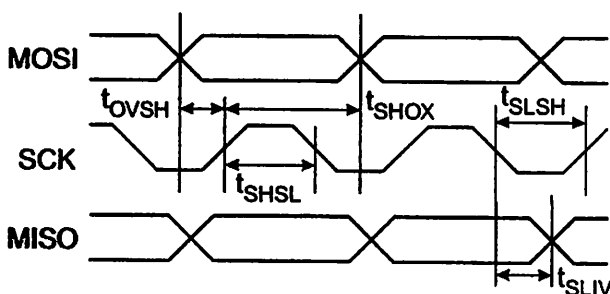


Table 9. Serial Programming Characteristics, $T_A = -40^\circ\text{C}$ to 85°C , $V_{CC} = 4.0 - 5.5\text{V}$ (Unless Otherwise Noted)

Symbol	Parameter	Min	Typ	Max	Units
$1/t_{CLCL}$	Oscillator Frequency	0		33	MHz
t_{CLCL}	Oscillator Period	30			ns
t_{SHSL}	SCK Pulse Width High	$8 t_{CLCL}$			ns
t_{SLSH}	SCK Pulse Width Low	$8 t_{CLCL}$			ns
t_{OVSH}	MOSI Setup to SCK High	t_{CLCL}			ns
t_{SHOX}	MOSI Hold after SCK High	$2 t_{CLCL}$			ns
t_{SLIV}	SCK Low to MISO Valid	10	16	32	ns
t_{ERASE}	Chip Erase Instruction Cycle Time			500	ms
t_{SWC}	Serial Byte Write Cycle Time			$64 t_{CLCL} + 400$	μs



Absolute Maximum Ratings*

Operating Temperature.....	-55°C to +125°C
Storage Temperature.....	-65°C to +150°C
Voltage on Any Pin with Respect to Ground.....	-1.0V to +7.0V
Maximum Operating Voltage.....	6.6V
DC Output Current.....	15.0 mA

*NOTICE: Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

C Characteristics

The values shown in this table are valid for $T_A = -40^\circ\text{C}$ to 85°C and $V_{CC} = 4.0\text{V}$ to 5.5V , unless otherwise noted.

Symbol	Parameter	Condition	Min	Max	Units
V_{IL}	Input Low Voltage	(Except \overline{EA})	-0.5	$0.2 V_{CC} - 0.1$	V
V_{IL1}	Input Low Voltage (\overline{EA})		-0.5	$0.2 V_{CC} - 0.3$	V
V_{IH}	Input High Voltage	(Except XTAL1, RST)	$0.2 V_{CC} + 0.9$	$V_{CC} + 0.5$	V
V_{IH1}	Input High Voltage	(XTAL1, RST)	$0.7 V_{CC}$	$V_{CC} + 0.5$	V
V_{OL}	Output Low Voltage ⁽¹⁾ (Ports 1,2,3)	$I_{OL} = 1.6 \text{ mA}$		0.45	V
V_{OL1}	Output Low Voltage ⁽¹⁾ (Port 0, ALE, \overline{PSEN})	$I_{OL} = 3.2 \text{ mA}$		0.45	V
V_{OH}	Output High Voltage (Ports 1,2,3, ALE, \overline{PSEN})	$I_{OH} = -60 \mu\text{A}$, $V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -25 \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -10 \mu\text{A}$	$0.9 V_{CC}$		V
V_{OH1}	Output High Voltage (Port 0 in External Bus Mode)	$I_{OH} = -800 \mu\text{A}$, $V_{CC} = 5\text{V} \pm 10\%$	2.4		V
		$I_{OH} = -300 \mu\text{A}$	$0.75 V_{CC}$		V
		$I_{OH} = -80 \mu\text{A}$	$0.9 V_{CC}$		V
I_{L}	Logical 0 Input Current (Ports 1,2,3)	$V_{IN} = 0.45\text{V}$		-50	μA
I_{L1}	Logical 1 to 0 Transition Current (Ports 1,2,3)	$V_{IN} = 2\text{V}$, $V_{CC} = 5\text{V} \pm 10\%$		-650	μA
I_{L2}	Input Leakage Current (Port 0, \overline{EA})	$0.45 < V_{IN} < V_{CC}$		± 10	μA
R_{RST}	Reset Pulldown Resistor		50	300	K Ω
C_{IO}	Pin Capacitance	Test Freq. = 1 MHz, $T_A = 25^\circ\text{C}$		10	pF
I_{CC}	Power Supply Current	Active Mode, 12 MHz		25	mA
		Idle Mode, 12 MHz		6.5	mA
	Power-down Mode ⁽²⁾	$V_{CC} = 5.5\text{V}$		50	μA

- Notes:
- Under steady state (non-transient) conditions, I_{OL} must be externally limited as follows:
 Maximum I_{OL} per port pin: 10 mA
 Maximum I_{OL} per 8-bit port:
 Port 0: 26 mA Ports 1, 2, 3: 15 mA
 Maximum total I_{OL} for all output pins: 71 mA
 If I_{OL} exceeds the test condition, V_{OL} may exceed the related specification. Pins are not guaranteed to sink current greater than the listed test conditions.
 - Minimum V_{CC} for Power-down is 2V.

AT89S51

C Characteristics

Under operating conditions, load capacitance for Port 0, ALE/PROG, and PSEN = 100 pF; load capacitance for all other inputs = 80 pF.

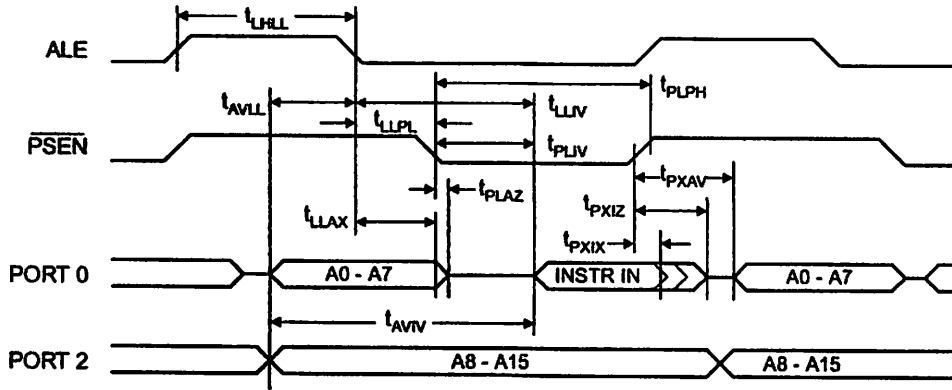
External Program and Data Memory Characteristics

Symbol	Parameter	12 MHz Oscillator		Variable Oscillator		Units
		Min	Max	Min	Max	
f_{osc}/t_{CLCL}	Oscillator Frequency			0	33	MHz
HL	ALE Pulse Width	127		$2t_{CLCL}-40$		ns
AVL	Address Valid to ALE Low	43		$t_{CLCL}-25$		ns
HLAX	Address Hold After ALE Low	48		$t_{CLCL}-25$		ns
HLIV	ALE Low to Valid Instruction In		233		$4t_{CLCL}-65$	ns
HLPL	ALE Low to PSEN Low	43		$t_{CLCL}-25$		ns
PLPH	PSEN Pulse Width	205		$3t_{CLCL}-45$		ns
PLIV	PSEN Low to Valid Instruction In		145		$3t_{CLCL}-60$	ns
PXIX	Input Instruction Hold After PSEN	0		0		ns
PXIZ	Input Instruction Float After PSEN		59		$t_{CLCL}-25$	ns
PXAV	PSEN to Address Valid	75		$t_{CLCL}-8$		ns
AVIV	Address to Valid Instruction In		312		$5t_{CLCL}-80$	ns
PLAZ	PSEN Low to Address Float		10		10	ns
RLRH	RD Pulse Width	400		$6t_{CLCL}-100$		ns
WLWH	WR Pulse Width	400		$6t_{CLCL}-100$		ns
RLDV	RD Low to Valid Data In		252		$5t_{CLCL}-90$	ns
RHDX	Data Hold After RD	0		0		ns
RHDZ	Data Float After RD		97		$2t_{CLCL}-28$	ns
LLDV	ALE Low to Valid Data In		517		$8t_{CLCL}-150$	ns
AVDV	Address to Valid Data In		585		$9t_{CLCL}-165$	ns
LWL	ALE Low to RD or WR Low	200	300	$3t_{CLCL}-50$	$3t_{CLCL}+50$	ns
AVWL	Address to RD or WR Low	203		$4t_{CLCL}-75$		ns
DVWX	Data Valid to WR Transition	23		$t_{CLCL}-30$		ns
DVWH	Data Valid to WR High	433		$7t_{CLCL}-130$		ns
MHQX	Data Hold After WR	33		$t_{CLCL}-25$		ns
RLAZ	RD Low to Address Float		0		0	ns
MHLH	RD or WR High to ALE High	43	123	$t_{CLCL}-25$	$t_{CLCL}+25$	ns

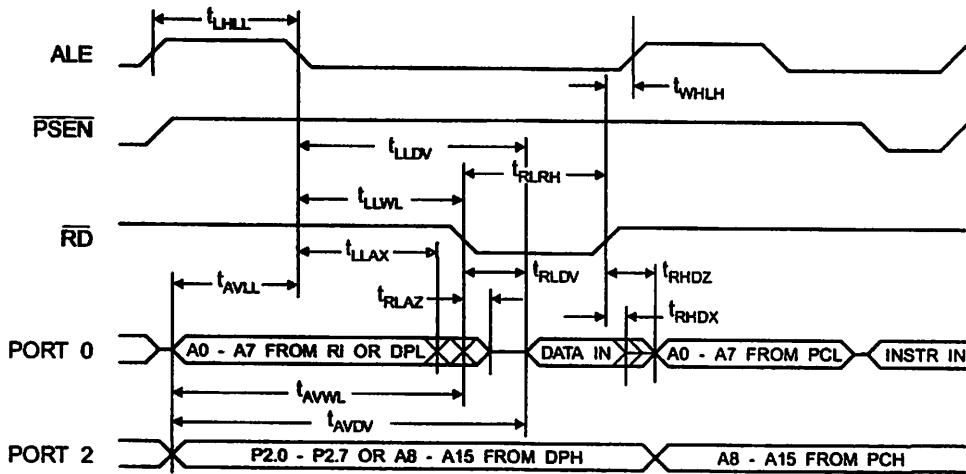




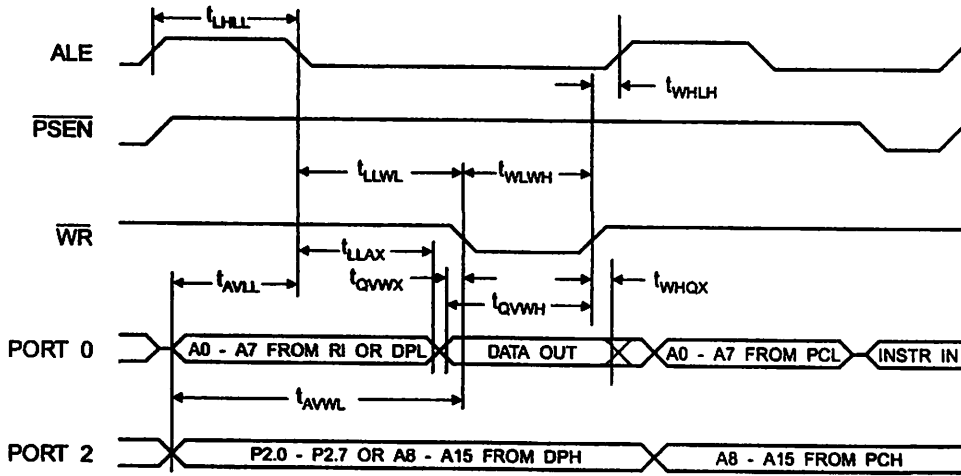
External Program Memory Read Cycle



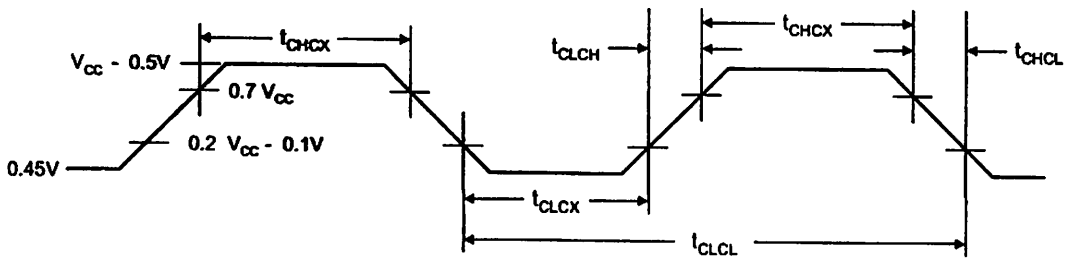
External Data Memory Read Cycle



External Data Memory Write Cycle



External Clock Drive Waveforms



External Clock Drive

Symbol	Parameter	Min	Max	Units
f_{CLCL}	Oscillator Frequency	0	33	MHz
T_{CLCL}	Clock Period	30		ns
t_{CHCX}	High Time	12		ns
t_{CLCX}	Low Time	12		ns
t_{CLCH}	Rise Time		5	ns
t_{CHCL}	Fall Time		5	ns

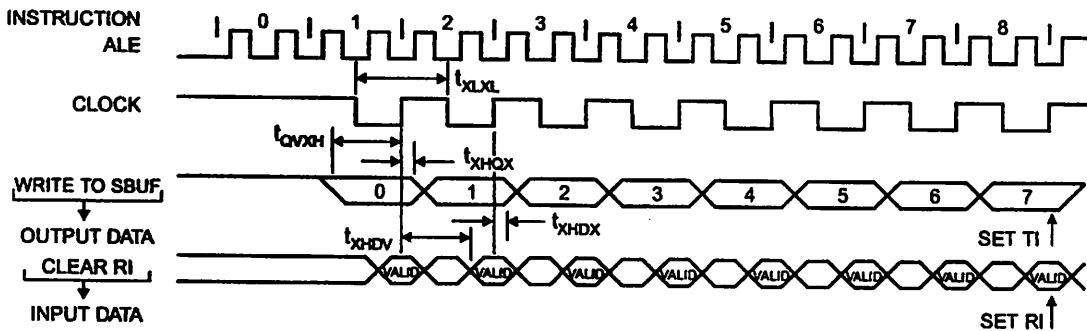


Serial Port Timing: Shift Register Mode Test Conditions

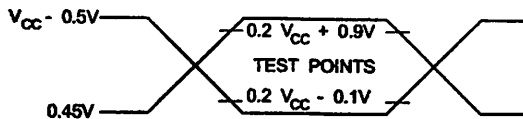
The values in this table are valid for $V_{CC} = 4.0V$ to $5.5V$ and Load Capacitance = 80 pF .

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{CLXL}	Serial Port Clock Cycle Time	1.0		$12t_{CLCL}$		μs
t_{QVXH}	Output Data Setup to Clock Rising Edge	700		$10t_{CLCL}-133$		ns
t_{XHQX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL}-80$		ns
t_{XHDX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHdV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL}-133$	ns

Shift Register Mode Timing Waveforms

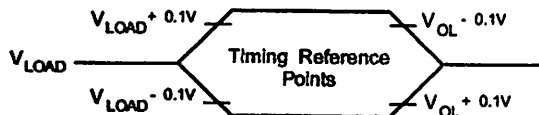


C Testing Input/Output Waveforms⁽¹⁾



Note: 1. AC Inputs during testing are driven at $V_{CC} - 0.5V$ for a logic 1 and $0.45V$ for a logic 0. Timing measurements are made at $V_{IH\text{ min}}$ for a logic 1 and $V_{IL\text{ max}}$ for a logic 0.

Load Waveforms⁽¹⁾



Note: 1. For timing purposes, a port pin is no longer floating when a 100 mV change from load voltage occurs. A port pin begins to float when a 100 mV change from the loaded V_{OH}/V_{OL} level occurs.

Ordering Information

Speed (MHz)	Power Supply	Ordering Code	Package	Operation Range
24	4.0V to 5.5V	AT89S51-24AC	44A	Commercial (0° C to 70° C)
		AT89S51-24JC	44J	
		AT89S51-24PC	40P6	
		AT89S51-24AI	44A	Industrial (-40° C to 85° C)
		AT89S51-24JI	44J	
		AT89S51-24PI	40P6	
33	4.5V to 5.5V	AT89S51-33AC	44A	Commercial (0° C to 70° C)
		AT89S51-33JC	44J	
		AT89S51-33PC	40P6	

 = Preliminary Availability

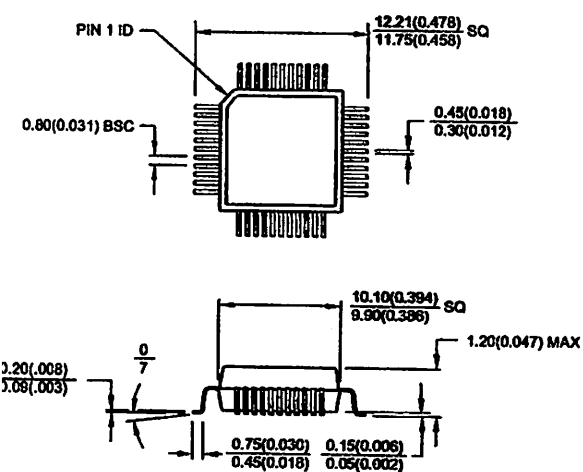
Package Type	
A	44-lead, Thin Plastic Gull Wing Quad Flatpack (TQFP)
J	44-lead, Plastic J-leaded Chip Carrier (PLCC)
P6	40-pin, 0.600" Wide, Plastic Dual Inline Package (PDIP)



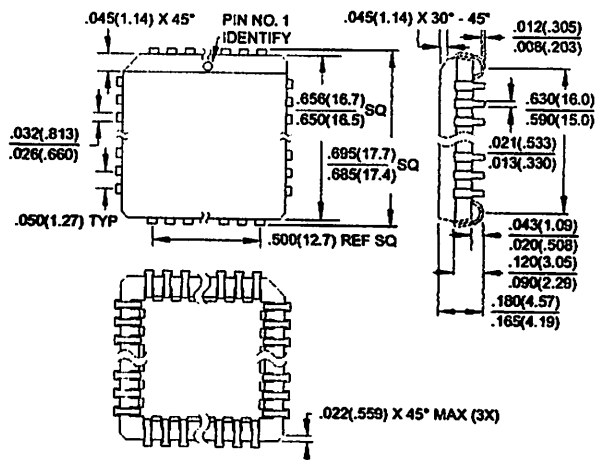


Packaging Information

44A, 44-lead, Thin (1.0 mm) Plastic Gull Wing Quad Flat Package (TQFP)
 Dimensions in Millimeters and (Inches)*

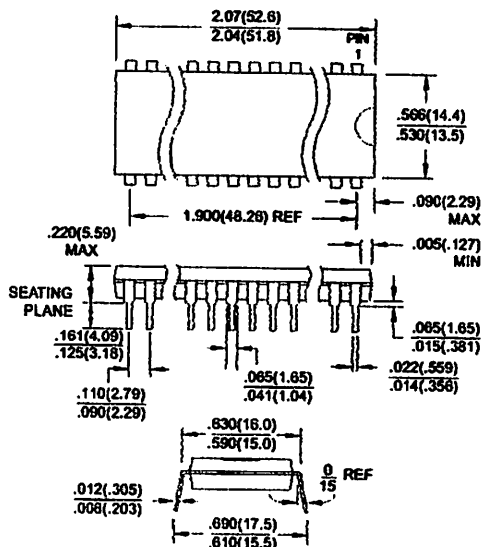


44J, 44-lead, Plastic J-leaded Chip Carrier (PLCC)
 Dimensions in Inches and (Millimeters)



Controlling dimension: millimeters

10P6, 40-pin, 0.600" Wide, Plastic Dual In Line Package (PDIP)
 Dimensions in Inches and (Millimeters)
 EDEC STANDARD MS-011 AC



AT89S51



Atmel Headquarters

Corporate Headquarters
25 Orchard Parkway
Framingham, MA 01937
TEL (408) 441-0311
FAX (408) 487-2600

Atmel Sarl
Route des Arsenaux 41
Case Postale 80
CH-1705 Fribourg
Switzerland
TEL (41) 26-426-5555
FAX (41) 26-426-5500

Atmel Asia, Ltd.
Room 1219
Fincham Golden Plaza
Mody Road Tsimshatsui
Kowloon
Hong Kong
TEL (852) 2721-9778
FAX (852) 2722-1369

Atmel Japan K.K.
1-1-1, Tonetsu Shinkawa Bldg.
24-8 Shinkawa
Chuo-ku, Tokyo 104-0033
Japan
TEL (81) 3-3523-3551
FAX (81) 3-3523-7581

Atmel Product Operations

Atmel Colorado Springs
1150 E. Cheyenne Mtn. Blvd.
Colorado Springs, CO 80906
TEL (719) 576-3300
FAX (719) 540-1759

Atmel Grenoble
Avenue de Rochepleine
BP 123
38521 Saint-Egreve Cedex, France
TEL (33) 4-7658-3000
FAX (33) 4-7658-3480

Atmel Heilbronn
Theresienstrasse 2
POB 3535
D-74025 Heilbronn, Germany
TEL (49) 71 31 67 25 94
FAX (49) 71 31 67 24 23

Atmel Nantes
La Chantrerie
BP 70602
44306 Nantes Cedex 3, France
TEL (33) 0 2 40 18 18 18
FAX (33) 0 2 40 18 19 60

Atmel Rousset
Zone Industrielle
13106 Rousset Cedex, France
TEL (33) 4-4253-6000
FAX (33) 4-4253-6001

Atmel Smart Card ICs
Scottish Enterprise Technology Park
East Kilbride, Scotland G75 0QR
TEL (44) 1355-357-000
FAX (44) 1355-242-743

e-mail
literature@atmel.com

Web Site
<http://www.atmel.com>

Atmel Corporation 2001.

Atmel Corporation makes no warranty for the use of its products, other than those expressly contained in the Company's standard warranty which is detailed in Atmel's Terms and Conditions located on the Company's web site. The Company assumes no responsibility for any errors that may appear in this document, reserves the right to change devices or specifications detailed herein at any time without notice, and does not make any commitment to update the information contained herein. No licenses to patents or other intellectual property of Atmel are granted to any Company in connection with the sale of Atmel products, expressly or by implication. Atmel's products are not authorized for use as critical components in life support devices or systems.

Atmel is the registered trademark of Atmel.

Intel is the registered trademark of Intel Corporation. Terms and product names in this document may be trademarks of others.



Printed on recycled paper.

2487A-10/01/xM

Rangkaian Transmitter

