

SKRIPSI

DESAIN DAN IMPLEMENTASI *BUILDING AUTOMATION SYSTEM* DENGAN WEB DAN WAP



Disusun Oleh

JONATHAN CHRISTIAN BESUNI
NIM: 04.12.212

KONSENTRASI TEKNIK ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
SEPTEMBER 2008

LEMBAR PENGESAHAN

**DESAIN DAN IMPLEMENTASI *BUILDING*
AUTOMATION SYSTEM DENGAN WEB DAN WAP**

SKRIPSI

*Disusun dan Diajukan Untuk Melengkapi dan Memenuhi Syarat
Guna Mencapai Gelar Sarjana Teknik*

Disusun Oleh:

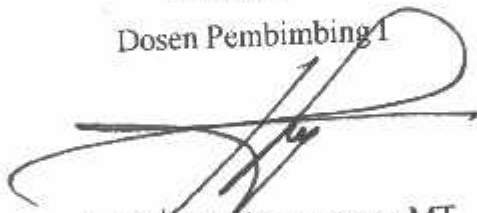
JONATHAN CHRISTIAN BESUNI

NIM: 04.12.212

Malang, Oktober 2008

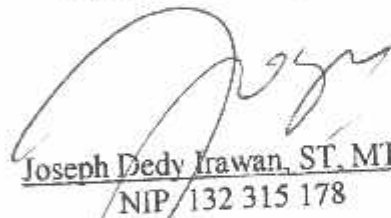
Disetujui

Dosen Pembimbing I



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

Dosen Pembimbing II

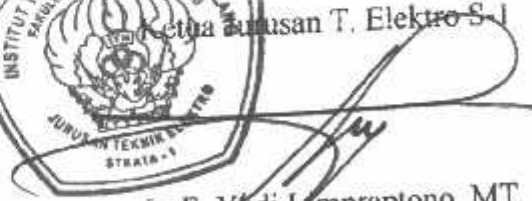


Joseph Dedy Irawan, ST, MT
NIP/ 132 315 178



Mengetahui

Ketua Jurusan T. Elektro S-1



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

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



INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura-gura No. 2
MALANG

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : Jonathan Christian Besuni
NIM : 04.12.212
Jurusan : Teknik Elektro S-I
Konsentrasi : Elektronika
Judul Skripsi : Desain dan Implementasi *Building Automation System* dengan Web dan WAP

Dipertahankan dihadapan Team Penguji Skripsi Jenjang Strata I
pada :

Hari : Selasa
Tanggal : 23-09-2008
Dengan Nilai : 85,05 *84*

Panitia Ujian Skripsi



Ir. Mochtar Asroni, MSME
Ketua

Ir.F. Yudi Limpraptono, MT
Sekretaris

Anggota Penguji

Komang Somawirata, ST, MT
Penguji Pertama

Irmalia Suryani F. ST, MT
Penguji Kedua

ABSTRAK

Building Automation System (BAS) merupakan suatu sistem dimana suatu gedung dapat diatur secara otomatis sesuai dengan keinginan pengguna. Pemanfaatan Sistem Otomatisasi Gedung kadangkala kurang efektif karena pengguna hanya dapat mengatur sistem ini dari jarak dekat (hanya pada komputer utama yang terhubung dengan alat pengatur) ataupun melakukan kontrol tanpa adanya fasilitas untuk memonitoring sistem otomatisasi gedung tersebut (dengan menyeting waktu lampu mati / menyala). Pada desain dan implementasi sistem otomatisasi gedung ini akan dirancang suatu sistem yang dapat mengatur dan memonitoring sistem otomatisasi gedung tanpa harus terikat waktu dan tempat si pengguna itu berada.

Sistem ini dibangun dengan menggunakan web dan wap sehingga sistem ini memungkinkan pengguna dapat mengatur dan melihat melalui browser (Internet Explorer, Mozilla Firefox, dll) pada PC ataupun wap browser pada handphone dengan cara mengetikkan alamat dari web/wapserver yang telah dibangun.

Hasil dari pengaturan ruangan tersebut akan disimpan di database mysql sehingga sistem ini dapat dimonitoring secara langsung dan sistem tidak akan rusak apabila terjadi gangguan dalam pengaturan ruangan

KATA PENGANTAR

Dengan memanjatkan puji syukur kepada Tuhan Yang Maha Lisa, atas berkat, rahmat dan karunia-Nya sehingga dapat menyelesaikan skripsi yang berjudul "Desain dan Implementasi *Building Automation System* dengan WEB dan WAP" ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan studi di jurusan Teknik Elektro S-1 konsentrasi Teknik Elektronika ITN Malang dan untuk mencapai gelar sarjana teknik.

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

1. Bapak Prof. Dr. Eng. Ir. Abraham Lomi, MSEE selaku Rektor ITN Malang.
2. Bapak Ir. Mochtar Asroni, MSME selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1 dan Dosen Pembimbing I.
4. Bapak Joseph Dedy Irawan, ST, MT selaku Dosen Pembimbing II.
5. Orangtua serta saudara-saudara kami yang telah memberikan doa restu, dorongan, semangat dan biaya.
6. Teman-teman instruktur Laboratorium Perancangan Elektronika yang telah membantu.
7. Kim Febriani Prasetyo yang telah membantu dan mendukung dalam penyelesaian.
8. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Penyusun telah berusaha semaksimal mungkin dan menyadari sepenuhnya akan keterbatasan pengetahuan dalam menyelesaikan laporan ini, untuk itu

penyusun mengharapkan saran dan kritik yang membangun dari pembaca demi kesempurnaan laporan ini.

Harapan penyusun semoga laporan ini memberikan manfaat bagi perkembangan ilmu pengetahuan bagi pembaca.

Malang, September 2008

Penulis

LEMBAR PERSEMBAHAN

Tuhan, Karuniakanlah diriku ketentraman batin untuk menerima hal-hal yang tak mungkin kuubah; Keberanian untuk mengubah hal-hal yang bisa kuubah; dan kebijaksanaan untuk mengetahui perbedaannya..

My Special thanks to:

My Family who always support me to finished my Last Assign

Mr. Yudi and Mr. Joseph who always give me suggest and support

Mr. Eko as Electrical Design Laboratory Leader who always give me a suggest, support and place in Laboratory

All of university level instructor who has give me knowledge in ITN Malang

All of my friends in ITN Malang who has give me support. Thanks Friends

All of the people in the world who give me a support.

Thanks All

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN	ii
BERITA ACARA UJIAN SKRIPSI.....	iii
ABSTRAK	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	ix
DAFTAR TABEL.....	xi
DAFTAR LAMPIRAN	xii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan	2
1.4 Batasan Masalah.....	3
1.5 Metodologi Perancangan.....	3
1.6 Sistematika Penulisan.....	4
BAB II DASAR TEORI.....	6
2.1 Pendahuluan.....	6
2.2 PHP	6
2.3 Mysql 5.....	8
2.4 Web Server.....	10
2.5 AT89S51	15
2.6 Relay	20
2.7 Wireless Application Protokol (WAP).....	21

BAB III PERANCANGAN DAN PEMBUATAN ALAT	37
3.1 Pendahuluan.....	26
3.2 Perancangan Perangkat Keras.....	28
3.2.1 Blok Diagram Alat.....	28
3.2.2 Mikrokontroler AT89S51.....	29
3.2.2.1 Rangkaian Clock.....	30
3.1.2.2 Rangkaian Reset.....	31
3.2.3 Rangkaian RS232 pada minimum sistem AT89S51.....	33
3.2.4 Rangkaian Driver Relay.....	34
3.3 Perancangan Perangkat Lunak.....	35
3.3.1 Diagram Alir Rangkaian.....	36
BAB IV PENGUJIAN DAN HASIL ANALISA	39
4.1 Pengujian Hardware.....	39
4.1.1 Pengujian Rangkaian Driver Relay.....	39
4.2 Perangkat Lunak (Software).....	41
4.2.1 Pengujian Program Pada Mikrokontroler.....	41
4.2.2 Pengujian Program Pada Web dan Wap.....	44
BAB V PENUTUP	48
5.1 Kesimpulan.....	48
5.2 Saran.....	48
Daftar Pustaka	50

DAFTAR GAMBAR

Gambar	Halaman
2.1 Hasil Eksekusi server side scripting.....	7
2.2 Blok Diagram Mikrokontroler AT89S51.....	16
2.3 Skema Pinout AT89S51.....	17
2.4 Osilator Eksternal AT89S51.....	20
2.5 Model Pemrograman WAP.....	22
2.6 Penggunaan Dari WAP Gateway.....	23
2.7 Sistem Jaringan WAP.....	23
2.8 Contoh Tampilan WAP IDE: Open Wave SDK.....	25
3.1 Diagram Blok Sistem.....	27
3.2 Blok Diagram Perangkat Keras Desain dan Implementasi Building Automation Sistem dengan WEB dan WAP.....	28
3.3 Perancangan Sistem Mikrokontroler AT89S51.....	30
3.4 Perancangan Rangkaian Clock.....	31
3.5 Perancangan Rangkaian Reset.....	32
3.6 Rangkaian RS232 Menggunakan IC MAX232.....	34
3.7 Rangkaian Driver pada ULN 2003.....	34
3.8 Rangkaian Driver Relay.....	35
3.9 Flowchart Sistem Secara Keseluruhan.....	36
3.10 Flowchart Proses Mematikan / Menyalakan Lampu.....	37
4.1 Rangkaian Driver Relay.....	40
4.2 Pengujian Rangkaian Driver Relay Saat On.....	40
4.3 Pengujian Rangkaian Driver Relay Saat Off.....	41
4.4 Diagram Blok Pengujian Software Mikrokontroler pada Sistem.....	41
4.5 Tampilan Pada Saat Ruangan 1 Menyala, Ruangan 2 Mati, Ruangan 3 Mati, dan Ruangan 4 Menyala.....	42
4.6 Alat Desain dan Implementasi Building Automation System Dengan Web dan WAP.....	44
4.7 Diagram Blok Pengujian Software pada Sistem Secara Keseluruhan.....	45

4.8	Tampilan Web Pada Saat Ruangan 1 Menyala, Ruangan 2 Mati, Ruangan 3 Mati, dan Ruangan 4 Menyala	45
4.9	Tampilan WAP Pada Saat Ruangan 1 Menyala, Ruangan 2 Mati, Ruangan 3 Mati, dan Ruangan 4 Menyala	46

DAFTAR TABEL

Tabel	Halaman
2.1 Fungsi Port 1	18
2.2 Fungsi Khusus pada Port 3.....	19
4.1 Hasil Pengujian Apache Server.....	47
4.2 Hasil Pengujian Web dan WAP.....	47

DAFTAR LAMPIRAN

1. Gambar Skematik Rangkaian Keseluruhan
2. Listing Program
3. Datasheet

BAB I

PENDAHULUAN

1.1 Latar Belakang

Sehubungan dengan perkembangan pola pikir manusia ilmu pengetahuan dan teknologi ternyata mengalami kemajuan terus menerus. Perbaikan terhadap teknologi yang sudah ada terus dilakukan agar menjadi lebih mudah. Salah satu bidang teknologi yang mengalami perkembangan lebih pesat adalah teknologi elektronika yang tidak terlepas dari tuntutan masyarakat yang terus – menerus berkembang sesuai dengan kondisi dan situasi yang dihadapi.

Seiring dengan perkembangan teknologi yang semakin berkembang, maka pemanfaatan teknologi dapat diterapkan dalam berbagai bidang, salah satunya pada bidang teknologi informasi. Sering kita melihat pada gedung-gedung terdapat kecenderungan lupa mematikan alat-alat listrik sehingga terjadi penggunaan daya listrik yang tidak diperlukan dan terkadang alat listrik yang tidak terpantau tersebut dapat menimbulkan arus pendek yang mengakibatkan kebakaran, maka diperlukan pengawasan terhadap listrik yang digunakan dalam suatu ruangan. Untuk itu penulis mencoba untuk memberikan solusi dalam proses kontrol listrik ruangan dengan memanfaatkan protokol http dan wap sebagai aplikasi web untuk memudahkan memantau keadaan gedung dan melakukan pengaturan ruangan dimana dan kapan saja (daerah yang dijangkau jaringan telekomunikasi).

Penggunaan sistem kontrol ini memiliki keuntungan yaitu kecepatan pengiriman data relatif tinggi. Komunikasi antara *client* dan *server* dapat dilakukan dengan jarak yang jauh. Sehingga dapat menjangkau jarak dari ruangan tersebut.

1.2 Rumusan Masalah

Dalam perencanaan dan pembuatan Desain dan Implementasi Building Automation System Berbasis WEB dan WAP maka dapat dirumuskan beberapa masalah yang akan dibahas yaitu:

1. Bagaimana merancang dan membuat sistem komunikasi dari komputer *server* ke handphone dan komputer *client* menggunakan Web dan WAP?
2. Bagaimana merancang dan membuat *Building Automation System* dengan menggunakan Mikrokontroler MCS51?
3. Bagaimana merancang dan membuat perangkat keras dan perangkat lunak atau *software* pada mikrokontroler dan komputer server?

1.3 Tujuan

Tujuan yang hendak dicapai dalam pembuatan Desain dan Implementasi Building Automation System Berbasis Web dan WAP adalah Tujuan dari perancangan sistem ini ialah untuk melakukan otomatisasi pada suatu gedung dengan memanfaatkan koneksi internet dan aplikasi *wireless* sehingga memudahkan pengguna untuk memantau dan mengatur penggunaan listrik suatu gedung sehingga dapat juga

menghindari terjadinya pemakaian daya listrik yang berlebihan serta adanya arus pendek akibat lupa mematikan alat listrik

1.4 Batasan Masalah

Agar pembahasan dari desain dan implementasi *Building Automation System* ini tidak terlalu meluas maka penyusun perlu membuat batasan-batasan masalah yang meliputi :

1. Tidak membahas pembagian daya tiap ruangan
2. Mikrokontroler yang digunakan adalah AT89S51
3. *Operating System* yang digunakan adalah Windows XP
4. Tidak membahas pengaman jaringan
5. Tampilan *web* hasil monitoring dapat diakses menggunakan *browser* yaitu *internet explorer, mozilla firefox, dan opera.*
6. Tidak membahas dasar teori WAP dan WEB

1.5 Metodologi Perancangan

Metodologi yang digunakan dalam perencanaan dan pembuatan Desain Implementasi *Buiding Automation System* berbasis WEB dan WAP adalah sebagai berikut :

- a. Studi literatur untuk memahami webserver dan *HTTP (Hypertext Transfer Protocol)* sebagai sarana untuk dapat menyajikan informasi/tampilan melalui *web*. Selain itu juga ditambah dengan memahami sistem dan rangkaian elektronika yang mendukung untuk merealisasikan alat secara keseluruhan.

- b. Percobaan dan eksperimen dilakukan untuk memperoleh data-data yang diperlukan dalam membuat rancangan sistem elektronika.
- c. Pada tahap realisasi alat yang dibuat, dilakukan perancangan alat yang meliputi merancang rangkaian untuk tiap-tiap blok dan rancangan rangkaian keseluruhan sistem, pembuatan PCB, dan perakitan hasil rancangan.
- d. Untuk mengetahui cara kerja alat, maka dilakukan pengujian tiap blok dan pengujian secara keseluruhan. Pengujian tiap blok meliputi pengujian :
 - i. Driver Relay
 - ii. Apache Server
 - iii. Program pada saat *Building Automation System* dijalankan
 - iv. Pengujian keseluruhan sistem dilakukan dengan menggabungkan seluruh rangkaian elektronik, mekanik alat dan program.
- e. Menganalisis hasil pengujian untuk membuat kesimpulan.

1.6 Sistematika Penulisan

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

BAB I. PENDAHULUAN

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

BAB II. DASAR TEORI

Pada bab ini dibahas tentang teori-teori yang mendukung dalam perencanaan dan pembuatan alat ini yang meliputi *relay*, *HTTP (Hypertext Transfer Protocol)*, *WAP Wireless Application System* dan mikrokontroler MCS51.

BAB III. PERENCANAAN DAN PEMBUATAN ALAT

Pada bab ini dibahas tentang perencanaan dan pembuatan keseluruhan sistem perangkat keras yaitu blok diagram rangkaian, prinsip kerja, perancangan setiap komponen yang digunakan, dan perancangan perangkat lunak yaitu *flowchart* kerja dari program yang dibuat.

BAB IV. PENGUJIAN ALAT DAN ANALISIS HASIL PENGUJIAN

Pada bab ini dibahas tentang proses serta hasil dari pengujian alat, yang didasarkan oleh pengukuran-pengukuran. Pengujian dilakukan pada tiap blok rangkaian yang digunakan meliputi, rangkaian driver *relay*, rangkaian minimum sistem mikrokontroler AT89S51, serta analisis data dari pengujian dari semua pengujian dan program yang telah dibuat. Program yang dibuat meliputi *WAP, Web Application*, dan *Visual Basic*

BAB V. PENUTUP

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

BAB II

DASAR TEORI

2.1. Pendahuluan

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

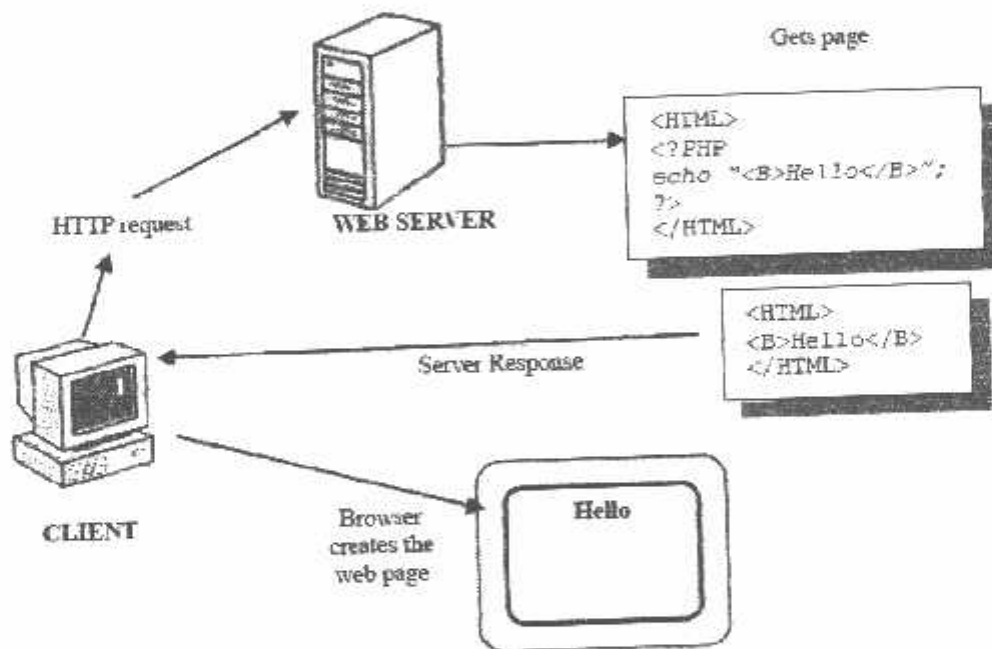
1. PHP
2. Mysql 5
3. Web Server
4. Relay
5. Mikrokontroler AT89S51
6. WAP

2.2. PHP

PHP (PHP: Hypertext Preprocessor) adalah sebuah *HTML-embedded scripting language*, yang berada dalam dokumen HTML, seperti halnya JavaScript atau VBScript. Tujuannya kurang lebih sama, yaitu untuk menciptakan halaman web yang interaktif dan dinamis. Lalu apa bedanya PHP dengan JavaScript atau VBScript? Perbedaannya adalah tempat dimana script tersebut dieksekusi. JavaScript dan VBScript merupakan *client-side scripting language* yang akan dieksekusi di sisi klien (*browser*), sedangkan PHP adalah *server-side*

scripting language yang akan dieksekusi di dalam *web server* ketika *script*-nya dipanggil.

Pada *client-side scripting*, ketika klien meminta sebuah dokumen yang mengandung *script*, VBScript atau JavaScript, *script* tersebut akan di-*download* dan dieksekusi di dalam *browser* yang bersangkutan. Sedangkan pada *server-side scripting*, dokumen yang diminta tetap berada di server, dijalankan di server dan hasilnya yang berupa html dikirimkan ke browser untuk ditampilkan. Sehingga dari segi keamanan, jelas lebih aman dengan *server-side scripting*, sebab klien tidak akan dapat melihat *script* atau *source code* asli yang ada di server, sebab yang ditampilkan di browser adalah hasil eksekusi *script* tersebut.



Gambar 2.1. Hasil Eksekusi *server side scripting* ⁽¹⁾

Sintaks-sintaks yang digunakan dalam PHP ini kebanyakan diambil dari bahasa-bahasa pemrograman yang populer seperti C, Perl, dan Java dengan penambahan beberapa kelebihan dan keunikan yang dimiliki oleh PHP.

PHP memiliki beberapa kemampuan yang sangat mendukung dalam pembuatan halaman web yang interaktif dan menarik. Kemampuan tersebut antara lain : perhitungan matematis, informasi jaringan, mail, *regular expression*, dan yang paling menonjol adalah kemampuan PHP dalam menyediakan antarmuka dengan beberapa server database yang populer di pasaran, seperti MySQL, Oracle, Sybase, PostgreSQL, mSQL, dan lainnya.

2.3. MYSQL 5

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU General Public License (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL.

Tidak sama dengan proyek-proyek seperti Apache, dimana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius.

Terdapat beberapa API tersedia yang memungkinkan aplikasi-aplikasi komputer yang ditulis dalam berbagai bahasa pemrograman untuk dapat mengakses basis data MySQL antara lain: bahasa pemrograman C, C++, C#,

bahasa pemrograman Eiffel, bahasa pemrograman Smalltalk, bahasa pemrograman Java, bahasa pemrograman Lisp, Perl, PHP, bahasa pemrograman Python, Ruby, REALbasic dan Tcl. Sebuah antarmuka ODBC memanggil MyODBC yang memungkinkan setiap bahasa pemrograman yang mendukung ODBC untuk berkomunikasi dengan basis data MySQL. Kebanyakan kode sumber MySQL dalam ANSI C.

MySQL mendukung banyak storage engine dengan berbagai tipe table, dimana storage engine itu sendiri terbagi atas 2 tipe, yaitu engine yang menagani transaction-safe table dan yang tidak. Pada umumnya, MyISAM merupakan engine yang non-transactional table, MyISAM menyediakan proses penyimpanan dan pengambilan data dengan cepat, serta dilengkapi dengan kemampuan fulltext searching. Hal ini menyebabkan MyISAM menjadi default engine pada MySQL. Beberapa storage engine yang lain:

1. MEMORY storage engine (HEAP engine), menyediakan in-memory tables. Engine ini dibantu oleh MERGE storage engine yang fungsinya meyatukan beberapa table menjadi satu.
2. InnoDB dan BDB storage engine menyediakan transaction-safe tables
3. EXAMPLE storage engine hanya merupakan stub engine yang tidak melakukan apapun. Walaupun dapat membuat table, tetapi tidak ada data yang bisa disimpan atau diambil. Kegunaan sebenarnya hanya sebagai ilustrasi bagaimana membuat storage engine baru.
4. NDB Cluster storage engine digunakan untuk menggabungkan table yang dibagi-bagi ke banyak computer walaupun terdapat perbedaan sistem operasi. Engine ini hanya dapat di-support oleh Linux, Solaris dan Mac.

5. ARCHIVE storage engine berguna untuk menyimpan data berukuran besar tanpa menggunakan index.
6. CSV storage engine menyimpan data dalam bentuk text file, dengan menggunakan tanda koma (,) sebagai separatornya.
7. BLACKHOLE storage engine dapat menerima data tetapi tidak menyimpannya, bila dilakukan pengambilan data maka akan me-return set kosong.
8. FEDERATE storage engine menyimpan data dalam remote database.

2.4 Web Server

Web Server adalah *software* server yang menjadi tulang belakang dari *World Wide Web* (WWW). *Web server* menunggu permintaan dari *client* yang menggunakan *browser* seperti netscape navigator, Internet Explorer, mozilla, dan *program browser* lainnya. Jika ada permintaan dari *browser*, maka *web server* akan memproses permintaan itu dan kemudian memberikan hasil prosesnya berupa data yang diinginkan kembali ke *browser*. Data ini mempunyai format yang standar disebut dengan format SGML (*Standard General Markup Language*). Data yang berupa format ini kemudian akan ditampilkan oleh browser sesuai dengan kemampuan browser itu. Contohnya ialah bila data yang dikirim berupa data gambar, browser yang hanya mampu menampilkan text (misalnya lynx) tidak akan mampu menampilkannya dan jika ada akan menampilkan alternatifnya saja.

Web Server, untuk berkomunikasi dengan clientnya (*web browser*) mempunyai protokol sendiri yaitu HTTP (*HyperText Transfer Protocol*). Dengan

protokol ini, komunikasi antar *web server* dengan clientnya (*browser*) dapat saling dimengerti dan lebih mudah. Seperti telah dijelaskan diatas, Standar format data pada *World Wide Web* adalah SGML. Tapi sudah menjadi hal yang umum bahwa para pengguna internet lebih banyak menggunakan format HTML (*HyperText Markup Language*) karena penggunaannya yang lebih sederhana dan mudah dipelajari.

Kata *HyperText* mempunyai arti bahwa seorang pengguna internet dengan web browse-nya dapat membuka dan membaca dokumen-dokumen yang ada dalam komputernya atau bahkan komputer yang jauh tempatnya sekalipun. Hal ini memberikan cita rasa dari suatu proses yang tridimensional, artinya pengguna internet dapat membaca dari satu dokumen ke dokumen yang lain hanya dengan mengklik beberapa bagian dari halaman-halaman dokumen (web) itu. Proses yang dimulai dari permintaan webclient (*browser*), diterima web server, diproses, dan dikembalikan hasil prosesnya oleh web server ke *webclient* lagi dilakukan secara transparan. Setiap orang dapat dengan mudah mengetahui apa yang terjadi pada tiap-tiap proses. Secara garis besarnya web server hanya memproses semua masukan yang diperolehnya dari web clientnya.

Dalam membangun Server menggunakan Windows untuk membuat sebuah web server, kita akan menemukan berbagai macam persoalan, dimulai dari pemilihan software web browser yang mana yang paling sesuai kebutuhan, apa spesifikasi hardware yang dibutuhkan, bagaimana kondisi interkoneksi jaringan internet yang ada, dan lain sebagainya. Belum lagi termasuk bagian pembuatan halaman-halaman webnya, mau menggunakan format apa (HTML, SGML, PHP, PHP3, CGI, dan lain-lain). Hal yang paling utama dalam proses pembuatan Web

Server adalah memilih software mana yang akan digunakan sebagai web server kita. Untuk itu, perlu adanya beberapa pertimbangan sebagai berikut :

1. Komersial lawan Freeware (software gratis).
2. Kemudahan instalasi
3. Kemudahan mengonfigurasi
4. Kemudahan untuk menambah atau mengubah periferalnya.
5. Kemampuan software.
6. Besar ruang yang dibutuhkan untuk menyimpan file-file minimal yang dibutuhkan agar software berfungsi dengan baik
7. Prospek software tersebut dimasa yang akan datang
8. Performasi dan konsumsi sumber daya yang digunakan software itu.
9. Fasilitas apa yang mampu didukung software itu.
10. Dukungan Teknis (mempunyai site-site atau mailis untuk bertanya bila terjadi masalah).
11. Dukungan Platform (jenis sistem operasi apa saja yang dapat menjalankan software tersebut)
12. Dukungan terhadap third party (Apakah software ini dapat ditambahkan software tambahan sebagai pelengkap)

Banyak sekali software web server yang berada di internet. Dengan berdasarkan pada 12 macam pertimbangan diatas, maka dapat dipilih software mana saja yang cocok dengan kebutuhan kita. Misalnya : Kita memasang web server untuk keperluan suatu perusahaan jasa internet (ISP), maka pertimbangan yang harus diambil adalah apakah mereka menginginkan software yang gratis atau komersial. Keuntungan dari software komersial adalah mereka punya dukungan

teknis dan dokumentasi yang lengkap, sedang pada kebanyakan software gratis tidak punya. Namun ada juga software gratisan yang mempunyai dukungan teknis dari pembuatnya dan dengan dokumentasi yang lengkap. Salah satu software web server gratisan seperti itu adalah Web server Apache. Web server Apache mempunyai kelebihan dilihat dari beberapa pertimbangan diatas:

1. Apache termasuk dalam kategori freeware (software gratisan)
2. Apache mudah sekali proses instalasinya jika dibanding web server lainnya seperti NCSA, IIS, dan lain-lain.
3. Mampu beroperasi pada berbagai platform sistem operasi seperti : AIX 3.1, BSDI 2.0, FreeBSD 2.1, HP-UX 9.07, IRIX 5.3, Linux, NetBSD 1.1, NEXTSTEP, SolarisX86 2.5, Solaris 2.4, Solaris 2.5, SunOS 4.1.3, UnixWare
4. Apache mudah untuk mengkonfigurasinya karena hanya mempunyai satu file konfigurasi.
5. Apache Web server mudah dalam menambahkan perferal lainnya ke dalam platform web servernya, misalnya : untuk menambahkan modul, cukup hanya menset file konfigurasinya agar mengikutsertakan modul itu kedalam kumpulan modul lain yang sudah dioperasikan.

Features atau ciri khas dari web server Apache adalah :

1. Dapat dijadikan pengganti bagi NCSA web server.
2. Perbaikan terhadap kerusakan dan error pada NCSA 1.3 dan 1.4
3. Apache web server dalam merespon client sangat cepat jauh melebihi server NCSA.
4. Mampu di kompilasi sesuai dengan spesifikasi HTTP yang sekarang.
5. Apache menyediakan feature untuk multihomed dan virtual server.

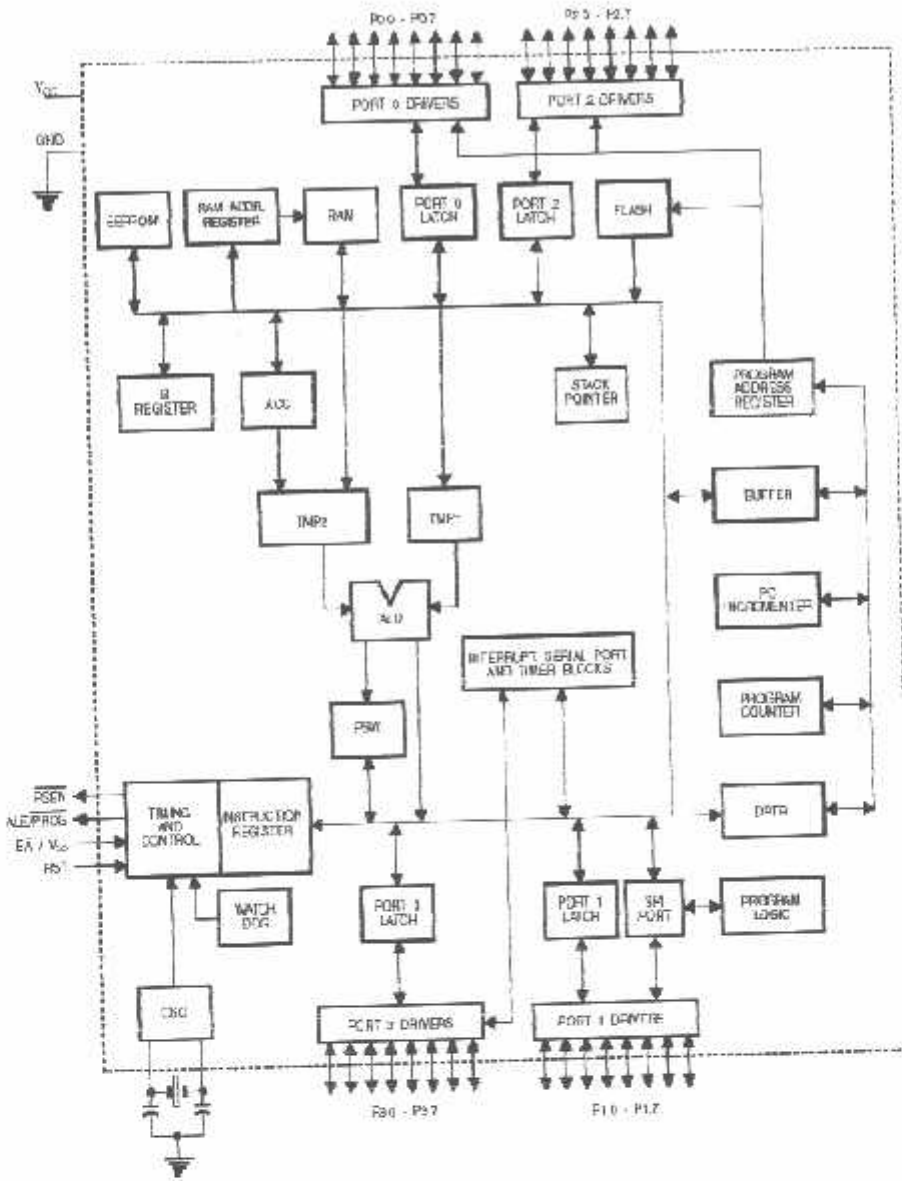
6. Kita dapat men-set respon error yang akan dikirim web server dengan menggunakan file atau skrip.
7. Server Apache dapat otomatis berkomunikasi dengan client browsernya untuk menampilkan yang tampilan terbaik pada client browsernya. Misalnya browser ingin menampilkan dalam bahasa Spanyol, maka Apache web server otomatis mencari dalam servicenya halaman-halaman dengan bahasa Spanyol.
8. Webserver Apache secara otomatis menjalankan file `index.html`, halaman utamanya, untuk ditampilkan secara otomatis pada clientnya.
9. Webserver Apache mempunyai level-level pengamanan.
10. Apache mempunyai komponen dasar terbanyak di antara webserver lain, yang berarti bahwa webserver Apache termasuk salah satu dari webserver yang lengkap.
11. Ditinjau dari segi sejarah perkembangan dan prospeknya di masa yang akan datang, Apache web server mempunyai prospek yang cerah. Apache berasal dari webserver NCSA yang kemudian dikembangkan karena NCSA masih mempunyai kekurangan dibidang kompatibilitasnya dengan sistem operasi lain. Sampai saat ini, webserver Apache terus dikembangkan oleh team dari `apache.org`.
12. Performansi dan konsumsi sumberdaya (*resource*) dari webserver apache tidak terlalu banyak, hanya sekitar 20 MB untuk file-file dasarnya dan setiap daemونها hanya memerlukan sekitar 950 KB memory *per-child*.
13. Mendukung transaksi yang aman (*secure transaction*) menggunakan SSL (*Secure Socket Layer*).
14. Mempunyai dukungan teknis melalui web.

15. Mempunyai kompatibilitas platform yang tinggi.
16. Mendukung *third party* berupa modul-modul tambahan.

2.5 AT89S51

Mikrokontroler AT89S51 memiliki berbagai fasilitas penting yang dimiliki oleh AT89S8253 sebagai sebuah pengendali, antara lain adalah sebagai berikut :

- ❖ Memiliki 4 Kilobytes Flash PEROM (*Programmable and Erasable Read Only Memory*) yang dapat diprogram ulang dengan fasilitas SPI (*Serial Programming Interface*). Memori Flash PEROM ini dapat bertahan untuk dihapus dan ditulis ulang kira-kira sebanyak 1000 kali.
- ❖ Beroperasi pada kisaran tegangan 4V sampai dengan 6V.
- ❖ Mendukung pewaktuan 0 Hz sampai 24 MHz *clock speed*.
- ❖ Fasilitas pengamanan program memori sebanyak tiga level pengamanan data (*program memory lock*).
- ❖ Memiliki RAM internal sebanyak 256 byte
- ❖ Memiliki 32 unit input dan output (*port pins*) yang dapat diprogram sebagai jalur masukan dan keluaran. Terbagi dalam 4 port paralel dan 1 port serial yang dapat berjalan dengan mode dua arah (*full duplex*).
- ❖ Memiliki fasilitas *Timer* dan *Counter* sebanyak dua buah.
- ❖ Memiliki fasilitas *watchdog timer* yang dapat diprogram sesuai keinginan.
- ❖ Memiliki satu buah DPTR (*Data Pointer*). Data pointer berguna untuk pengalamatan memori dengan alamat 16 bit.

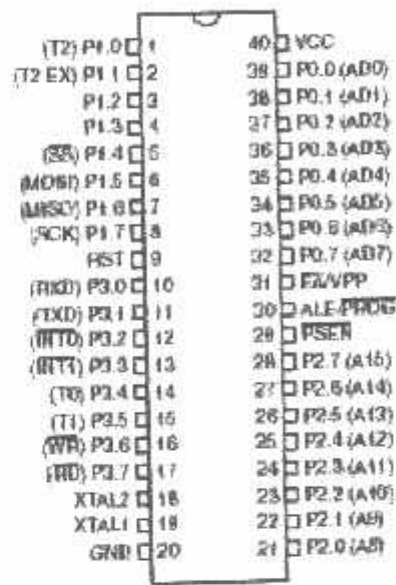


Gambar 2.2. Blok Diagram Mikrokontroler AT89S51^[2]

Seperti halnya mikrokontroler yang lainnya, mikrokontroler AT89S51 memiliki dimensi yang cukup kecil. Dimensinya yang cukup kecil membuat mikrokontroler sangat berguna dalam perancangan *embedded control application*. Berikut ini adalah penjelasan dari dimensi jalur-jalur penting dalam dimensi mikrokontroler, yang juga dikenal dengan nama *pin*.

Mikrokontroler AT89S51 terdiri dari 40 pin dengan pin *out/in* sebagai

berikut:



Gambar 2.3. Skema Pinout AT89S51^[2]

Fungsi-fungsi tiap pinnya adalah sebagai berikut :

- ❖ VCC (Supply tegangan), pin 40
- ❖ GND (Ground) , pin 20
- ❖ Port 0, pin 32-39

Merupakan port input-output dua arah, tanpa internal pull-up dan konfigurasi sebagai multipleks bus alamat rendah (A0-A7) dan data selain pengaksesan program memory dan data memory eksternal.

- ❖ Port 1, pin 1-8

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

Fungsi khusus pin-pin pada port1 sebagai berikut :

Tabel 2.1. Fungsi Port 1¹²⁾

Nama Pin	Kegunaan Khusus
P1.0	T2 (<i>input timer</i> atau <i>counter 2</i> dari sumber <i>clock</i>
P1.1	eksternal)
P1.4	T2EX (<i>Timer/Counter 2 capture/reload trigger</i> dan <i>direction control</i>)
P1.5	SS (untuk memilih <i>slave port</i>)
P1.6	MOSI (<i>Master data output</i> , masukan <i>slave data input pin</i> untuk fasilitas SPI)
P1.7	MISO (<i>Master data input</i> , keluaran <i>slave data output pin</i> untuk fasilitas SPI)
	SCK (<i>Master clock output</i> , masukan untuk pin <i>slave clock</i> untuk fasilitas SPI channel)

❖ Port 2, pin 21-28

Merupakan port input-output dengan *internal pull-up*. Mengeluarkan alamat tinggi selama pengambilan program memori eksternal.

❖ Port 3, pin 10-17

Merupakan port input-output dengan *internal pull-up*, dimana port 3 juga memiliki fungsi khusus dan dapat dilihat pada table berikut :

Tabel 2.2. Fungsi Khusus Pada Port 3 ^[2]

Nama Pin	Kegunaan Khusus
P3.0	RXD (port penerima komunikasi serial)
P3.1	TXD (port pengirim komunikasi serial)
P3.2	$\overline{INT0}$ (masukan untuk fasilitas interupsi eksternal 0)
P3.3	$\overline{INT1}$ (masukan untuk fasilitas interupsi eksternal 1)
P3.4	T0 (masukan untuk fasilitas timer eksternal 0)
P3.5	T1 (masukan untuk fasilitas timer eksternal 1)
P3.6	\overline{WR} (sinyal tulis memori data eksternal)
P3.7	\overline{RD} (sinyal baca memori data eksternal)

❖ RST (Reset), pin 9

Input reset merupakan reset master untuk AT89S51

❖ ALE /Prog (*Address Latch Enable*), pin 30

Digunakan untuk menahan alamat memori eksternal selama pelaksanaan intruksi.

❖ PSEN (*Program Store Enable*), pin 29

Merupakan sinyal pengontrol yang memperbolehkan program memori eksternal masuk ke dalam bus.

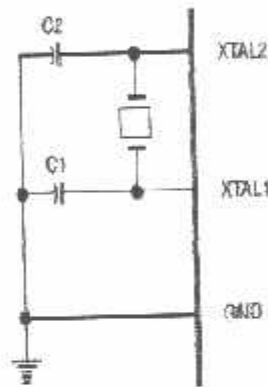
❖ EA/VPP (*External Access*), pin 31

Dapat diberikan logika rendah (Ground) atau logika tinggi (+5 Volt). Jika diberikan logika tinggi maka mikrokontroller akan mengakses program dari ROM internal (EEPROM/Flash Memori), dan jika diberikan logika

rendah maka mikrokontroller akan mengakses program dari memori eksternal.

❖ X-TAL 1 dan X-TAL 2, pin 18,19

Pin ini dihubungkan dengan kristal bila menggunakan osilator internal. X-TAL 1 merupakan masukan ke rangkaian osilator internal sedangkan X-TAL 2 keluaran dari rangkaian osilator internal. Untuk keperluan ini diperlukan kapasitor penstabil sebesar 30 pF. Dan nilai dari X-TAL tersebut antara 4-24Mhz. untuk lebih jelasnya dapat dilihat gambar pemasangan X-TAL serta kapasitor yang digunakannya.



Gambar 2.4. Osilator Eksternal AT89S51^[2]

2.6. Relay

Relay merupakan rangkaian yang bersifat elektronis sederhana dan tersusun oleh

- saklar
- medan elektromagnet (kawat koil)
- poros besi

Cara kerja komponen ini dimulai pada saat mengalirnya arus listrik melalui koil, lalu membuat medan magnet sekitarnya merubah posisi saklar sehingga menghasilkan arus listrik yang lebih besar. Disinilah keutamaan komponen sederhana ini yaitu dengan bentuknya yang minimal bisa menghasilkan arus yang lebih besar

Komponen sederhana ini dalam perkembangannya digunakan (atau pernah digunakan) sebagai komponen dasar berbagai perangkat elektronika, lampu kendaraan bermotor, jaringan elektronik, televisi, radio, bahkan pada tahun 1930an pernah digunakan sebagai perangkat dasar komputer yang keberadaannya kini digantikan oleh mikroprosesor seperti Intel Corp. dan AMD. Semua itu karena pemakaian relay mempunyai Keuntungan yaitu :

- Dapat mengontrol sendiri arus serta tegangan listrik yang diinginkan
- Dapat memaksimalkan besarnya tegangan listrik hingga mencapai batas maksimalnya
- Dapat menggunakan baik saklar maupun koil lebih dari satu, disesuaikan dengan kebutuhan

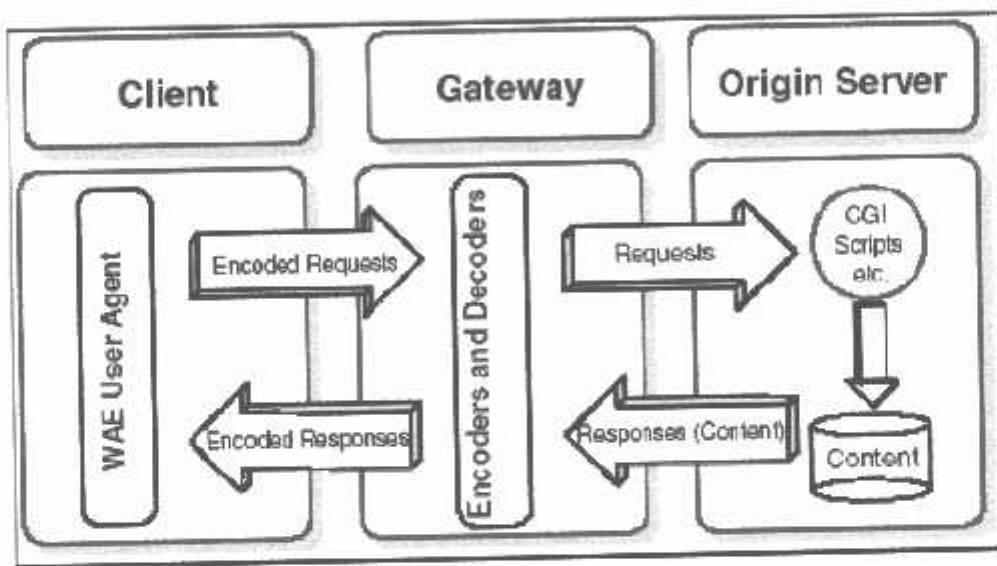
2.7 WIRELESS APPLICATION PROTOCOL (WAP)

Merupakan standard di seluruh dunia dalam menyediakan komunikasi internet dan mengedepankan layanan pada digital mobile devices, seperti telepon seluler, pagers, dan alat wireless lainnya. Protokol ini adalah suatu spesifikasi global yang mengijinkan bagi user yang memiliki alat digital maka dengan leluasa

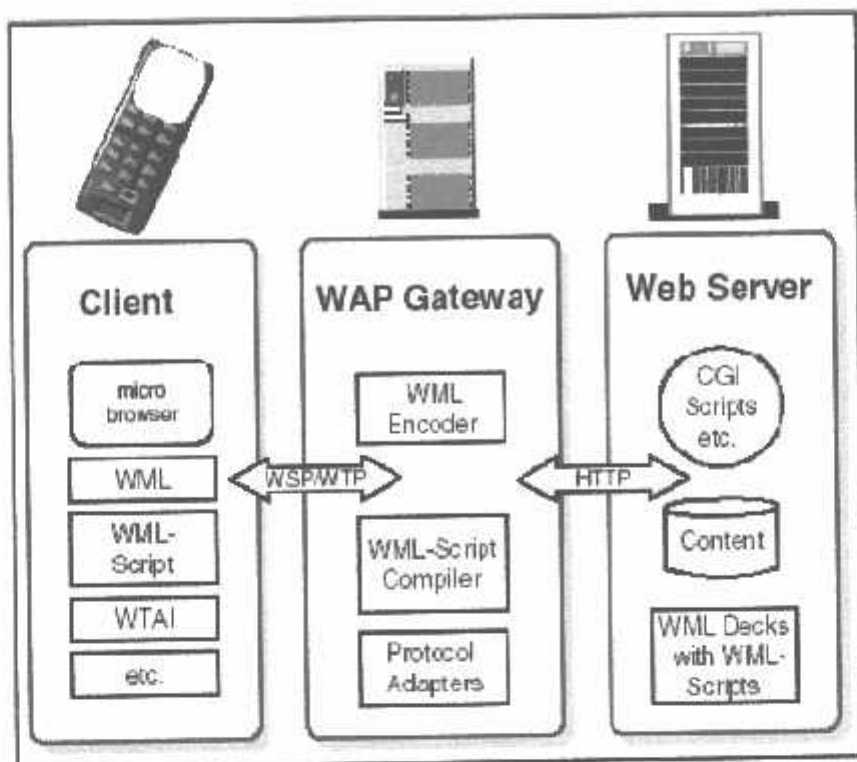
user tersebut untuk mengakses dan saling berhubungan dengan layanan aplikasi Internet, intranet, dan extranet.

WAP digambarkan suatu arsitektur standard dan protokol untuk menerapkan akses internet secara wireless. Elemen-elemen penting dari spesifikasi WAP meliputi :

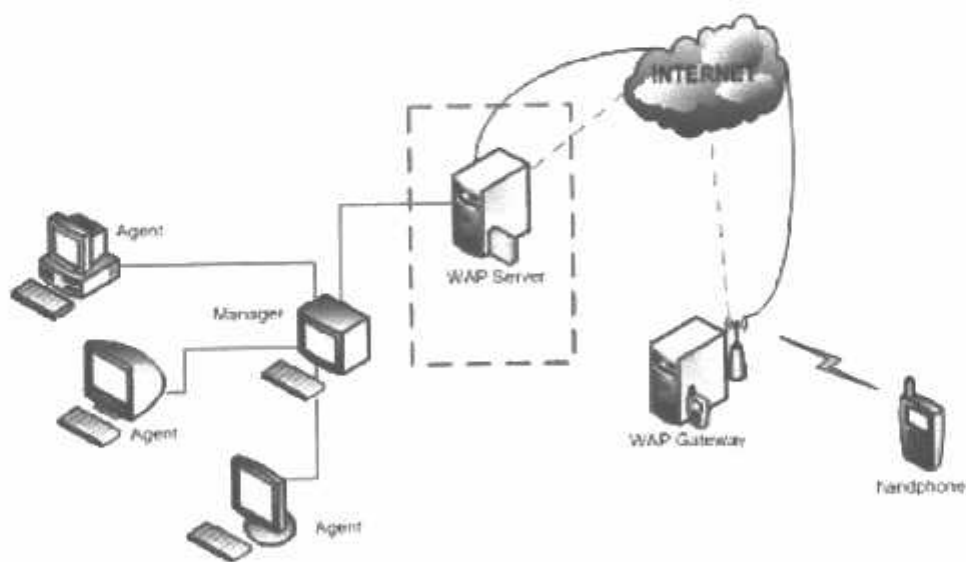
- WAP client (alat handheld atau WAP terminal)
- WAP gateway
- Web server



Gambar 2.5 Model Pemrograman WAP^[3]



Gambar 2.6 Penggunaan dari WAP gateway^[3]



Gambar 2.7 System Jaringan WAP^[3]

Keterangan :

1. WAP Gateway

WAP Gateway disini dalam dunia GSM, adalah provider layanan jasa telekomunikasi seperti Telkomsel, Indosat atau Satelindo. Teknologi yang digunakan untuk berkomunikasi antara handphone dengan WAP Gateway adalah GPRS (General Packet Radio Service)

GPRS merupakan implementasi teknologi packet-switching pada lingkungan GSM, sebagai pengembangan lebih lanjut dari teknologi GSM generasi kedua (2G), yang dasarnya adalah sebuah teknologi circuit-switching. GPRS sering juga disebut sebagai teknologi GSM Fase 2+ sebelum menuju teknologi 3G yang merupakan teknologi packet-switching. Dalam teknologi packet-switching, koneksi ke jaringan hanya dilakukan pada saat ada data yang dikirim sekaligus dalam satu paket sehingga lebih efisien dibanding koneksi permanen pada teknologi circuit-switching, serta memungkinkan kecepatan transmisi data sampai dengan 115Kbps, dibandingkan dengan 9,56 Kbps pada sistem GSM 900.

Implementasi GPRS memungkinkan penerapan Internet Protocol (IP) pada jaringan GSM disamping interkoneksi dengan jaringan data lain melalui protokol standard seperti TCP/IP atau X.25.

Untuk setting masing-masing handphone bisa menghubungi provider terkait, misal kartu Simpati bisa dengan membuka halaman web telkomsel di www.telkomsel.com

2. WAP Server

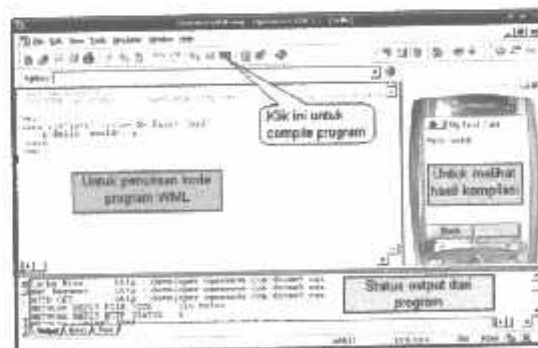
Komponen di WAP Server :

- a. Web Server, dalam hal ini bisa menggunakan Apache.
- b. Database, sebagai penyimpan informasi di WEB, bisa menggunakan MySQL.
- c. PHP, sebagai bahasa script disisi server untuk pengembangan WAP.
- d. WML, sebagai interface ke HP sehingga kode-kode PHP bisa terbaca di ponsel yang terbatas kemampuannya.

Pengembangan aplikasi WAP pada sisi client dilakukan dengan pemrograman WML dan WMLScript. Program aplikasi yang dapat digunakan ada banyak, bisa menggunakan notepad atau dengan IDE (Integrated Development Environment) yang banyak memberikan fasilitas.

Ukuran halaman WML dibatasi tidak melebihi 1024 byte dalam bentuk binary WMLC (WML Compiled, WML yang telah diencode). Sehingga terkadang kode-kode WML yang berukuran besar dapat berjalan baik di emulator tapi mungkin tidak untuk sebagian besar perangkat WAP.

Salah satu WAP IDE (sudah termasuk WAP Emulator) yang dapat digunakan adalah Open Wave SDK buatan phone.com, yang dapat didownload di <http://www.openwave.com>



Gambar 2.8 Contoh tampilan WAP IDE : Open Wave SDK^[3]

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

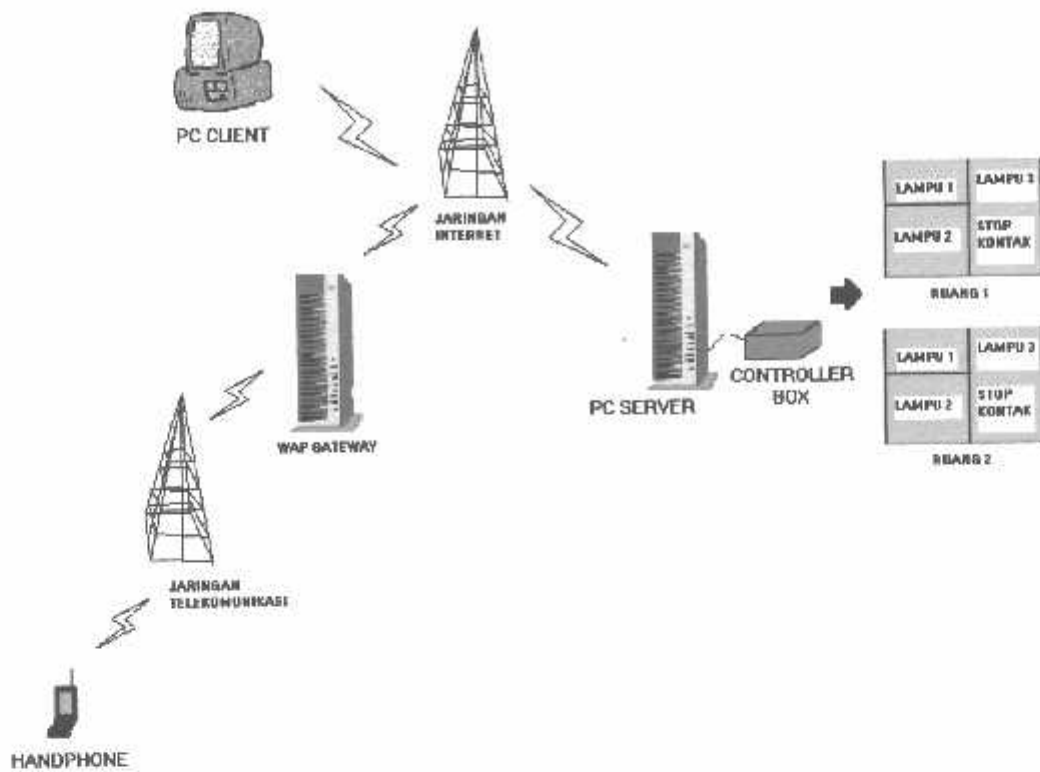
3.1. Pendahuluan

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

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

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

Perangkat keras sendiri terdiri dari rangkaian Driver Relay, minimum sistem AT89S51 dan komputer yang digunakan untuk menampilkan data ke monitor. Pada mikrokontroler menggunakan bahasa pemrograman *assembly* untuk mikrokontroler keluarga MCS51.



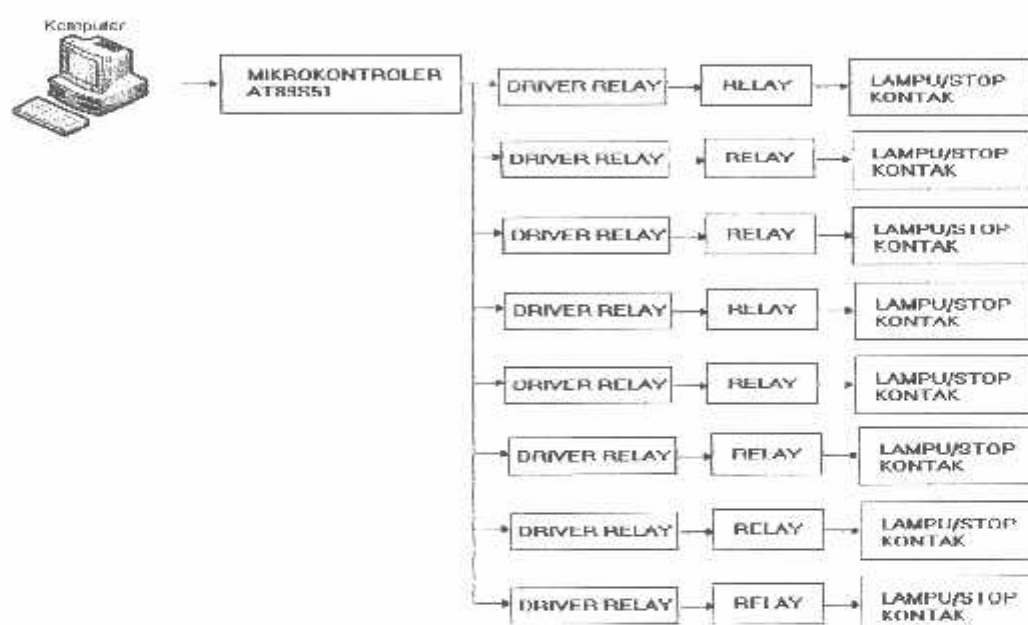
Gambar 3.1. Diagram Blok Sistem

Secara umum gambar sistem kerja keseluruhan adalah Website ataupun WAP berfungsi untuk mengubah nilai pada database mysql dimana database tersebut akan dibaca oleh Visual Basic. Setelah database tersebut dibaca, visual basic akan mengeksekusi dan mengirimkan data secara serial ke mikrokontroler AT89S51. Mikrokontroler AT89S51 berfungsi untuk menyalakan / mematikan relay berdasarkan perintah yang dikirimkan oleh software Visual Basic secara serial. Relay tersebut terhubung dengan lampu/stop kontak gedung sehingga lampu/stop kontak pada gedung tersebut dapat menyala ataupun mati secara otomatis sesuai dengan keinginan user.

3.2. Perancangan Perangkat Keras

Perancangan perangkat keras yang direncanakan meliputi: pembuatan mekanik, blok diagram keseluruhan dan prinsip kerja alat, pembuatan skema seluruh rangkaian yang direncanakan, penghitungan nilai komponen yang digunakan, pembuatan PCB, dan perakitan seluruh komponen.

3.2.1. Blok Diagram Alat



Gambar 3.2. Blok diagram perangkat keras desain dan implementasi BAS dengan web dan wap

Penjelasan tiap blok :

1. Komputer.

Komputer pada perancangan ini berfungsi sebagai Server web dan wap yang terhubung dengan database mysql. Database mysql tersebut akan dibaca datanya oleh visual basic yang nantinya software visual basic berfungsi sebagai eksekutor berdasarkan database mysql dan kemudian hasil eksekusi tersebut akan dikirimkan secara serial ke mikrokontroler.

2. Minimum sistem AT89S51 :

Rangkaian ini berfungsi sebagai pengolah data yang dikomunikasikan secara serial oleh komputer. Kemudian hasil pengolahan data yang dilakukan oleh mikrokontroler akan dikirimkan ke driver relay sehingga relay akan menjadi *off/on*.

3. Driver Relay

Driver relay berisi rangkaian ULN 2003 yang berfungsi untuk menguatkan arus yang diberikan oleh mikrokontroler untuk mengaktifkan relay.

4. Relay

merupakan rangkaian yang bersifat elektronis sederhana dan tersusun oleh saklar, medan elektromagnetik dan poros besi.

5. Lampu / Stop kontak

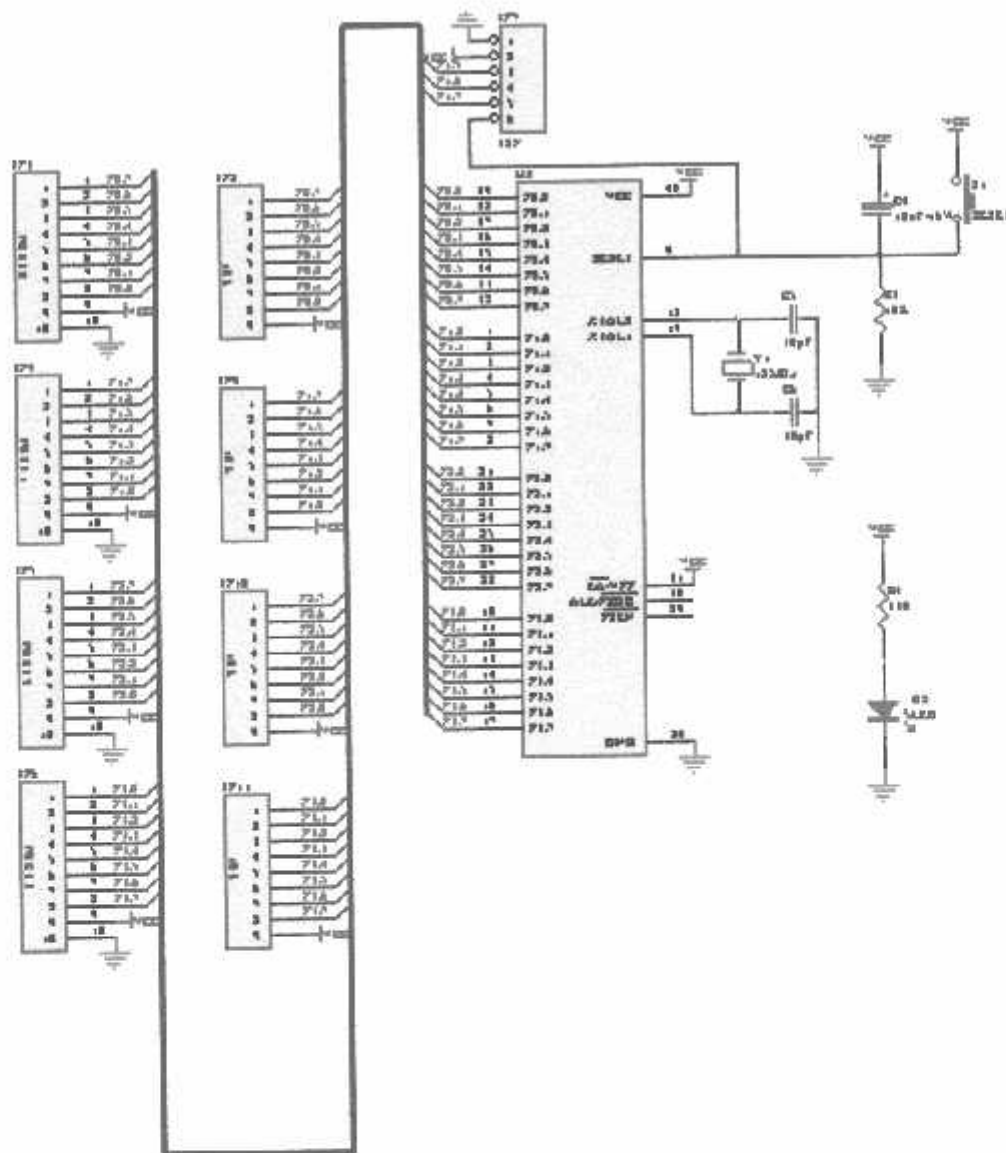
Merupakan obyek yang akan dikontrol melalui web dan wap.

3.2.2. Mikrokontroler AT89S51

Rangkaian mikrokontroler AT89S51 berfungsi sebagai pengolah data yang diperoleh dari komunikasi serial dengan komputer dan berfungsi sebagai pengontrol relay yang terhubung dengan lampu / stop kontak.

Pengaturan jalur *input* dan *output* pada rangkaian mikrokontroler untuk sebuah rancangan terprogram, sangat berkaitan erat dengan program yang kita buat. Agar tidak terjadi kesalahan saat pembacaan data, mikrokontroler menyediakan jalur-jalur 32 *input-output* yang dapat digunakan secara berkelompok atau bersamaan untuk tiap kelompok terisi 8 bit. Untuk lebih

jelasnya, hubungkan mikrokontroller dengan perangkat pendukungnya dapat dilihat dalam gambar di bawah ini :



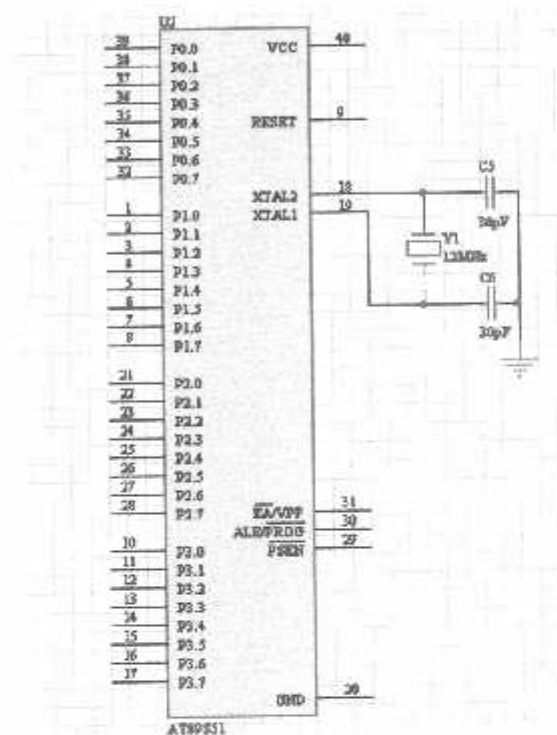
Gambar 3.3. Perancangan Sistem Mikrokontroller AT89S51

3.2.2.1. Rangkaian Clock

Kecepatan proses yang dilakukan oleh mikrokontroller ditentukan oleh sumber clock (pewaktuan) yang mengendalikan mikrokontroller tersebut. Sistem yang akan dirancang ini akan menggunakan osilator internal yang sudah tersedia

dalam chip mikrokontroler AT89S51. Untuk menentukan frekuensi osilatornya cukup dengan cara menghubungkan kristal pada pin 19 (XTAL 1) dan pin 18 (XTAL 2) serta dua buah kapasitor ke pentanahan (ground).

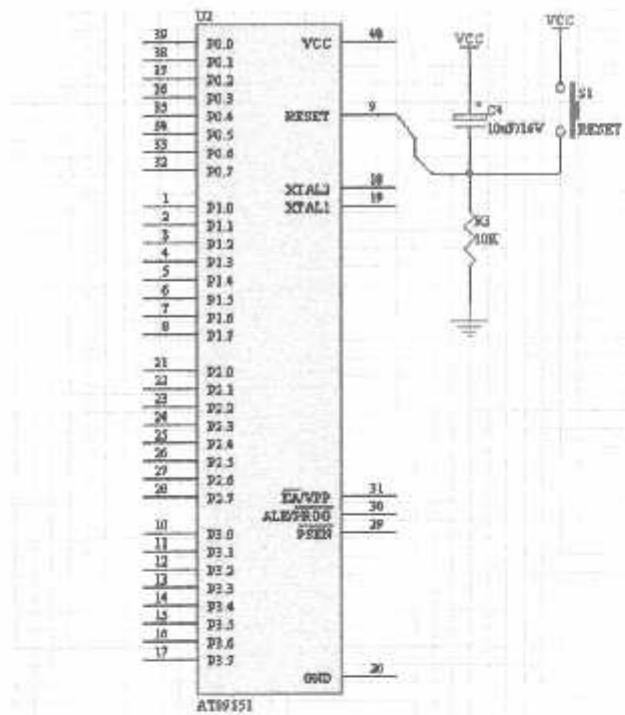
Besarnya kapasitansi, disesuaikan dengan spesifikasi pada lembar data AT89S8252 yaitu 30 pF. Kristal yang digunakan adalah 11,0592 MHz. Gambar 3.4 memperlihatkan rangkaian *clock* yang digunakan.



Gambar 3.4. Perancangan Rangkaian Clock

3.2.2.2. Rangkaian Reset

Untuk mereset mikrokontroler AT89S51, maka pin RST diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal *reset* kapasitor dihubungkan dengan Vcc dan sebuah resistor yang dihubungkan ke *ground*. Rangkaian reset ditunjukkan dalam gambar 3.5 sebagai berikut :



Gambar 3.5 Perancangan Rangkaian Reset

Karena kristal yang digunakan mempunyai frekuensi sebesar 11,0592 MHz, maka satu periode membutuhkan waktu sebesar :

$$\tau = \frac{1}{f} = \frac{1}{11,0592 \text{ MHz}} \cdot S = 9,042 \times 10^{-8} \text{ s}$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk mereset mikrokontroller adalah :

$$\begin{aligned} \text{Reset (min)} &= \tau \times \text{periode yang dibutuhkan} \\ &= 9,042 \times 10^{-8} \times 24 = 2,17 \mu\text{s} \end{aligned}$$

Jadi mikrokontroller membutuhkan waktu minimal 2,17 μs untuk mereset.

Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C.

Dari persamaan konstanta waktu $\tau = R \times C$ (William H Hyat, 1998, h132 [1])

dan jika nilai R ditentukan sebesar 10 k Ω , maka nilai C adalah :

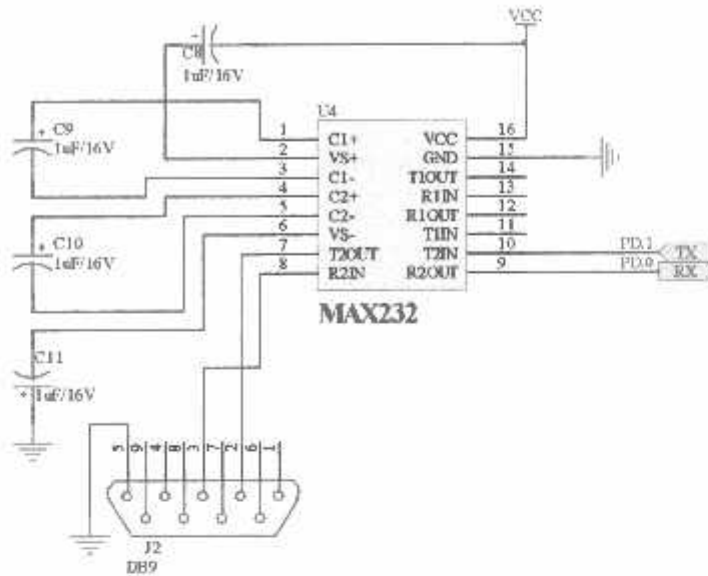
$$\begin{aligned}
 C &= \frac{\tau}{R} \\
 &= \frac{2,17 \times 10^{-6}}{10 \times 10^{-12}} \\
 &= 217 \times 10^{-12} \text{ F}
 \end{aligned}$$

Kapasitor minimal yang dibutuhkan adalah 217 pF. Dengan menggunakan kapasitor sebesar 10 μ F, maka akan menjamin waktu reset di atas nilai minimal waktu yang dibutuhkan untuk mereset mikrokontroler.

3.2.3 Rangkaian RS232 pada minimum sistem AT89S51

Level tegangan pada sinyal data serial yang ada pada mikrokontroler dengan computer tidaklah sama. Pada mikrokontroler level tegangan untuk sinyal data serial ialah 5V(Level TTL) sedangkan pada komputer memiliki tegangan untuk sinyal data serial level tegangan RS232. Untuk itu diperlukan rangkaian RS232 untuk dapat mengkonversi level tegangan sinyal data serial TTL menjadi level tegangan RS232.

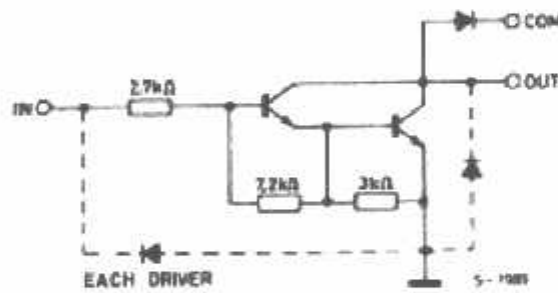
Rangkaian RS232 yang digunakan pada mikrokontroler AT89S51 menggunakan IC MAX232 yang dapat digunakan untuk mengkonversi level tegangan sinyal data serial TTL menjadi level RS232. Berikut ialah gambar rangkaian RS232 menggunakan IC MAX232 :



Gambar 3.6. Rangkaian RS232 menggunakan IC MAX232

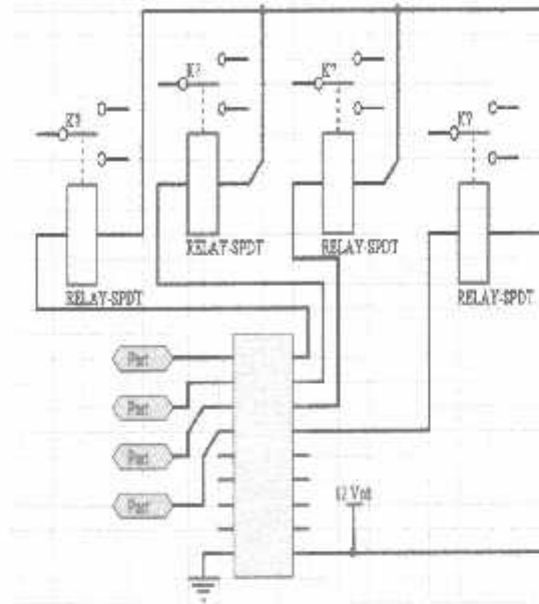
3.2.4. Rangkaian Driver Relay

Untuk pembuatan driver relay digunakan ULN 2003 dimana tiap pin input sampai pin output terdapat rangkaian



Gambar 3.7. Rangkaian driver pada ULN 2003

Dengan menggunakan rangkaian tersebut, maka relay dapat diatur melalui mikrokontroler sebagai inputan pada pin *in* ULN 2003



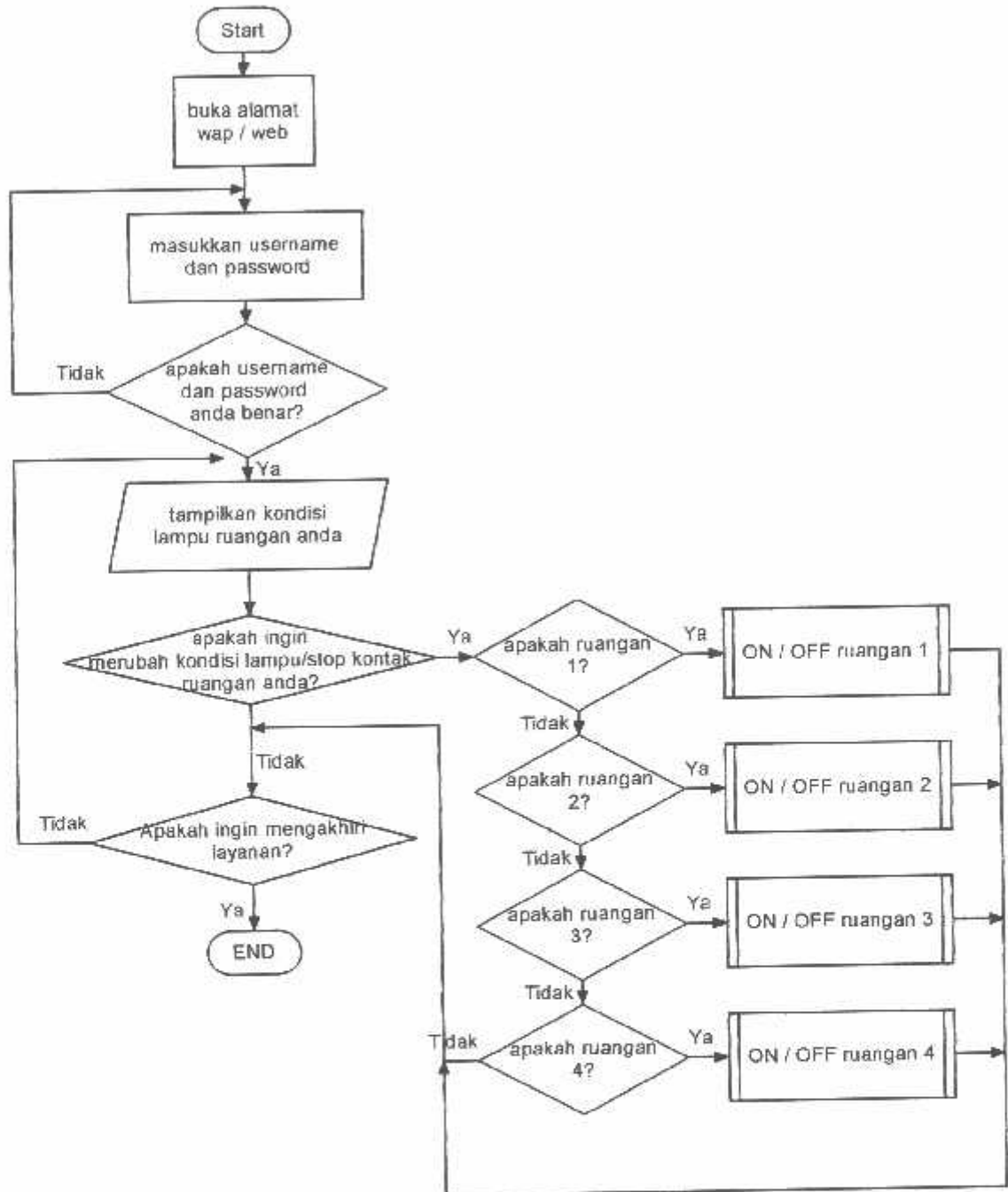
Gambar 3.8. Rangkaian Driver Relay

3.3. Perancangan Perangkat Lunak

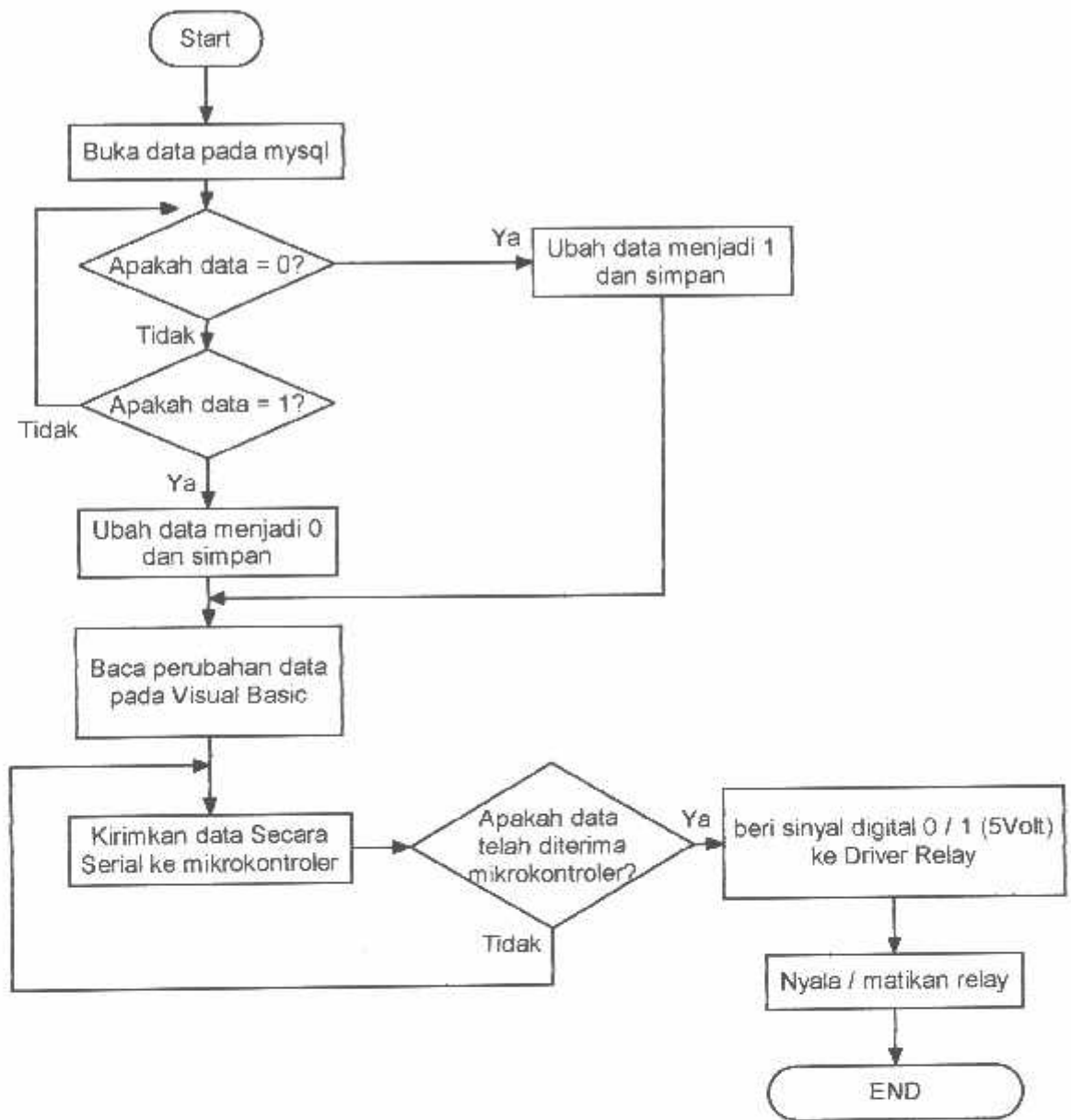
Perancangan perangkat lunak (*software*) yang digunakan dalam perencanaan dan pembuatan alat desain dan implementasi BAS dengan web dan wap akan dipaparkan dalam *flowchart* sistem secara keseluruhan. Pembuatan software hanya dilakukan pada mikrokontroler menggunakan bahasa pemrograman *Assembly*.

3.3.1. Diagram Alir Perancangan :

Flowchart rangkaian secara keseluruhan sebagai berikut :



Gambar 3,9 Flowchart system secara keseluruhan



Gambar 3.10 Flowchart proses mematikan / menyalakan lampu

Penjelasan FlowChart

1. *User* dapat mengakses ruangan yang dimiliki dengan mengakses alamat website atau melalui *wap browser* (pada *handphone*) penyedia fasilitas ini.
2. *User* diminta untuk memasukkan *username* dan *password* yang akan diproses dan dicari data dari *database*.

3. Apabila *user* terdaftar dalam *database* pengaturan ruangan ini, maka *user* dapat mengakses lampu / stop kontak dari tiap ruangan.
4. Apabila lampu / stop kontak ruangan yang dipilih dalam kondisi menyala, maka lampu / stop kontak tersebut akan dimatikan, begitu pula sebaliknya. Kondisi tersebut kemudian akan disimpan ke *database user*.
5. Adanya perubahan *database* tersebut akan dibaca oleh Visual Basic pada komputer server sehingga data tersebut akan dieksekusi dan dikirimkan secara serial ke mikrokontroler.
6. Setelah data yang dikirim oleh komputer secara serial tersebut telah diterima mikrokontroler, maka mikrokontroler tersebut akan mengirimkan sinyal digital 1 / 0 ke driver relay sesuai dengan eksekusi yang dilakukan oleh program Visual Basic.
7. Setelah kondisi lampu / stop kontak sudah sesuai dengan yang diinginkan *user*, maka *user* dapat *sign out*.

BAB IV

PENGUJIAN DAN HASIL ANALISA

Pengujian dilakukan untuk mengetahui sejauh mana peralatan dapat bekerja sesuai dengan perencanaan. Langkah pengujian dilakukan melalui 2 tahap, yakni pengujian pada setiap blok dan pengujian pada sistem keseluruhan. Tahap pertama dimaksudkan untuk mengetahui sejauh mana blok-blok rangkaian dapat berjalan, sedangkan tahap kedua dilakukan setelah diperoleh kepastian bahwa tiap blok rangkaian telah berjalan sesuai rencana. Pada tahap ini setiap blok rangkaian diintegrasikan menjadi desain dan implementasi *Building Automation System* dengan web dan wap.

4.1. Pengujian *Hardware*

4.1.1 Pengujian rangkaian driver relay

1. Alat yang digunakan :

- Rangkaian driver relay
- Multimeter Digital SANWA CD800a
- Catu daya 5, 12 dan 220 Volt

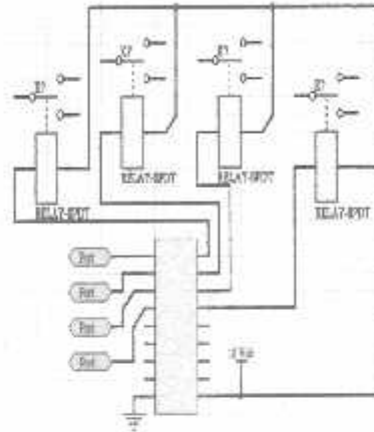
2. Metode Pengujian :

Gambar 4.1 menunjukkan rangkaian driver relay yang terdiri dari ULN 2003.

3. Langkah Pengujian :

- Merangkai rangkaian driver relay seperti pada gambar 4.1
- Memasang catu daya DC 5, 12, 220 V pada rangkaian.

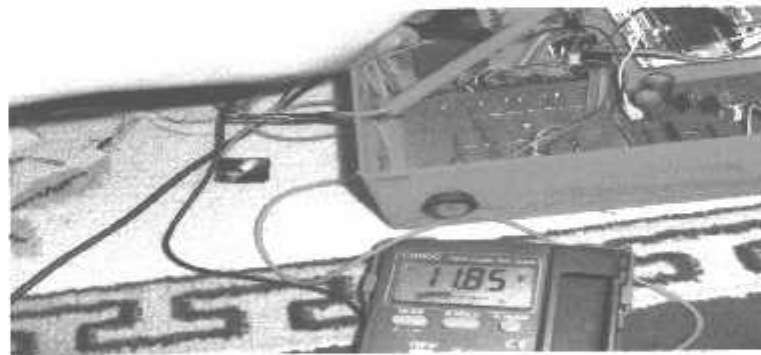
- Mengukur tegangan yang dilalui *relay* dengan *AVO meter* apabila relay dalam keadaan *on* dan *off*.



Gambar 4.1. Rangkaian Driver Relay

4. Hasil dan Analisa Pengujian Driver Relay

Hasil pengujian driver relay ditunjukkan dalam gambar 4.2, 4.3 dibawah ini.



Gambar 4.2 Pengujian rangkaian driver relay saat on



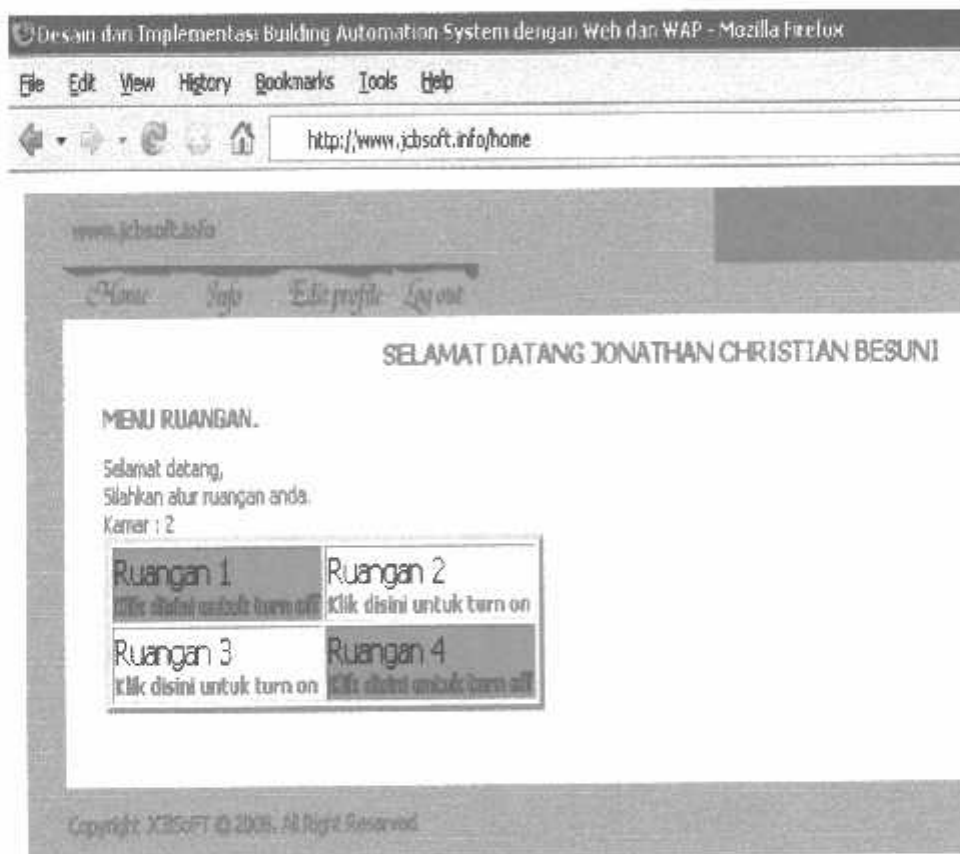
Gambar 4.3 Pengujian rangkaian driver relay saat off

3. Langkah Pengujian

- Menghubungkan box controller pada komputer secara serial.
- Mengakses halaman *web* dengan menggunakan *browser* dengan alamat *http://localhost/jcbsoft*

4. Hasil dan Analisa Pengujian

Hasil pengujian keseluruhan sistem setelah dijalankan ialah pada Gambar 4.5



Gambar 4.5. Tampilan pada saat ruangan 1 menyala, ruangan 2 mati, ruangan 3 mati dan ruangan 4 menyala.

Hasil pengujian alat dan program memberikan tampilan sesuai dengan kondisi ruangan pada tiap kamar. Jika ruangan dalam kondisi menyala, maka akan menampilkan tampilan informasi di web berupa

indikator warna merah pada blok ruangan dan jika ruangan dalam kondisi mati, maka akan menampilkan tampilan informasi di web berupa indikator warna putih pada blok ruangan.

Analisa program dalam mikrokontroler dapat dilakukan dengan mengikuti alur *flowchart* yang telah dibuat. Program pada mikrokontroler dibuat menggunakan bahasa *assembly* dengan MIDE-51 untuk MCS-51. Berikut ialah penjelasan mengenai program yang terdapat pada mikrokontroler AT89S51.

1. Bagian awal program terdiri dari program timer dengan kristal osilator mikrokontroler sebesar 11,059 MHz, dan baudrate yang akan digunakan sebesar 9,600 bit per second.

```
Start      : MOV TMOD,#20H
           : MOV TH1,#0FDH
           : MOV SCON,#50H
           : SETB TR1
           : SETB ES
           : SETB EA
```

2. Selanjutnya dilakukan dengan proses inisialisasi yang berkaitan dengan fasilitas atau fitur yang digunakan pada mikrokontroler AT89S51. Inisialisasi yang dilakukan ialah inisialisasi serial port, dimana Mikrokontroler akan mengirimkan karakter "O" dan "K" secara serial yang akan dikenali nantinya oleh program Visual Basic.

```
SERI_INT  : JB R1,YA
           : CLR T1
           : RET
YA        : MOV A,SBUF
           : CLR R1
           : MOV P1,A
           : MOV P0,P1
           : MOV R1,A
           : MOV A,#"O"
           : CLR T1
           : MOV SBUF,A
CEK_O    : JNB T1,$
           : MOV A,#"K"
```

```

          CLR T1
          MOV SBUF,A
CEK_K    : JNB T1,$
          CLR T1
          MOV A,R1
          MOV SBUF,A
          RET
END

```

3. Setelah proses inisialisasi selesai akan dilakukan dengan pengecekan terhadap karakter yang dikirim oleh mikrokontroler. Maka data serial tersebut akan dikirimkan oleh visual basic dan akan ditampilkan di P1(*Relay*) dan P0(*LED Display*)

Berikut ialah foto alat secara keseluruhan :



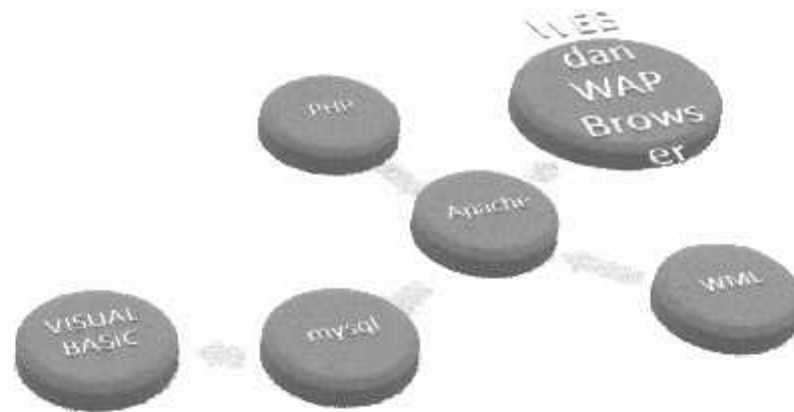
Gambar 4.6. Alat Desain dan Implementasi *Building Automation System* dengan Web dan WAP

4.2.2 Pengujian Program pada web dan wap

1. Peralatan yang digunakan :

- PC dengan operating sistem *Windows XP*.
- Web Browser dan WAP Browser

2. Metode Pengujian :



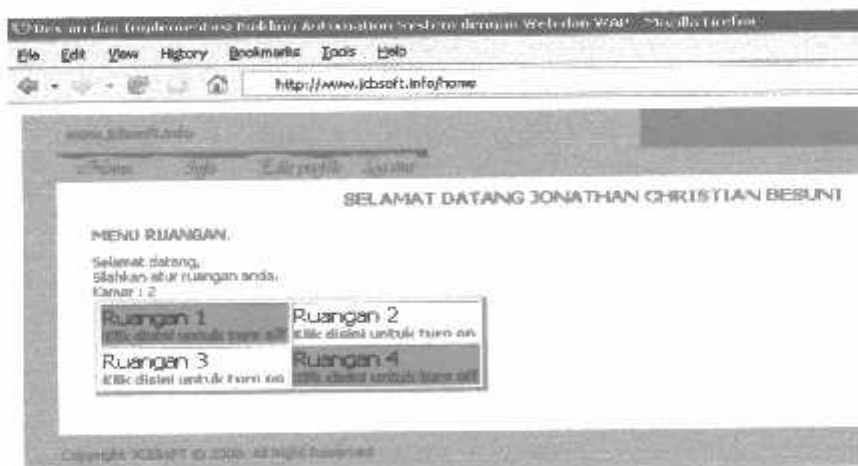
Gambar 4.7. Diagram blok pengujian *software* pada sistem secara keseluruhan

3. Langkah Pengujian

- Mengaktifkan apache server dan database mysql
- Mengakses halaman *web* dengan menggunakan *browser* dengan alamat <http://localhost/jcbsoft> dan mengakses halaman WAP dengan WAP browser dengan alamat <http://localhost/jcbsoft/wap>

4. Hasil dan Analisa Pengujian

Hasil pengujian keseluruhan sistem setelah dijalankan ialah pada Gambar 4.8 dan Gambar 4.9



Gambar 4.8. Tampilan web pada saat ruangan 1 menyala, ruangan 2 mati, ruangan 3 mati dan ruangan 4 menyala .



Gambar 4.9. Tampilan WAP pada saat ruangan 1 menyala, ruangan 2 mati, ruangan 3 mati dan ruangan 4 menyala .

Hasil pengujian program memberikan tampilan sesuai dengan kondisi ruangan pada tiap kamar. Kondisi yang ditampilkan pada web browser sesuai dengan tampilan yang dihasilkan pada wap browser. Pada Web Browser, jika ruangan dalam kondisi menyala, maka akan menampilkan tampilan informasi di web berupa indikator warna merah pada blok ruangan dan jika ruangan dalam kondisi mati, maka akan menampilkan tampilan informasi di web berupa indikator warna putih pada blok ruangan. Sedangkan pada WAP Browser, kondisi ruangan ditampilkan dengan indikator tulisan “menyala” atau “mati”.

Untuk menguji koneksi webservice dan wapbrowser pada server, maka dilakukan uji koneksi apakah web dan wap tersebut dapat dibuka di hp ataupun komputer dan tidak terjadi *lost connection*.

Tabel 4.1 Hasil pengujian Apache Server

Pengujian pada HP ke-	Kondisi	Pengujian pada Komputer ke-	Kondisi
1	<i>Connect</i>	1	<i>Connect</i>
2	<i>Connect</i>	2	<i>Connect</i>
3	<i>Connect</i>	3	<i>Connect</i>
4	<i>Connect</i>	4	<i>Connect</i>
5	<i>Connect</i>	5	<i>Connect</i>
6	<i>Connect</i>	6	<i>Connect</i>
7	<i>Connect</i>	7	<i>Connect</i>
8	<i>Connect</i>	8	<i>Connect</i>
9	<i>Connect</i>	9	<i>Connect</i>
10	<i>Connect</i>	10	<i>Connect</i>

Pengujian juga dilakukan dengan mengakses web dan wap secara bersamaan, dan melihat seberapa besarkah error yang dihasilkan saat aplikasi dijalankan bersama-sama

Tabel 4.2 Hasil pengujian WEB dan WAP

Pengujian dengan jumlah koneksi	Pengujian pada WAP	Pengujian pada Web
1 koneksi	Ruangan dapat dinyalakan /dimatikan tanpa <i>error</i>	Ruangan dapat dinyalakan /dimatikan tanpa <i>error</i>
Lebih dari 1 koneksi	Ruangan dapat dinyalakan /dimatikan tanpa <i>error</i> Namun kondisi ruangan sesuai dengan kondisi terakhir yang mengakses ruangan pada WAP/WEB	Ruangan dapat dinyalakan /dimatikan tanpa <i>error</i> Namun kondisi ruangan sesuai dengan kondisi terakhir yang mengakses ruangan pada WAP/WEB

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan perancangan, pembuatan, pengujian dan analisis alat Desain dan Implementasi Building Automation System dengan Web dan WAP, dapat disimpulkan:

1. Desain dan implementasi Building Automation System dengan Web dan WAP memberikan kemudahan bagi penggunanya dalam mengatur dan mengawasi suatu ruangan dimanapun dan kapan pun juga.
2. Pada pengujian driver relay, relay akan aktif saat diberi tegangan 11,85 volt yang merupakan tegangan output dari driver relay saat driver relay diberi inputan 0 V.
3. Dari pengujian koneksi Web dan WAP ke sever, penggunaan Apache sebagai server web dan wap cukup stabil.
4. Pada saat aplikasi ini diakses, mysql merupakan media penyimpanan data yang cukup stabil walaupun diakses melalui beberapa bahasa pemrograman lain sehingga dalam penggunaannya tidak terjadi *error* dalam database.
5. Kondisi yang ditampilkan pada ruangan, merupakan kondisi terakhir yang diupdate melalui wap dan web.

5.2 Saran

Pada alat hasil perancangan ini masih mempunyai kekurangan-kekurangan, untuk itu ada beberapa hal yang dapat dilakukan pengembangan lebih lanjut, yaitu:

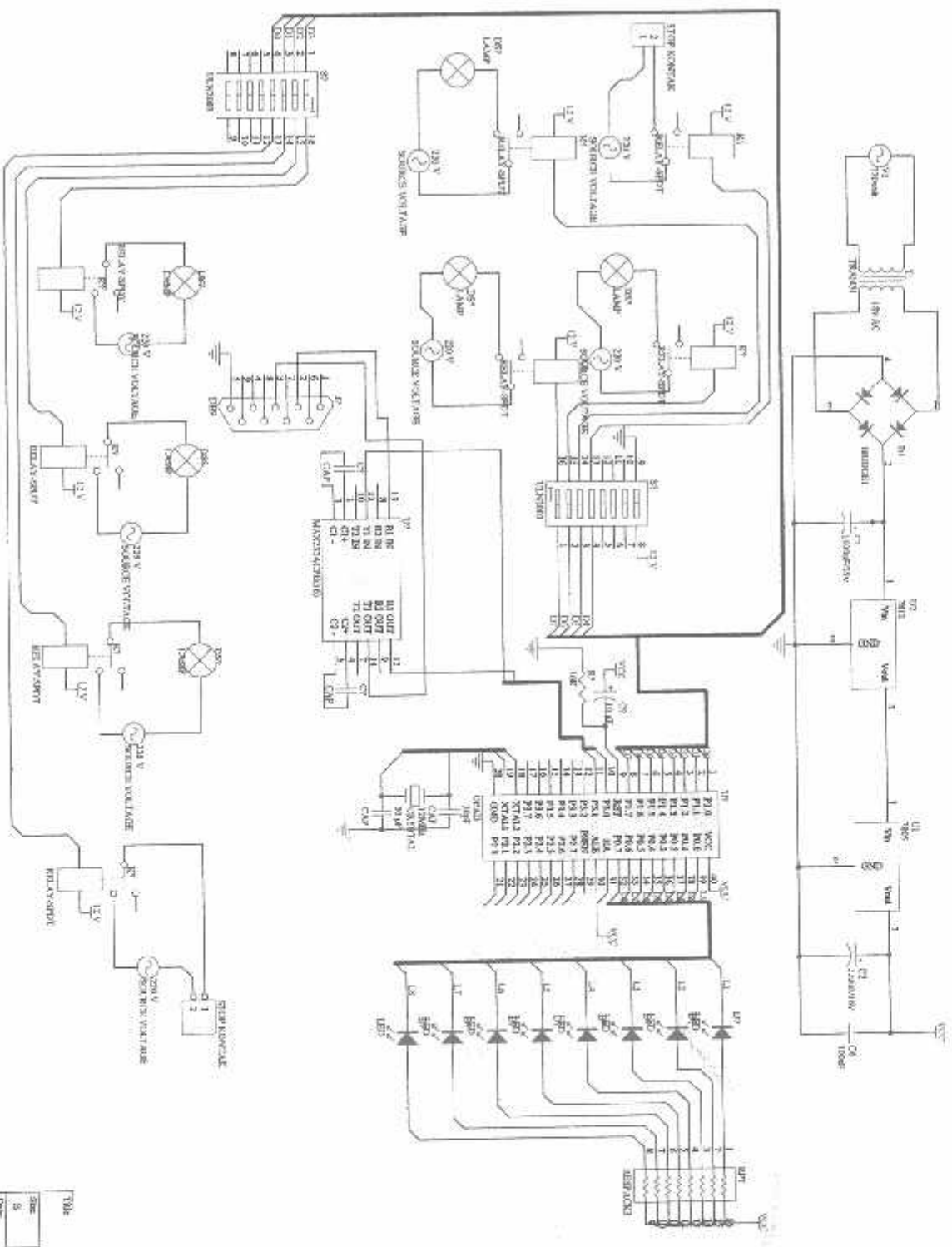
1. Pada aplikasi ini, tidak dapat mendeteksi apakah koneksi listrik yang hendak dikontrol mengalami gangguan atau tidak, sedangkan pada aplikasi ini hanya berfungsi sebagai saklar otomatis saja.
2. Pada web, bisa ditambahkan pengingat password berupa email yang dikirimkan ke pengguna berisi username dan password saat si pengguna lupa password.

DAFTAR PUSTAKA

- [1] en.wikipedia.org/wiki/php
 - [2] www.atmel.com
 - [3] Simarmata, Janner. *Aplikasi Mobile Commerce menggunakan PHP dan MySQL*. 2006. Yogyakarta: Penerbit ANDI
 - [4] Prasetya, Retno. *Inteface Port Paralel dan Port Serial Komputer dengan Visual Basic 6.0*. 2004. Yogyakarta: Penerbit ANDI
 - [5] Purwanto, Yudhi. *Pemrograman Web dan PHP*. 2000. Elex Media Komputindo
 - [6] Nugroho, Bunafit. *Database Relasional dengan MySQL*. 2004. Yogyakarta: Penerbit ANDI
-

LAMPIRAN

SKEMATIK RANGKAIAN



Tipe	
Size	Medium
Doc	23.340.008
File	1.047.0001.011
Sheet	1 from 1
Author	Barbara

**LISTING PROGRAM
VISUAL BASIC**

LISTING PROGRAM VISUAL BASIC

```
Private Sub mnConnect_Click()
```

```
Get_Connect
```

```
End Sub
```

```
Private Sub mnDisconnect_Click()
```

```
Get_Disconnect
```

```
End Sub
```

```
Private Sub mnExit_Click()
```

```
Unload Me
```

```
End Sub
```

```
Public Sub Get_Connect()
```

```
Dim i As Integer
```

```
On Error GoTo Handle_Error
```

```
MSComm1.PortOpen = True
```

```
MSComm1.Output = Chr$(Get_Value)
```

```
mnDisconnect.Checked = False
```

```
mnDisconnect.Enabled = True
```

```
mnConnect.Checked = True
```

```
mnConnect.Enabled = False
```

```
Exit Sub
```

```
Handle_Error:
```

```
Get_Disconnect
```

```
End Sub
```

```
Private Sub Get_Disconnect()
```

```
If MSComm1.PortOpen = True Then
```

```
MSComm1.PortOpen = False
```

```
End If
```

```
mnConnect.Checked = False
```

```
mnConnect.Enabled = True
```

```
mnDisconnect.Checked = True
```

```
mnDisconnect.Enabled = False
```

```
End Sub
```

```
Function Get_Value() As Integer
```

```
Get_Value = Label9.Caption
```

```
End Function
```

Private Static Sub MSComm1_OnComm()

Dim Buffer As Variant

Select Case MSComm1.CommEvent

Case comEvReceive

If MSComm1.InBufferCount = 3 Then

Buffer = CStr(MSComm1.Input)

Debug.Print "comEvReceive - " & StrConv(Buffer, vbUnicode) 'menampilkan data yang diterima pada jendela debug

If Mid(Buffer, 1, 1) = "O" Then

If Mid(Buffer, 2, 1) = "K" Then

End If

End If

End If

End Select

End Sub

Private Sub Timer1_Timer()

Adodc1.Refresh

Adodc2.Refresh

Adodc3.Refresh

Label9.Caption = (Label11.Caption * 128) + (Label2.Caption * 64) + (Label3.Caption * 32) + (Label4.Caption * 16) + (Label5.Caption * 8) + (Label6.Caption * 4) + (Label7.Caption * 2) + (Label8.Caption * 1)

Get_Connect

End Sub

Private Sub Timer2_Timer()

If MSComm1.PortOpen = True Then

Empty_Buffer = MSComm1.Input

MSComm1.Output = Chr\$(Get_Value)

End If

End Sub

LISTING PROGRAM

LISTING PROGRAM WEB

Update.php

```
<br />
```

```
<?>
```

```
$kondisi = $_GET['kondisi'];  
ECHO "<BR>";  
$ruang = $_GET['ruang'];  
ECHO "<BR>";  
$kamar = $_GET['kamar'];  
require 'common/config.inc.php';  
$sql="UPDATE tbl_ruangan  
      SET '$ruang.'" = "'$kondisi.'" where id_ruangan="'$kamar.'";  
$result = mysql_query($sql);  
?>
```

```
Update Ruangan <? echo $ruang;?> Berhasil..
```

```
<br />
```

```
<br />
```

```
<a href="masuk.php?module=utama">Lihat Ruangan Lagi</a>
```

Index.php

```
<?>
```

```
// mengambil fungsi sistem koneksi  
require './common/config.inc.php';  
// melemparkan sistem ke halaman default  
header ('location:default.php?module='.Shome);  
?>
```

Logout.php

```
<?>
```

```
session_start();  
session_destroy();  
require './common/config.inc.php';  
$info = sayLogout;  
header ('location:default.php?module='.Shome);  
?>
```

Login.php

```
<?
require './common/config.inc.php';
session_start();
if (array_key_exists("USERID", $_SESSION))
{
    define("USERID", $_SESSION["USERID"]);
    define("USERNAME", $_SESSION["USERNAME"]);
    define("USERREALNAME", $_SESSION["USERREALNAME"]);
    define("USERPASS", $_SESSION["USERPASS"]);
    define("USEREMAIL", $_SESSION["USEREMAIL"]);
    define("USERCLASS", $_SESSION["USERCLASS"]);
    define("USERCLASSNAME", $_SESSION["USERCLASSNAME"]);
}
else
{
    $username = $_POST['username'];
    $password = $_POST['password'];
    $sql = 'select a.id, a.username, a.password, a.nama, a.email, b.id as
kelas_id, b.nama as kelas from '.$db.$pengguna.' a join '.$db.$kelas.' b on a.kelas=b.id
where a.username = "'.$username.'" And a.password="'.$password.'" And a.aktif="1"';
    $result = mysql_query($sql);
    $row = mysql_fetch_object($result);
    if (($row->username==$username) AND ($row->password==$password)){
        $userclass = $row->kelas_id;
        define(USERID, $row->id);
        define(USERNAME, $row->username);
        define(USERREALNAME, $row->nama);
        define(USERPASS, $row->password);
        define(USEREMAIL, $row->email);
        define(USERCLASS, $row->kelas_id);
        define(USERCLASSNAME, $row->kelas);
        $_SESSION["USERID"] = USERID;
        $_SESSION["USERNAME"] = USERNAME;
        $_SESSION["USERREALNAME"] = USERREALNAME;
        $_SESSION["USERPASS"] = USERPASS;
        $_SESSION["USEREMAIL"] = USEREMAIL;
        $_SESSION["USERCLASS"] = USERCLASS;
        $_SESSION["USERCLASSNAME"] = USERCLASSNAME;
        header ('location:masuk.php?module='.$Shome.'');
    }
    else
    {
        header ('location:index.php');
    }
}
}
```


Ruangan.php

<?

```
require 'common/config.inc.php';
```

```
$sql = 'select alamat
        from tbl_pengguna
        where id="' . $userid . '";
$result = mysql_query($sql);
$row = mysql_fetch_object($result);
```

```
echo "Kamar : ";
echo $row->alamat;
```

```
$sql = 'select *
        from tbl_ruangan
        where id_ruangan="' . $id_ruangan . '";
$result = mysql_query($sql);
$row = mysql_fetch_object($result);
$L1 = $row->L1;
$L2 = $row->L2;
$L3 = $row->L3;
$L4 = $row->L4;
```

>

<table border="3">

<tr>

<td bgcolor="<? if (\$L1=='0')

```
{
echo "white";
$skondisi="1";
$skata = 'Klik disini untuk turn on';
}
```

else if (\$L1=='1')

```
{
echo "red";
$skondisi="0";
$skata = 'Klik disini untuk turn off';
}
```

?>> Ruang 1
<a href="masuk.php?module=lampu&ruang=L1&kamar=<? echo

\$id_ruangan ?>&kondisi=<? echo \$skondisi ?>> <? echo \$skata; ?></td>

<td bgcolor="<? if (\$L2=='0')

```
{
echo "white";
```

```

Skondisi="1";
Skata = 'Klik disini untuk turn on';
}
else if ($L2=='1')
{
echo "red";
Skondisi="0";
Skata = 'Klik disini untuk turn off';
}
?>"> Ruang 2 <br /><a href="masuk.php?module=lampu&ruang=L2&kamar=<? echo
Sid_ruangan ?>&kondisi=<? echo $kondisi ?>"> <? echo $kata; ?></a></td>
</tr>
<tr>
<td bgcolor="<? if ($L3=='0')
{
echo "white";
Skondisi="1";
Skata = 'Klik disini untuk turn on';
}
else if ($L3=='1')
{
echo "red";
Skondisi="0";
Skata = 'Klik disini untuk turn off';
}
?>"> Ruang 3 <br /><a href="masuk.php?module=lampu&ruang=L3&kamar=<? echo
Sid_ruangan ?>&kondisi=<? echo $kondisi ?>"> <? echo $kata; ?></a></td>
<td bgcolor="<? if ($L4=='0')
{
echo "white";
Skondisi="1";
Skata = 'Klik disini untuk turn on';
}
else if ($L4=='1')
{
echo "red";
Skondisi="0";
Skata = 'Klik disini untuk turn off';
}
?>"> Ruang 4 <br /><a href="masuk.php?module=lampu&ruang=L4&kamar=<? echo
Sid_ruangan ?>&kondisi=<? echo $kondisi ?>"> <? echo $kata; ?></a></td>
</tr>
</table>

```

Config.inc

<?

```
// konfigurasi sistem
$dbhostname = "localhost"; // setting nama host
$dbusername = "root";      // setting username (tergantung setting sistem)
$dbpassword = "";         // setting password (diisi jika pada setting awal
memang menggunakan password)
$dbdatabase = "jcbsoft_x"; // setting nama database
$db          = "tbl_";     // setting prefix nama database
// cek koneksi sistem
mysql_connect($dbhostname,$dbusername,$dbpassword) or die ("Tidak bisa Koneksi,
cek hostname, username atau password pada sistem Anda!");
// cek koneksi database
mysql_select_db($dbdatabase) or die ("Tidak bisa terkoneksi dengan
database" . $dbdatabase);
```

?>

right_user.php

```
<?
include "table_description.php";
?>
<table width="100%" border="0" cellspacing="0" cellpadding="0">
  <tr>
    <td width="24" background="<?= $rowtmp->template;
?>images/image_r6_c2.jpg">&nbsp;&nbsp;&nbsp;</td>
    <td>
      <table width="100%" border="0" cellspacing="0" cellpadding="5">
        <tr>
          <td width="170">
            <? if ($module==home){ ?>
              
            <? } else if ($module==berita){ ?>
              
            <? } else if ($module==pengguna){ ?>
              
            <? } else if ($module==kelas){ ?>
              
            <? } else if ($module==stemplate){ ?>
              
            <? } else if ($module==foldertemplate){ ?>
              
            <? } ?>
          </td>
          <td class="big" align="left"><?= $infoLogin; ?></td>
        </tr>
      </table>
    </td>
  </tr>
  <tr>
    <td colspan="2" class="bold">
      Update Kamar </td>
  </tr>
  <tr>
    <td colspan="2" class="small"><?

```

```

echo "lampu";
include ("update.php");?>
</td>
</tr>
</
}
if ($module==$home){
    if ($userclass == 1){
?>
        <tr>
        <td colspan="2" class="bold">
            MENU ADMINISTRATOR.</td>
        </tr>
        <tr>
            <td colspan="2" class="small">Sebagai Super Admin, Anda memiliki
akses penuh untuk membuat hirarki user, memperkenankan user untuk mengakses
website Anda sesuai dengan tingkatannya masing-masing. Anda hanya diperkenankan
untuk menambah, mengedit, memblokir, dan menghapus keanggotaan dari anggota
website.</td>
        </tr>
</?
        } else if ($userclass == 2){
?>
            <tr>
            <td colspan="2" class="bold">
                MENU OPERATOR.</td>
            </tr>
            <tr>
                <td colspan="2" class="small">Sebagai operator, Anda memiliki akses
untuk mengontrol semua data yang telah masuk.</td>
            </tr>
</?
        } else {
?>
            <tr>
            <td colspan="2" class="bold">
                MENU RUANGAN.</td>
            </tr>
            <tr>
                <td colspan="2" class="small">Selamat datang,<br /> Silahkan atur
ruangan anda.<br />
            </tr>
            <?php
            include('ruangan.php');
            ?>

```

```
</td>
</tr>
<?
}
} else if ($module==Stable) {
    include "table.php";
}
?>
</td>
</tr>
<tr>
<td>&nbsp;</td>
</tr>
</table>
</td>
</tr>
</table>
```

Right.php

```
<table width="100%" border="0" cellspacing="0" cellpadding="0">
  <tr>
    <td width="24" background="<?=$rowtmp->template;
?>images/image_r6_c2.jpg">&nbsp;</td>
    <td>
      <table width="100%" border="0" cellspacing="0" cellpadding="5">
        <tr>
          <td>
            <? if ($module== $home) { ?>
              
            <? } else if ($module== $news) { ?>
              
            <? } else if ($module== $register) { ?>
              
            <? } ?>
          </td>
        </tr>
        <tr>
          <td class="small">
            <table width="100%" border="0" cellspacing="0" cellpadding="5">
              <?
              if (isset($info)) {
                ?>
                <tr valign="top">
                  <td colspan="3" class="warning"><?=$info; ?></td>
                </tr>
              <?
              }
            <?
          <?
          if ($module== $pengguna) {
            $username = $_POST['username'];
            $password = $_POST['password'];
            $password1 = $_POST['re-password'];
            $class = $_POST['kelas'];
            $email = $_POST['email'];
            $alamat = $_POST['alamat'];
            $nama = $_POST['nama'];
            if ($action== $add) {
              $field = (!empty($username) && !empty($password1)
&& !empty($nama) && !empty($email));
```

```

        <td class="small">Nama* </td>
        <td class="small"> </td>
        <td class="small"><input name="nama" value="<?= $rowdata-
>nama; ?>" size="30" type="text" class="search" /></td>
    </tr>
    <tr valign="top">
        <td class="small">Email* </td>
        <td class="small"> </td>
        <td class="small"><input name="email" value="<?= $rowdata-
>email; ?>" size="30" type="text" class="search" /></td>
    </tr>
    <tr valign="top">
        <td class="small">Alamat </td>
        <td class="small"> </td>
        <td class="small"><textarea name="alamat" rows="4" cols="50"
class="search"><?= $rowdata->alamat; ?></textarea></td>
    </tr>
    <tr valign="top">
        <td colspan="3"><input name="action" value="add"
type="hidden" />
        <input name="submit" class="search" type="submit"
value="kirim" /><input name="reset" class="search" type="reset" value="batal" /></td>
    </tr>
</form>
<?
} else if (($module==$berita) || ($module==$home)){
    if (isset($_GET['id'])){
        $id = $_GET['id'];
        $where = "where aktif='1' and id='".$id."'";
    } else {
        $where = "where aktif='1'";
    }
    $first = "select * from ".$db.$berita." ".$where."";
    $second = "select count(*) from ".$db.$berita." ".$where."";
    // beri batas data
    if ($module==$home) {
        $limit = 1;
    } else {
        $limit = 10;
    }
    // merubah nama query
    $sqltable = $first;
    // menset halaman dimulai dari halaman 1
    $page = 1;
    // memeriksa apakah ada pemanggilan page -- page-
    if(isset($_POST['page'])) {

```



```

    Stambah = "insert into ".Sdb.$pengguna." values
(default, '".Susername."', '".Spassword1."', '".Semail."', '".Snama."', '".Salamat."', '3',
NOW(), '1')";

```

```

    if ($field){
        // merubah nama variabel query
        $insert = $stambah;
        // jalankan query
        $rstbh = mysql_query($insert);
        // jika query yang dijalankan salah
        if (!$rstbh){
            $info = _sayError;
        }
        // jika eksekusi query benar
        else {
            $info = "Data baru telah ditambah!";
        }
    }
    // jika salah satu atau beberapa field tidak terisi
    else {
        $info = _sayFill;
    }
}
}

```

```

?>
<form method="post">
<tr valign="top">
    <td colspan="3" class="big">Formulir Pendaftaran</td>
</tr>
<tr valign="top">
    <td width="19%" class="small">Username</td>
    <td width="1%" class="small"></td>
    <td class="small"><input name="username" value="<?=$rowdata->username; ?>" size="30" type="text" class="search" /></td>
</tr>
<tr valign="top">
    <td class="small">Password* </td>
    <td class="small"></td>
    <td class="small"><input name="password" value="<?=$rowdata->password; ?>" size="30" type="password" class="search" /></td>
</tr>
<tr valign="top">
    <td class="small">Ulangi Password* </td>
    <td class="small"></td>
    <td class="small"><input name="re-password" value="<?=$rowdata->password; ?>" size="30" type="password" class="search" /></td>
</tr>
<tr valign="top">

```

```

        $page = $_POST["page"];
    }
    // merubah nama query yang kedua (menghitung jumlah data)
    $sqlsecond = $second;
    // menjalankan query kedua
    $rssecond = mysql_query($sqlsecond);
    // cek jumlah data
    $jumlah = mysql_fetch_row($rssecond);
    // untuk mengetahui jumlah halaman
    $pi = ceil($jumlah[0]/$limit);

    // jika halaman pada hasil query lebih kecil dengan setting halaman
    yang telah dibuat, maka set halaman tetap menjadi halaman 1
    if ($pi < $page ) {
        $page=1;
    }

    // menjalankan query dengan sistem pagging, menggunakan limit
    data yang akan ditampilkan
    $sqltable = " $sqltable." order by id desc limit " . $limit." offset
    ". ($page-1)*$limit;
    $rstable = mysql_query($sqltable);
    if (($jumlah[0]==0) && (!isset($_GET['id']))) {
?>
    <tr>
    <td colspan="2" class="warning"><?-_sayNodata; ?></td>
    </tr>
    <?
    } else if (($module!= $home) && (!isset($_GET['id']))) {
?>
    <tr>
    <td colspan="2" class="small">
    <form method="post">
    Total : <?- $jumlah[0]; ?>, Halaman :
    <select class="search" name="page">;
    <?
    // jika halaman bukan 1
    if ($page!=1) {
    echo ' <option selected class="search"
value="" . $page.'">'. $page '</option>',
    }
    for($page = 1; $page <= $pi; $page++) {
    echo ' <option class="search"
value="" . $page.'">'. $page '</option>',
    }
?>

```

```

        </select>
        <input name="submit" type="submit" class="search" value="lihat"
    </form>
</td>
</tr>
<?
    }

    while($rownews = mysql_fetch_object($rstable)){
?>
    <tr valign="top">
        <td width="1%" class="big">&bull;</td>
        <td width="99%" class="big"><a href="<?=
$PHP_SELF."?module=".$berita."&id=".$rownews->id; ?>"><?= $rownews->judul;
?></a></td>
    </tr>
    <?
        if (($module==Shome) || (isset($_GET['id']))) {
?>
        <tr valign="top">
            <td colspan="2" class="small">
                <?
                    if ($rownews->gambar!=""){
                        echo ';
                    }
                <?
                    <?= nl2br($rownews->isi); ?></td>
            </tr>
            <tr valign="top">
                <td colspan="2" class="very_small">Oleh : <?= $rownews-
>nama; ?>, Tanggal : <?= $rownews->tanggal; ?></td>
            </tr>
        <?
        }
    }
?>
</table>
</td>
</tr>
</table>
</table>

```

LISTING PROGRAM WAP

Index.php

```
<?
session_start();
session_destroy();
header("content-type:text/vnd.wap.wml");
echo "<?xml version='1.0'?'>";
?>

<wml>
<card id="card1" title="==HOME==">
<p align="center"><br />
<b>Selamat Datang di Sistem <I>Building Automation System</I> berbasis WAP.</b>
<br />
<a href="#card2">Login Here</a>
<br />
</p>
Sdate=date("D, M d y");
print "<small>$date</small>";
?>
</p>
</card>
<card id="card2" title="==Login==">

Username :<input type="text" name="nim" label="">
Password :<input type="password" name="pas" label="">

    <anchor>
    <go method="post" href="login.php">
    <postfield name="nim" value="$nim"/>
    <postfield name="pas" value="$pas"/>
    </go>
    Login
    </anchor>
</card>
</wml>
```

Login.php

```
<?
session_start();
header("content-type:text/vnd.wap.wml");
echo "<?xml version='1.0'?">";
?>

<wml>
<card id="card1" title="—HOME—">
<?
include("dbase/koneksi.inc");
$username=$_POST['nim'];
$password=$_POST['pas'];
$query=mysql_query("select * from tbl_pengguna where username='".$username.'" and
password='".$password.'''");
if(mysql_num_rows($query))
{
$data=mysql_fetch_object($query);
$nama=$data->nama;
$alamat=$data->alamat;
$id=$data->id;

session_register('nama');
session_register('username');
session_register('alamat');

print "<b><u> Login Sukses </u> </u> <br> <br> <small>Selamat
datang".$nama."<br>Silahkan periksa dan atur ruangan anda.
<br>Klik <a
href='ruangan.php?alamat=".$alamat."&id=".$id."&nama=".$nama.'">disini</a> untuk
lanjut.</small>";

}
else
{
print "Login salah";

}
?>
</card>
</wml>
```

Ruangan.php

```
<?
session_start();
/*
$nama = $_SESSION['nama'];
$username = $_SESSION['username'];
$alamat = $_SESSION['alamat'];
*/
header("content-type:text/vnd.wap.wml");
echo "<?xml version='1.0'?>";
include ("config.php");
?>

<wml>
<card id="card1" title="== <? echo $nama; ?> ==">
Silahkan pilih ruangan yang akan diatur:<br />
Ruangan 1:<br />
kondisi=
<?
if ($L1=='0')
{
echo "mati";
$ruang="L1";
$skondisi ="1";
}
else if ($L1=='1')
{
echo "nyala";
$ruang="L1";
$skondisi ="0";
}

$Sb="alamat=".$alamat."&id=".$Sid."&nama=".$nama."&kondisi=".$skondisi."&ruang=".$S
ruang;
?>
<br />
<a href="update.php?<? echo $Sb; ?>">Ubah</a> <br />

Ruangan 2:<br />
kondisi=
<?
if ($L2=='0')
{
echo "mati";
$ruang="L2";
$skondisi ="1";
```

```
}  
else if ($L2=='1')  
{  
echo "nyala";  
$ruang="L2";  
$kondisi="0";  
}
```

```
$b="alamat=".Salamat."&id=".Sid."&nama=".Snama."&kondisi=".Skondisi."&ruang=".S  
ruang;  
?>  
<br />  
<a href="update.php"?<? echo $b; ?>">Ubah</a> <br />
```

Ruangan 3:


```
kondisi=  
<?  
if ($L3=='0')  
{  
echo "mati";  
$ruang="L3";  
$kondisi="1";  
}  
else if ($L3=='1')  
{  
echo "nyala";  
$ruang="L3";  
$kondisi="0";  
}
```

```
$b="alamat=".$alamat."&id=".$id."&nama=".$nama."&kondisi=".$kondisi."&ruang=".$  
ruang;  
?>  
<br />  
<a href="update.php"?<? echo $b; ?>">Ubah</a> <br />
```

Ruangan 4:


```
kondisi=  
<?  
if ($L4=='0')  
{  
echo "mati";  
$ruang="L4";  
$kondisi="1";  
}  
else if ($L4=='1')
```



```
{
echo "nyala";
$ruang="L4";
$skondisi="0";
}
```

```
$b="alamat=". $alamat. "&id=". $id. "&nama=". $nama. "&kondisi=". $skondisi. "&ruang=". $ruang;
```

```
?>
```

```
<br />
```

```
<a href="update.php"?<? echo $b; ?>">Ubah</a> <br />
```

```
<a href="index.php">LogOut</a>
```

```
</card>
```

```
</wml>
```

Update.php

```
<?
session_start();
header("content-type:text/vnd.wap.wml");
echo "<?xml version='1.0'?>";
include ("config.php");
$skondisi = $_GET['kondisi'];
$ruang = $_GET['ruang'];
?>

<wml>
<card id="card1" title="" == <? echo $nama; ?> ==">

<?
$sql="UPDATE tbl_ruangan
        SET ".$ruang." = ".$skondisi." where id_ruangan=".$alamat."";

        mysql_query($sql);
?>

Update Ruang <? echo $ruang; ?> Berhasil<br />
Klik <a href='ruangan.php'?<? echo $a;?>'>disini</a> untuk kembali.

</card>
</wml>
```

Config.php

```
<?
include("dbase/koneksi.inc");

$nama = $_GET['nama'];
$alamat = $_GET['alamat'];
$id = $_GET['id'];

Sa="alamat=".$alamat."&id=".$id."&nama=".$nama;

echo $id;
echo $alamat;

$sql = "select * from tbl_ruangan where id_ruangan='".$alamat."'";
$query=mysql_query($sql);

if(mysql_num_rows($query))
{

$data=mysql_fetch_object($query);
$L1=$data->L1;
$L2=$data->L2;
$L3=$data->L3;
$L4=$data->L4;

}
?>
```

Koneksi.inc

```
<?
session_start();
$hostname="localhost";
$dbuser="root";
$password="";
$dbname="jcbsoft_x";
if(!mysql_connect($hostname,$dbuser,$password)) echo "ga konek";
if(!mysql_select_db($dbname)) echo "database ga konek";
?>
```

**LISTING PROGRAM
ASSEMBLY AT89S51**

```

;awal kode program pada memori Flash PEROM
    ORG 0000H
    SJMP START ;lompat ke START
;vektor interupsi serial
    ORG 0023H ;lokasi vektor interupsi port serial
    SJMP SERI_INT ;jika terjadi interupsi lompat ke SERI_INT
START: MOV TMOD,#20H ;TIMER1 mode 2 (8 bit isi ulang)
    MOV TH1,#0FDH ;9600 bit per second
    MOV SCON,#50H ;mode serial 8 bit UART
    SETB TR1 ;start TIMER1
    SETB ES
    SETB EA ;interupsi enable
KALANG: SJMP KALANG ;program akan berputar disini
;penanganan interupsi serial
SERI_INT: JB RI, YA ;jika interupsi penerimaan data maka lompat ke YA
    CLR TI ;jika interupsi pengiriman data, hapus TI
    RETI ;abaikan interupsi
YA: MOV A,SBUF ;baca karakter yang diterima pada register A
    CLR RI ;hapus RI agar bisa menerima lagi
    MOV P1,A
    mov p0,p1 ;set PORT1 sesuai isi karakter
    MOV R1,A ;simpan isi register A di register R1
    MOV A,#'O' ;akan dikirim karakter 'O' ke komputer
    CLR TI
    MOV SBUF,A ;mengirimkan karakter 'O' ke komputer

```

CEK_O: JNB TI,CEK_O ;cek apakah karakter sudah selesai dikirimkan

MOV A,#'K'

CLR TI

MOV SBUF,A ;mengirimkan karakter 'K' ke komputer

CEK_K: JNB TI,CEK_K

CLR TI

MOV A,R1

MOV SBUF,A ;kirim kembali karakter yang diterima ke komputer

RETI ;kembali ke pemanggil

END

DATASHEET

ULN 2003



LEMBAR BIMBINGAN SKRIPSI

Nama : Jonathan Christian Besuni
Nim : 04.12.212
Jurusan : Teknik Elektro Strata -1
Konsentrasi : Elektronika
Judul Skripsi : Desain dan Implementasi *Building Automation System* dengan Web dan WAP
Tanggal Pengajuan : 7 Juli 2008
Tanggal Penyelesaian : 23 September 2008
Dosen Pembimbing I : Ir. F. Yudi Limpraptono, MT
Dosen Pembimbing II : Yoseph Dedy Irawan, ST, MT
Telah Dievaluasi Dengan Nilai I : 90 (sembilan puluh)
Telah Dievaluasi Dengan Nilai II : 87 (delapan puluh enam)

Malang, Oktober 2008
Mengetahui,

Dosen Pembimbing I

Ir F. Yudi Limpraptono, ST, MT

Dosen Pembimbing II

Joseph Dedy Irawan, ST, MT



FORMULIR BIMBINGAN SKRIPSI

Nama : JONATHAN CHRISTIAN BESUNI
Nim : 0412212
Masa Bimbingan : 7 Juli 2008 s/d 7 Januari 2009
Judul Skripsi : Desain dan Implementasi Building Automation System Berbasis Web dan WAP

NO	Tanggal	Uraian	Paraf Pembimbing
1	2/8/08	Pendahuluan Bab I & II	
2		Bab III - RUMAH	
3		Bab IV	
4		Simpulan	
5		Bab V	
6			
7			
8			
9			
10			

Malang,
Dosen Pembimbing

Ir. Yudi Limpraptono, MT.
NIP.Y. 1039500274

Form



FORMULIR BIMBINGAN SKRIPSI

Nama : JONATHAN CHRISTIAN BESUNI
Nim : 0412212
Masa Bimbingan : 7 Juli 2008 s/d 7 Januari 2009
Judul Skripsi : Desain dan Implementasi Building Automation System Berbasis Web dan WAP

NO	Tanggal	Uraian	Paraf Pembimbing
1	20/8 '08	Bab I & II Ok	
2	20/8 '08	Bab III (Revisi)	
3	1/9 '08	Bab III Ok	
4	4/9 '08	Seminar	
5	4/9 '08	Bab V Ok	
6	22/9 '08	Final KOMPM	
7			
8			
9			
10			

Malang,
Dosen Pembimbing

Joseph Deddy Irawan, MT
NIP. 152315178

Form S-4b



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

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

NAMA
NIM
Perbaikan meliputi:

SOUMITHANU S. 13
014 12 219

1) kurang penyusunan di tabel!
Kemudian kembali ke 1!

Haspind front & Spacer.



Formulir Perbaikan Ujian Skripsi

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

NAMA : Jonathan Cristian B
NIM : 04.13.212
Perbaikan meliputi :

Diliat dibuku banyak sepeali yg
harus diperbaiki mulai bab 1 s/d bab V
dng tata tulis laporan.

Malang, 23 Sept 2008


(_____)



FORMULIR PERBAIKAN SKRIPSI

Dari hasil ujian komprehensif jenjang strata satu (S-1) Jurusan Teknik Elektro
Konsentrasi Elektronika yang diselenggarakan pada

Hari : Selasa

Tanggal : 23 September 2008

Telah dilakukan perbaikan skripsi oleh saudara:

Nama : Jonathan Christian Besuni

Nim : 04.12.212

Perbaikan meliputi:

No	Materi Perbaikan	Keterangan
1	Hasil Pengeluaran ditulis, kemudian analisa	
2	Masalah font dan spasi	
3	Dilihat di buku banyak sekali yang harus diperbaiki mulai bab 1 s/d bab 5 tentang tata tulis laporan	

Penguji Pertama

Komang Somawirata, ST, MT

Penguji Kedua

Irmalia Suryani F, ST, MT

Mengetahui,

Dosen Pembimbing I

Ir F. Yudi Limpraptono, ST, MT

Dosen Pembimbing II

Joseph Dedy Irawan, ST, MT

SURAT PUAS TUGAS / PRAKTIKUM

Nomor : ITN 083/SKATS/IX/SPT / 2008

Menerangkan bahwa mahasiswa :

Nama : JONATHAN CHRISTIAN B NIM : 0412212

Jurusan : Teknik Elektronika SI

Angkatan : 2004 (.....)

Telah menyelesaikan tugas : SKATS

Semester : 90 (Semester kedua)

Dengan Hasil : 90 (Semester kedua)

Malang, 22-9-2008

Dosen / Asisten yhs

R. F. Galil, MS

SURAT PUAS TUGAS / PRAKTIKUM

Nomor : ITN 001 / TA / 09 / SPT / 200 8

Menerangkan bahwa mahasiswa :

Nama : JONATHAN CHRISTIAN NIM : 09120212

Jurusan : Teknik Elektro 2

Angkatan : ()

Telah menyelesaikan tugas : SKRIPSI

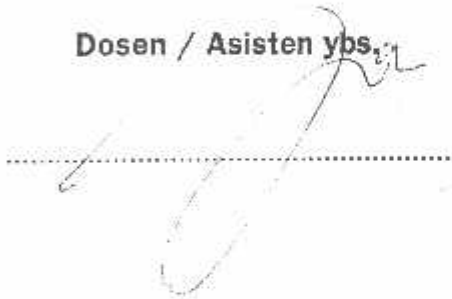
Semester : ()

Dengan Hasil : ()

Malang, 21 - 9 - 200 8

87

Dosen / Asisten ybs.



DATASHEET



DATASHEET
MIKROKONTROLER
AT89S51

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 clock 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

2487B-MICRO-12/03

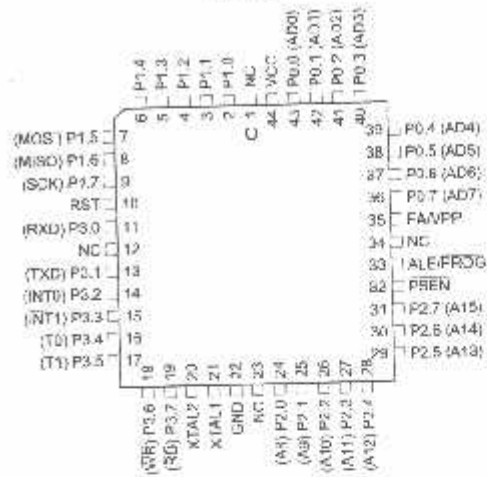


Pin Configurations

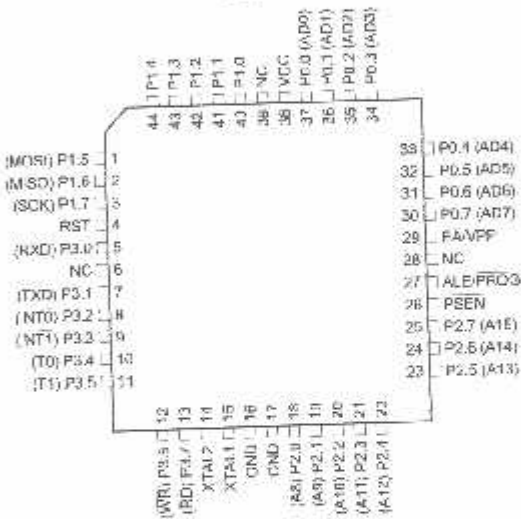
PDIP



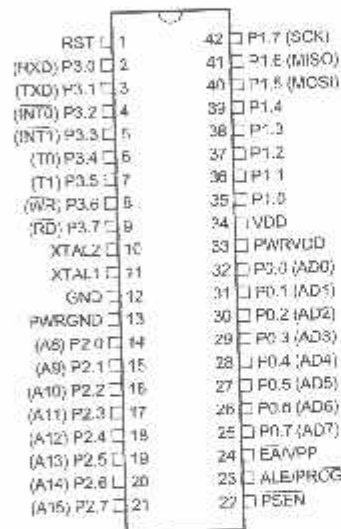
PLCC



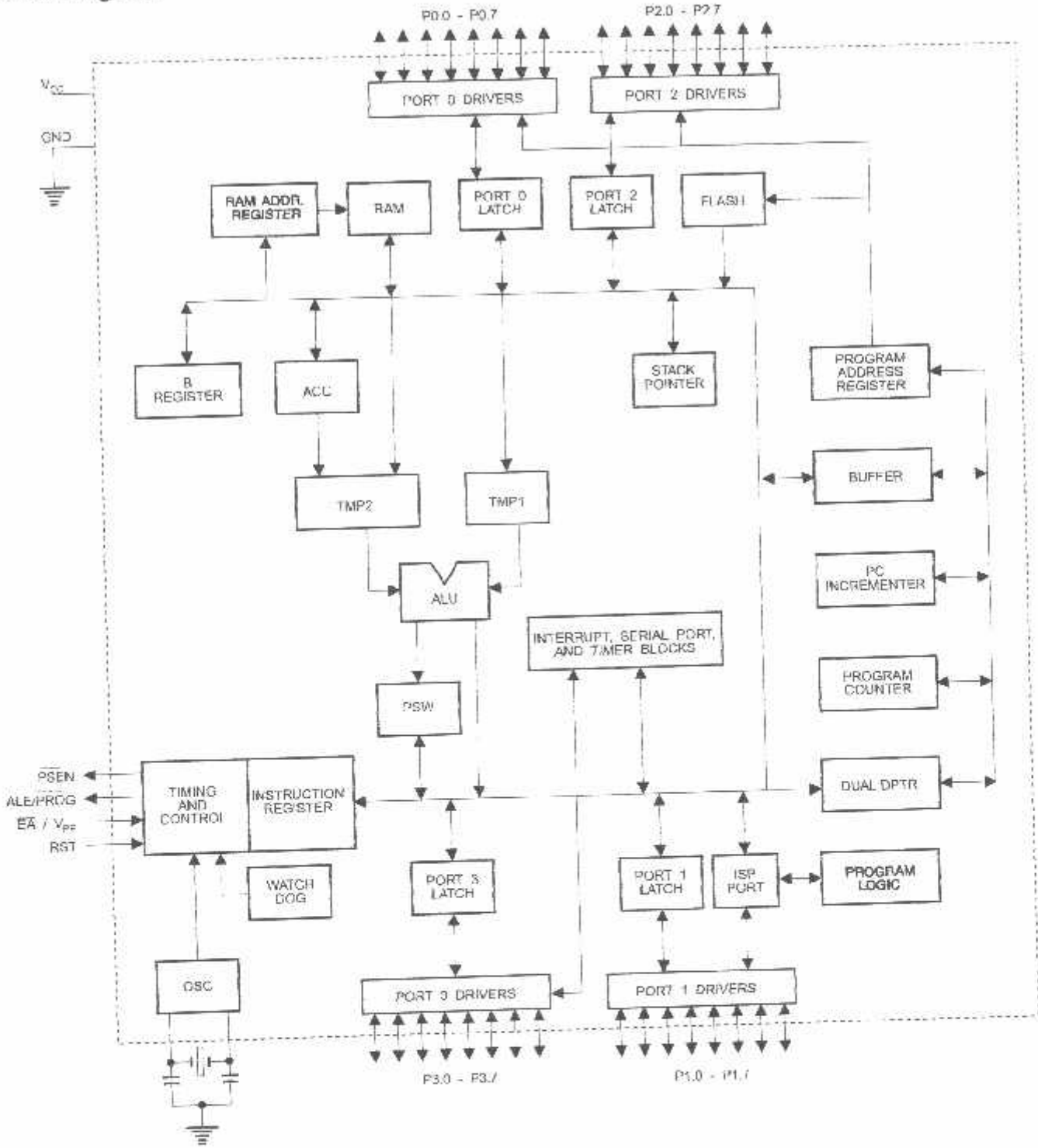
TQFP



PDIP



Block Diagram



Pin Description

VCC Supply voltage (all packages except 42-PDIP).

GND Ground (all packages except 42-PDIP; for 42-PDIP GND connects only the logic core and the embedded program memory).

VDD Supply voltage for the 42-PDIP which connects only the logic core and the embedded program memory.

PWRVDD Supply voltage for the 42-PDIP which connects only the I/O Pad Drivers. The application board **MUST** connect both VDD and PWRVDD to the board supply voltage.

PWRGND Ground for the 42-PDIP which connects only the I/O Pad Drivers. PWRGND and GND are weakly connected through the common silicon substrate, but not through any metal link. The application board **MUST** connect both GND and PWRGND to the board ground.

Port 0 Port 0 is an 8-bit open drain bi-directional 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 bi-directional 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 bi-directional 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 bi-directional 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	INT0 (external interrupt 0)
P3.3	INT1 (external interrupt 1)
P3.4	T0 (timer 0 external input)
P3.5	T1 (timer 1 external input)
P3.6	WR (external data memory write strobe)
P3.7	RD (external data memory read strobe)

RST

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.

ALE/PROG

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 (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.

PSEN

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.

EA/VPP

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.

XTAL1

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

XTAL2

Output from the inverting oscillator amplifier

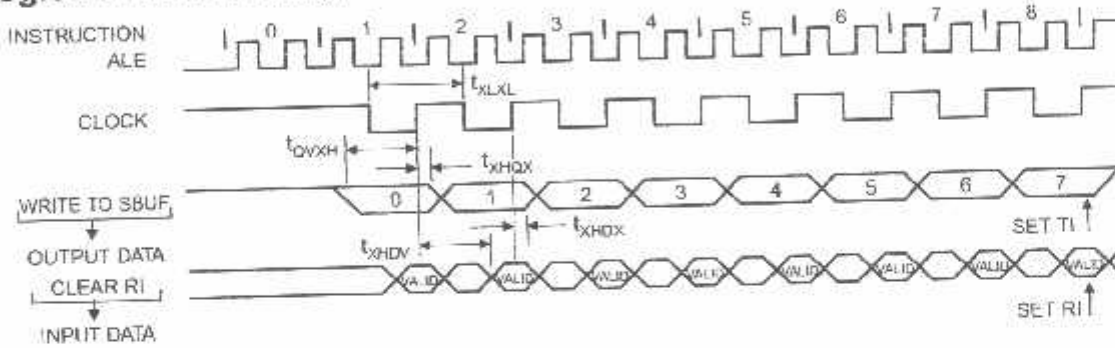


Serial Port Timing: Shift Register Mode Test Conditions

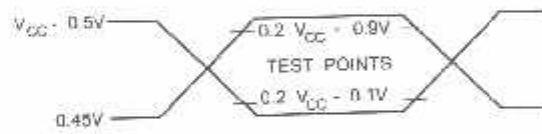
The values in this table are valid for $V_{CC} = 4.0V$ to $5.5V$ and Load Capacitance = $80 pF$.

Symbol	Parameter	12 MHz Osc		Variable Oscillator		Units
		Min	Max	Min	Max	
t_{XLX1}	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_{XHGX}	Output Data Hold After Clock Rising Edge	50		$2t_{CLCL} - 80$		ns
t_{XHGX}	Input Data Hold After Clock Rising Edge	0		0		ns
t_{XHGV}	Clock Rising Edge to Input Data Valid		700		$10t_{CLCL} - 133$	ns

Shift Register Mode Timing Waveforms

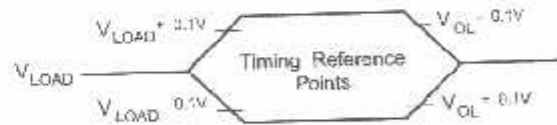


AC Testing Input/Output Waveforms⁽¹⁾



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

Float 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.

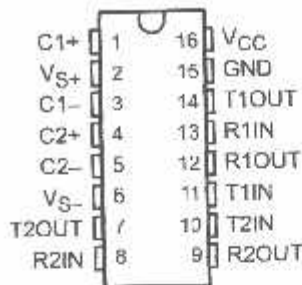
DATASHEET
IC MAX232

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS017I – FEBRUARY 1999 – REVISED OCTOBER 2002

- Meet or Exceed TIA/EIA-232-F and ITU Recommendation V.28
- Operate With Single 5-V Power Supply
- Operate Up to 120 kbit/s
- Two Drivers and Two Receivers
- ± 30 -V Input Levels
- Low Supply Current . . . 8 mA Typical
- Designed to be Interchangeable With Maxim MAX232
- ESD Protection Exceeds JESD 22 – 2000-V Human-Body Model (A114-A)
- Applications
 - TIA/EIA-232-F
 - Battery-Powered Systems
 - Terminals
 - Modems
 - Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
MAX232I . . . D, DW, OR N PACKAGE
(TOP VIEW)



description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ± 30 -V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

TA	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube	MAX232N	MAX232N
	SOIC (D)	Tube	MAX232D	MAX232
		Tape and reel	MAX232DR	
	SOIC (DW)	Tube	MAX232DW	MAX232
		Tape and reel	MAX232DWR	
SOP (NS)	Tape and reel	MAX232NSR	MAX232	
-40°C to 85°C	PDIP (N)	Tube	MAX232IN	MAX232IN
	SOIC (D)	Tube	MAX232ID	MAX232I
		Tape and reel	MAX232IDR	
	SOIC (DW)	Tube	MAX232IDW	MAX232I
		Tape and reel	MAX232IDWR	

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.



Please be aware that an important notice concerning availability, standard warranty, and use in critical applications of Texas Instruments semiconductor products and disclaimers thereto appears at the end of this data sheet.

LinASIC is a trademark of Texas Instruments.

PRODUCTION DATA information is current as of publication date. Products conform to specifications per the terms of Texas Instruments standard warranty. Production processing does not necessarily include testing of all parameters.

**TEXAS
INSTRUMENTS**

POB 1 OFFICE BOX 655303 • DALLAS, TEXAS 75265

Copyright © 2002, Texas Instruments Incorporated

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLLS047I - FEBRUARY 1989 - REVISED OCTOBER 2002

Function Tables

EACH DRIVER

INPUT T _{IN}	OUTPUT T _{OUT}
L	H
H	L

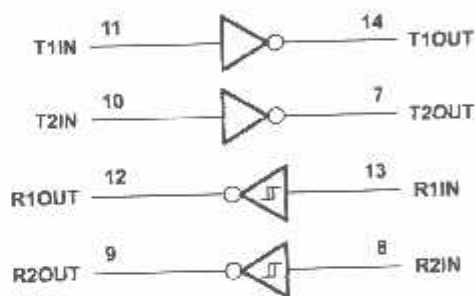
H = high level, L = low level

EACH RECEIVER

INPUT R _{IN}	OUTPUT R _{OUT}
L	H
H	L

H = high level, L = low level

logic diagram (positive logic)



**TEXAS
INSTRUMENTS**

POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

DUAL EIA-232 DRIVERS/RECEIVERS

SLLS247I – FEBRUARY 1989 – REVISED OCTOBER 2002

absolute maximum ratings over operating free-air temperature range (unless otherwise noted)†

Input supply voltage range, V_{CC} (see Note 1)	-0.3 V to 6 V
Positive output supply voltage range, V_{S+}	$V_{CC} - 0.3$ V to 15 V
Negative output supply voltage range, V_{S-}	-0.3 V to -15 V
Input voltage range, V_I : Driver	-0.3 V to $V_{CC} + 0.3$ V
Receiver	± 30 V
Output voltage range, V_O : T1OUT, T2OUT	$V_{S-} - 0.3$ V to $V_{S+} + 0.3$ V
R1OUT, R2OUT	-0.3 V to $V_{CC} + 0.3$ V
Short-circuit duration: T1OUT, T2OUT	Unlimited
Package thermal impedance, θ_{JA} (see Note 2): D package	73°C/W
DW package	57°C/W
N package	67°C/W
NS package	64°C/W
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds	260°C
Storage temperature range, T_{stg}	-65°C to 150°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

NOTE 1: All voltage values are with respect to network ground terminal.

2. The package thermal impedance is calculated in accordance with JESD 51-7.

recommended operating conditions

		MIN	NOM	MAX	UNIT
V_{CC}	Supply voltage	4.5	5	5.5	V
V_{IH}	High-level input voltage (T1IN, T2IN)	2			V
V_{IL}	Low-level input voltage (T1IN, T2IN)			0.8	V
R1IN, R2IN	Receiver input voltage			+30	V
T_A	Operating free-air temperature	MAX232	0	70	°C
		MAX232I	-40	85	

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature (unless otherwise noted) (see Note 3 and Figure 4)

PARAMETER	TEST CONDITIONS	MIN	TYP‡	MAX	UNIT
I_{CC} Supply current	$V_{CC} = 5.5$ V, All outputs open, $T_A = 25^\circ\text{C}$		8	10	mA

‡ All typical values are at $V_{CC} = 5$ V and $T_A = 25^\circ\text{C}$.

NOTE 3: Test conditions are C1-C4 = 1 μF at $V_{CC} = 5 \text{ V} \pm 0.5 \text{ V}$.

MAX232, MAX232I DUAL EIA-232 DRIVERS/RECEIVERS

SLL5047I - FEBRUARY 1989 - REVISED OCTOBER 2002

DRIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 3)

PARAMETER		TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V _{OH}	High-level output voltage	T1OUT, T2OUT R _L = 3 kΩ to GND	5	7		V
V _{OL}	Low-level output voltage‡	T1OUT, T2OUT R _L = 3 kΩ to GND		-7	-5	V
r _o	Output resistance	T1OUT, T2OUT V _{S+} = V _{S-} = 0, V _O = ±2 V	300			Ω
I _{OS} §	Short-circuit output current	T1OUT, T2OUT V _{CC} = 5.5 V, V _O = 0		-10		mA
I _{IS}	Short-circuit input current	T1IN, T2IN V _I = 0			200	μA

† All typical values are at V_{CC} = 5 V, T_A = 25°C.

‡ The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

§ Not more than one output should be shorted at a time.

NOTE 3: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 3)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
SR	Driver slew rate	R _L = 3 kΩ to 7 kΩ, See Figure 2			30	V/μs
SR(t)	Driver transition region slew rate	See Figure 3		3		V/μs
	Data rate	One TOUT switching		120		kbit/s

NOTE 3: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

RECEIVER SECTION

electrical characteristics over recommended ranges of supply voltage and operating free-air temperature range (see Note 3)

PARAMETER		TEST CONDITIONS	MIN	TYP†	MAX	UNIT
V _{OH}	High-level output voltage	R1OUT, R2OUT I _{OH} = -1 mA	3.5			V
V _{OL}	Low-level output voltage‡	R1OUT, R2OUT I _{OL} = 3.2 mA			0.4	V
V _{IT+}	Receiver positive-going input threshold voltage	R1IN, R2IN V _{CC} = 5 V, T _A = 25°C		1.7	2.4	V
V _{IT-}	Receiver negative-going input threshold voltage	R1IN, R2IN V _{CC} = 5 V, T _A = 25°C	0.8	1.2		V
V _{hys}	input hysteresis voltage	R1IN, R2IN V _{CC} = 5 V	0.2	0.5	1	V
r _i	Receiver input resistance	R1IN, R2IN V _{CC} = 5, T _A = 25°C	3	5	7	kΩ

† All typical values are at V_{CC} = 5 V, T_A = 25°C.

‡ The algebraic convention, in which the least positive (most negative) value is designated minimum, is used in this data sheet for logic voltage levels only.

NOTE 3: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see Note 3 and Figure 1)

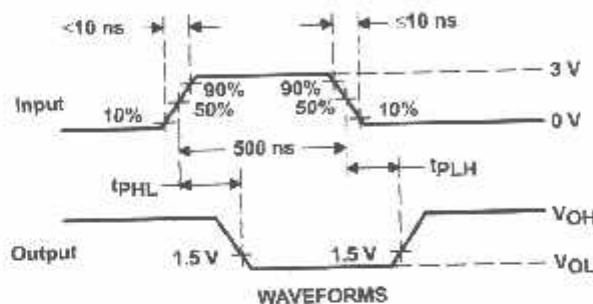
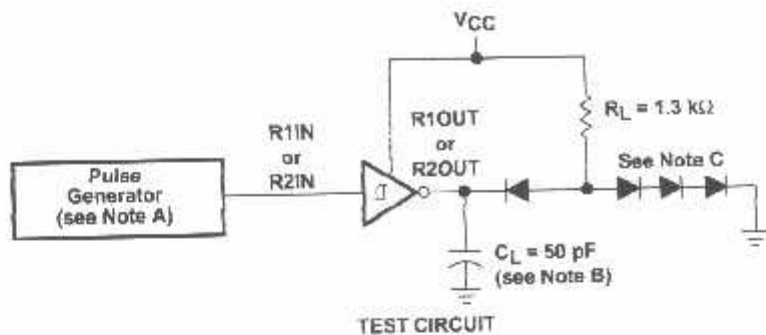
PARAMETER		TYP	UNIT
t _{PLH(R)}	Receiver propagation delay time, low- to high-level output	500	ns
t _{PHL(R)}	Receiver propagation delay time, high- to low-level output	500	ns

NOTE 3: Test conditions are C1-C4 = 1 μF at V_{CC} = 5 V ± 0.5 V.



POST OFFICE BOX 655303 • DALLAS, TEXAS 75265

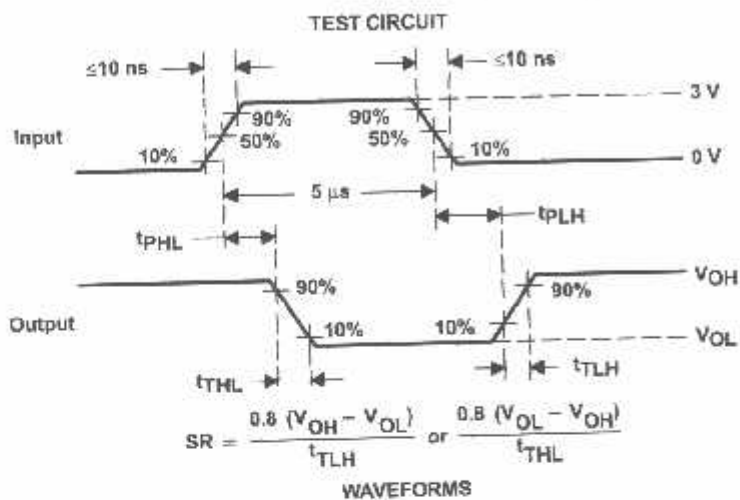
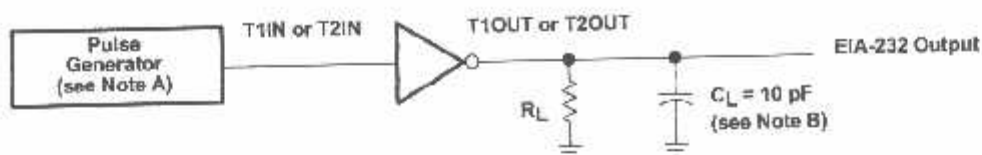
PARAMETER MEASUREMENT INFORMATION



- NOTES: A. The pulse generator has the following characteristics: $Z_O = 50 \Omega$, duty cycle $\leq 50\%$.
 B. C_L includes probe and jig capacitance.
 C. All diodes are 1N3064 or equivalent.

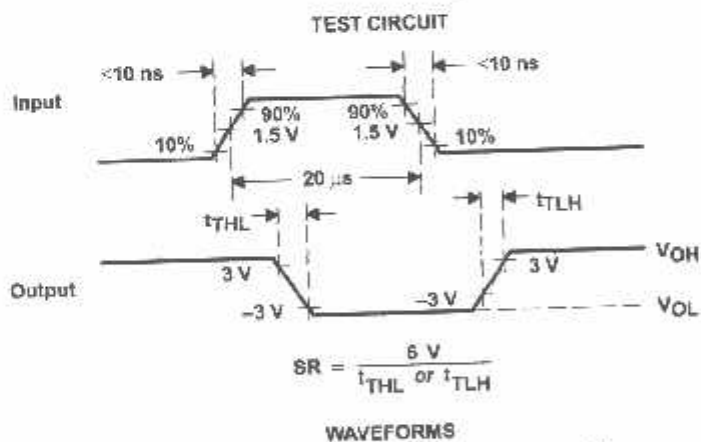
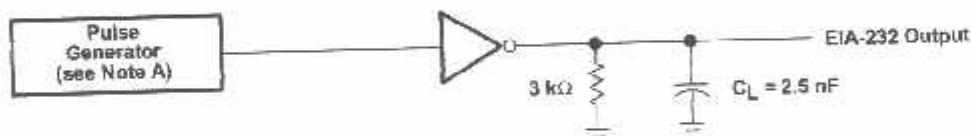
Figure 1. Receiver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements

PARAMETER MEASUREMENT INFORMATION



NOTES: A. The pulse generator has the following characteristics: $Z_O = 50 \Omega$, duty cycle $\leq 50\%$.
B. C_L includes probe and jig capacitance.

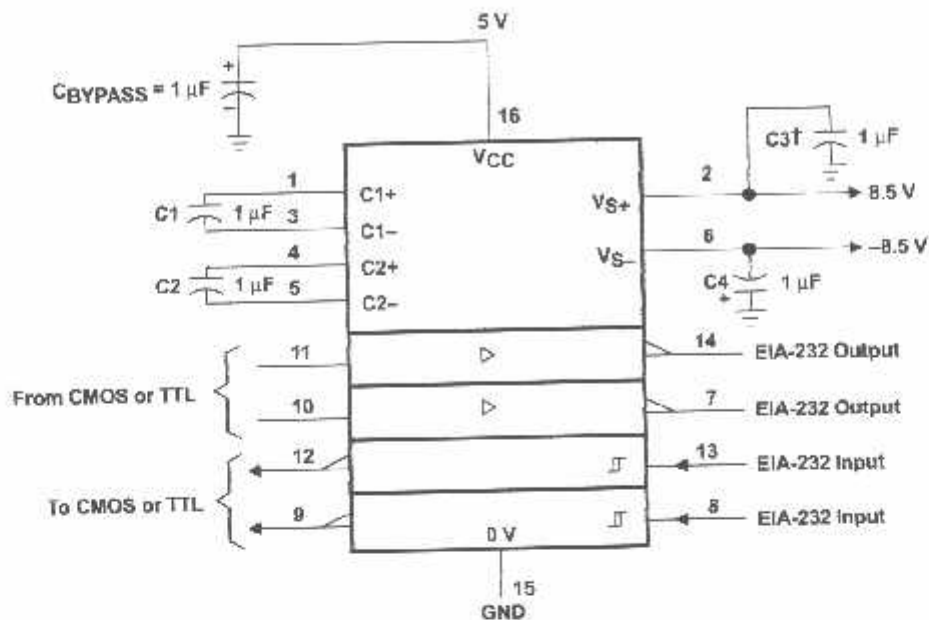
Figure 2. Driver Test Circuit and Waveforms for t_{PHL} and t_{PLH} Measurements (5- μ s Input)



NOTE A: The pulse generator has the following characteristics: $Z_O = 50 \Omega$, duty cycle $< 50\%$.

Figure 3. Test Circuit and Waveforms for t_{THL} and t_{TLH} Measurements (20- μ s Input)

APPLICATION INFORMATION



† C3 can be connected to VCC or GND.

Figure 4. Typical Operating Circuit



ULN200xA ULN200xD1

Seven darlington array

Features

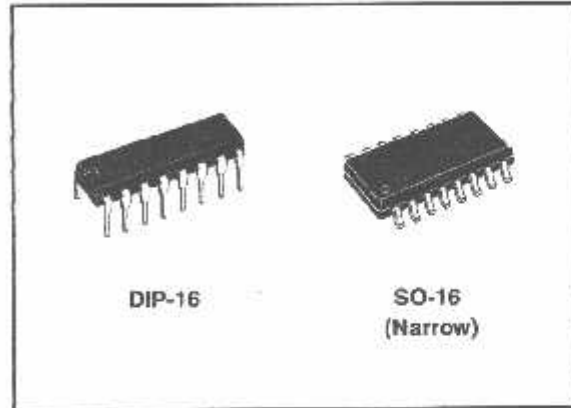
- Seven darlington pairs per package
- Output current 500 mA per driver (600 mA peak)
- Output voltage 50 V
- Integrated suppression diodes for inductive loads
- Outputs can be paralleled for higher current
- TTL/CMOS/PMOS/DTL Compatible inputs
- Inputs pinned opposite outputs to simplify layout

Description

The ULN2001, ULN2002, ULN2003 and ULN2004 are high voltage, high current darlington arrays each containing seven open collector darlington pairs with common emitters. Each channel rated at 500 mA and can withstand peak currents of 600 mA. Suppression diodes are included for inductive load driving and the inputs are pinned opposite the outputs to simplify board layout.

The versions interface to all common logic families:

- ULN2001 (general purpose, DTL, TTL, PMOS, CMOS)
- ULN2002 (14-25V PMOS)
- ULN2003 (5V TTL, CMOS)
- ULN2004 (6-15V CMOS, PMOS)



These versatile devices are useful for driving a wide range of loads including solenoids, relays DC motors, LED displays filament lamps, thermal printheads and high power buffers.

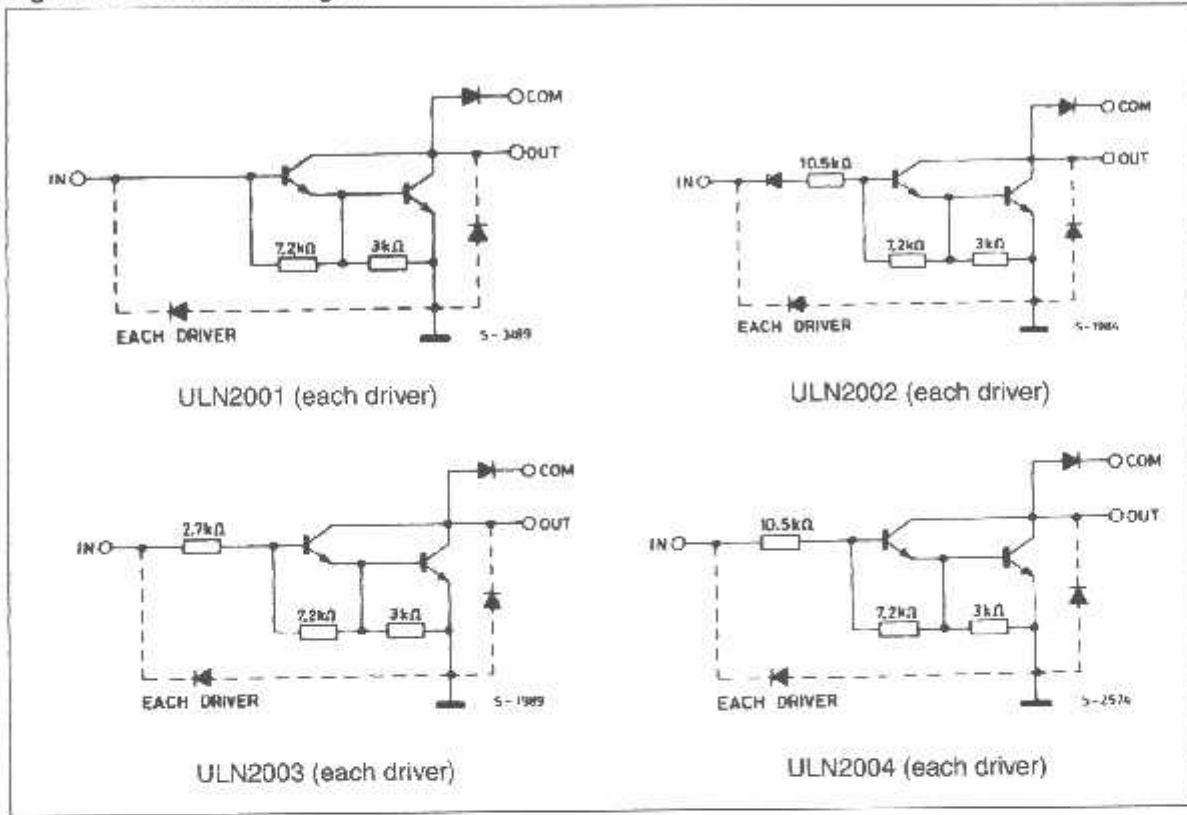
The ULN2001A/2002A/2003A and 2004A are supplied in 16 pin plastic DIP packages with a copper leadframe to reduce thermal resistance. They are available also in small outline package (SO-16) as ULN2001D1/2002D1/2003D1/2004D1.

Table 1. Device summary

Order code	
ULN2001A	ULN2001D1013TR
ULN2002A	ULN2002D1013TR
ULN2003A	ULN2003D1013TR
ULN2004A	ULN2004D1013TR

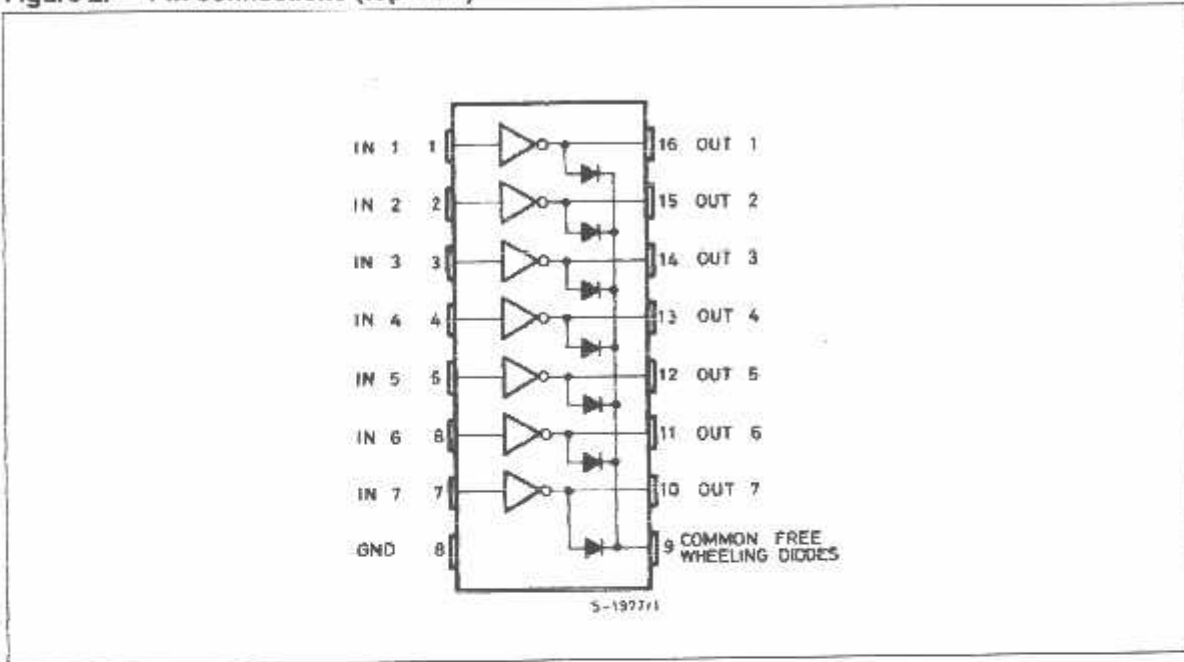
1 Diagram

Figure 1. Schematic diagram



2 Pin configuration

Figure 2. Pin connections (top view)



3 Maximum ratings

Table 2. Absolute maximum ratings

Symbol	Parameter	Value	Unit
V_O	Output voltage	50	V
V_I	Input voltage (for ULN2002A/D - 2003A/D - 2004A/D)	30	V
I_C	Continuous collector current	500	mA
I_B	Continuous base current	25	mA
T_A	Operating ambient temperature range	- 20 to 85	°C
T_{STG}	Storage temperature range	- 55 to 150	°C
T_J	Junction temperature	150	°C

Table 3. Thermal data

Symbol	Parameter	DIP-16	SO-16	Unit
R_{thJA}	Thermal resistance junction-ambient, Max.	70	120	°C/W

4 Electrical characteristics

Table 4. Electrical characteristics
($T_A = 25^\circ\text{C}$ unless otherwise specified).

Symbol	Parameter	Test condition	Min.	Typ.	Max.	Unit
I_{CEX}	Output leakage current	$V_{\text{CE}} = 50\text{ V}$, (Figure 3.)			50	μA
		$T_A = 70^\circ\text{C}$, $V_{\text{CE}} = 50\text{ V}$ (Figure 3.)			100	
		$T_A = 70^\circ\text{C}$ for ULN2002, $V_{\text{CE}} = 50\text{ V}$, $V_I = 6\text{ V}$ (Figure 4.)			500	
		$T_A = 70^\circ\text{C}$ for ULN2002, $V_{\text{CE}} = 50\text{ V}$, $V_I = 1\text{ V}$ (Figure 4.)			500	
$V_{\text{CE(SAT)}}$	Collector-emitter saturation voltage (Figure 5.)	$I_C = 100\text{ mA}$, $I_B = 250\text{ }\mu\text{A}$		0.9	1.1	V
		$I_C = 200\text{ mA}$, $I_B = 350\text{ }\mu\text{A}$		1.1	1.3	
		$I_C = 350\text{ mA}$, $I_B = 500\text{ }\mu\text{A}$		1.3	1.6	
$I_{\text{(ON)}}$	Input current (Figure 6.)	for ULN2002, $V_I = 17\text{ V}$		0.82	1.25	mA
		for ULN2003, $V_I = 3.85\text{ V}$		0.93	1.35	
		for ULN2004, $V_I = 5\text{ V}$		0.35	0.5	
		$V_I = 12\text{ V}$		1	1.45	
$I_{\text{(OFF)}}$	Input current (Figure 7.)	$T_A = 70^\circ\text{C}$, $I_C = 500\text{ }\mu\text{A}$	50	65		μA
$V_{\text{(ON)}}$	Input voltage (Figure 8.)	$V_{\text{CE}} = 2\text{ V}$, for ULN2002 $I_C = 300\text{ mA}$			13	V
		for ULN2003 $I_C = 200\text{ mA}$			2.4	
		$I_C = 250\text{ mA}$			2.7	
		$I_C = 300\text{ mA}$			3	
		for ULN2004 $I_C = 125\text{ mA}$			5	
		$I_C = 200\text{ mA}$			6	
		$I_C = 275\text{ mA}$			7	
		$I_C = 350\text{ mA}$			8	
h_{FE}	DC Forward current gain (Figure 5.)	for ULN2001, $V_{\text{CE}} = 2\text{ V}$, $I_C = 350\text{ mA}$	1000			
C_i	Input capacitance			15	25	pF
t_{PLH}	Turn-on delay time	$0.5 V_I$ to $0.5 V_O$		0.25	1	μs
t_{PHL}	Turn-off delay time	$0.5 V_I$ to $0.5 V_O$		0.25	1	μs
I_{R}	Clamp diode leakage current (Figure 9.)	$V_{\text{R}} = 50\text{ V}$			50	μA
		$T_A = 70^\circ\text{C}$, $V_{\text{R}} = 50\text{ V}$			100	
V_{F}	Clamp diode forward voltage (Figure 10.)	$I_{\text{F}} = 350\text{ mA}$		1.7	2	V

5 Test circuits

Figure 3. Output leakage current

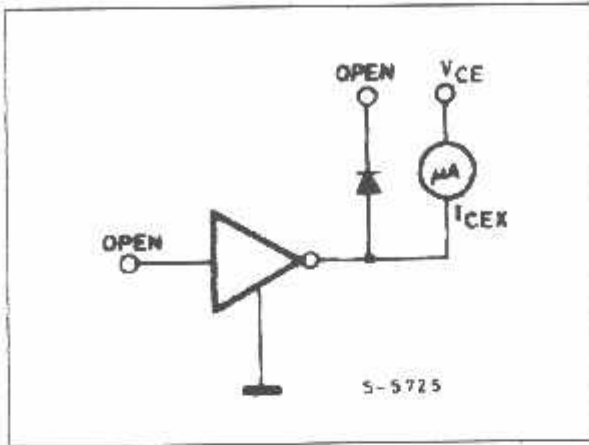


Figure 4. Output leakage current (for ULN2002 only)

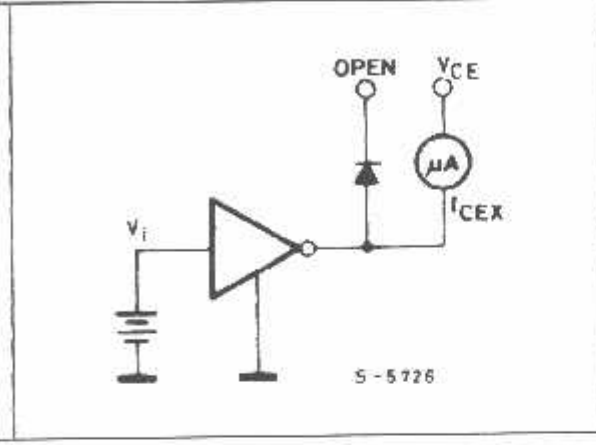


Figure 5. Collector-emitter saturation voltage Figure 6. Input current (ON)

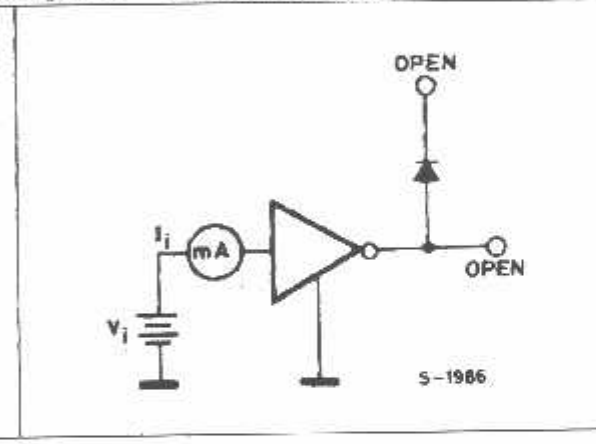
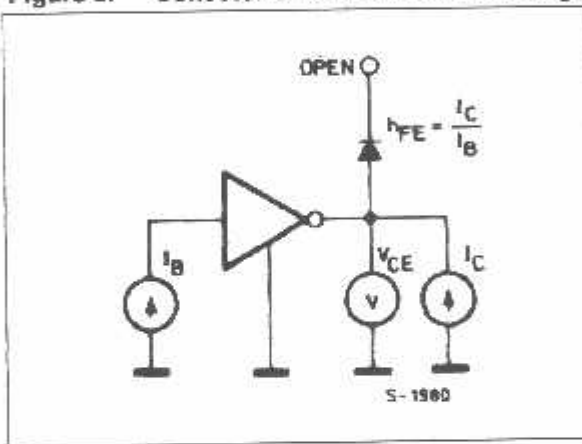


Figure 7. Input current (OFF)

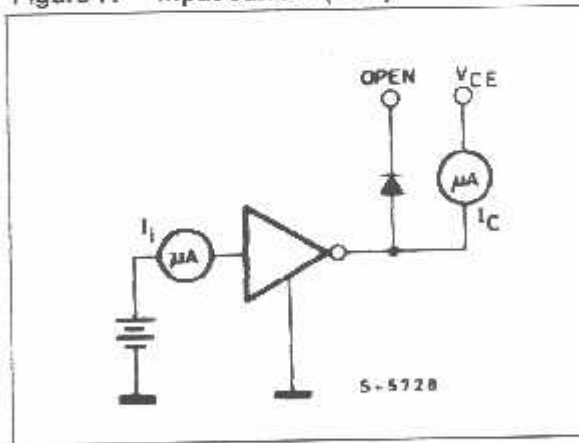


Figure 8. Input voltage

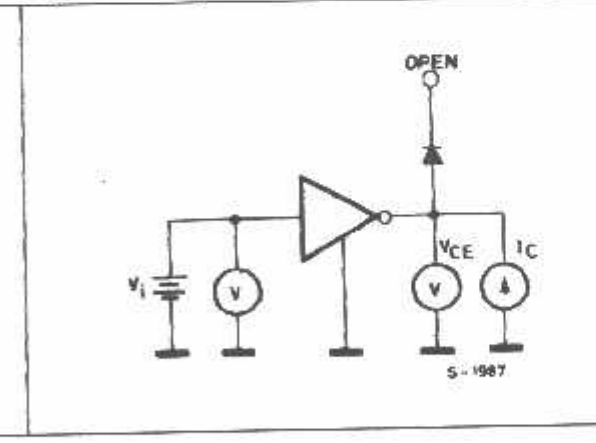


Figure 9. Clamp diode leakage current

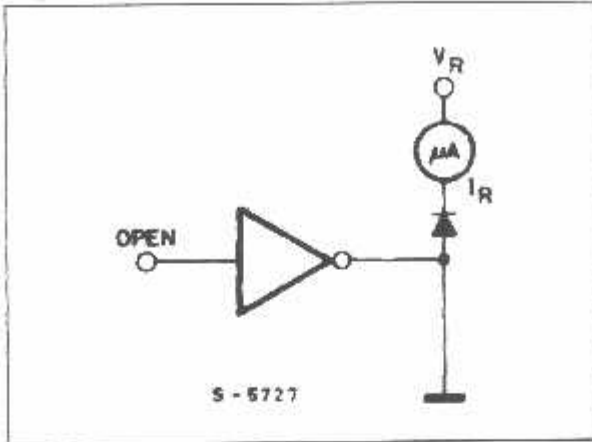


Figure 10. Clamp diode forward voltage

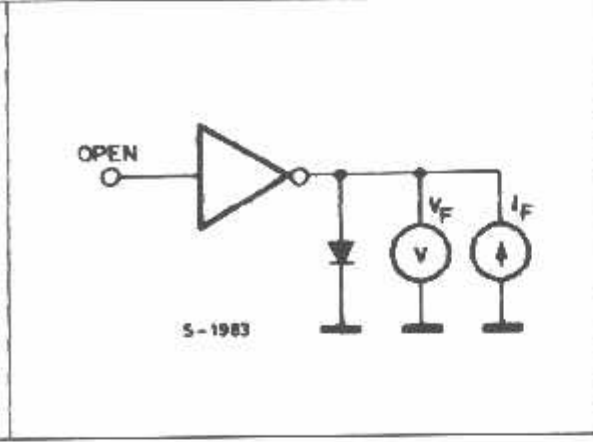


Figure 11. Collector current vs input current

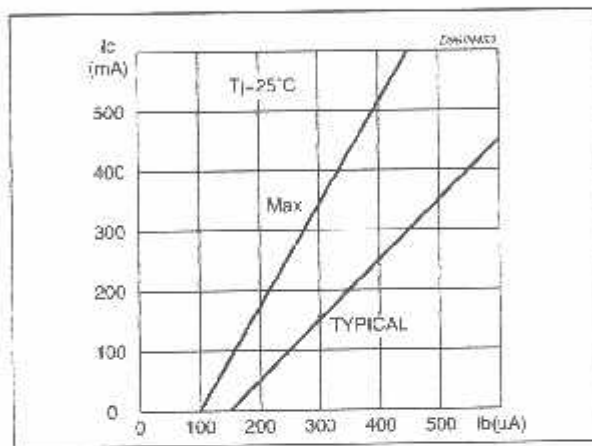


Figure 12. Collector current vs saturation voltage

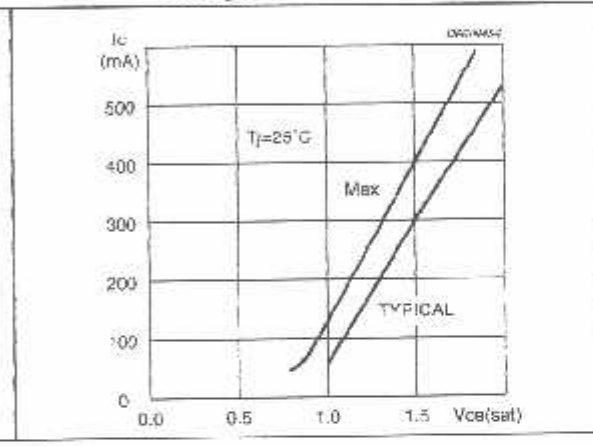


Figure 13. Peak collector current vs duty cycle

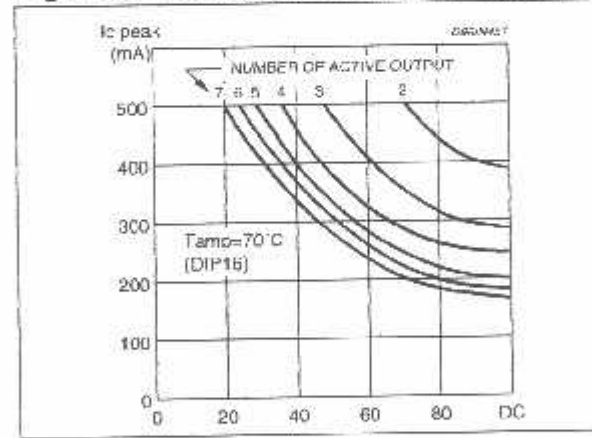


Figure 14. Peak collector current vs duty cycle

