

SKRIPSI

MERANCANG APLIKASI SISTEM PENGONTROLAN PERALATAN ELEKTRONIK VIA WEB



Disusun Oleh :
MIRA ORISA
05.12.588

PERPUSTAKAAN
ITN MALANG

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
SEPTEMBER 2009

LOGGING

ЧЕСТНОЕ ПОДПИСАНО
ЖЕЛАЮЩИЕ ПОТАКТИЧЕСКОГО ПОДПИСАНИЯ
KOD AV

• ДЕНЬ РОЖДЕНИЯ
АВИАКОМПАНИИ
800.27.50

Л-6 ОБРАЗЕЦ ПИСЬМА НАГРУЗКА
ЛЮБОДОМСКИЙ КОМПЛЕКС СЕВЕРНЫХ МАСТЕРСКАХ
БЕЛОРУССКОГО ГОСУДАРСТВА
СТАРИЙ ДОМ ОБРАЗОВАНИЯ
СООБЩЕСТВО ПРЕДПРИЯТИЙ

LEMBAR PERSETUJUAN

MERANCANG APLIKASI SISTEM PENGONTROLAN
PERALATAN ELEKTRONIK VIA WEB

SKRIPSI

*Disusun dan Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Komputer dan Informatika Strata Satu (S-1)*

Disusun Oleh :

MIRA ORISA

NIM : 05.12.588

Diperiksa dan Disetujui

Dosen Pembimbing I

Ir. F. Yudi Limpraptono, MT

NIP.Y 103 950 0274

Dosen Pembimbing II

Joseph Dedy Irawan MT

NIP. 132315178

Mengetahui,

Ketua Jurusan Teknik Elektro S-1



Irf. F. Yudi Limpraptono, MT

NIP.Y 103 950 0274

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2009



**INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNIK INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER & INFORMATIKA S-1**

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : Mira Orisa
NIM : 05.12.588
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika S-1
Judul Skripsi : Merancang Aplikasi Sistem Pengontrolan Peralatan Elektronik
Via Web

Dipertahankan dihadapan tim penguji skripsi jenjang Strata Satu (S-1) pada:

Hari : Senin
Tanggal : 7 September 2009
Dengan Nilai : 88,75 (A) *By*

PANITIA UJIAN SKRIPSI



Ir. H. Sidik Noertjahjono, MT.
NIP.Y. 1028700163

SEKRETARIS



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

ANGGOTA PENGUJI

PENGUJI I



Sandy Nataly Mantja, SKom.

PENGUJI II



(Ir.TH.Mimien Mustikawati,MT)
NIP.P.1030000352

**MERANCANG APLIKASI SISTEM PENGONTROLAN PERALATAN
ELEKTRONIK VIA WEB**

Mira Orisa

Jurusan Teknik Elektro S-1, Kosentrasi Teknik Komputer dan Informatika

Institut Teknologi Nasional Malang

Jl. Raya Karanglo Km.02 Malang, Indonesia

Email : mir4_orisa@yahoo.co.id

ABSTRAKSI

Di dalam perkembangannya, komputer banyak diterapkan dalam merancang system pengontrolan. Solusi untuk kerja apapun akan dapat dilakukan melalui media komputer termasuk dalam bidang pengontrolan alat-alat elektronik. Misalnya pada pengontrolan alat elektronik di suatu gedung bertingkat, pelaku tidak perlu mendatangi secara langsung ruangan untuk menyalakan atau mematikan alat-alat elektronik hanya dengan komputer sudah dapat mengontrol situasi dan kondisi alat-alat elektronik di ruangan tersebut. Sehingga sistem yang sudah terkomputerisasi akan mengontrol kondisi saklar menjadi lebih efektif dan efisien.

Pengontrolan peralatan elektronik via web merupakan suatu sistem kontrol sekaligus monitoring kondisi peralatan elektronik di ruangan melalui sebuah web. Client akan mendownload program yang ada di server tersebut sehingga client dapat melakukan pengontrolan apakah itu menhidupkan atau mematikan peralatan elektronik. Apabila client meminta halaman web dari sisi server dan alamat yang dituju itu benar maka server akan mengirimkan halaman web yang ada pada web server. jaringan yang digunakan adalah LAN (Lokal Area Network). PC-Client akan melakukan pemanggilan alamat lokal IP, tentu dibutuhkan sebuah web browser dan yang digunakan adalah internet explorer untuk memasukkan alamat tersebut..

Didalam halaman web itu akan ditampilkan informasi kondisi alat elektronik terkini, serta memungkinkan client untuk merubah kondisi alat elektronik (mengontrol saklar alat-alat elektronik), serta menampilkan informasi kondisi alat elektronik setelah pengontrolan.

Kata Kunci: *web,client, server,database,mikrokontroler AT89S51.*

KATA PENGANTAR

Dengan mengucap syukur kehadirat Allah SWT atas rahmat, hidayah dan berkat segala kemurahan-Nya telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul “Merancang Aplikasi Sistem Pengontrolan Peralatan Elektronik Via Web”.

Pembuatan skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata I di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis banyak mendapat bantuan baik moril maupun materiil, saran dan dorongan semangat dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Eng. Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang
2. Bapak Ir.H.Sidik Noertjahjono,MT selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang dan sekaligus selaku Dosen Pembimbing I.
4. Bapak Yoseph Dedy Irawan,ST,MT selaku Dosen Pembimbing II.
5. Kedua orang tua saya serta keluarga besar terutama Ibunda yang telah memberikan kepercayaan, do'a restu, dorongan serta semangat yang luar biasa.
6. Kakek dan Nenek saya yang telah membiayai studi saya selama ini.
7. Yusuf Sendy Jatmiko,ST terimakasih atas dukungan dan perhatiannya selama ini.
8. Ibu Kiptiah yang telah memberikan dukungan serta perhatiannya.

9. Rekan-rekan mahasiswa/i Teknik Komputer dan Informatika S-1 dan
T.Elektronika S-1.

10. Semua pihak yang telah membantu penulis dalam menyelesaikan
skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis telah berusaha semaksimal mungkin dan menyadari sepenuhnya
akan keterbatasan pengetahuan dalam menyelesaikan laporan ini.Untuk itu
Penulis mengharapkan kritik dan saran yang membangun dari pembaca demi
kesempurnaan laporan ini.

Harapan penulis semoga skripsi ini memberikan manfaat bagi
perkembangan ilmu pengetahuan dan pembaca.

Malang, September 2009

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	ii
BERITA ACARA	iii
ABSTRAKSI.....	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Tujuan.....	2
1.3. Rumusan Masalah	2
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian	3
1.5.1. Metode Pengumpulan Data	3
1.5.2. Metode Pengembangan Sistem	4
1.6. Sistematika Penulisan.....	6
BAB II DASAR TEORI.....	7
2.1. World Wide Web	7
2.2. Model Client Server	7
2.3. Dasar Arsitektur (TCP/IP).....	8
2.3.1. Nerwork Interface Layer	9
2.3.2. Internet Layer	10

2.3.3. Transport Layer.....	11
2.3.4. Application Layer	12
2.4. Borland Delphi 7	13
2.5. Kabel UTP	16
2.6. Web Server	18
2.7. Personal Home Page (PHP).....	20
2.8. Microsoft SQL.....	20
2.8.1. SQL (Structured Query Laanguage	20
2.8.2. MYSQL.....	21
2.9. ODBC(Open Database Connectivity)	25
2.10. Komunikasi Data.....	25
2.10.1. Port Komunikasi Serial	26
2.11. Saluran RS232.....	27
2.11.1. MAX232	28
2.12. Mikrokontroler AT89S51	29
2.12.1. Kontruksi AT89S51	30
2.12.2. Susunan Pin AT89S51	31
BAB III PERANCANGAN SISTEM.....	35
3.1. Umum.....	35
3.2. Arsitektur Sistem.....	36
3.2.1. Web Browser.....	36
3.2.1.1. Desain Antarmuka Aplikasi Pada Web Browser.....	36
3.2.1.1.1. Halaman Login.....	37
3.2.1.1.2. Halaman Info.....	37

3.2.1.1.3. Halaman Kontrol.....	38
3.2.1.1.4. Halaman View Status.....	38
3.2.2. Web Server.....	39
3.2.3. Borland Delphi API	42
3.2.4. Database	44
3.2.5. Perencanaan Perangkat Lunak Mikrokontroler	45
3.3. Perencanaan Perangkat Keras	46
3.3.1. Rangkaian Mikrokontroler AT89S41	46
3.3.2. Rangkaian IC MAX232	48
3.4. Perencanaan Dan Pembuatan Program Aplikasi.....	49
BAB IV IMPLEMENTASI DAN PENGUJIAN	53
4.1. Implementasi Sistem	53
4.2. Pengujian Koneksi LAN	53
4.3. Pengujian Web Server.....	54
4.4. Pengujian Database MYSQL	56
4.5. Pengujian Perangkat keras	59
4.5.1. Menguji Mikrokontroler dengan software	59
4.5.2. Pengujian MAX232	61
4.6. Pengujian Sistem	63
BAB V PENUTUP.....	68
5.1. Kesimpulan.....	68
5.2. Saran.....	68
DAFTAR PUSTAKA	69
LAMPIRAN	70

DAFTAR GAMBAR

2.1. Gambar Model World Wide Web	8
2.2. Gambar Kabel UTP	17
2.3. Gambar Port Serial D9	28
2.4. Gambar Sususunan Pin MAX232	29
2.5. Gambar Blok Diagram AT89S51.....	31
2.6. Gambar Susunan Pin AT89S51.....	32
3.1. Gambar Blok Diagram Sistem	35
3.2. Gambar Halaman Login.....	37
3.3. Gambar Halaman Sebelum Pengontrolan	37
3.4. Gambar Halaman Kontrol	38
3.5. Gambar Halaman View Status	38
3.6. Gambar Aphache Web Server	40
3.7. Gambar Delphi Programing	43
3.8. Gambar Flowchart Untuk Program Kontrol.....	44
3.9. Gambar Flowchart Untuk Program MCU	45
4.10. Gambar Rangkaian Osilator	46
3.11. Gambar Rangkaian MCU AT89S51	47
3.12. Gambar Rangkaian IC MAX232.....	48
3.13. Gambar Algoritma Tampilan Program	50
3.14. Gambar PHPMyadmin	51
4.1. Gambar Run Menu	54
4.2. Gambar Tampilan Pengujian Dengan Web Browser	55

4.3. Gambar Menu pada Database.....	57
4.4. Gambar Tabel Database	57
4.5. Gambar penulisan Perintah SQL.....	58
4.6. Gambar Hasil Perubahan Database	58
4.7. Gambar Led Aktif Tinggi.....	59
4.8. Gambar Rangkaian Pengujian RS232 to TTL.....	51
4.9. Gambar Rangkaian Pengujian TTL to RS232.....	52
4.10. Gambar Diagram Pengujian Sistem	64
4.11. Gambar Program Aplikasi Pada Komputer Kontrol	64
4.12. Gambar Tampilan Kondisi Sebelum Dikontrol	65
4.13. Gambar Form Update Kondisi Saklar.....	66
4.14. Gambar Kondisi Sesudah Dikontrol	66
4.15. Kondisi Alat	66

DAFTAR TABEL

2.1. Tabel Layer TCP/IP	9
2.2. Tabel Kategori Untuk Kabel UTP.....	17
2.3. Tabel Susunan Fungsi Pin-pin Konektor D25 dan D9	26
4.1. Tabel Hasil Pengujian RS232 to TTL	62
4.2. Tabel Hasil pengujian TTL to RS232	63
4.3. Tabel Pengujian Sistem Pengontrolan.....	67

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dalam era seperti saat ini, begitu banyak sektor kehidupan yang tidak terlepas dari peran serta dan penggunaan teknologi komputer, khususnya pada bidang-bidang dari lingkup pekerjaan. Semakin hari, kemajuan teknologi komputer, baik dibidang piranti lunak maupun perangkat keras berkembang dengan sangat pesat, disisi lain juga berkembang kearah yang sangat mudah dari segi pengaplikasian dan murah dalam biaya. Solusi untuk bidang kerja apapun akan ada cara untuk dapat dilakukan melalui media komputer termasuk dalam bidang pengontrolan alat-alat elektronik.

Pada saat ini banyak sekali kita jumpai gedung-gedung bertingkat yang menjulang tinggi dengan banyak ruangan didalamnya baik itu gedung perkantoran, gedung industri, gedung sekolah, dan tempat-tempat umum lainnya, dimana setiap ruangan pada setiap lantai terdapat banyak sekali alat-alat elektronik seperti lampu, ac dan sebagainya. Dalam upaya menghemat energi listrik maka kita harus meminimalisasi pemakaian alat-alat elektronik dari seluruh ruangan yang ada, seperti menggunakan alat-alat elektronik pada saat yang diperlukan saja.

Namun hal diatas akan menjadi sesuatu yang sulit jika seluruh aktivitas selalu dilakukan secara manual. Komputer menjadi jawaban yang utama dalam kasus-kasus seperti ini. Dengan adanya komputer bisa menjadikan pekerjaan yang membutuhkan ketelitian tinggi menjadi ringan. Pelaku tidak perlu mendatangi secara

langsung ruangan untuk menyalakan atau mematikan alat-alat elektronik hanya dengan komputer sudah dapat mengontrol kondisi saklar alat-alat elektronik di ruangan tersebut.

1.2. Tujuan

Tujuan dari Skripsi ini yaitu merancang aplikasi sistem pengontrolan peralatan elektronik Via Web agar mempermudah pengontrolan kondisi alat-alat elektronik pada suatu instansi sehingga akan dapat membantu penghematan energi listrik.

1.3. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan pada bagian sebelumnya, maka dapat dirumuskan beberapa masalah yang akan dibatasi yaitu:

1. Bagaimana mengkonfigurasi *server* agar dapat menerima dan mengolah *request* dari *client*, kemudian mengirimkan respon menuju *client* kembali.
2. Bagaimana merencanakan dan membuat suatu tampilan sederhana dan dinamis untuk *client* sehingga dapat mengendalikan peralatan kontrol yang dirancang .
3. Bagaimana membuat software kendali berbasis Delphi untuk menghubungkan sistem database dengan perangkat .

1.4. Batasan masalah

Dalam penulisan skripsi ini diperlukan batasan-batasan masalah tertentu agar nantinya tidak terjadi penyimpangan dari maksud dan tujuan penulisan skripsi ini. Skripsi ini dibatasi hanya pada hal-hal sebagai berikut:

1. Perencanaan sistem dan konfigurasi *server* agar dapat menerima dan mengolah request dari *client*, kemudian mengirimkan respon menuju *client* kembali.
2. Prinsip kerja *server operating system Windows Xp*.
3. Pembuatan software tampilan berbasis web yang dinamis untuk *client* sehingga dapat mengendalikan peralatan elektronik.
4. Pembuatan software kendali berbasis *Delphi* yang nantinya ditempatkan pada komputer server.
5. Tidak membahas secara rinci perangkat keras pengontrolan.
6. Tidak membahas secara rinci mengenai sistem jaringan yang digunakan.
7. Pengontrolan yang dilakukan pada saklar peralatan elektronik.

1.5. Metodologi Penelitian

1.5.1. Metode Pengumpulan data

Data yang digunakan dalam pengembangan sistem aplikasi ini haruslah berkualitas dan bermanfaat. Oleh karena itu dalam pengambilan data harus dengan

metode yang tepat. Dalam pengumpulan data penyusun menggunakan metode sebagai berikut :

1. Studi Pustaka / Literatur

Pengumpulan data ini dilakukan dengan cara mencari bahan-bahan kepustakaan sebagai landasan teori yang ada hubungannya dengan permasalahan yang dijadikan obyek penelitian. Studi literatur dilakukan untuk mempelajari teori-teori dasar yang berkaitan dengan perancangan sistem, meliputi:

- Studi tentang perangkat yang akan dibutuhkan dalam mengkonfigurasi *server*.
- Studi tentang protokol-protokol jaringan yang akan mendukung sistem, khususnya pada *server*.
- Studi tentang pembuatan program tampilan (*web browser programming*) untuk dapat diimplementasikan ke dalam sistem.

1.5.2. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan adalah sebagai berikut :

1. Analisi dan perencanaan sistem

Pada tahap ini dilakukan analisa kebutuhan sistem yang digunakan sebagai acuan untuk melakukan perancangan sistem aplikasi. Perancangan sistem aplikasi dilakukan setelah diketahui melalui analisa kebutuhan sistem sebelumnya.

2. Implementasi dan pengujian sistem

Implementasi dilakukan dengan menempatkan hasil perancangan ke dalam kode – kode program yang akan diintegrasikan menjadi suatu unit program yang dapat dijalankan. Pengujian sistem dilakukan untuk mengetahui apakah spesifikasi sistem sudah terpenuhi.

- Mengimplementasikan kebutuhan perangkat keras dan perangkat lunak bagi *server*.
- Membuat program tampilan pada *client* dengan bantuan diagram alir.
- Membuat blok-blok diagram dengan komponen yang sesuai.

3. Integrasi dan pengujian system

Integrasi sistem dilakukan untuk menggabungkan setiap unit program menjadi suatu sistem yang lengkap dan tahap selanjutnya adalah tahap pengujian yang bertujuan untuk mengetahui apakah sistem tersebut memenuhi persyaratan.

- Pengujian program tampilan pada *Client*.
- Pengujian dengan menggabungkan seluruh sistem.

1.6. Sistematika Penulisan

Untuk mempermudah dan memahami pembahasan skripsi ini, maka peneliti menyajikan secara sistematis sebagai berikut :

1. BAB I PENDAHULUAN

Membahas tentang hal-hal yang melatar belakangi pembuatan sistem, tujuan, perumusan masalah, batasan permasalahan, metodologi dan sistematika pada penulisan skripsi ini.

2. BAB II LANDASAN TEORI

Menguraikan teori-teori tentang sisi client dan sisi server yang mendasari dan mendukung dalam perencanaan dan implementasi sistem.

3. BAB III ANALISIS DAN DISAIN SISTEM

Membahas tentang perencanaan dan rekayasa perangkat lunak maupun perangkat keras yang akan dibutuhkan dalam sistem ini. Serta konfigurasi *Server*. Serta perangkat keras sistem control dan menerangkan cara-cara pengoperasian *Web browser*.

4. BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM

Membahas tentang pengujian sistem serta analisis dari hasil pengujian tersebut.

5. BAB V PENUTUP

Berisikan kesimpulan yang diperoleh dari hasil pengujian serta saran-saran yang dibutuhkan untuk kesempurnaan sistem.

BAB II

LANDASAN TEORI

2.1. World Wide Web

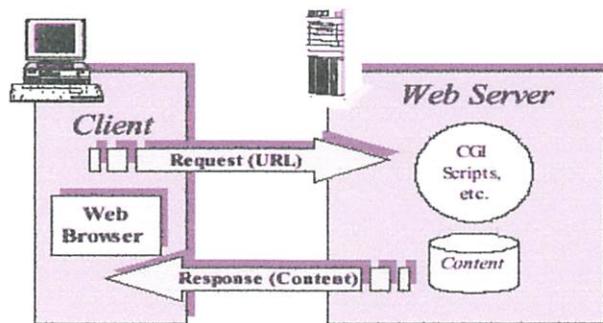
Model WWW yang digunakan pada *internet*, memungkinkan *client* untuk menerima *content* dari *web server*. Komunikasinya ditangani melalui protokol-protokol jaringan standar, seperti HTTP dan TCP/IP. Untuk mencapai *content* pada *server*, *client* menggunakan alamat-alamat dalam standar penamaan yang dikenal dengan *Uniform Resource Locator* (URL), prosesnya ditunjukkan pada gambar 2.1. Agar dapat menampilkan *content* dalam suatu format misalnya dalam *HyperText Markup Language* (HTML), bahasa-bahasa *script* (*JavaScript*) maupun format-format yang lainnya, *client* dapat menggunakan sebuah *web browser*.

2.2. Model Client Server

client-server merupakan sebuah paradigma dalam teknologi informasi yang merujuk kepada cara untuk mendistribusikan aplikasi ke dalam dua pihak: pihak klien dan pihak server.

Dalam model klien/server, sebuah aplikasi dibagi menjadi dua bagian yang terpisah, tapi masih merupakan sebuah kesatuan yakni komponen klien dan komponen server. Komponen klien juga sering disebut sebagai *front-end*, sementara komponen server disebut sebagai *back-end*. Komponen klien dari aplikasi tersebut dijalankan dalam sebuah workstation dan menerima masukan data dari pengguna.

Komponen klien tersebut akan menyiapkan data yang dimasukkan oleh pengguna dengan menggunakan teknologi pemrosesan tertentu dan mengirimkannya kepada komponen server yang dijalankan di atas mesin server, umumnya dalam bentuk *request* terhadap beberapa layanan yang dimiliki oleh server. Komponen server akan menerima request dari klien, dan langsung memprosesnya dan mengembalikan hasil pemrosesan tersebut kepada klien. Klien pun menerima informasi hasil pemrosesan data yang dilakukan server dan menampilkannya kepada pengguna, dengan menggunakan aplikasi yang berinteraksi dengan pengguna.



Gambar 2.1 Model World Wide Web
(2009 :04(www.wapforum.org))

2.3. Dasar Arsitektur TCP/IP

TCP/IP terdiri atas sekumpulan protokol yang masing-masing bertanggung jawab atas bagian-bagian tertentu dari komunikasi data. Pada TCP/IP tugas masing-masing protokol menjadi lebih jelas dan sederhana. Protokol yang satu tidak perlu mengetahui cara kerja protokol yang lain, sepanjang ia masih bisa saling mengirim

dan menerima data. Dengan prinsip ini, TCP/IP menjadi protokol komunikasi data yang fleksibel. Protokol TCP/IP dapat diterapkan dengan mudah di setiap jenis komputer dan *interface* jaringan, karena sebagian besar isi kumpulan protokol ini tidak spesifik terhadap satu komputer atau peralatan jaringan tertentu. Agar TCP/IP dapat berjalan di atas *interface* jaringan tertentu, maka hanya perlu dilakukan perubahan pada protokol yang berhubungan dengan *interface* jaringan saja.

Sekumpulan protokol TCP/IP dimodelkan dengan *layer-layer* sebagaimana terlihat pada gambar berikut:

Tabel 2.1 Layer TCP/IP

Layer	Protocols						
Application	HTTP	FTP (files)	SMTP (mail)	DNS (names)	NTP (time)	TELNET (login)	
Transport	TCP				UDP		
Internet	IP						
Network Interface	Ethernet	ISO 8802-2		X.25	SLIP	PPP	PLIP
		ISO 8802-3 IEEE 802.3	ISO 8802-5 IEEE 802.5 (token ring)	ISO 7776 X.25 LAPB (HDLC)			
	(CSMA/CD)			Serial		Parallel	
Physical	Fiber Optic	Coaxial	UTP	Satellite	Microwave		

(I Made Wiryana, Koneksi ke Internet : II-3)

2.3.1. Network Interface Layer

Protokol pada lapisan ini bertanggung jawab mengirim dan menerima data ke dan dari media fisik. Media fisiknya dapat berupa kabel, serat optik atau gelombang radio (Wi-fi dan Bluetooth).

Protokol yang akan diuraikan pada lapisan ini adalah PPP (*Point to Point Protocol*), yaitu sebuah protokol yang sering digunakan untuk menangani *dial - in* melalui penggunaan sebuah bluetooth.

2.3.2. Internet Layer

Protokol yang berada pada *layer* ini bertanggung jawab dalam proses pengiriman paket data ke alamat yang tepat. Pada *layer* ini terdapat tiga macam protokol yaitu IP, ARP dan ICMP.

1. IP (Internet Protocol)

IP merupakan inti dari protokol TCP/IP. Seluruh data yang berasal dari protokol pada layer di atas IP harus dilewatkan, diolah oleh protokol IP dan dipancarkan sebagai paket IP agar sampai ke tujuan..

2. ICMP (Internet Control Message Protocol)

ICMP merupakan sebuah protokol yang bertugas mengirimkan pesan-pesan kesalahan dan kondisi lain yang memerlukan perhatian khusus. Pesan / paket ICMP dikirim jika terjadi masalah pada lapisan IP dan lapisan di atasnya (TCP/UDP).

3. ARP (Address Resolution Protocol)

ARP bertugas untuk menerjemahkan alamat IP ke alamat *ethernet*. Pada jaringan lokal, paket IP biasanya dikirim melalui *card ethernet*. Untuk dapat berkomunikasi atau mengenali *ethernet* yang lain digunakan suatu alamat *ethernet*. Dalam hal ini setiap *card ethernet* memiliki alamat *ethernet* yang

berbeda-beda. Pada saat hendak mengirimkan data ke komputer dengan IP tertentu, suatu *host* pada jaringan *ethernet* perlu mengetahui, di atas alamat *ethernet* yang manakah tempat IP tersebut terletak. Untuk keperluan pemetaan alamat IP dengan alamat *ethernet* ini digunakanlah protokol ARP.

2.3.3. Transport Layer

Pada *transport layer* berisi protokol yang bertanggung jawab untuk mengadakan komunikasi antara dua *host* / komputer. Protokol tersebut adalah TCP (*Transport Control Protocol*) dan UDP (*User Datagram Protocol*).

TCP (*Transport Control Protocol*) menyediakan layanan yang dikenal dengan *connection oriented, reliable* dan *byte stream service*. *Connection oriented* berarti sebelum melakukan pertukaran data, dua aplikasi pengguna TCP harus melakukan pembentukan hubungan (*handshaking*) terlebih dahulu. *Reliable* berarti TCP menerapkan proses deteksi kesalahan paket dan retransmisi. Sedangkan *byte stream service* memiliki arti bahwa paket dikirimkan dan sampai ke tujuan secara berurutan.

UDP (*User Datagram Protocol*) merupakan protokol transport yang sederhana. Berbeda dengan TCP yang *connection oriented*. UDP bersifat *connectionless*. Dalam UDP tidak ada pengurutan kembali data yang datang, *acknowledgement* terhadap paket yang datang, atau retransmisi jika paket mengalami masalah di tengah jalan.

Kemiripan UDP dengan TCP ada pada penggunaan *port number*. Sebagaimana digunakan pada TCP, UDP menggunakan *port number* ini untuk

membedakan pengiriman *datagram* ke beberapa aplikasi berbeda yang terletak pada komputer yang sama.

2.3.4. Application Layer

Pada *layer* inilah terletak semua aplikasi yang menggunakan protokol TCP/IP. Salah satunya yang akan diuraikan adalah *Hypertext Transfer Protocol* (HTTP).

Protokol transfer utama yang digunakan oleh *World Wide Web* adalah HTTP, yaitu protokol aplikasi berbasis *client server* sederhana yang dibangun di atas TCP. Transaksi yang khas dari HTTP ini yaitu :

1. *Client* melakukan koneksi dengan *server* HTTP dengan mengirim *request* terhadap *content* yang diinginkan dan menunggu respon dari *server*.
2. Setelah menerima *request* dari *client*.
3. *Server* akan memproses *request* tersebut, mengirimkan respon dan menutup koneksi.

Respon dapat berupa *content* yang diinginkan atau dapat juga berupa hasil dari eksekusi dalam kasus *content* yang diminta adalah *content* yang dapat dieksekusi. *Client* mengalamatkan *content* yang dibutuhkan dengan URL yaitu alamat fisik dari objek yang dapat diambil dengan menggunakan protokol jaringan seperti HTTP. URL terdiri atas nama *server* (DNS atau alamat IP) dan nama dari *content* yang diminta. Nama *server* dan nama *content* dipisah dengan tanda “/”. Sebagai

contoh, di bawah ini menunjukkan *request* HTTP yang berisi URL dari *content* yang diinginkan.

http:// 10.10.10.1/trd/index.php

10.10.10.1 adalah IP dari *server*, dan *skripsi/index.php* adalah nama dari *content*. Ada beberapa operasi HTTP, sebagai contoh adalah GET dan POST. Operasi GET digunakan untuk meminta *content* dari *server*. Sedangkan operasi POST digunakan untuk mengirim informasi ke *content* yang ada di *server*. Selanjutnya *server* akan merespon dengan *header* yang diikuti oleh bagian data.

2.4. Borland Delphi 7

Borland Delphi merupakan bahasa pemrograman yang bekerja dalam lingkup *MS Windows*, seperti *MS Visual Basic*. Borland Delphi dapat memanfaatkan fasilitas Windows secara maksimal serta memiliki kemampuan untuk merancang program aplikasi seperti tampilan program aplikasi lainnya.

Kemampuan secara umum, yaitu menyediakan komponen-komponen yang memungkinkan kita membuat program aplikasi yang sesuai dengan tampilan dan cara kerja *MS Windows*, serta didukung dengan pemrograman terstruktur yang sangat andal, yaitu pemrograman Objek pascal.

Delphi merupakan aplikasi database yang berorientasi objek dengan kemampuannya untuk mengakses *SQL* yang merupakan bahasa query.

IDE (Integrated Development Environment) merupakan bagian dari Delphi yang digunakan untuk menciptakan aplikasi. Melalui IDE (*Integrated Development Environment*) inilah secara visual kita bisa merancang tampilan untuk client dan menuliskan *script*.

ADO (ActiveX Data Objects) adalah antarmuka tingkat tinggi dari Microsoft untuk pengaksesan database. *ADO (ActiveX Data Objects)* diimplementasikan oleh teknologi *OLE DB* dan dapat mengakses database relational dan nonrelasional seperti email, sistem-sistem file, dan objek-objek bisnis. *ADO (ActiveX Data Objects)* adalah mesin database yang tidak tergantung pada server dan mendukung *server lokal* dan *server SQL*. *ADO (ActiveX Data Objects)* adalah mesin yang berat dengan konfigurasi yang disederhanakan (karena tidak tersentralisasi). Kita tidak perlu menginstalasikannya karena merupakan bagian dari windows.

ADO (ActiveX Data Objects) memberikan beberapa keuntungan untuk pemakaian *server SQL* karena driver Microsoft untuk databasenya sendiri mempunyai kualitas lebih baik dari provider *OLE DB* pada umumnya. ADO hanya dapat dipakai pada platfrom Windows.

ADO (ActiveX Data Objects) memiliki beberapa komponen yang merupakan turunan dari *TDataset*. Kumpulan *ADO* tersebut sering disebut dengan *ADOExpress* yang mempunyai beberapa komponen terletak dihalaman *ADO* pada Component Palette.

- Komponen untuk koneksi

Agar *ADO* dapat berjalan dan saling berbagi koneksi dalam mencari data, maka dilakukan dengan menghubungkan beberapa komponen *ADODataset* dan perintah *ADO* ke dalam komponen *TADOConnection*.

- Komponen untuk akses *ADO*

Dalam melakukan proses untuk mengakses database, komponen untuk mengakses *ADO* terdiri atas *TADODataset* dan *TADOCommand* yang dioperasikan pada data *ADO* menggunakan komponen utamanya, yaitu *TADODATASET*.

TADODataset mempunyai fungsi-fungsi seperti *Ttable*, *Tquery* dan *TstoredProc*. Dengan mempunyai fasilitas ini, komponen ini mempunyai kemampuan untuk mengakses data terutama dalam manipulasi tabel-tabel dan menjalankan query SQL dan Stored Procedure, bahkan mampu berinteraksi langsung dengan sebuah data store atau koneksi melalui *TADOConnection*.

Untuk *TADOCommand* ini mempunyai fungsi menjalankan perintah-perintah SQL, agar dapat dipakai untuk menjalankan SQL yang menghasilkan record, gunakan perintah seperti berikut:

ADODataset.Recordset:=ADOCommand.Execete;

- Komponen *ADOTable*

Komponen kompatibilitas terdiri atas *TADOTable*, *Tquery* dan *TstoredProc*.

Pada prinsipnya Delphi dirancang untuk memudahkan para user dalam merancang sebuah aplikasinya. Fasilitas seperti komponen *ADOTable* merupakan komponen yang dirancang untuk mendukung dalam pembuatan aplikasi.

- Komponen *ADOConnection*

Dalam pemakaian beberapa dataset untuk menyatakan koneksi, maka dataset akan membuat objek connection yang mengakibatkan pemborosan resource dan akan timbul masalah lisensi pengaksesan karena dengan beberapa data set akan ada beberapa koneksi. Sehingga kita bisa memanfaatkan fasilitas dari *ADO* yang lebih memudahkan dalam berkoneksi.

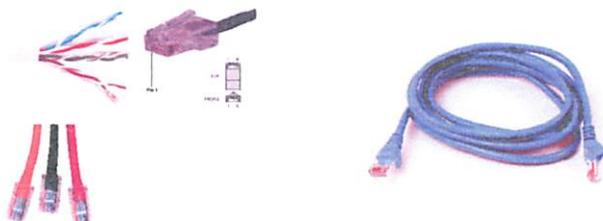
- Komponen *ADODataset*

Komponen selanjutnya yang sangat penting dalam mencari dan mengubah data dari beberapa tabel yang digunakan adalah komponen *ADODataset*.

2.5. Kabel UTP(*Unshielded Twisted Pair*)

Kabel “Unshielded twisted pair” (UTP) digunakan untuk LAN dan sistem telepon. Kabel UTP terdiri dari empat pasang warna konduktor tembaga yang setiap pasangnya berpilin. Pembungkus kabel memproteksi dan menyediakan jalur bagi tiap pasang kawat. Kabel UTP terhubung ke perangkat melalui konektor modular 8 pin

yang disebut konektor RJ-45. Semua protokol LAN dapat beroperasi melalui kabel UTP. Kebanyakan perangkat LAN dilengkapi dengan RJ-45.



Gambar 2.3.Kabel UTP

(*Kabel Local Area Network (LAN).htm*)

Kabel UTP dikelompokkan menjadi beberapa kategori, mulai kategori 1 sampai 5, masing-masing dengan karakteristik tertentu. Secara singkat kategori-kategori tersebut adalah sebagai berikut:

Tabel 2.2.Kategori Untuk Kabel UTP

Tipe	Penggunaan
Kategori 1	Hanya untuk suara, biasanya untuk kabel telepon
Kategori 2	Maksimum laju data sampai 4 Mbps (Local Talk)
Kategori 3	Maksimum laju data sampai 10 Mbps (Ethernet)
Kategori 4	Maksimum laju data sampai 20 Mbps (16 Mbps Token Ring)
Kategori 5	Maksimum laju data sampai 100 Mbps (fast Ethernet)

Kategori 5e	Maksimum laju data sampai 100 Mbps (perbaikan kategori 5)
Kategori 6	Maksimum laju data sampai 250 Mbps
Kategori 7	Maksimum laju data sampai 600 Mbps

(Jaringan Komputer teori dan implementasi berbasis linux/hal:42,wagito:2005)

Kabel UTP ini terkoneksi dengan komputer melalui konektor RJ-45. Pada komputer dipasang pada port yang ada di LAN Card computer masing-masing.

Keuntungan yang dapat diperoleh dalam menggunakan kabel UTP antara lain:

- Materialnya sangat banyak dan variasi instalasi kabel yang luas tidak asing dengan peralatan instalansi
- Dapat menggantikan kabel telepon jika diperlukan kualitas yang cukup tinggi

Merupakan kabel dengan harga paling murah dan mudah dikerjakan

2.6. Web Server

Web server atau lebih tepatnya *World Wide Web server* merupakan sebuah *server internet* yang mampu melayani koneksi transfer data dalam protokol HTTP. Agar dapat diakses oleh *Web browser*, *content* (data-data) pada *web server* dibuat dengan menggunakan bahasa pemrograman web. Program untuk *content* tersebut bersifat dinamis dan berhubungan erat dengan sebuah *database*.

Pada sistem ini, perangkat lunak yang digunakan sebagai *web server* adalah *Apache web server*. Pertimbangan yang diambil dalam memilih *web server* tersebut dikarenakan oleh performasi, kecepatan serta distribusinya yang gratis.

Dalam sistem operasi *Windows XP*, setelah proses instalasi *Apache web server* akan terdapat banyak file dan direktori, diantaranya adalah /Xampp/apache/conf yang merupakan tempat file-file konfigurasi yaitu httpd.conf, srm.conf, mime.types dan access.conf. File-file tersebut digunakan untuk pemeliharaan konfigurasi *server*.

File httpd.conf merupakan file konfigurasi utama dari *Apache server* yang berisikan petunjuk konfigurasi yang mengontrol *server*, lokasi *logfile*, *user ID* dan *listening port*.

File srm.conf menyimpan informasi tentang lokasi dokumen *web*, lokasi dari program CGI, dan prioritas bahasa. File access.conf berisi tipe akses, ketersediaan dokumen, dan setting keamanan. Sedangkan file mime.types (*Multipurpose Internet Mail Extension*) merupakan sebuah sistem *e-mail* yang memungkinkan penyertaan file-file *binary* seperti *image*, *program executable*, file suara dan termasuk pula untuk dokumen-dokumen HTML.

Dari keempat file tersebut di atas, dalam implementasinya hanya httpd.conf yang digunakan, karena file tersebut telah terkonfigurasi dan dapat diaplikasikan secara langsung. Sedangkan isi dari file srm.conf, mime.types dan access.conf telah menjadi satu di dalam file httpd.conf tersebut.

2.7. Personal Home Page(PHP)

PHP merupakan modul tambahan *server Apache* dan dianggap satu bagian integralnya, sehingga pembuatan aplikasi *web* dengan menggunakan PHP akan dapat memberikan hasil yang jauh lebih baik dalam hal kecepatan dan efisiensi server. Terdapat dua pilihan aplikasi PHP yaitu PHP4 dan PHP3. Dalam pembuatan sistem ini yang digunakan adalah aplikasi PHP4 karena merupakan aplikasi yang terbaru dan telah mengalami banyak perkembangan dari versi sebelumnya.

2.8. Microsoft SQL

2.8.1. SQL (Structured Query Language)

SQL adalah standart bahasa pemrograman query database yang dipakai untuk menyelesaikan segala permasalahan database. SQL telah dipakai sejak tahun 80-an dan tahun 90-an. Pada saat ini SQL menjadi standart untuk bekerja dengan database client/server. SQL mempunyai fungsi sebagai berikut :

- Data Definition. Anda dapat mendefinisikan struktur tabel, tipe data dari setiap field didalam tabel dan hubungan referensial dari field-field tertentu ke field-field pada tabel yang lain.
- Data Retrieval. Aplikasi client memakai SQL untuk meminta data dari server. SQL juga mengijinkan client mendefinisikan data yang akan dibaca dan bagaimana data itu di-retrieve (diperoleh informasinya) misalnya urutannya dan field-field apa yang di-retrieve.

- Data Integrity. Anda dapat menjaga integritas data dengan berbagai konstantin integritas sebagai bagian dari tabel atau terpisah dari tabel.
- Data Processing. SQL mengijinkan client mengubah, menambah, menghapus data dari server.
- Security. Anda dapat memproteksi data dengan mendefinisikan hak akses, view dan batasan pengaksesan pada objek-objek database.
- Concurrent access. SQL mengatur pengaksesan secara simultan sehingga satu pengaksesan tidak mengganggu pengaksesan yang lain.

SQL dapat diakses dari delphi dan delphi dapat memakai perintah SQL untuk memanipulasi tabel. Untuk memakai bahasa SQL dari delphi anda memerlukan komponen query. Komponen query dapat anda pada tab data access dari VCL. Hal penting pada komponen query adalah property SQL, yaitu tempat anda memasukkan pernyataan SQL yang akan siap dijalankan.

2.8.2. MySQL

MySQL merupakan terobosan solusi yang tepat dalam aplikasi database. Sifatnya yang open source serta dukungan oleh ribuan bahkan jutaan komunitas pengguna di internet, menjadikan MySQL sebagai software database yang cukup banyak digunakan. Tak hanya itu, kemampuannya yang bisa digunakan pada berbagai sistem operasi juga menjadikan MySQL sebagai software database pilihan.

Pada dasarnya, keandalan suatu sistem database (DBMS) dapat diketahui dengan melihat cara kerja optimizer-nya dalam melakukan proses perintah-perintah

SQL. Baik itu yang dibuat oleh user, maupun oleh program-program aplikasinya. MySQL merupakan database server yang dapat dikatakan lebih unggul dibandingkan database server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dari Interbase.

Dengan kemampuan inilah database MySQL mampu berinteraksi dengan database desktop seperti Delphi 7, terutama dalam hal melakukan koneksi data.

Sebagai software database dengan konsep database modern, MySQL memiliki banyak kelebihan, antara lain:

- **Probability**

MySQL dapat digunakan dengan stabil tanpa kendala berarti pada berbagai sistem operasi seperti: Windows, Linux, FreeBSD, Mac OS X Server, Solaris, Amiga, Hp-UX dana masih banyak lagi.

- **Open Source**

Sebagaimana telah diuraikan sebelumnya, MySQL didistribusikan secara open source dibawah lisensi GPL, sehingga kita dapat memperoleh dan menggunakannya secara Cuma-Cuma tanpa dipungut biaya.

- **Multiuser**

MySQL dapat digunakan untuk mengatasi beberapa user dalam waktu yang bersamaan tanpa mengalami konflik. Hal ini akan memungkinkan sebuah

database server MySQL dapat diakses client secara bersamaan dalam waktu yang bersamaan pula.

- **Performance Tuning**

Dalam hal performansi, MySQL memiliki kecepatan yang cukup menakjubkan dalam menangani query-query sederhana, serta mampu memproses lebih banyak SQL per satuan waktu.

- **Column Types**

MySQL mendukung tipe kolom(tipe data) yang sangat komplek, seperti signed/unsigned integer, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year, set serta enum.

- **Command dan Functions**

MySQL mempunyai operator dan fungsi secara penuh yang mendukung perintah SELECT dan WHERE dalam query.

- **Security**

Sekuritas sistem merupakan hal yang sangat diperhatikan pada MySQL. Terbukti adanya beberapa lapisan sekuritas seperti level subtemask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

- **Scalability dan limits**

Dalam hal batas kemampuan, MySQL terbukti mampu menangani database dalam skala yang cukup besar, dengan jumlah record lebih dari 50 juta dan 60

ribu tabel serta 5 miliar baris. Selain itu, batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.

- **Connectivity**

MySQL mampu melakukan koneksi dengan client menggunakan protokol TCP/IP, Unix soket (Unix), Named Pipes(NT).

- **Localisation**

Localisation merupakan salah satu kemampuan MySQL dalam mendekripsi pesan kesalahan (error code) pada client dengan menggunakan lebih dari dua puluh bahasa . Dengan demikian, pengguna cukup menyesuaikan bahasa yang digunakan.

- **Interface**

Sama halnya dengan software-software database lainnya MySQL memiliki interface (antarmuka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (Application Programming Interface)

- **Client dan tools**

MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database, dan pada setiap tool yang ada disertakan petunjuk online.

- Struktur tabel

Struktur tabel MySQL cukup baik, serta cukup fleksibel misalnya ketika menangani ALTER TABLE dibandingkan database lainnya semacam PostgreSQL ataupun Oracle.

2.9. ODBC(Open Database Connectivity)

ODBC adalah software dalam paket windows yang berfungsi untuk menyatukan aplikasi-aplikasi database menjadi satu. Di dunia komputer saat ini banyak sekali aplikasi-aplikasi penyusun database, seperti MYSQL, dBase, FoxPro, Ms Acces dan sebagainya. Dengan ODBC kita dapat mengambil data ke dalam beberapa format aplikasi database tersebut secara bersama-sama.

2.10. Komunikasi Data

Mikroprosesor dalam komputer bekerja atas dasar prinsip data paralel, mulai mula banyak dipakai microprocessor dengan data paralel 8 bit dan kini sudah dipakai data paralel 32 bit, tapi dalam hal komunikasi data yang dipakai tetap adalah teknik pengiriman data secara seri.

Alasan utama pemakaian teknik pengiriman seri karena saluran komunikasi data paralel yang panjang harganya sangat mahal dan tidak praktis. Sinyal digital dari IC TTL (Transistor Transistor Logic) mungkin hanya bisa dikirimkan sejauh 1 sampai 2 meter tergantung kecepatan pengiriman data. Agar bisa dikirimkan lebih jauh, sinyal TTL tadi harus di-buffer dengan rangkaian buffer khusus, biaya pengolahan sinyal ini cukup mahal. Untuk mengirim data secara paralel diperlukan satu saluran untuk setiap bit data, dengan demikian pengiriman data paralel 8 bit,

membutuhkan biaya 8 kali lebih mahal dibandingkan dengan komunikasi data secara seri yang hanya perlu satu saluran saja.

2.10.1. Port Komunikasi Serial

Port komunikasi serial pada komputer hadir dalam dua tipe konektor yaitu konektor D25 dan konektor D9. Kedua-duanya adalah konektor male, sehingga kita membutuhkan konektor female untuk memanfaatkan fasilitas port serial ini. Di bawah ini susunan pin-pin pada konektor D25 dan konektor D9.

Tabel 2.3. Susunan dan fungsi pin-pin konektor D25 dan konektor D9

D25	D9	Fungsi	Keterangan
Pin 2	Pin 3	TD (Transmit Data)	Output data serial (TX)
Pin 3	Pin 2	RD (Receive Data)	Input Data Serial (RX)
Pin 4	Pin 7	RTS (Request To Send)	Menginformasikan pada modem bahwa UART siap menukar data
Pin 5	Pin 8	CTS (Clear To Send)	Mengindikasikan bahwa modem siap untuk menukar data
Pin 6	Pin 6	DSR (Data Set Ready)	Menginformasikan pada UART bahwa modem siap mengadakan hubungan
Pin 7	Pin 5	SG (Signal Ground)	Saluran Ground (Tanah)
Pin 8	Pin 1	CD (Carrier Detect)	Mendeteksi sinyal carrier dari modem lain
Pin	Pin 4	DTR (Data Terminal)	Menginformasikan pada modem

20		Ready)	bahwa UART siap mengadakan hubungan
Pin 22	Pin 9	RI (Ring Indicator)	Aktif jika modem menerima sinyal ring dari PSTN

(Interfacing the Serial / RS232 Port V5.0, 1998: hal 5)

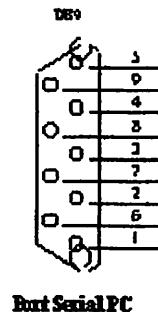
2.11. Saluran RS232

Ditinjau dari standart elektronik, dewasa ini dikenal dua macam saluran data, yang pertama adalah transmisi data tunggal (single-ended/unbalanced data transmission) yang dipakai RS232 yakni saluran yang memenuhi ketentuan standard EIA RS232, saluran data yang kedua adalah saluran ganda (differential-balanced data transmission) yang dipakai RS485 yaitu saluran yang memenuhi ketentuan standard EIA RS422/485 atau saluran arus 20 mA (20 mA current loop).

Transmisi saluran tunggal memakai satu utas kabel untuk mengirimkan satu sinyal. Informasi logika ditafsirkan dari beda tegangan terhadap ground. Saluran RS232 bersifat Full Duplex artinya dapat mengirimkan dan mengirimkan dan menerima data secara bersamaan.

Saluran RS232 banyak dipakai untuk menghubungkan komputer dengan alat pendukungnya, misalnya komputer dengan printer (sekarang hampir tidak ada printer seri), komputer dengan modem, dengan syarat hubungan antara alat yang tidak lebih jauh dari 50 feet (sekitar 15 meter). Komputer sebagai DTE (Data Terminal Equipment) dan alat pendukung yang lain sebagai DCE (Data Communications

Equipment). Panjang/jarak maksimum pengiriman data pada saluran RS232 adalah 50 feet dengan kecepatan 19,2 Kbps. Logika ‘1’ ditandai dengan range tegangan –5 V Min dan –15 V Max. Sedangkan logika ‘0’ ditandai pada range tegangan 5 V Min dan 15 V Max.



Gambar 2.5 Port serial D9.

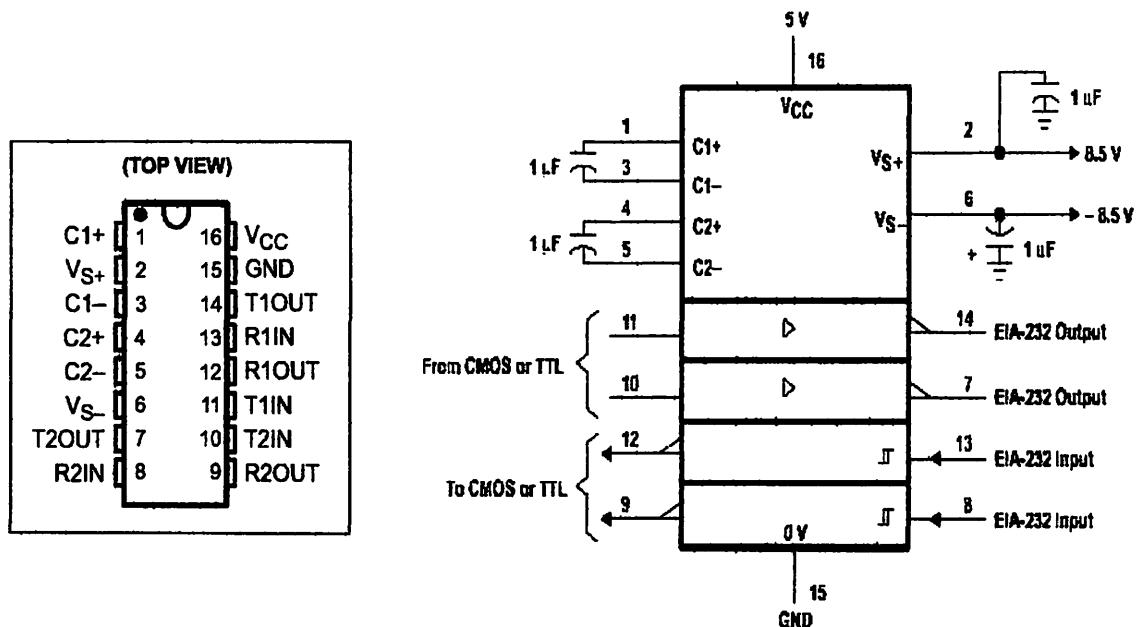
(*Panduan Praktis Teknik Antarmuka dan Pemrograman Mikrokontroler : 90*)

2.11.1. MAX232(RS232 Lever Converter)

Dalam ketentuan RS232, level logika ‘1’ dinyatakan dengan tegangan antara –3 sampai –25 Volt, dan level logika ‘0’ dinyatakan dengan tegangan antara +3 sampai +25 Volt. Mengingat Hampir semua perangkat digital menggunakan level tegangan logika TTL atau CMOS sumber tegangan +5 Volt, dan level logika ‘0’ dinyatakan dengan tegangan antara 0,8 sampai 0 Volt dan level logika ‘1’ dinyatakan dengan tegangan 3,5 sampai 5 Volt, maka antara rangkaian digital dan saluran RS232 biasanya disisipkan IC RS232-TTL Level Converter MAX232.

Pada IC MAX232 sudah termasuk didalamnya receive dan transmit data, yang waktu dulu IC receive data dan transmit data ini sendiri-sendiri. Bahkan pada

MAX232 ini ada 2 receive/transmit data. Berikut ini gambar susunan kaki serta rangkaian komunikasi pada MAX232 :



Gambar 2.6 Susunan Pin dan Rangkaian MAX232

(*Data Sheet MAX232 , Texas Instruments, 1998*)

2.12. Mikrokontroler AT89S51

Mikrokontroler adalah otak dari segala piranti elektronik yang aplikatif disekitar kita. Ada 2 mainstream dari family mikrokontroller didunia elektronika. Pertama dari keluarga Motorola (68HCXX) dan yang lain keluarga Intel (8XCXX). Intel mengenalkan IC Mikrokontroler type 8051 pada awal tahun 1980-an, 8051 termasuk sederhana dan harganya murah sehingga banyak digemari, banyak pabrik IC besar lain yang ikut memproduksnya, seperti pabrik IC Atmel ikut menambah

anggota keluarga MCS51. Atmel merupakan pabrik IC yang sangat menguasai teknologi pembuatan Flash PEROM (*Programable Eraseable Read Only Memory*).

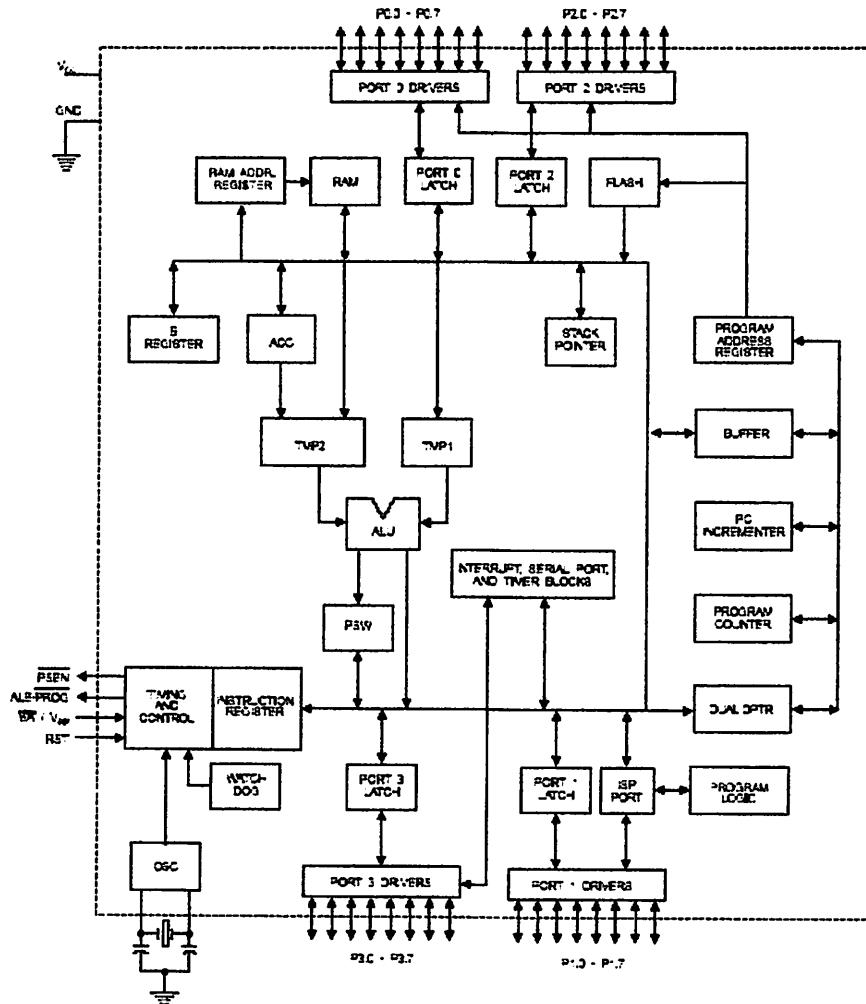
2.12.1. Kontruksi AT89S51

AT89S51 menyediakan beberapa fasilitas sebagai berikut :

- 8 bit CPU (Central Proccessing Unit) dengan register A dan B
- 16 bit PC (Program Counter) dan DPTR (Data Pointer)
- 8 bit PSW (Program Status Word)
- 8 bit SP (Stack Pointer)
- 4K Bytes flash PEROM (Programable Eraseable Read Only Memory)
dengan daya tahan 1000 kali penulisan dan penghapusan.
- 128 bytes internal RAM (Random Acces memory), terdiri dari :
 - 4 register bank masing masing berisi 8 register
 - 16 byte memori yang dapat dialamati pada bit level
 - 80 byte memori general purpose
- 32 pin/jalur I/O (Input/Output) tersusun sebagai 4 port masing-masing 8 bit
- 2 Timer/Counter 16-Bit yaitu T0 dan T1
- Kestabilan kerja yang tinggi : 0 Hz – 24 Hz
- 6 sumber intrupsi
- Komunikasi serial full duplex : SBUF

- Membutuhkan catu daya rendah

Blok Diagram AT89S51

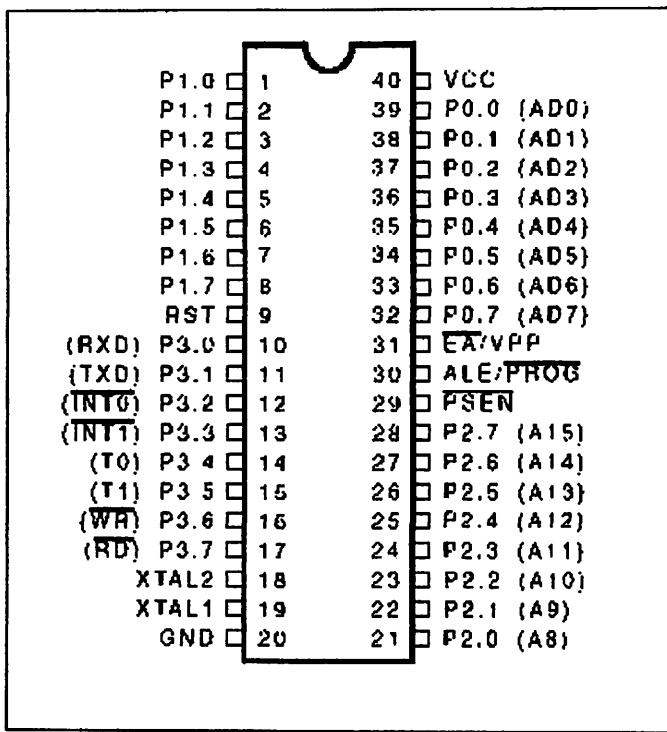


Gambar 2.7 Blok Diagram AT89S51

(*Data Sheet Microcontroller Atmel 89S51*)

2.12.2. Susunan Pin AT89S51

Mikrokontroller AT89S51, terdiri atas 40 pin dengan susunan dan penjelasan sebagai berikut :



Gambar 2.8 Susunan Pin AT89S51

(*Data Sheet AT8951*)

- Pin 1 sampai 8 (Port1) merupakan Port I/O 8 bit bidireksional (dua Arah) dengan internal Pullups. Port ini bisa digunakan untuk berbagai keperluan.
- Pin 9 (Reset) adalah masukan reset aktif tinggi. Pulsa transisi dari rendah ke tinggi akan mereset mikrikontroller.
- Pin 10 sampai 17 (Port 3) merupakan Port I/O 8 bit bidireksional (dua Arah) dengan internal Pullups. Port 3 ini mempunyai fungsi-fungsi alternative atau pengganti meliputi TxD (*Transmit Data*), RxD (*Receive Data*), Int 0 (*Interrupt 0*), Int 1 (*Interrupt 1*), T0 (*Timer 0*), T1 (*Timer 1*), WR (*Write*) dan RD (*Read*). Jika

fungsi ini tidak dipakai, pin-pin ini bisa digunakan sebagai port paralel serba guna.

- Pin 18 (XTAL 1) adalah pin masukan ke rangkaia osiltor internal. Sebuah osilator kristal atau sumber osilator luar dapat dipakai.
- Pin 19 (XTAL 2) adalah pin masukan ke rangkaian osiltor internal. Sebuah osilator kristal atau sumber osilator luar dapat dipakai.
- Pin 20 (Ground) adalah pin yang dihubungkan ke Vss atau ground.
- Pin 21 sampai 28 (Port 2) merupakan Port I/O 8 bit bidireksional (dua Arah) dengan internal Pullups. Port ini dapat juga dimultiplex untuk address atau data bus (MSB) selama mengakses program dan data memori external.
- Pin 29 (PSEN) atau Program Store Enable adalah pin yang membaca memori eksternal. Ketika AT89S51 sedang mengeksekusi kode dari eksternal program memori, PSEN ini diaktifkan dua kali setiap perputaran mesin.
- Pin 30 (ALE/PROG) atau Address Latch Enable adalah pin untuk mengurus alamat rendah selama mengakses memori eksternal. Pin ini juga untuk pulsa input program (PROG) selama memrograman PEROM. ALE juga dapat di buat disable (tidak aktif) dengan menyetting bit 0 dari SFR pada alamt 8EH. ALE diaktifkan dengan memberikan perintah MOVX atau MOVC.
- Pin 31 (EA/Vpp) adalah pin bila diberi logika tinggi akan melaksanakan instruksi dari intenal memori program (PEROM). Bila diberi logika rendah, mikrokontroller akan melaksanakan seluruh instruksi dari memori program luar.

- Pin 32 sampai 39 (Port 0) merupakan Port I/O 8 bit open drain dua arah. Bila digunakan untuk mengakses memori luar, port ini akan memultiplex alamat memori dengan data. Karena tidak mempunyai internal pullup maka untuk memberikan logika tinggi output dari port ini butuh eksternal pullup (resistor).
- Pin 40 (Vcc) dihubungkan dengan tegangan DC (+5 Volt).

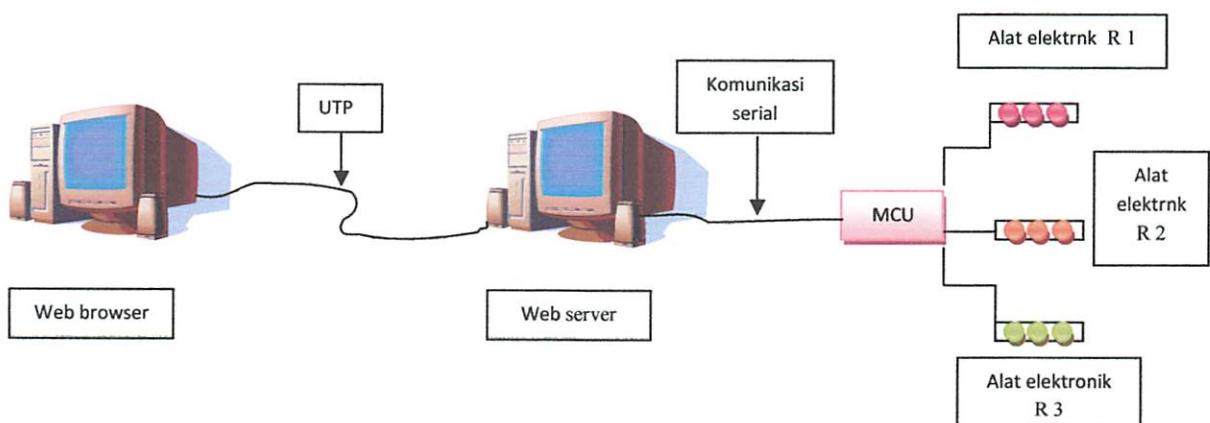
BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1. Umum

Bab ini menjelaskan masalah perencanaan dan implementasi *server* sebagai pusat komunikasi dalam sistem pengontrolan peralatan elektronik. Pengontrolan peralatan elektronik via *web* adalah sistem kontrol sekaligus monitoring kondisi peralatan elektronik di ruangan melalui sebuah *Client* yang memiliki fitur akses ke *server*. Namun, seperti telah diterangkan sebelumnya kita memakai *Apache web server* sehingga nantinya client akan mendial-in server dan akan mendownload program yang ada di server tersebut.

Pada aplikasi untuk melakukan pengontrolan sekaligus monitoring, pertama kali client akan mendial server ruangan dan akan mendownload program yang ada di *server*.



Gambar 3.1 Blok Diagram Sistem

3.2. Arsitektur Sistem

Ada beberapa komponen dalam sub sistem web browser dan web server ini, yaitu :

1. *Web Browser*
2. *Web Server*
 - *Borland Delphi API (Aplication Programing Interfacing)*
 - *Database*
3. *Sistem Kontrol*
 - *Mikrokontroler AT89S51*

Pada halaman sebelumnya telah diperlihatkan arsitektur sistem pengontrolan secara lengkap (*gambar 3.1*). Selanjutnya perencanaan tersebut akan diimplementasikan menjadi suatu sistem, dengan mempertimbangkan hal-hal sebagai berikut :

3.2.1. Web Browser

Agar dapat melakukan *browsing* dengan menggunakan *client*, kita harus merancang *web* dengan menggunakan *PHP*. memungkinkan *client* untuk menerima *content* dari *web server*. Komunikasinya ditangani melalui protokol-protokol jaringan standar, seperti HTTP dan TCP/IP.

3.2.1.1. Desain Antarmuka Aplikasi pada web browser

Sesuai dengan spesifikasi sistem di atas, sistem Pengontrolan ini diharapkan dapat dengan mudah dipakai oleh banyak orang dan operator yang mengoperasikan sistem ini. Untuk itu harus dibuat desain antarmuka yang mudah dipahami dan tidak terlalu rumit.

Ada beberapa macam desain antarmuka pada aplikasi ini baik halaman untuk login maupun halaman sistem pengontrolan itu sendiri, yaitu desain halaman Utama,

halaman info, halaman kontrol, view status setelah pengontrolan, halaman logout. Sedangkan desain menu ditampilkan berdasarkan user dan password setiap ruangan.

3.2.1.1.1. Halaman Utama



Gambar 3.2a Halaman Login pada Web Browser

Desain halaman Utama, terdapat dua entri untuk user dan password.

3.2.1.1.2. Halaman info



Gambar 3.3 halaman info

Halaman ini akan muncul setelah login berhasil. halaman ini akan menghubungkan kita kepada halaman yang sesuai dengan user dan password yang kita masukkan.

3.2.1.1.3. Halaman kontrol



Gambar 3.4 Halaman kontrol

Halaman ini yang akan memungkinkan client melakukan perubahan kondisi saklar alat-alat elektronik.

3.2.1.1.4. Halaman view status



Gambar 3.5 Halaman view status

Pada halaman ini kita bisa meklik pada View Status untuk melihat kondisi setelah penyetelan

3.2.1.1.5. Halaman logout



Gambar 3.6 Halaman logout

3.2.2. Web Server

Web server atau lebih tepatnya *World Wide Web server* merupakan sebuah *server internet* yang mampu melayani koneksi transfer data dalam protokol HTTP. Agar dapat diakses oleh *web browser*, *content* (data-data) pada *web server* dibuat dengan menggunakan bahasa pemrograman *web*. Program untuk *content* tersebut bersifat dinamis dan berhubungan erat dengan sebuah *database*.

Perangkat-perangkat lunak yang dibutuhkan untuk aplikasi *server*, berjalan di atas sistem operasi *Windows Xp*. Sistem operasi ini digunakan karena memiliki beberapa kelebihan, diantaranya :

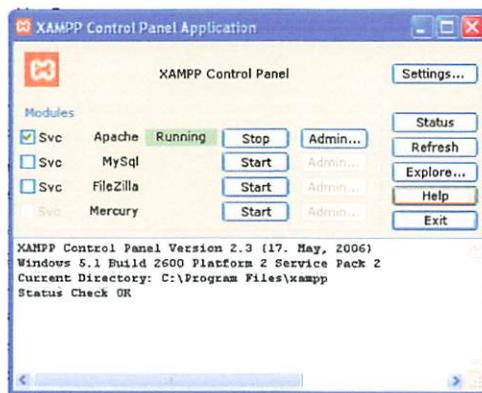
- *Familiar*.
- Mendukung aplikasi *multiuser* dan *multitasking*.
- Mendukung protokol TCP/IP.

Selain itu terdapat pertimbangan lain khususnya untuk aplikasi sistem yang dibuat, yaitu :

- Mendukung *Apache web server*.

- Mendukung *Borland Delphi*

Pada sistem ini, perangkat lunak yang digunakan sebagai *web server* adalah *Apache web server*. Pertimbangan yang diambil dalam memilih *web server* tersebut dikarenakan oleh performasi, kecepatan serta distribusinya yang gratis.



Gambar 3.7 Apache Web Server

Dalam sistem operasi *Windows Xp*, setelah proses instalasi *Apache web server* akan terdapat banyak file dan direktori, diantaranya adalah C:/program files/xampp/apache yang merupakan tempat file-file konfigurasi yaitu httpd.conf, srm.conf, mime.types dan access.conf. File-file tersebut digunakan untuk pemeliharaan konfigurasi *server*.

File httpd.conf merupakan file konfigurasi utama dari *Apache server* yang berisikan petunjuk konfigurasi yang mengontrol *server*, lokasi *logfile*, *user ID* dan *listening port*.

File srm.conf menyimpan informasi tentang lokasi dokumen *web*, lokasi dari program CGI, dan prioritas bahasa.

File access.conf berisi tipe akses, ketersediaan dokumen, dan setting keamanan.

Dari kesemua file tersebut di atas, dalam implementasinya hanya httpd.conf yang digunakan, karena file tersebut telah terkonfigurasi dan dapat diaplikasikan secara langsung. Sedangkan isi dari file srm.conf dan access.conf telah menjadi satu di dalam file httpd.conf tersebut.

Selanjutnya akan diuraikan hal-hal terpenting dalam konfigurasi file httpd.conf :

ServerType standalone

Sintaks : ServerType mode

Dengan menggunakan metode *standalone*, maka program *Apache* dapat dijalankan secara manual atau secara otomatis oleh komputer pada saat *booting*.

Port 80

Sintaks : Port nomor_port.

Perintah ini digunakan di dalam menentukan *port server* untuk aplikasi HTTP.

DocumentRoot "C:/Program Files/xampp/htdocs/mr"

Sintaks : DokumentRoot /lokasi/dokumen/html

Lokasi dokumen atau halaman-halaman *web* yang digunakan, ditentukan oleh perintah ini.

```
<Directory "C:/Program Files/xampp/htdocs">
```

Sintaks : DirectoryIndex nama-file-index

Perintah ini digunakan untuk menentukan *file indeks*. *File indeks* adalah *file* yang akan ditampilkan saat seorang *user* mengakses suatu alamat *homepage*.

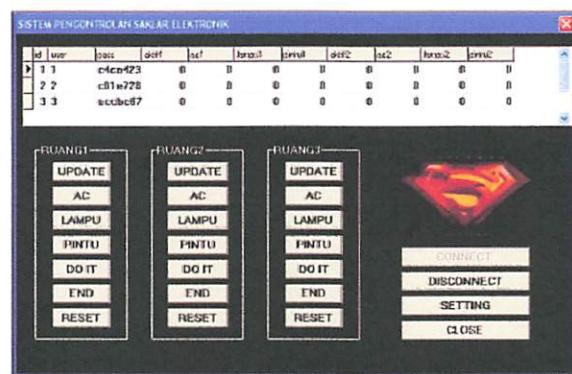
Pada *Apache* kita dapat menentukan *content type* apa yang akan dikirimkan ke *browser*. *Content type* ini akan membuat *browser* mengerti bagaimana menampilkan suatu *file*. Contohnya jika *content type* adalah *text/vnd.web.php*, maka *browser* mengerti bahwa ia harus menampilkan dokumen PHP.

Setelah melakukan instalasi dan konfigurasi pada *Apache web server*, kegiatan selanjutnya adalah melakukan instalasi modul PHP. PHP merupakan modul tambahan *server Apache* dan dianggap satu bagian integralnya, sehingga pembuatan aplikasi *web* dengan menggunakan PHP akan dapat memberikan hasil yang jauh lebih baik dalam hal kecepatan dan efisiensi server.

3.2.3. Borland Delphi API (*Aplication Programing Interfacing*)

Pada perencanaan software di komputer *server* ini kita membutuhkan dua macam software yaitu software untuk komunikasi dengan mikrokontroller dan software untuk komunikasi dengan *server windows Xp*.

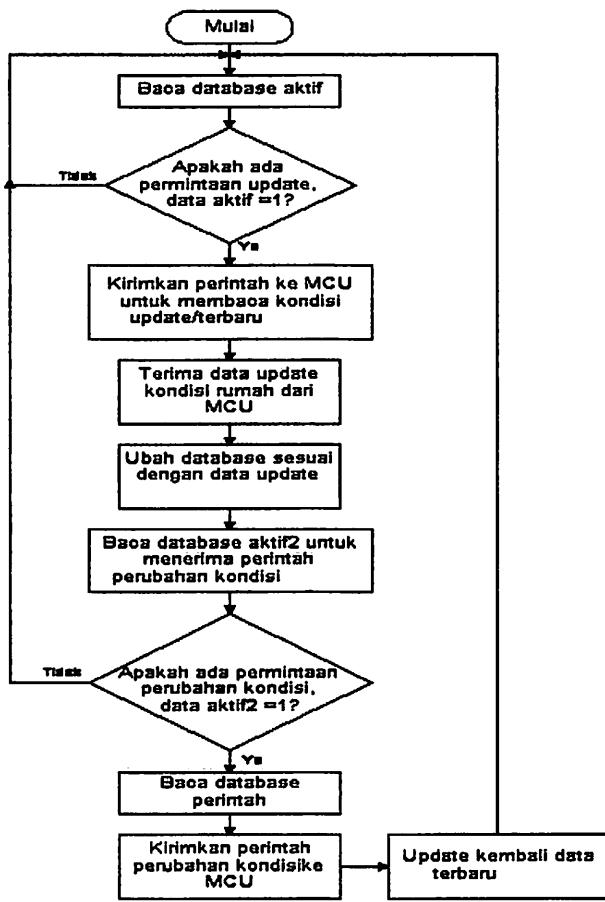
Software untuk komunikasi dengan *MCU* direncanakan menggunakan *Delphi* 7. Dengan Delphi ini pembuatan aplikasi menjadi lebih mudah selain didukung oleh VCL (*Visual Component Library*) untuk membuat tampilan yang menarik juga pada aplikasi database dengan komponen querynya. Sedangkan untuk aplikasi komunikasi kita harus menginstalnya secara terpisah yaitu Componen *CPORT Library*. Dengan komponen dalam *CPORT* ini, yaitu *Comport* kita akan membuat aplikasi komunikasi serial dengan mudah, karena didalamnya sudah disediakan berbagai pengesetan parameter komunikasi serial seperti *baudrate*, *port yang digunakan*, *parity*, *bitstop* dan sebagainya.



Gambar 3.8 Delphi Programing

Sedangkan untuk komunikasi dengan *server database MySQL windows Xp*, Delphi juga telah menyediakan komponen untuk membangun komunikasi, yaitu *ADO Connection*. Pada *ADO Connection* ini kita tinggal mengisi String Connection yaitu Informasi yang berisi komputer yang akan dihubungi, database dan sebagainya yang juga akan dihubungkan dengan *ODBC MySQL*.

Adapun flowchart program pada komputer kontrol ini seperti ditunjukkan pada gambar 3.4.



Gambar 3.9 Flowchart untuk Program Kontrol

3.2.4. Database

Database yang diaplikasikan dalam sistem ini akan memberikan layanan informasi kepada *client* tentang kondisi Ruangan di Kantor Jurusan, selain itu juga memberikan suatu keamanan kepada mereka dengan pemberian sebuah autentikasi yang berupa *user name* dan *password*.

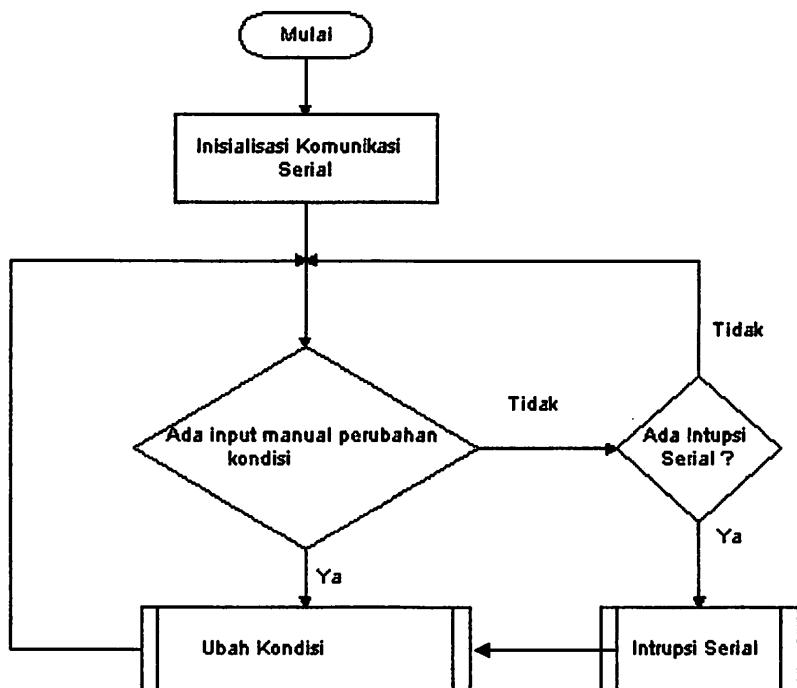
Program *database server* yang digunakan adalah *MySQL*. Pertimbangan yang diambil dalam memilih perangkat lunak ini adalah fungsi *interface* yang dimiliki oleh

PHP untuk dapat melihat secara interaktif data pada *MySQL*. Dengan kata lain *MySQL database* sangat mendukung PHP.

3.2.5. Perencanaan Perangkat Lunak Mikrokontroller

Perencanaan perangkat lunak (*software*) pada *mikrokontroller* ini berdasarkan bahasa *assembler 8051* dengan *ekstensi H51*. Softwae ini nantinya harus di kompile dengan software konversi sampai jadi *file .BIN*, yang nantinya akan dimasukkan kedalam *AT89S51* sebagai program yang siap dijalankan.

Untuk lebih memudahkan dalam perencanaan program h51 ini maka dibuat flowchart program, yakni gambaran jalannya program secara umum. Adapun flowchart pada *mikrokontroller* ini adalah seperti gambar 3.9

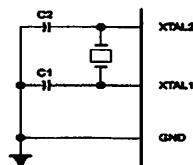


Gambar 3.11 Flowchart program pada MCU

3.3. Perencanaan Perangkat Keras

3.3.1. Rangkaian Mikrokontroller AT89S51

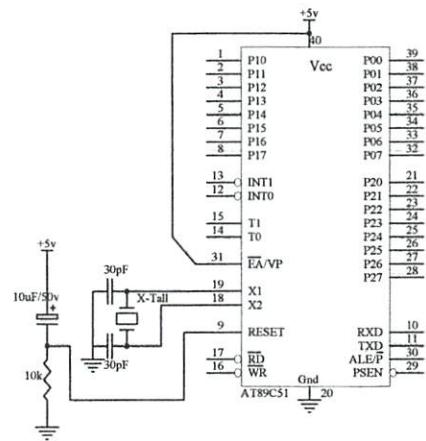
Untuk membuat rangkaian mikrokontroller selain sebuah *MCU* *AT89S51* diperlukan juga beberapa komponen pendukung diantaranya rangkaian osilator dan rangkaian reset. Rangkaian Osilator adalah gabungan antara kristal, kapasitor dan rangkaian inverter yang ada di dalam *IC* yang akan membangkitkan pulsa difrekuensi kristalnya. Kristal merupakan jantung dari *89S51* yaitu untuk membangkitkan pulsa *clock* yang mensinkronkan semua operasi internal dalam *MCU*. Biasanya sebuah kristal dan kapasitor dihubungkan seperti gambar 3.6 dan frekuensi kristal ini adalah frekuensi dasar *clock* internal dari mikrokontroller.



Gambar 3.12 Rangkaian Osilator

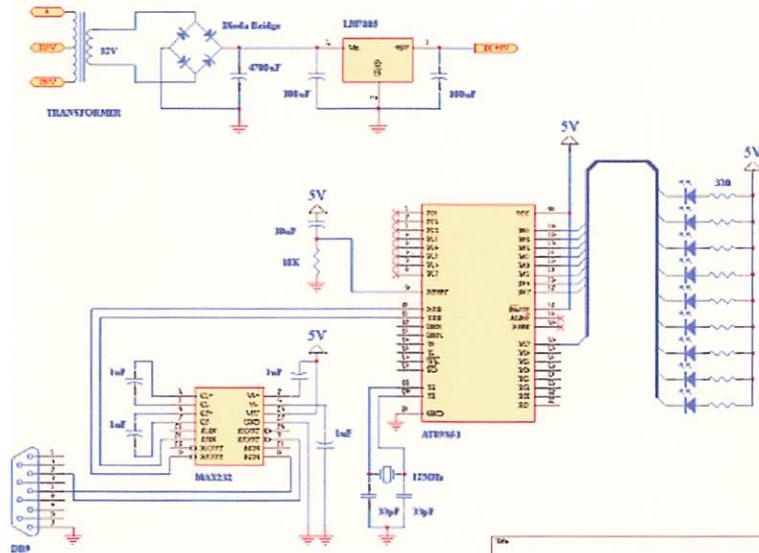
(*Datasheet AT89S51, 2009*)

Komunikasi juga membutuhkan frekuensi osilator yang dihubungkan dengan *internal counter* yang menentukan kecepatan *clock* dasar dari standar kecepatan komunikasi bit per detik (Baud). Bila frekuensi dasar *clock* tidak sesuai dengan yang disarankan maka frekuensi komunikasi sudah tidak standar lagi dan tentunya juga komunikasi dengan mikrokontroller tidak terjadi (gagal komunikasi).



Gambar 3.13 Rangkaian Mikrokontroller AT89S51

(Data Sheet Microcontroller Atmel 89S51)



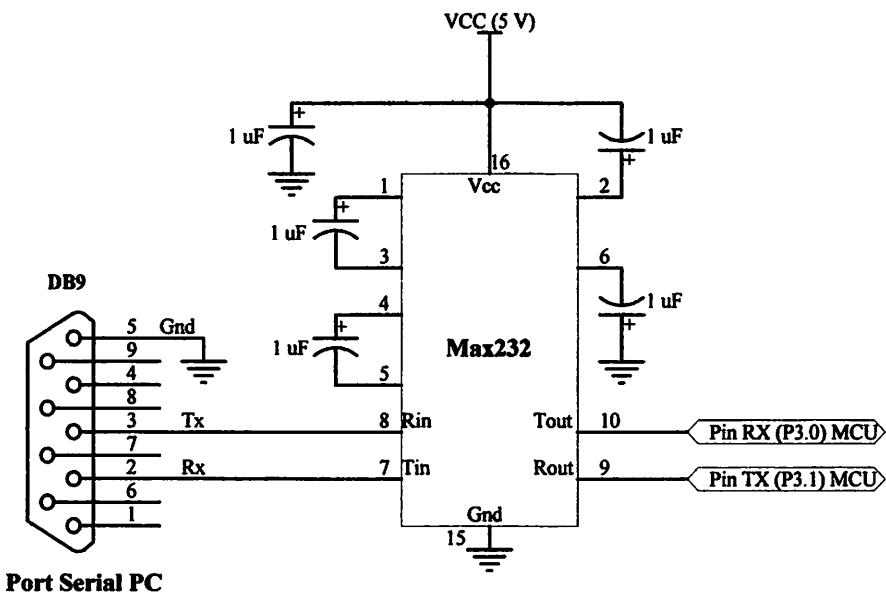
Gambar 3.14 skematik Rangkaian

Untuk nilai kapasitor C1 dan C2, karena resonator yang dipakai adalah kristal, maka nilai kapasitor yang dibutuhkan adalah sebesar 30pF, sesuai dengan datasheet.

Rangkaian reset tersusun dari sebuah kapasitor $10 \mu\text{F}$ dan sebuah resistor $10 \text{k}\Omega$. Saat catu daya dinyalakan, Rangkaian reset akan menahan logika rendah pada pin RST, sehingga program langsung dieksekusi. Untuk mereset atau mengembalikan MCU bekerja mulai dari instruksi awal maka kita bisa memasukkan logika tinggi pada pin RST dengan menshortkan kaki antara kapasitor $10 \mu\text{F}$. Pin EA (Enable All) kita hubungkan ke Vcc karena kita menggunakan program internal pada MCU.

3.3.2. Rangkaian IC MAX232

Mikrokontroller AT89S51 juga dilengkapi dengan port serial. Port serial ini memungkinkan kita mengirim data dalam format serial. Apabila hendak menghubungkan mikrokontroller AT89S51 dengan PC (komputer pribadi) melalui port serial maka level tegangan TTL yang dikeluarkan oleh *IC AT89S51* harus diubah menjadi level tegangan RS232 port serial pada PC. Pemilihan kapasitor disesuaikan dengan nilai pada datasheet *IC MAX232* yaitu $1\mu\text{F}$. Hubungannya diperlihatkan pada gambar 3.15



Gambar 3.15 Rangkaian IC MAX232

(*Data Sheet MAX232*)

3.4. Perencanaan dan Pembuatan Program Aplikasi

Dalam pembuatan program aplikasi ini, bahasa pemrograman yang digunakan adalah *HyperText Markup Language (HTML)* yang dipadukan dengan *script / kode-kode PHP*.

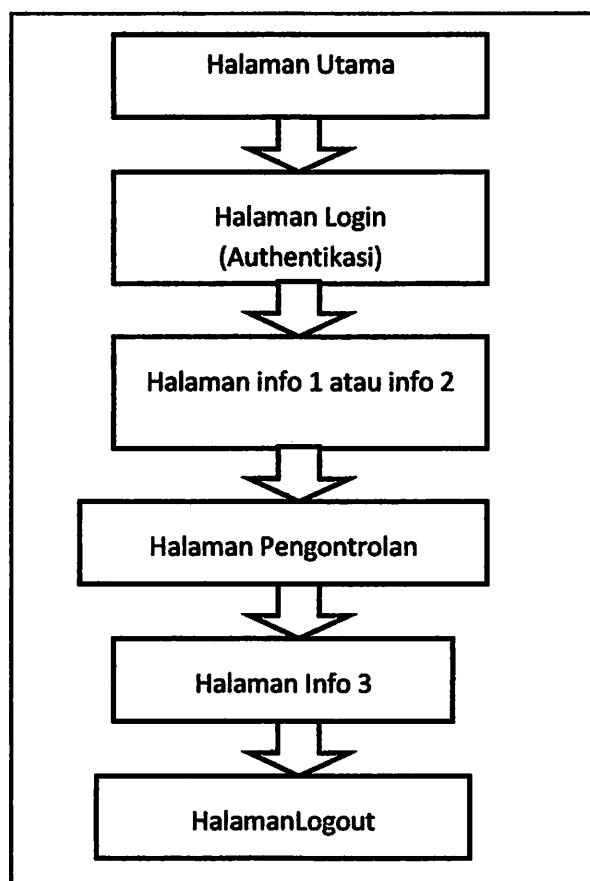
Secara umum, tampilan program pengontrolan memiliki alur sebagai berikut :

- Pertama, *client* akan dihadapkan dengan sebuah halaman utama yang merupakan halaman pembuka dari sistem telekontrol peralatan elektronik.
- Pada halaman berikutnya, *client* dihadapkan pada suatu sistem authentikasi yang terdiri dari *user name* dan *password*. Apabila *client* memberikan data yang benar, maka pengontrolan akan dapat segera

dilakukan. Namun jika sebaliknya, maka *client* akan diberikan kesempatan untuk mengulang proses tersebut (*Info 1*).

- Sebelum proses pengontrolan, *server* akan memberikan informasi (*Info 2*) kepada *client* kondisi terkini dari ruangan.
- Pada proses selanjutnya *client* akan diberikan kesempatan untuk melakukan pengontrolan, kemudian kondisi ruangan terkini dapat segera tertampilkan (*Info 3*).

Dari alur tersebut di atas dapat dipresentasikan dengan membuat sebuah algoritma :



Gambar 3.16 Algoritma Tampilan Program Pengontrolan

Untuk pembuatan *database* digunakan sebuah *software* bantu interaktif yang dinamakan *phpMyAdmin 2.2.Orc3*. Dengan perangkat lunak tersebut, kita dapat membuat *database* hanya dengan mengisi kolom-kolom yang telah tersedia.



Gambar 3.17 Php Myadmin

Dalam program telekontrol ini, *database* yang dibuat diberi nama **kontrol**. *Database* ini terdiri dari tabel *database* yang bernama **info**. Tabel info terdiri dari komponen-komponen seperti yang terlihat pada gambar di bawah ini :

	id	user	pass	aktif1	ac1	lampu1	pintu1	aktif2	ac2	lampu2	pintu2
1	ruang1	7825bec0a1aafbfba03d27752b5ee996		0	0	0	0	0	0	0	0
2	ruang2	0bfcdcd8191a53e117fd8d10954810e8		0	0	0	0	0	0	0	0
3	ruang3	0bfcdcd8191a53e117fd8d10954810e8		0	0	0	0	0	0	0	0

Tabel 3.1 Komponen-komponen Table Database kontrol

Kolom **aktif1** digunakan untuk memberi informasi kepada komputer kontrol bahwa terdapat *user* yang hendak melakukan pengontrolan. Dengan demikian komputer kontrol harus segera mengirimkan kondisi terkini yakni dengan mengisi kolom-kolom **ac1**, **lampu1** dan **pintu1**. Setelah kolom-kolom tersebut terisi maka *server* akan mengirimkan isi kolom-kolom tersebut ke *user*.

Sedangkan kolom **aktif2** digunakan oleh *server* untuk menginstruksikan komputer kontrol agar segera melakukan pengontrolan setelah kolom-kolom **ac2**, **pintu2** dan **lampu2** telah diisi oleh *user*.

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1. Implementasi Sistem

Tahap implementasi merupakan lanjutan dari tahap analisa sistem dan perancangan sistem yang sudah dibahas pada bab sebelumnya. Dalam aplikasi sistem pengontrolan alat elektronik via web ini, program yang digunakan dalam pengimplementasian digunakan apache web server, Borland Delphi 7 dan sebagai media penyimpanan data (basisdata) aplikasi ini menggunakan basis data MySQL.

Dalam pengujian pada sistem pengontrolan alat elektronik via web ini dibedakan menjadi dua, yaitu pengujian server dan database serta yang kedua yaitu pengujian program kontrol dan alat. Perangkat yang digunakan dalam pengujian , adalah sebagai berikut:

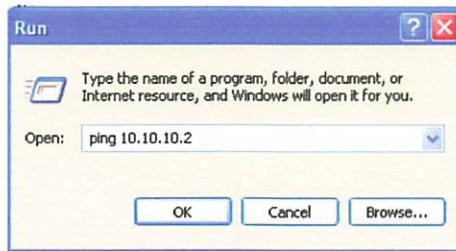
1. Perangkat Client
2. Perangkat Server
3. MCU AT89S51

4.2. Pengujian Koneksi LAN

Tujuan dari pengujian LAN ini (*subbab 2.5*) yaitu untuk mengkonfigurasi perangkat *server* agar dapat terkoneksi dengan perangkat *Client*.

- Langkah – langkah yang dilakukan pada pengujian ini adalah:
 1. menekan sebanyak 2x icon Local Area Connections pada taskbar windows. Atur internet protocol (TCP/IP) properties nya sesuai dengan IP address komputer client sebelum mengirimkan packet data, atur juga internet protocol (TCP/IP) properties pada komputer client isikan IP address komputer server.

2. mengirimkan packet data dengan cara memberikan perintah ping dari Run Menu kepada *client* untuk mengetahui apakah server sudah terkoneksi dengan client, gambar 4.1 Run Menu



Gambar 4.1 Run Menu

Hasil Pengujian Dari koneksi LAN adalah sebagai berikut :

```
Pinging 10.10.10.2 with 32 bytes of date:  
Reply from 10.10.10.2: bytes=32 time<1ms TTL=128  
Ping statistics for 10.10.10.2:  
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Kesimpulan dari pengujian diatas yaitu dimana packet data yang dikirimkan server diterima oleh client dengan sempurna (*0% loss*) yang berarti koneksi antara sisi server telah terkoneksi dengan client.

4.3. Pengujian Web Server

Tujuan dari pengujian diatas adalah untuk menguji *web server* (*sub bab 3.2.2*) apakah sudah dapat mengirimkan halaman pengontrolan kepada client. Pengujian komponen tersebut dengan menggunakan Web Browser pada sisi *client*. Adapun langkah Pengujian adalah sebagai berikut:

1. Dua buah komputer dihubungkan dengan kabel LAN

2. Salah satu sebagai komputer *client* menggunakan sebuah *Web browser*
3. Komputer *server* diberikan alamat IP 10.10.10.1 dan *client* dengan IP 10.10.10.2/5. Pemetaan IP dilakukan oleh komponen apache Web Server pada sisi *server*.
4. *Client* dapat membuka halaman *web* dengan menggunakan perintah <http://10.10.10.1/skripsi/index.php>

- Hasil Pengujian

Tampilan halaman *Browser* yang dihasilkan adalah :



Gambar 4.2 Tampilan Pengujian dengan *Web Browser*

Dari hasil pengujian diatas client dapat menampilkan halaman web pada sisi client maka dapat ditarik kesimpulan bahwa program aplikasi dapat berjalan dengan baik. Pemrosesan *request*, beserta respon dari *web server* dapat ditampilkan dengan perintah:

```
# copy /xampp/log/httpd/access_log con
```

hasilnya sebagai berikut:

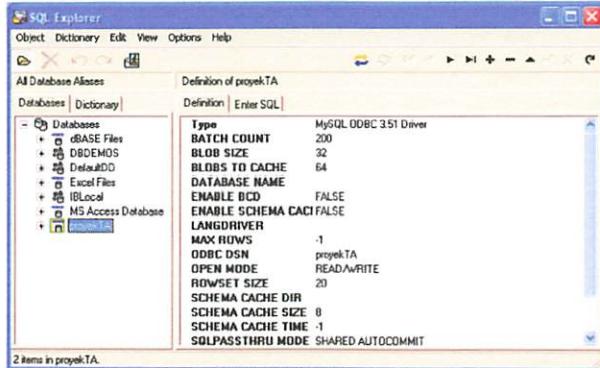
```
10.10.10.2 - - [20/july/2009:19:56:55 +0700] "GET /skripsi/ HTTP/1.1" 200 1098
10.10.10.2 - - [20/july/2009:19:56:55 +0700] "GET /skripsi/skull.wbmp HTTP/1.1"
304
10.10.10.2 - - [20/july/2009:19:57:06 +0700] "POST /skripsi/auth.php HTTP/1.1"
200 306
10.10.10.2 - - [20/july/2009:19:57:08 +0700] "GET /skripsi/infos.php?user=ruang1
HTTP/1.1" 200 426
10.10.10.2 - - [20/july/2009:19:57:42 +0700] "GET /skripsi/logout.php?user=ruang1
HTTP/1.1" 200 292
```

4.4. Pengujian Database MySQL

Tujuan pengujian ini akan mencoba membaca database MySQL. komponen databases ini digunakan untuk merubah kondisi saklar pada alat serta menginformasikan kondisi sebelum dan sesudah pengontrolan.

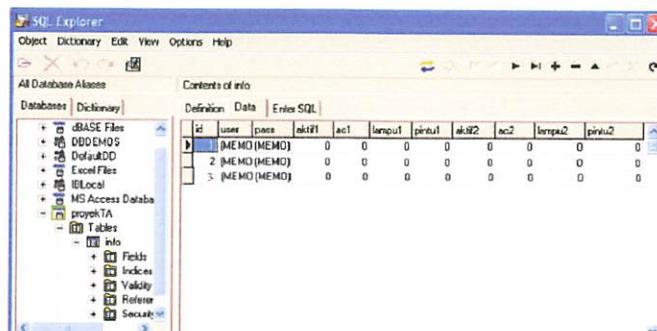
- **Langkah Pengujian**

Cara paling mudah yang bisa dicoba adalah membuka database kontrol menggunakan SQL Explorer Delphi. Jalankan SQL Explorer dan klik dua kali pada database ProyekTA seperti tampilan windows berikut ini :



Gambar 4.3 Menu Database pada SQL Explorer

Setelah itu akan muncul tampilan windows authentikasi, masukkan username dan password sesuai dengan username dan password yang telah diisi pada ODBC MySQL. dan jika tidak ada masalah maka database server akan langsung terbuka yang berisi table info dan users. Kliklah pada table info dan tabel Data maka anda akan dapat melihat database server seperti



Gambar 4.4 Tabel Database Server

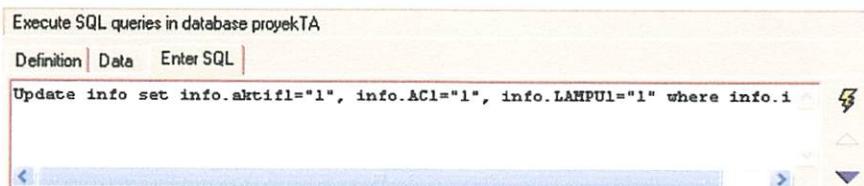
- Hasil Pengujian

Dari gambar tabel info diatas maka dapat disimpulkan bahwa *client* sudah bisa membaca database kontrol pada *server*. Hal ini berarti ODBC telah bekerja dengan baik, sehingga tabel info pada database server dapat dibaca.

Pengujian selanjutnya adalah dilakukan perubahan database dengan memberikan perintah SQL pada Tab ‘Enter SQL’ pada gambar 4.5. Pada perintah SQL akan dapat merubah kondisi ac1, lampu1 dan pintu1 pada ruang1 menjadi bernilai “1”. Perintah SQLnya sebagai berikut :

```
Update info set info.aktif1="1", info.AC1="1",
info.LAMPU1="1" where info.id="1"
```

Hasilnya seperti pada gambar 4.4



Gambar 4.5 Penulisan Perintah SQL

Anda eksekusi perintah SQL tersebut dengan menekan icon petir dan tutuplah SQL Explorer kemudian bukalah kembali database proyekTA dan didapat hasil data pada tabel info yang baru seperti pada gambar 4.6

Enter SQL										
id	user	pass	aktif1	ac1	lampu1	pintu1	aktif2	ac2	lampu2	pintu2
1	[MEMO (MEMO)]		1	1	1	0	0	0	0	0
2	[MEMO (MEMO)]		0	0	0	0	0	0	0	0
3	[MEMO (MEMO)]		0	0	0	0	0	0	0	0

Gambar 4.6 hasil Perubahan database

Dari hasil pengujian diatas, dapat ditarik kesimpulan bahwa hasil perubahan database diatas dapat diketahui. *client* tidak hanya sekedar membaca data tabel saja tetapi sudah bisa merubah database server. Keberhasilan membaca dan mengubah database server karena *client* telah terdaftar sebagai user di server database MySQL.

4.5. Pengujian Perangkat Keras

4.5.1. Menguji Mikrokontroller dengan Software

Tujuan dari pengujian ini adalah untuk menguji respon program yang diisikan pada mikrokontroller, apabila hasilnya sesuai dengan yang diharapkan berarti mikrokontroller dalam keadaan baik.

- Langkah Pengujian

1. Diisikan program yang sederhana, misalnya mengeluarkan output pada salah satu port AT89S51. Dibutuhkan led peraga agar dapat mengetahui dengan benar output yang dikeluarkan.
2. Disusun rangkaian sederhana mikrokontroller dengan led peraga.

kemudian diisikan program sederhana misalnya sebagai berikut :

	Org	00h
Mulai:	Mov	P0, #00001111B
	Acall	Delay
	Mov	P0, #11110000B
	Acall	Delay
	Sjmp	Mulai
Delay:	Mov	R0, #0

```
Delay1: Mov      R1,#0h
        Djnz    R1,Delay2
        Djnz    R0,Delay1
        Ret
        End
```

- **Hasil Pengujian**

Led menyala pada selang waktu tertentu pada P0.0-P0.3 dan P0.4-P0.7 secara bergantian.

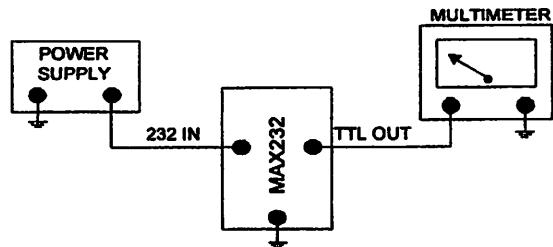
Dari hasil pengujian, program berjalan dengan semestinya yaitu hidup matinya led pada selang waktu tertentu pada P0.0-P0.3 dan P0.4-P0.7 secara bergantian. Hal ini berarti MCU dapat mengambil instruksi kerja sesuai dengan alamat pada PC (program Counter). Kesimpulan dari pengujian diatas adalah MCU dapat menjalankan program, berarti MCU dalam kondisi baik.

4.5.3. Pengujian MAX232

Tujuan pengujian ini yaitu untuk mengetahui IC MAX 232 apakah dapat bekerja dengan baik atau tidak.

- **Langkah Pengujian**

Pengujian pada IC MAX232 dibagi 2 yaitu pengujian respon pada RS 232_{IN} to TTL_{OUT} Converter dan pengujian respon pada TTL_{IN} to RS232_{OUT}. Untuk menguji rangkaian RS 232 to TTL converter, dipakai rangkaian sebagai berikut :



Gambar 4.7 Rangkaian Pengujian RS 232 to TTL

Langkah pengujian RS 232 to TTL

1. Rangkaian dihubungkan seperti gambar 4.8.
2. Diberikan input tegangan dengan logika ‘1’ yaitu antara -3V sampai -15V pada pin RS 232 input.
3. Ukur besarnya tegangan pada TTL output.
4. Diulangi langkah 3 dan 4 untuk logika ‘0’, yaitu +3V sampai +15V.

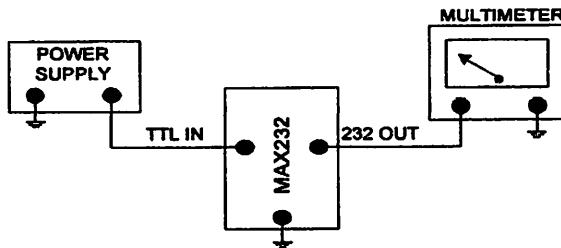
Data yang diperoleh pada pengujian ini adalah:

Table 4.1 Hasil pengujian RS 232 to TTL

Vcc	RS 232 input	TTL output
+5V	+10	0
+5V	-10	+5

Pengujian selanjutnya adalah TTL_{IN} to RS232_{OUT}. Langkah pengujian sebagai berikut :

1. Rangkaian dihubungkan seperti gambar 4.12, diberikan input logika ‘1’ yaitu antara +2,4V sampai +5V pada pin TTL input.



Gambar 4.8 Rangkaian Pengujian TTL to RS 232

2. Ukur besarnya tegangan pada RS 232 output.
 3. Diulangi langkah 3 dan 4 untuk logika ‘0’, yaitu 0V sampai +1,8V.
- Hasil Pengujian

Data yang diperoleh pada pengujian ini adalah:

Table 4.2 Hasil pengujian TTL to RS 232

Vcc	TTL input	RS 232 output
+5V	+5	-8,5
+5V	0	+8,5

Dari teori dasar disebutkan bahwa ketentuan RS232, level logika ‘1’ dinyatakan dengan tegangan antara -3 sampai -25 Volt, dan level logika ‘0’ dinyatakan dengan tegangan antara +3 sampai +25 Volt. Mengingat Hampir semua perangkat digital menggunakan level tegangan logika TTL atau CMOS sumber tegangan +5 Volt, dan level logika ‘0’ dinyatakan dengan tegangan antara 0,8 sampai 0 Volt dan level logika ‘1’ dinyatakan dengan tegangan 3,5 sampai 5 Volt.

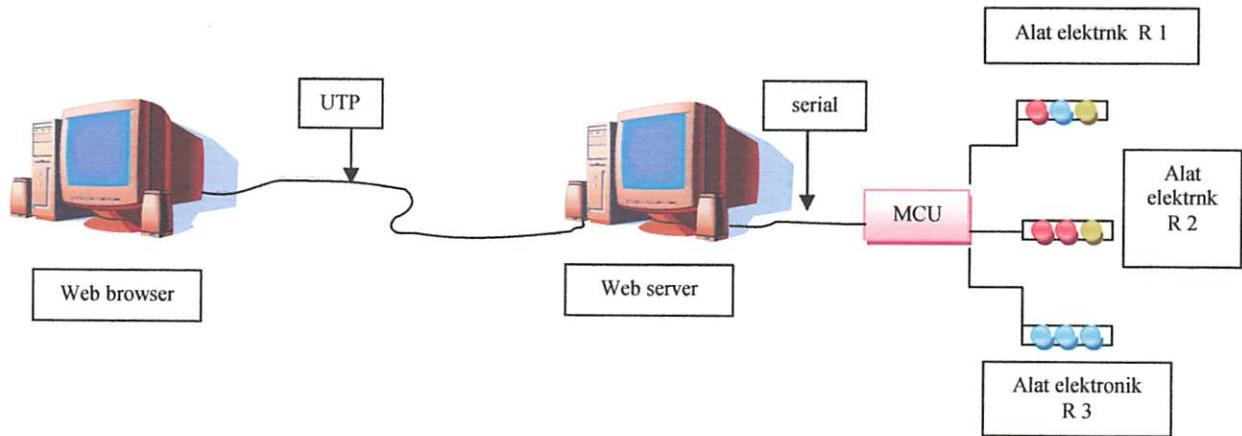
Dibandingkan dengan hasil pengujian, level tegangan yang dihasilkan dalam pengujian berada dalam range tegangan yang ditentukan untuk RS 232. Dengan melihat level tegangan RS 232 dan TTL, maka data yang diperoleh pada

pengujian ternyata sesuai dan masuk dalam range tegangannya, sehingga pada blok rangkaian RS 232 to TTL converter yang diuji sudah bekerja dengan baik.

4.6. Pengujian Sistem Sebagai Telekontrol Peralatan Elektronik

Tujuan pengujian keseluruhan sistem telekontrol ini yaitu akan mengubah database server dari client untuk mengontrol peralatan elektronik ruang.

- Diagram Pengujian

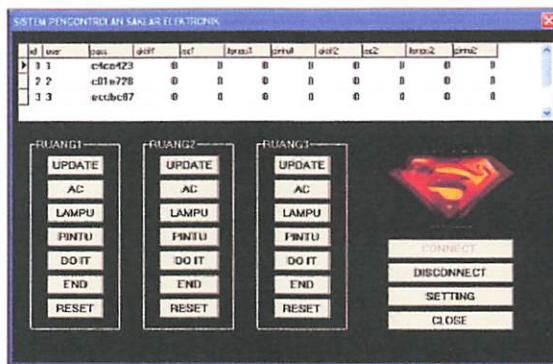


Gambar 4.9 Diagram Pengujian Sistem Telekontrol

- Langkah Pengujian

Pada Program kontrol, Delphi akan menjalankan Program khusus skripsiKU yang akan selalu berhubungan dengan server, software aplikasi ini dibiarkan dalam kondisi running sambil menunggu perubahan database server sebagai pengaktifan pengontrolan. Adapun langkah pengujian sebagai berikut :

1. Program pada Program kontrol dijalankan, klik tombol “connect”.

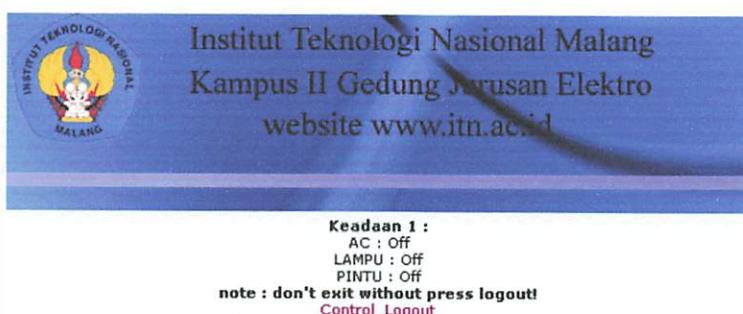


Gambar 4.10 Program Aplikasi pada Komputer Kontrol

2. Database server di-reset agar setiap perubahan data dapat diamati
3. Peralatan dikontrol secara manual mis. AC dan Lampu pada ruang1 dihidupkan.
4. Amati kondisi update yang ditampilkan pada layar Web browser
5. Kondisi peralatan diubah misalnya AC off, lampu Off dan Pintu On.
6. Amati kondisi setelah pengontrolan.

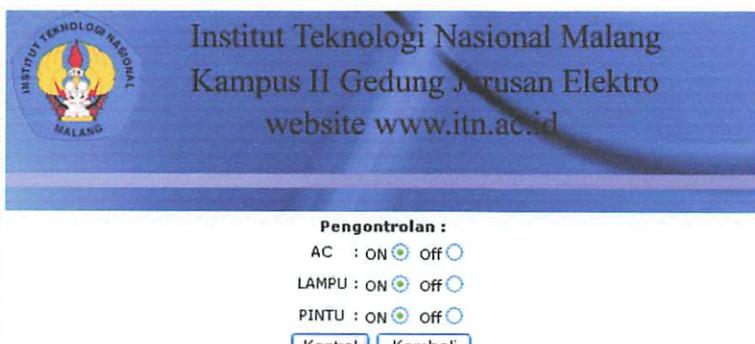
- Hasil Pengujian

Setelah dilakukan langkah pengujian diatas ternyata kondisi update AC dan lampu dapat di tampilkan dengan benar seperti terlihat pada gambar:



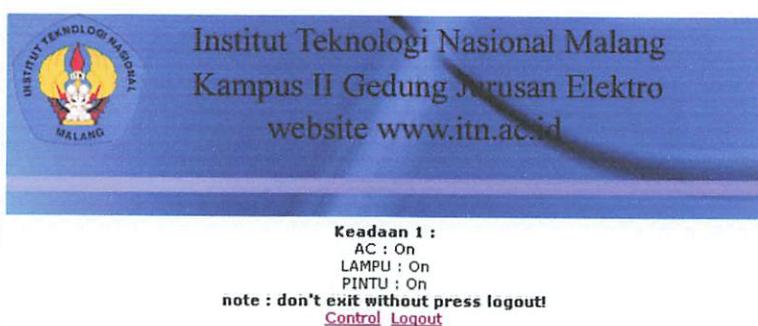
Gambar 4.11 Kondisi sebelum dikontrol

Setelah client melakukan login pada form utama,maka akan ditampilkan kondisi saklar alat elektronik terkini,seperti gambar 4.11. Apabila kita ingin melakukan perubahan kondisi saklar maka klik 'control', maka kita akan dibawa kehalaman update data sebagai berikut:



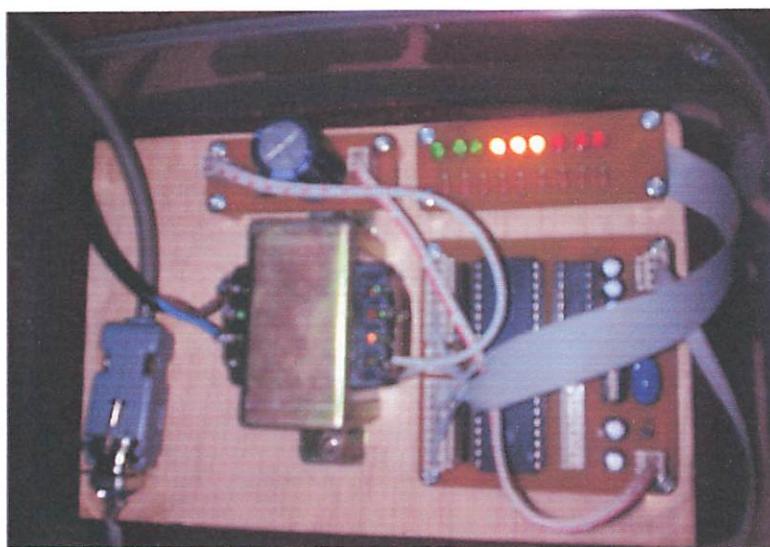
Gambar 4.12 form update kondisi saklar

Jika kita klik tombol kontrol,maka akan ditampilkan form kondisi setelah pengontrolan.



Gambar 4.14 Kondisi sesudah dikontrol

Kondisi pada peralatan pun berubah sesuai dengan perubahan kondisi yang telah dilakukan.



Gambar 4.15 Kondisi alat

Tabel 4.3. Pengujian sistem pengontrolan

input			Kondisi ruang 1			Kondisi ruang 2			Kondisi ruang 3		
			Lampu	AC	Pintu	Lampu	AC	Pintu	Lampu	AC	Pintu
0	0	0	Off	Off	Off	Off	Off	Off	Off	Off	off
0	0	1	Off	Off	On	Off	Off	On	Off	Off	On
0	1	0	Off	On	Off	Off	On	Off	Off	On	Off
0	1	1	Off	On	On	Off	On	On	Off	On	On
1	0	0	On	Off	Off	On	Off	Off	On	Off	Off
1	0	1	On	Off	On	On	Off	On	On	Off	On
1	1	0	On	On	Off	On	On	Off	On	On	Off
1	1	1	On	On	On	On	On	On	On	On	On

Dari hasil pengujian diatas ternyata kondisi update bisa langsung terlihat dengan cepat karena refresh database dari komputer kontrol ke komputer server dilakukan dengan cepat pula (<1 detik). Begitu juga halnya dengan melihat kondisi setelah dilakukan pengontrolan, kondisi terakhir pun bisa langsung terlihat kembali. Jika proses refresh database dilakukan begitu lama maka server akan mengambil data tabel sebelumnya sehingga data yang ditampilkan di layar tidak valid.

Hal ini bisa diatasi jika menunggu dulu proses refresh database dari komputer kontrol sampai selesai sehingga data nantinya akan valid. Dari pengujian diatas dapat ditarik kesimpulan :

1. Dari hasil perubahan kondisi peralatan elektronik diatas dapat diketahui bahwa pengujian yang dilakukan tidak hanya sekedar membaca data tabel saja tetapi sudah bisa mengubah database kontrol melalui client.
2. Komputer kontrol dapat mengakses server database MySQL untuk dapat melakukan pembacaan dan perubahan data, jika sudah terdaftar sebagai user pada server MySQL. Pengontrolan peralatan elektronik ruang juga bisa dikontrol melalui komputer kontrol yang sekaligus dapat mengubah database server.

Dari hasil pengujian pengontrolan yang dilakukan dua user pada waktu bersamaan, dimana user yang bisa melakukan pengontrolan adalah user yang login pertama. Hal ini dikarenakan proses pembacaan request oleh mikrokontroler yaitu karena jalur untuk komunikasi data seri RXD dan TXD diletakkan berhimpitan dengan P1.0 dan P1.1 dikaki nomor 2 dan 3. Jadi kalau sarana UART ini dipakai untuk jalur input/output seri maka P1.0 dan P1.1 tidak bisa dipakai untuk jalur input/output paralel. Sehingga proses request akan diproses satu persatu.Jika ada dua user yang melakukan pengontrolan secara bersamaan maka mikrokontroler akan mengirimkan pesan bahwa saluran sedang sibuk. Mikrokontroler akan membaca perintah dari user yang login pertama kali dengan delay 50 ms(< dari 1 detik) dari user lainnya.Sehingga setelah user 1 melakukan logout maka user yang lain bisa melakukan pengontrolan.

BAB V

PENUTUP

5.1 Kesimpulan

Dari pembahasan Perancangan dan pembuatan sistem pengontrolan alat elektronik via web, dapat diambil kesimpulan,yaitu:

- Dari hasil pengujian pengontrolan yang dilakukan dua user pada waktu bersamaan, dimana user yang bisa melakukan pengontrolan adalah user yang login pertama.Sedangkan user yang lain akan bisa melakukan pengontrolan apabila user 1 telah logout.
- Dari hasil pengujian alat, pada saat client menginputkan, misal **0** [**0= Off=**] maka lampu LED peraga akan mati. Jika client menginputkan **1**[**1= On**] maka lampu LED peraga akan menyala.
- Dari hasil perubahan kondisi peralatan elektronik diatas dapat diketahui bahwa pengujian yang dilakukan tidak hanya sekedar membaca data tabel saja tetapi sudah bisa mengubah database kontrol melalui client.
- Aplikasi Sistem Pengontrolan Peralatan Elektronik Via Web ini memiliki kontribusi yang sangat menguntungkan jika diterapkan langsung pada suatu instansi yang memiliki banyak ruangan seperti gedung perkantoran, perusahaan-perusahaan dan sebagainya. Yaitu mempermudah dalam pengontrolan alat elektronik dari sekian banyak ruangan tersebut. Sehingga Sistem pengontrolan baik itu menghidup atau mematikan peralatan elektronik dapat dilakukan dengan mudah yaitu cukup dengan mengakses web pengontrolannya saja..

5.2 Saran

Diharapkan system aplikasi pengontrolan ini dapat ditingkatkan dengan mengimplementasikan sistem monitoring camera sehingga kondisi disekitar ruang kontrol dapat termonitoring secara *real – time*.

DAFTAR PUSTAKA

- ✚ Ir.Inge Martina,36 Jam Belajar Komputer Pemrograman Internet dengan Delphi,Penerbit PT.Elex Media Koputindo,Jakarta,2002
- ✚ R.Budhi Widodo & Joseph Dedy Irawan, Interfacing Paralel & Serial menggunakan Delphi, Edisi Pertama, Penerbit:Graha Ilmu, Yogyakarta,2007.
- ✚ T.Marcus, Agus Prijono,Josef Widiadhi, Pemrograman Delphi Dengan ADOExpress, Edisi Revisi, Penerbit:Informatika, Bandung,2005.
- ✚ Taryana Suryana & Jonathan Sarwono, E-Commerce Menggunakan PHP dan MySQL, Edisi Pertama,Penerbit: Graha Ilmu,Yogyakarta,2007.
- ✚ Fathansyah, Basis Data, Edisi Pertama, Penerbit:CV.Informatika, Bandung,1999.
- ✚ www.data sheet AT89S8252.com
- ✚ Delphi.About.com
- ✚ www.yourwap.com

ZACKENRACH



FORMULIR BIMBINGAN SKRIPSI

Nama :MIRA ORISA
Nim :05.12.538
Masa Bimbingan :15 JUNI 2009 s/d 15 DESEMBER 2009
Judul Skripsi :MERANCANG APLIKASI SISTEM PENGONTROLAN PERALATAN ELEKTRONIK PADA GEDUNG JURUSAN TEKNIK ELEKTRO VIA WEB SERVER MENGGUNAKAN DELPHI 7

NO	TANGGAL	URAIAN	PARAF BIMBINGAN
1.	26/7/09	Bab I & II diperbaiki	
2.	27/7/09	Bab III	
3.	5/8/09	Bab IV	
4.		Demo	
5.		Survei dan Analisa	
6.	3/9/09	Laporan Akhir	
7.			
8.			
9.			
10.			

Malang,

Dosen Pimbimbing :

Ir. F. Yudi Limpraptono, MT
NIP. Y.10395900274



FORMULIR BIMBINGAN SKRIPSI

Nama : MIRA ORISA
 Nim : 06.12.588
 Masa Bimbingan : 15 JUNI 2009 s/d 15 DESEMBER 2009
 Judul Skripsi : MERANCANG APLIKASI SISTEM PENGONTROLAN PERALATAN ELEKTRONIK PADA GEDUNG JURUSAN TEKNIK ELEKTRO VIA WEB SERVER MENGGUNAKAN DELPHI 7

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1.	29/7/09	BAB I & II diperbaiki	✓
2.	28/7/09	BAB III	✓
3.	5/8/09	BAB IV	✓
4.	10/8/09	BAB V	✓
5.	13/8/09	Demo	✓
6.	18/8/09	Makalah seminar hasil	✓
7.	1/9/09	Laporan Akhir	✓
8.	4-9-2009	Anu rompi	✓
9.			
10.			

Malang,

Dosen Pembimbing II

Joseph Dedy Irawan, ST, MT
NIP.132315178

Form S-4B



FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata satu (S-1) Jurusan Teknik Elektro konsentrasi Teknik Komputer dan Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : MIRA ORISA
NIM : 05.12.588
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika
Masa Bimbingan : 15 Juni 2009 s/d 15 Desember 2009
Judul Skripsi : MERANCANG APLIKASI SISTEM PENGONTROLAN PERALATAN ELEKTRONIK VIA WEB.

No	Tanggal	Uraian	Paraf Pembimbing
1		Penambahan pengujian antar client yang login terlebih dahulu pada bab IV.	
2		Kesimpulan database apa yang dirubah melalui client	
3		Rumusan masalah tidak perlu menggunakan tanda tanya	

Dosen Penguji,

Penguji

Sandy Nataly Mantja, SKom.

Mengetahui

Dosen Pembimbing I

Ir. F. Yudi Limpraptono, MT

NIP.Y. 1039500274

Dosen Pembimbing II

Joseph Dedy Irawan, ST,MT

NIP.132315178

Program-program Kontrol

```
unit UtamaFrm;  
  
interface  
  
uses  
  
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
  StdCtrls, Grids, DBGrids, ExtCtrls, CPort, CPortCtl;  
  
type  
  
  TFmain = class(TForm)  
    dbgProyek: TDBGrid;  
    RUANG3: TGroupBox;  
    btDOIT3: TButton;  
    BTLAMPU3: TButton;  
    BTPIN3: TButton;  
    BTAC3: TButton;  
    btCLOSE: TButton;  
    ComPort1: TComPort;  
    RUANG1: TGroupBox;  
    RUANG2: TGroupBox;  
    BTLAMPU1: TButton;  
    BTPIN1: TButton;  
    BTAC1: TButton;
```

```
BTDOIT1: TButton;  
BTLAMPU2: TButton;  
BTPIN2: TButton;  
BTAC2: TButton;  
BTDOIT2: TButton;  
btUPDATE1: TButton;  
btUPDATE2: TButton;  
btUPDATE3: TButton;  
BtEND1: TButton;  
BTEND2: TButton;  
BTEND3: TButton;  
Btsetting: TButton;  
BConnect: TButton;  
Bdisconnect: TButton;  
Bres1: TButton;  
Bres2: TButton;  
Bres3: TButton;  
CekAktif: TTimer;  
Image1: TImage;  
ComDataPacket1: TComDataPacket;  
procedure btCLOSEClick(Sender: TObject);  
procedure BTLAMPU3Click(Sender: TObject);  
procedure BTPIN3Click(Sender: TObject);  
procedure BTAC3Click(Sender: TObject);  
procedure ComPort1RxChar(Sender: TObject; Count: Integer);  
procedure btDOIT3Click(Sender: TObject);
```

```
procedure btUPDATE3Click(Sender: TObject);
procedure btUPDATE2Click(Sender: TObject);
procedure btUPDATE1Click(Sender: TObject);
procedure BTLAMPU2Click(Sender: TObject);
procedure BTPIN2Click(Sender: TObject);
procedure BTAC2Click(Sender: TObject);
procedure BTDOIT2Click(Sender: TObject);
procedure BTLAMPU1Click(Sender: TObject);
procedure BTPIN1Click(Sender: TObject);
procedure BTAC1Click(Sender: TObject);
procedure BTDOIT1Click(Sender: TObject);
procedure BtEND1Click(Sender: TObject);
procedure BTEND2Click(Sender: TObject);
procedure BTEND3Click(Sender: TObject);
procedure BtsettingClick(Sender: TObject);
procedure BConnectClick(Sender: TObject);
procedure BdisconnectClick(Sender: TObject);
procedure Bres1Click(Sender: TObject);
procedure Bres2Click(Sender: TObject);
procedure Bres3Click(Sender: TObject);
procedure CekAktifTimer(Sender: TObject);
procedure UlangTimer(Sender: TObject);
procedure ComDataPacket1Packet(Sender: TObject; const Str: String);
procedure FormCreate(Sender: TObject);
private
  Procedure refresh(Sender: TObject);
```

```
{ Private declarations }

public

{ Public declarations }

end;

var

Fmain: TFmain;

UDSQL,STSQL,RESETSQL,UPDATE1,UPDATE2,UPDATE3,DOIT1,DOIT2,DOIT3,COL1,LAM1,PIN1,
COL2,LAM2,PIN2,COL3,LAM3,PIN3,DOCOL1,DOLAM1,DOPIN1,DOCOL2,DOLAM2,DOPIN2,
DOCOL3,DOLAM3,DOPIN3: STRING;
DTKIRIM, DATA,TEMP,DataA,DataB,Data1,Data2,Data3,Data4,
Data5,Data6,Data7,dtPINTU3,dtlampu3,dtac3,lamp,
aman,dtlampu1,dtPINTU1,dtac1,dtlampu2,dtPINTU2,dtac2:byte;
implementation

uses pDm;

{$R *.DFM}

procedure TFmain.btCLOSEClick(Sender: TObject);
begin
COMPORT1.Connected:=FALSE;
Application.terminate;
end;

procedure TFmain.BTLAMPU3Click(Sender: TObject);
begin
STSsql:='Select * from info Where info.id="3"';
dmP.opSQL(dmP.ADOsee,stssql);
LAM3:= dmP.ADOsee.Fields[5].AsString ;

```

```

IF LAM3='0' THEN
  udSQL:='Update info set info.LAMPU2="1" where info.id="3"'
ELSE UDSQL:='Update info set info.LAMPU2="0" where info.id="3"';
dmP.exSQL(dmP.ADOsee,UDsql);
end;

Procedure TFmain.refresh(Sender: TObject);
Begin
dmP.ADOtabel.close;
dmP.ADOtabel.open;
//dmP.ADOtabel.Refresh;
end;

procedure TFmain.BTPIN3Click(Sender: TObject);
begin
STsql:='Select * from info Where info.id="3"';
dmP.opSQL(dmP.ADOsee,stsql);
PIN3:= dmP.ADOsee.Fields[6].AsString ;
IF PIN3='0' THEN
  udSQL:='Update info set info.PINTU2="1" where info.id="3"'
ELSE UDSQL:='Update info set info.PINTU2="0" where info.id="3"';
dmP.exSQL(dmP.ADOsee,UDsql);
end;

procedure TFmain.BTAC3Click(Sender: TObject);
begin
STsql:='Select * from info Where info.id="3"';
dmP.opSQL(dmP.ADOsee,stsql);
COL3:= dmP.ADOsee.Fields[4].AsString ;

```

```
IF COL3='0' THEN
  UDSQL:='Update info set info.AC2="1" where info.id="3"'
ELSE UDSQL:='Update info set info.AC2="0" where info.id="3"';
dmP.exSQL(dmP.ADOsee,UDsql);
end;
procedure TFmain.ComPort1RxChar(Sender: TObject; Count: Integer);
begin
//comport1.Read(data,1);
{ if data and $80=$80 Then COL1 := '0' else COL1 := '1';
if data and $40=$40 then LAM1 := '0' else LAM1 := '1';
if data and $20=$20 then PIN1 := '0' else PIN1 := '1';

if data and $01=$01 then COL2 := '0' else COL2 := '1';
if data and $02=$02 then LAM2 := '0' else LAM2 := '1';
if data and $04=$04 then PIN2 := '0' else PIN2 := '1';

{case data of
  49: COL1:='1';
  50: LAM1:='1';
  51: PIN1:='1';
  52: COL2:='1';
  53: LAM2:='1';
  54: PIN2:='1';
  55: COL3:='1';
  56: LAM3:='1';
  57: PIN3:='1';
  65: COL1:='0';
```

```
66: LAM1 :='0';
67: PIN1 :='0';
68: COL2 :='0';
69: LAM2 :='0';
70: PIN2 :='0';
71: COL3 :='0';
72: LAM3 :='0';
73: PIN3 :='0';
else
CASE TEMP OF
83:BEGIN
    dataA:=data;
    //CARI BINERNYA
    data1 :=dataA div 2;
    dtAC2 :=dataA mod 2;
    data2 :=data1 div 2;
    dtLAMPU2:=data1 mod 2;
    data3 :=data2 div 2;
    dtPINTU2:=data2 mod 2;
    data4 :=data3 div 2;
    aman :=data3 mod 2;
    data5 :=data4 div 2;
    lamp :=data4 mod 2;
    data6 :=data5 div 2;
    dtAC1 :=data5 mod 2;
    data7 :=data6 div 2;
```

```
dtLAMPU1:=data6 mod 2;  
dtPINTU1:=data7 mod 2;  
  
//DETEKSI KONDISI RUANG1  
if dtAC1=1 then COL1:='1'  
else COL1:='0';  
if dtLAMPU1=1 then LAM1:='1'  
else LAM1:='0';  
if dtPINTU1=1 then PIN1:='1'  
else PIN1:='0';  
  
//DETEKSI KONDISI RUANG2  
if dtAC2=0 then COL2:='1'  
else COL2:='0';  
if dtLAMPU2=0 then LAM2:='1'  
else LAM2:='0';  
if dtPINTU2=0 then PIN2:='1'  
else PIN2:='0';  
END;
```

84: begin

```
dataB :=data;  
data1 :=dataB div 16;  
data5 :=data1 div 2;  
data6 :=data5 div 2;  
dtPINTU3:=data5 mod 2;
```

```
data7 :=data6 div 2;  
dtLAMPU3:=data6 mod 2;  
dtAC3 :=data7 mod 2;  
if dtAC3=0 then COL3:='1'  
else COL3:='0';  
if dtLAMPU3=0 then LAM3:='1'  
else LAM3:='0';  
if dtPINTU3=0 then PIN3:='1'  
else PIN3:='0';  
end;  
end;  
END;}  
END;
```

```
procedure TFmain.btDOIT3Click(Sender: TObject);  
begin  
UDSQL:='Update info set info.AKTIF2="1" where info.id="3"';  
dmP.exSQL(dmP.ADOsee,UDsql);  
end;
```

```
procedure TFmain.btUPDATE3Click(Sender: TObject);  
begin  
STsql:='Select * from info Where info.id="3"';  
dmP.opSQL(dmP.ADOsee,stsql);  
update3:= dmP.ADOsee.Fields[3].AsString ;  
if (update1='1') or (update2='1') then
```

```
showmessage('Maaf saluran sedang sibuk ')  
else  
begin  
IF update3='0' THEN  
    udSQL:='Update info set info.aktif1="1" where info.id="3"'  
ELSE UDSQL:='Update info set info.aktif1="0" where info.id="3"';  
dmP.exSQL(dmP.ADOsee,UDsql);  
end;  
end;  
  
procedure TFmain.btUPDATE2Click(Sender: TObject);  
begin  
STsql:='Select * from info Where info.id="2"';  
dmP.opSQL(dmP.ADOsee,stsql);  
update2:= dmP.ADOsee.Fields[3].AsString ;  
if (update1='1') or (update3='1') then  
showmessage('Maaf saluran sedang sibuk ')  
else  
begin  
IF update2='0' THEN  
    udSQL:='Update info set info.aktif1="1" where info.id="2"'  
ELSE UDSQL:='Update info set info.aktif1="0" where info.id="2"';  
dmP.exSQL(dmP.ADOsee,UDsql);  
end;  
end;
```

```
procedure TFmain.btUPDATE1Click(Sender: TObject);
begin
  STsql:='Select * from info Where info.id="1"';
  dmP.opSQL(dmP.ADOsee,stsql);
  update1:= dmP.ADOsee.Fields[3].AsString ;
  if (update2='1') or (update3='1') then
    showmessage('Maaf saluran sedang sibuk ')
  else
    begin
      IF update1='0' THEN
        udSQL:='Update info set info.aktif1="1" where info.id="1"'
      ELSE UDSQL:='Update info set info.aktif1="0" where info.id="1"';
      dmP.exSQL(dmP.ADOsee,UDsql);
    end;
end;

procedure TFmain.BTLAMPU2Click(Sender: TObject);
begin
  STsql:='Select * from info Where info.id="2"';
  dmP.opSQL(dmP.ADOsee,stsql);
  LAM2:= dmP.ADOsee.Fields[5].AsString ;
  IF LAM2='0' THEN
    udSQL:='Update info set info.LAMPU2="1" where info.id="2"'
  ELSE UDSQL:='Update info set info.LAMPU2="0" where info.id="2"';
  dmP.exSQL(dmP.ADOsee,UDsql);
end;
```

```
procedure TFmain.BTPIN2Click(Sender: TObject);
begin
  STsql:='Select * from info Where info.id="2"';
  dmP.opSQL(dmP.ADOsee,stsql);
  PIN2:= dmP.ADOsee.Fields[6].AsString ;
  IF PIN2='0' THEN
    udSQL:='Update info set info.PINTU2="1" where info.id="2"'
  ELSE
    UDSQL:='Update info set info.PINTU2="0" where info.id="2"';
  dmP.exSQL(dmP.ADOsee,UDsql);
end;
```

```
procedure TFmain.BTAC2Click(Sender: TObject);
begin
  STsql:='Select * from info Where info.id="2"';
  dmP.opSQL(dmP.ADOsee,stsql);
  COL2:= dmP.ADOsee.Fields[4].AsString ;
  IF COL2='0' THEN
    udSQL:='Update info set info.AC2="1" where info.id="2"'
  ELSE
    UDSQL:='Update info set info.AC2="0" where info.id="2"';
  dmP.exSQL(dmP.ADOsee,UDsql);
end;
```

```
procedure TFmain.BTDOIT2Click(Sender: TObject);
begin
  UDSQL:='Update info set info.AKTIF2="1" where info.id="2"';
```

```
dmP.exSQL(dmP.ADOsee,UDsql);

end;

procedure TFmain.BTLAMPU1Click(Sender: TObject);
begin
  STsql:='Select * from info Where info.id="1"';
  dmP.opSQL(dmP.ADOsee,stsql);
  LAM1:= dmP.ADOsee.Fields[5].AsString ;
  IF LAM1='0' THEN
    udSQL:='Update info set info.LAMPU2="1" where info.id="1"'
  ELSE UDSQL:='Update info set info.LAMPU2="0" where info.id="1"';
  dmP.exSQL(dmP.ADOsee,UDsql);
end;

procedure TFmain.BTPIN1Click(Sender: TObject);
begin
  STsql:='Select * from info Where info.id="1"';
  dmP.opSQL(dmP.ADOsee,stsql);
  PIN1:= dmP.ADOsee.Fields[6].AsString ;
  IF PIN1='0' THEN
    udSQL:='Update info set info.PINTU2="1" where info.id="1"'
  ELSE UDSQL:='Update info set info.PINTU2="0" where info.id="1"';
  dmP.exSQL(dmP.ADOsee,UDsql);
end;

procedure TFmain.BTAC1Click(Sender: TObject);
begin
  STsql:='Select * from info Where info.id="1"';
  dmP.opSQL(dmP.ADOsee,stsql);
```

```
COL1:= dmP.ADOsee.Fields[4].AsString ;  
  
IF COL1='0' THEN  
    udSQL:='Update info set info.AC2="1" where info.id="1"';  
ELSE UDSQL:='Update info set info.AC2="0" where info.id="1"';  
dmP.exSQL(dmP.ADOsee,UDsql);  
end;  
  
procedure TFmain.BTDOIT1Click(Sender: TObject);  
begin  
    UDSQL:='Update info set info.AKTIF2="1" where info.id="1"';  
    dmP.exSQL(dmP.ADOsee,UDsql);  
end;  
  
procedure TFmain.BtEND1Click(Sender: TObject);  
begin  
    UDSQL:='Update info set info.aktif1="0" where info.id="1"';  
    dmP.exSQL(dmP.ADOsee,UDsql);  
end;  
  
procedure TFmain.BTEND2Click(Sender: TObject);  
begin  
    UDSQL:='Update info set info.aktif1="0" where info.id="2"';  
    dmP.exSQL(dmP.ADOsee,UDsql);  
end;  
  
procedure TFmain.BTEND3Click(Sender: TObject);  
begin  
    UDSQL:='Update info set info.aktif1="0" where info.id="3"';  
    dmP.exSQL(dmP.ADOsee,UDsql);  
end;
```

```
procedure TFmain.BtsettingClick(Sender: TObject);
begin
  ComPort1.ShowSetupDialog;
end;

procedure TFmain.BConnectClick(Sender: TObject);
begin
  dmP.ADOTabel.Open;
  CekAktif.Enabled:=true;
  Bconnect.Enabled:=false;
  Bdisconnect.Enabled:=true;
end;

procedure TFmain.BdisconnectClick(Sender: TObject);
begin
  CekAktif.Enabled:=false;
  Bconnect.Enabled:=true;
  Bdisconnect.Enabled:=false;
end;

procedure TFmain.Bres1Click(Sender: TObject);
begin
  UDSQL:='Update info set info.aktif1="0",
info.AC1="0",info.LAMPU1="0",info.PINTU1="0",info.AKTIF2="0",info.AC2="0",info.LAMPU2="0
",info.PINTU2="0" where info.id="1"';
  dmP.exSQL(dmP.ADOsee,UDsql);
end;

procedure TFmain.Bres2Click(Sender: TObject);
begin
```

```

UDSQL:='Update info set
info.aktif1="0",info.AC1="0",info.LAMPU1="0",info.PINTU1="0",info.AKTIF2="0",info.AC2="0",i
nfo.LAMPU2="0",info.PINTU2="0" where info.id="2"'; 

dmP.exSQL(dmP.ADOsee,UDsql);

end;

procedure TFmain.Bres3Click(Sender: TObject);
begin

    UDSQL:='Update info set
info.aktif1="0",info.AC1="0",info.LAMPU1="0",info.PINTU1="0",info.AKTIF2="0",info.AC2="0",i
nfo.LAMPU2="0",info.PINTU2="0" where info.id="3"'; 

dmP.exSQL(dmP.ADOsee,UDsql);

end;

procedure TFmain.CekAktifTimer(Sender: TObject);
begin

// APAKAH ADA PERMINTAAN AKTIF1 RUANG1

stssql:='Select * from info Where info.id="1"'; 

dmP.opSQL(dmP.AdoQuery1,stssql);

UPDATE1:= dmP.AdoQuery1.Fields[3].AsString ;

//UPDATE DATA RUMAH1

IF UPDATE1='1' THEN

BEGIN

TEMP:=83;

comport1.write(TEMP,1);

Sleep(50);

Application.ProcessMessages;

UDSQL:='Update info set info.AC1="'+COL1+'", info.LAMPU1="'+LAM1+'",
info.PINTU1="'+PIN1+'" where info.id="1"'; 

dmP.exSQL(dmP.AdoQuery1,UDsql);

```

```
// APAKAH ADA PERMINTAAN AKTIF2 RUANG1
stssql:='Select * from info Where info.id="1"';
dmP.opSQL(dmP.AdoQuery1,stssql);
DOIT1:= dmP.AdoQuery1.Fields[7].AsString ;
//UPDATE DATA RUMAH1
IF DOIT1='1' THEN
BEGIN
stssql:='Select * from info Where info.id="1"';
dmP.opSQL(dmP.AdoQuery1,stssql);
//AMBIL PERINTAH
DOCOL1:= dmP.AdoQuery1.Fields[8].AsString ;
DOLAM1:= dmP.AdoQuery1.Fields[9].AsString ;
DOPIN1:= dmP.AdoQuery1.Fields[10].AsString ;
//UNTUK AC1
if DOCOL1='1' THEN
begin
dtkirim:=ORD('C');
Comport1.Write(dtirim,1);
end
ELSE
Begin
dtkirim:=ORD('3');
Comport1.Write(dtirim,1);
end;
//UNTUK LAMPU1
if DOLAM1='1' THEN
```

```
begin
  dtkirim:=ORD('B');
  Comport1.Write(dtkirim,1);
end
ELSE
Begin
  dtkirim:=ORD('2');
  Comport1.Write(dtkirim,1);
end;
//UNTUK PINTU1
if DOPIN1='1' THEN
begin
  dtkirim:=ORD('A');
  Comport1.Write(dtkirim,1);
end
ELSE
Begin
  dtkirim:=ORD('1');
  Comport1.Write(dtkirim,1);
end;
UDSQL:='Update info set info.AKTIF2="0" where info.id="1"';
dmP.exSQL(dmP.AdoQuery1,UDsql);
END;
END
ELSE
BEGIN
```

```
// APAKAH ADA PERMINTAAN AKTIF1 RUANG2
stssql:='Select * from info Where info.id="2"';
dmP.opSQL(dmP.AdoQuery1,stssql);

UPDATE2:= dmP.AdoQuery1.Fields[3].AsString ;
//UPDATE DATA RUMAH2

IF UPDATE2='1' THEN
BEGIN
TEMP:=83;
comport1.write(TEMP,1);
Sleep(50);
Application.ProcessMessages;

UDSQL:='Update info set info.AC1="'+COL2+'", info.LAMPU1="'+LAM2+'",
info.PINTU1="'+PIN2+'" where info.id="2"';
dmP.exSQL(dmP.AdoQuery1,UDsql);

// APAKAH ADA PERMINTAAN AKTIF2 RUMAH2
stssql:='Select * from info Where info.id="2"';
dmP.opSQL(dmP.AdoQuery1,stssql);

DOIT2:= dmP.AdoQuery1.Fields[7].AsString ;
//UPDATE DATA RUMAH2

IF DOIT2='1' THEN
BEGIN
stssql:='Select * from info Where info.id="2"';
dmP.opSQL(dmP.AdoQuery1,stssql);

//AMBIL PERINTAH

DOCOL2:= dmP.AdoQuery1.Fields[8].AsString ;
DOLAM2:= dmP.AdoQuery1.Fields[9].AsString ;
DOPIN2:= dmP.AdoQuery1.Fields[10].AsString ;
```

```
//UNTUK AC2
if DOCOL2='1' THEN
begin
dtkirim:=54;
Comport1.Write(dtkirim,1);
end
ELSE
Begin
dtkirim:=70;
Comport1.Write(dtkirim,1);
end;
//UNTUK LAMPU2
if DOLAM2='1' THEN
begin
dtkirim:=53;
Comport1.Write(dtkirim,1);
end
ELSE
Begin
dtkirim:=69;
Comport1.Write(dtkirim,1);
end;
//UNTUK PINTU2
if DOPIN2='1' THEN
begin
dtkirim:=52;
```

```
Comport1.Write(dtkirim,1);
end
ELSE
Begin
dtkirim:=68;
Comport1.Write(dtkirim,1);
end;

UDSQL:='Update info set info.AKTIF2="0" where info.id="2"';
dmP.exSQL(dmP.AdoQuery1,UDsql);
END;
END
else
begin
// APAKAH ADA PERMINTAAN AKTIF1 RUANG3
stsql:='Select * from info Where info.id="3"';
dmP.opSQL(dmP.AdoQuery1,stsql);
UPDATE3:= dmP.AdoQuery1.Fields[3].AsString ;
//UPDATE DATA RUMAH3
IF UPDATE3='1' THEN
BEGIN
TEMP:=83;
comport1.write(TEMP,1);
Sleep(50);
Application.ProcessMessages;
UDSQL:='Update info set info.AC1="'+COL3+'", info.LAMPU1="'+LAM3+'",
info.PINTU1="'+PIN3+'" where info.id="3"';
dmP.exSQL(dmP.AdoQuery1,UDsql);
```

```
// APAKAH ADA PERMINTAAN AKTIF2 RUANG3  
stsqli:='Select * from info Where info.id="3"';  
dmP.opSQL(dmP.AdoQuery1,stsqli);  
DOIT3:= dmP.AdoQuery1.Fields[7].AsString ;  
//UPDATE DATA RUMAH3  
IF DOIT3='1' THEN  
BEGIN  
stsqli:='Select * from info Where info.id="3"';  
dmP.opSQL(dmP.AdoQuery1,stsqli);  
//AMBIL PERINTAH  
DOCOL3:= dmP.AdoQuery1.Fields[8].AsString ;  
DOLAM3:= dmP.AdoQuery1.Fields[9].AsString ;  
DOPIN3:= dmP.AdoQuery1.Fields[10].AsString ;  
//UNTUK AC3  
if DOCOL3='1' THEN  
begin  
dtkirim:=57;  
Comport1.Write(dtkirim,1);  
end  
ELSE  
Begin  
dtkirim:=73;  
Comport1.Write(dtkirim,1);  
end;  
//UNTUK LAMPU3  
if DOLAM3='1' THEN
```

```
begin
  dtkirim:=56;
  Comport1.Write(dtkirim,1);
end
ELSE
Begin
  dtkirim:=72;
  Comport1.Write(dtkirim,1);
end;
//UNTUK PINTU3
if DOPIN3='1' THEN
begin
  dtkirim:=55;
  Comport1.Write(dtkirim,1);
end
ELSE
Begin
  dtkirim:=71;
  Comport1.Write(dtkirim,1);
end;
UDSQL:='Update info set info.AKTIF2="0" where info.id="3"';
dmP.exSQL(dmP.AdoQuery1,UDsql);
END;
END;
end;
end;
```

```
refresh(self);

dbgProyek.Columns[0].Width := 20;

dbgProyek.Columns[1].Width := 60;

dbgProyek.Columns[2].Width := 60;

dbgProyek.Columns[3].Width := 60;

dbgProyek.Columns[4].Width := 60;

dbgProyek.Columns[5].Width := 60;

dbgProyek.Columns[6].Width := 60;

dbgProyek.Columns[7].Width := 60;

dbgProyek.Columns[8].Width := 60;

dbgProyek.Columns[9].Width := 60;

dbgProyek.Columns[10].Width := 60;

end;

procedure TFmain.UlangTimer(Sender: TObject);
begin

refresh(self);

end;

procedure TFmain.ComDataPacket1Packet(Sender: TObject; const Str: String);
var
Dt1,
Dt2: Byte;
begin
Dt1 := Ord(Str[1]);
Dt2 := Ord(Str[2]);
// Caption := IntToStr(Dt2);

```

```
if dt1 and $80=$80 Then COL1 := '0' else COL1 := '1';
if dt1 and $40=$40 then LAM1 := '0' else LAM1 := '1';
if dt1 and $20=$20 then PIN1 := '0' else PIN1 := '1';

if dt1 and $04=$04 then COL2 := '0' else COL2 := '1';
if dt1 and $02=$02 then LAM2 := '0' else LAM2 := '1';
if dt1 and $01=$01 then PIN2 := '0' else PIN2 := '1';

if dt2 and $20=$20 then COL3 := '0' else COL3 := '1';
if dt2 and $40=$40 then LAM3 := '0' else LAM3 := '1';
if dt2 and $80=$80 then PIN3 := '0' else PIN3 := '1';

end;
end.
```

```

index.php

php
$LID=$_POST['user'];
$PWD=md5($_POST['password']);
include"kon.php";
$sql="SELECT * FROM info WHERE user='$LID' AND pass='$PWD'";
$result=@mysql_query($sql,$db);
if (mysql_num_rows($result) != 0)
{
    $myrow = mysql_fetch_row($result);
    mysql_query("select aktif1 from info where user='".$user."','".$db');
    mysql_query("update info set aktif1='1' where user='".$user."','".$db');
    print"<script
language=javascript>window.location='infos.php?user=$_POST[user]';</script>";
}

DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"
xml ns="http://www.w3.org/1999/xhtml">
head>
meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
title>...Aplikasi Sistem Pengontrolan Peralatan Elektronik Via Web:...</title>
link rel="stylesheet" type="text/css" href="style/css/css.css">
head>
body>





```

```

infos.php
-----  

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>...:Pengontrolan Peralatan Elektronik:...</title>
<link rel="stylesheet" type="text/css" href="style/css/css.css">
</head>
<body>





```

```

control.php
-----  

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>...Pengontrolan Peralatan Elektronik...</title>
<link rel="stylesheet" type="text/css" href="style/css/css.css">
</head>
<body>
<table style="border:1px solid" align="center">
    <tr>
        <td></td>
    </tr>
    <tr>
        <td align="center">
            <form action="set.php?user=<?php echo $_GET['user'];?>" method="get">
                <p>
                    <?php
                        $db = mysql_pconnect("localhost", "root", "");
                        mysql_select_db("kontrol", $db);
                        mysql_query("update info set aktif1='0' where
                        user='". $_GET['user']."'","");
                        $query = "select ac1, pintu1, lampu1 from info where
                        user='". $_GET['user']."'";
                        $result = @mysql_query($query);
                        print "<b>Pengontrolan :</b> <br/>";
                        if (mysql_num_rows($result)) :
                            while ($row = mysql_fetch_array($result)) :
                                $ac1 = "$row[ac1]";
                                $pintu1 = "$row[pintu1]";
                                $lampu1 = "$row[lampu1]";
                            endwhile;
                        else:
                            print "Waiting for Info.";
                        endif;
                    ?>
                <table>
                    <tr>
                        <td>AC</td>
                        <td>:</td>
                        <td>ON<input type="radio" name="ac2" value="1" /> Off<input
                        type="radio" name="ac2" value="0" /></td>
                    </tr><!--<input title="set" name="ac2" value="<? //echo $ac2
                        ?"/>-->
                    <tr>
                        <td>LAMPU</td>
                        <td>:</td>
                        <td>ON<input type="radio" name="lampa2" value="1" />
                        <input type="radio" name="lampa2" value="0" /></td>
                    </tr><!--<input title="set" name="lampa2" value="<? //echo
                        $lampa2 ?"/>-->
                    <tr>
                        <td>PINTU</td>
                        <td>:</td>
                        <td>ON<input type="radio" name="pintu2" value="1" />
                        <input type="radio" name="pintu2" value="0" /></td>
                    </tr><!--<input title="set" name="pintu2" value="<? //echo
                        $pintu2 ?"/>-->
                </table>
                <input type="hidden" name="user" value="<?php echo $_GET['user'];?>">
                <input type="submit" name="masuk" value="Kontrol" />
                <input type="button" value="Kembali" />
                <click="document.location='infos.php?user=<?php echo $_GET['user'];?>'" />
            </p>
        </form>
    </td>
</tr>

```

control.php

```
<tr> <td>  
      </td>  
</tr>  
<table>  
<body>  
<html>
```

```

set.php
-----  

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  

  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  

<html xmlns="http://www.w3.org/1999/xhtml">  

<head>  

  <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />  

  <title>...: Pengontrolan Peralatan Elektronik:...</title>  

  <link rel="stylesheet" type="text/css" href="style/css/css.css">  

</head>  

<body>  

  <table style="border:1px solid" align="center">  

    <tr>  

      <td></td>  

    </tr>  

    <tr>  

      <td align="center">  

        $LID=chop("".$_GET['user']);  

        $PWD=chop("".$_GET['pass']);  

        $user = $_GET['user'];  

        $ac2 = $_GET['ac2'];  

        $lampu2 = $_GET['lampu2'];  

        $pintu2 = $_GET['pintu2'];  

        $db = mysql_pconnect("localhost", "root","");
        mysql_select_db("kontrol", $db);
        mysql_query("select aktif2, ac2, pintu2, lampu2 from info where
        user='".$_GET['user']."'",$db);
        mysql_query("update info set aktif2='1', ac2='$ac2', pintu2='$pintu2',
        lampu2='$lampu2' where user='".$_GET['user']."'",$db);
        mysql_query("select aktif1 from info where
        user='".$_GET['user']."'",$db);
        //mysql_query("ac1, pintu1, lampu1 from info where user='".$user"', $db);
        mysql_query("update info set aktif1='1'where
        user='".$_GET['user']."'",$db);
        mysql_query("update info set ac1='$ac2', pintu1='$pintu2',
        lampu1='$lampu2' where user='".$user"'",$db);
        mysql_close();  

        Setting Success ...</b><br/>
        <a href="infos.php?user=<?php echo $_GET['user'];?>">View Status</a>
      </td>
    </tr>
    <tr>
      <td>
        </td>
    </tr>
  </table>
</body>
</html>

```

```
logout.php
DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
xml xmlns="http://www.w3.org/1999/xhtml">
head>
meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
meta http-equiv="refresh" content="2;URL=index.php">
title>...::Pengontrolan Peralatan Elektronik:::</title>
link rel="stylesheet" type="text/css" href="style/css/css.css">
head>
body>





```

Program-program Kontrol

```
unit UtamaFrm;  
  
interface  
  
uses  
  
  Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,  
  StdCtrls, Grids, DBGrids, ExtCtrls, CPort, CPortCtl;  
  
type  
  
  TFmain = class(TForm)  
  
    dbgProyek: TDBGrid;  
  
    RUMAH3: TGroupBox;  
  
    btDOIT3: TButton;  
  
    BTLAMPU3: TButton;  
  
    BTPIN3: TButton;  
  
    BTAC3: TButton;  
  
    btCLOSE: TButton;  
  
    ComPort1: TComPort;  
  
    RUMAH1: TGroupBox;  
  
    RUMAH2: TGroupBox;  
  
    BTLAMPU1: TButton;  
  
    BTPIN1: TButton;  
  
    BTAC1: TButton;
```

```
if dt1 and $80=$80 Then COL1 := '0' else COL1 := '1';
if dt1 and $40=$40 then LAM1 := '0' else LAM1 := '1';
if dt1 and $20=$20 then PIN1 := '0' else PIN1 := '1';

if dt1 and $04=$04 then COL2 := '0' else COL2 := '1';
if dt1 and $02=$02 then LAM2 := '0' else LAM2 := '1';
if dt1 and $01=$01 then PIN2 := '0' else PIN2 := '1';

if dt2 and $20=$20 then COL3 := '0' else COL3 := '1';
if dt2 and $40=$40 then LAM3 := '0' else LAM3 := '1';
if dt2 and $80=$80 then PIN3 := '0' else PIN3 := '1';

end;
end.
```

TAMPILAN WEB BROWSER PADA SISI CLIENT

Institut Teknologi Nasional Malang
Kampus II Gedung Jurusan Elektro
website www.itn.ac.id

LOGIN, Please.....

user

password

Welcome to ITN Malang

Aplikasi Pengontrolan Alat Elektronik
Gedung Jurusan Elektro
TA 2009

TAMPILAN KOMPUTER KONTROL PADA SISI SERVER

SISTEM PENGONTROLAN SAKLAR ELEKTRONIK

id	user	pass	oktfl1	act1	lampu1	pintu1	oktfl2	act2	lampu2	pintu2
1	1	c4cc4123	0	0	0	0	0	0	0	0
2	2	c01e728	0	0	0	0	0	0	0	0
3	3	eccdbc87	0	0	0	0	0	0	0	0

-RUANG1-

UPDATE
AC
LAMPU
PINTU
DO IT
END
RESET

-RUANG2-

UPDATE
AC
LAMPU
PINTU
DO IT
END
RESET

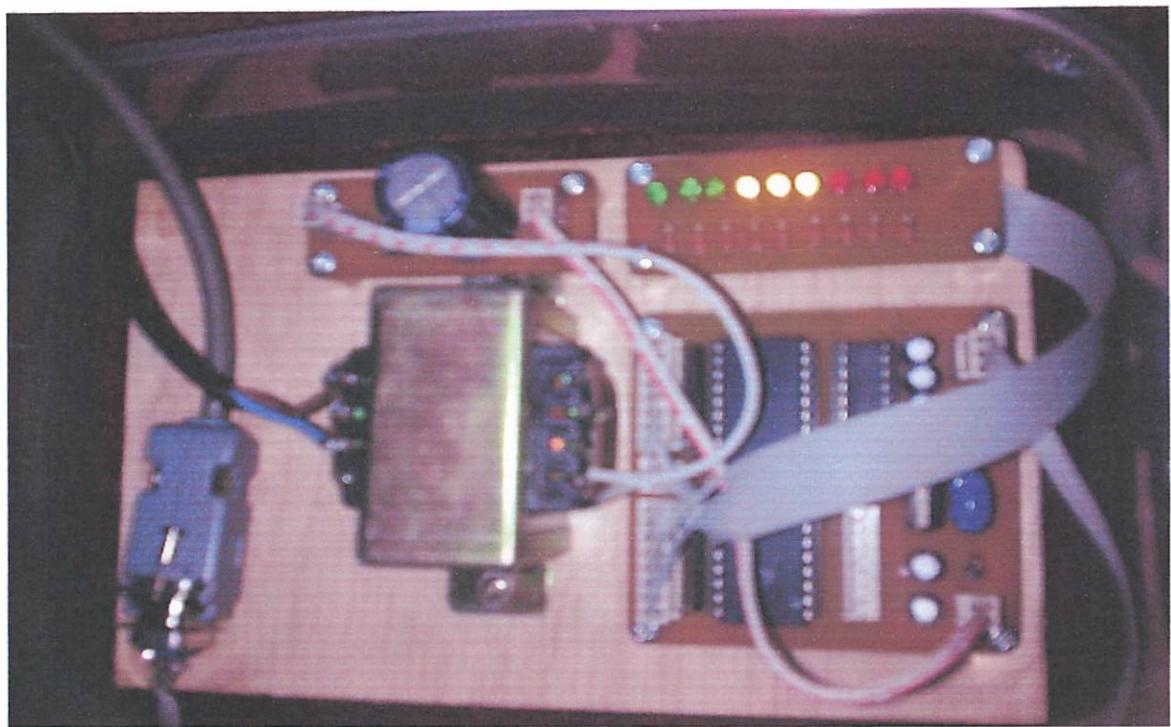
-RUANG3-

UPDATE
AC
LAMPU
PINTU
DO IT
END
RESET

CONNECT
DISCONNECT
SETTING
CLOSE



TAMPILAN ALAT



SKEMATIK RANGKAIAN

