

**PERANCANGAN DAN PEMBUATAN SYSTEM
INFORMASI NILAI AKADEMIK DENGAN
MENGUNAKAN RFID**

SKRIPSI

*Disusun dan Dajukan Sebagai Salah Satu Syarat Untuk Memperoleh
Gelar Sarjana Teknik Elektronika Strata Satu (S-1)*

Disusun oleh:

HENDRAS PRASETYO

NIM : 00.17.275



**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2009**

LEMBAR PENGESAHAN
PERANCANGAN DAN PEMBUATAN SYSTEM INFORMASI
NILAI AKADEMIK DENGAN MENGGUNAKAN RFID


SKRIPSI

Disusun dan diajukan sebagai salah satu syarat untuk memperoleh
Gelar Sarjana

Disusun Oleh :
HENDRAS PRASETYA
NIM : 00.17.275

Menyetujui

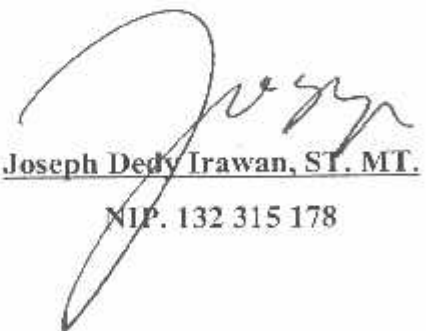
Ketua Jurusan Elektro S-1



Ir. F. Yudi Limpraptono, MT.

NIP.Y 1039500274

Dosen Pembimbing



Joseph Dedy Irawan, ST. MT.

NIP. 132 315 178

**JURUSAN TEKNIK ELEKTRO S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2009**



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

BERITA ACARA UJIAN TUGAS AKHIR
FAKULTAS TEKNOLOGI INDUSTRI

NAMA : HENDRAS PRASETYO
NIM : 00.17.275
JURUSAN : TEKNIK ELEKTRO S-1
KONSENTRASI : ELEKTRONIKA
JUDUL : PERANCANGAN DAN PEMBUATAN SYSTEM INFORMASI
NILAI :
AKADEMIK DENGAN MENGGUNAKAN RFID

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

HARI : Selasa
TANGGAL : 06 OKTOBER 2009
NILAI : B+

Panitia Ujian Skripsi



Ketua Majelis Penguji
Ir. H. Sidik Noertjahjono, MT
NIP.Y.1028700163

Sekretaris Majelis Penguji



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

Anggota Penguji

Penguji I



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

Penguji II



Komang Somawirata ST, MT
NIP.P.1030100367

ABSTRAKSI

PERANCANGAN DAN PEMBUATAN SYSTEM INFORMASI NILAI AKADEMIK DENGAN MENGGUNAKAN RFID

Disusun Oleh : Hendras Prasetyo
Dosen Pembimbing : Josep Dedy Irawan, ST,MT.
Kata kunci : RFID

ERA biometric memang kini mulai jadi trend. Pilihan kian beragam, mulai dari sidik jari, pola wajah, pola suara hingga lapisan iris mata. Pemakaiannya sudah meluas ke berbagai hal, khususnya wilayah – wilayah yang sensitive terhadap factor keamanan. Bahkan, kini sudah mulai dicoba penerapan biometric dengan cara memasukkan piranti chip elektronika berukuran mikro ke dalam tubuh sebagai pengenal identitas pribadi.

Ini memang bukan mimpi. Sudah ada beberapa orang yang mencoba menanam (mengimplantasikan) piranti tersebut kedalam tubuhnya. Salah satunya adalah Amal Graafsta yang tinggal di Wasington DC Amerika Serikat. Kisah implementasikan dimuat dalam majalah IEEE Spectrum edisi maret 2007. Sebagai impiannya digunakan piranti berbasis “ Radio Frekuensi Identification “ atau lebih dikenal RFID.

RFID (Radio Frekuensi Identification) atau identifikasi frekuensi radio adalah sebuah metode identifikasi dengan menggunakan sarana yang disebut label RFID atau transponder untuk menyimpan dan mengambil data jarak jauh. Label atau kartu RFID adalah sebuah benda yang bisa dipasang atau dimasukkan didalam sebuah produk hewan, atau bahkan manusia dengan tujuan untuk identifikasi menggunakan gelombang radio. Label RFID terdiri atas microchip silicon dan antena. Label yang pasif tidak membutuhkan sumber tenaga, sedangkan label yang aktif membutuhkan sumber tenaga untuk dapat berfungsi. Seiring dengan perkembangan teknologi mengalami kemajuan yang sangat pesat, menyebabkan manusia tidak akan lepas dari penggunaan berbagai macam peralatan elektronik yang ada, baik itu penggunaan peralatan yang berupa perangkat keras maupun perangkat lunak.

Untuk mereka yang bergerak di bidang manufaktur, logistik, pergudangan, pasar swalayan, pelayanan keamanan, dll ada teknologi yang mungkin akan segera digunakan secara besar-besaran. Teknologi ini dinamakan Radio Frequency Identification (RFID).

Juga mereka yang terlibat dalam pembuatan surat-surat identitas, paspor, dan lain-lain. Secara sederhana dapat dijelaskan bahwa teknologi ini akan tampak dalam bentuk label, stiker, kartu, gantungan kunci, koin, gelang, atau bentuk-bentuk tag lainnya. Hanya saja, tag atau kartu ini dapat memancarkan gelombang radio berupa kode (ID dan informasi lainnya).

Banyak perusahaan yang beralih menggunakan perangkat lunak dalam sistem absensi karyawannya. Dengan adanya kartu RFID maka bisa juga dibuat sebagai kartu identitas karyawan sekaligus sebagai alat untuk absen, sehingga absen para karyawan dapat dilakukan dengan mudah dan singkat, data dari absen tersebut bisa diolah sesuai kebutuhan.

ITN Malang merupakan sebuah instansi pendidikan yang memiliki Karyawan dan Dosen yang tentunya akan sangat memerlukan adanya sebuah absensi dengan menggunakan RFID yang dapat digunakan sebagai sarana informasi akademik. Informasi yang nantinya akan diolah bisa digunakan untuk dilhat siapa saja yang aktif masuk serta jam berapa. Penerapan ilmu pengetahuan dan teknologi sering menimbulkan kontroversi. Kontroversi yang sering terdengar untuk penggunaan RFID adalah masalah pelanggaran terhadap kebebasan pribadi. Contoh yang terjadi saat ini adalah dipertimbangkannya pembatalan pilot project penggunaan RFID pada paspor di Amerika Serikat. Alasan penolakan RFID adalah bahwa dengan adanya RFID di paspor, pembawa dengan mudah dapat dipantau di mana-mana (dengan asumsi bahwa nantinya di mana-mana akan terpasang RFID reader).

Kata Pengantar

Alhamdulillah penulis panjatkan kehadirat Ilahi Robbi atas curahan nikmat dan karunia-Nya serta shalawat dan salam semoga tetap terlimpah pada Rasulullah Muhammad beserta keluarga dan para sahabatnya atas terselesaikannya laporan skripsi ini. Laporan skripsi ini disusun sebagai salah satu prasyarat untuk memperoleh gelar sarjana di Fakultas Industri, Teknik Elektronika, ITN Malang

Atas terselesaikannya laporan Skripsi ini, penulis mengucapkan banyak terimakasih kepada :

- ❖ Kedua orang tuaku atas curahan kasih sayang, kesabaran mendidiku, support dan perhatiannya baik secara moril dan materil.
- ❖ My Wife atas curahan kasih sayang, kesabaran, support dan perhatiannya baik secara moril dan materil.
- ❖ Joseph Dedy Irawan, ST, MT selaku dosen pembimbing I atas kesabaran dan bantuannya saat membimbing.
- ❖ I Komang Somawirata, ST, MT selaku dosen pembimbing II atas kesabaran dan bantuannya saat membimbing.
- ❖ Saudaraku dan Sahabat-sahabatku ELKA '00 yang tersayang atas kasih sayang dan perhatiannya.
- ❖ Semua pihak yang telah membantu terselesaikannya laporan ini.

Penulis berharap semoga melalui laporan ini, dapat memberikan manfaat bagi civitas akademika fakultas industri pada khususnya dan civitas akademika institut teknologi nasional Malang pada umumnya. Akhirnya penulis menyadari bahwa segala kekurangan yang melekat pada laporan ini. Untuk itu saran dan kritik dari para pembaca merupakan hal yang saya harapkan. Semoga segala ikhtiyar saya diridhoi oleh Allah S. W. T. Amin ya Robbal 'alamin.

Malang, Mei 2009

Penulis

DAFTAR ISI

LEMBAR PENGESAHAN	i
ABSTRAKSI	ii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL	ix

BAB I PENDAHULUAN

1.1. Latar Belakang	1
1.2. Tujuan	2
1.3. Rumusan Masalah	2
1.4. Batasan Masalah	3
1.5. Metodologi	3
1.6. Sistematika	4

BAB II LANDASAN TEORI

2.1. Pendahuluan	5
2.2. RFID Starter Kit ID-20	5
2.3. Borland Delphi 7.0	11
2.3.1. IDE Delphi	11
2.3.2. Menu Borland Delphi	14
2.3.3. Komponen Visual dan Non Visual	16
2.3.4. Informasi Sintak Umum	18

2.3.4.1. Komentar	18
2.3.4.2. Type Data	19
2.4. Dasar Basis Data	29
2.4.1. Konsep Basis Data	29
2.4.2. Definisi	29
2.4.3. Operasi Basis Data	31
2.5. Microsoft Access	32
2.5.1. Keunggulan Microsoft Access	33
2.5.2. Database didalam MS Access	33
2.6 RS-232	35

BAB III PERANCANGAN DAN PEMBUATAN ALAT

3.1. Pendahuluan	37
3.1.1. Blok Diagram Keseluruhan	37
3.2. Prinsip Kerja Alat	38
3.3. Komunikasi Data antara RFID Tag dengan Komputer	40
3.4. Flowchart	43
3.5. Perancangan Perangkat Lunak (<i>Software</i>)	44
3.5.1. Program Aplikasi Komputer	44
3.5.2. Perancangan Pembuatan Software Menggunakan Bahasa Pemrograman Borland Delphi7 dan MS Access	45
3.5.2.1. Penyusunan Desain Software	45

BAB IV PENGUJIAN RANGKAIAN DAN ANALISA DATA

4.1. Pengujian Penerimaan Data dari Database	56
4.1.1. Tujuan	56
4.1.2. Peralatan yang digunakan	56
4.1.3. Langkah Pengujian	56
4.1.4. Hasil Pengujian	57

BAB V PENUTUP

5.1 Kesimpulan	62
5.2. Saran	62

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2.1. Fisik Reader ID-12	6
Gambar 2.2. Timing Diagram	10
Gambar 2.3. Lembar Kerja Delphi	12
Gambar 2.4. Database dalam MS Access	34
Gambar 2.5. Object Tabel	34
Gambar 2.6. Konfigurasi Pin Konektor RS-232	35
Gambar 3.1. Blok Diagram RFID Starter Kit	37
Gambar 3.2. Rangkaian RFID	39
Gambar 3.3. Connection Hyper-Terminal	40
Gambar 3.4. Conect Using	40
Gambar 3.5. Serial Port Parameter	41
Gambar 3.6. Tampilan Kode Tag RFID pada Hyper_terminal	41
Gambar 3.7. Flowhart	42
Gambar 3.8. Tampilan Utama MS Access	45
Gambar 3.9. Blank Database	46
Gambar 3.10 Gambar New Database	46
Gambar 3.11. Membuat tabel database	47
Gambar 3.12. Struktur Tabel	48
Gambar 3.13. Save Tabel	48
Gambar 3.14. Menyusun Tampilan Design Menu Utama	49
Gambar 3.15. Menu User	50
Gambar 3.16. Menu Pilihan	51
Gambar 3.17. Menu Data Mahasiswa	52

Gambar 3.18. Menu Data Nilai	54
Gambar 4.1. Menu Utama Program	57
Gambar 4.2. Menu	58
Gambar 4.3. Menu User.....	59
Gambar 4.4. Lembar Admin	60
Gambar 4.5. Tampilan Menu Setting	60

DAFTAR TABEL

Tabel 2.1. Tabel Pin Deskription & Output Data Format	7
Tabel 2.2. Tabel Karakteristik Fisik	7
Tabel 2.3 Macam-macam Komponen Standart Delphi	16
Tabel 2.4. Tabel Tipe Data Integer	20
Tabel 2.5. Tabel Tipe Data Real	21
Tabel 2.6. Tabel Tipe Data Boolean	22
Tabel 2.7. Tabel Tipe Data Character	22
Tabel 2.8. Tabel Tipe Data String	23
Tabel 2.9. Tabel operator aritmatika pada Delphi	25
Tabel 2.10 Tabel operator relasi pada Delphi	26
Tabel 2.11. Tabel operator logika pada Delphi	27
Tabel 2.12. Fungsi PIN dalam DB-9	35
Tabel 2.13. Spesifikasi RS 232	36
Tabel 3.1. Merubah isi Properti pada Layar Utama	49
Tabel 3.2. Merubah isi Properti pada Layar Biodata Karyawan	50
Tabel 3.3. Merubah isi Properti pada menu user	52
Tabel 3.4. Merubah isi Properti pada menu data mahasiswa.....	53
Tabel 3.5. Merubah isi Properti pada menu data nilai.....	54
Tabel 4.1. Jarak sinyal yang diterima oleh RFID	61

BAB I PENDAHULUAN

1.1 Latar Belakang

Sciring dengan perkembangan teknologi yang mengalami kemajuan yang sangat pesat, menyebabkan manusia tidak akan lepas menggunakan berbagai macam peralatan elektronik yang ada, baik itu menggunakan peralatan yang berupa perangkat keras maupun perangkat lunak. Manusia semakin dituntut untuk melakukan segala sesuatu bukan hanya secara tepat, tetapi juga harus cepat serta efisien. Untuk itu manusia senantiasa mencari akal guna mempermudah pekerjaan sehari-hari.

Penggunaan teknologi Radio Frequency Identification (RFID) bukanlah hal yang baru dalam bidang jasa (seperti pendidikan, perhubungan, dan industri jasa lainnya), pada beberapa industri di Indonesia telah menggunakannya. Secara sederhana dapat dijelaskan bahwa teknologi ini akan tampak dalam bentuk label, stiker, kartu, gantungan kunci, koin, gelang, atau bentuk-bentuk tag lainnya. Hanya saja, tag atau kartu ini dapat memancarkan gelombang radio berupa kode (ID dan informasi lainnya).

Banyak perguruan tinggi yang beralih menggunakan perangkat lunak dalam sistem informasi nilai akademik. Dengan adanya kartu RFID maka bisa juga dibuat sebagai kartu identitas karyawan sekaligus sebagai alat untuk informasi nilai akademik, sehingga para mahasiswa dapat dilakukan dengan mudah dan singkat.

ITN Malang merupakan sebuah instansi pendidikan yang memiliki Karyawan dan Dosen yang tentunya akan sangat memerlukan adanya terobosan baru dengan menggunakan RFID yang dapat digunakan sebagai sarana informasi nilai akademik. Informasi yang nantinya akan diolah bisa digunakan untuk dilihat siapa saja yang aktif masuk serta jam berapa. Penerapan ilmu pengetahuan dan teknologi sering menimbulkan kontroversi. Kontroversi yang sering terdengar untuk penggunaan RFID adalah masalah pelanggaran terhadap kebebasan pribadi. Contoh yang terjadi saat ini adalah dipertimbangkannya pembatalan pilot project penggunaan RFID pada paspor di Amerika Serikat. Alasan penolakan RFID adalah bahwa dengan adanya RFID di paspor, pembawa dengan mudah dapat dipantau di mana-mana (dengan asumsi bahwa nantinya di mana-mana akan terpasang RFID reader).

1.2 Tujuan

Adapun tujuan dari pembuatan RFID ini adalah untuk membuat kartu identitas yang dapat digunakan melihat nilai akademik mahasiswa dengan hanya mendekatkan saja kartu identitas tersebut ke reader.

1.3 Rumusan Masalah

Berdasarkan Latar Belakang yang telah diuraikan pada bagian sebelumnya, maka akan didapat beberapa masalah yang akan ditemui dalam merancang dan membuat software sehingga dapat mengoperasikan *RFID* untuk melihat informasi nilai akademik.

1.4 Batasan Masalah

Dalam menyusun tugas akhir ini diperlukan suatu batasan masalah agar tidak menyimpang dari ruang lingkup yang akan dibahas. Adapun batasan masalahnya adalah sebagai berikut:

- a) RFID yang digunakan ID-12
- b) Merancang database menggunakan *Microsoft Access*.
- c) Membuat software untuk membaca kartu menggunakan *Borland Delphi-7*.

1.5 Metodologi

Metodologi yang dipakai dalam pembuatan TA ini adalah:

1. Studi Literatur

Dengan mencari referensi-referensi yang berhubungan dengan perencanaan pembuatan perangkat yang akan dibuat.

2. Field Research

Dengan melakukan penelitian secara langsung mengenai objek-objek yang berhubungan langsung dengan perencanaan perangkat yang akan dibuat.

3. Design dan Pembuatan Perangkat

Yaitu meliputi pembuatan Software pada program borland Delphi-7 serta pada Microsoft Access.

4. Pengujian Perangkat

Dengan melakukan pengujian perblok perangkat dan kerja seluruh sistem pada perangkat tersebut.

BAB II

LANDASAN TEORI

2.1 Pendahuluan

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

- RFID Starter Kit ID-20
- Borland Delphi-7
- Microsoft Access
- RS-232

2.2 RFID Starter Kit ID-20

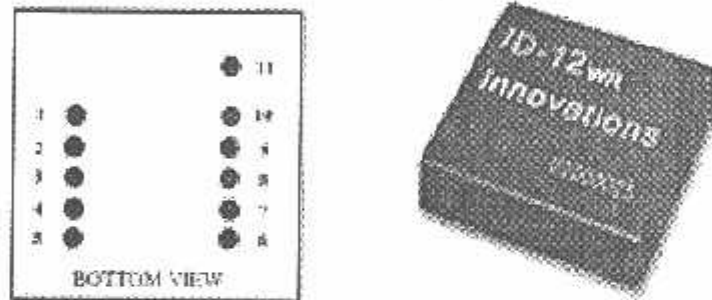
RFID Starter Kit merupakan suatu sarana pengembangan RFID berbasis reader tipe ID-12 yang telah dilengkapi dengan jalur komunikasi RS-232 serta indikator buzzer dan LED. Modul ini dapat digunakan dalam aplikasi mesin absensi RFID, RFID access controller, dsb.

Spesifikasi

1. Berbasis RFID reader ID-12 dengan frekuensi kerja 125 kHz untuk kartu performat EM4001/sejenis dan memiliki jarak baca maksimal 12 cm.
 2. Kompatibel dengan varian RFID reader lainnya, antara lain: ID-2, ID-10, dan ID-20.
 3. Mendukung Varian RFID reader/writer, antara lain : ID-2RW, ID-12RW, dan ID-20RW.
-

4. Mendukung varian RFID reader/writer, antara lain: ID-2RW, ID-12RW, dan ID-20RW.
5. Mendukung format data ASCII (UART TTL/RS-232), Wiegand26, maupun Magnetic ABA Track2 (Magnet Emulation).
6. Dilengkapi dengan buzzer sebagai indikator baca, serta LED sebagai indikator tulis.
7. Tersedia jalur komunikasi serial UART RS-232 dengan konektor RJ11.
8. Tegangan input catu daya 9 - 12 VDC (J2).

Susunan pin pada reader ID-12 adalah sebagai berikut:



Gambar 2.1 Fisik Reader ID-12

(www.google.com ID_SERIES_DATASHEET_2005)

Tabel 2.1 Tabel *Pin Description & Output Data Formats*

Pin No.	Description	ASCH	Magnet Emulation	Wiegand26
Pin 1	Zero Volts and Tuning Capacitor Ground	GND 0V	GND 0V	GND 0V
Pin 2	Strap to +5V	Reset Bar	Reset Bar	Reset Bar
Pin 3	To External Antenna and Tuning Capacitor	Antenna	Antenna	Antenna
Pin 4	To External Antenna	Antenna	Antenna	Antenna
Pin 5	Card Present	No function	Card Present *	No function
Pin 6	Future	Future	Future	Future
Pin 7	Format Selector (+/-)	Strap to GND	Strap to Pin 10	Strap to +5V
Pin 8	Data 1	CMOS	Clock *	One Output *
Pin 9	Data 0	TTL Data (inverted)	Data *	Zero Output *
Pin 10	3.1 kHz Logic	Beeper / LED	Beeper / LED	Beeper / LED
Pin 11	DC Voltage Supply	+5V	+5V	+5V

* Requires 4K7 Pull-up resistor to +5V

Karakteristik Fisik

Tabel 2.2 Karakteristik fisik

Parameters	ID-2	ID-12	ID-20
Read Range	N/A (no internal antenna)	12+ cm	16+ cm
Dimensions	21 mm x 19 mm x 6 mm	26 mm x 25 mm x 7 mm	40 mm x 40 mm x 9 mm
Frequency	125 kHz	125 kHz	125 kHz
Card Format	EM 4001 or compatible	EM 4001 or compatible	EM 4001 or compatible

Encoding	Manchester 64-bit, modulus 64	Manchester 64-bit, modulus 64	Manchester 64-bit, modulus 64
Power Requirement	5 VDC @ 13mA nominal	5 VDC @ 30mA nominal	5 VDC @ 65mA nominal
I/O Output Current	+/-200mA PK	-	-
Voltage Supply Range	+4.6V through +5.4V	+4.6V through +5.4V	+4.6V through +5.4V

Ada dua macam RFID, yaitu RFID aktif dan RFID pasif. RFID aktif terdiri dari suatu rangkaian chip untuk menyimpan identitas dan informasi lainnya, pemancar, antena, dan baterai. RFID aktif memancarkan sinyal dengan tenaga dari baterai.

Pada umumnya RFID tidak memancarkan sinyal terus-menerus. Untuk menghemat baterai, RFID hanya akan memancarkan sinyalnya apabila ada sinyal pemicu yang sesuai dengan tata cara pengiriman dan penerimaannya (protokol). Sinyal pemicu ini biasanya ditempatkan menjadi satu pada alat pemancar/penerima (reader dan antena).

Secara singkat dapat dijelaskan bahwa sebenarnya RFID tag dan reader/antena keduanya merupakan transceiver (transmitter-receiver). Jarak jangkauan RFID aktif ini ada yang menjanjikan dapat sampai 100 meter. Bentuk RFID aktif umumnya mempunyai ketebalan beberapa milimeter untuk tempat batraainya.

Sedangkan ukurannya bervariasi, ada yang sebesar uang logam Rp 1.000, ada yang berupa gantungan kunci, ada yang berupa kartu nama, dan lain-lain. RFID aktif ini umumnya digunakan untuk sistem keamanan (mobil, barang-barang berharga), pembayaran tol dan parkir otomatis, lokasi keberadaan

(misalnya keberadaan dokter, perawat dan pasien di dalam lingkungan rumah sakit), dan lain-lain.

RFID pasif tidak mempunyai baterai. Sinyal dikirim oleh reader/antena (transceiver) diterima oleh RFID tag, kemudian rangkaian dalam tag dengan menggunakan energi sinyal tadi mengirim data ke antena dan reader kembali. Oleh karena itu sinyalnya sangat lemah. Paling jauh hanya sekitar tiga meter.

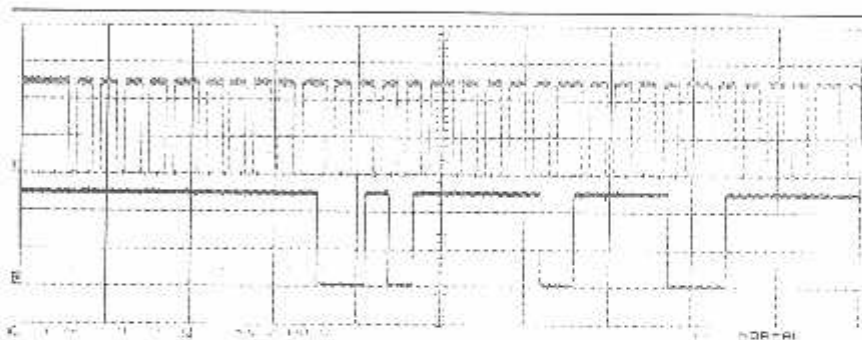
Banyak aplikasi untuk RFID pasif ini yang efektif berjalan di bawah 50 cm. Bentuk RFID pasif ini sebagian besar berupa label, baik label yang digantung pada barang-barang seperti pakaian atau label yang ditempel di kotak pembungkus/karton. Ukurannya bermacam-macam. Yang umum dipakai adalah 10 x 15 cm dan 1,5 x 3 inci.

RFID ada yang menggunakan frekuensi 125-134 KHz, biasanya untuk label barang, kartu-kartu untuk akses masuk ke suatu tempat. Frekuensi 13.56 Mhz untuk pekerjaan-pekerjaan yang memerlukan kecepatan lebih tinggi dan jarak lebih jauh, misalnya untuk tol, parkir, sortir surat di kantor pos, dan bagasi pesawat.

RFID menggunakan frekuensi radio untuk membaca informasi dari sebuah devais kecil yang disebut tag atau *transponder* (*Transmitter + Responder*). Pada sistem RFID umumnya, tag atau *transponder* ditempelkan pada suatu obyek. Setiap tag dapat membawa informasi yang unik, diantaranya: serial number, model, warna, tempat perakitan, dan data lain dari obyek tersebut.

Sistem RFID terdiri dari empat komponen, diantaranya seperti berikut :

- Tag : Ini adalah devais yang menyimpan informasi untuk identifikasi objek. Tag RFID sering juga diseebut sebagai *transponder*. Format dari Tag pada perencangan ini adalah EM4001 atau tag kompatibel lainnya.
- Antena : Untuk menstransmisikan sinyal frekuensi radio antara pembaca RFID dengan tag RFID.
- Pembaca RFID : Adalah devais yang kompatibel dengan tag RFID yang akan berkomunikasi secara *wireless* dengan tag. Digunakan ID-12 sebagai RFID reader pada perancangan ini.



Gambar 2.2 Timing Diagram

(www.google.com ID2_12_20_RW.pdf)

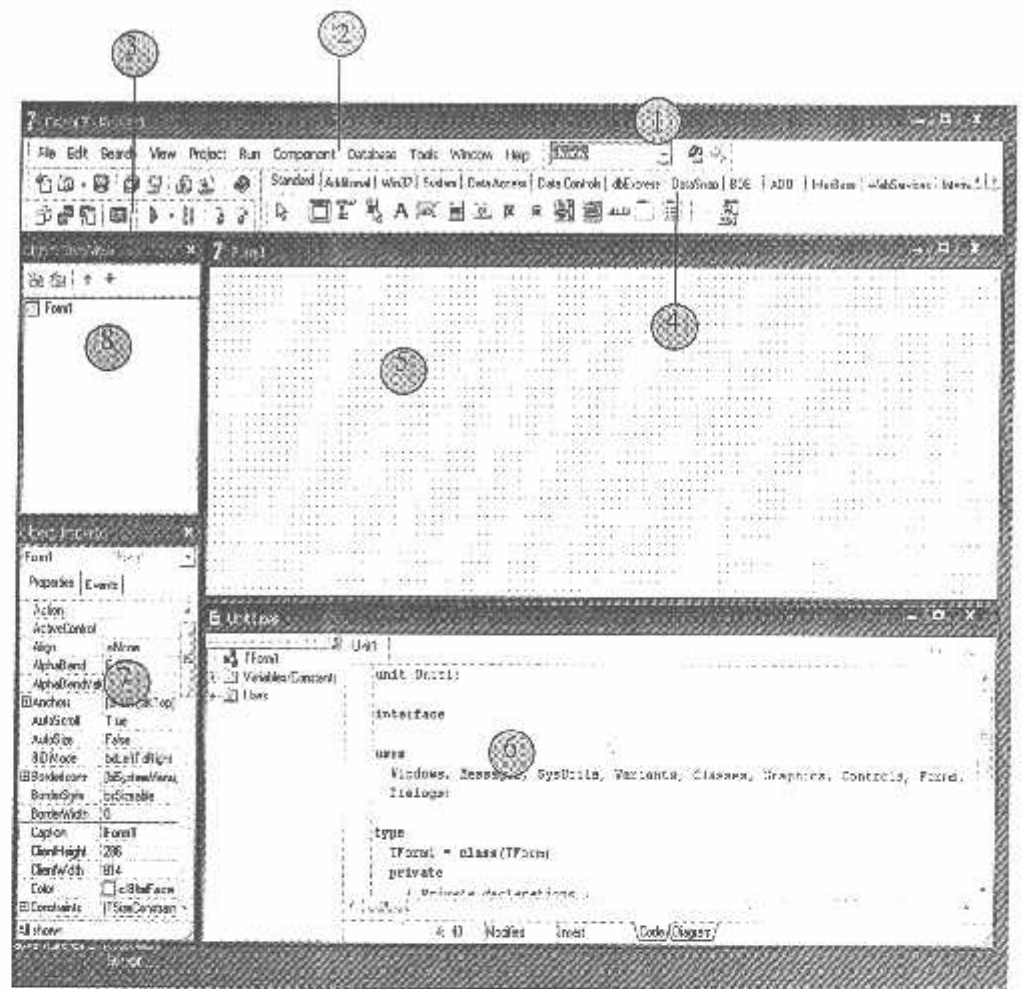
2.3. Borland Delphi 7.0

Delphi adalah salah satu pemrograman visual dilingkungan sitem operasi Microsoft Windows, berbasis bahasa PASCAL, sehingga bila kita telah mengetahui konsep dasar dan aturan-aturan yang berlaku dalam pemrograman PASCAL maka untuk masuk ke Delphi akan lebih mudah. Untuk mempermudah pemrograman dalam membuat suatu program aplikasi, Delphi menyediakan fasilitas pemrograman yang sangat lengkap. Fasilitas pemrograman tersebut dibagi dalam du kelompok, yaitu object dan bahasa pemrograman.

Secara ringkas, object adalah suatu komponen yang mempunyai bentuk fisik, object biasanya dipakai untuk melakukan tugas tertentu dan mempunyai batasan-batasan tertentu. Sedangkan bahasa pemrograman dapat disebut sebagai sekumpulan teks yang mempunyai arti tertentu dan disusun dengan aturan tertentu serta untuk menjalankan tugas tertentu.

2.3.1. IDE Delphi

Lingkungan pengembangan terpadu atau *Integrated Development Environment* (IDE) dalam program Delphi terbagi menjadi enam bagian utama yaitu *Main Window*, *ToolBar*, *Componen Palette*, *Form Designer*, *Code Editor* dan *Object Inspector*. Untuk lebih jelasnya terlihat pada gambar dibawah ini:



Gambar 2.3. Lembar Kerja Delphi

(Delphi ForAccounting hal 6)

IDE merupakan sebuah lingkungan dimana semua tombol perintah yang diperlukan untuk mendesain aplikasi, menjalankan dan menguji sebuah aplikasi disajikan dengan baik untuk memudahkan mengembangkan program.

Jendela IDE Delphi 7.0 mempunyai perangkat-perangkat yang dapat dipergunakan untuk memudahkan seseorang programmer dalam membuat program. Perangkat-perangkat tersebut seperti terlihat pada gambar 2.3 diantaranya adalah:

1. *Main Window*

Jendela Utama ini adalah bagian dari IDE yang mempunyai fungsi yang sama dengan semua fungsi utama dari program aplikasi Windows lainnya. Jendela utama Delphi terbagi menjadi tiga bagian, diantaranya Main Menu, Toolbar, dan Componen Pallete.

2. *Main Menu*

Menu Utama pada Delphi memiliki kegunaan yang sama seperti program aplikasi windows lainnya. Dengan menggunakan fasilitas menu, dapat memanggil atau menyimpan program. Pada dasarnya semua perintah yang diberikan dapat ditemukan pada bagian menu utama.

3. *Toolbar*

Dengan *toolbar* dapat melakukan beberapa operasi pada menu utamadengan sebuah klik tunggal. Setiap tombol pada *toolbar* mempunyai sebuah *tooltip* yang berisi informasi mengenai fungsi dari tombol tersebut.

4. *Component Palette*

Component Palette merupakan bagian yang digunakan untuk meletakkan berbagai komponen yang sesuai dengan kategorinya. Komponen tersebut berguna untuk mendesain user interface (antarmuka pemakai)dari program yang sedang dibuat.

5. *Form Designer*

Merupakan sebuah bidang jendela yang masih kosong. Ketika menjalankan Borland Delphi 7, secara otomatis *Form Designer* akan memanggil sebuah form yang bernama Form1.

6. *Code Editor*

Pada bidang ini dapat menuliskan kode-kode program dan logika program dan bahasa Delphi untuk mengatur jalannya program. Antara Form Designer dan Code Editor merupakan dua bagian yang berkaitan.

7. *Object Inspector*

Dengan perangkat ini dapat mengubah *property* dan *event* pada setiap object atau komponen. Object atau komponen yang satu dengan yang lain mempunyai *property* dan *event* yang berbeda. Jika menempatkan sebuah komponen pada suatu Form, komponen tersebut akan berisi nilai *default* dari Delphi. Misalnya komponen Label akan berisi nilai 13 dan 32 untuk *property height* dan *width*. Nilai *property* tersebut dapat diubah baik pada saat perancangan interface program maupun pada saat program berjalan dengan menggunakan kode program.

8. *Object Tree View*

Merupakan sebuah diagram pohon yang menggambarkan hubungan logis antara komponen visual dan non visual yang terletak pada form, data module atau frame. Semua object yang anda pakai pada form, data module atau frame akan muncul pada *object Tree View*.

2.3.2. Menu Borland Delphi

1. *Menu File*

Berisi Perintah dasar menu yang sering digunakan yang berhubungan dengan pengoperasian file.

2. *Menu Edit*

Berisi perintah yang digunakan untuk menyunting teks program dalam jendela code editor, menyunting komponen-komponen yang terletak pada bagian form design dan beberapa item lainnya.

3. *Menu Search*

Berisi perintah yang digunakan untuk menyunting teks program dalam jendela code editor, menyunting komponen-komponen yang terletak pada bagian form designer dan beberapa item lainnya.

4. *Menu View*

Berisi tentang perintah yang digunakan untuk menampilkan atau menyembunyikan jendela-jendela tertentu dalam program Delphi.

5. *Menu Project*

Berisi tentang perintah yang berhubungan dengan pengolahan suatu project.

6. *Menu Run*

Digunakan untuk menjalankan program dan melihat jalannya program. Dan dapat memantau jalannya program dengan memperhatikan procedure yang dijalankan.

7. *Menu Component*

Digunakan untuk menambah atau menginstal komponen baru.

8. *Menu Database*

Digunakan untuk membuat, mengubah atau melihat database.

9. *Menu Tools*

Digunakan untuk mengubah option atau memanggil database desktop dari menu Delphi.

10. *Menu Help*

Menu Help digunakan untuk membuka lembaran kerja bantu Delphi.

2.3.3. **Komponen Visual dan Nonvisual**

Visual Component Library (VCL) adalah Library Delphi dalam lingkungan windows. Delphi membagi komponen-komponen ke dalam dua kelompok besar berdasarkan visualisasinya, yaitu:



1. **Komponen Visual**











Yaitu komponen yang akan ditampilkan dilayar pada saat aplikasi dijalankan. Komponen visual dipakai untuk membangun antarmuka dengan pemakai. Komponen ini dapat diubah ukurannya dengan cara drag pada salah satu kotak kecil yang ada disekeliling komponen. Contoh komponen visual adalah Button, Edit, Memo, Label, dan lain-lain.






2. **Komponen Nonvisual**

Yaitu komponen yang tidak ditampilkan dilayar pada saat aplikasi dijalankan, tetapi diperlukan dalam aplikasi. Ukuran komponen ini tidak dapat diubah.

Tabel 2.3. *Macam-macam Komponen Standar Delphi*

Komponen	Fungsi
 Pointer	Pointer adalah komponen khusus dan terdapat disetiap tab dalam component palatte. Komponenpointer adalah komponen select yang digunakan untuk memilih komponen dalam form designer.
 Frames	Sebuah kontainer yang digunakan untuk menampung komponen frame dapat diletakkan dalam form atau frame-frame yang lain.

 Main Menu	Komponen yang digunakan untuk membuat menu bar dan menu drop down. Komponen ini bersifat invisible.
 PopUp Menu	Komponen yang digunakan untuk membuat menu popup yang akan muncul jika pemakai melakukan proses klik kanan. Komponen ini bersifat invisible.
 Label	Komponen ini hanya digunakan untuk menambahkan teks di dalam form
 Edit	Komponen yang digunakan untuk menerima satu baris teks yang merupakan data input pemakai. Komponen ini juga dapat digunakan untuk menampilkan teks.
 Memo	Komponen Memo dipakai untuk memasukkan atau menampilkan beberapa baris teks di dalam form.
 Button	Komponen yang dipakai untuk membuat button yang akan dipakai untuk memilih pilihan di dalam aplikasi. Jika mengklik komponen button tersebut maka suatu perintah atau kejadian akan dijalankan.
 Check Box	Komponen yang digunakan untuk memilih atau membatalkan suatu pilihan, yaitu dengan cara mengklik komponen.
 Radio Button	Komponen yang digunakan untuk memberikan sekelompok pilihan dan hanya ada satu pilihan yang dapat dipilih. Untuk memilih salah satu pilihan adalah dengan mengklik tombol pilihan yang diinginkan.
 List Box	Komponen yang digunakan untuk membuat sebuah daftar pilihan, dimana hanya ada satu pilihan yang dapat dipilih. Untuk mencari dan kemudian memilih salah satu pilihan yang terdapat di dalam daftar dapat menggunakan batang penggulung.
 Combo Box	Komponen yang digunakan untuk membuat sebuah daftar pilihan, dimana hanya ada satu pilihan yang dapat dipilih. Untuk mencari dan kemudian memilih salah satu pilihan yang terdapat di dalam daftar dapat menggunakan mengklik tombol

	drop down.
 Scroll Bar	Komponen ini mempunyai fungsi yang sama seperti batang penggulung yang terdapat dalam program-program berbasis Windows pada umumnya.
 Group Box	Sebuah kontainer yang dapat digunakan untuk mengelompokkan komponen-komponen lain seperti Radio Button, CheckBox dan komponen kontainer yang lain.
 Radio Group	Komponen ini merupakan komponen kombinasi dari GroupBox dan didesain untuk membuat sekelompok Radio Button.
 Panel	Sebuah kontainer yang dapat digunakan untuk membuat StatusBar, ToolBar dan ToolPalette.
 Action List	Komponen yang berisi daftar action yang digunakan bersama-sama dengan komponen dan control seperti item menu dan button.

2.3.4. Informasi Sintak Umum

Untuk memulai penulisan listing program pada Delphi 7, terdapat hal-hal yang perlu diperhatikan dan harus dimengerti terlebih dahulu. Sehingga dalam membuat sebuah program atau aplikasi yang telah direncanakan, dapat berjalan dengan baik. Hal-hal tersebut antara lain;

2.3.4.1. Komentar

Komentar Dipakai untuk memberikan penjelasan atau keterangan di dalam baris program. Teks yang ditulis sebagai komentar tidak akan dikompilasi oleh kompilator pada saat program aplikasi di jalankan.

Untuk menuliskan sebuah komentar, programmer dapat menggunakan salah satu tanda dari tiga bentuk tanda yang disediakan, yaitu :

- Kurung Kurawal => { Komentar Program }
- Kurung Bintang => (* Komentar Program *)
- Slash Ganda => // Komentar Program

Komentar dengan tanda kurung kurawal dan kurung bintang menggunakan tanda pembuka dan tanda penutup, sehingga teks yang dianggap sebagai komentar adalah teks yang terletak di antara tanda pembuka dan tanda penutup. Sedangkan tanda komentar slash ganda tidak menggunakan tanda penutup, sehingga semua teks yang terletak di belakan tanda slash ganda dianggap sebagai komentar.

2.3.4.2. Tipe Data

Pemilihan tipe data yang tepat akan sangat berguna dalam penghematan memori, kecepatan proses, ketelitian penghitungan dan lain-lain. Ada beberapa hal yang perlu diperhatikan dalam menentukan tipe data :

1. Penggunaan Memori

Masing-masing tipe data memiliki perbedaan dalam hal penggunaan memori. Dalam hal ini usahakan untuk menggunakan tipe data yang memiliki memori yang kecil.

2. Ketelitian Penghitungan

Walaupun unsur penghematan memori sangat penting untuk diperhatikan namun kebenaran dalam hal ketelitian penghitungan jauh lebih penting. Oleh sebab itu gunakan tipe data yang mempunyai tingkat ketelitian (presisi) paling tinggi.

Borland Delphi menyediakan banyak tipe data. Jenis maupun tipe data yang dapat digunakan antara lain adalah :

- **Tipe Integer**

Tipe data Integer digunakan untuk bilangan bulat atau bilangan yang tidak memiliki angka desimal. Tipe data Integer memiliki beberapa tipe yang tergantung pada rentang nilai dan ukuran penggunaan memori.

Tabel 2.4 Tabel Tipe Data Integer

Tipe	Rentang Nilai	Byte
Byte	0 .. 255	1
Word	0 .. 65535	2
ShortInt	-128 .. 127	2
SmallInt	-32768 .. 32767	2
Integer	-2147483648 .. 2147483648	4
Cardinal	0 .. 2147483647	4
LongInt	-2147483648 .. 2147483648	4

Semakin besar rentang nilai bilangan, semakin besar pula memori yang diperlukan.

- **Tipe Real**

Tipe data Real digunakan untuk bilangan yang memiliki desimal (ditulis dalam $a \times 10^b$). Tipe data Real memiliki beberapa tipe yang tergantung pada rentang nilai dan ukuran penggunaan memori.

Tabel 2.5 Tabel Tipe Data Real

Tipe	Rentang Nilai	Byte
Real48	$2,9 \times 10^{-38} .. 1,7 \times 10^{38}$	4
Single	$1,5 \times 10^{-45} .. 3,4 \times 10^{45}$	4
Double	$5,0 \times 10^{-324} .. 1,7 \times 10^{324}$	8
Real	$5,0 \times 10^{-324} .. 1,7 \times 10^{324}$	8
Extended	$3,6 \times 10^{-4931} .. 1,1 \times 10^{4931}$	10
Comp	$-2^{63+1} .. 2^{63-1}$	4
Currency	-922337203685477,5808 .. 922337203685477,5807	4

Tipe Double ekuivalen dengan tipe Real yang umum dipakai. Tipe Real48 digunakan untuk menjaga kompatibilitas dengan versi sebelumnya. Tipe Real48 menghasilkan proses yang lebih lambat. Tipe Extended menghasilkan ketelitian yang lebih baik tetapi kurang kompatibilitas jika program nantinya didesain untuk sharing dengan platform lain. Tipe Comp adalah tipe asli yang digunakan CPU Intel yang sebetulnya mewakili tipe Integer 64 bit, tetapi dikelompokkan dalam tipe Real karena tidak mempunyai sifat ordinal. Comp dipakai untuk mempunyai kompatibilitas dengan versi sebelumnya. Sebaiknya gunakan Int64 untuk kerja yang lebih baik. Keuntungan dari penggunaan tipe data Currency adalah :

- ❖ Tipe data Currency mempunyai ketelitian yang lebih tinggi dalam menangani bilangan yang cukup besar.
- ❖ Tipe data Currency dipakai dalam CurrencyField dan komponen lain dan kompatibel dengan tipe database yang menyatakan uang.

- **Tipe Boolean**

Tipe data Boolean digunakan untuk data logika yang hanya berisi True (Benar) dan False (Salah). Tipe data Boolean yang dapat digunakan antara lain :

Tabel 2.6 Tabel Tipe Data Boolean

Tipe	Byte
Boolean	1
ByteBool	1
WordBool	2
LongBool	4

Dari beberapa tipe yang ada disarankan untuk menggunakan tipe Boolean, sedangkan untuk tipe lain hanya untuk menjaga kompatibilitas dengan program lain yang menggunakan tipe yang sama. Variabel tipe data Boolean dapat menerima penggunaan operator logika AND, OR, dan NOT.

- **Tipe Character**

Tipe data Character digunakan untuk menyatakan sederetan karakter satu huruf dalam bentuk tiga tipe, yaitu :

Tabel 2.7 Tabel Tipe Data Character

Tipe	Rentang Nilai	Byte
Char	1 Karakter ANSI	1
AnsiChar	1 Karakter ANSI	1
WideChar	1 Karakter Unicode	2

- **Tipe String**

Tipe data string digunakan untuk menyatakan sederetan karakter, misalnya nama, alamat, kota, dan lain-lain. Adapun tipe-tipe dari data string, yaitu :

Tabel 2.8 Tabel Tipe Data String

Tipe	Rentang Nilai	Byte
ShortString	2 - 256	256 Karakter
AnsiString	4 - 2 GB	2 ³¹ Karakter
String	255 - 3 GB	2 ³¹ Karakter
WideString	4 - 2 GB	2 ³¹ Karakter

Tipe ShortString berfungsi untuk menyesuaikan kompatibilitas dengan versi sebelumnya, sedangkan AnsiString dan WideString dapat digunakan untuk menyimpan karakter Unicode. Variabel dengan tipe data string mampu menangani data String yang hampir tidak terbatas (3 GB).

3. Variabel

Variabel adalah suatu tempat yang dialokasikan dalam memori yang diberi nama (sebagai pengenal) untuk menampung suatu data. Pendeklarasian variabel mempunyai susunan sebagai berikut :

```
var
nama_variabel : tipe_variabel;
```

Contoh:

```
var
a : Integer;
```

Contoh diatas mendeklarasikan variabel a dengan tipe Integer. Untuk mendeklarasikan beberapa variabel, perintah var cukup dituliskan hanya satu kali.

Contoh :

```
var
a : Integer;
b : String;
c : Real;
```

Contoh diatas mendeklarasikan variabel a, b dan c yang masing-masing bertipe Integer, String dan Real. Untuk mendeklarasikan beberapa variabel dengan tipe yang sama, tipe datanya cukup dituliskan hanya satu kali.

Contoh :

```
var
a, b, c : Integer;
x, y, z : String;
```

4. Operator

Dalam melaksanakan proses pengolahan data, Delphi menyediakan berbagai operator dengan urutan atau derajat proses pelaksanaan yang berbeda untuk beberapa operator yang dilibatkan pada suatu proses. Berikut ini adalah derajat proses pengolahan data yang dimiliki delphi :

Apabila terdapat ekspresi $A + B * C$, maka operasi yang akan dikerjakan terlebih dahulu $B * C$, baru setelah itu hasilnya ditambahkan dengan A. Urutan proses tersebut terjadi karena proses perkalian mempunyai urutan derajat pengoperasian yang lebih tinggi dari penjumlahan. Namun jika bentuk ekspresi diubah menjadi $(A + B) * C$, maka operasi yang didahulukan $(A + B)$, kemudian hasilnya dikalikan dengan C. Hal ini menunjukkan bahwa tanda kurung digunakan untuk mendahulukan proses.

Apabila ada proses yang terdiri dari beberapa operator yang mempunyai urutan derajat yang sama, maka proses pengoperasian akan mendahulukan proses yang ditulis paling kiri.

- **Operator Assignment (Penugasan)**

Assignment atau operator penugasan dituliskan dengan bentuk “:=” (titik dua sama dengan) dan berfungsi untuk memasukkan suatu nilai data ke dalam sebuah variabel, dengan bentuk penulisan :

Nama_Variabel := Ekspresi;

Contoh :

Harga := 500;

Jumlah := 20;

Total := Harga * Jumlah;

- **Operator Aritmatika**

Operator aritmatika digunakan untuk melakukan operasi aritmatika. Delphi mempunyai beberapa operator aritmatika, yaitu seperti tabel dibawah ini.

Tabel 2.9 Tabel operator aritmatika pada Delphi

Operator	Fungsi	Tipe yang Diproses	Tipe Hasil Proses
*	Perkalian	Integer, Real	Integer, Real
/	Pembagian Real	Integer, Real	Integer, Real
+	Penjumlahan	Integer, Real	Integer, Real
-	Pengurangan	Integer, Real	Integer, Real
div	Pembagian Integer	Integer	Integer
mod	Sisa Hasil Bagi	Integer	Integer

Contoh :

Angka := 15 * 2; //Hasil 30

Angka := 15 / 2; //Hasil 7.5

Angka := 15 + 2; //Hasil 17

Angka := 15 - 2; //Hasil 13

Angka := 15 div 2; //Hasil 7

Angka := 15 mod 2; //Hasil 1

Untuk operasi perpangkatan, Delphi tidak me-nyediakan operator untuk pemangkatan. Sebagai gantinya, dapat digunakan rumus :

$$ax = \exp(b * \ln(a))$$

Contoh :

```
Angka1 := 5;
Angka2 := 2;
Hasil1 := exp(Angka1 * ln(Angka2)); //Hasil: 32
Hasil2 := exp(Angka2 * ln(Angka1)); //Hasil: 25
```

• Operator Relasi (Pembandingan)

Operotor relasi berfungsi untuk membandingkan suatu nilai (ekspresi) dengan nilai (ekspresi) lain yang akan meng-hasilkan suatu nilai logika (Boolean) yaitu True atau False. Kedua data yang dibandingkan tersebut harus memiliki tipe data yang sama. Berikut ini adalah tabel operator relasi yang disediakan Delphi :

Tabel 2.10 Tabel operator relasi pada Delphi

Operator	Keterangan
=	Sama Dengan
>	Lebih Besar
<	Lebih Kecil
>=	Lebih Besar atau Sama Dengan
<=	Lebih Kecil atau Sama Dengan
<>	Tidak Sama Dengan

Contoh :

```
x := 21 + 9 > 20; //menghasilkan True
x := 21 - 9 > 20; //menghasilkan False
```

• Operator Logika

Operator logika digunakan untuk mengekspresikan suatu satu atau lebih data atau ekspresi logika (Boolean), menghasilkan data logika (boolean) baru. Tabel operator logika dengan hierarki dari atas ke bawah adalah sebagai berikut :

Tabel 2.11 Tabel operator logika pada Delphi

Operator	Keterangan
not	Tidak
and	Dan
or	Atau
xor	Inklusiva Or

❖ Operator Logika Not

Operator logika Not merupakan operator yang menyatakan kondisi kebalikan dari suatu ekspresi.

Contoh :

```
x := not (21 > 9); //Hasil False
x := not (21 < 9); //Hasil True
x := not (21 = 9); //Hasil True
```

❖ Operator Logika And

Operator logika And hanya akan menghasilkan nilai True jika lebih dari satu ekspresi yang menggunakan operator And bernilai True. Apabila ada satu ekspresi atau lebih bernilai False maka operator logika And akan menghasilkan nilai False.

Contoh :

```
x := (21 > 9) and (19 < 71); //Hasil True
x := (21 < 9) and (19 < 71); //Hasil False
x := (21 < 9) and (19 > 71); //Hasil False
```

❖ Operator Logika Or

Operator logika Or akan menghasilkan nilai True jika salah satu atau seluruh ekspresi yang menggunakan operator Or bernilai True. Operator Or hanya akan bernilai False jika semua ekspresi yang menggunakan operator Or bernilai False.

Contoh :

```
x := (21 > 9) or (19 < 71); //Hasil True
x := (21 < 9) or (19 < 71); //Hasil True
x := (21 < 9) or (19 > 71); //Hasil False
```

❖ Operator Logika Xor

Operator logika Xor akan menghasilkan nilai true jika ekspresi yang terletak di kiri operator Xor berbeda dengan ekspresi disebelah kanannya. Operator Xor hamper memiliki fungsi yang sama dengan operator tidak sama dengan.

Contoh :

```
x := (9 < 21) xor (19 < 71); //Hasil False
x := (9 > 21) xor (19 > 71); //Hasil False
x := (9 < 21) xor (19 > 71); //Hasil True
x := (9 > 21) xor (19 < 71); //Hasil True
```

• Operator String

Operator String fungsinya untuk menggabungkan dua data string. Operator setring dilambangkan dengan bentuk "+".

Penggabungan dua buah string tersebut menghasilkan data yang berjenis string juga.

Contoh :

```
x := 'Mata Kuliah ';
y := 'Pemrograman Lanjutan';
z := x + y;
Maka z akan berisi 'Mata Kuliah Pemrograman
Lanjutan'
```

2.4. Dasar Basis Data

2.4.1. Konsep Basis Data

Basis Data (Database) dapat dibayangkan sebagai sebuah lemari arsip. Jika kita memiliki sebuah almari arsip dan berwenang untuk mengelolanya, maka kemungkinan besar kita akan melakukan hal-hal sebagai berikut:

- Memberi sampul/map pada kumpulan/bundel arsip yang akan disimpan
- Menentukan kelompok/jenis arsip
- Memberi penomoran dengan pola tertentu yang nilainya unik pada setiap sampul/map
- Menerapkan arsip-arsip tersebut dengan cara/urutan tertentu dalam almari dll

Upaya penyusunan/penempatan ini baru kita lakukan jika kita rasakan bahwa buku tersebut sudah cukup banyak.

2.4.2. Definisi

Basis data terdiri dari 2 kata, yaitu basis dan data. Basis adalah markas/gudang, tempat bersarang atau berkumpul, sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan dan sebagainya yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi dan kombinasinya.

Basis data dapat didefinisikan dari beberapa sudut pandang:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali secara cepat dan mudah.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu untuk memenuhi berbagai kebutuhan.
3. Kumpulan file/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Prinsip kerja dari basis data dan lemari arsip adalah sama, terutama adalah untuk pengaturan data/arsip. Selain itu, tujuan utamanya juga sama, yaitu untuk kemudahan dan kecepatan dalam pengambilan kembali data/arsip. Perbedaannya hanya terletak pada media penyimpanan yang digunakan. Jika lemari arsip menggunakan lemari besi atau kayu, maka basis data menggunakan media penyimpanan elektronik seperti disk (disket atau harddisk).

Satu hal yang harus diperhatikan bahwa basis data bukan hanya sekedar media penyimpanan data secara elektronik (dengan bantuan komputer), artinya tidak semua bentuk penyimpanan data elektronik dapat disebut basis data. Kita dapat menyimpan dokumen berisi data dalam file teks (dengan program pengolah kata), file spreadsheet dll, tetapi tidak dapat disebut sebagai basis data. Karena didalamnya tidak ada pemilahan dan pengelompokan data sesuai jenis/fungsi data, sehingga akan menyulitkan pencarian data kelak. Yang sangat ditorjalkan dalam basis data adalah pengaturan/ pemilahan/ pengelompokan/ pengorganisasian data yang akan kita simpan sesuai fungsi atau jenisnya.

Pemilahan/ pengelompokan/ pengorganisasian ini dapat berbentuk sejumlah file/ tabel terpisah atau dalam bentuk pendefinisian kolom-kolom/ field-field data setiap file/tabel.

2.4.3. Operasi Dasar Basis Data

Dalam sebuah disk, basis data dapat diciptakan dan dapat pula ditiadakan. Di dalam sebuah disk, kita dapat pula menempatkan beberapa (lebih dari satu) basis data. Sementara dalam sebuah basis data kita dapat menempatkan satu atau lebih file/tabel. Pada file/tabel inilah sesungguhnya data disimpan/ditempatkan.

Sebuah basis data umumnya dibuat untuk mewakili sebuah semesta data yang spesifik. Misalnya basis data kepegawaian, basis data akademik, basis data inventori (pergudangan) dan sebagainya. Dalam basis data akademik, misalnya kita dapat menempatkan file mata pelajaran, file guru, file jadwal, file nilai dan seterusnya.

Oleh karena itu operasi-operasi dasar yang dapat kita lakukan berkenaan dengan basis data dapat meliputi:

- Pembuatan basis data baru (*create database*) yang identik dengan pembuatan lemari rasip yang baru.
- Penghapusan basis data (*drop database*) yang identik dengan perusakan lemari rasip.
- Pembuatan file/ tabel baru ke suatu basis data (*create table*), identik dengan penambahan map arsip baru.
- Penghapusan file/tabel dari suatu basis data (*drop tabel*).
- Penambahan/pengisian data baru ke sebuah file/table di sebuah basis data.

- Pengambilan data dari sebuah file/tabel.
- Pengubahan data dari sebuah file/tabel.
- Penghapusan data dari sebuah file/tabel.

Operasi-operasi yang berkaitan dengan pembuatan objek (basis data dan tabel) merupakan operasi awal yang hanya dilakukan sekali dan berlaku seterusnya, sedangkan operasi-operasi yang berkaitan dengan pengisian tabel merupakan operasi-operasi rutin yang dilakukan berulang-ulang dan karena itu operasi-operasi inilah yang lebih tepat mewakili aktivitas pengelolaan (management) dan pengolahan (processing) data dalam basis data

2.5. Microsoft Access

Microsoft Access (atau Microsoft Office Access) adalah sebuah program aplikasi basis data komputer relasional yang ditujukan untuk kalangan rumahan dan perusahaan kecil hingga menengah. Aplikasi ini merupakan anggota dari beberapa aplikasi Microsoft Office, selain tentunya Microsoft Word, Microsoft Excel, dan Microsoft PowerPoint. Aplikasi ini menggunakan mesin basis data Microsoft Jet Database Engine, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna. Versi terakhir adalah Microsoft Office Access 2007 yang termasuk ke dalam Microsoft Office System 2007.

Microsoft Access dapat menggunakan data yang disimpan di dalam format Microsoft Access, Microsoft Jet Database Engine, Microsoft SQL Server, Oracle Database, atau semua kontainer basis data yang mendukung standar ODBC. Para pengguna/ *programmer* yang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para

programmer yang kurang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang sederhana. Access juga mendukung teknik-teknik pemrograman berorientasi objek, tetapi tidak dapat digolongkan ke dalam perangkat bantu pemrograman berorientasi objek.

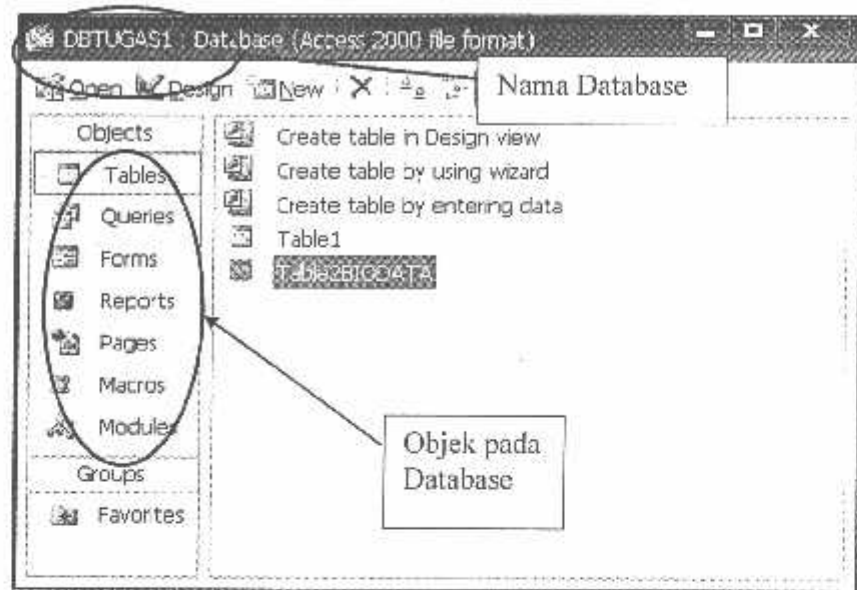
2.5.1. Keunggulan Microsoft Access

Salah satu keunggulan Microsoft Access dilihat dari perspektif *programmer* adalah kompatibilitasnya dengan bahasa pemrograman Structured Query Language (SQL); query dapat dilihat dan disunting sebagai statement-statement SQL, dan statement SQL dapat digunakan secara langsung di dalam Macro dan VBA Module untuk secara langsung memanipulasi tabel data dalam Access. Para pengguna dapat mencampurkan dan menggunakan kedua jenis bahasa tersebut (VBA dan Macro) untuk memprogram *form* dan logika dan juga untuk mengaplikasikan konsep berorientasi objek.

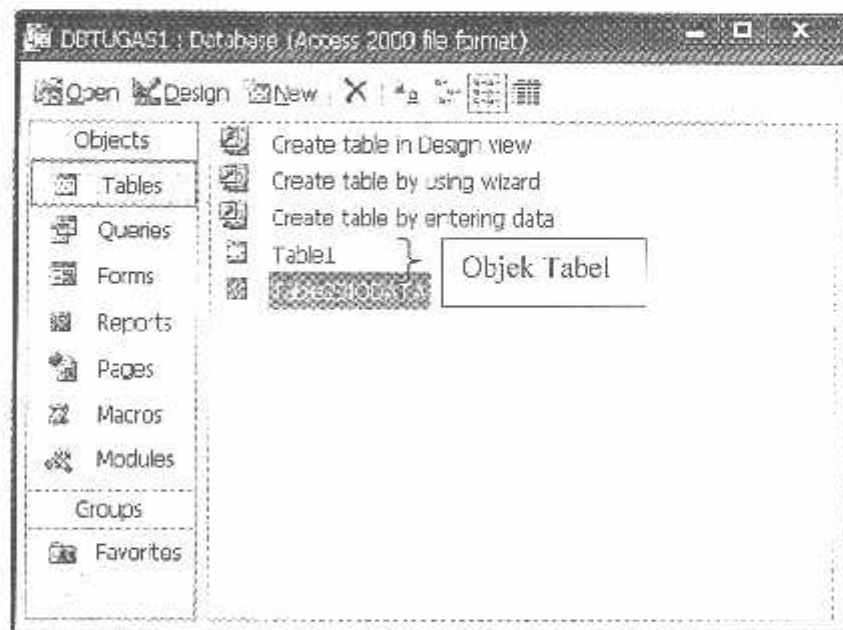
Microsoft SQL Server Desktop Engine (MSDE) 2000, yang merupakan sebuah versi mini dari Microsoft SQL Server 2000, dimasukkan ke dalam Office XP Developer Edition dan dapat digunakan oleh Microsoft Access sebagai alternatif dari Microsoft Jet Database Engine.

2.5.2. Database di dalam MS Access

Pengertian *database* pada MS Access adalah sekumpulan objek yang terdiri dari *Tabel, Query, Form, Report, Pages, Macro* dan *Module*. Object-object ini ditampung dalam satu wadah atau database. Jadi MS Access yang dimaksud database adalah kumpulan dari *Tabel, Query, Form, Report, Pages, Macro* dan *Module*. Database di MS Access selalu terdiri dari objek terdapat juga groups.



Gambar 2.4. Database dalam MS Access
(Microsoft Access 2003)



Gambar 2.5. Object Tabel
(Microsoft Access 2003)

Didalam MS Access sebuah database disimpan dalam *satu file* berekstensi *mdb*. Jadi dalam Access satu file mewakili atau mempresentasikan satu database. dimana satu database ini bisa terdiri atas beberapa *Tabel, Query, Form, Report, Pages, Macro* dan *Module*.

2.6. RS-232

Karakteristik listrik dari RS-232 meliputi level tegangan, yaitu mempunyai tegangan keluaran +5V sampai +15V untuk mewakili level rendah (logika '0') dan -5V sampai -15V untuk mewakili level tinggi (logika '1').

Untuk kebutuhan *interface*, sinyal RS-232 memanfaatkan sebuah konektor dengan jumlah pin 9 atau 25. Konektor tersebut sedang disebut DB 9 *conector*. Gambar tipe konektor DB 9 ditunjukkan dengan gambar di bawah ini



Gambar 2.6. Konfigurasi Pin Konektor RS-232
(www.google.com DB-9)

Fungsi masing-masing pin adalah sebagai berikut:

Tabel 2.12. Fungsi Pin dalam DB 9

No	Nama	Fungsi
1	DCR (<i>Data carrier Detect</i>)	Mendeteksi sinyal carrier dari modem lain

2	RD (<i>Receive Data Line</i>) (RxD)	Pengiriman data serial dari DCE ke DTE
3	TD (<i>Transmit Data Line</i>) (TxD)	Pengiriman data serial dari DTE ke DCE
4	DTR (<i>Data Terminal Ready</i>)	Memberitahukan DCE bahwa DTE telah aktif dan siap untuk bekerja
5	Ground	Referensi semua tegangan untarmuka
6	DSR (<i>Data Set Ready</i>)	Memberitahukan DTE dan DCE telah aktif dan siap untuk bekerja
7	RTS (<i>Request To Send</i>)	Memberitahukan DCE bahwa DTE akan mengirim data
8	CTS (<i>Clear To Send</i>)	Memberitahukan DTE bahwa DCE siap menerima data
9	RI (<i>Ring Indikator</i>)	Aktif jika modem menerima sinyal ring pada jalur telepon

Untuk spesifikasi RS-232 dapat dilihat dalam tabel 2.13

Tabel 2.13. Spesifikasi RS-232

Keistimewaan	Karakteristik
Jenis Operasi	Single ended (tak seimbang)
Jenis Penggerak dan Penerimaan per jalur	1 driver 1 receiver
Data rate maksimum	20 kbps
Panjang saluran maksimum	50 ft (15 m)
Tegangan keluaran penggerak	$\pm 5 - \pm 15$ volt
Sensitivitas penerima	± 3 volt

BAB III

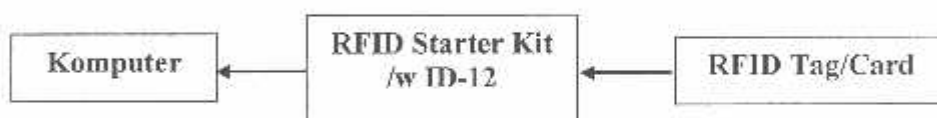
PERANCANGAN DAN PEMBUATAN ALAT

3.1. Pendahuluan

Dalam bab ini akan membahas tentang perencanaan dan pembuatan sistem yang digunakan untuk pengoperasian RFID pada suatu Perusahaan atau pada suatu Universitas terutama di ITN Malang.

3.1.1. Blok Diagram Keseluruhan

Adapun blok diagram sistem secara keseluruhan adalah sebagai berikut:



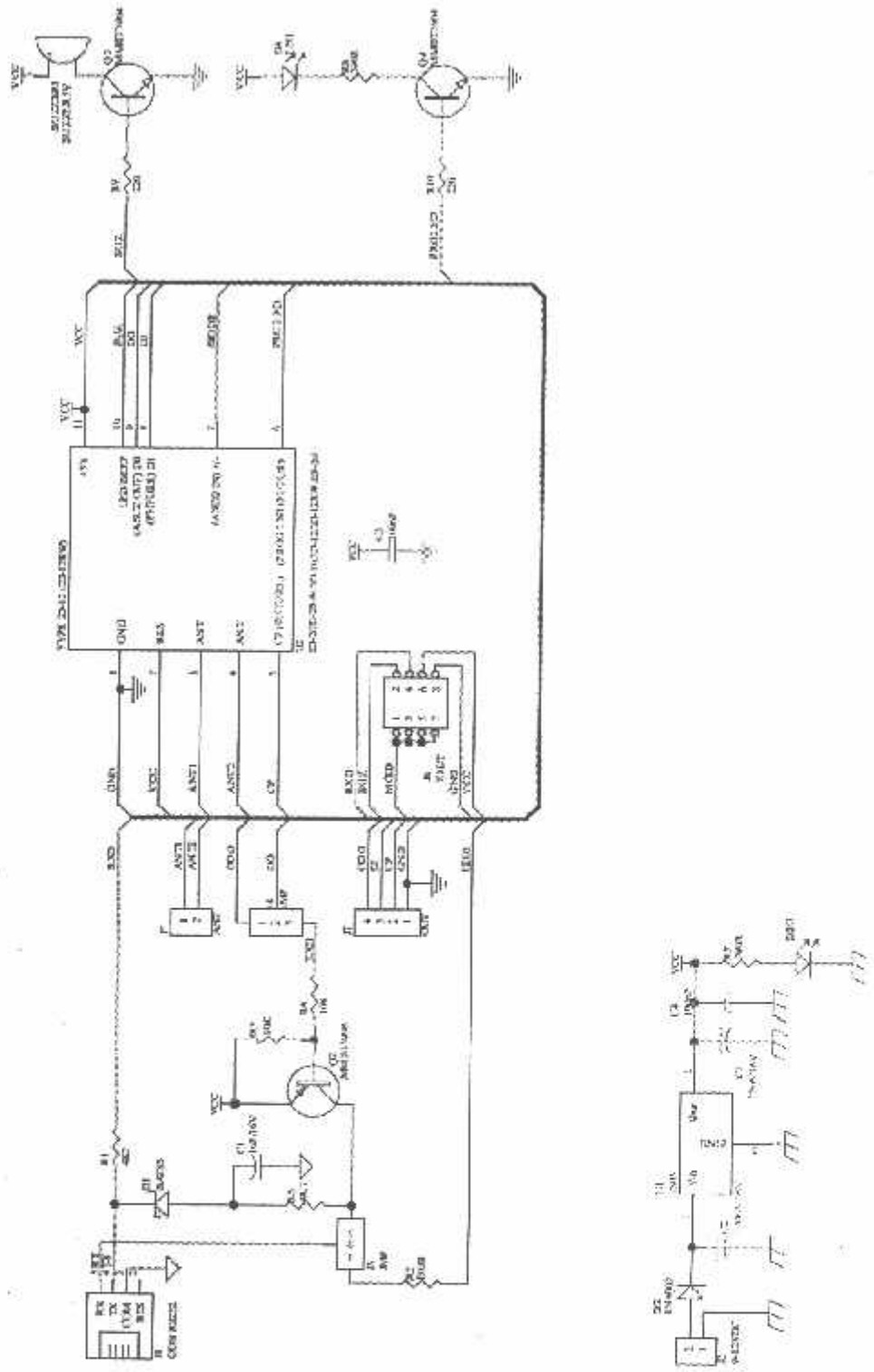
Gambar 3.1 Blok Diagram RFID Starter Kit

RFID Starter Kit dihubungkan ke komputer melalui *COM port* dengan menggunakan kabel serial RFID Starter Kit yang tersedia. Pada RFID Starter Kit aturlah *jumper* J3 & J4 pada posisi 2-3 (UART RS-232) dan *jumper* J6 pada posisi 4 (mode ASCII). RFID *reader* ID-12 yang digunakan dalam AN ini dapat diganti dengan ID-2 atau ID-20. Setelah semua rangkaian dan sumber tegangan terhubung dengan benar, program RFID dapat dijalankan dan pilih *COM port* yang digunakan.

3.2. Prinsip Kerja Alat

Kerja dari alat ini merupakan sebuah aplikasi sederhana yang menggunakan RFID Starter Kit untuk membaca nomor ID dari *RFID transponder* dengan bantuan komputer. RFID Starter Kit dihubungkan ke komputer melalui *COM port* dan nomor ID dari *RFID transponder* akan ditampilkan pada sebuah jendela aplikasi yang dikembangkan dalam bahasa Borland Delphi. Aplikasi ini hanya membutuhkan RFID Starter Kit dengan kabel serialnya serta bantuan sebuah komputer.

Pada umumnya RFID tidak memancarkan sinyal terus-menerus, RFID hanya akan memancarkan sinyalnya apabila ada sinyal pemicu yang sesuai dengan tata cara pengiriman dan penerimaannya (protokol). Sinyal pemicu ini biasanya ditempatkan menjadi satu pada alat pemancar/penerima (reader dan antena) dengan range kisaran pembaca 8 cm serta bekerja pada frekuensi 125 KHz.



Gambar 3.2. Rangkaian RFID

3.3. Komunikasi Data antara RFID Tag dengan Komputer

Cara membaca Tag RFID adalah sebagai berikut :

- Bukalah Hyper-Terminal Setup kemudian pilih ikon dan berilah nama, kemudian klik OK.



Gambar 3.3. Connection Hyper-Terminal

- Lalu muncul tampilan seperti gambar 3.3, pilihlah *connect using COM1* atau COM2 sesuai dengan serial yang terhubung lalu kil OK.



Gambar 3.4. Conect Using

- Kemudian muncul tampilan *serial port parameter* aturlah kecepatan dengan 9600 bps, 8 Data bits, 1 Stop bits, Parity *None*, dan Flow Control *None*, kemudian klik OK.



Gambar 3.5 serial port parameter

- Kemudian muncul tampilan seperti gambar dibawah ini maka modul RFID terhubung dengan serial port pada PC menggunakan Hyper-Terminal - RFID tag, sehingga nomor pada tag bisa digunakan.



Gambar 3.6. Tampilan Kode Tag RFID pada Hyper-Termina

Program RFID ini secara garis besar bekerja sebagai berikut:

1. Program melakukan inisialisasi *COM port* yaitu pada *baudrate* 9600 bps, 8 bit data, 1 bit stop, tanpa bit *parity*, tanpa *flow control*. Kemudian program melakukan deklarasi variabel yang akan digunakan antara lain:
 - Counter adalah variabel bertipe *byte* dan digunakan sebagai penghitung jumlah byte data yang diterima.
 - Nomor adalah variabel bertipe *string* dan digunakan untuk menampung data ID dari *RFID transponder*.
 - Data adalah variabel bertipe *byte* yang digunakan untuk menampung data serial dari *COM port*.
2. Kemudian program memberi nilai nol pada variabel counter dan membuat variabel nomor menjadi *null* (kosong).
3. Program menunggu datangnya data serial dari *COM port*. Bila data serial datang maka program akan menyimpannya pada variabel data dan melanjutkan ke langkah berikutnya.
4. Program akan menambah variabel counter dengan 1, dan memeriksa nilai variabel counter. Jika variabel counter bernilai 1 maka program akan memeriksa apakah variabel data bernilai 2. Jika variabel data tidak bernilai 2 maka program akan memberi nilai variabel counter dengan 0 dan membuat variabel nomor menjadi *null*.
5. Jika variabel counter bernilai 4 hingga 11 maka program akan mengkonversi nilai variabel data menjadi *string* dan menambahkannya ke variabel nomor (sintaksis \rightarrow nomor:= nomor + chr[data]).

6. Jika nilai variabel counter adalah 16 maka program akan memeriksa apakah variabel data bernilai 3. Jika variabel data bernilai 3 maka program akan mengkonversi data dalam variabel nomor yang berformat heksadesimal menjadi format desimal lalu menampilkannya pada jendela program (melalui label1). Jika data tidak bernilai 3 maka program akan langsung melanjutkan ke langkah berikutnya.
7. Kemudian program memberi nilai nol pada variabel counter dan membuat variabel nomor menjadi *null* (kosong).
8. Lalu program kembali ke langkah 3.

3.5. Perancangan Perangkat Lunak (*Software*)

Perangkat lunak adalah sebuah jembatan yang menghubungkan keseluruhan komponen yang ada pada sebuah computer. Pada perancangan ini akan membahas tentang perangkat lunak untuk computer menggunakan *MS Access* dan *Delphi7*

3.5.1. Program Aplikasi Komputer

Program aplikasi adalah *software* di dalam komputer yang berfungsi untuk melakukan pengendalian. Program ini bertujuan untuk mengorganisasi komunikasi antara *MS Access* dengan *Delphi7*, sehingga perangkat lunak dalam program komputer perlu diketahui terlebih dahulu aturan-aturan atau protocol komunikasi yang digunakan untuk mengatur jalannya komunikasi antara *MS Access* dengan *Delphi7*.

3.5.2. Perancangan Pembuatan Software Menggunakan Bahasa Pemrograman *Borland Delphi7* dan *MS Access*.

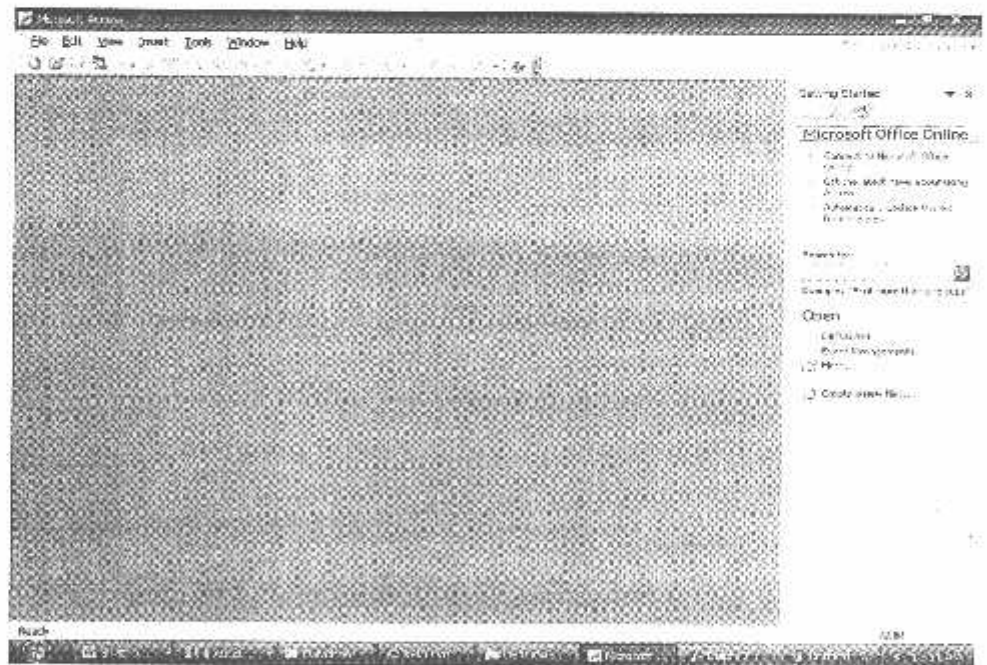
3.5.2.1. Penyusunan Desain Software

Dalam Penyusunan desain tampilan software perlu diperhatikan dengan teliti terutama pada saat membuat database pada MS Access. Sebelum masuk ke Delphi7 terlebih dahulu buat database.

1. Membuat *Database*

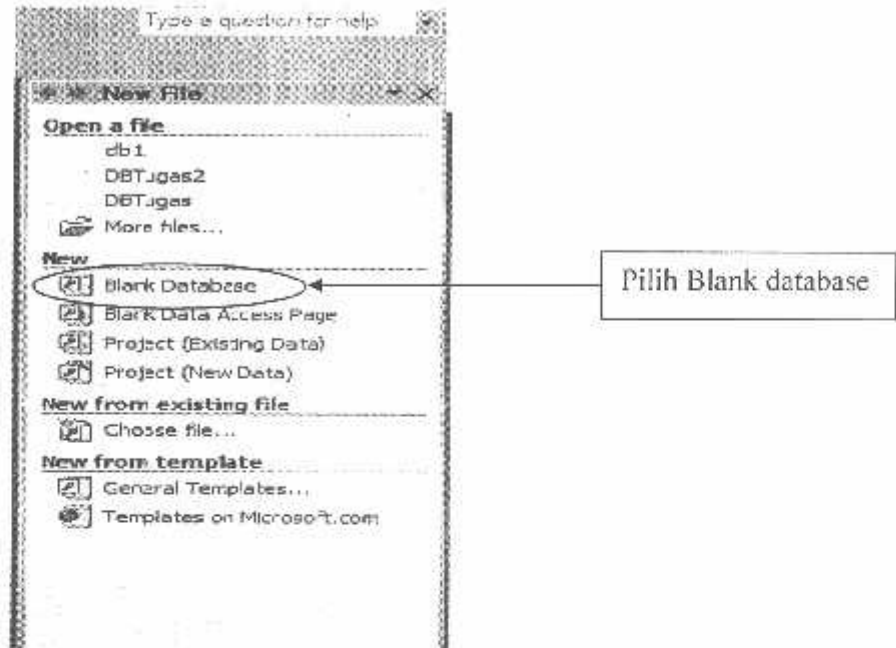
Langkah pertama yang dilakukan dalam membuat suatu Aplikasi *Database* adalah membuat *database*. Adapun langkah-langkah membuat database sebagai berikut:

- Membuka aplikasi MS Access akan muncul tampilan seperti gambar 3.8



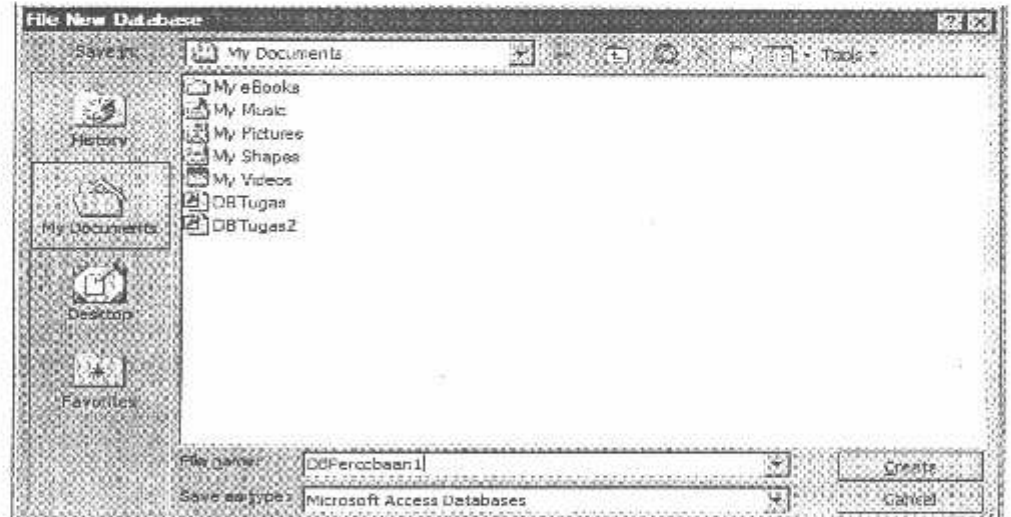
Gambar 3.8. Tampilan Utama MS Access

- Setelah muncul tampilan utama pilihlah 'Blank Database' untuk membuat Database baru



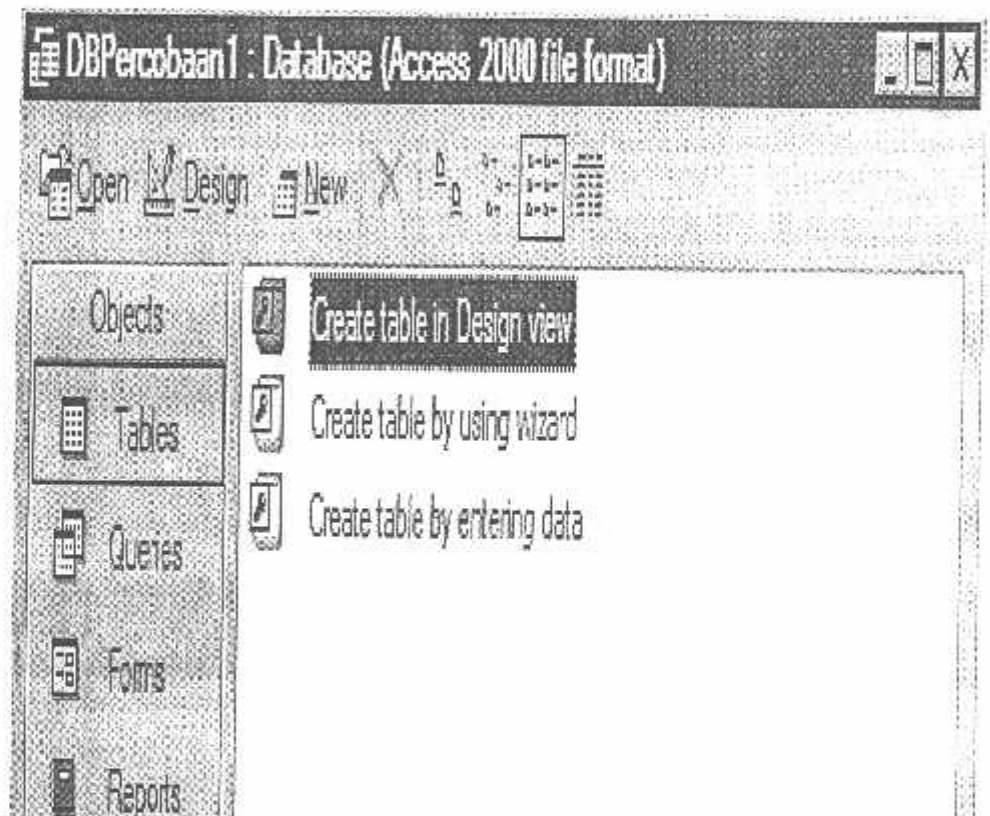
Gambar 3.9. Blank Database

- Setelah dipilih muncul tampilan untuk membuat folder beserta nama file/nama database yang akan dibuat



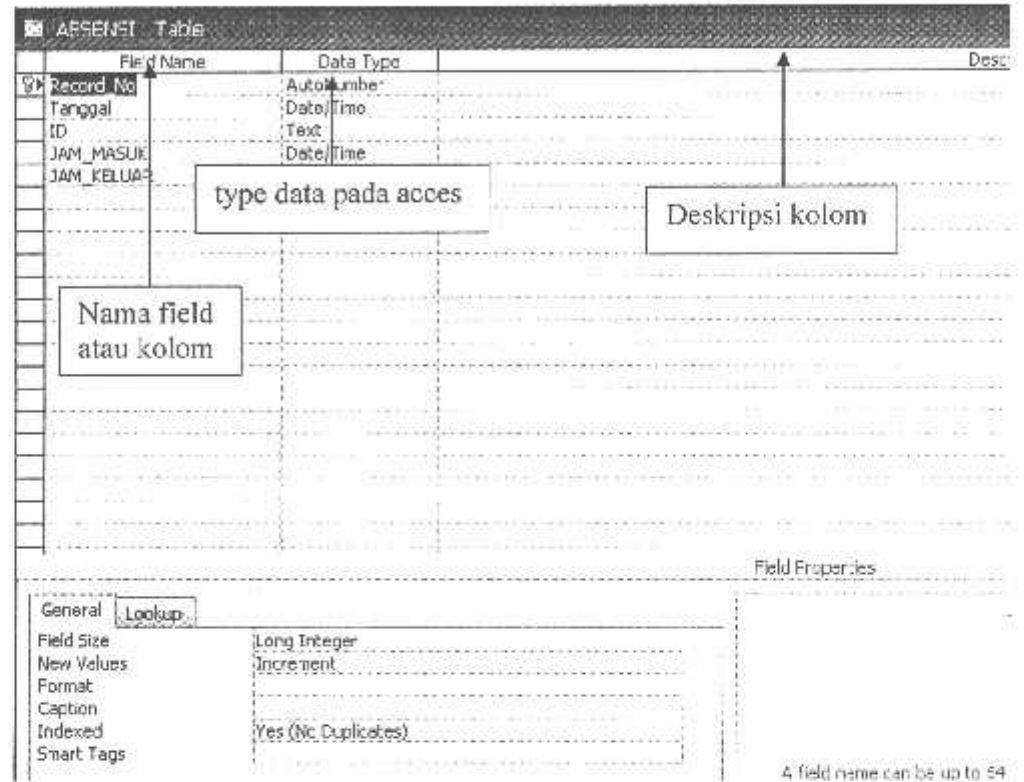
Gambar 3.10. Gambar New Database

- Setelah sudah diberi nama misalnya 'DBPercobaan1' tekan tombol *create* untuk membuat database dengan nama 'DBPercobaan1'. Didalam MS Access satu databasedisimpan dalam satu file. Dengan kata lain file DBPercobaan1 adalah database DBPercobaan1.
Setelah membuat database langkah selanjutnya adalah membuat isi dari database DBPercobaan1 yaitu berupa tabel-tabel.
- Selanjutnya untuk membuat tabel baru di MS Access double klik di '*Create table in Design view*'



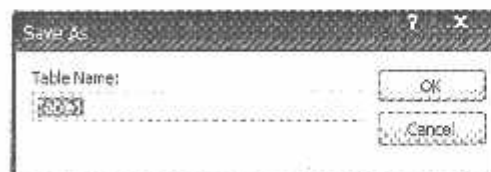
Gambar 3.11. Membuat tabel database

- Maka akan muncul struktur tabel yang akan dibuat, terdiri dari *Field Name* (Nama Kolom), *Data Type* (Type Data), dan *Description* (Keterangan mengenai kolom tertentu).



Gambar 3.12. Struktur Tabel

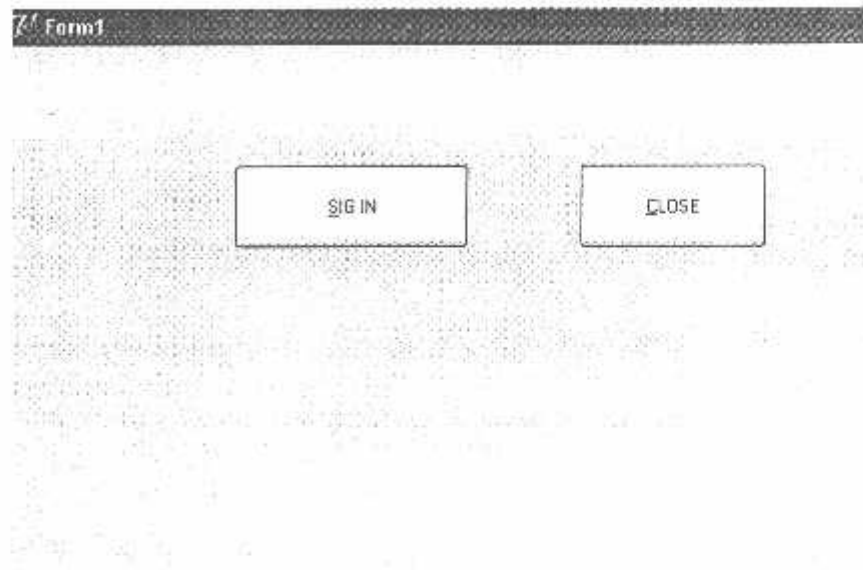
- Setelah Tabel dibuat, Jika ingin menyimpan Tabel database klik close pada lembar struktur tabel.



Gambar 3.13 Save Tabel

2. Membuat tampilan Desain koneksi dengan menggunakan Delphi7

- Menyusun desain Menu Utama seperti gambar dibawah ini :



Gambar 3.14. Menyusun Tampilan Design Menu Utama

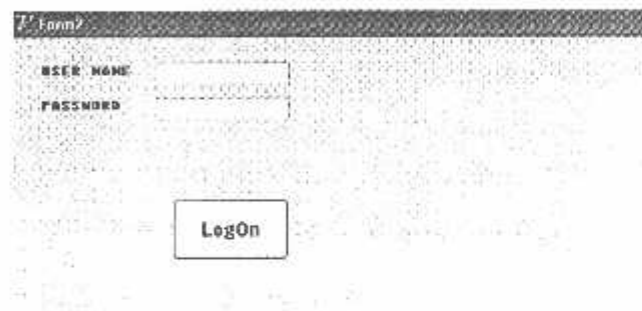
Untuk Mendesain Tampilan diatas diperlukan komponen *ADO Connection*, 2 buah *Speed button*. Kemudian merubah properti dari masing-masing komponen seperti dibawah ini.

Tabel 3.1 Merubah isi Properti pada Layar Utama

Komponen	Properti	isi
ADOConnection1 (Pada palet ADO)	Name	ADOConnection1
	Connection string	Microsoft.jet 4.0 OLE DB Provider
	LoginPrompt	False
	Connected	True

ADOQuery1 (Pada Palet ADO)	Connection	ADOCnection1
	Name	ADOTable1
Speed Buttom 1	Name	Speed Buttom 1
	Caption	Sig In
Speed Buttom 2	Name	Speed Buttom 2
	Caption	Close

- Menyusun Desain Tampilan



Gambar 3.15. Menu User

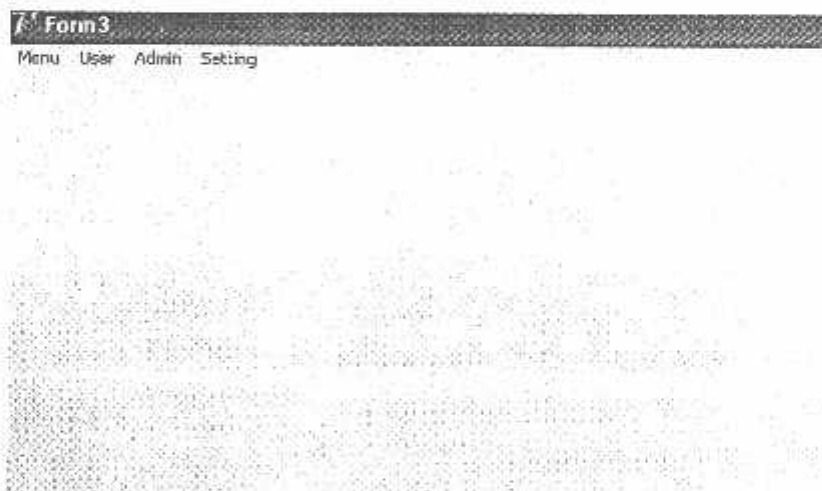
Untuk mendesain Tampilan seperti diatas perlu beberapa komponen diantaranya ADO Query 1, 2 buah DBEdit, 2 buah Label, Speed buttom. Kemudian mengubah properti seperti tabel dibawah ini :

Tabel 3.2. Merubah isi Properti pada menu user

Komponen	Property	Isi
ADO Query 1 (Pada palet ADO)	Name	ADO Query 1
	Connection	Form 1 ADO connection 1
	Aktif	False

DBEdit 1 (Pada Palet Data Controls)	Name	DBEdit 1
DBEdit 2 (Pada Palet Data Controls)	Name	DBEdit 2
Label 1 (pada palet standart)	Name	Label 1
	Caption	User Name
	FocusControl	Kosong
Label 2 (pada palet standart)	Name	Label 2
	Caption	Password
	FocusControl	Kosong
Speed Buttom	Name	Speed Buttom 1
	Caption	Log On

- Menyusun Desain Tampilan



Gambar 3.16. Menu Pilihan

Untuk mendesain Tampilan seperti diatas perlu beberapa komponen diantaranya Data souerce 1, main menu, User, Admin, Setting, Status bar .Kemudian mengubah properti seperti tabel dibawah ini :

Tabel 3.3. Merubah isi Properti pada menu user

Komponen	Properti	isi
DataSource1	Name	Form 3
	Caption	Form 3
Main menu	Name	Main menu 1
	Caption	Menu
User	Name	User 1
	Caption	User
Admin	Name	Admin 1
	Caption	Admin
Setting	Name	Setting 1
	Caption	Setting
Status Bar	Name	Name

- Menyusun Desain Tampilan

The screenshot shows a Windows application window titled "Form3". At the top, there is a data grid with the following data:

NIM	Nama Mahasiswa	Program Studi	Alamat
0412557	esi	Informatika	Jl. Tembus 10 Batu
951709	iem	Industri	Jl. Tembus 15 Jombang
0020145	budi	elektro	Jl. Siguregara 3 Malang
9811271	tepa	mesin	Jl. Dunte 5 Gemarang

Below the grid, there are several input fields and a button:

- NIM:
- Nama Mahasiswa:
- Program Studi: with navigation buttons (Next, Previous, Home, End, Refresh).
- Alamat:
- Jenis Kelamin: with a "Menu" button.
- TTL:
- No Telp:

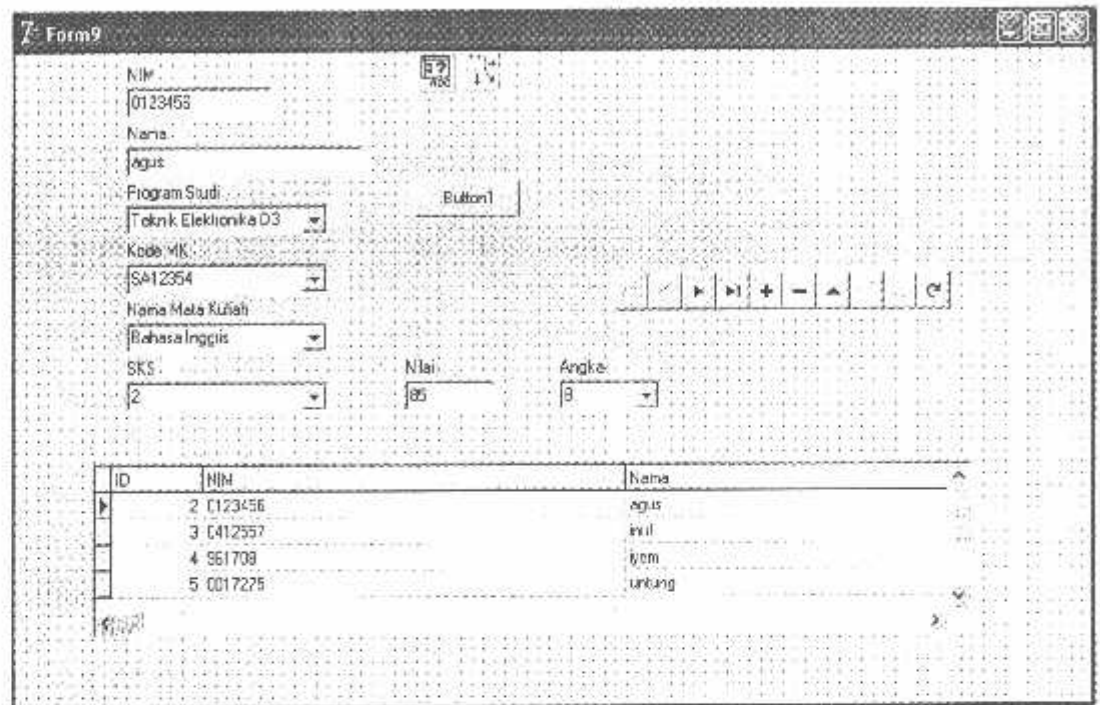
Gambar 3.17. Menu Data Mahasiswa

Untuk mendesain Tampilan seperti diatas perlu beberapa komponen diantaranya 7 buah DBEdit, 7 buah Label, DB Grid1, DB Navigator. Kemudian mengubah properti seperti tabel dibawah ini :

Tabel 3.4. Merubah isi Properti pada menu data mahasiswa

Komponen	Properti	isi
Buttom 1	Caption	Menu
	Name	Buttom 1
DBEdit1	Name	DBEdit1
	DataFailed	Nama Mahasiswa
DBEdit2	Name	DBEdit2
	DataFailed	Program Studi
DBEdit3	Name	DBEdit3
	DataFailed	Alamat
DBEdit4	Name	DBEdit4
	DataFailed	Jenis Kelamin
DBEdit5	Name	DBEdit5
	DataFailed	NIM
DBEdit6	Name	DBEdit6
	DataFailed	TTL
DBEdit7	Name	DBEdit7
	DataFailed	No Tlp
DPGrid1	Name	DPGrid1
	DataSource	DataSource2
DBNavigator	Name	DBNavigator
	DataSource	DataSource2
DBLabel2	Name	DBLabel2
	Caption	Nama Mahasiswa
DBLabel3	Name	DBLabel3
	Caption	Program Studi
DBLabel4	Name	DBLabel4
	Caption	Alamat
DBLabel5	Name	DBLabel5
	Caption	Jenis Kelamin
DBLabel6	Name	DBLabel6
	Data field	TTL
DBLabel7	Name	DBLabel7
	Caption	No Tlp
DBLabel8	Name	DBLabel8
	Caption	NIM

- Menyusun Desain Tampilan data nilai



Gambar 3.18. Menu Data Nilai

Untuk mendesain Tampilan seperti diatas perlu beberapa komponen diantaranya 5 buah DBCombo Box, 1 buah DBGrid1, 1 buah DBNavigator, 3 buah DBEdit, 8 buah Label. Kemudian mengubah properti seperti tabel dibawah ini :

Tabel 3.5. Merubah isi Properti pada menu data nilai

Komponen	Properti	isi
Button1	Name	Button1
	Caption	Button1
DBCombo Box1	Data Field	Program Studi
	Name	DBCombo Box1

DBCombo Box2	Data Field	Kode MK
	Name	DBCombo Box2
DBCombo Box3	Data Field	Nama MK
	Name	DBCombo Box3
DBCombo Box4	Data Field	SKS
	Name	DBCombo Box4
DBCombo Box5	Data Field	Angka
	Name	DBCombo Box5
DBEdit1	Name	DBEdit1
	Data field	NIM
DBEdit2	Name	DBEdit2
	Data field	Nama
DBEdit3	Name	DBEdit3
	Data field	Nilai
DBGrid1	Name	DBGrid1
	Data Source	Data Source2
DBNavigator	Name	DBNavigator
	Data Source	Data Source2
DBLabel1	Name	DBLabel1
	Caption	NIM
DBLabel2	Name	DBLabel2
	Caption	Nama
DBLabel3	Name	DBLabel3
	Caption	Program Studi
DBLabel4	Name	DBLabel4
	Caption	Kode Mata kuliah
DBLabel5	Name	DBLabel5
	Caption	Nama Mata Kuliah
DBLabel6	Name	DBLabel6
	Caption	SKS
DBLabel7	Name	DBLabel7
	Caption	Nilai
DBLabel8	Name	DBLabel8
	Caption	Angka

BAB IV

PENGUJIAN RANGKAIN DAN ANALISA DATA

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

4.1 Pengujian Penerimaan Data dari database

4.1.1. Tujuan

Untuk mengetahui apakah data yang diterima oleh Delphi sesuai dengan apa yang ada pada database.

4.1.2. Peralatan yang digunakan

1. Komputer
2. Rangkaian RFID
3. Kabel Serial RS-232
4. Adaptor 9 – 12 Volt

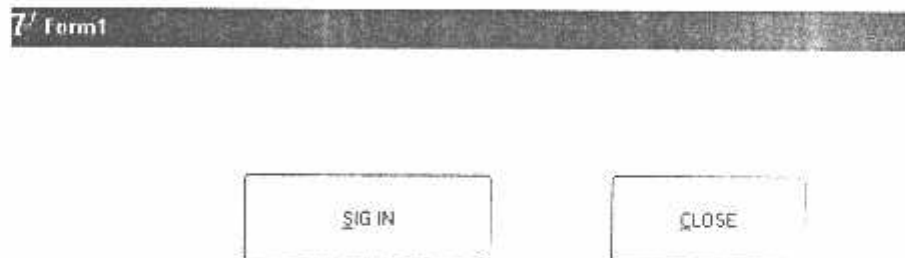
4.1.3. Langkah Pengujian

1. Membuka program MS Access, Kemudian buat Database.
 2. Membuka Program Delphi dan menjalankan program aplikasi.
 3. Menghubungkan RFID ke PC menggunakan kabel serial RS-232.
 4. Program dijalankan
-

4.1.4. Hasil Pengujian

Hasil pengujian Penerimaan Data dengan RFID dari suatu Komputer sebagai berikut :

- Setelah membuat Database kemudian pada program delphi dijalankan maka akan muncul suatu program yang sudah dibuat,yaitu seperti gambar dibawah ini

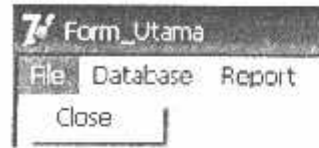


Gambar 4.1. Menu Utama program

Prinsip kerja dari program ini adalah hanya untuk sign in ke program selanjutnya.

1. Menu pada program

- File

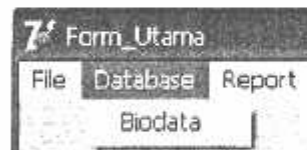


Gambar 4.2. Menu File

Pada menu file terdapat pilihan CLOSE listing programnya sebagai berikut

```
procedure TForm_Utama.Close1Click(Sender: TObject);
begin
    Close;
end;
```

- Database



Gambar 4.3. Menu Database

Pada menu Database terdapat pilihan Biodata ini merupakan pilihan kemana kita akan melihat tampilan dari Biodata, adapun program untuk bisa menampilkan form Biodata sebagai berikut.

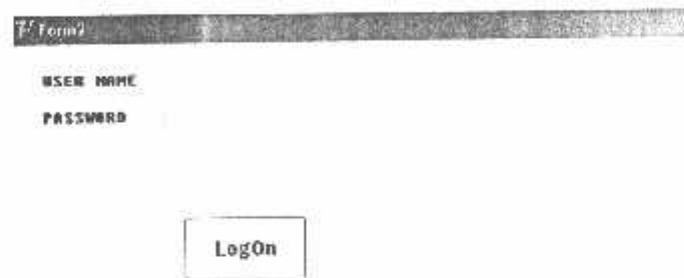
```
procedure TForm_Utama.Biodata1Click(Sender: TObject);
```

```
begin
```

```
Form_Biodata.ShowModal;
```

```
end;
```

Ini merupakan tampilan dari Database Biodata Karyawan

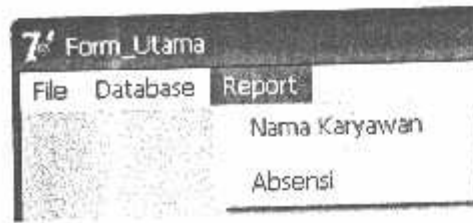


The image shows a screenshot of a login form. At the top, there is a title bar that says "Form". Below the title bar, there are two input fields: "USER NAME" and "PASSWORD". Below these fields, there is a button labeled "LogOn".

Gambar 4.4. Lembar Biodata

Kerja dari Database tersebut adalah suatu identitas karyawan yang dimiliki suatu Perusahaan atau Universitas yang mana apabila ada keperluan mencari seorang karyawan, tinggal melihat database Biodata Mahasiswa tersebut

- Report



Gambar 4.5. Tampilan menu Report

Pada pilihan menu Report akan muncul 'Nama Karyawan' dan 'Absensi'

Pada report merupakan suatu aplikasi untuk bisa dibuat untuk mencetak/print suatu data

```
procedure TForm_Utama>NamaKaryawan1Click(Sender: TObject);
```

```
begin
```

```
Form_Report_Biodata.ADOQuery1.Active := False;
```

```
Form_Report_Biodata.ADOQuery1.Active := True;
```

```
Form_Report_Biodata.Preview;
```

```
end;
```

```
procedure TForm_Utama.Absensi1Click(Sender: TObject);
```

```
begin
```

```
Form_Report_Absensi.ADOQuery1.Active := False;
```

```
Form_Report_Absensi.ADOQuery1.Active := True;
```

```
Form_Report_Absensi.Preview;
```

```
end;
```

Adapun tampilan dari Report yang digunakan dalam database

Gambar 4.6. Tampilan Report Daftar Nama Karyawan

Gambar 4.7. Tampilan Report Laporan Absensi

Jarak yang diterima oleh RFID *reader* melalui Tag bisa diambil data sebagai berikut :

Tabel 4.1. Jarak sinyal yang diterima oleh RFID

NO	Jarak Tag ke RFID reader	Kartu 1	Kartu 2	Kartu 3	Kartu5	Kartu6	Kartu 7
1	1 cm	√	√	√	√	√	√
2	3 cm	√	√	√	√	√	√
3	5 cm	√	√	√	√	√	√
4	8 cm	√	√	√	√	√	√
5	10 cm	√	√	√	√	√	√
6	12 cm	√	√	√	√	√	√
7	13 cm	x	x	x	x	x	x
8	14 cm	x	x	x	x	x	x

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dari Pengujian yang dilakukan dengan menggunakan alat ini, maka dapat diambil kesimpulan bahwa :

1. Dengan menggunakan komponen VaComm, data yang diterima sesuai dengan data asli yang dikirim melalui Tag RFID.
2. RFID menerima sinyal dari Tag \pm 12 cm.
3. Jika RFID menerima sinyal dari luar / Tag lain yang tidak terdaftar pada database maka program akan menolak.

5.2. Saran

Diharapkan dengan penyusunan Aplikasi ini dapat dijadikan sebagai program aplikasi yang dapat memberikan kemudahan dalam proses absensi dan memberikan nilai tambah dalam kedisiplinan terhadap suatu karyawan..

DAFTAR PUSTAKA

- Komunikasi dengan tim Pengembang P3TIE-BPPT
 - M.M Sarinanto. Prosiding Seminar Teknologi untuk negeri, BPPT. Maret 2002
 - Proceedings, Computer dan Sistem Intelijen. Comit 2002
 - Smart Cards Security Basics < <http://www.cardlogix.com> >
 - URL : <http://www.inn.bppt.go.id/info/kantaya/>
-

INSTITUT TEKNOLOGI NASIONAL

Jl. Bendungan Sigura-gura No. 2

M A L A N G

Lampiran : 1 (satu) berkas

Pembimbing Skripsi

Kepada : Yth. Bapak Joseph Deddy Irawan. ST. MT.

Dosen Institut Teknologi Nasional

M A L A N G

Yang bertanda tangan di bawah ini :

Nama : Hendras Prasetyo.

Nim : 0017275

Jurusan : Teknik Elektro S-1

Kosentrasi : **Teknik Elektronika**

Dengan ini mengajukan permohonan, kiranya Bapak bersedia menjadi Dosen Pembimbing Utama / Pendamping *), untuk penyusunan Skripsi dengan judul (proposal terlampir) :

“ Perancangan dan pembuatan system informasi nilai akademik dengan menggunakan RFID “.

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.

Demikian permohonan kami dan atas kesediaan Bapak kami ucapkan terima kasih.

Malang, Juli 2009

Mengetahui
Ketua Jurusan Teknik Elektro S-1

Hormat kami,

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


Hendras Prasetyo.

*) coret yang tidak perlu

INSTITUT TEKNOLOGI NASIONAL

Jl. Bendungan Sigura-gura No. 2

MALANG

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa :

Nama : Hendras Prasctyo.

Nim : 0017275.

Semester : XV111 (Delapan Belas)

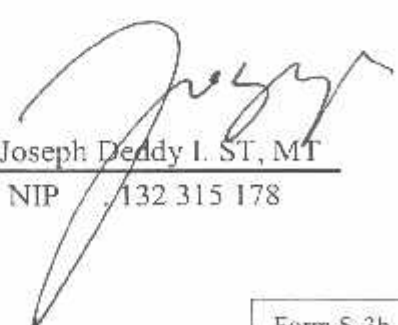
Jurusan : Teknik Elektronika

Dengan ini menyatakan bersedia / tidak bersedia *) Membimbing Skripsi dari mahasiswa tersebut, dengan judul :

“ Perancangan dan pembuatan system informasi nilai akademik dengan menggunakan RFID “.

Demikian Surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang, Juli 2009
Kami yang membuat pernyataan

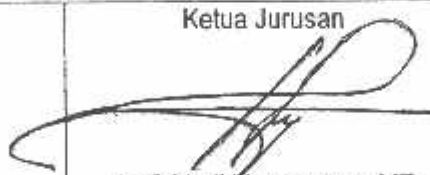
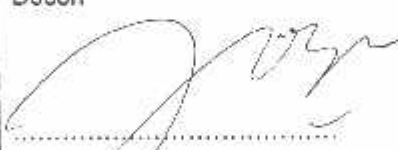

Joseph Dedy I. ST, MT
NIP .132 315 178

Form S-3b



LEMBAR PENGAJUAN JUDUL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika/Teknik Komputer & Informatika*)

1.	Nama Mahasiswa: HENDRAS PRASETYO	Nim: 0017275								
2.	Waktu Pengajuan	Tanggal: _____ Bulan: _____ Tahun: _____								
3.	Spesifikasi Judul (berilah tanda silang)**)									
	<table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none;">a. Sistem Tenaga Elektrik</td> <td style="width: 50%; border: none;">e. Elektronika & Komponen</td> </tr> <tr> <td style="border: none;">b. Energi & Konversi Energi</td> <td style="border: none;"><input checked="" type="checkbox"/> f. Elektronika Digital & Komputer</td> </tr> <tr> <td style="border: none;">c. Tegangan Tinggi & Pengukuran</td> <td style="border: none;">g. Elektronika Komunikasi</td> </tr> <tr> <td style="border: none;">d. Sistem Kendali Industri</td> <td style="border: none;">h. lainnya</td> </tr> </table>		a. Sistem Tenaga Elektrik	e. Elektronika & Komponen	b. Energi & Konversi Energi	<input checked="" type="checkbox"/> f. Elektronika Digital & Komputer	c. Tegangan Tinggi & Pengukuran	g. Elektronika Komunikasi	d. Sistem Kendali Industri	h. lainnya
a. Sistem Tenaga Elektrik	e. Elektronika & Komponen									
b. Energi & Konversi Energi	<input checked="" type="checkbox"/> f. Elektronika Digital & Komputer									
c. Tegangan Tinggi & Pengukuran	g. Elektronika Komunikasi									
d. Sistem Kendali Industri	h. lainnya									
4.	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen*) <i>Joseph Deddy I. ST MT</i>	Ketua Jurusan  Ir. F. Yudi Limpraptono, MT NIP. P. 1039500274								
5.	Judul yang diajukan mahasiswa:	<i>Berancangan dan pembuatan sistem informasi Nilai akademik dengan menggunakan smart card untuk login</i>								
6.	Perubahan judul yang disetujui Dosen sesuai materi bidang ilmu	<i>MENYATAKAN HALIK AKSES</i>								
7.	Persetujuan Judul skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu	Disetujui Dosen <i>20 - 12 - 2008</i> 								

Perhatian:

1. Formulir pengajuan ini harap dikembalikan kepada jurusan paling lambat satu minggu setelah disetujui kelompok dosen keahlian dengan dilampirkan proposal skripsi beserta persyaratan skripsi sesuai form S-1
2. Keterangan: *) Coret yang tidak perlu
**) dilingkari a, b, c, atau g sesuai bidang keahlian



FORMULIR BIMBINGAN SKRIPSI

Nama : Hendras Prasetyo
Nim : 00.17.275
Masa Bimbingan : 29-Mei-2009 s/d 29-November -2009
Judul Skripsi : **PERANCANGAN DAN PEMBUATAN SYSTEM INFORMASI
NILAI AKADEMIK DENGAN MENGGUNAKAN RFID**

No	Tanggal	Uraian	Paraf Pembimbing
1			
2			
3			
4			
5			
6			
7	3-12-2009	A-a sumptu	
8			
9			
10			

Malang, Mei 2009

Dosen pembimbing I

Joseph Dedy Irawan, ST,MT
NIP.132315178

Form S-4a



Formulir Perbaikan Ujian Skripsi

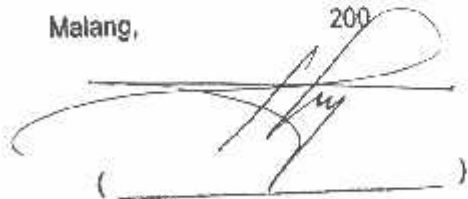
Dalam pelaksanaan Ujian Skripsi Jenjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T. Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : HENDRAS P
NIM :
Perbaikan meliputi :

- ① Penulisan abstrak 1 bps, terdiri dari latar blg, ~~is~~ sistem yg dibuat dan hasil² pengujian.
- ② penulisan di Bab 2, gambar, tabel hrs ada sumber. Penulisan sumber hrs standar, pakai usul [1] atau pada metode lain.
- ③ Kesimpulan diambil dari hasil² pengujian
hasil : pengujian RFID reader, komunikasi serial, program dll
- ④ Kata pengantar belum ada.
- ⑤ Belajar lagi !!!

Malang,

200





INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

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

NAMA : HENDRAS P
NIM : 06.17.295
Perbaikan meliputi :

* hal. 34 gambar schematic. file kecil !
* Kesimpulan dari hasil pengujian.
Perbaikan lagi !

Malang, 06-10-2009

Hendras P.

DATA FORMATS

Output Data Structure – ASCII

STX (02h)	DATA (10 ASCII)	CHECK SUM (7 ASCII)	CR	LF	ETX (03h)
-----------	-----------------	---------------------	----	----	-----------

[The 1 byte (2 ASCII characters) Check sum is the "Exclusive OR" of the 5 hex bytes (10 ASCII) Data characters.]

Output Data Structure – Wiegand26

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
P	E	E	E	E	E	E	E	E	E	E	E	E	O	O	O	O	O	O	O	O	O	O	O	O	O	P
Even parity (E)													Odd parity (O)													

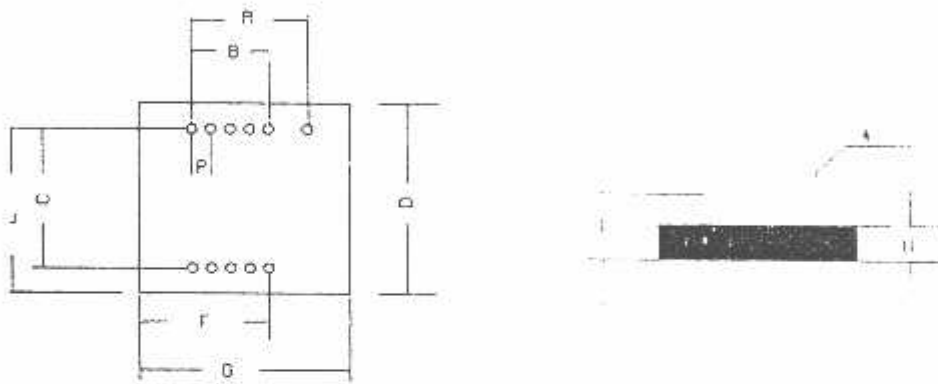
P = Parity start bit and stop bit

Output Data Magnetic ABA Track2

10 Leading Zeros	SS	Data	ES	LCR	10 Ending Zeros
------------------	----	------	----	-----	-----------------

[SS is the Start Character of 11010, ES is the end character of 11111, LRC is the Longitudinal Redundancy Check.]

Dimensions (Top View) (mm)



	ID-2			ID-12			ID-20		
	Nom.	Min.	Max.	Nom.	Min.	Max.	Nom.	Min.	Max.
A	12.0	11.6	12.4	12.0	11.6	12.4	12.0	11.6	12.4
B	8.0	7.6	8.4	8.0	7.6	8.4	8.0	7.6	8.4
C	15.0	14.6	15.4	15.0	14.6	15.4	15.0	14.6	15.4
D	20.5	20.0	21.5	25.3	24.9	25.9	40.3	40.0	41.0
E	18.5	18.0	19.2	20.3	19.8	20.9	27.8	27.5	28.5
F	14.0	13.0	14.8	16.3	15.8	16.9	22.2	21.9	23.1
G	22.0	21.6	22.4	26.4	26.0	27.1	38.5	38.2	39.2
P	2.0	1.8	2.2	2.0	1.8	2.2	2.0	1.8	2.2
H	5.92	5.85	6.6	6.0	5.8	6.6	6.8	6.7	7.0
J	9.85	9.0	10.5	9.9	9.40	10.5	9.85	9.4	10.6
W	0.66	0.62	0.67	0.66	0.62	0.67	0.66	0.62	0.67

Note - measurements do not include any burring of edges.

NOTICE - Innovated Devices reserve the right to change these specifications without prior notice.

DATA FORMATS

Output Data Structure – ASCII

STX (02h)	DATA (10 ASCII)	CHECK SUM (2 ASCII)	CR	LF	ETX (03h)
-----------	-----------------	---------------------	----	----	-----------

[The 1-byte (2 ASCII characters) Check sum is the "Exclusive OR" of the 5 hex Bytes (10 ASCII) Data characters.]

Output Data Structure – Wiegand26

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
P	E	E	E	E	E	E	E	E	E	E	E	O	O	O	O	O	O	O	O	O	O	O	O	O	P
Even parity (E)												Odd parity (O)													

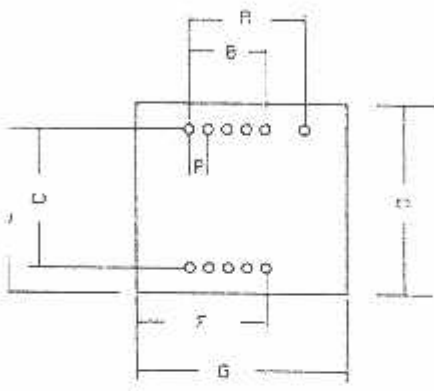
P = Parity start bit and stop bit

Output Data Magnetic ABA Track2

10 Leading Zeros	SS	Data	ES	LCR	10 Ending Zeros
------------------	----	------	----	-----	-----------------

[SS is the Start Character of 11010, ES is the end character of 11111, LRC is the Longitudinal Redundancy Check.]

Dimensions (Top View) (mm)



	ID-2			ID-12			ID-20		
	Nom.	Min.	Max.	Nom.	Min.	Max.	Nom.	Min.	Max.
A	12.0	11.6	12.4	12.0	11.6	12.4	12.0	11.6	12.4
B	8.0	7.6	8.4	8.0	7.6	8.4	8.0	7.6	8.4
C	15.0	14.6	15.4	15.0	14.6	15.4	15.0	14.6	15.4
D	20.5	20.0	21.5	25.3	24.9	25.9	40.3	40.0	41.0
E	18.5	18.0	19.2	20.3	19.8	20.9	27.8	27.5	28.5
F	14.0	13.0	14.8	16.3	15.8	16.9	22.2	21.9	23.1
G	22.0	21.6	22.4	26.4	26.1	27.1	38.5	38.2	39.2
P	2.0	1.8	2.2	2.0	1.8	2.2	2.0	1.8	2.2
H	5.92	5.85	6.6	6.0	5.8	6.6	6.8	6.7	7.0
J	9.85	9.0	10.5	9.9	9.40	10.5	9.85	9.4	10.6
W	0.66	0.62	0.67	0.66	0.62	0.67	0.66	0.62	0.67

Note: measurements do not include any burring of edges.

NOTICE - Innovated Devices reserve the right to change these specifications without prior notice.

DATA FORMATS

Output Data Structure – ASCII

STX (02h)	DATA (10 ASCII)	CHECK SUM (2 ASCII)	CR	LF	ETX (03h)
-----------	-----------------	---------------------	----	----	-----------

The 1 byte (2 ASCII characters) Check sum is the "Exclusive OR" of the 8 hex bytes (10 ASCII) Data characters.

Output Data Structure – Wiegand26

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	
P	E	E	E	E	E	E	E	E	E	E	E	E	O	O	O	O	O	O	O	O	O	O	O	O	O	P
Even parity (E)													Odd parity (O)													

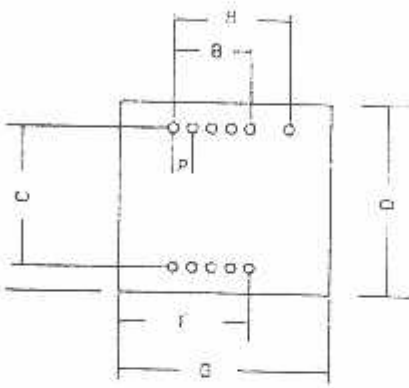
P = Parity start bit and stop bit

Output Data Magnetic ABA Track2

10 Leading Zeros	SS	Data	ES	LCR	10 Ending Zeros
------------------	----	------	----	-----	-----------------

SS is the Start Character of 11010. ES is the end character of 11111. LCR is the Longitudinal Redundancy Check.

Dimensions (Top View) (mm)

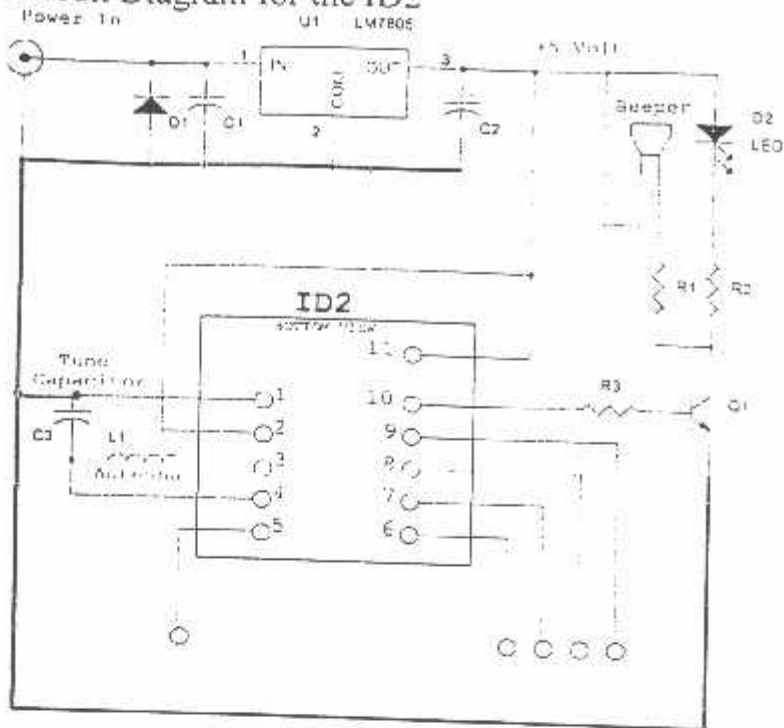


	ID-2			ID-12			ID-20		
	Nom.	Min.	Max.	Nom.	Min.	Max.	Nom.	Min.	Max.
A	12.0	11.6	12.4	12.0	11.6	12.4	12.0	11.6	12.4
B	8.0	7.6	8.4	8.0	7.6	8.4	8.0	7.6	8.4
C	15.0	14.6	15.4	15.0	14.6	15.4	15.0	14.6	15.4
D	20.5	20.0	21.5	25.3	24.9	25.9	40.3	40.0	41.0
E	18.5	18.0	19.2	20.3	19.8	20.9	27.8	27.5	28.5
F	14.0	13.0	14.8	16.3	15.8	16.9	22.2	21.9	23.1
G	22.0	21.6	22.4	26.4	26.1	27.1	38.5	38.2	39.2
P	2.0	1.8	2.2	2.0	1.8	2.2	2.0	1.8	2.2
H	5.92	5.85	6.6	6.0	5.8	6.6	6.8	6.7	7.0
J	9.85	9.0	10.5	9.9	9.40	10.5	9.85	9.4	10.6
W	0.66	0.62	0.67	0.66	0.62	0.67	0.66	0.62	0.67

Note – measurements do not include any burring of edges.

NOTICE – Innovated Devices reserve the right to change these specifications without prior notice.

Circuit Diagram for the ID2



COMPONENT LIST

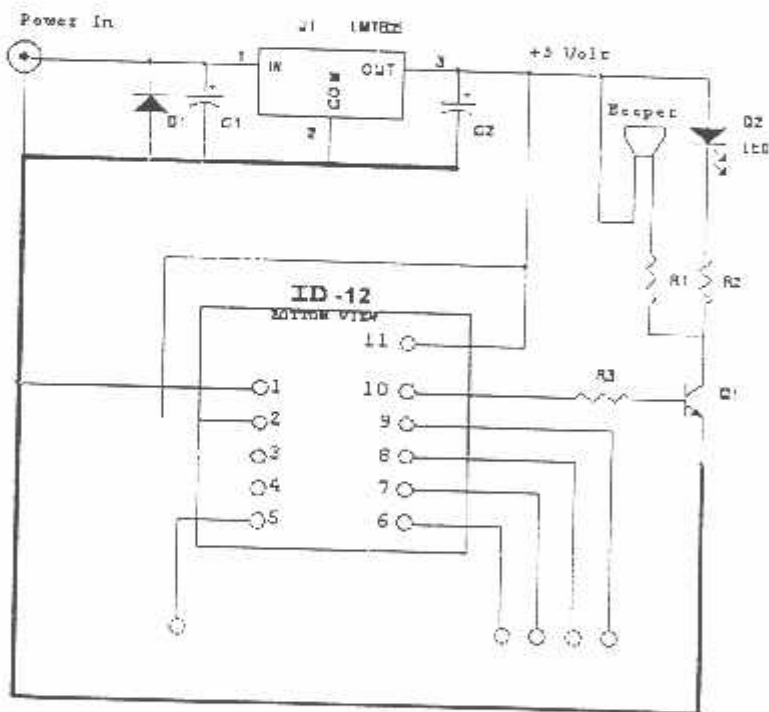
- R1 = 100R
- R2 = 1K
- R3 = 1K
- C1 = 100uF 16V
- C2 = 100uF 10V
- C3 = 1nF COG 100V *
- Beeper = 2.7-3.5KHz 100R
- D1 = 1N4001
- D2 = GREEN LED
- U1 = LM7805
- Q1 = UTC8050 (NPN)
- L1 = 640UH

ID2 = ID Innovations ID2

* Please Note the ID2 has an internal tuning capacitor of 1.5nF and this makes the total tuning capacity = 2.5nF

The 3.1KHz Beeper Logic is centered for most Beepers in range 2.7-3.5KHz

Circuit Diagram for the ID-12/ID20



COMPONENT LIST

- R1 = 100R
- R2 = 1K
- R3 = 1K
- C1 = 100uF 16V
- C2 = 100uF 10V
- Beeper = 2.7-3.5KHz 100R
- D1 = 1N4001
- D2 = GREEN LED
- U1 = LM7805
- Q1 = UTC8050 (NPN)
- ID2 = ID Innovations ID2

* Please Note the ID2 has an internal tuning capacitor of 1.5nF and this makes the total tuning capacity = 2.5nF

The 3.1KHz Beeper Logic is centered for most Beepers in range 2.7-3.5KHz

ID Innovations

Advanced Digital Reader Technology

--- Better by Design

Designing Coils for ID2

The recommended Inductance is 1.08mH to be used with an internal tuning capacitor of 1n5. In general the bigger the antenna the better, provided the reader is generating enough field strength to excite the tag. The ID-2 is relatively low power so a maximum coil size of 15x15cm is recommended if it is intended to read ISO cards. If the reader is intended to read glass tags the maximum coil size should be smaller, say 10x10cm.

There is a science to determine the exact size of an antenna but there are so many variables that in general it is best to get a general idea after which a degree of 'Try it and see' is unavoidable.

If the reader is located in a position where there is a lot of heavy interference then less range cannot be avoided. In this situation the coil should be made smaller to increase the field strength and coupling.

It is difficult to give actual examples of coils for hand winding because the closeness and tightness of the winding will significantly change the inductance. A professionally wound coil will have much more inductance than a similar hand wound coil.

For those who want a starting point into practical antenna winding it was found that 63 turns on a 120mm diameter former gave an inductance of 1.08mH. For those contemplating adding an additional tuning capacitor it was found that 50 turns on a 120mm diameter former gave 700uH. The wire diameter is not important.

Anybody who wishes to be more theoretical we recommend a trip to the Microchip Website where we found an application sheet for Loop Antennas, <http://www1.microchip.com/downloads/en/AppNotes/00831b.pdf>

The Tuning Capacitor

It is recommended that the internal 1n5 capacitor is used for tuning, however a capacitor may be also be added externally. The combined capacitance should not exceed 2n7. Do not forget that the choice of tuning capacitor can also substantially affect the quality of your system. The ID12 is basically an ID2 with an internal antenna. The loss in an ID12 series antenna is required to be fairly high to limit the series current. A low Q will hide a lot of the shortcomings of the capacitor, but for quality and reliability and repeatability the following capacitors are recommend.

Polypropylene	Good Readily available. Ensure AC voltage at 125kHz is sufficient.
COG/NPO	Excellent. Best Choice
Silver Mica	Excellent but expensive
Polycarbonate	Good Readily available. Ensure AC voltage at 125kHz is sufficient.

Voltage Working.

A capacitor capable of withstanding the RMS voltage at 125KHz MUST be chosen. The working voltage will depend on the coil design. I suggest the designer start with rugged 1n5 Polypropylene 630v capacitor to do his experiments and the come down to a suitable size/value. The capacitor manufacturer will supply information on their capacitors. Do not simply go by the DC voltage. This means little. A tolerance of 2% is preferable. A tolerance of 5% is acceptable.

Fine Tuning

We recommend using an oscilloscope for fine-tuning. Connect the oscilloscope to observe the 125KHz AC voltage across the coil. Get a sizeable piece of ferrite and bring it up to the antenna loop. If the voltage increases then you need more inductance (or more capacitance). If the voltage decreases as you bring the ferrite up to the antenna then the inductance is too great. If you have no ferrite then a piece of aluminum sheet may be used for testing in a slightly different way. Opposing currents will flow in the aluminum and it will act as a negative inductance. If the 125kH AC voltage increases as the aluminum sheet approaches the antenna then the inductance is too high. Note it may be possible that the voltage will first maximize then decrease. This simply means that you are near optimum tuning. If you are using ferrite then the coil is a little under value and if you are using an aluminum sheet then the coil is a over under value.

ID Innovations

Advanced Digital Reader Technology

---Better by Design