SKRIPSI

PERANCANGAN DAN PEMBUATAN ALAT DAFTAR KIR KENDARAAN BERMOTOR BERBASIS SMARTCARD DENGAN MENGGUNAKAN MODUL FT232BM-USB YANG DIKENDALIKAN OLEH R8C/13 TINY

Disusun Oleh :

EDDY NUR IMAMSYAH

NIM : 03.17.018

JURUSAN TEKNIK ELEKTRO S - 1

KONSENTRASI ELEKTRONIKA

FAKULTAS TEKNOLOGI INDUSTRI

INSTITUT TEKNOLOGI NASIONAL MALANG

MARET 2008

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN ALAT DAFTAR KIR KENDARAAN BERMOTOR BERBASIS SMARTCARD DENGAN MENGGUNAKAN MODUL FT232BM-USB YANG DIKENDALIKAN OLEH R8C/13 TINY

SKRIPSI

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

Disusun Oleh :

EDDY NUR IMAMSYAH
NIM : 03.17.018

Diperiksa dan Disetujui

Dosen Pembimbing I	Dosen Pembimbing II
Ir. F. Yudi Limpraptono, MT	**Joseph Dedy Irawan, ST,MT**
NIP.Y 1039500274	**NIP.132315178**

Mengetahui

Ketua Jurusan Teknik Elektro S-1

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

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

2008

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

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama	: EDDY NUR IMAMSYAH
NIM	: 03.17.018
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Judul Skripsi	: Perencanaan dan Pembuatan Alat Daftar KIR Kendaraan Bermotor Berbasis Smartcard Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C/13 Tiny.

Dipertahankan di hadapan majelis penguji Skripsi jenjang Strata satu (S-1) pada :

Hari	: Sabtu
Tanggal	: 15 Maret 2008
Dengan Nilai	: 84 (A) 84

INSTITUT TEKNOLOGI NASIONAL MALANG

Ketua Majelis Penguji

(Ir. Mochtar Asroni, MS,ME)
NIP.Y.1018100036

Sekretaris Majelis Penguji

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

Penguji I

(Ir. M. Abdul Hamid, MT)
NIP.Y.1018800188

Penguji II

(I Komang Somawirata, ST, MT)
NIP.P.1030100361

ABSTRAKSI

PERANCANGAN DAN PEMBUATAN ALAT DAFTAR KIR KENDARAAN BERMOTOR BERBASIS SMARTCARD DENGAN MENGGUNAKAN MODUL FT232BM-USB YANG DIKENDALIKAN OLEH R8C/13 TINY

Eddy Nur Imamsyah
03.17.018

Jurusan Teknik Elektronika – Institut Teknologi Nasional Malang
Jln. Raya Karanglo Km 2 Malang
Eddy_elka03@yahoo.com

Dosen Pembimbing : I. Ir.F.Yudi Limpraptono,MT.
II. Joseph Dedy Irawan,ST,MT.

Kata Kunci : Smartcard,Mikrokontroller R8C/13 tiny series ,Modul USB,PC.

Perkembangan ilmu pengetahuan dan teknologi semakin cepat. Berdasarkan perkembangan teknologi ini kita harus bisa memberikan informasi dengan cepat sesuai dengan data yang dibutuhkan Misalkan kita ingin memberikan informasi tentang data – data kendaraan dalam memenuhi wajib lapor kendaraan setiap beberapa tahun sekali yaitu seperti memberikan informasi mengenai jenis kendaraan,tahun pembuatan kendaraan,nomor mesin,nomor rangka,merk/type kendaraan dan lain - lain.Dari gagasan tersebut diatas maka muncul suatu pemikiran untuk membuat sebuah sistem yang dapat memberikan informasi yang dibutuhkan dengan cepat tanpa harus mengecek data – data yang ada di dalam arsip kendaraan.Sistem kendali utamanya adalah sebuah mikrokontroller yang diberi masukan kartu smartcard yang diberi data – data informasi dari kendaraan tersebut yang kemudian disamakan dengan data yang ada pada database,dalam pengujian yang dikalukan pada sitem ini yaitu dengan melakukan pengisian serta terhadap lima buah kartu smartcard kemudian kartu tersebut debaca berulang-ulang sebanyak 25 kali,selain itu pengujian juga mengunakan kabel perpanjanagan USB dengan panjang 3 meter dimana hasil yang didapat dari pengujian tersebut tidak terdapat eror baik dalam proses pembacaan maupun dalam proses penulisan.

KATA PENGANTAR

Alhamdulillah, puji syukur kehadirat-Mu Ya Allah yang telah memberikan rahmat dan hidayah-Nya, sehingga dapat menyelesaikan skripsi yang berjudul "Perancangan dan Pembuatan Alat Daftar Kir Kendaraan Bermotor Berbasis SMARTCARD Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C/13 TINY" ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika ITN Malang dan untuk mencapai gelar Sarjana Teknik.

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

1. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1.
2. Bapak Ir. F. Yudi Limpraptono, MT selaku Dosen PembimbingI.
3. Bapak Joseph Dedy Irawan, ST, MT Dosen Pembimbing II dan Ka. Laboratorium Elektronika Digital.
4. Ayah dan Ibu serta saudara-saudara kami yang telah memberikan do'a restu, dorongan, semangat, dan biaya.
5. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

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

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

Malang, April 2008

Penyusun

DAFTAR ISI

DAFTAR GAMBAR

❖ Timer Mode

Mempunyai timer sebanyak 4 yaitu timer X,Y,Z,C. berikut adalah mode – mode timernya :

Table 2.2 Mode – mode Timer[3].

Item		Timer X	Timer Y	Timer Z	Timer C
Configuration		8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	8-bit timer with 8-bit prescaler	16-bit free-run timer
Count		Down	Down	Down	Up
Count source		• f_1 • f_2 • f_8 • f_{32}	• f_1 • f_8 • f_{RING} • Input from $CNTR_1$ pin	• f_1 • f_2 • f_8 • Timer Y underflow	• f_1 • f_8 • f_{32} • $f_{RING\text{-}fast}$
Function	Timer mode	provided	provided	provided	not provided
	Pulse output mode	provided	not provided	not provided	not provided
	Event counter mode	provided	provided(1)	not provided	not provided
	Pulse width measurement mode	provided	not provided	not provided	not provided
	Pulse period measurement mode	provided	not provided	not provided	not provided
	Programmable waveform generation mode	not provided	provided	provided	not provided
	Programmable one-shot generation mode	not provided	not provided	provided	not provided
	Programmable wait one-shot generation mode	not provided	not provided	provided	not provided
	Input capture mode	not provided	not provided	not provided	provided
	Output compare mode	not provided	not provided	not provided	provided
Input pin		$CNTR_0$	$CNTR_1$	$\overline{INT_0}$	TC_{IN}
Output pin		$CNTR_0$ $\overline{CNTR_0}$	$CNTR_1$	TZ_{OUT}	$CMP0_0$ to $CMP0_2$ $CMP1_0$ to $CMP1_2$
Related interrupt		Timer X int $\overline{INT_1}$ int	Timer Y int $\overline{INT_2}$ int	Timer Z int $\overline{INT_0}$ int	Timer C int $\overline{INT_3}$ int compare 0 int compare 1 int
Timer stop		provided	provided	provided	provided

❖ Low Voltage Detect (LVD)

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

❖ Watchdog Timer

Watchdog berfungsi untuk mendeteksi ketika program di luar control.

- **On Chip Debugger**

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

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

- **Rangkaian Osilator**

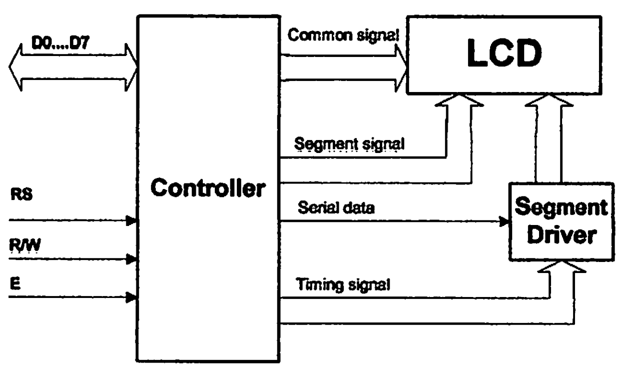
  Pada osilator utama menggunakan Kristal luar sampai dengan 20MHz,dengan memilii fitur Clock Stop Detect.kemudian untuk On Chip Osilator disediakan kecepatan Low 125 Khz dan High 8Mhz.saat setelah reset,default clock adalah kecepatan rendah On Chip osilator 125 KHz.

2.3 LCD (Liquid Crystal Display) M1632

Modul peraga yang digunakan dalam aplikasi ini adalah LCD modul M1632. Modul LCD ini membutuhkan daya yang kecil dan dilengkapi dengan panel LCD dengan tingkat kontras yang cukup tinggi serta pengendali LCD

CMOS yang terpasang dalam modul tersebut. Pengendali mempunyai pembangkit karakter ROM/RAM dan display data RAM. Semua fungsi display diatur oleh instruksi-instruksi, sehingga modul LCD ini dapat dengan mudah dihubungkan dengan unit mikroprosesor. LCD tipe ini tersusun sebanyak dua baris dengan 16 karakter.

Masukan yang diperlukan untuk mengendalikan modul berupa bus data yang masih termutiplek dengan bus alamat serta 3 bit sinyal kontrol. Sementara pengendalian LCD dilakukan secara internal oleh kontroler yang sudah terpasang dalam modul LCD. Diagram blok untuk LCD dapat dilihat dalam Gambar 2.11



Gambar 2.11 Diagram blok LCD M1632[4].

LCD modul M1632 mempunyai spesifikasi perangkat keras sebagai berikut:

- 16 karakter dan 2 baris tampilan yang terdiri dari 5x7 dot matrik ditambah kursor
- Pembangkit karakter ROM untuk 192 jenis karakter
- Pembangkit karakter RAM untuk 8 jenis karakter
- 80 x 8 display data RAM (maksimum 8 karakter)

- ➢ Osilator internal
- ➢ Catu daya +5volt
- ➢ Secara otomatis akan reset saat catu daya dinyalakan

LCD M1632 mempunyai 16 pin atau penyemat yang mempunyai fungsi-fungsi seperti ditunjukkan dalam Tabel 2.3 :

Tabel 2.3 Fungsi pin-pin LCD M1632[4].

No	Nama Penyemat	Fungsi
1	Vss	Terminal ground
2	Vcc	Tegangan catu +5 volt
3	Vee	Drive LCD
4	RS	Sinyal pemilih register 0: Instruksi register (tulis) 1: Data Register (tulis dan baca)
5	R/W	Sinyal seleksi tulis atau baca 0: Tulis 1: Baca
6	E	Sinyal operasi awal, sinyal ini mengaktifkan data tulis dan baca
7 – 14	DB0-DB7	Merupakan saluran data, berisi perintah dan data yang akan ditampilkan
15	V+ BL	Pengendali kecerahan latar belakang LCD 4 - 4,42 V dan 50 – 500 mA
16	V-BL	Pengendali kecerahan latar belakang LCD 0 V

2.4 Smartcard

Smartcard adalah sebuah kartu yang memiliki memory yang dapat menyimpan data-data.Kartu chip secara umum di golongkan menjadi dua macam berdasarkan penggunaanya yaitu :

- Kartu chip yang lebih berupa *memory*.
- Kartu chip yang mempunyai *microprocessor*.

Untuk jenis kartu chip pertama lebih sering digunakan sebagai kartu telpon,kartu internet dan fasilitas – fasilitas prabayar yang tidak memerlukan informasi penguna kartu.Kartu jenis ini fungsinya lebih ditekankan pada penyimpanan sejumlah kredit atau poin untuk transaksi-transaksi,contohnya pada kartu telpon chip.Pada beberapa jenis kartu yang baru telah dilengkapi dengan enkripsi atau pin identifikasi yang seperti ini biasa disebut dengan *smart memorycard.*

Sedangkan untuk jenis kartu chip kedua adalah kartu yang lebih cangih yang dilengkapi mikroprosessor untuk fungsi-fungsi tertentu,selain menyimpan identitas pemilik kartu juga dapat memproses sejumlah data.Kartu jenis ini juga memiliki memory berupa ROM dan EEPROM,data dilindungi dengan algoritma enkripsi.jenis ini umum disebut dengan *smartcard* atau kartu cerdas (walau sebenarya *memorycard* adalah juga merupakan *smartcard*).

Spesifikasi kartu telpon chip generasi pertama :

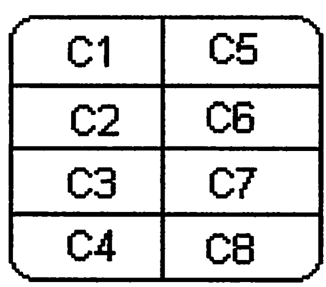
- Synchronous protocol
- Teknologi N-Mos atau CMOS untuk yang lebih baru.
- Organisasi memory 256 x 1 bit.
- 96 bit proteksi penulisan dengan clock out fuse.
- Pemakaian daya rendah 85 mW pada read mode.
- 21 volt programming voltage.
- Acces time 500 ms.
- Operating temperature range – 10°C sampai +70°C.
- Deteksi data hingga 10 thn.

Sedangkan untuk generasi kedua adalah

- ISO 7816 – ½ compatible protocol.
- Penggunaan tegangan supply tunggal 5 volt.
- Teknologi NMOS.
- Konsumsi daya rendah.

2.4.1 Posisi Pin pada Smartcard.

Posisi pin pada smartcard mengunakan standart ISO7816 dimana letaknya akan diperhatikan pada gambar berikut :



Gambar 2.12 Konfigurasi Pin Smartcard[8].

Fungsi kontak-kontak pada gambar diatas adalah :

- C1 digunakan untuk *input power supply* (Vcc) dari piranti antarmuka.
- C2 untuk RST dan digunakan oleh piranti antarmuka untuk mengirim sinyal reset ke mikrosirkit kartu.
- C3 untuk *clock* (CLK) dan sinyal-sinyal pewaktuan dikirimkan ke kartu melalui C3.
- C5 sebagai tegangan referensi (GND), nilai tegangan itu dianggap 0 volt.
- C7 menyelenggaraakan komunikasi ke dan dari kartu, dan disebut I/O.
- C4, C6 dan C8 tidak digunakan.

2.4.2 System komunikasi Data Smartcard.

Komunikasi antara Smartcard dengan interface devais (reader) melalui beberapa proses yaitu :

- Kontak dan Aktivasi oleh reader.
- Reset dari smartcard.
- Answer to Reset (ATR) oleh smartcard.
- Pertukaran data antara smartcard dan reader.
- Deaktivasi kontak oleh reader.

2.4.2.1 kontak dan aktivasi oleh reader.

Untuk menghindari kerusakan yang mungkin terjadi pada kartu yang disebabkan oleh reader pada saat kartu dimasukan maka sirkuit diharapkan tidak diaktifkan dahulu sebelum pin - pin pada kartu benar – benar terhubung dengan readernya.adapun didalam mengaktifkan reader harus memperhatikan beberapa hal berikut yaitu :

- Reset berada pada status " low".
- Vcc telah diberikan tegangan tertentu.
- I/O pada reader berada pada reception mode.
- CLK harus diberikan clock yang sesuai dan stabil.

2.4.2.2 Reset Oleh Smartcard.

Setiap card reset dikenali oleh reader ,dimana kartu harus merespon dengan ATR seperti yang dijelaskan pada bagian berikut

Pada saat akhir aktivasi kontak oleh reader (RST berada pada state "low" ,Vcc telah diberi tegangan tertentu,I/O pada reader berada pada *reception mode*

,CLK harus diberi clock yang sesuai dan stabil),maka kartu akan merespon secara asinkron bahwa card siap di reset.

Jika kartu merespon secara sinkron,seperti pada kartu kredit misalnya maka semua pin berada pada posisi "low" Vcc telah diberikan pada tegangan tertentu,Vpp diset pada *idle state*,CLK dan RST berada tetap pada "Low",I/O berada pada *reception mode*,Reset harus berada pada kondisi "High" sekurang – kurangnya 50 us sebelum kembali kekondisi "Low".

2.4.2.3 Answer To Reset (ATR).

Terdapat dua buah tipe transmisi pengiriman data pada saat answer to reset yaitu

- Pengiriman secara asinkron

  Karakter yang dikirim melalui I/O (half duplex) secara asinkron dimana setiap karakter berupa Byte (8 bit).

- Pengiriman secara Sinkron.

  Sekelompok bit yang dikirim secara half duplex melalui clock pada CLK.

Smartcard digunakan sebagai kartu GSM atau kartu telpon biasanya megunakan transmisi asinkron,sedangkan transmisi sinkron biasanya digunakan pada kartu kredit.

2.4.2.4 Deaktivasi Kontak oleh Reader.

Yaitu Jika pertukaran data putus atau dibatalkan yang disebabkan semisalnya oleh pengambilan kartu atau kartu yang tidak merespon maka kontak pada pin harus dinonaktifkan,adapun proses tersebut meliputi :

- RST pada kondisi "Low".

- CLK pada Kondisi "Low".
- Vcc tidak diberi tegangan.

2.4.3 Spesifikasi dari modul ACR30SP ini yaitu

1. Memiliki tegangan kerja 5 volt DC.
2. Tersedia 1 LED sebagai indicator dalam mendeteksi adanya kartu smartcard.
3. Interfacing dengan komunikasi serial RS-232 atau USB.
4. Mendukung kartu mikroprosesor dengan protokol T=0 dan T=1.
5. ISO 7816 compatible Smartcard interface.
6. Mendukung kartu memori SLE4432 dan SLE 4442.

Adapun gambar fisik dari modul AC30SP dapat digambarkan sebagai berikut:

Gambar 2.13 Modul ACR30SP[6].

Pada modul ACR30 ini terhubung memalui interface serial asinkron yang mengacu pada protocol RS-232.dengan parameter-parameter sebagai berikut :

- Protocol transmisi : Serial asinkron
- Bit paritas : Tidak ada
- Bit data : 8 bit
- Handshake : 1 bit
- Baudrate : 9600 bps

2.4.4 Protokol Komunikasi

Dalam kondisi normal, ACR30 berlaku sebagai perangkat pasif. Maksudnya adalah, ACR30 menunggu *command* dari PC atau mikrokontroller, menerima command dan mengirim respons setelah mengeksekusi command tersebut. Command berikutnya hanya bisa ditransmisikan setelah command sebelumnya selesai dieksekusi. Ada dua kondisi dimana ACR30 mengirim respons tanpa menunggu command dari PC, yaitu pada saat mengirim *reset message* dan *card status message*.

2.4.4.1 Command

Adapun perintah – perintah yang di berikan kepada mikrokontroler dalam mengakses reader smartcard ACR30 :

➢ **Normal command (kurang dari 255 byte)**

Normal command terdiri dari 4 byte protokol dan N byte data (variabel) dengan struktur sebagai berikut :

byte 1	2	3	1≤N≤255	4
HEADER	INSTRUCTION	DATA LENGTH (N)	DATA	CHECKSUM

Gambar 2.14 Normal Command[6].

Dengan perincian masing-masing byte sebagai berikut :

- Header adalah Start of command (01h)
- Instruction adalah Kode instruksi
- Data Length adalah Panjang data atau variabel
- Data Data, variabel atau parameter

- Checksum adalah perintah untuk melakukan operasi XOR pada semua byte mulai dari start of command (Header),instruction,data linght dan semua data.

➢ **Select_Card_Type**

Dalam pembacaan kartu oleh reader ACR30 dibutuhkan sebuah perintah didalam mengakses kartu smartcard tersebut dengan format perintahnya yaitu :

Instruction Code	Data length	Data
		TYPE
02_H	01_H	

Gambar 2.15 Format Select_Card_Type[6].

TYPE : merupakan tambahan data untuk perintah menggenai keterangan jenis kartu yang digunakan.

2.4.5 Kartu Smartcard

Untuk jenis kartu yang digunakan dalam penyusunan tugas akhir ini adalah type SLE 4442 dengan kapasaitas memory kartu 256 byte.

Adapun perintah – perintah yang dikeluarkan reader dalam mengaakses kartu ini yaitu:

1. Reset

   Di dalam melaksanakan reset pada saat kartu dimasukkan ke dalam reader harus dapat dilakukan dengan cara mingikuti format berikut:

Format perintah

Instruction Code	Data length
80 H	00 H

Gambar 2.16 Format Reset [6].

2. Power_Off

   Dalam mengakifkan reader pada saat kartu dimasukkan dapat mengunakan format sebagai berikut :

   Format perintah

Instruction Code	Data length
81 H	00 H

Gambar 2.17 Format Power_off [6].

3. Read_data

   Pembacaan kartu dilakukan dengan jalan melakukan perintah pada saat kartu dimasukkan yaitu :

   Format perintah

Instruction Code	Data length	Data		
		ADDR		LEN
90 H	03 H			

Gambar 2.18 Format Read_Data[6].

ADDR : Alamat byte pada saat pembacaan kartu.

LEN : Jumlah data N yang dibaca dari kartu ($0 < N \leq MAX\_R$)

4. Write_Data

   Pada saat kartu dimasukkan penulisan dilakukan pada alamat tertentu dengan format sebagai berikut :

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1. Latar Belakang

Informasi merupakan suatu hal yang penting bagi kehidupan manusia. Ada banyak informasi yang dapat kita ketahui dengan cara-cara tertentu,seperti misalnya untuk mengetahui daftar kir suatu kendaran dan kemudian menyamakannya dengan data dari kendaraan tersebut pada arsip atau juga pada saat melakukan suatu pemeriksaan kendaraan dengan batas waktu untuk melakukan uji kelayakan kendaraan berikutnya,sehingga semua itu dapat dilakukan secara mudah dengan mengunakan sebuah kartu yang berisi data – data tersebut.

Dengan berkembangnya teknologi yang merupakan suatu sistem komunikasi yang *reliable*, maka pemanfaatan teknologi sebuah kartu smartcard dapat digunakan sebagai media untuk mengetahui secara cepat informasi yang diinginkan. Smartcard diharapkan dapat menjadi sebuah media untuk mengetahui semua informasi yang diharapkan dengan cepat dan tepat sehingga dapat mengurangi kesalahan pada waktu melakukan pemeriksaan seminimal mungkin dalam hal ini komputer juga berperan sangat penting.

Hal tersebut yang menjadi dasar pada tugas akhir ini untuk menciptakan suatu sistem yang dapat dengan cepat memberikan informasi mengenai kendaraan tersebut sehingga semua dapat diketahui secara cepat .

1.2. Rumusan Masalah

Mengacu pada permasalahan yang ada, maka dalam perencanaan dan pembuatan alat ini diutamakan pada hal-hal sebagai berikut :

1. Bagaimana merancang system pembuatan Alat Daftar Kir Kendaraan Bermotor.
2. Bagaimana cara menghubungkan antara modul FT232BM-USB , Mikrokontroller dan SMARTCARD.
3. Bagaimana merencanakan dan membuat sebuah program pada Mikrokontroller serta mengunakan delphi sebagai tampilan dari informasi.

1.3. Batasan Masalah

Dalam laporan akhir "Perancangan dan Pembuatan Alat Daftar Kir Kendaraan Bermotor Berbasis SMARTCARD Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C/13TINY", penulis akan memberikan batasan-batasan masalah agar tidak terjadi penyimpangan maksud dan tujuan utama penyusunan skripsi ini.

1. Mikrokontroller yang digunakan adalah R8C/13 TINY.
2. Program yang digunakan sebagai tampilan pada PC adalah Delphi.
3. Type Smartcard yang digunakan adalah ACR30.
4. Database yang dipakai adalah database Paradox.

1.4. Tujuan

Tujuan dari penulisan skripsi ini adalah untuk merancang dan membuat Alat Daftar Kir Kendaraan Bermotor Berbasis SMARTCARD Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C/13 TINY.

1.5. Metodelogi Penelitian

Guna merealisasikan Tugas Akhir sebagai tesebut diatas, maka metodelogi penulisan yang digunakan adalah sebagai berikut :

1. Studi Pustaka

   Memperoleh data dengan cara membaca dan mempelajari buku *literature* yang berhubungan dengan penyusunan skripsi ini.

2. Studi Lapangan

   Memperoleh data dengan cara praktek secara lansung untuk menunjang pembuatan alat.

3. Pengujian alat

   Melakukan pengujian permodul untuk memastikan bahwa masing-masing modul telah bekerja dengan baik

1.6. Sistematika Penulisan

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

BAB I Pendahuluan

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

BAB II Landasan Teori

Pada bab ini dibahas tentang teori-teori yang mendukung dalam perencanaan dan pembuatan alat.

BAB III Perencanaan Dan Pembuatan Alat

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

BAB IV Pengujian Alat

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

BAB V Penutup

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

BAB II

LANDASAN TEORI

Landasan teori sangat membantu untuk dapat memahami suatu sistem. Landasan teori juga dapat digunakan sebagai acuan di dalam merencanakan suatu sistem. Dengan pertimbangan hal-hal tersebut, maka landasan teori merupakan bagian yang harus dipahami untuk pembahasan lebih lanjut. Dalam landasan teori ini akan dibahas teori dasar yang berhubungan dengan *FT232BM-USB, Mikrokontroler Renesas R8C/13 TINY,LCD,SmartCard ACR30 dan MAX232.*

2.1 Modul FT232BM-USB

Perkembangan teknologi komputer dimana seiring dengan perkembangan ilmu pengetahuan yang semakin cepat pada PC atau Laptop – laptop untuk keluaran terbaru jumlah port serial RS232 semakin lama semakin berkurang,jika pada PC lama biasanya terdapat dua buah konektor RS232 maka sekarang hanya terdapat satu buah konektor saja sehingga keberadaan port serial RS232 sekarang telah digantikan oleh port USB yang mempunyai banyak kelebihan dibandingkan port serial RS232.adapun contoh pengiriman informasi secara serial melalui sebuah mikrokontroler yang dikirimkan ke PC melalui port USB seperti Modul FT232BM-USB yaitu modul interface yang digunakan untuk aplikasi dari mikrokontroller ke USB,hubungan ini dilakukan secara serial dengan kata lain sebuah modul yang dapat mengkonversikan data serial yang berasal dari mokrokontroller.secara garis besar modul ini berfungsi untuk mengubah data USB yang berasal dari port USB menjadi data serial dengan level tegangan TTL sehingga pengguna dapat melakukan komunikasi data serial (UART) melalui port USB. Keunggulan digunakannya modul ini adalah kemampuannya untuk mengirim data lebih cepat dibandingkan dengan komunikasi serial dengan mengunakan Port RS232 serta untuk kecepatan transfer data serial yang dapat di pakai oleh IC ini yaitu sebesar 300bps sampai 9600 bps.

Modul FT232BM-USB mempunyai spesifikasi sebagai berikut:

1. Memiliki tegangan kerja 4,4- 5,25 volt DC.
2. Tersedia 2 LED untuk indicator Tx dan Rx data pada komunikasi serial.
3. Memiliki boudrate 3Mbps (TTL), 1Mbps (RS-232), 3Mbps (RS-422/RS-485).
4. Pin sinyal kontrol (arah) untuk komunikasi RS-485 yang bekerja secara otomatis.
5. Kompatibel dengan USB 1.1 dan USB 2.0.

   Modul mengunakan konfigurasi daya Self Powered.
6. Memiliki output dengan level TTL 5 volt .
7. Memiliki EEPROM eksternal untuk menyimpan data PID,VID,nomor serial,dan deskripsi produk.pengisian datanya melalui USB.
8. Virtual COM port driver (VCP) dan D2xx (USB Direct Drivers + DLL S/W Interface) untuk windows 98,98SE,ME,2000 dan XP.
9. Mendukung format UART dengan 7/8 bit data , 1 / 2 stop bit dan Odd / Even / Mark / Space / No Parity.

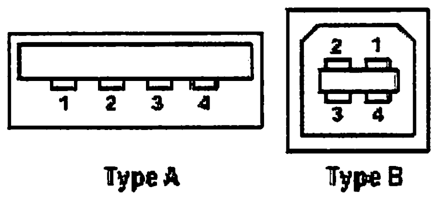
Adapun gambar fisik dari modul FT232BM-USB dapat digambarkan sebagai berikut:

Gambar 2.1 Modul FT232BM-USB[1]

Gambar diatas merupakan modul dari FT232BM-USB secara fisik yang terdiri dari konektor USB TIPE B dan chip FT232BM sebagai komponen utama dari modul COM to USB tersebut serta komponen-komponen pendukung lainnya.

2.1.1 Konektor USB (Universal Serial Bus).

Di dalam menghubungkan modul FT232BM-USB dengan komputer maka dibutuhkan kabel konektor USB dimana hanya terdapat ada 2 macam konektor yang digunakan dalam menghubungkan modul tersebut, yaitu konektor type A dan konektor type B seperti terlihat dalam Gambar 1. Konektor type A dipakai untuk menghubungkan kabel USB ke terminal USB yang ada pada bagian computer sedangkan Konektor type B dipakai untuk menghubungkan kabel USB ke terminal USB yang ada pada modul FT232BM-USB sedangkan untuk peralatan USB yang sederhana, misalnya mouse, biasanya tidak pakai konektor B melainkan konektor tipe A hal ini dilakukan untuk menghemat biaya sehingga kabel langsung dihubungkan ke bagian dalam mouse.

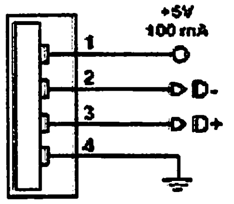


Gambar 2.2 Konektor USB[2]

Kabel USB terdiri dari 4 kabel ditambah konduktor pembungkus kabel seperti pelindung yang biasanya dijumpai dalam kabel audio.

Kabel nomor 1 dipakai untuk menyalurkan sumber daya dengan tegangan 5 Volt, jika diperlukan peralatan USB boleh mengambil daya dari saluran ini tidak lebih dari 100 mA. Komputer yang dilengkapi dengan kemampuan USB, wajib menyediakan daya sebesar 500 mA untuk keperluan ini. Peralatan USB yang memerlukan daya lebih dari ketentuan tersebut di atas, harus menyediakan sendiri sumber daya untuk keperluan kerja peralatan tersebut.

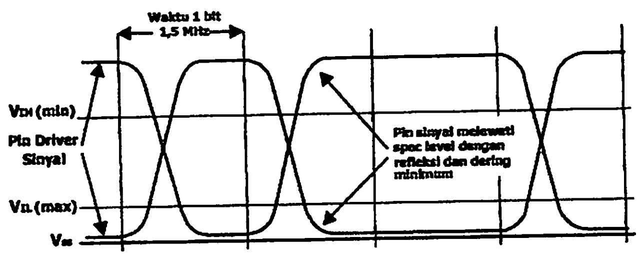
Kabel nomor 4 adalah ground sebagai saluran balik sumber tegangan 5 Volt.Kabel nomor 2 dan nomor 3 dipakai untuk pengiriman sinyal. Kabel nomor 2 bernama D- dan kabel nomor 3 bernama D+, tegangan pada dua saluran ini berubah antara 0 Volt dan 3,3 Volt seperti terlihat pada gambar 2 berikut:



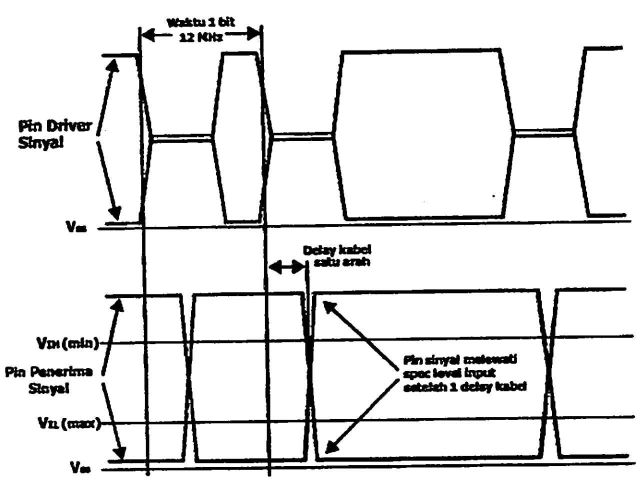
Gambar 2.3 Konektor Pin USB Tipe A[2]

2.1.2 Karakteristik USB (Universal Serial Bus).

Rentang tegangan USB adalah 0,3 volt hingga 3,6 volt (pada beban 1,5 kΩ).logika tinggi didapat jika tegangan sudah melebihi 2,8 volt terhadap ground pada beban 15kΩ.pada piranti USB yang berkecepatan rendah dan penuh untuk deferensial "1" dikirim dengan menarik D+ hingga lebih besar dari 2,8 volt dengan sebuah resistor 15 kΩ terhubung ke ground dan sekaligus menarik D- hinga dibawah 0,3 volt dengan dengan sebuah resistor 1,5kΩ terhubung ke 3,6 lebih rendah dari 0,3 volt dengan resistor pull-up dan pull-down yang sama.Di bagian penerima ,deferensial "1" dideferensialkan sebagai D+ lebih besar 200mV dari D-,dan deferensial "0" berarti D+ lebih kecil dari 200mV dibanding D-.Pada USB berkecepatan tinggi (480MBit/s) digunakan sumber arus tetap mA untuk mengurangi *noise*. Adapun bentuk sinyal gelombang pengiriman USB dengan kecepatan tinggi maupun rendah yang ditunjukkan pada gambar 2.4a dan 2.4b :



Gambar 2.4a Gelombang sinyal USB 1,5 MHz[2].

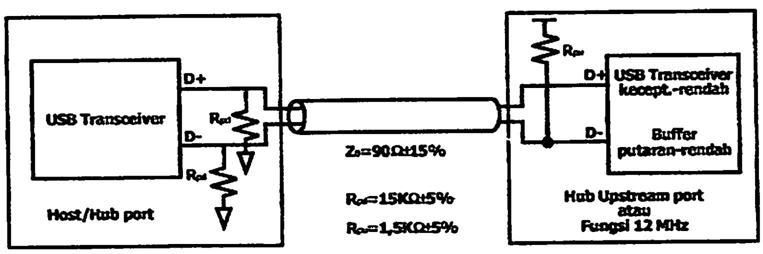


Gambar 2.4b Perkabelan USB Kecepatan tinggi 12MHz[2].

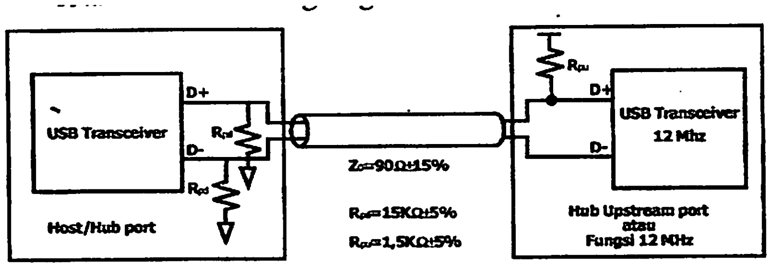
Kecepatan piranti USB dibagi menjadi dua yaitu kecepatan rendah (low-speed) dan kecepatan tinggi (full-speed) yang dihubungkan dengan cara koneksi resistor dan kabel USB oleh posisi resistor *pull-up* di ujung kabel *downstream* yaitu sebagai berikut :

- Piranti *full-speed* diterminalkan dengan resistor pull-up terhubung di D+ seperti gambar 2.5a.
- Piranti low-speed diterminalkan dengan resistor pull-up terhubung di D-,seperti gambar 2.5b.

- Terminator pull-down di port down-stream adalah resistor 15kΩ ±5% terhubung ke ground.



Gambar 2.5a Perkabelan USB Kecepatan Rendah[2].

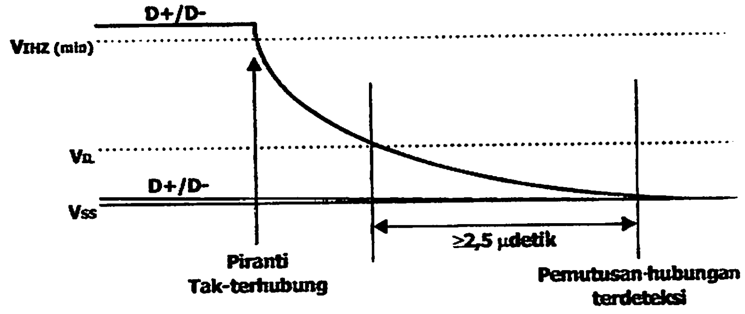


Gambar 2.5b Perkabelan USB kecepatan penuh[2].

2.1.3 Sinyal Penyambung dan Pemutusan

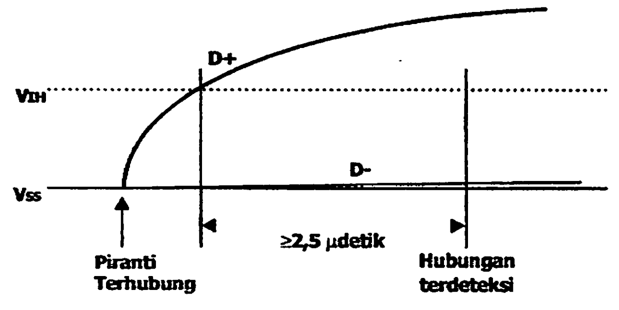
Pada sinyal penyambungan dan pemutusan USB yaitu ketika tidak ada piranti terhubung ke host atau komputer maka pull-down resistor yang ada akan menyebabkan D+ dan D- tertarik hingga di bawah ambang logika rendah host atau komputer ,hal ini menyebabkan munculnya keadaan SE0 (*singel-Ended 0*) di port downsteram.Kondisi tak-terhubungnya suatu piranti USB dari port akan dideteksi jika host atau komputer tidak mendrive *header* (di jalur data) selama lebih dari 2,5 μdetik.

Sedangkan jika sebuah piranti dihubungkan,maka host atau komputer akan mendeteksi bahwa salah satu jalur datanya tertarik hingan lebih besar dari ambang V_{IH} selama lebih dari 2,5µdetik.host kemudian bisa (option) mendeteksi kecepatan piranti yang baru terhubung dengan men-sempel keadaan bus segera sebelum men-drive SE0 untuk mereset piranti.jika diinginkan,host dapat membuat bus mengambang sesudah meresetnya dan menjalankan evaluasi bus sesudah 2,5µdetik tersebut.lebih jelasnya seperti tampak pada gambar-gambar berikut :

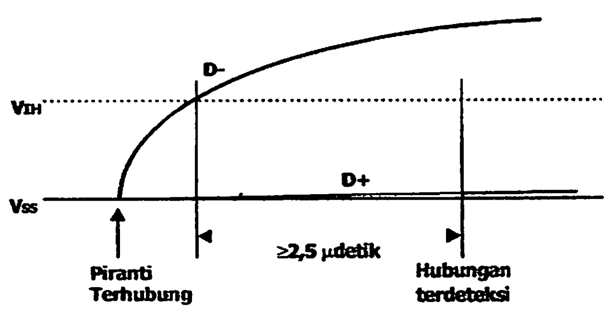


Gambar 2.6 Sebuah piranti dilepas dari port USB[2].

)



Gambar 2.7a Sebuah piranti USB kecepatan tinggi dihubungkan ke port USB pada host/komputer [2].



Gambar 2.7b Sebuah piranti USB kecepatan rendah dihubungkan ke port USB pada host/komputer [2].

Untuk mengirimkan paket data,USB menerapkan enkode data NRZI (*Non Return to Zero Invert*).Dalam NRZI ini.logika "1" berarti tidak ada perubahan level tegangan dan logika "0" ditunjukkan dengan adanya perubahan level tegangan.Paket data dikirimkan ke bus USB berurutan dari bit yang berbobot paling rendah (LSB,*Least Significant Bit*),diikuti LSB berikutnya dan terakhir adalah bit yang berbobot paling tinggi (MSB,*Mos Significant Bit*).

2.2 Mikrokontroler Renesas R8C/13 Tiny

Renesas technology adalah produsen semikonduktor tingkat internasional.Renesas terbangun dari dua produsen semikonduktor,renesas juga mengeluarkan berbagai jenis keluarga mikrokontroler (MK).

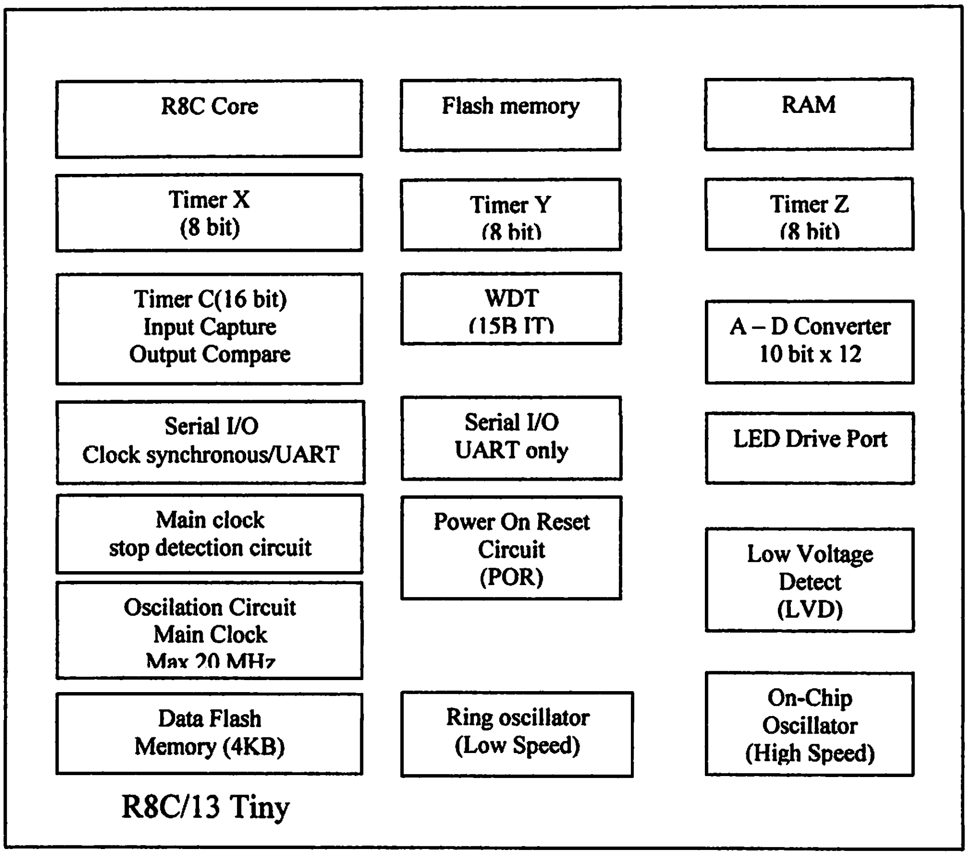
Renesas R8C adalah salah satu jenis keluarga MK M16C.CPU R8C sama dengan CPU CISC 16-bit M16C,hanya saja lebar jalur data R8C adalah 8-bit.Kerena menggunakan CPU yang sama maka R8C memiliki *instruction set* hamper sama dengan M16C.Perbadaanya hanya terletak pada 2 instruksi,yaitu

R8C tidak memiliki instruksi JMPS (*Jump Special Page*) dan JSRS (*Jump Subroutine Special Page*).R8C/13 adalah salah satu tipe MK dalam seri R8C.MK ini memiliki kemasan 32-*pin* LQFP.dalam perancangan pada skripsi ini menggunakan MK MK seri R5F21134FP,yaitu R8C/13 Tiny yang memiliki flash ROM 16KB(1000 E/W *cycles*) dan RAM sebesar 1KB.

2.2.1 Spesifikasi R8C/13Tiny

Berikut ini adalah spesifikasi R8C/13 Tiny dengan peta peripheral dan memori–memorinya :

- Mempunyai CPU core (16-bit) 1-20 MHz, 3.0 – 5.5 Volt dan 1 – 10 MHz 2.7 – 5.5 Volt.
- Rangkaian Clock,kecepatan *low/high On-Chip Oscillator*.Clock utama dengan Xin/Xout.
- Memory (ROM/SRAM) 16Kbyte / 1 Kbyte, 2 x 2 Kbyte data flash pada R8C/13 Tiny.
- Kemasan 32 pin LQFP (7 mm x 7 m)



Gambar 2.8 Blok Diagram R8C/13 Tiny dan Peta *Peripheral*-nya[3].

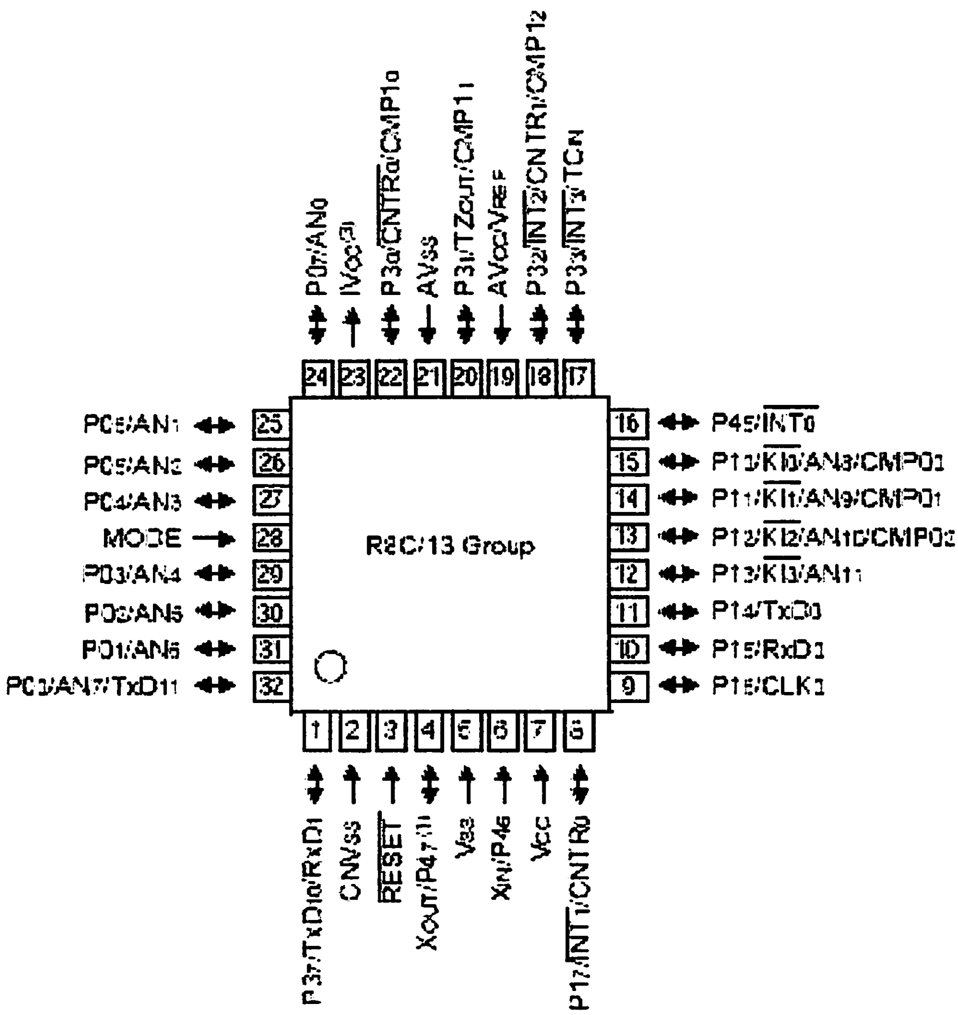
2.2.2 Kelebihan - kelebihan R8C/13 Tiny

Banyak kelebihan – kelebihan yang dimiliki R8C/13 Tiny diantaranya adalah :

- Kompatibel dengan M16C yaitu kompatibel dalam instruksi dan kode.
- *Peripheral* lebih terintegrasi jadi lebih hemat.
- *Electromagnetic Compatibility* (EMC) mempunyai EMI rendah,EMS tinggi.
- *Development Tool (Compiler dan Debugger)* didapat dengan murah dan difasilitasi *On-Chip Debugger*.

- Mempunyai *fitur fail – safe* yaitu pengamanan terhadap kegagalan system.
- Konsumsi daya rendah.
- 16-bit CISC CPU dengan kecepatan maksimal 20 MHz (1 : 1).
- 89 instruksi CISC lebih hemat ROM kira – kira 20 % RAM sampai 1 KB.
- Waktu konversi ADC 3 μS.

2.2.3 Konfigurasi Pin R8C/13 Tiny



Gambar 2.9 Konfigurai Pin R8C/13 Tiny[3].

Gambar diatas adalah konfigurasi pin – pin dari R8C/13 Tiny untuk lebih jelasnya dapat diamati pada table deskripsi pin – pin berikut ini :

Tabel 2.1 Konfigurasi pin – pin dari R8C/13 Tiny[3]

Nama sinyal	**Nama Pin**	**Type I/O**	**Fungsi**
Masukan Catu Daya	Vcc, Vss	I	Tegangan 2.7 V – 5.5 V pada pin Vcc Tegangan 0 V pada Vss pin
I Vcc	Ivcc	O	Pin ini untuk men_stabilkan catu daya *Internal*,pin ini dihubungkan pada Vss melalui kapasitor 100nF.jangan dihubungkan pada Vcc.
Input Catu Daya Analog	Avcc,Avss	I	Ini adalah catu daya pada ADC.A Vcc dihubungkan pada Vcc,A Vss dihubungkan ke Vss.Dianjurkan untuk menghubungkan kapasitor diantara pin A Vcc dan AVss.
Input Reset	RESET	I	"L" untuk masukan ini mereset MCU.
CNVss	CNVss	I	Pin ini dihubungkan pada Vss melalui sebuah resistor.
MODE	MODE	I	Pin ini dihubungkan pada Vcc melalui sebuah resistor.
Input Clock Utama	Xin	I	Pin ini dsediakan untuk membangkitkan rangkaian I/O Clock utama.Dihubungkan

Output clock Utama	Xout	O	dengan sebuah keramik resonator atau kristal diantara pin Xin dan Xout.Jika digunakan clock internal maka pin Xin dan Xout dalam keadaan terbuka.
Inpu Kunci Interupsi	K10-K13	I	Pin ini sebagai masukan kunci interupsi.
TIMER X	CNTR 0	I/O	Pin I/O ini adalah untuk Timer X.
	CNTR 0	O	Pin Output untuk Timer X.
Timer Y	CNTR 1	I/O	Pin I/O untuk Timer Y.
Time Z	TZout	O	Pin Output untuk Timer Z.
Timer C	TC in	I	Pin input untuk Timer C.
	CMP00 CMP03 CMP10 CMP13	O	Pin Output untuk Timer C.
Serial Inteface	CLK 0	I/O	Pin I/O untuk memindahkan Clock.
Serial Inteface	RXD0 RXD1	I	Pin input untuk data Serial.
	TXD0 TXD10 TXD11	O	Pin output untuk data Serial.
Input	Vref	I	Tegangan referensi input untuk ADC.

Tegangan Referensi			Vref pin dihubungkan ke Vcc.
ADC,penguba h dari analog ke digital	AN0-AN11	I	Pin analog input pada ADC.
Port I/O	P00-P07, P10-P17, P30-P33, P37-P45	I/O	Merupakan port I/O CMOS 8-bit.Setiap port mempunyai pilihan register pengarah sebagai input output.tiap port dapat dialamati per bit.Dapat di-set menggunakan pull up resistor dengan program.P10-P17 mempunyai driver transistor.
Port input	P46,P47	I	Pin ini hanya bias digunakan sebagai input.

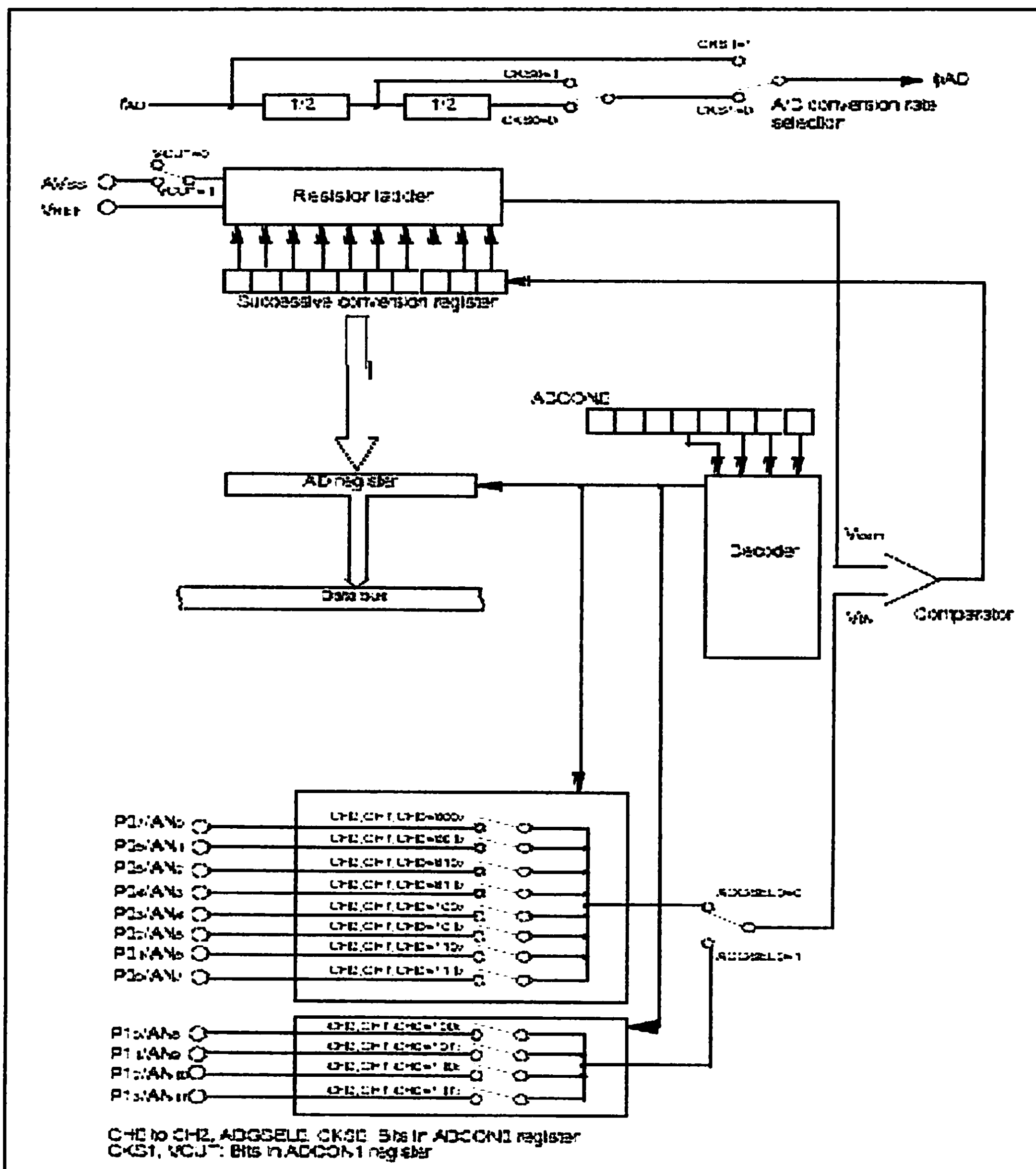
2.2.4 *Periperal R8C/13 Tiny*

Mikrokontroler R8C/13 Tiny mempunyai beberapa *peripheral – peripheral* yang banyak digunkan pada beberapa aplikasi – aplikasi penting,diantaranya adalah sebagai berikut :

❖ **Analog To Digital Converter (ADC)**

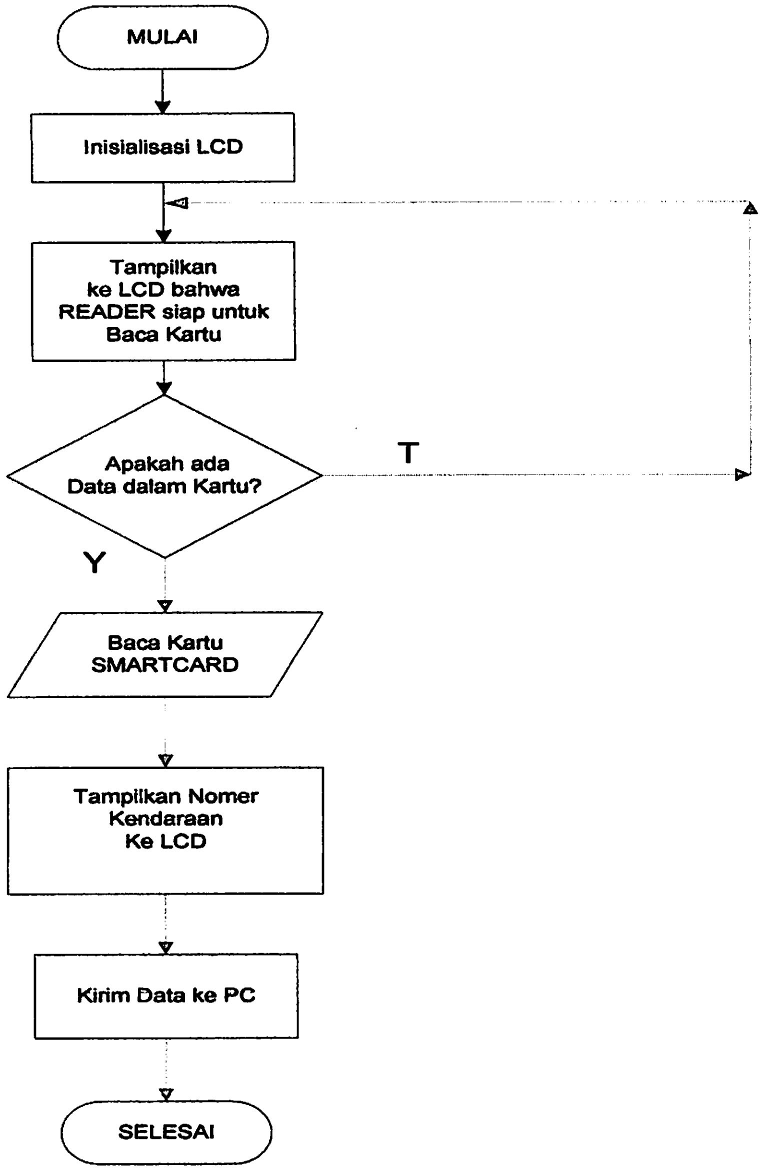
Dengan 12 SAR ADC S/H yang mempunyai resolusi 8-bit atau 10-bit.

Mode Operasinya menggunakan *One-Shot* dan *Repeat* dengan waktu konversi 2.8 µS (pada clock 10 MHz).berikut gambar diagram blok ADC *built in* pada mikrokontroler :



Gambar 2.10 Diagram blok ADC[3].

3.9 FlowChart



Gambar 3.20

Diagram Alir Program Mikrokontroler

Format perintah

Instruction Code	Data length	Data					
	LEN	ADDR		BYTE 1	...	...	BYTE N
91_H							

Gambar 2.19 Format Write_data[6].

LEN : Banyaknya byte data pada penulisan di kartu mulai dari awal data hinga akhir data.

ADDR : Alamat byte di dalam kartu pada saat memulai penulisan.

BYTE x : Nilai byte pada kartu mulai dari alamat ADDR dimana dimana BYTE 1 adalah penulisan alamat ADDR dan BYTE N adalah penulisan untuk alamat ADDR+N-1.

5. Present_Code

Memberikan kode rahasia dari dari kartu serta memungkinkan penulisan operasi dengan kartu SLE4442.

Format present_Code tersebut adalah :

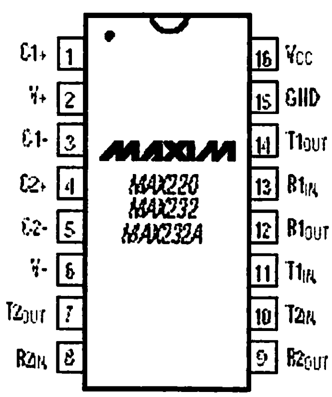
Instruction Code	Data length	Data		
		CODE		
92_H	03_H			

Gambar 2.20 Format Present_Code[6].

CODE : Tiga byte rahasia code (PIN)

2.5 Komunikasi Serial

Berfungsi untuk hubungan komunikasi antara reader smartcard dengan mikrokontroler R8C/13 Tiny, digunakan RS-232, dimana pada RS232 ini terdapat fungsi-fungsi untuk Tx (pengiriman data), Rx (penerimaan data),CTS(Clear To Send) dan DTR. Untuk melakukan *transfer* data dari Reader Smartcard ke mikrokontroler digunakan IC MAX232, yang merupakan rangkaian terpadu untuk antarmuka komunikasi serial.



Gambar 2.21 RS 232 (MAXIM) [7].

RS MAX232 tersusun dari 2 bagian yaitu *RS232 Line Driver* yang berfungsi mengubah level tegangan TTL ke level tegangan RS232 dan *RS232 Line Receiver* yang berfungsi mengubah level tegangan RS232 ke level tagangan TTL.

Alat ini merupakan standart yang dipakai untuk mengirimkan aliran bit seri antar *interface*. Komunikasi serial dapat dibagi menjadi dua sifat dasar pola komunikasi. Yang pertama adalah komunikasi asinkron, dimana pola-pola bit tertentu dipakai untuk memisahkan bit-bit karakter. yang kedua adalah komunikasi seri *sinkron,* yang memungkinkan karakter dikirim secara berurutan, namun membutuhkan karakter *sinkronisasi* khusus pada awal setiap karakter dan karakter semua khusus untuk dikirimkan ketika tidak ada informasi yang sedang dikirim.

2.5.1 Protokol Komunikasi pada RS 232

Beberapa protokol dalam interface RS 232 adalah:

- Start Bit

  Merupakan sebuah bit dengan logic "0" dimana bit ini yang menandakan bahwa akan ada karakter atau data yang mengikutinya. Bit ini langsung diberikan oleh sinyal device tanpa harus mensetnya terlebih dahulu.

- Data Bit

  Merupakan bit yang mewakili dari karakter yang diikutinya *data bit* ini dapat diset sepanjang antara 5 sampai 8 bit.

- Pariti Bit

  Merupakan bit yang digunakan sebagai *error checking* pada *receiver* , apabila terjadi kesalahan maka *receiver* akan menset *error flag* (parity error) pada special register. *Parity bit* ini menghitung jumlah data yang berlogic '1' pada data bit. Perhitungan jumlah data bit tersebut tergantung dari jenis *parity* yang diset. Untuk parity *EVEN* maka jumlah data bit yang berlogic '1' ditambah dengan *parity bit* akan menghasilkan jumlah yang ganjil. Sedangkan untuk *parity MARK* merupakan *parity bit* selalu berlogic '1' begitu pula pada space, parity bit selalu berlogic ' 0' dan *parity NONE* disini *parity bit* yang diabaikan.

- Stop Bit

  Merupakan bit yang menandakan akhir dari suatu paket data (biasanya 1 byte data). Seperti pada start bit, bit ini langsung diberikan dari serial device. *Stop bit* ini dapat diset panjangnya menjadi satu bit, satu setengah dan dua bit.

- Baut Rate

  Sebenarnya *baut rate* berarti pergantian kondisi tiap detik *(State Change of the Line persecond),* tetapi karena hanya ada 2 kondisi pada serial *(logic* 0 *dan 1)* maka dapat juga digunakan untuk menunjukkan kecepatan dari transmisi *(bits per second).*

BAB III

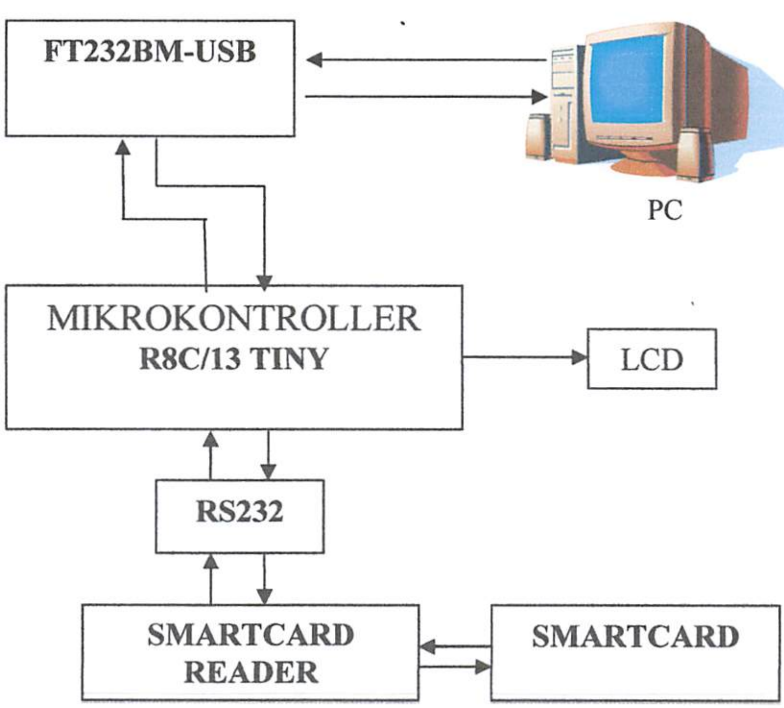
PERANCANGAN DAN PEMBUATAN ALAT

3.1. Pendahuluan

Dalam bab ini akan membahas tentang perencanaan dan pembuatan keseluruhan sistem perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan dalam sistem pembuatan alat daftar kir kendaraan bermotor berbasis SMARTCARD yang diinformasikan via USB (*Universal Serial Bus*).

3.1.1. Blok Diagram Keseluruhan Sistem

Perancangan dan pembuatan alat ditunjukkan dengan gambar blok diagram dibawah ini :



Gambar 3.1. Diagram Blok Keseluruhan Sistem.

Keterangan fungsi dari masing-masing blok diagram diatas sebagai berikut :

- **PC.**

  Pada bagian ini berfungsi sebagai tampilan dengan menampilkan.

- **FT232BM-USB.**

  Merupakan bagian yang berfungsi sebagai modul penghubung COM TO USB.

- **MIKROKONTROLLER R8C13/TINY.**

  Merupakan bagian yang berfungsi sebagai menerima data yang berasal dari smartcard reader.

- **RS 232**

  Berfungsi sebagai interface antara mikrokontroler dengan Smartcard Reader.

- **SMARTCARD READER.**

  Berfungsi sebagai penerima dan menulis data pada kartu smartcard.

- **SMARTCARD.**

  Berfungsi sebagai kartu yang memberikan inputan data kendaraan.

- **LCD.**

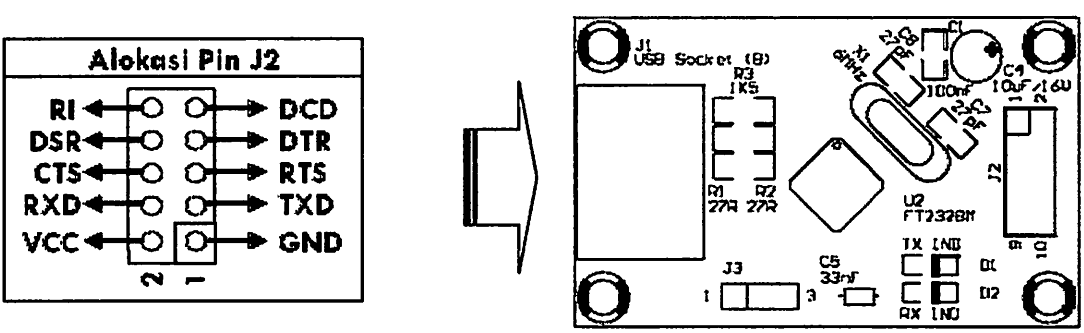
  Berfungsi untuk menapilkan informasi.

3.2 Modul FT232BM.

Pada skripsi ini Modul Modul FT232BM berfungsi sebagai Modul USB (Universal Serial Bus) yang menjembatani antara mikrokontroler (R8C13/Tiny) dengan komputer. Pada modul USB ini ada beberapa pin yang mesti di pakai dengan seksama sesuai dengan kegunaannya yang terdiri pin J2 sebagai port serial sebagai I/O.

3.2.1 Penggunaan Pin-Pin Modul FT232BM.

Gambar 3.2 menunjukkan pin-pin Modul FT232BM, namun dalam perancangan sistem ini beberapa pin saja yang dipakai.



Gambar 3.2. Pemakaian PIN pada port Serial dari Modul FT232BM.

Pin-pin yang digunakan dalam perancangan ini:

- Pin 1, GND (Ground)

  Pin 1 pada J2 adalah pin yang digunakan sebagai ground pada modul USB ini,sedangkan pada IC FT232BM ground terletak pada pin 9,17,dan 29.

- Pin 2, VCC

  Modul USB ini di operasikan dengan tegangan supply 4,4V – 5,25 V DC,pada IC FT232BM Vcc berada pada pin 3,13,26,30.

- Pin 3, RxD

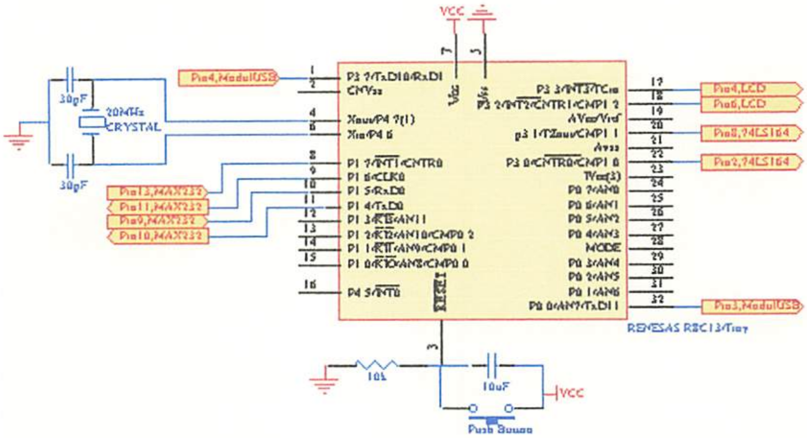
  Pin 3 pada J2 merupakan pin yang berada pada pin 24 dari IC FT232BM. Dalam perancangan pin digunakan sebagai port input serial.

- Pin 4, TxD

  Pin 4 pada J2 merupakan pin 25 dari IC FT232BM dan dalam perencanaannya, pin ini digunakan sebagai output serial.

3.3 Perencanaan MCU Renesas R8C/13 Tiny.

MCU bekerja jika ada kartu yang dimasukkan ke reader smartcard . Alasan penggunaan MCU Renesas R8C/13 Tiny adalah rangkaiannya yang praktis (bentuk fisik IC-nya yang sangat kecil) karena tersusun dalam satu modul yang sangat mudah untuk kita gunakan, sederhana dan tidak memakan tempat. Jumlah Tx (Transmitter) dan Rx (Receiver) ada 2, yaitu untuk Tx pada kaki ke-11 (port 1.4) dan kaki ke-32 (port 0.0). Sedangkan untuk Rx pada kaki ke-1 (port 3.7) dan kaki ke-10 (port 1.5). Pada MCU Renesas R8C/13 Tiny rangkaian ADC (Analog to Digital Converter) sudah ada sehingga kita tidak perlu membuat rangkaian ADC baru. Mikrokontroler RENESAS dibangun menggunakan proses gerbang *silicon* CMOS dengan kemampuan tinggi menggunakan CPU seri R8C/13Tiny dan dikemas dalam modul plastik dengan jumlah pin sebanyak 32. Mikrokontroler ini beroperasi menggunakan perintah canggih khususnya efisiensi perintah dengan level tinggi. Mikrokontroler ini mempunyai 1 Mbytes kapasitas alamat, yang bisa digunakan untuk mengeksekusi perintah dengan kecepatan tinggi. Data flash ROM sebesar 2 KB x 2 blocks.



Gambar 3.3 Port pada MCU Renesas R8C/13 Tiny yang dipakai pada sistem.

Fungsi port pada MCU yang digunakan pada alat adalah sebagai berikut :

- Port 0 memiliki fungsi sebagai berikut :
  - Port 0.0 digunakan sebagai output data ke Modul FT232BM (Tx).
- Port 1 memiliki fungsi sebagai berikut :
  - Port 1.4 digunakan sebagai output data ke RS232(Tx).
  - Port 1.5 digunakan sebagai input data dari RS232(Rx).
  - Port 1.6 digunakan sebagai DTR ke RS232..
  - Port 1.7 digunakan sebagai CTS dari RS232.
- Port 3 memiliki fungsi sebagai berikut :
  - Port 3.0 digunakan sebagai penyemat RS pada LCD.
  - Port 3.1 digunakan sebagai penyemat Enable pada LCD.
  - Port 3.2 digunakan sebagai penyemat clock pada LCD.
  - Port 3.3 digunakan sebagai input data dari LCD.
  - Port 3.7 digunakan sebagai input data dari Modul FT232BM.

- Port 4 memilki fungsi sebagai berikut :
  - Port 4.7 dengan port 4.6 disambungkan dengan rangkaian osilator.

3.3.1 Rangkaian Reset

Untuk me-*reset* mikrokontroler Renesas R8C/13 Tiny, maka pin RST diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal *reset* kapasitor dihubungkan dengan V_{CC} dan sebuah resistor yang dihubungkan ke *ground.*

Karena kristal yang digunakan mempunyai frekuensi sebesar 20 MHz, maka satu periode dapat dihitung dari persamaan :

$$T = \frac{1}{fxtal} = \frac{1}{20MHz} s = 5 \times 10^{-8} \text{ s}$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk me*reset* mikrokontroler dapat dihitung menggunakan persamaan :

$t_{reset(min)}$ = T x periode yang dibutuhkan

Jadi mikrokontroler membutuhkan waktu minimal 1.2 µs untuk me*reset*. Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C. Dari Persamaan dengan menentukan nilai R = 22 kΩ, dan C = 10 µF, maka:

$t = 0{,}357 \text{ R.C}$

```
d[0]=kon(tangkap1[1]/16);
d[1]=kon(tangkap1[1]%16);
d[2]=kon(tangkap1[2]/16);
d[3]=kon(tangkap1[2]%16);
d[8]=kon(tangkap1[7]/16);
d[9]=kon(tangkap1[7]%16);
d[10]=kon(tangkap1[8]/16);
d[11]=kon(tangkap1[8]%16);
for(i=4;i<8;i++) d[i]=tangkap1[i-1];
for(i=12;i<60;i++) d[i]=tangkap1[i-3];
}
void dta(unsigned char aa)
{
unsigned char bb;
bb=aa/16+48;kirim0(bb);
bb=aa%16+48;kirim0(bb);
}
unsigned char atd(unsigned char aa,unsigned char bb)
{
unsigned char cc;
cc=(aa-48)*16;
cc=cc+(aa-48);
}
void tunggu()
{
while(tt==0);
while(tt==1);
}
void oncard()
{       kirim0(2);
        kirimt0("01800081");
        kirim0(3);
        tunggu();
        tsb=0;
for(i=0;i<6;i++)
{       if(tangkap[i+1]!=fbreset[i]) tsb=1; }
                }
void offcard()
{       kirim0(2);
        kirimt0("01810080");
        kirim0(3);
        }
void milihcard()
{       kirim0(2);
        kirimt0("0102010604");
```

```
        kirim0(3);
        }
void presetcode()
{       kirim0(2);
        kirimt0("019203FFFFFF6F");
        kirim0(3);}
void awalproses()
{
        dtr=0;delay(100);dtr=1;delay(100);dtr=0;
        delay(1000);
        milihcard();delay(1000);
        oncard();delay(1000);
        presetcode();delay(1000);
}
void tangkapcard()
{       kirim0(2);
        kirimt0("01810080");
        kirim0(3);}
void tuliscard()
{
unsigned char aa,bb;
awalproses();
kirim0(2);
kirimt0("019120001B");

aa=0;bb=0xAB;
for(i=0;i<60;i++)
        {
        if(i%2==0) aa=nok(d[i])*16;
        else {aa=nok(d[i])+aa;bb=bb^aa;}
        }

for(i=0;i<60;i++) kirim0(d[i]);

kirim0(kon(bb/16));
kirim0(kon(bb%16));
kirim0(3);
}
void bacacard()
{
awalproses();
kirim0(2);
kirimt0("019003001B1E97");
kirim0(3);
}
```

```
void cleardata()
{
for(i=0;i<100;i++)
d[i]='0';
}
#pragma INTERRUPT rx
void rx(void)
{
while(ri_u1c1 == 0);
        ri_u1c1 =0;
        if(u1rb=='a') {tcom=1;i2=0;}
        else if(u1rb=='z') {tcom=0;}
        if(tcom==1) {tangkap1[i2]=u1rb;i2++;}
        u1c1 |= 1;
        }
#pragma INTERRUPT rx0
void rx0(void)
{
while(ri_u0c1 == 0);
        ri_u0c1 =0;

        if(u0rb==2) {tt=1;i=0;}
        else if(u0rb==3) tt=0;
        if(tt==1) {tangkap[i]=u0rb;i++;}

        u0c1 |= 1;
        }
#pragma INTERRUPT tmx
void tmx(void)
{
        txund=0;
        itimer++;
        prex = 99;
        tx = 199;
}
void main(void)
{
/* Inisialisasi Variable*/
        /* Inisialisasi Awal MK */
        asm("FCLR I");          // Interrupt disable
        prcr = 1;               // Protect off
        cm13 = 1;               // X-in X-out = Clock External
        cm15 = 1;               // XCIN-XCOUT drivecapacity select bit : HIGH
        cm05 = 0;               // X-in on
        cm16 = 0;               // Main clock = No division mode
```

```
cm17 = 0;
cm06 = 0;               // CM16 and CM17 enable
asm("nop");
asm("nop");
asm("nop");
asm("nop");
ocd2 = 0;               // Main clock change (x-tal)
prcr = 0;               // Protect on
s1ric=2;
s0ric=1;

/* p1 sebagai keluaran dan p0 sebagai masukan */
pd1=0x80;
pd1_4=1;
pd0=0;
prc2=1;pd0_5=1;
prc2=1;pd0_4=1;
pd3=0xff;
pd3_7=0;
pd1_6=1;pd1_7=0;
asm("FSET I");
/* Proses memasukkan dan mengeluarkan data */
initlcd();
initSerial();
tcom=0;
cetak(1,1,"EDDY NUR IMAM S.");
cetak(2,1,"     ITN      ");
for(i=0;i<60;i++) {d[i]='0';tangkap1[i]='0';}
dtr=0;delay(100);dtr=1;delay(100);dtr=0;
delay(100);
milihcard();delay(100);
oncard();delay(100);
presetcode();delay(100);
oncard();
delay(2500);busek();cetak(1,1,"siap");
while (1)
{
tsb=0;
while(tt==0)
{
if(tcom==1){cetak(1,1,"hub computer    ");while(tcom==1);
                        cetak(1,1,"proses transfer ");pindahdata();
                        tuliscard();
                        offcard();
                        busek();cetak(1,1,"siap");
```

```
                        }
        }
        while(tt==1);
        tsb=0;
        for(i=0;i<10;i++)
        {if(tangkap[i+1]!=fbmasuk[i]) tsb=1;}
        if(tsb==0) {busek();cetak(1,1,"deteksi masuk  ");
                        bacacard();tunggu();busek();
                        pos(1,1);

        a=nok(tangkap[9])*16+nok(tangkap[10]);kirim_serial(a);dataout(a);

        a=nok(tangkap[11])*16+nok(tangkap[12]);kirim_serial(a);dataout(a);
                        dataout(' ');
        kirim_serial(tangkap[13]);kirim_serial(tangkap[14]);
        kirim_serial(tangkap[15]);kirim_serial(tangkap[16]);
                        dataout(tangkap[13]);dataout(tangkap[14]);
                        dataout(tangkap[15]);dataout(tangkap[16]);
                        dataout(' ');
        a=nok(tangkap[17])*16+nok(tangkap[18]);kirim_serial(a);dataout(a);
        a=nok(tangkap[19])*16+nok(tangkap[20]);kirim_serial(a);dataout(a);
                        for(i=21;i<57;i++)kirim_serial(tangkap[i]);
                        }
                        offcard();
        tsb=0;
        for(i=0;i<10;i++)
        {if(tangkap[i+1]!=fbkeluar[i]) tsb=1;}
        if(tsb==0) {busek();cetak(1,1,"deteksi keluar  ");delay(2000);busek();
        cetak(1,1,"siap");}


        }
}
```

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics,
Controls, Forms,
  Dialogs, DB, ADODB, Menus, ExtCtrls, StdCtrls, CPort,
DBCtrls, Grids,
  DBGrids, jpeg, DBTables, ExtDlgs, Mask;

type
  TForm1 = class(TForm)
    Notebook1: TNotebook;
    MainMenu1: TMainMenu;
    File1: TMenuItem;
    close1: TMenuItem;
    ADOConnection1: TADOConnection;
    ADOQuery1: TADOQuery;
    ADOConnection2: TADOConnection;
    ADOQuery2: TADOQuery;
    ComPort1: TComPort;
    EdtdataMK: TEdit;
    btncek: TButton;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    Label6: TLabel;
    Label7: TLabel;
    Label8: TLabel;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Foto: TLabel;
    data1: TMenuItem;
    Image1: TImage;
    BtnOK: TButton;
    Label12: TLabel;
    Label13: TLabel;
    Label14: TLabel;
    Label15: TLabel;
    Label16: TLabel;
    Label17: TLabel;
    Label19: TLabel;
    proses1: TMenuItem;
```

```
    Button1: TButton;
    Table1: TTable;
    DBGrid1: TDBGrid;
    DBNavigator1: TDBNavigator;
    setting1: TMenuItem;
    Button2: TButton;
    Button3: TButton;
    DBEdit1: TDBEdit;
    DBEdit2: TDBEdit;
    DBEdit3: TDBEdit;
    DBEdit4: TDBEdit;
    DBEdit5: TDBEdit;
    DBEdit6: TDBEdit;
    DBEdit7: TDBEdit;
    DBEdit8: TDBEdit;
    DBEdit9: TDBEdit;
    DBEdit10: TDBEdit;
    OpenPictureDialog1: TOpenPictureDialog;
    SavePictureDialog1: TSavePictureDialog;
    DataSource1: TDataSource;
    Button4: TButton;
    DBLookupComboBox1: TDBLookupComboBox;
    Melihatdatakeseluruhan1: TMenuItem;
    ambahUbahdata1: TMenuItem;
    Button5: TButton;
    Image2: TImage;
    Label1: TLabel;
    Label2: TLabel;
    Label18: TLabel;
    procedure kirimdata(data:string);
    procedure close1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure ComPort1RxChar(Sender: TObject; Count: Integer);
    procedure proses1Click(Sender: TObject);
    procedure BtnOKClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure setting1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
    procedure Button3Click(Sender: TObject);
    procedure Button4Click(Sender: TObject);
    procedure Melihatdatakeseluruhan1Click(Sender: TObject);
    procedure ambahUbahdata1Click(Sender: TObject);
    procedure Button5Click(Sender: TObject);
  private
    { Private declarations }
```

```
 public
   { Public declarations }
 end;

var
 Form1: TForm1;

implementation

{$R *.dfm}

procedure TForm1.kirimdata(data:string);
var data_kirim:string;
   i:integer;
begin
i:=0;
repeat
 i:=i+1;
 data_kirim:=copy(data,i,1);
 ComPort1.WriteStr(data_kirim);
 Sleep(200);
until i=length(data);
end;

procedure TForm1.close1Click(Sender: TObject);
begin
close;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
Notebook1.ActivePage:='utama';
end;

procedure   TForm1.ComPort1RxChar(Sender:   TObject;   Count:
Integer);
var data:string;
begin
ComPort1.ReadStr(data,Count);
EdtdataMK.Text:=EdtdataMK.Text+data;
if Length(EdtdataMK.Text)=44 then
  begin
   if Table1.FindKey([copy(EdtdataMK.Text,1,8)]) then
     begin
     Notebook1.ActivePage:='tampilan';
```

```
    Button5.Click;
    end;
   EdtdataMK.Text:='';
  end
else if Length(EdtdataMK.Text)>44 then EdtdataMK.Text:='';
end;

procedure TForm1.proses1Click(Sender: TObject);
begin
Notebook1.ActivePage:='proses';
end;

procedure TForm1.BtnOKClick(Sender: TObject);
var
no,jenis,berlaku,berat,jbb,jbi,mst,orang,barang,kelas,data_dikirim:s
tring;
begin
if DBEdit1.Text='' then
begin
  ShowMessage('nomer kendaraan kosong');
end
else
begin

Image1.Picture.SaveToFile(ExtractFilePath(Application.ExeName
)+'\gambar\'+Table1.fieldbyname('no').AsString+'.jpg');
  Table1.Edit;
  Table1.FieldByName('foto').AsString:='ada';
  Table1.Post;


data_dikirim:='a'+DBEdit1.Text+DBEdit3.Text+DBEdit4.Text+D
BEdit5.Text

+DBEdit6.Text+DBEdit7.Text+DBEdit8.Text+DBEdit9.Text+DB
Edit10.Text+'z';
 kirimdata(data_dikirim);
end;
end;

procedure TForm1.Button1Click(Sender: TObject);
begin
kirimdata('a'+EdtdataMK.Text+'z');
end;
```

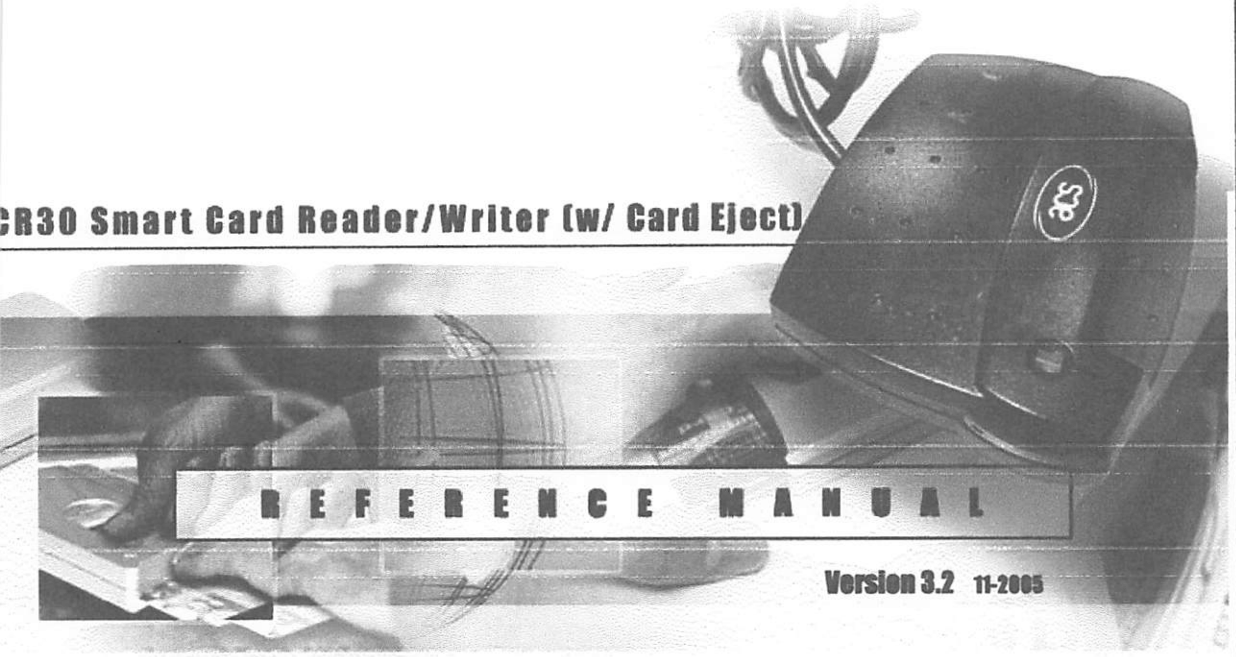
```
procedure TForm1.setting1Click(Sender: TObject);
begin
  ComPort1.ShowSetupDialog ;
  ComPort1.Connected := true ;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
if OpenPictureDialog1.Execute then begin
  Image1.Visible:=true;
  Image1.Picture.LoadFromFile (OpenPictureDialog1.FileName);
end;

end;

procedure TForm1.Button3Click(Sender: TObject);
begin
Table1.Cancel;
end;

procedure TForm1.Button4Click(Sender: TObject);
begin
Table1.Append;
Image1.Visible:=false;
end;

procedure            TForm1.Melihatdatakeseluruhan1Click(Sender:
TObject);
begin
Notebook1.ActivePage:='tblkendaraan';
end;

procedure TForm1.ambahUbahdata1Click(Sender: TObject);
begin
Notebook1.ActivePage:='tampilan';
end;

procedure TForm1.Button5Click(Sender: TObject);
var data_dikirim:string;
begin
IF Table1.FieldByName('foto').IsNull then ShowMessage('gak da
gambar')
else begin
 Image1.Visible:=true;
```

```
Image1.Picture.LoadFromFile(ExtractFilePath(Application.ExeNa
me)+'\gambar\'+Table1.fieldbyname('no').AsString+'.jpg');
  end;
end;

end.
```

Modul Smartcard

Advanced Card Systems Ltd.

CR30 Smart Card Reader/Writer (w/ Card Eject)



REFERENCE MANUAL

Version 3.2 11-2005

Unit 1008, 10th Floor, Hongkong International Trade and Exhibition Centre
1 Trademart Drive, Kowloon Bay, Hong Kong
Tel: +852 2796 7873 Fax: +852 2796 1286 Email: info@acs.com.hk Website: www.acs.com.hk

ntents

te:

– SAM Reader does not support for memory cards

– SAM Reader only

Introduction

ACS Smart Card Reader/Writer ACR30 is an interface for the communication between a computer (for mple, a PC) and a smart card. Different types of smart cards have different commands and different nmunication protocols. This prevents in most cases the direct communication between a smart card and omputer. The ACR30 Reader/Writer establishes a uniform interface from the computer to the smart card a wide variety of cards. By taking care of the card specific particulars, it releases the computer software grammer of getting involved with the technical details of the smart card operation, which are in many es not relevant for the implementation of a smart card system.

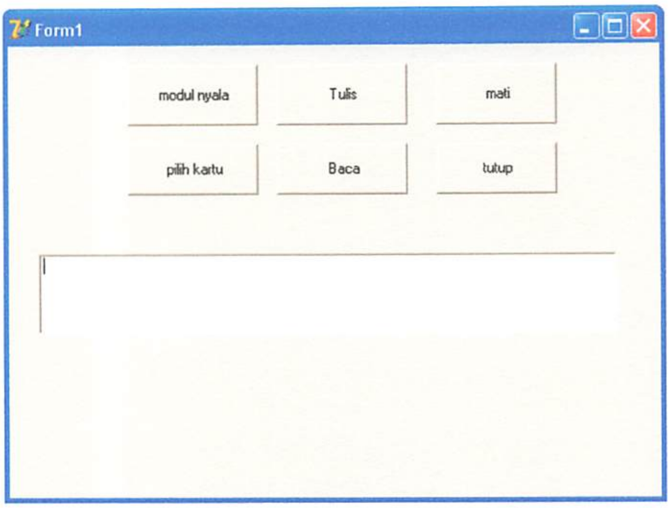
ACR30 Smart Card Reader/Writer is connected to the computer through a serial asynchronous rface (RS-232) or USB interface. The reader accepts commands from the computer, carries out the cified function at the smart card and returns the requested data or status information.

TE - Although the ACR30 is a true *card reader/writer* as it can read and write smart cards, the terms *card reader* or *reader* will be d indifferently to refer to the ACR30, for the sake of readability and because these designations are commonly in use for this kind of ces.

Features

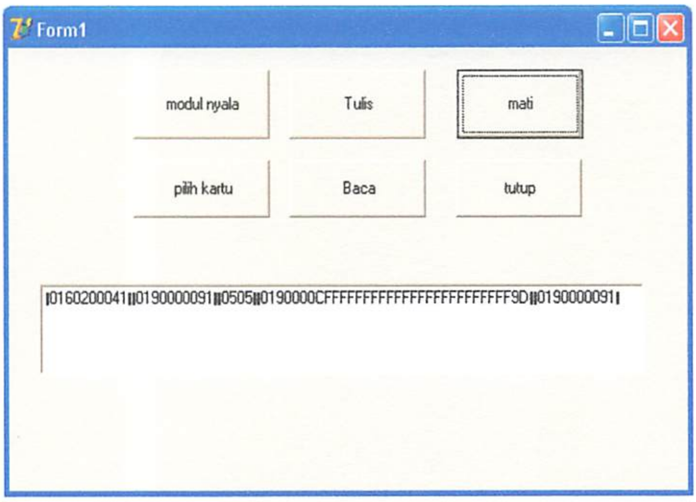
ISO7816-1/2/3 compatible smart card interface
Supports CPU-based cards with T=0 and/or T=1 protocol
(*) Supports commonly used memory cards (I2C, SLE4406, SLE4418/28, SLE4432/42)
Support PPS (Protocol and Parameters Selection) with 9600 – 96000 bps in reading and writing smart cards
RS-232 interface or USB interface to PC with simple command structure
Supports memory cards SLE4436 and SLE5536 (firmware 2.10 onwards)
Security application modules (SAM) inside the reader supporting CPU-based cards with T=0 and/or T=1 protocol (SAM Reader only)

e (*) – SAM Reader does not support for memory cards



Gambar 4.17. Tampilan Program Dijalankan.

e. Setelah itu ditekan tombol modul nyala, pilih kartu, tulis, baca dan mati sehingga pada program tersebut akan muncul :



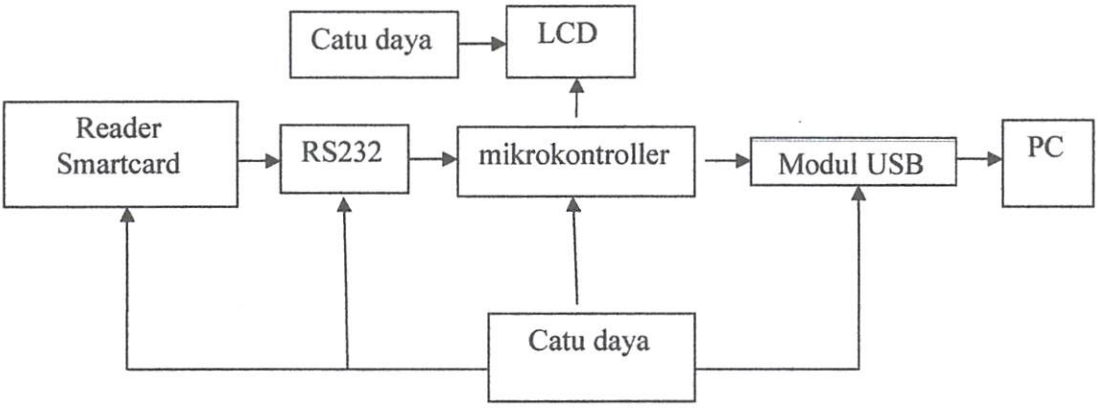
Gambar 4.18. Tampilan pada saat tombol ditekan.

f. Setelah program dijalankan berikut adalah tampilan pada reader smartcard:

Gambar 4.19 Pengujian Reader Smartcard.

4.6 Pengujian Keseluruhan Sistem

Tujuan dari pengujian Sistem ini adalah untuk mengetahui apakah alat yang dibuat dan dirancang dapat bekerja sesuai yang diinginkan. Adapun prosedur pengujian sistem ini yaitu



Gambar 4.20. Blok Pengujian Sistem Alat Daftar Kir Kendaraa

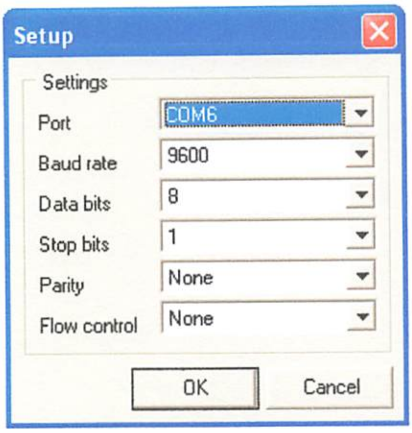
- Menghubungkan keseluruhan rangkaian sesuai dengan diagram blok
- Menjalankan program Database Delphi
- Melakukan proses identifikasi COM
- Memasukkan Kartu smartcard pada Reader
- Menyamakan data pada kartu dengan database
- Mengisi data bila tidak ada data pada kartu dan database

4.6.1 Hasil pengujian dari sistem :

1. Tampilan Informasi Pada LCD
   a. Menampilkan Tulisan " Siap ".
   b. Menampilkan Tulisan " Deteksi Masuk ".
   c. Menampilkan Tulisan " Nomor Kendaraan ".
   d. Menampilkan Tulisan " Deteksi Keluar ".
   e. Menampilkan Tulisan " Hub Computer ".
   f. Menampilkan Tulisan " Proses Transfer ".
2. Tampilan Informasi Pada Program Delphi
   a. Menampilkan Form Utama
   b. Menampilkan Form Tambah Data atau Ubah Data
   c. Menampilkan Form Melihat Data Keseluruhan

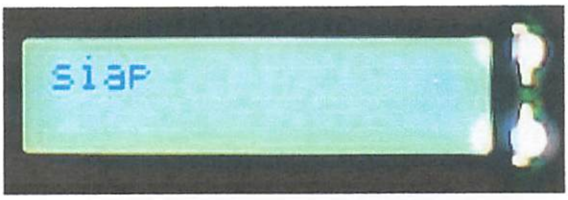
4.6.2 Proses Identifikasi dan Sistem Dari Alat Daftar Kir Kendaraan.

a. Setelah semua perangkat dihubungkan dan program Delphi dijalankan terlebih dahulu yang dilakukan adalah memilih COM yang dideteksi oleh PC atau Komputer seperti gambar 4.20 seting ini dilakukan pada saat program delphi dijanankan dan pada menu setting .



Gambar 4.21 Tampilan Setting Port.

b. Pada saat sistem diberi tegangan dan terkoneksi dengan PC atau Laptop, LCD akan menampilkan tulisan seperti gambar 4.22:

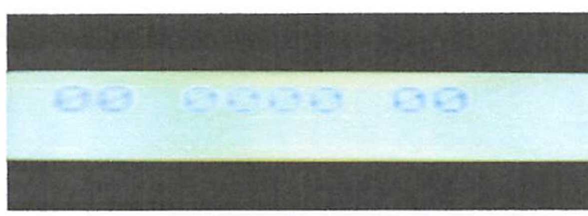


Gambar 4.22 Tampilan LCD Siap.

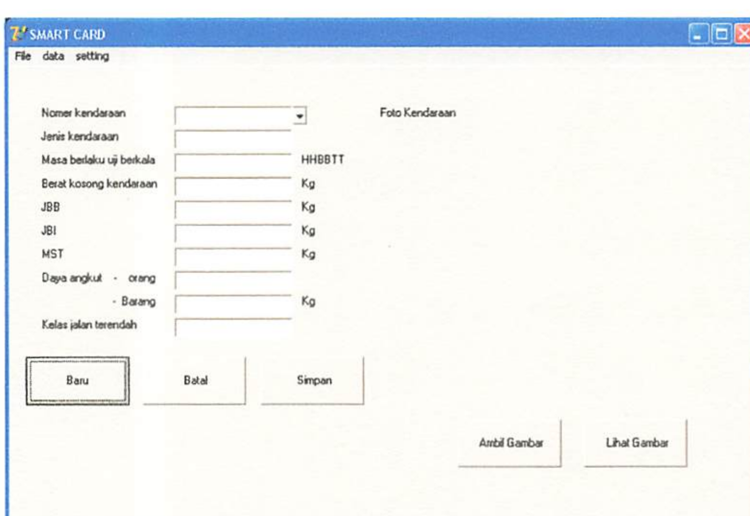
c. Pada saat kartu smartcard dimasukkan pada Reader maka LCD akan menampilkan tulisan seperti gambar 4.23. kemudian jika terdapat data pada kartu maka LCD akan menampilkan tulisan seperti gambar 4.24 jika tidak terdapat data seperti gambar 4.25,serta Delphi akan menampilkan data dari kartu tersebut apabila terdapat data pada database maka akan ditampilkan,jika tidak terdapat data maka mengisinya dengan menekan tombol baru seperti pada gambar 4.26 setelah data selesai diisi maka selanjutnya menekan tombol simpan :

Gambar 4.23 Tampilan LCD Deteksi Masuk.

Gambar 4.24 Tampilan LCD Data Nomor Kendaraan.



Gambar 4.25 Tampilan LCD Tidak Ada Data.

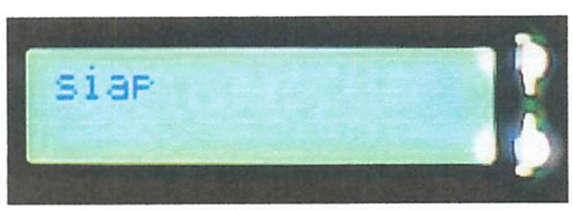


Gambar 4.26 Tampilan Layar Monitor Data Kendaraan.

d. Pada saat melakukan proses pengisian dengan menekan tombol simpan LCD akan menampilkan pemberitahuan seperti pada gambar 4.27 dan gambar 4.28 setelah data selesai di transfer maka LCD akan menampilkan karakter seperti gambar 4.29 :

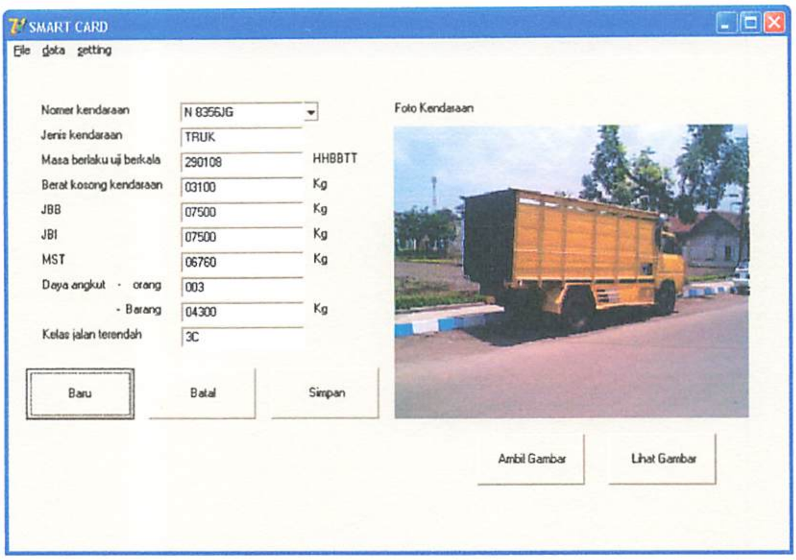
Gambar 4.27 Tampilan LCD Hub Komputer.

Gambar 4 .28 Tampilan LCD Proses Transfer.



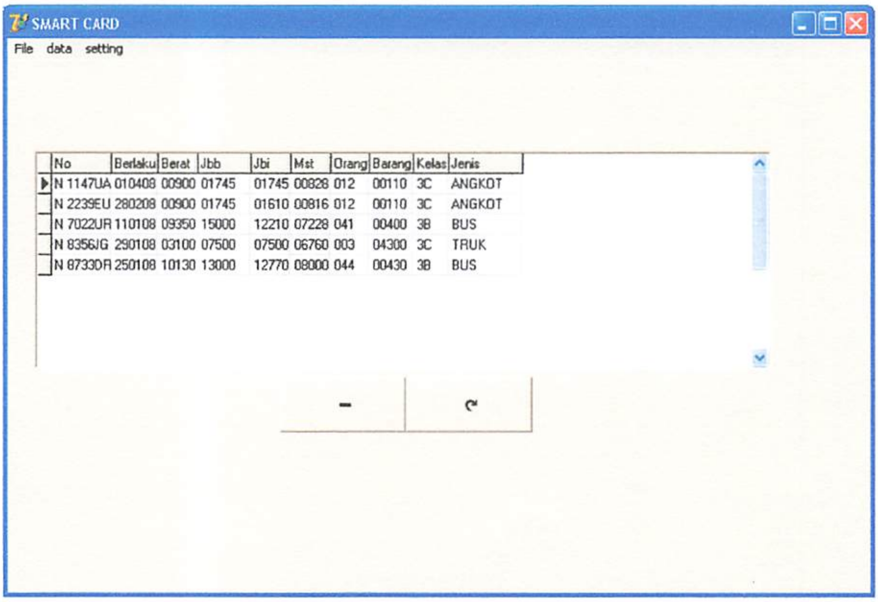
Gambar 4.29 Tampilan LCD Siap.

e. Apabila terdapat data pada database maka program delphi akan menampilkan data dari kartu tersebut pada program seperti gambar :



Gambar 4.30 Pengiriman data pada Program Delphi.

f. Jika kita ingin melihat data keseluruhan yang ada pada database maka dapat dilihat dengan memilih menu Data pada program delphi seperti yang ditampilakan pada gambar 4.31 dibawah:

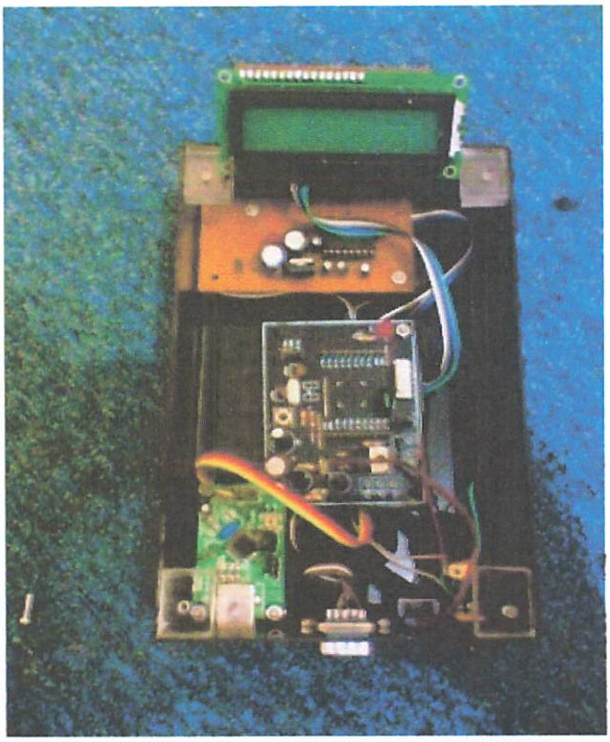


Gambar 4.31 Data Kendaraan pada Database.

4.5 Spesifikasi Alat

➢ **Alat daftar kir kendaraan**

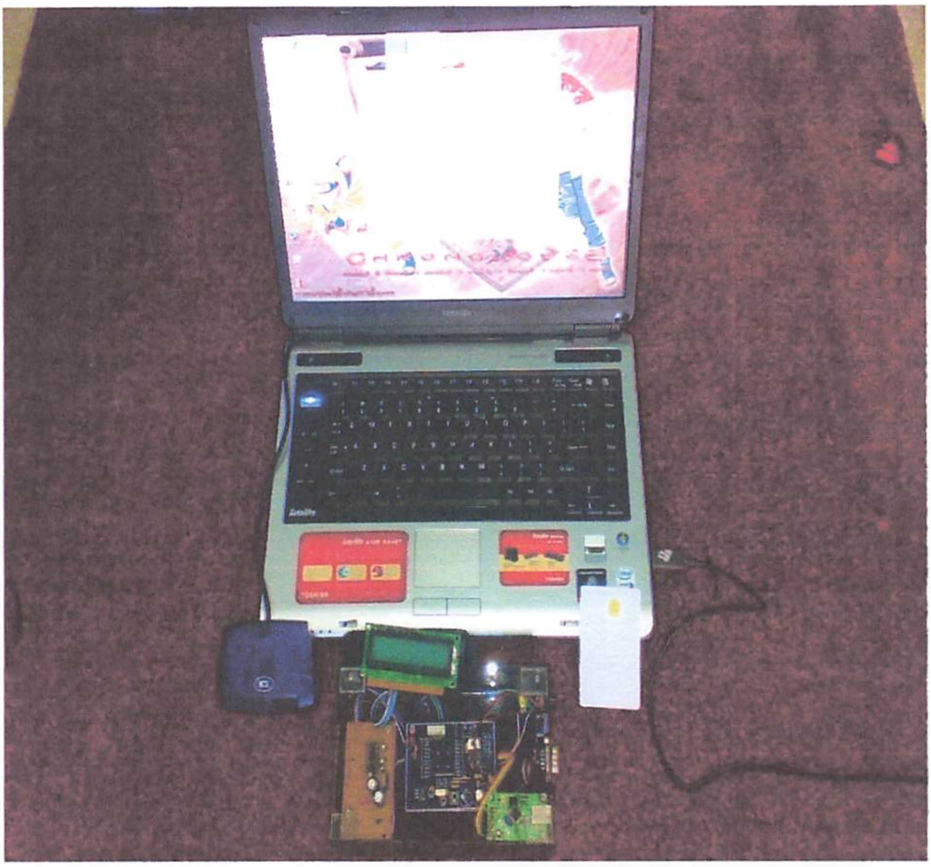
- Dimensi : 18 x 11,5 x 6 cm
- Mikrokontroler R8C/13 Tiny
- LCD M163
- Modul Reader Smartcard ACR30
- MAX 232
- IC 74LS164
- Kartu Smartcard Type Siemens SLE 4442



Gambar 4.32 Alat Daftar Kir Kendaraan

➢ **Alat Keseluruhan**

- Laptop Toshiba Satellite A135-S4487
- Perangkat Lunak Pemrograman Delphi7
- Modul ACR30
- Alat daftar kir
- Kartu Smartcard Type Siemens SLE 4442



Gambar 4.33 Alat Keseluruhan.

BAB IV

ANALISIS DAN PENGUJIAN ALAT

4.1. Pendahuluan

Dalam bab ini membahas tentang pengujian dari modul yang digunakan. Secara umum pengujian ini bertujuan untuk mengetahui apakah piranti yang telah direalisasikan dapat bekerja sesuai dengan perencanaan yang telah direncanakan.

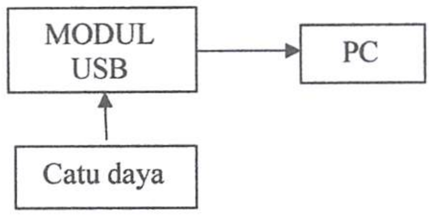
4.2 Tujuan Pengujian

Tujuan Pengujian pada masing-masing blok pada perancanggan dan pembuatan alat daftar kir kendaraan ini bertujuan:

- Untuk mengetahui apakah tiap blok rangkaian dapat berfungsi dengan baik.
- Untuk mengetahui sistem kerja alat daftar kir kendaraan dengan menggunakan mikrokontroller R8C/13 Tiny sebagai pengendali utama.

4.3 Pengujian Modul USB

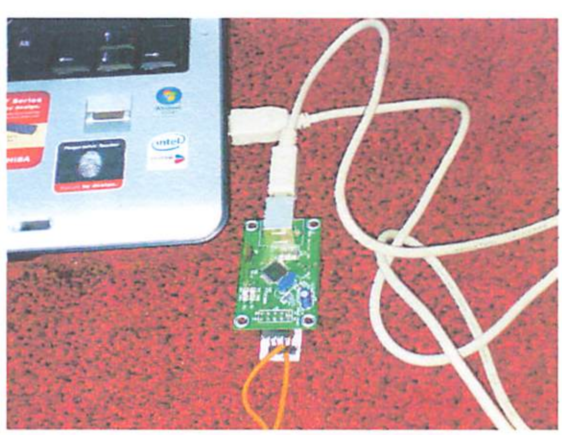
Tujuan dari pengujian ini adalah untuk mengetahui apakah Modul USB bisa digunakan untuk mengirim dan menerima data atau tidak. Adapun cara pengujianya dilakukan dengan cara memilih com yang dideteksi oleh PC kemudian mencoba melakukan pengiriman data melalui program sederhana yang telah dibuat ,data akan dikirim melalui kabel USB menuju Modul USB dengan menggunakan indikator apabila data di kirim dari PC ke USB maka led indikator pada modul (Rx) akan nyala dan sebaliknya pada saat data dikirim kembali dari USB ke PC maka indikator pada modul (Tx) akan menyala,dengan prosedur pengujian seperti dibawah ini :



Gambar 4.1. Blok Pengujian Modul USB

- Menghubungkan rangkaian Modul USB ke PC menggunakan kabel USB.
- Menyambung/Menjumper pin Tx dan Rx pada Modul USB.
- Menginstal driver modul USB.
- Membuka program pada delphi.

4.3.1 Sebelum melakukan proses penginstallan driver USB maka kita menghubungkan modul dengan PC mengunakan kabel USB seperti gambar 4.2 yaitu kabel konektor USB tipe A terhubung ke PC dan konektor tipe B terhubung pada Modul USB :



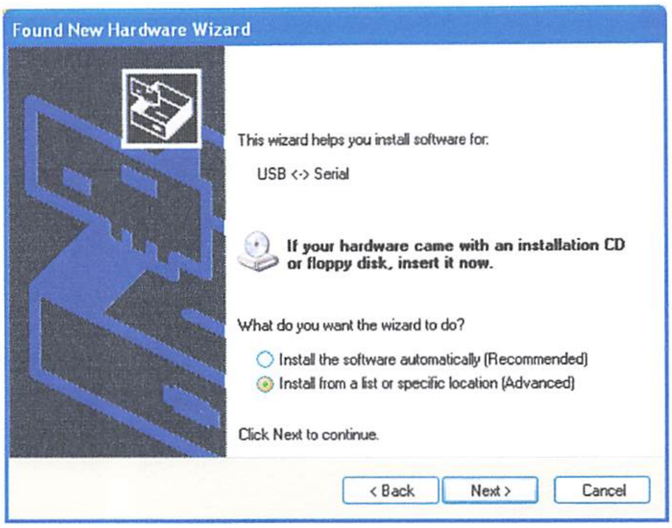
Gambar 4.2. Menghubungkan Modul USB dengan PC .

4.3.2 Setelah Modul terhubung dengan PC maka proses selanjutnya kita menginstal driver Modul USB tersebut yaitu :

- Pada saat kabel konektor USB dihubungkan pada PC maka PC akan mendeteksi adayan perangkat baru seperti pada gambar 4.3:

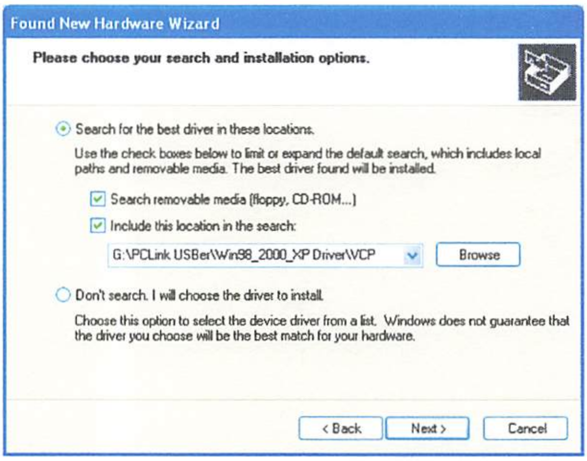
Gambar 4.3 PC Mendeteksi Hardware baru.

- Setelah terdeteksi maka perses selanjutnya menekan Button *Next*,setelah button ditekan maka maka akan muncul seperti gambar 4.4 Perintah tersebut merupakan permberitahuan apakah penginstallan dilakukan secara otomatis atau manual,tetapi pada saat menginstall dipilih yang manual:



Gambar 4.4 Perintah Untuk Memilih Cara Penginstallan.

- Proses selanjutnya saat button *Next* ditekan maka pc akan menampilkan perintah untuk memberitahu dimana lokasi driver USB tersebut seperti gambar 4.5 dan gambar 4.6 disini driver berada pada (drive G → PCLink → USBer → Win98_2000_XP Driver VCP) :



Gambar 4.5 Perintah Memberitahu Lokasi Driver Modul USB.

Gambar 4.6 Lokasi Driver.

- Apabila proses diatas telah selesai maka selanjutnya adalah proses pengisntallan dengan menekan button next yang selanjutnya PC akan momproses driver tersebut seperti gambar 4.7 dan setelah proses transfer selesai maka akan tampil bahwa driver baru telah selesai di install seperti gambar 4.8 :

Gambar 4.7 Proses Transfer Driver USB.

Gambar 4.8 Proses Penginstallan Selesai .

4.3.3 Setelah Proses penginstallan selesai maka untuk melakukan pengujian terhadap modul USB,digunakan sebuah program simpel delphi yang dibuat agar dapat digunakan untuk menguji modul USB tersebut yaitu :

❖ Progam Uji Komunikasi Modul USB :

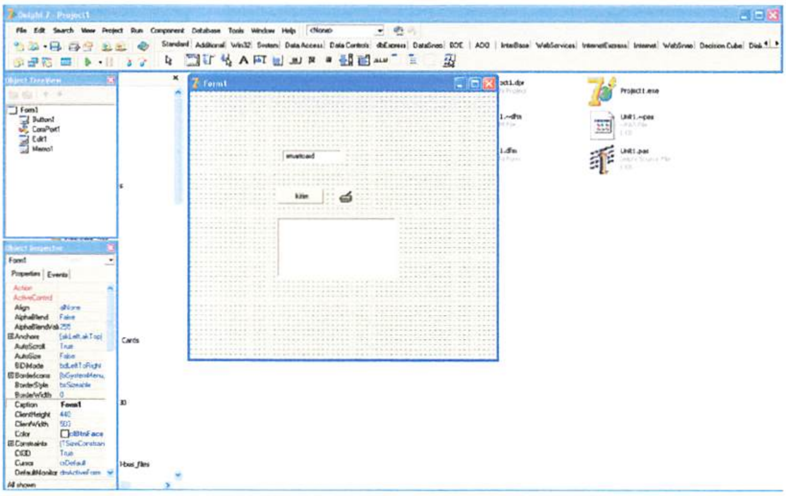
" Cuplikan program uji komunikasi serial di komputer menggunakan program Delphi 7.0 "

```
procedure TForm1.Button1Click(Sender: TObject);
begin
ComPort1.WriteStr('smartcard')
end;
procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
var data:string;
begin
ComPort1.ReadStr(data,count);
Memo1.Text:=Memo1.Text+data;
end;
end.
```

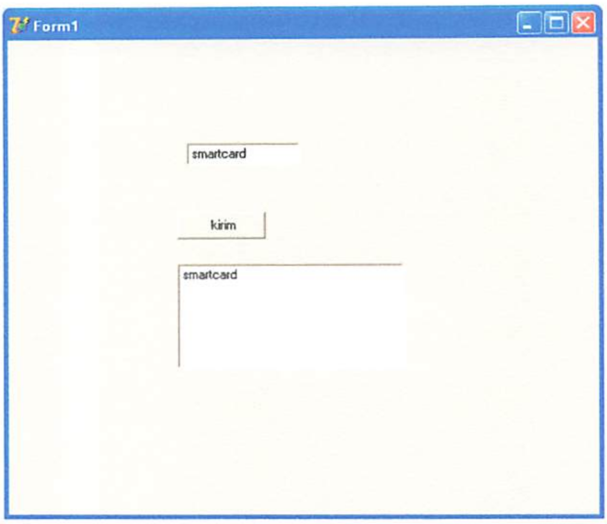
Program diatas sebagai percobaan mengirim data melalui PC ke modul USB di mana komponen yang digunakan pada delphi adalah

- Comport : komponen ini digunakan sebagai koneksi antara program delphi dengan modul USB dan pada percobaan ini PC mendeteksi COM6 untuk melakukan pengiriman
- Edit : Komponen ini digunakan sebagai input pada text memo.
- Memo : Komponen ini berfubgsi untuk menampilkan data.
- Button : Komponen ini merupakan tombol untuk mengirim data

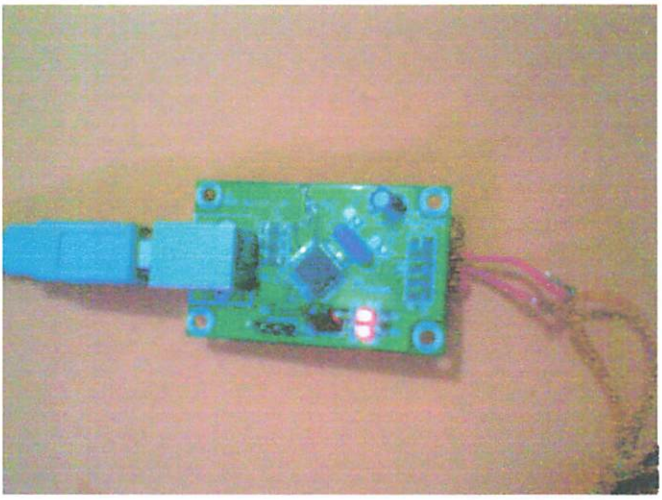
❖ Sebelum program dijalankan terlebih dahulu kita menyeting com yang dideteksi oleh PC seperti gambar 4.9 setelah itu program di runing Tampilan pada saat program dijalankan seperti gambar 4.10 dan tampilan pada Modul USB pada saat data dikirim seperti gambar 4.11:



Gambar 4.9 Tampilan Delphi pada Saat Penyetinggan Komponen Comport.



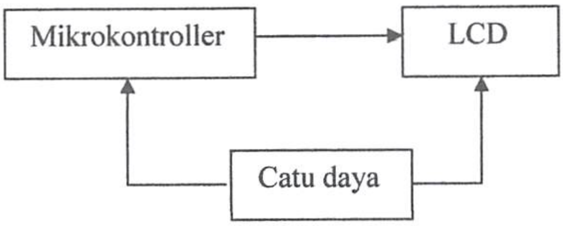
Gambar 4.10 Tampilan Delphi pada Saat Program diJalankan.



Gambar 4.11 Tampilan pada Modul USB.

4.4 Pengujian LCD

Tujuan dari pengujian ini adalah untuk mengetahui apakah LCD dapat berfungsi untuk menampilkan data yang diinginkan dengan baik atau tidak,adapun prosedur pengujianya yaitu



Gambar 4.12. Blok Pengujian LCD

1. Menghubungkan Mikrokontroller dengan LCD dan diberi tegangan atau catu daya.
2. Mengisi mikrokontroller dengan program seperti di bawah ini :

```
#include "sfr_r8c13.h"
#include "lcdku.c"
//---------------
// Program Utama
// ---------------------------------------------------------
void main ()
{//* Begin of Main
 initlcd();
 busek();
 cetak(1,1,"  SMARTCARD   ");
 cetak(2,1,"  03.17.018        ");
}//* End of Main
```

3. Apabila lampu menyala dan terdapat karakter pada LCD maka rangkaian berfungsi.

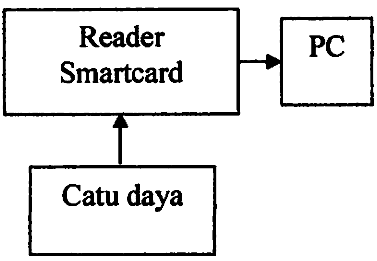
Dari hasil pengujian maka di dapat data seperti yang terlihat pada gambar berikut ini :

Gambar 4.13. Hasil Pengujian LCD .

4.5 Pengujian Modul Smartcard ACR30.

Tujuan dari pengujian ini adalah untuk mengetahui apakah Reader Smartcard dapat membaca kartu atau tidak.Adapun cara melakukan pengujian modul smartcard tersebut yaitu pada reader smartcard akan dilakukan proses reader aktif,pilih kartu,tulis,baca dan reader mati.

Adapun prosedur pengujian smartcard ini :

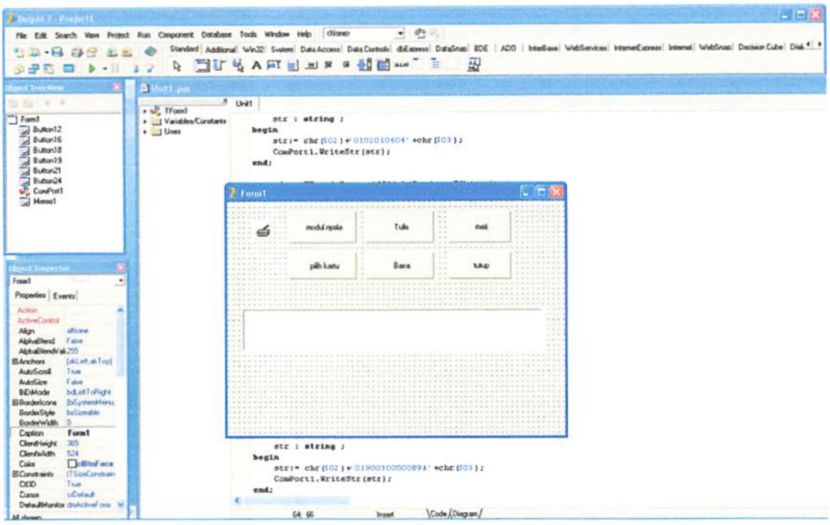


Gambar 4.14. Blok Pengujian Modul Smartcard ACR30

4.5.1 prosedure pengujian modul Smartcard.

a. Menghubungkan modul smartcard → PC.

b. Membuka program pada delphi yang telah dibuat untuk menguji reader smartcard .



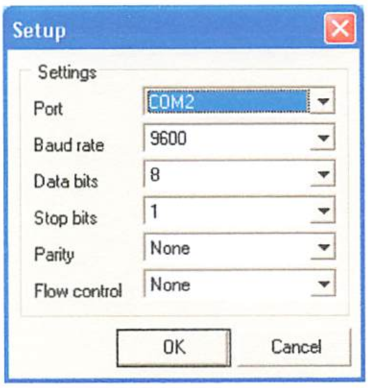
Gambar 4.15 Program Delphi.

Program Delphi:

```
procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
begin
  ComPort1.ReadStr(data,Count);
  memo1.Text := memo1.Text+data ;
end;
procedure TForm1. ButtonPilihkartu(Sender: TObject);
var
  str : string ;
begin
  str:= chr($02)+'0102010604'+chr($03);
  ComPort1.WriteStr(str);
end;
procedure TForm1. ButtonModulnyala(Sender: TObject);
var
  str : string ;
begin
  str:= chr($02)+'01800081'+chr($03);
  ComPort1.WriteStr(str);
end;
procedure TForm1.ButtonTutup(Sender: TObject);
begin
  ComPort1.Destroy ;
  close;
end;
procedure TForm1.Buttonmati(Sender: TObject);
var
  str : string ;
```

```
begin
   str:= chr($02)+'01810080'+chr($03);
   ComPort1.WriteStr(str);
end;
procedure TForm1.ButtonBaca(Sender: TObject);
var
   str : string ;
begin
   str:= chr($02)+'0190030000089A'+chr($03);
   ComPort1.WriteStr(str);
end;
procedure TForm1.Buttontulis(Sender: TObject);
var
   str : string ;
begin
   str:=
chr($02)+'0190000CFFFFFFFFFFFFFFFFFFFFFFFF9D'+chr($03);
   ComPort1.WriteStr(str);
end;
```

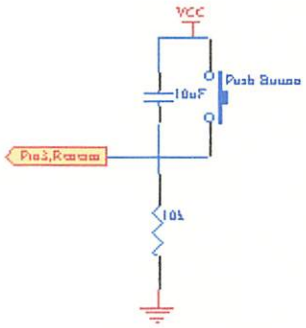
c. Setelah program dibuka terlebih dahulu dilakukan penyetingan COM yang dideteksi oleh PC:



Gambar 4.16. Tampilan Setting COM.

d. Setelah penyetingan COM selesai dan PC mendeteksi adanya hardware baru pada COM2 maka program dijalankan :

Rangkaian reset ditunjukkan dalam gambar di bawah ini :



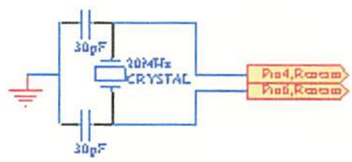
Gambar 3.4 Perencanaan Rangkaian Reset.

3.3.2 Rangkaian Clock

Kecepatan proses yang diperlukan oleh mikrokontroller Renesas R8C/13Tiny ditentukan oleh sumber *clock* yang mengendalikan mikrokontroller tersebut. Mikrokontroller Renesas R8C13/Tiny memiliki internal *clock* generator yang berfungsi sebagai sumber *clock* yang diperlukan. Untuk kristal *clock* dipasang Kristal dan resonator keramik yang berfungsi sebagai pembangkit *clock* osilator yang ada pada mikrokontroller.

Rangkaian ini terdiri dari dua buah kapasitor dan sebuah kristal. Untuk mengendalikan frekuensi osilatornya cukup dengan menghubungkan Kristal pada pin 4 (P4.7/X_{out}) dan pin 6 (P4.6/X_{in}) serta dua buah kapasitor ke *ground.*

Dalam minimum kristal ini, menggunakan kristal 20 Mhz dan $C_1 = C_2$ yaitu sebesar 30 pF. Dengan rangkaian sebagai berikut :



Gambar 3.5 Perencanaan Rangkaian *Clock.*

Dengan menggunakan nilai kristal dan kapasitor di atas maka dapat dihitung waktu yang diperlukan untuk 1 siklus mesin yaitu :

Diketahui : F = 20 MHz

$$T = \frac{1}{F}$$

Maka $T = \frac{1}{20\ MHz} = \frac{1}{20}\ \mu s$

Maka untuk 1 siklus mesin dari mikrokontroller Renesas R8C/13 Tiny adalah sebesar:

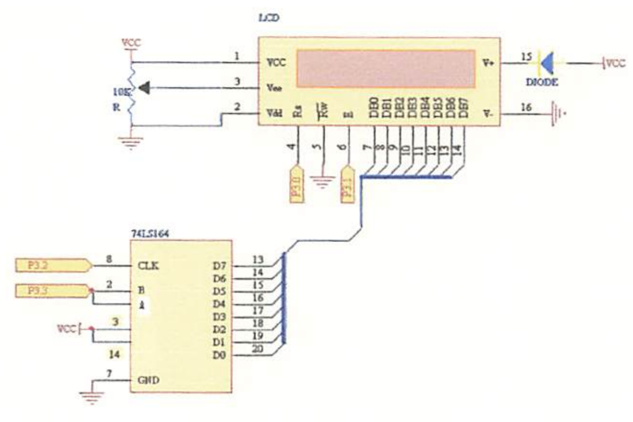
$$Tm = 20 \text{ x } T$$

$$= 20 \text{ x } \frac{1}{20}\ \mu S$$

$$= 1 \mu S$$

3.4 Perencanaan Rangkaian *Liquid Crystal Display* (LCD)

Dalam aplikasi ini menggunakan sebuah layar LCD (*Liquid Crystal Display*) yaitu jenis *Seiko Instrument* M1632 yang merupakan LCD dua baris dengan setiap barisnya terdiri 16 karakter dan menggunakan IC 74LS164 yang merupakan register geser 8 bit yang memiliki jalan masuk deret tergerbang. Gambar hubungan antara LCD, IC74LS164 dan mikrokontroler dapat dilihat dalam gambar berikut :



Gambar 3.6 Perencanaan Rangkaian LCD.

LCD dot matrik ini membutuhkan sepuluh buah pin masukan/keluaran dari mikrokontroler dan IC 74164 Adapun dua buah pin yakni port 3.0 pada penyemat RS yang digunakan sebagai sinyal pemilih register dan port 3.1 pada penyemat Enable digunakan sebagai sinyal operasi awal, sinyal enable ini mengaktifkan data tulis atau baca oleh mikrokontroler, penyemat DB0-DB7 yang dihubungakan ke pin data IC74164 digunakan untuk menampilkan karakter yang dikehendaki oleh mikrokontroler. Ketika terdapat data pada jalur data, data tersebut akan ditahan dengan memberikan *clock* pin E pada LCD. Pin RS menentukan apakah data yang ditahan akan digunakan sebagai instruksi untuk mengatur *setting* tampilan pada LCD atau sebagai kode karakter yang diperlukan LCD untuk menampilkan suatu karakter. Sedangkan untuk pin R/W pada LCD dihubungkan ke *ground* karena dalam hal ini LCD hanya melakukan operasi write atau operasi menampilkan karakter.

Untuk pin Vcc pada LCD dihubungkan ke supply +Vcc dan Vss dihubungkan ke *ground.* Pin V_{EE} beserta pin Vcc dan Vss dihubungkan ke *trimer potensio* atau kadang disebut dengan *trimpot. Trimpot* ini digunakan untuk mengatur kontras dari tampilan LCD dengan cara mengubah tegangan pada pin V_{EE}. Daftar tabel fungsi penyemat pada LCD dapat dilihat dalam Tabel 3.1.

Tabel 3.1. Fungsi penyemat LCD.

Penyemat	Fungsi
DB0 – DB7	Merupakan saluran data, berisi perintah dan data yang akan ditampilkan di LCD.
Enable	Sinyal operasi awal, sinyal ini mengaktifkan data tulis atau baca.
R/W	Sinyal seleksi tulis atau baca 0: tulis 1: baca
RS	Sinyal pemilih *register* 0: masukan data 1: masukan instruksi

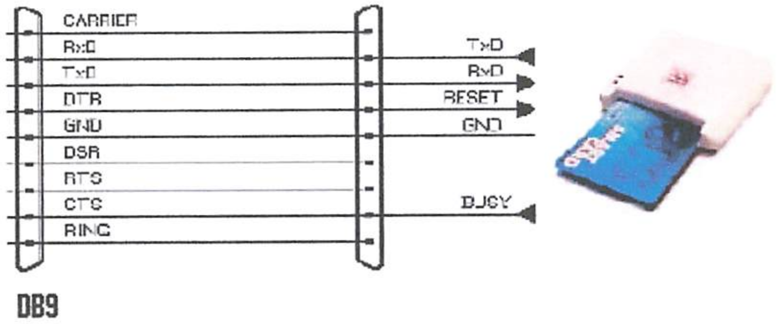
Seperti telah disebutkan sebelumnya bahwa data yang terdapat pada jalur data selain dianggap sebagai kode karakter dapat digunakan sebagai suatu perintah instruksi untuk mengatur setting dari tampilan LCD. Cara pemakaian data antara sebagai instruksi dengan kode karakter berbeda. Perbedaan hanyalah keadaan pin RS ketika data yang ada dijalur data ditahan oleh LCD dengan memberikan *clock* pada pin E.

Pin – pin yang digunakan adalah

- Pin DB0-DB7 terhubung pada IC 74LS164 yang kemudian pada IC tersebut terhubung pada renesas dengan pin 2 (Data) terhubung pada P 3.0 dan pin 8 (CLK) terhubung pada port 3.1 sedangkan pin 7 (Ground).
- Pin Enable pada LCD terhubung pada renesas yaitu port 3.2.
- Pin RS pada LCD terhubung pada renesas yaitu port 3.3.

3.5 Smartcard Reader ACR30

Pada pembuatan skripsi,ACR30 merupakan sebuah modul reader smartcard yang dapat membaca kartu jenis SLE4442 dengan kapasitas memori 256 byte.di dalam melakukan komunikasi dengan mikrokontroler (R8C/13Tiny) smartcard reader terhubung dengan RS232,adapun pin – pin yang digunakan dalam modul yang akan terhubung dengan DB9/RS232.



Gambar 3.7 Pin yang digunakan pada Smartcard.

Pini – pin yang digunakan adalah

- Pin 2

  Pin ini merupakan pin Tx dari modul ACR30.

- Pin 3

  Pin ini merupakan pin Rx dariModul ACR30.

- Pin 4

  Pin ini adalah pin reset pada ACR30 yang terhubung pada mikrokontroler (DTR)

- Pin 5

  Pin ini berfungsi sebagai ground

- Pin 8

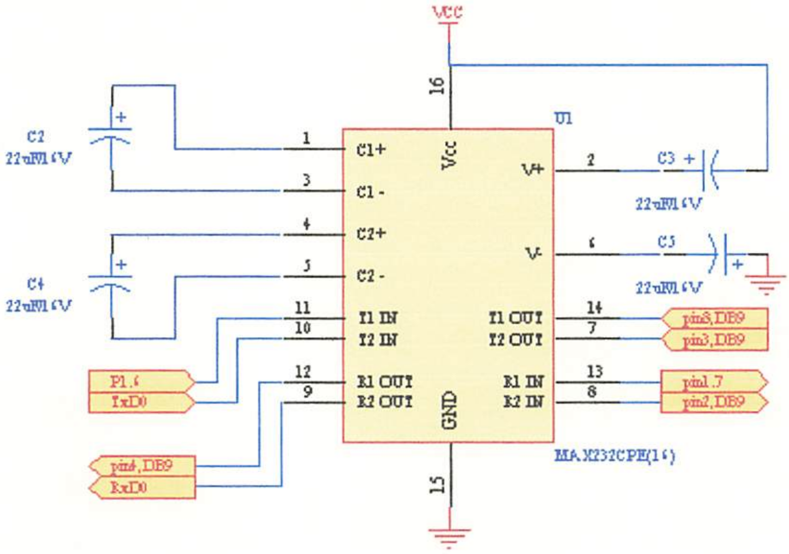
  Pin ini sebagai pin sebagai CTS (Clear To Send).

3.6 Rangkaian Antar Muka RS-232

Pada perencanaan hubungan antara *MCU* dan *Modul Smartcard ACR30* menggunakan komunikasi data secara *serial.* Adapun kaki/ pin-pin yang dipakai adalah

- pin 2 pada DB9 sebagai Rs.
- pin 3 pada DB9 sebagai Tx.
- Pin 4 pada DB9 Sebagai DTR
- Pin 5 pada DB9 sebagai ground.
- Pin 8 pada DB9 sebagai CTS

Level tegangan dari RS-232 harus disesuaikan ke level tegangan TTL menggunakan IC MAX 232. Kecepatan transfer data per *bit* menggunakan 9600 bps. Rangkaian Interface RS-232 diperlihatkan pada gambar 3.8.



Gambar 3.8 Rangkaian Antar Muka RS-232.

3.7 Perancangan Perangkat Lunak

Setelah semua perangkat keras telah selesai dikerjakan pada tahap selanjutnya perangkat lunak (Sofware) yang akan menangani sistem rangkaian .Pada perangkat lunak inilah kita dapat menentukan bagaimana sistem rangkaian ini akan bekarja,pada bagian inilah semua tata kerja rangkaian ditentukan .dalam merancang perangkat lunak ini ,mengunakan dua buah sofware yaitu sofware pada mikrokontroler dan sofware pada delphi yaitu :

- Pada pemrograman di mikrokontroler mengunakan renesas R8C13/Tiny mengunakan bahasa C dengan compiler yang dipaket bersama pada suatu IDE yaitu HEW (High-Performance Embedded Workshop).
- Pada PC (komputer) mengunakan sefware delphi 7,di dalam delphi7 ini ada beberapa komponen yang di gunakan untuk

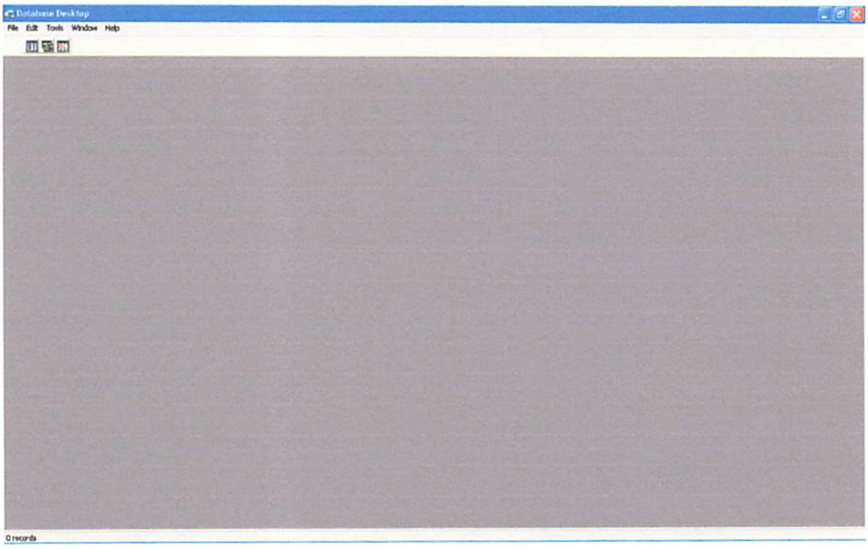
mengakses/menghubungkan sofware pada mikrokontroler dengan PC baik itu dalam menampilkan data maupun dalam mengakses database. Komponen tersebut adalah :

- ✓ Main menu : komponen ini terletak pada bagian standart yang fungsinya untuk mendesain dan menciptakan menu bar yang ada pada form.
- ✓ Table : komponen ini terletak pada bagian BDE yang funsinya untuk menghubungkan tabel pada suatu database dengan program yang dibuat.
- ✓ DataSource : komponen ini terletak pada bagian Data Access yang fungsinya untuk menghubungkan komponen Table atau Query dengan komponen tempat data akan ditampilkan.
- ✓ DBGrid : komponen ini terletak pada Data Controls yang fungsinya untuk menampilkan data-data dalam bentuk baris dan kolom.
- ✓ DBNavigator : komponen ini terletak pada Data Controls yang fungsinya untuk membuat pengontrol yang bisa menavigasi database dan mempunyai kemampuan untuk mengubah data tersebut
- ✓ Comport : komponen ini terletak pada CPortLib yang fungsinya untuk menyeting COM,Baud rate,Data bit,Stop bit,Parity dan Flowcontrol.
- ✓ OpenPictureDialog / SavePictureDialog : komponen ini berfungsi untuk menggambil dan menyimpan gambar yang akan dimasukkan.

3.8 perancangan database

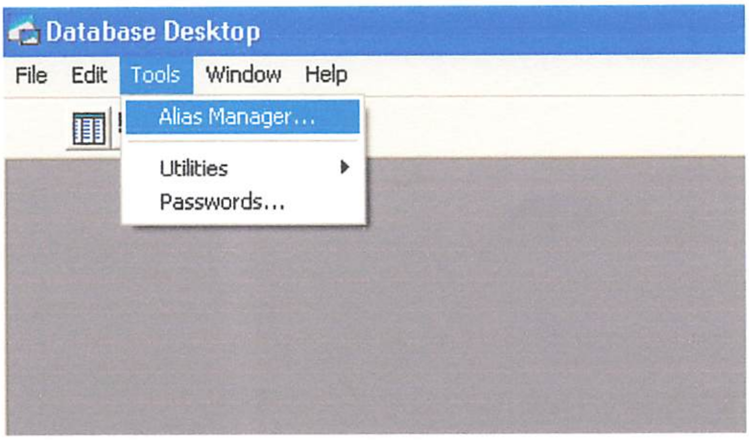
Pada perancangan database disini menggunakan database paradox7 dengan mengunakan adapun tahap-tahap dalam perancangan tersebut yaitu :

a) Membuka file database desktop pada delphi.

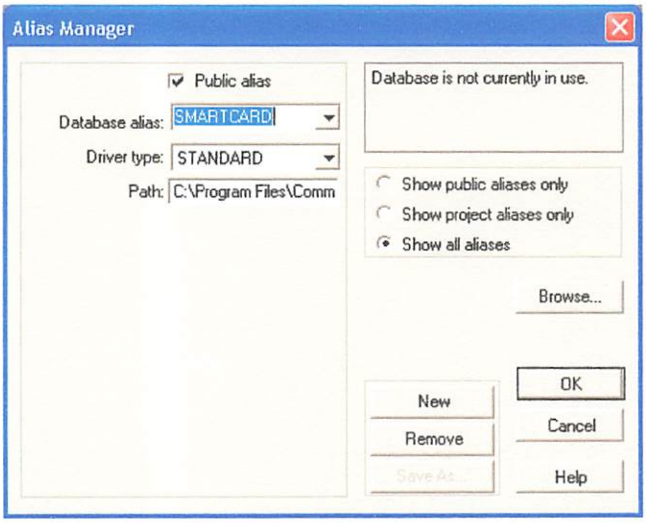


Gambar 3.9 Membuka File Database dekstop.

b) Mengisikan Alias dari database yang akan kita buat pada kolom database yaitu dengan cara memilih Tolls→ Alias Manager.

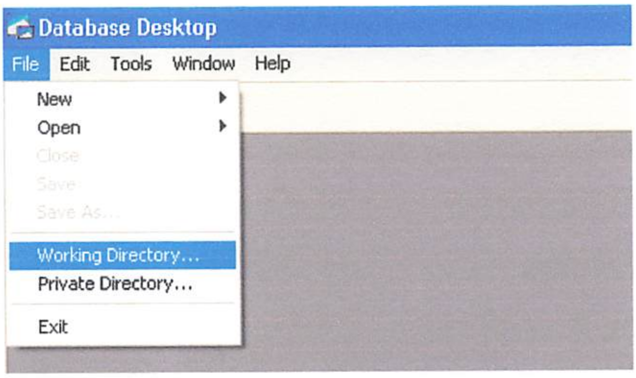


Gambar 3.10 Membuka Alias Manager.



Gambar 3.11 Mengisi Nama Alias.

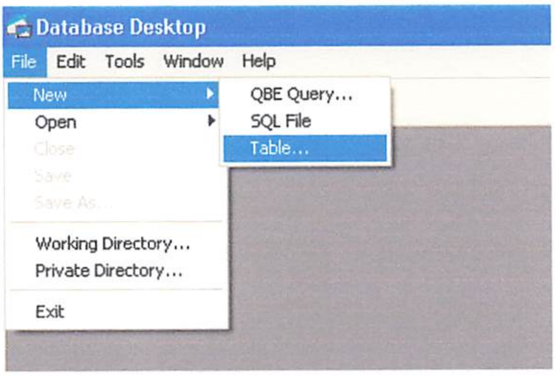
c) Setelah langkah diatas selesai maka selanjutnya adalah menentuka Working Directory dengan cara pilih file → Working Directory seperti berikut :



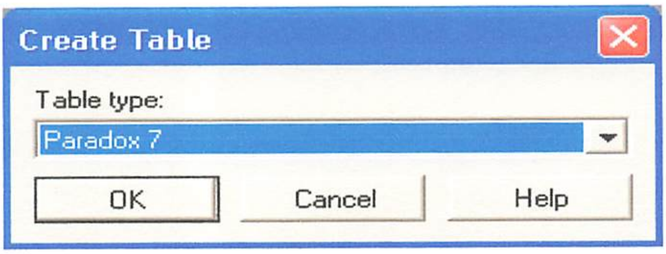
Gambar 3.12 Working Directory.

Gambar 3.13 Menentukan Working Directory.

d) Setelah langkah diatas semua selesai dikerjakan maka selanjutnya adalah langkah untuk membuat Table baru yaitu dengan cara pilih File → New→ Tabel selanjutnya pilih tipe tabel Paradox7.

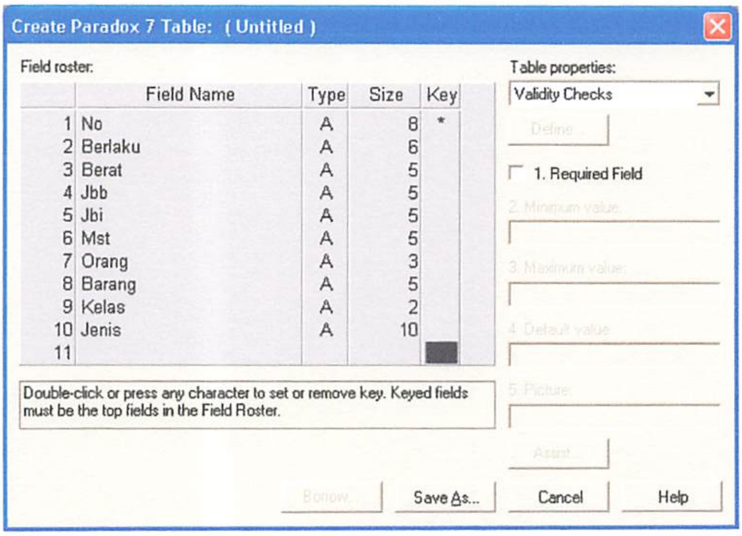


Gambar 3.14 Membuat Table Baru.



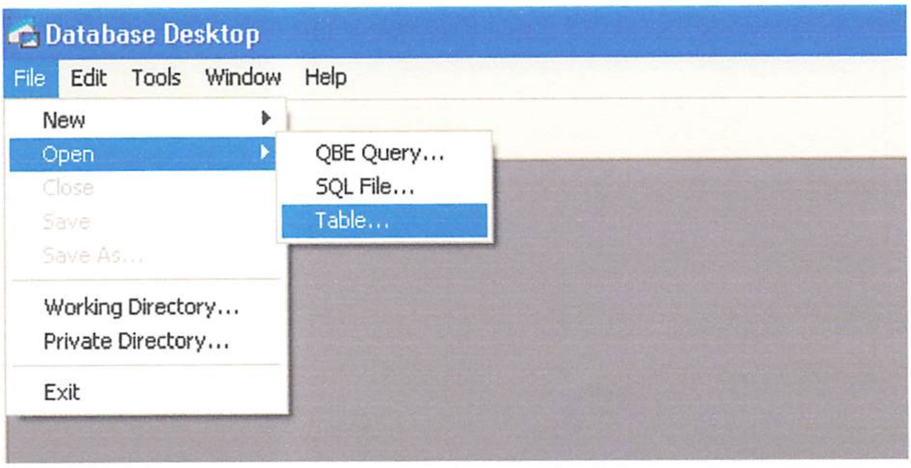
Gambar 3.15 Menentukan Tipe Tabel.

e) Selanjutnya mengisi tabel yang akan dibuat :

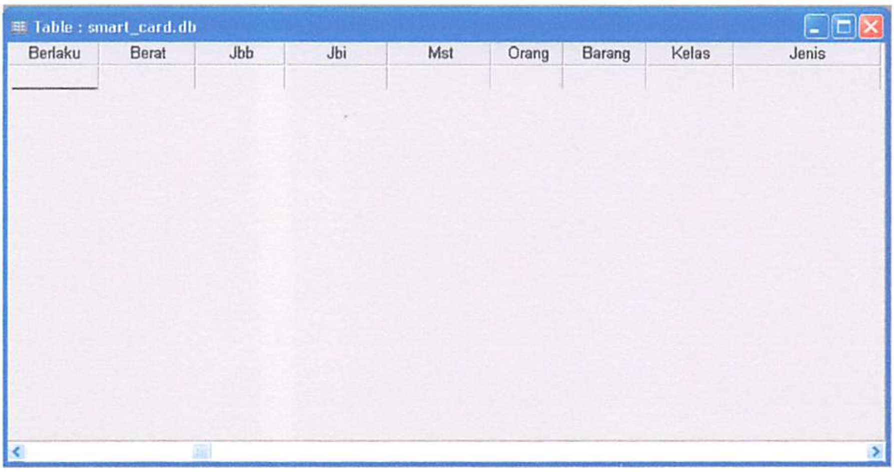


Gambar 3.16 Mengisi data yang akan dijadikan Table.

f) Selanjutnya file table disimpan kemudian untuk melihat tabel yang telah kita buat yaitu dengan cara pilih file → open → tabel.setelah itu tabel dapat kita hubungkan dengan delphi.

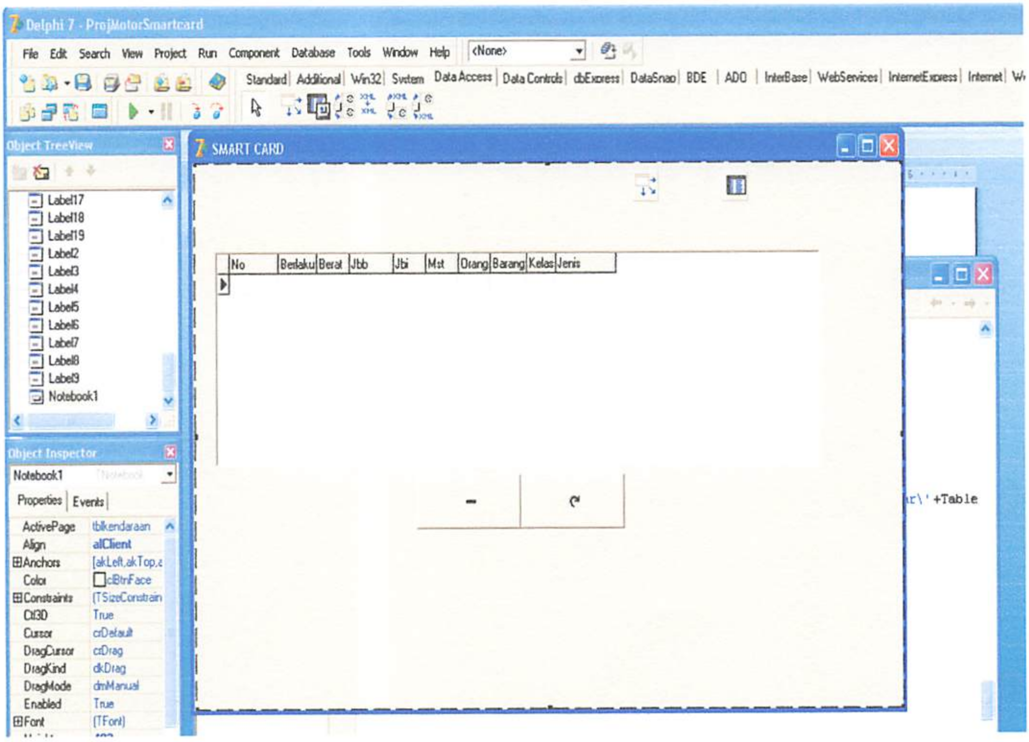


Gambar 3.17 Membuka Tabel.



Gambar 3.18 Gambar Tabel Yang Telah Dibuat.

g) Selanjutnya kita menghubungkan tabel tersebut pada delphi dengan menggunakan komponen Table dan Data Source



Gambar 3.19 Gambar menghubungkan tabel dengan delphi.

BAB V

PENUTUP

5.1. Kesimpulan

Dari perancangan dan pembuatan alat daftar kir kendaraan bermotor berbasis smartcard ini maka dapat diambil kesimpulan sebagai berikut:

1. Dari haril pengujian,pada saat konektor kabel USB tipe A dihubungkan ke PC dan di pindah–pindah maka PC akan mendeteksi COM yang berbeda - beda.
2. Program pada PC tidak berjalan apa bila tidak sesuai Hardware yang dibuat dan tidak sesuai COM yang dideteksi oleh PC.
3. Dari hasil pengujian dengan 5 buah kartu yang dimasukkan berulang–ulang sebanyak 25 kali tidak terdapat eror dalam pembacaan kartu.
4. Dari hasil pengujian, dalam proses pembacaan dan proses penulisan dengan menggunakan kabel perpanjangan USB sepanjang 3 meter tidak terdapat eror.

5.2. Saran

Aplikasi sistem dari daftar kir kendaraan yang dikendalikan oleh mikrokontroler R8C/13 Tiny ini masih memiliki keterbatasan, nantinya diharapkan dapat dikembangkan untuk mengatasi keterbatasan itu.Seperti Untuk pengembangan database selain informasi mengenai daftar kir bisa ditambahkan untuk informasi lain yang berkaitan tentang kendaraan tersebut dengan memanfaatkan teknologi Smartcard.

DAFTAR PUSTAKA

[1] Innovative_electronics.

[2] Sutawi, Dwi ,"I/O BUS & Motherboard"Andi,Yogyakarta,2004.

[3] Data Sheet Renesas R8C/13 Tiny.

[4] Data Sheet LCD M1632

[5] Margoselo,Bambang Dwi Cahyo" Tinjauan Smart Card untuk Pengamanan Database Berbasis Komputer"Bandung,2003.

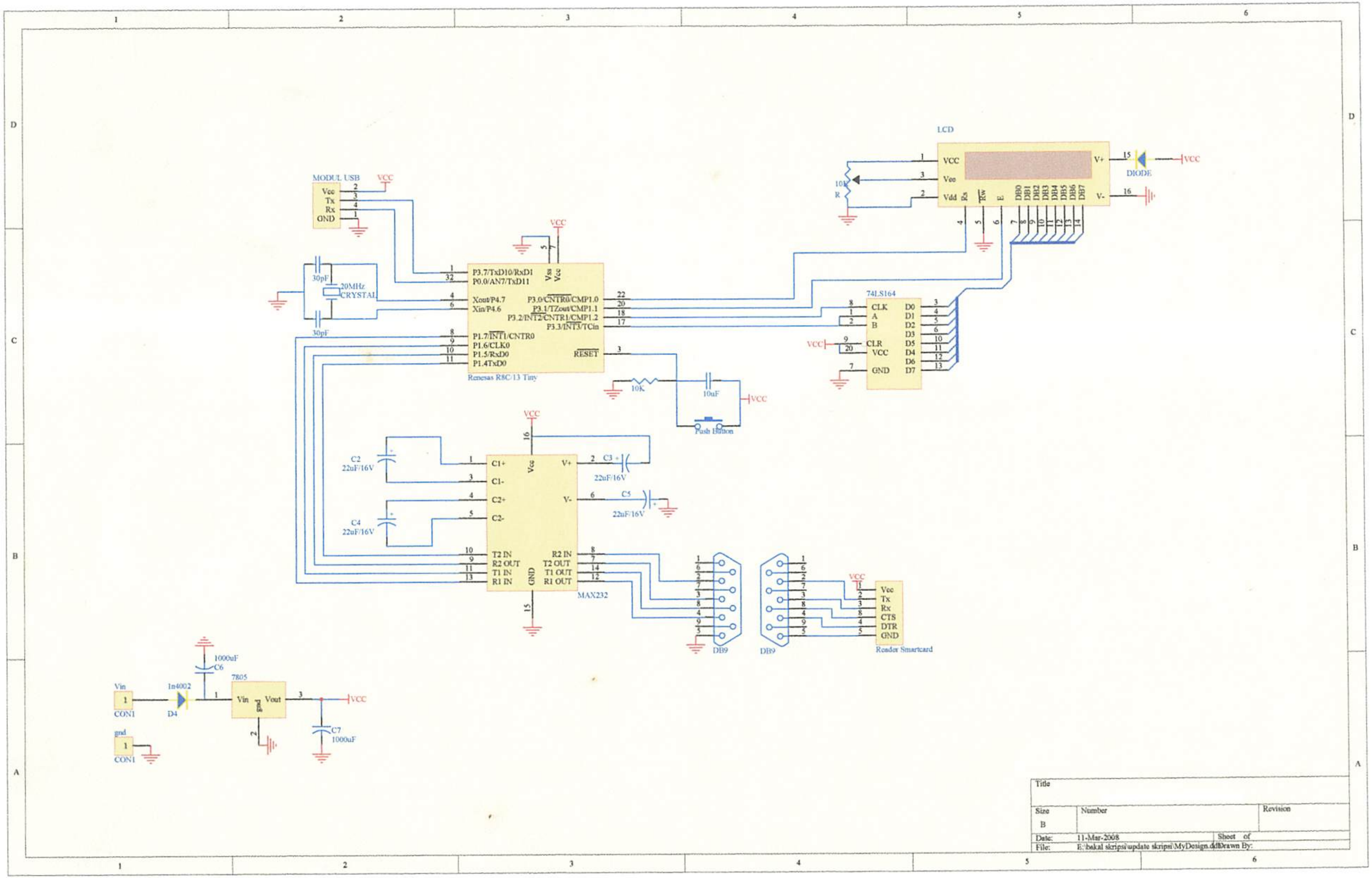
[6] Reference Manual ACR30 Smartcard Reader version 3.2,11-2005.

[7] Data Sheet Max 232.

[8] Data Sheet SLE 4442

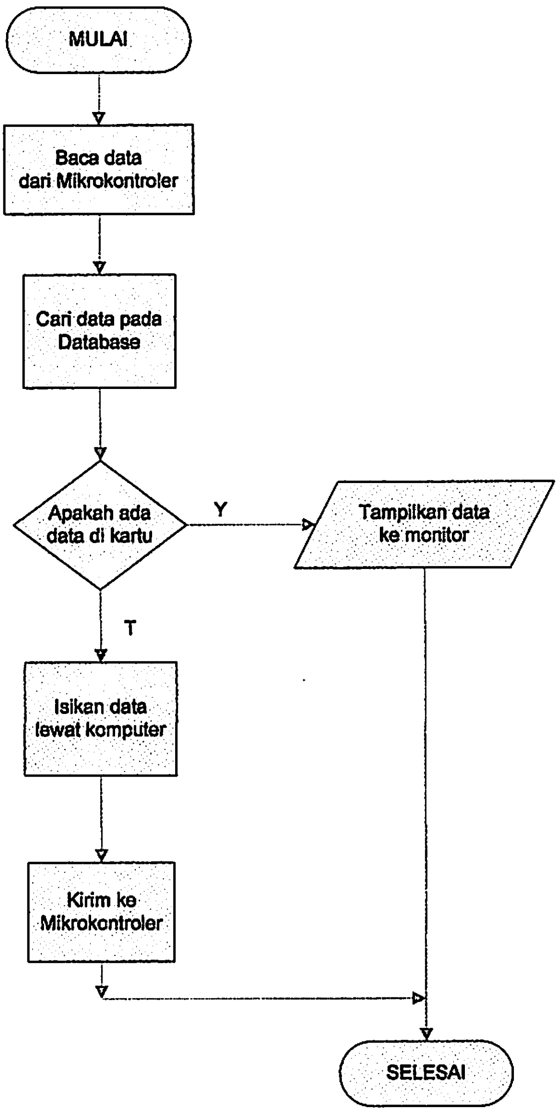
LAMPIRAN

Rangkaian


LCD
VCC
Vee
Vdd
Rs
RW
E
DB0 DB1 DB2 DB3 DB4 DB5 DB6 DB7
V+
V-
DIODE
VCC
10K
R
MODUL USB
Vcc
Tx
Rx
GND
30pF
20MHz
CRYSTAL
P3.7/TxD10/RxD1
P0.0/AN7/TxD11
Xout/P4.7
Xin/P4.6
P1.7/INT1/CNTR0
P1.6/CLK0
P1.5/RxD0
P1.4TxD0
Vss
Vcc
P3.0/CNTR0/CMP1.0
P3.1/TZout/CMP1.1
P3.2/INT2/CNTR1/CMP1.2
P3.3/INT3/TCin
RESET
Renesas R8C/13 Tiny
74LS164
CLK
A
B
CLR
VCC
GND
D0 D1 D2 D3 D4 D5 D6 D7
10K
10uF
Push Button
C2
22uF/16V
C4
22uF/16V
C1+
C1-
C2+
C2-
T2 IN
R2 OUT
T1 IN
R1 IN
Vcc
GND
V+
V-
R2 IN
T2 OUT
T1 OUT
R1 OUT
MAX232
C3
22uF/16V
C5
22uF/16V
DB9
DB9
Vcc
Tx
Rx
CTS
DTR
GND
Reader Smartcard
Vin
CON1
gnd
CON1
1n4002
D4
1000uF
C6
7805
Vin
Vout
gnd
C7
1000uF
Title
Size
B
Number
Revision
Date:
11-Mar-2008
Sheet of
File:
E:\bakal skripsi\update skripsi\MyDesign.ddb
Drawn By:

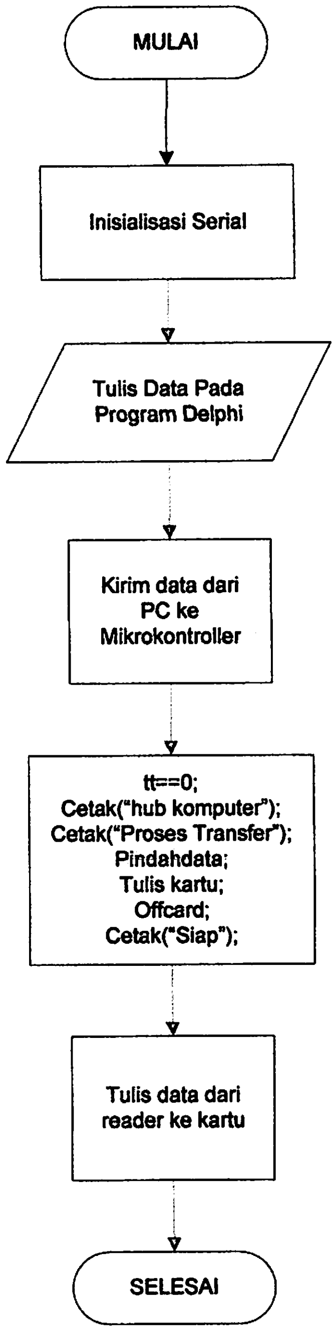
PROGRAM

```c
/*************************************************************/
/*                                                           */
/* FILE        :tes.c                                        */
/* DATE        :Thu, DES 28, 2007                            */
/* DESCRIPTION :Main Program                                 */
/* CPU TYPE    :Other                                        */
/*                                                           */
/* This file is generated by Renesas Project Generator (Ver.4.0).  */
/*                                                           */
/*************************************************************/

#include <stdio.h>
#include "sfr_r813.h"
#include "timer.h"
#include "lcdku.h"
#include "serial.h"

#define fbreset"0190000091"
#define fbmasuk        "01FF0100FF"
#define fbkeluar "01FF0200FC"
#define dtr            p1_6
#define cts            p1_7

union byte_def tanda_addr;
#define   tanda   tanda_addr.byte
#define   tt   tanda_addr.bit.b0
#define   tsb   tanda_addr.bit.b1
#define   tkm   tanda_addr.bit.b2
#define   tcom   tanda_addr.bit.b3
unsigned char a,tangkap[100],i,i2,d[100],tangkap1[100];
long itimer;
unsigned char kon(unsigned char n)
  {
   if (n <10) return(n+0x30);
   else return(n+0x37);

  }
unsigned char nok(unsigned char n)
  {
   if (n < 0x3a) return(n-0x30);
   else return(n-0x37);

  }
void pindahdata()
{unsigned char aa;
```



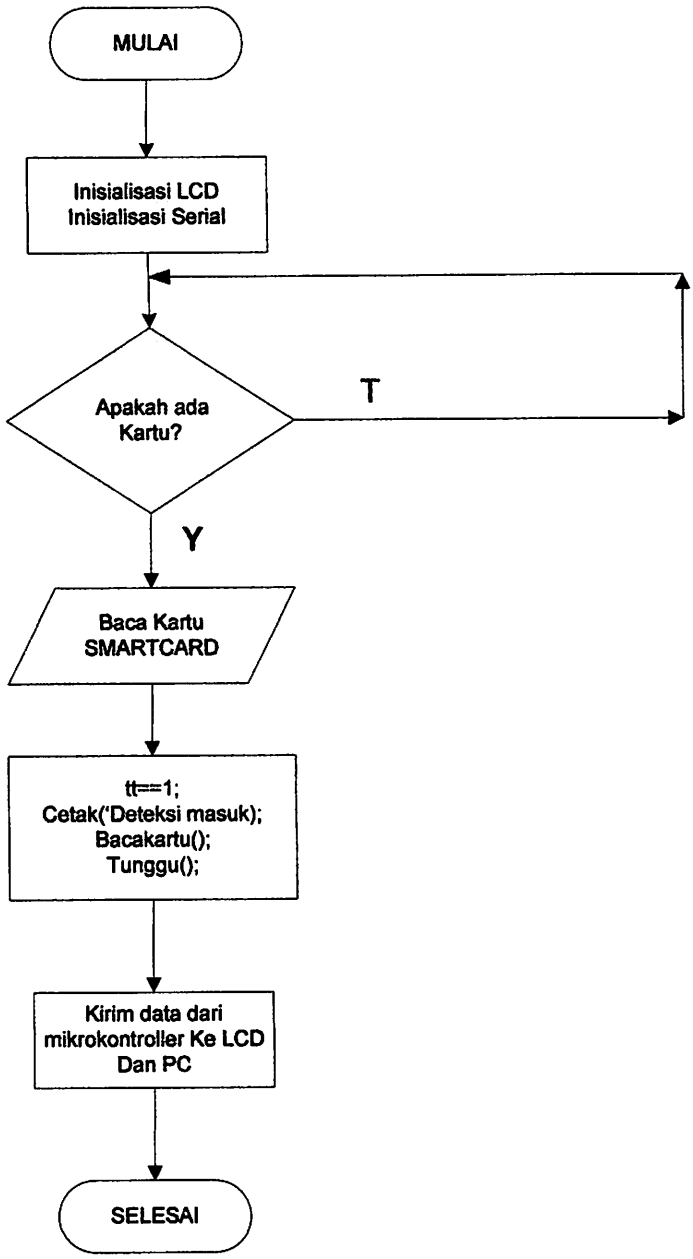
Gambar 3.21

Diagram Alir Program Komputer.



Gambar 3.22

Diagram Alir Tulis Kartu Smartcard.

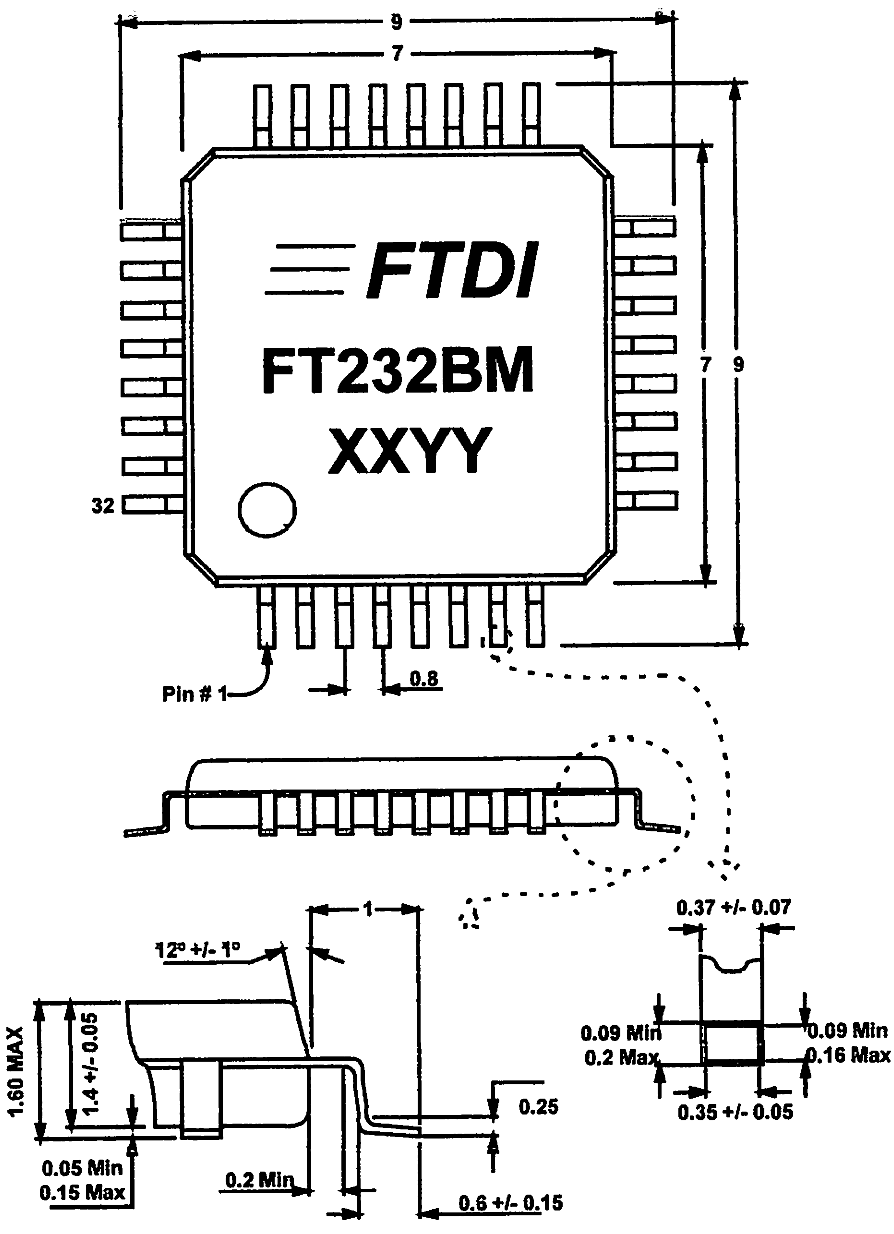


Gambar 3.23

Diagram Alir Baca Kartu Smartcard.

ıckage Outline

; – 32 LD LQFP Package Dimensions



ᐧT232BM is supplied in a 32 LD LQFP package as standard. This package has a 7mm x 7mm body (9mm ɲ including leads) with leads on a 0.8mm pitch. An alternative 5mm x 5mm leadless chip scale package is available on special request for projects where package area is critical.

The above drawing shows the LQFP-32 package – all dimensions are in millimetres.

XXYY = Date Code (XX = 2 digit year number, YY = 2 digit week number.

bsolute Maximum Ratings

'e the absolute maximum ratings for the FT232BM device in accordance with the Absolute Maximum Rating System 34). Exceeding these may cause permanent damage to the device.

age Temperature .... –65°C to + 150°C
ient Temperature (Power Applied)...................................... 0°C to + 70°C
Supply Voltage .... -0.5v to +6.00v
nput Voltage - Inputs .... -0.5v to VCC + 0.5v
nput Voltage - High Impedance Bidirectionals .... -0.5v to VCC + 0.5v
)utput Current – Outputs 24mA
)utput Current – Low Impedance Bidirectionals 24mA
er Dissipation (VCC = 5.25v) 500mW
trostatic Discharge Voltage (I < 1uA) .... +/- 2000v
n Up Current (Vi < 0 or Vi > Vcc) 100mA

.C. Characteristics

racteristics (Ambient Temperature = 0 .. 70°C)

ng Voltage and Current

eter	Description	Min	Typ	Max	Units	Conditions
	VCC Operating Supply Voltage	4.4	5.0	5.25	V	
	VCCIO Operating Supply Voltage	3.0	-	5.25	V	
	Operating Supply Current	-	25	-	mA	Normal Operation
	Operating Supply Current	-	180	200	uA	USB Suspend ** Note 1

· Supply current excludes the 200uA nominal drawn by the external pull-up resistor on USB DP.

) Pin Characteristics (VCCIO = 5.0v)

eter	Description	Min	Typ	Max	Units	Conditions
	Output Voltage High	4.4	-	4.9	V	I source = 2mA
	Output Voltage Low	0.1	-	0.7	V	I sink = 4 mA
	Input Switching Threshold	1.1	1.5	1.9	V	Note 2
	Input Switching Hysteresis		200		mV	

) Pin Characteristics (VCCIO = 3.3v)

eter	Description	Min	Typ	Max	Units	Conditions
	Output Voltage High	2.7	-	3.2	V	I source = 2mA
	Output Voltage Low	0.1	-	0.7	V	I sink = 4 mA
	Input Switching Threshold	1.0	1.4	1.8	V	Note 2
	Input Switching Hysteresis		200		mV	

- Inputs have an internal 200k pull-up resistor to VCCIO.

OUT Pin Characteristics

er	Description	Min	Typ	Max	Units	Conditions
	Output Voltage High	4.0	-	5.0	V	Fosc = 6MHz
	Output Voltage Low	0.1	-	1.0	V	Fosc = 6MHz
	Input Switching Threshold	1.8	2.5	3.2	V	

TEST, EECS, EESK, EEDATA, IO Pin Characteristics

er	Description	Min	Typ	Max	Units	Conditions
	Output Voltage High	4.4	-	4.9	V	I source = 2mA
	Output Voltage Low	0.1	-	0.7	V	I sink = 4 mA
	Input Switching Threshold	1.1	1.5	1.9	V	Note 3
	Input Switching Hysteresis		200		mV	

EECS and EEDATA pins have an internal 200k pull-up resistor to VCC

Pin Characteristics

er	Description	Min	Typ	Max	Units	Conditions
	Output Voltage High	3.0	-	3.6	V	I source = 2mA
	Leakage Current Tri-State	-	-	5	uA	

in Characteristics

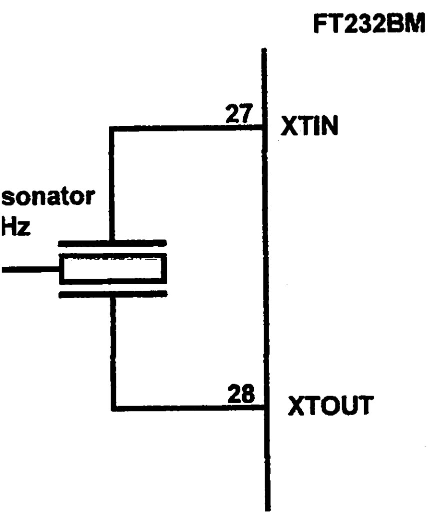
er	Description	Min	Typ	Max	Units	Conditions
	IO Pins Static Output (High)	2.8		3.6v	V	RI = 1k5 to 3V3Out (D+) RI = 15k to GND (D-)
	IO Pins Static Output (Low)	0		0.3	V	RI = 1k5 to 3V3Out (D+) RI = 15k to GND (D-)
	Single Ended Rx Threshold	0.8		2.0	V	
	Differential Common Mode	0.8		2.5	V	
	Differential Input Sensitivity	0.2			V	
	Driver Output Impedance	29		44	ohm	Note 4

Driver Output Impedance includes the external 27R series resistors on USBDP and USBDM pins.

vice Configuration Examples

cillator Configurations

I
eramic Resonator
ration



llustrates how to use the FT232BM with eramic Resonator such as Murata Part # /I00G53 or equivalent. 3-Pin resonators have capacitors built into the resonator so no external apacitors are required. This makes for an al configuration. Though the typical accuracy of sonator is +/- 0.5% and is technically out-with specification, it has been calculated that using evice will work satisfactorily in practice with the /I design.

Figure 5
Crystal or 2-Pin Ceramic Resonator Configuration

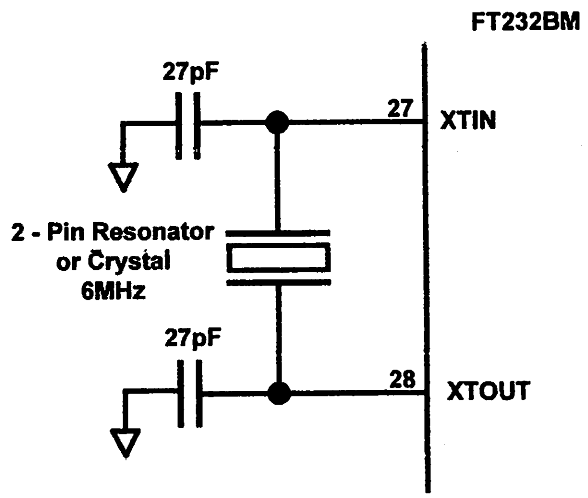


Figure 5 illustrates how to use the FT232BM with a 6MHz Crystal or 2-Pin Ceramic Resonator. In this case, these devices do not have in-built loading capacitors so these have to be added between XTIN, XTOUT and GND as shown. A value of 27pF is shown as the capacitor in the example – this will be good for many crystals and some resonators but do select the value based on the manufacturers recommendations wherever possible. If using a crystal, use a parallel cut type. If using a resonator, see the previous note on frequency accuracy.

:PROM Configuration

;
ı Configuration

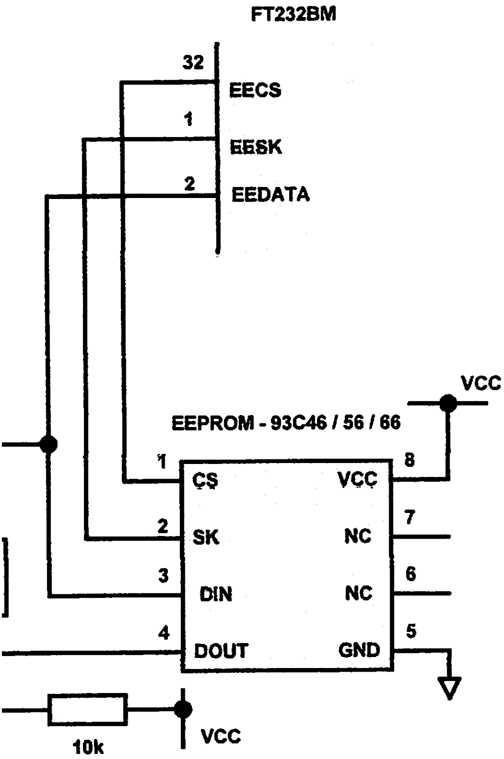


Figure 6 illustrates how to connect the FT232BM to the 93C46 (93C56 or 93C66) EEPROM. EECS (pin 32) is directly connected to the chip select (CS) pin of the EEPROM. EESK (pin 1) is directly connected to the clock (SK) pin of the EEPROM. EEDATA (pin 2) is directly connected to the Data In (Din) pin of the EEPROM. There is a potential condition whereby both the Data Output (Dout) of the EEPROM can drive out at the same time as the EEDATA pin of the FT232BM. To prevent potential data clash in this situation, the Dout of the EEPROM is connected to EEDATA of the FT232BM via a 2k2 resistor.

Following a power-on reset or a USB reset, the FT232BM will scan the EEPROM to find out a) if an EEPROM is attached to the Device and b) if the data in the device is valid. If both of these are the case, then the FT232BM will use the data in the EEPROM, otherwise it will use it's built-in default values. When a valid command is issued to the EEPROM from the FT232BM, the EEPROM will acknowledge the command by pulling it's Dout pin low. In order to check for this condition, it ;ary to pull Dout high using a 10k resistor. If the command acknowledge doesn't happen then EEDATA will high by the 10k resistor during this part of the cycle and the device will detect an invalid command or no 1 present.

: two varieties of these EEPROMs on the market – one is configured as being 16 bits wide, the other ıred as being 8 bits wide. These are available from many sources such as Microchip, ST, SIS etc. The 1 requires EEPROMs with a 16-bit wide configuration such as the Microchip 93LC46B device. The EEPROM capable of reading data at a 1Mb clock rate at a supply voltage of 4.4v to 5.25v. Most available parts are of this.

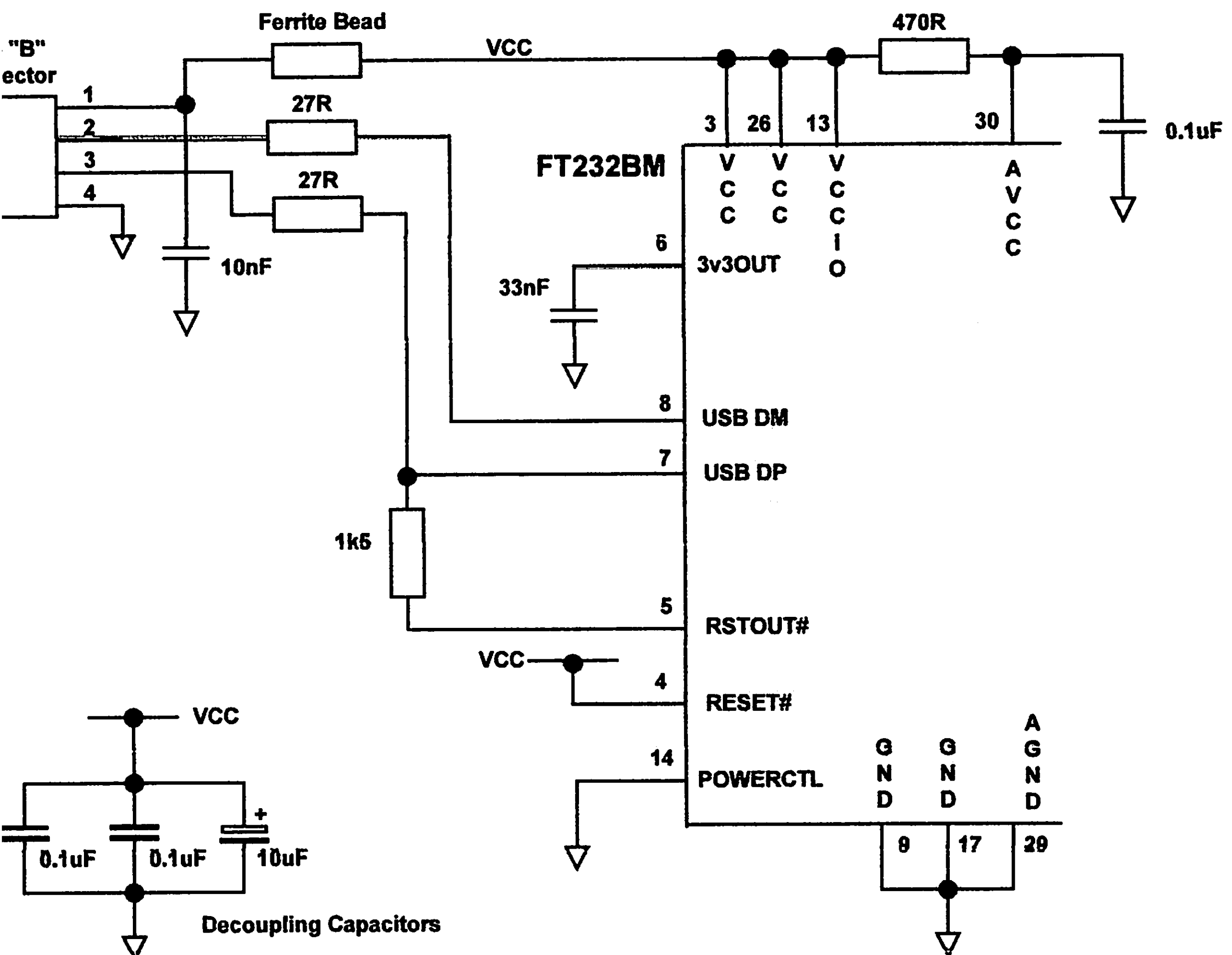
e manufacturers data sheet to find out how to connect pins 6 and 7 of the EEPROM. Some devices specify no-connect, others use them for selecting 8 / 16 bit mode or for test functions. Some other parts have their tated by 90° so please select the required part and it's options carefully.

ible to "share" the EEPROM between the FT232BM and another external device such as an MCU. However, only be done when the FT232BM is in it's reset condition as it tri-states it's EEPROM interface at that time. configuration would use four bit's of an MCU IO Port. One bit would be used to hold the FT232BM reset (:SET#) on power-up, the other three would connect to the EECS, EESK and EEDATA pins of the FT232BM o read / write data to the EEPROM at this time. Once the MCU has read / written the EEPROM, it would take high to allow the FT232BM to configure itself and enumerate over USB.

ɜB Bus Powered and Self Powered Configuration

7

ıs Powered Configuration



illustrates a typical USB bus powered configuration. A USB Bus Powered device gets its power from the USB ic rules for USB Bus power devices are as follows –

lug-in, the device must draw no more than 100mA

ISB Suspend the device must draw no more than 500uA.

s Powered High Power Device (one that draws more than 100mA) should use the PWREN# pin to keep the ɪnt below 100mA on plug-in and 500uA on USB suspend.

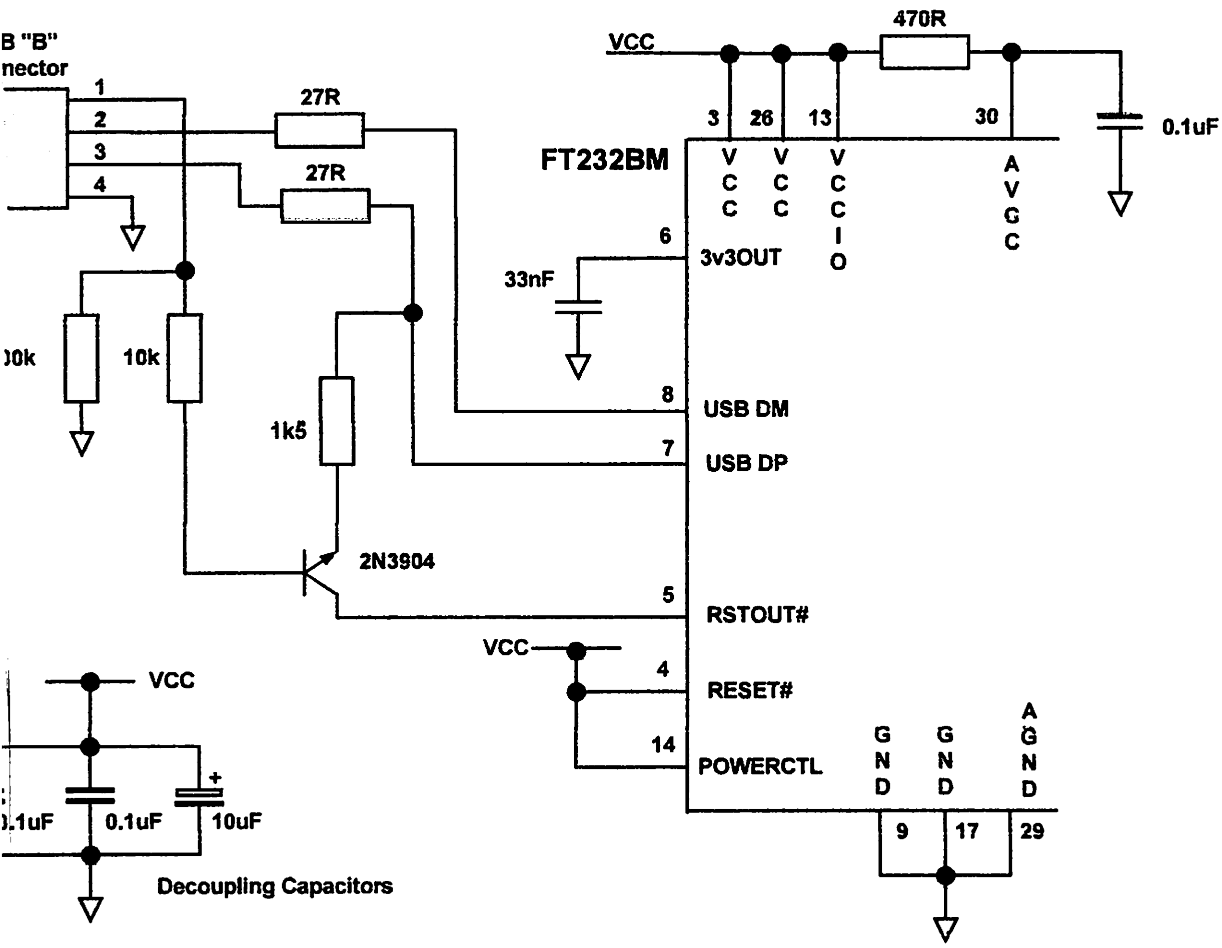
vice that consumes more than 100mA can not be plugged into a USB Bus Powered Hub

evice can draw more that 500mA from the USB Bus.

CTL (pin 14) is pulled low to tell the device to use a USB Bus Power descriptor. The power descriptor in the Ʌ should be programmed to match the current draw of the device.

Bead is connected in series with USB power to prevent noise from the device and associated circuitry (EMI ɜdiated down the USB cable to the Host. The value of the Ferrite Bead depends on the total current required rcuit – a suitable range of Ferrite Beads is available from Steward (www.steward.com) for example Steward **I0805K400R-00 also available as DigiKey Part #** 240-1035-1.

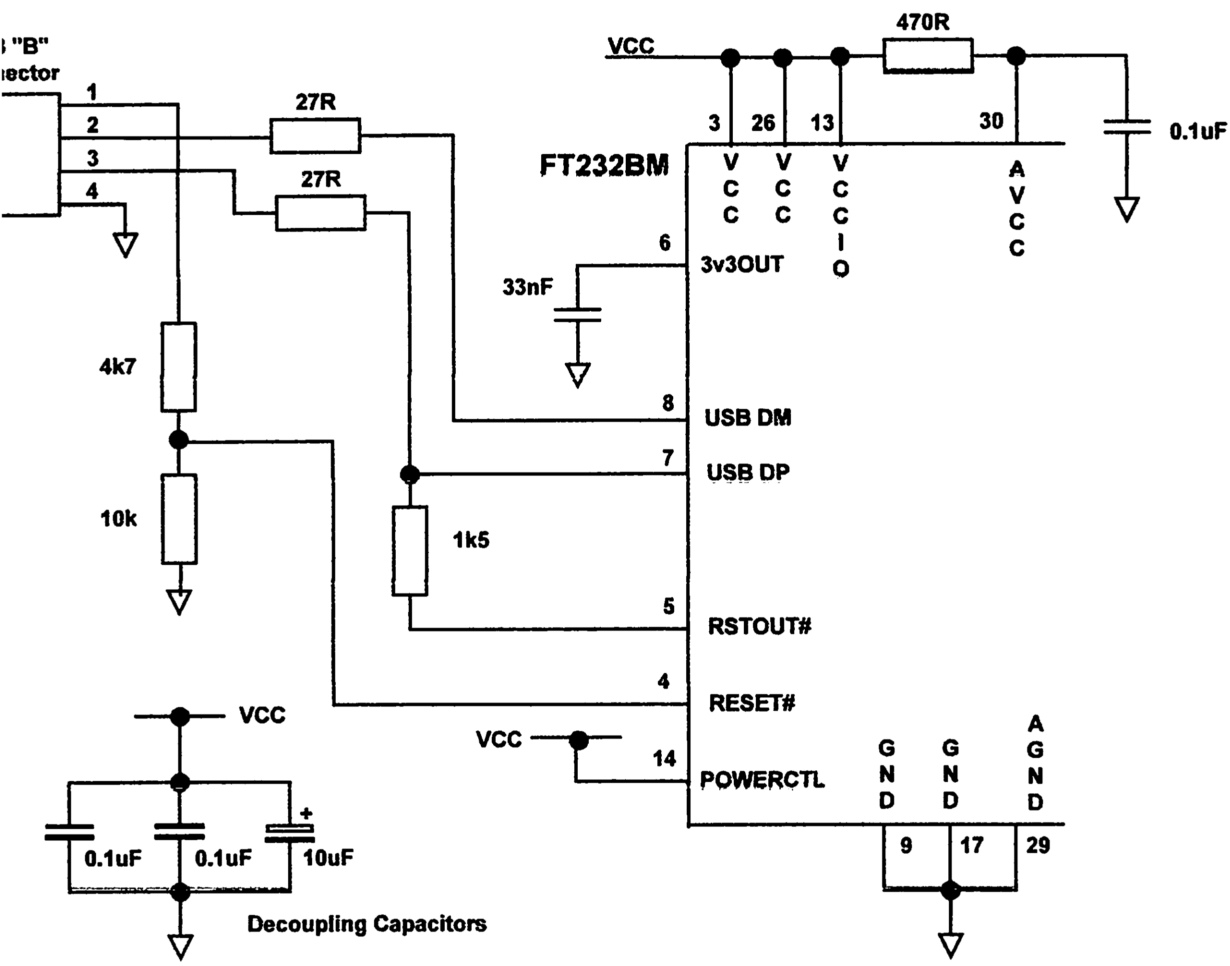
8
elf Powered Configuration (1)



; illustrates a typical USB self powered configuration. A USB Self Powered device gets its power from its own
{ SUPPLY and does not draw current from the USB bus. Basic rules for USB Self power devices are as follows

elf-Powered device should not force current down the USB bus when the USB Host or Hub Controller is
'ered down.
elf Powered Device can take as much current as it likes during normal operation and USB suspend as it has its
ı POWER SUPPLY.
elf Powered Device can be used with any USB Host and both Bus and Self Powered USB Hubs
{CTL (pin 14) is pulled high to tell the device to use a USB Bus Power descriptor. The power descriptor in the
M should be programmed to a value of zero.
: requirement a), the 1k5 pull-up circuit on USB DP has to be modified to prevent the device forcing current into
} DP line via the 1k5 pull-up when the host or hub is powered down. Failure to do this may cause some USB
ıub controllers to power up erratically. A NPN small signal transistor (2N3906) is used to sense the power on
} bus. It is connected as an emitter-follower circuit so that when there is power on the USB bus the transistor
rate and pull the 1k5 resistor to the voltage of RSTOUT#. When the USB power is off, the transistor will turn
preventing current flow into the USB DP line.

9
ɘlf Powered Configuration (2)



illustrates a variant of the circuit shown in Figure 8. This time, the 1k5 pull-up resistor on USB DP is ıd to RSTOUT# as per the bus-power circuit. However, the USB Bus Power is used to control the RESET# ɘ FT232BM device. When the USB Host or Hub power is off, RESET# will go low and the device will be held As RESET# is low, RSTOUT# will also be low, so no current will be forced down USB DP via the 1k5 pull-up

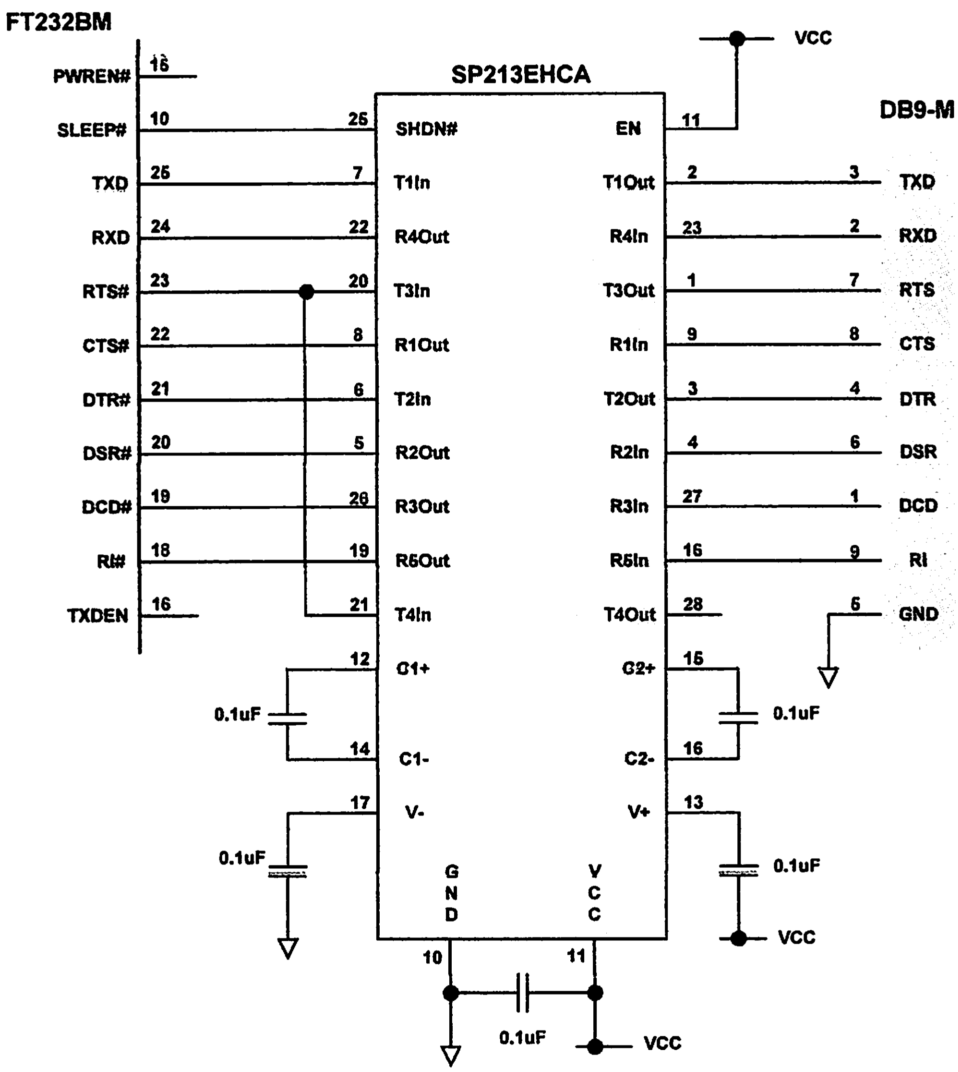
'hen the FT232B is in reset, the UART interface pins all go tri-state. These pins have internal 200k pull-up to VCC-IO so they will gently pull high unless driven by some external logic.

the two configurations to use depends on the nature of the peripheral design. With the configuration of the FT232BM is "live" – when power to the USB port is shut off, there will be no activity on the USB bus ɘevice will enter low power sleep mode after a few milliseconds. In this configuration, the RESET# pin is still ı if required.

ȷure 9 configuration, the FT232BM is held in reset when the USB power is off. In reset, the FT232BM 6MHz will still be running and the device will not be in low power mode.

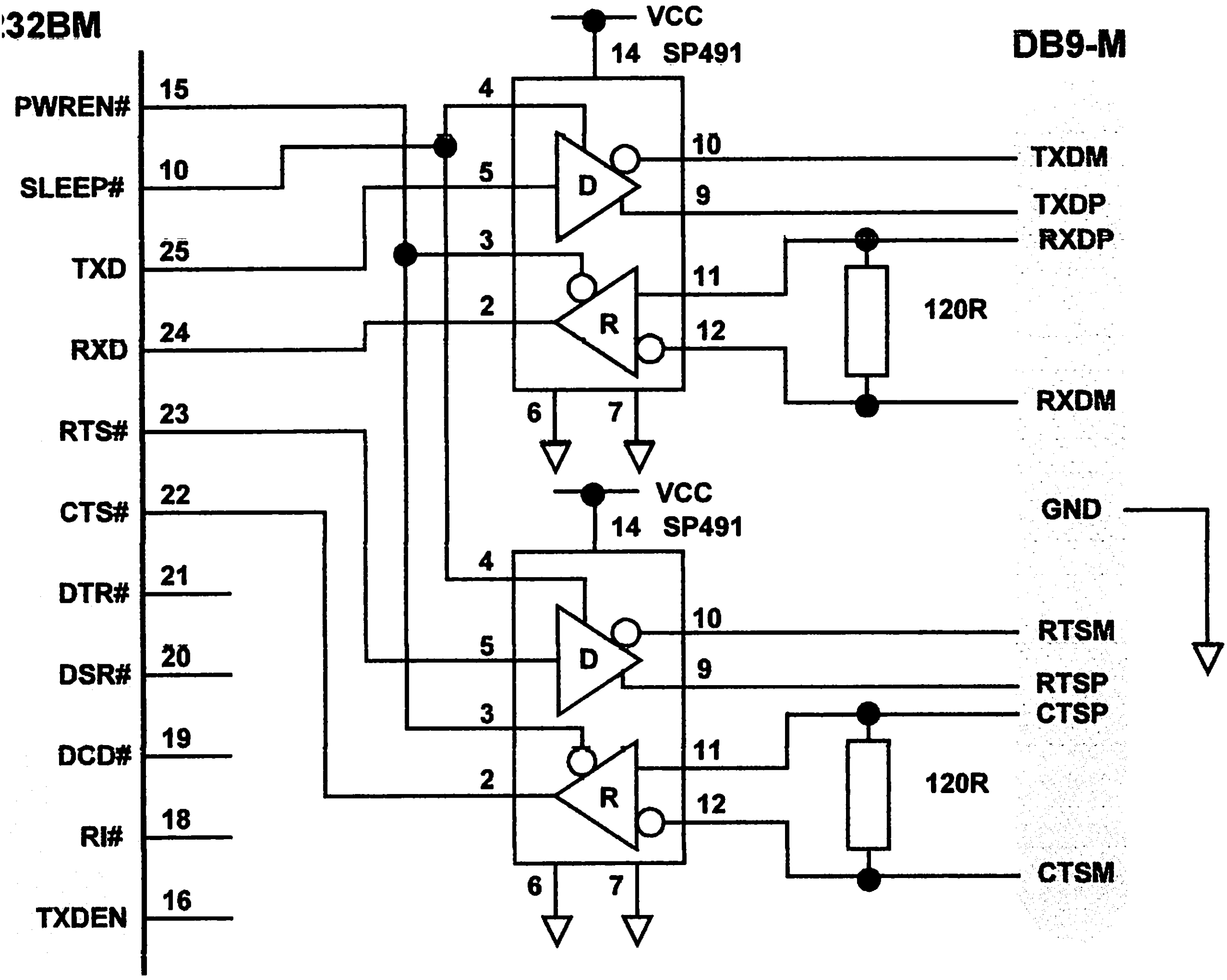
ART Interface Configuration

10
· RS232 Converter Configuration



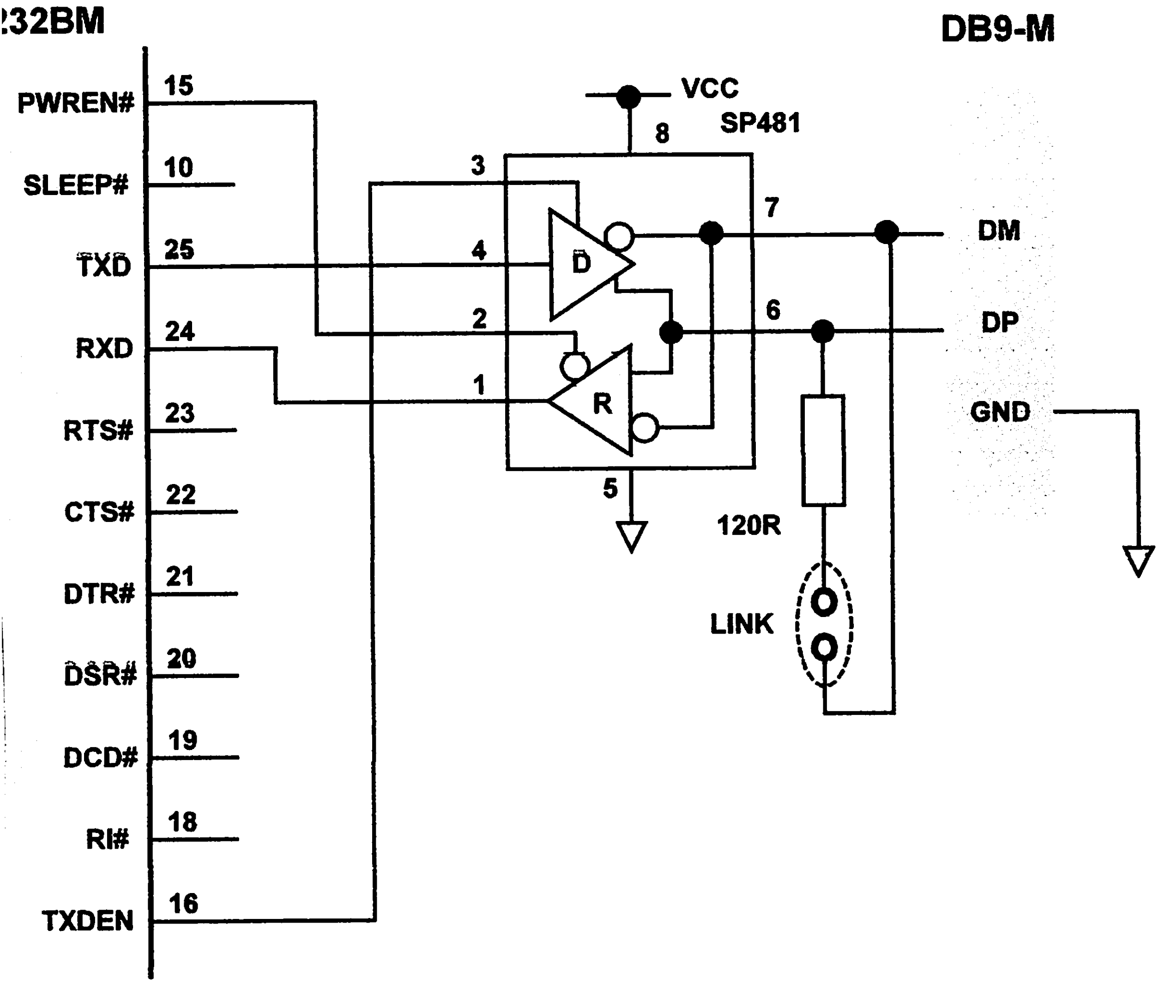
) illustrates how to connect the UART interface of the FT232BM to a TTL – RS232 Level Converter I.C. to JSB -> RS232 converter using the popular "213" series of TTL to RS232 level converters. These devices ansmitters and 5 receivers in a 28 LD SSOP package and feature an in-built voltage converter to convert ıominal) VCC to the +/- 9volts required by RS232. An important feature of these devices is the SHDN# pin n power down the device to a low quiescent current during USB suspend mode
ce used in the example is a Sipex SP213EHCA which is capable of RS232 communication at up to 500k ı lower baud rate is acceptable, then several pin compatible alternatives are available such as Sipex ЭA , Maxim MAX213CAI and Analog Devices ADM213E which are good for communication at up to 115,200 ı higher baud rate is desired, use a Maxim MAX3245CAI part which is capable of RS232 communication at ıp to 1M baud. The MAX3245 is not pin compatible with the 213 series devices, also it's SHDN pin is active ;onnect this to PWREN# instead of SLEEP#.

11
· RS422 Converter Configuration



l illustrates how to connect the UART interface of the FT232BM to a TTL – RS422 Level Converter I.C. to
JSB -> RS422 converter. There are many such level converter devices available – this example uses Sipex
evices which have enables on both the transmitter and receiver. Because the transmitter enable is active
connected to the SLEEP# pin. The receiver enable is active low and is connected to the PWREN# pin. This
that both the transmitters and receivers are enabled when the device is active, and disabled when the device
} suspend mode. If the design is USB BUS powered, it may be necessary to use a P-Channel logic level
Γ (controlled by PWREN#) in the VCC line of the SP491 devices to ensure that the USB standby current of
met.
91 is good for sending and receiving data at a rate of up to 5M Baud – in this case the maximum rate is
ı 3M Baud by the FT232BM.

12
▸ RS485 Converter Configuration



2 illustrates how to connect the UART interface of the FT232BM to a TTL – RS485 Level Converter I.C. to USB => RS485 converter. This example uses the Sipex SP491 device but there are similar parts available xim and Analog Devices amongst others. The SP491 is a RS485 device in a compact 8 pin SOP package. parate enables on both the transmitter and receiver. With RS485, the transmitter is only enabled when a r is being transmitted from the UART. The TXDEN pin on the FT232BM is provided for exactly that purpose he transmitter enable is wired to TXDEN. The receiver enable is active low, so it is wired to the PWREN# pin e the receiver when in USB suspend mode.

s a multi-drop network – i.e. many devices can communicate with each other over a single two wire cable ion. The RS485 cable requires to be terminated at each end of the cable. A link is provided to allow the cable minated if the device is physically positioned at either end of the cable.

xample the data transmitted by the FT232BM is also received by the device that is transmitting. This is a ı feature of RS485 and requires the application software to remove the transmitted data from the received eam. With the FT232BM it is possible to do this entirely in hardware – simply modify the schematic so that the FT232BM is the logical OR of the SP481 receiver output with TXDEN using an HC32 or similar logic gate.

) Interface

13
ED Configuration

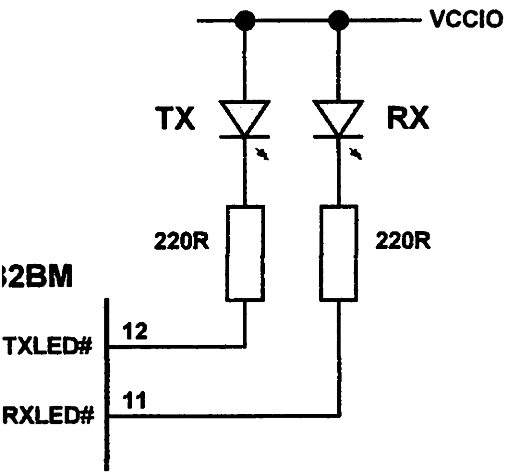
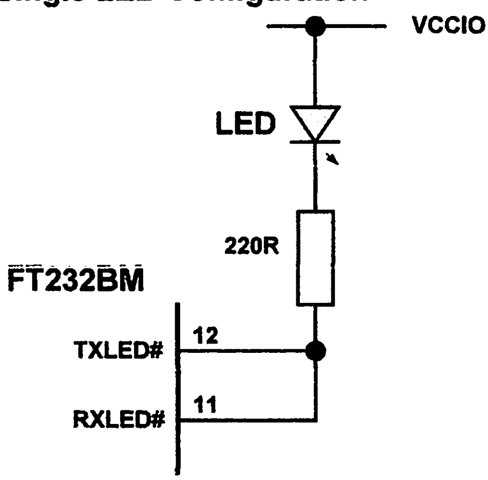


Figure 14
Single LED Configuration

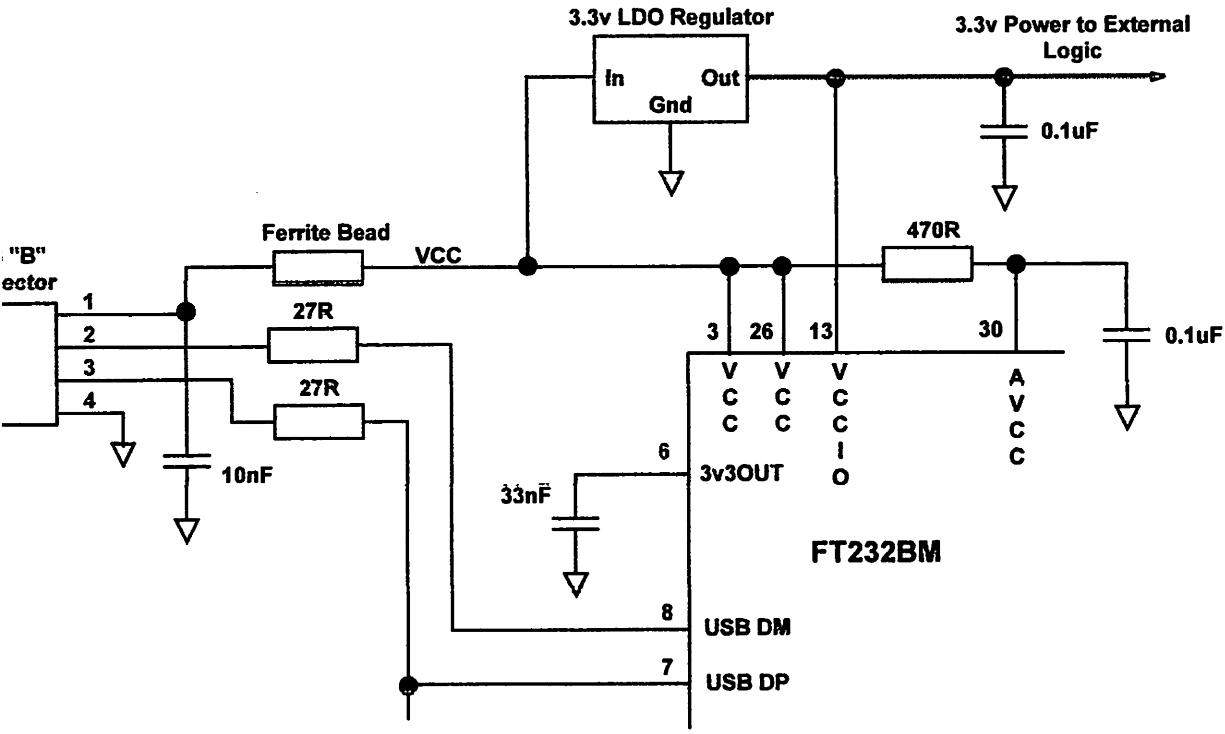


32BM has two IO pins dedicated to controlling LED status indicators, one for transmitted data the other 'ed data. When data is being transmitted / received the respective pins drive from tri-state to low in order e indication on the LEDs of data transfer. A digital one-shot timer is used so that even a small percentage ansfer is visible to the end user. Figure 13 shows a configuration using two individual LED's – one for ed data the other for received data. In Figure 14, the transmit and receive LED indicators are wire-or'd to give a single LED indicator which indicates any transmit or receive data activity.
ossibility (not shown here) is to use a 3 pin common anode tri-color LED based on the circuit in Figure 13 single LED that can display activity in a variety of colors depending on the ratio of transmit activity compared e activity.

rfacing to 3.3v Logic

15

wered Circuit with 3.3v logic drive / supply voltage



; shows how to configure the FT232BM to interface with a 3.3v logic device. In this example, a discrete 3.3v is used to supply the 3.3v logic from the USB supply. VCCIO is connected to the output of the 3.3v regulator, turn will cause the UART interface IO pins to drive out at 3.3v level. For USB bus powered circuits some ations have to be taken into account when selecting the regulator –

'egulator must be capable of sustaining its output voltage with an input voltage of 4.4 volts. A Low Drop Out ( ) regulator must be selected.

quiescent current of the regulator must be low in order to meet the USB suspend total current requirement of)0uA during USB suspend.

ple of a regulator family that meets these requirements is the MicroChip (Telcom) TC55 Series. These can supply up to 250mA current and have a quiescent current of under 1uA.

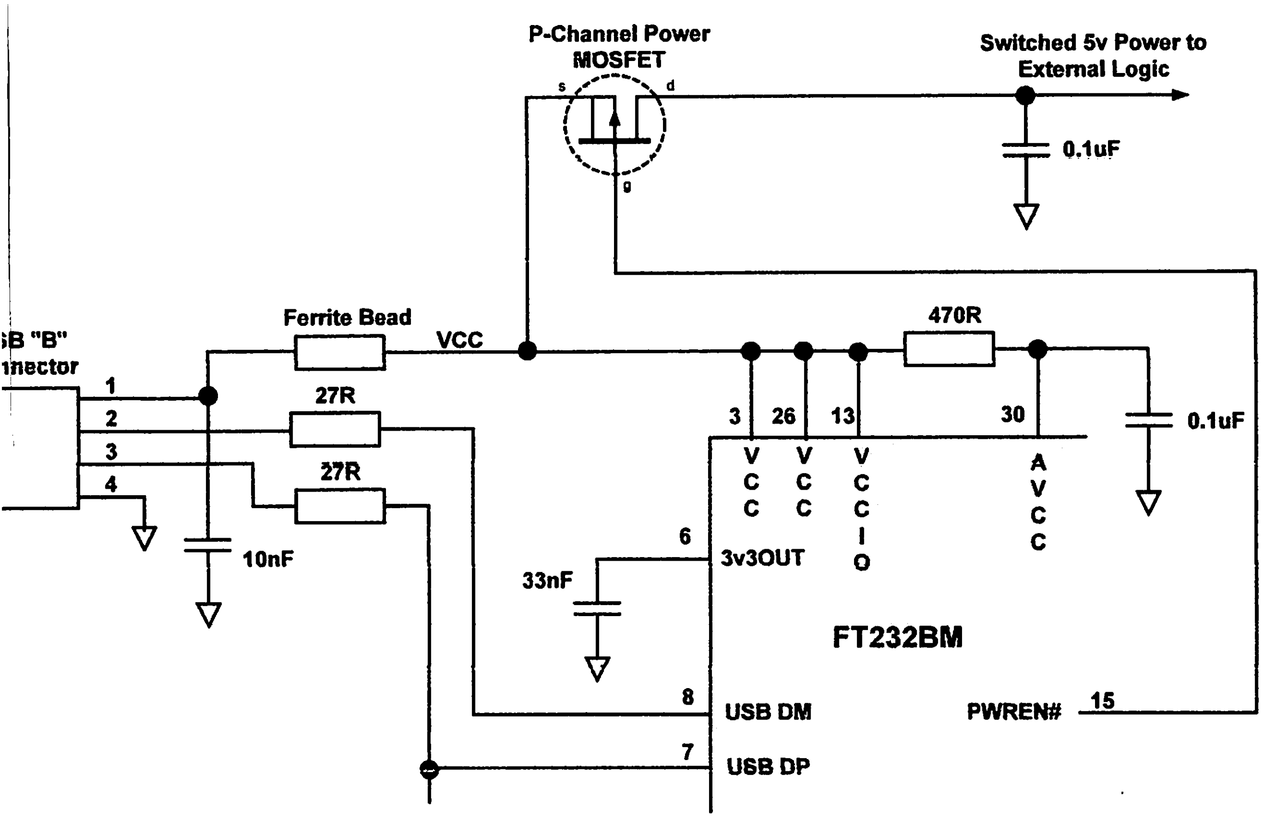
ing the FT232BM in a self powered USB design, simply connect VCCIO to the 3.3v supply rail of the external :. Suspend current is not a consideration for self powered designs.

cases, where only a small amount of current is required (< 5mA) , it may be possible to use the in-built of the FT232BM to supply the 3.3v without any other components being required. In this case, connect) the 3v3OUT pin of the FT232BM.

›wer Switching

ͅ 16

owered Circuit (<= 100mA) with Power Control



ıs powered circuits need to be able to power down in USB suspend mode in order to meet the <= 500uA total d current requirement (including external logic). Some external logic can power itself down into a low current / monitoring the POWEREN# pin. For external logic that cannot power itself down in that way, the FT232BM s a simple but effective way of turning off power to external circuitry during USB suspend.

16 shows how to use a discrete P-Channel Logic Level MOSFET to control the power to external logic circuits. ɔle device could be a Fairchild NDT456P or equivalent. This configuration is suitable for powering external logic :he normal supply current is <= 100mA and the logic to be controlled does not generate an appreciable current ıt power-up. For power switching external logic that takes over 100mA or generates a current surge on power- ecommend that a dedicated power switch i.c with inbuilt “soft-start” is used instead of a MOSFET. A suitable ͅwitch i.c. for such an application would be a Micrel (www.micrel.com) MIC2025-2BM or equivalent.

note the following points in connection with power controlled designs –

ͅ logic to be controlled must have it's own reset circuitry so that it will automatically reset itself when power is ɜpplied on coming out of suspend.

: the soft pull-down option bit in the FT232BM EEPROM.

· 3.3v power controlled circuits VCCIO must not be powered down with the external circuitry (PWREN# gets VCC supply from VCCIO). Either connect the power switch between the output of the 3.3v regulator and the ɜrnal 3.3v logic OR if appropriate power VCCIO from the 3v3OUT pin of the FT232BM.

cument Revision History

3 Version 1.0 – Initial document created 30 April 2002.
3 Version 1.1 – Updated 04 August 2002
SET# Pin description corrected (RESET# does not have an internal 200k pull-up to VCC as previously ed).
re 2 pin-out corrected (EECS = Pin 32).

sclaimer

duct and its documentation are supplied on an as-is basis and no warranty as to their suitability for ticular purpose is either made or implied.
Technology Devices International Ltd. will not accept any claim for damages howsoever arising as a f use or failure of this product. Your statutory rights are not affected.

duct or any variant of it is not intended for use in any medical appliance, device or system in which re of the product might reasonably be expected to result in personal injury.

cument provides preliminary information that may be subject to change without notice.

ontact Information

Technology Devices Intl. Limited
rge's Studios
t. George's Road,
v G3 6JA,
Kingdom.
4 (0)141 353 2565
14 (0)141 353 2656

E-Mail (Sales) : sales@ftdichip.com
E-Mail (Support) : support@ftdichip.com
E-Mail (General Enquiries) : admin@ftdichip.com
Web Site URL : http://www.ftdichip.com

and Sales Representatives

me of writing our Sales Network covers over 40 different countries world-wide. Please visit the Sales Network our Web Site site for the contact details our distributor(s) in your country.

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

FORMULIR BIMBINGAN SKRIPSI

Nama : Eddy Nur Imamsyah
Nim : 03.17.018
Masa Bimbingan : 03-November-2007 s/d 03-Mei-2008
Judul Skripsi : Perancangan Dan Pembuatan Alat Daftar Kir Kendaraan Berbasis Smartcard Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C13/TINY

No	Tanggal	Uraian	Paraf Pembimbing
1	12/12 07	Bab I → diperbaiki Bab II → acuan diperbaiki	[signature]
2	9/1 08	Bab III → revisi	[signature]
3	5/2 08	Perbaiki Bab IV & V	[signature]
4	11/2 08	Bab IV & V	[signature]
5	18/2 08	makalah seminar.	[signature]
6	6/3 08	Acc Akhir	[signature]
7			
8			
9			
10			

Malang,

Dosen pembimbing I

[signature]

Ir. F. Yudi Limpraptono, MT
Nip. 1039500274

Form S-4a

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

FORMULIR BIMBINGAN SKRIPSI

Nama : Eddy Nur Imamsyah
Nim : 03.17.018
Masa Bimbingan : 03-November-2007 s/d 03-Mei-2008
Judul Skripsi : Perancangan Dan Pembuatan Alat Daftar Kir Kendaraan Berbasis Smartcard Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C13/TINY

No	Tanggal	Uraian	Paraf Pembimbing
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Malang,
Dosen pembimbing I

Ir. F. Yudi Limpraptono, MT
Nip. 1039500274

Form S-4a

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

FORMULIR BIMBINGAN SKRIPSI

Nama : Eddy Nur Imamsyah
Nim : 03.17.018
Masa Bimbingan : 03-November-2007 s/d 03-Mei-2008
Judul Skripsi : Perancangan Dan Pembuatan Alat Daftar Kir Kendaraan Berbasis Smartcard Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C13/TINY

No	Tanggal	Uraian	Paraf Pembimbing
1	12/12/07	Bab I dan II Revisi penulisan.	[paraf]
2	3/01/08	Acc Bab I dan II Bab III Revisi flowchart.	[paraf]
3	7/02/08	Acc Bab III Bab IV Revisi Rangkaian dan Gambar	[paraf]
4	15/02/08	Acc Bab IV Bab V Revisi kesimpulan	[paraf]
5	19/02/08	Acc Bab V.	[paraf]
6	20/02/08	Acc Bab I, II, III, IV dan V.	[paraf]
7	27/02/08	Acc Seminar hasil.	[paraf]
8	04/03/08	Acc Ujian kompre. dan laporan Skripsi	[paraf]
9			
10			

Malang, 4 maret 2008
Dosen pembimbing II

[tanda tangan]

Joseph Dedy Irawan ST, MT
NIP. 132315178

Form S-4a 599

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

FORMULIR BIMBINGAN SKRIPSI

Nama : Eddy Nur Imamsyah
Nim : 03.17.018
Masa Bimbingan : 03-November-2007 s/d 03-Mei-2008
Judul Skripsi : Perancangan Dan Pembuatan Alat Daftar Kir Kendaraan Berbasis Smartcard Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C13/TINY

No	Tanggal	Uraian	Paraf Pembimbing
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Malang

Dosen pembimbing II

Joseph Dedy Irawan ST, MT
NIP. 132315178

Form S-4a

INSTITUT TEKNOLOGI NASIONAL
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
Jl. Karanglo KM. 2 Malang

FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan Ujian Skripsi Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama	: Eddy Nur Imamsyah
NIM	: 03 17 018
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Elektronika
Masa Bimbingan	: 3 November 2007 s/d 3 Mei 2008
Judul Skripsi	: Perencanaan dan Pembuatan Alat Daftar KIR Kendaraan Bermotor Berbasis Smartcard Dengan Menggunakan Modul FT232BM-USB Yang Dikendalikan Oleh R8C/13 Tiny.

Penguji/Tanggal	Uraian	Paraf
Penguji II 15 Maret 2008	Abstraksi meliputi Latar Blakang,Tujuan,Metodologi, Hasil	
	Hal 9 Gambar di edit ,hal 24 tabel diatas	
	Ditambah Flowchart baca/tulis kartu	
	Perancangan database ditambah meliputi table dan menghubungkannya,dst	

Dosen Penguji,

Penguji II

I Komang Somawirata,ST, MT
NIP.P.1030100361

Mengetahui,

Dosen Pembimbing I

Ir. F. Yudi Limpraptono, MT
NIP. 1039500274

Dosen Pembimbing II

Joseph Dedy Irawan ST, MT
NIP.132315178

Applications Information

Capacitor Selection

The capacitor type used for C1–C4 is not critical for proper operation. The MAX202E, MAX206–MAX208E, MAX211E, and MAX213E require 0.1µF capacitors, and the MAX232E and MAX241E require 1µF capacitors, although in all cases capacitors up to 10µF can be used without harm. Ceramic, aluminum-electrolytic, or tantalum capacitors are suggested for the 1µF capacitors, and ceramic dielectrics are suggested for the 0.1µF capacitors. When using the minimum recommended capacitor values, make sure the capacitance value does not degrade excessively as the operating temperature varies. If in doubt, use capacitors with a larger (e.g., 2x) nominal value. The capacitors' effective series resistance (ESR), which usually rises at low temperatures, influences the amount of ripple on V+ and V-.

Use larger capacitors (up to 10µF) to reduce the output impedance at V+ and V-. This can be useful when "stealing" power from V+ or from V-. The MAX203E and MAX205E have internal charge-pump capacitors.

Bypass V_{CC} to ground with at least 0.1µF. In applications sensitive to power-supply noise generated by the charge pumps, decouple V_{CC} to ground with a capacitor the same size as (or larger than) the charge-pump capacitors (C1–C4).

V+ and V- as Power Supplies

A small amount of power can be drawn from V+ and V-, although this will reduce both driver output swing and noise margins. Increasing the value of the charge-pump capacitors (up to 10µF) helps maintain performance when power is drawn from V+ or V-.

Driving Multiple Receivers

Each transmitter is designed to drive a single receiver. Transmitters can be paralleled to drive multiple receivers.

Driver Outputs when Exiting Shutdown

The driver outputs display no ringing or undesirable transients as they come out of shutdown.

High Data Rates

These transceivers maintain the RS-232 ±5.0V minimum driver output voltages at data rates of over 120kbps. For data rates above 120kbps, refer to the Transmitter Output Voltage vs. Load Capacitance graphs in the *Typical Operating Characteristics*. Communication at these high rates is easier if the capacitive loads on the transmitters are small; i.e., short cables are best.

Table 2. Summary of EIA/TIA-232E, V.28 Specifications

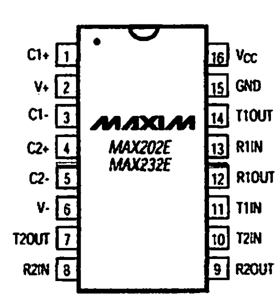
PARAMETER		CONDITIONS	EIA/TIA-232E, V.28 SPECIFICATIONS
Driver Output Voltage	0 Level	3kΩ to 7kΩ load	+5V to +15V
	1 Level	3kΩ to 7kΩ load	-5V to -15V
Driver Output Level, Max		No load	±25V
Data Rate		3kΩ ≤ R_L ≤ 7kΩ, C_L ≤ 2500pF	Up to 20kbps
Receiver Input Voltage	0 Level		+3V to +15V
	1 Level		-3V to -15V
Receiver Input Level			±25V
Instantaneous Slew Rate, Max		3kΩ ≤ R_L ≤ 7kΩ, C_L ≤ 2500pF	30V/µs
Driver Output Short-Circuit Current, Max			100mA
Transition Rate on Driver Output		V.28	1ms or 3% of the period
		EIA/TIA-232E	4% of the period
Driver Output Resistance		-2V < V_{OUT} < +2V	300Ω

Table 3. DB9 Cable Connections Commonly Used for EIA/TIAE-232E and V.24 Asynchronous Interfaces

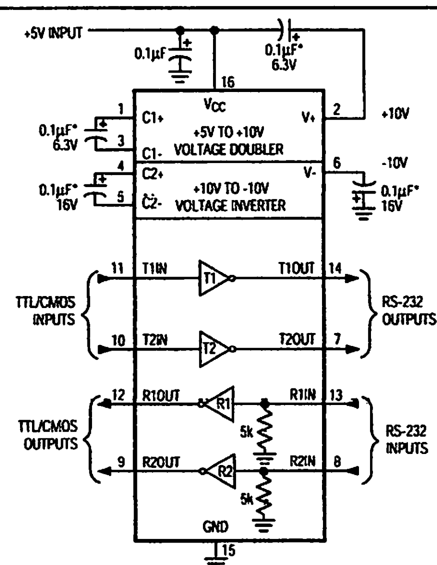
PIN	CONNECTION	
1	Received Line Signal Detector (sometimes called Carrier Detect, DCD)	Handshake from DCE
2	Receive Data (RD)	Data from DCE
3	Transmit Data (TD)	Data from DTE
4	Data Terminal Ready	Handshake from DTE
5	Signal Ground	Reference point for signals
6	Data Set Ready (DSR)	Handshake from DCE
7	Request to Send (RTS)	Handshake from DTE
8	Clear to Send (CTS)	Handshake from DCE
9	Ring Indicator	Handshake from DCE

Pin Configurations and Typical Operating Circuits (continued)

TOP VIEW



DIP/SO



PIN NUMBERS ON TYPICAL OPERATING CIRCUIT REFER TO DIP/SO PACKAGE, NOT LCC.
* 1.0μF CAPACITORS, MAX232E ONLY.

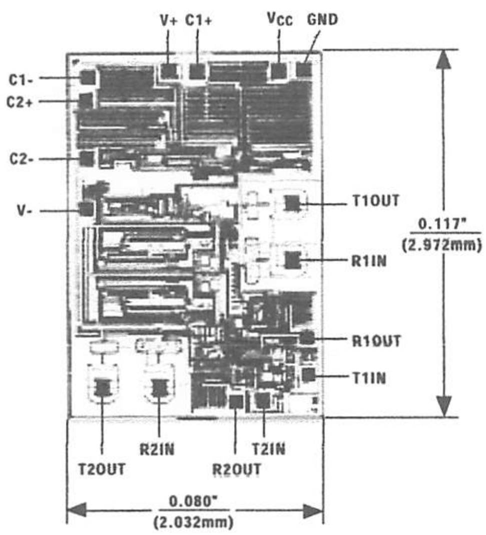
Ordering Information

PART	TEMP. RANGE	PIN-PACKAGE
MAX202ECPE	0°C to +70°C	16 Plastic DIP
MAX202ECSE	0°C to +70°C	16 Narrow SO
MAX202ECWE	0°C to +70°C	16 Wide SO
MAX202EC/D	0°C to +70°C	Dice*
MAX202EEPE	-40°C to +85°C	16 Plastic DIP
MAX202EESE	-40°C to +85°C	16 Narrow SO
MAX202EEWE	-40°C to +85°C	16 Wide SO
MAX203ECPP	0°C to +70°C	20 Plastic DIP
MAX203ECWP	0°C to +70°C	20 SO
MAX203EEPP	-40°C to +85°C	20 Plastic DIP
MAX203EEWP	-40°C to +85°C	20 SO
MAX205ECPG	0°C to +70°C	24 Wide Plastic DIP
MAX205EEPG	-40°C to +85°C	24 Wide Plastic DIP
MAX206ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX206ECWG	0°C to +70°C	24 SO
MAX206ECAG	0°C to +70°C	24 SSOP
MAX206EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX206EEWG	-40°C to +85°C	24 SO
MAX206EEAG	-40°C to +85°C	24 SSOP
MAX207ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX207ECWG	0°C to +70°C	24 SO
MAX207ECAG	0°C to +70°C	24 SSOP
MAX207EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX207EEWG	-40°C to +85°C	24 SO
MAX207EEAG	-40°C to +85°C	24 SSOP
MAX208ECNG	0°C to +70°C	24 Narrow Plastic DIP
MAX208ECWG	0°C to +70°C	24 SO
MAX208ECAG	0°C to +70°C	24 SSOP
MAX208EENG	-40°C to +85°C	24 Narrow Plastic DIP
MAX208EEWG	-40°C to +85°C	24 SO
MAX208EEAG	-40°C to +85°C	24 SSOP
MAX211ECWI	0°C to +70°C	28 SO
MAX211ECAI	0°C to +70°C	28 SSOP
MAX211EEWI	-40°C to +85°C	28 SO
MAX211EEAI	-40°C to +85°C	28 SSOP
MAX213ECWI	0°C to +70°C	28 SO
MAX213ECAI	0°C to +70°C	28 SSOP
MAX213EEWI	-40°C to +85°C	28 SO
MAX213EEAI	-40°C to +85°C	28 SSOP
MAX232ECPE	0°C to +70°C	16 Plastic DIP
MAX232ECSE	0°C to +70°C	16 Narrow SO
MAX232ECWE	0°C to +70°C	16 Wide SO
MAX232EC/D	0°C to +70°C	Dice*
MAX232EEPE	-40°C to +85°C	16 Plastic DIP
MAX232EESE	-40°C to +85°C	16 Narrow SO
MAX232EEWE	-40°C to +85°C	16 Wide SO
MAX241ECWI	0°C to +70°C	28 SO
MAX241ECAI	0°C to +70°C	28 SSOP
MAX241EEWI	-40°C to +85°C	28 SO
MAX241EEAI	-40°C to +85°C	28 SSOP

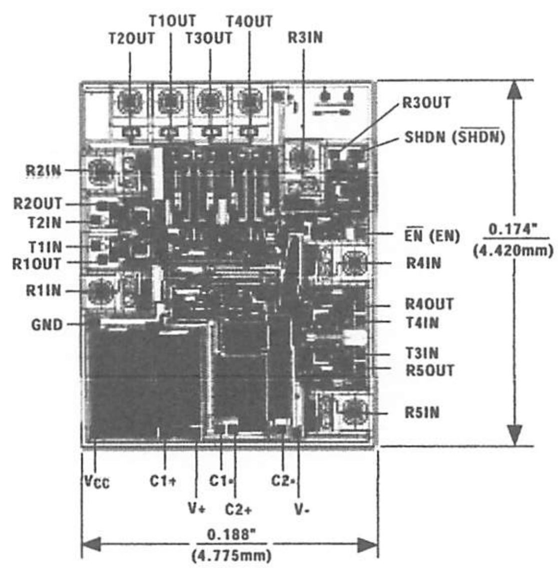
Dice are specified at T_A = +25°C.

Chip Topographies

MAX202E/MAX232E



MAX211E/MAX213E/MAX241E



( ) ARE FOR MAX213E ONLY

TRANSISTOR COUNT: 123
SUBSTRATE CONNECTED TO GND

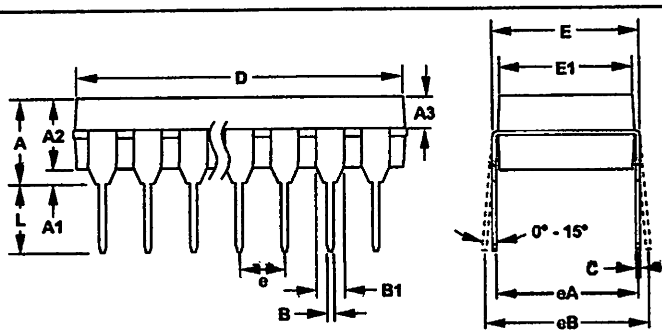
TRANSISTOR COUNT: 542
SUBSTRATE CONNECTED TO GND

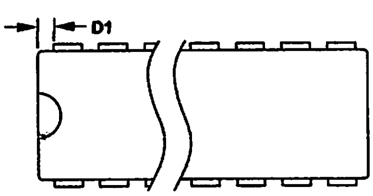
Chip Information

MAX205E/MAX206E/MAX207E/MAX208E

TRANSISTOR COUNT: 328
SUBSTRATE CONNECTED TO GND

Package Information



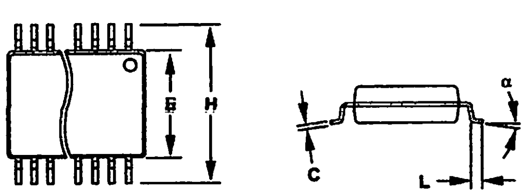


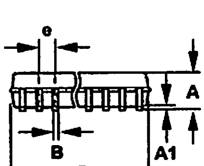
DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	–	0.200	–	5.08
A1	0.015	–	0.38	–
A2	0.125	0.175	3.18	4.45
A3	0.055	0.080	1.40	2.03
B	0.016	0.022	0.41	0.56
B1	0.045	0.065	1.14	1.65
C	0.008	0.012	0.20	0.30
D1	0.005	0.080	0.13	2.03
E	0.300	0.325	7.62	8.26
E1	0.240	0.310	6.10	7.87
e	0.100	–	2.54	–
eA	0.300	–	7.62	–
eB	–	0.400	–	10.16
L	0.115	0.150	2.92	3.81

Plastic DIP
PLASTIC DUAL-IN-LINE PACKAGE (0.300 in.)

PKG.	DIM	PINS	INCHES		MILLIMETERS	
			MIN	MAX	MIN	MAX
P	D	8	0.348	0.390	8.84	9.91
P	D	14	0.735	0.765	18.67	19.43
P	D	16	0.745	0.765	18.92	19.43
P	D	18	0.885	0.915	22.48	23.24
P	D	20	1.015	1.045	25.78	26.54
N	D	24	1.14	1.265	28.96	32.13

21-0043A





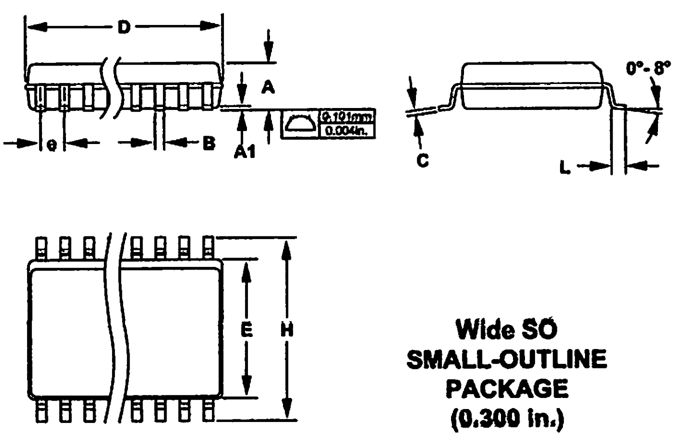
DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.068	0.078	1.73	1.99
A1	0.002	0.008	0.05	0.21
B	0.010	0.015	0.25	0.38
C	0.004	0.008	0.09	0.20
D	SEE VARIATIONS			
E	0.205	0.209	5.20	5.38
e	0.0256 BSC		0.65 BSC	
H	0.301	0.311	7.65	7.90
L	0.025	0.037	0.63	0.95
α	0°	8°	0°	8°

SSOP
SHRINK SMALL-OUTLINE PACKAGE

DIM	PINS	INCHES		MILLIMETERS	
		MIN	MAX	MIN	MAX
D	14	0.239	0.249	6.07	6.33
D	16	0.239	0.249	6.07	6.33
D	20	0.278	0.289	7.07	7.33
D	24	0.317	0.328	8.07	8.33
D	28	0.397	0.407	10.07	10.33

21-0056A

Package Information (continued)



Wide SO
SMALL-OUTLINE
PACKAGE
(0.300 in.)

DIM	INCHES		MILLIMETERS	
	MIN	MAX	MIN	MAX
A	0.093	0.104	2.35	2.65
A1	0.004	0.012	0.10	0.30
B	0.014	0.019	0.35	0.49
C	0.009	0.013	0.23	0.32
E	0.291	0.299	7.40	7.60
e	0.050		1.27	
H	0.394	0.419	10.00	10.65
L	0.016	0.050	0.40	1.27

DIM	PINS	INCHES		MILLIMETERS	
		MIN	MAX	MIN	MAX
D	16	0.398	0.413	10.10	10.50
D	18	0.447	0.463	11.35	11.75
D	20	0.496	0.512	12.60	13.00
D	24	0.598	0.614	15.20	15.60
D	28	0.697	0.713	17.70	18.10

21-0042A

Maxim cannot assume responsibility for use of any circuitry other than circuitry entirely embodied in a Maxim product. No circuit patent licenses are implied. Maxim reserves the right to change the circuitry and specifications without notice at any time.

FT232BM

232BM is the 2nd generation of FTDI's popular USB UART i.c. This device not only adds extra functionality T8U232AM predecessor and reduces external component count, but also maintains a high degree of pin ibility with the original, making it easy to upgrade or cost reduce existing designs as well as increasing the l for using the device in new application areas.

-eatures

VARE FEATURES

- ingle Chip USB ⇔Asynchronous Serial Data ransfer
- ull Handshaking & Modem Interface Signals
- ART I/F Supports 7 / 8 Bit Data, 1 / 2 Stop Bits nd Odd/Even/Mark/Space/No Parity
-)ata rate 300 => 3M Baud (TLL)
-)ata rate 300 => 1M Baud (RS232)
-)ata rate 300 => 3M Baud (RS422/RS485)
- 84 Byte Receive Buffer / 128 Byte Transmit Buffer or high data throughput
- djustable RX buffer timeout
- ull hardware assisted hardware or X-On / X-Off andshaking
- -built support for event characters and line break ondition
- uto Transmit Buffer control for RS485
- upport for USB Suspend / Resume through LEEP# and RI# pins
- upport for high power USB Bus powered devices rough PWREN# pin
- tegrated level converter on UART and control ignals for interfacing to 5v and 3.3v logic
- tegrated 3.3v regulator for USB IO
- tegrated Power-On-Reset circuit
- tegrated 6MHz – 48Mhz clock multiplier PLL
- SB Bulk or Isocronous data transfer modes
- .4v to 5.25v single supply operation
- HCI / OHCI / EHCI host controller compatible
- SB 1.1 and USB 2.0 compatible
- 3 VID, PID , Serial Number and Product cription strings in external EEPROM
- EPROM programmable on-board via USB

VIRTUAL COM PORT (VCP) DRIVERS for

- Windows 98 and Windows 98 SE
- Windows 2000 / ME / XP
- Windows CE **
- MAC OS-8 and OS-9
- MAC OS-X **
- Linux 2.40 and greater

D2XX (USB Direct Drivers + DLL S/W Interface)

- Windows 98 and Windows 98 SE
- Windows 2000 / ME / XP

APPLICATION AREAS

- USB ⇔RS232 Converters
- USB ⇔ RS422 / RS485 Converters
- Upgrading RS232 Legacy Peripherals to USB
- Cellular and Cordless Phone USB data transfer cables and interfaces
- Interfacing MCU based designs to USB
- USB Audio and Low Bandwidth Video data transfer
- PDA ⇔ USB data transfer
- USB Smart Card Readers
- Set Top Box (S.T.B.) PC - USB interface
- USB Hardware Modems
- USB Wireless Modems
- USB Instrumentation
- USB Bar Code Readers

[** = In planning or under development]

;nhancements

:tion summarises the enhancements of the 2nd generation device compared to it's FT8U232AM predecessor. ıer details, consult the device pin-out description and functional descriptions.

- **ıtegrated Power-On-Reset (POR) Circuit**
  ıe device now incorporates an internal POR nction. The existing RESET# pin is maintained order to allow external logic to reset the device here required, however for many applications ıis pin can now be either left N/C or hard wired to CC. In addition, a new reset output pin (RSTO# is provided in order to allow the new POR circuit ı provide a stable reset to external MCU and other evices. RSTO# was the TEST pin on the previous eneration of devices.

- **ıtegrated RCCLK Circuit**
  ı the previous devices, an external RC circuit 'as required to ensure that the oscillator and lock multiplier PLL frequency was stable prior ı enabling the clock internal to the device. This ircuit is now embedded on-chip – the pin assigned ı this function is now designated as the TEST pin nd should be tied to GND for normal operation.

- **ıtegrated Level Converter on UART interface nd control signals**
  'he previous devices would drive the UART and ontrol signals at 5v CMOS logic levels. The ew device has a separate VCC-IO pin allowing ıe device to directly interface to 3.3v and other ıgic families without the need for external level onverter i.c.'s

- **Improved Power Management control for USB Bus Powered, high current devices**
  The previous devices had a USBEN pin, which became active when the device was enumerated by USB. To provide power control, this signal had to be externally gated with SLEEP# and RESET#. This gating is now done on-chip - USBEN has now been replaced with the new PWREN# signal which can be used to directly drive a transistor or P-Channel MOSFET in applications where power switching of external circuitry is required. A new EEPROM based option makes the device pull gently down it's UART interface lines when the power is shut off (PWREN# is High). In this mode, any residual voltage on external circuitry is bled to GND when power is removed thus ensuring that external circuitry controlled by PWREN# resets reliably when power is restored.

- **Lower Suspend Current**
  Integration of RCCLK within the device and internal design improvements reduce the suspend current of the FT232BM to under 200uA (excluding the 1.5k pull-up on USB DP) in USB suspend mode. This allows greater margin for peripherals to meet the USB Suspend current limit of 500uA.

- **Support for USB Isocronous Transfers**
  Whilst USB Bulk transfer is usually the best choice for data transfer, the scheduling time of the data is not guaranteed. For applications where scheduling latency takes priority over data integrity such as transferring audio and low bandwidth video data, the new device now offers an option of USB Isocronous transfer via an option bit in the EEPROM.

rogrammable Receive Buffer Timeout
the previous device, the receive buffer timeout sed to flush remaining data from the receive uffer was fixed at 16ms timeout. This timeout is ow programmable over USB in 1ms increments om 1ms to 255ms, thus allowing the device to e better optimised for protocols requiring faster esponse times from short data packets.

XDEN Timing fix
XDEN timing has now been fixed to remove the xternal delay that was previously required for S485 applications at high baud rates. TXDEN ow works correctly during a transmit send-break ondition.

elaxed VCC Decoupling
he 2nd generation devices now incorporate a level f on-chip VCC decoupling. Though this does ot eliminate the need for external decoupling apacitors, it significantly improves the ease of pcb esign requirements to meet FCC,CE and other MI related specifications.

nproved PreScaler Granularity
he previous version of the Prescaler supported ivision by (n + 0), (n + 0.125), (n + 0.25) and n + 0.5) where n is an integer between 2 and 6,384 (2^{14}). To this we have added (n + 0.375 (n + 0.625), (n + 0.75) and (n+ 0.875) which an be used to improve the accuracy of some baud ates and generate new baud rates which were reviously impossible (especially with higher baud ates).

- **Bit Bang Mode**

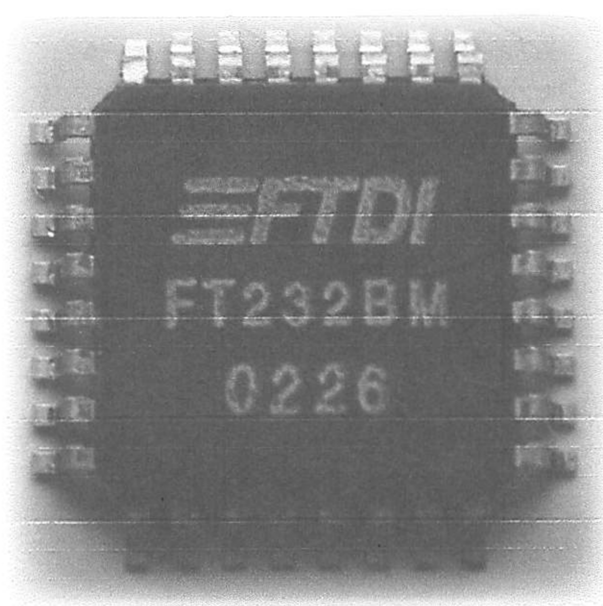
The 2nd generation device has a new option referred to as "Bit Bang" mode. In Bit Bang mode, the eight UART interface control lines can be switched between UART interface mode and an 8-bit Parallel IO port. Data packets can be sent to the device and they will be sequentially sent to the interface at a rate controlled by the prescaler setting. As well as allowing the device to be used stand-alone as a general purpose IO controller for example controlling lights, relays and switches, some other interesting possibilities exist. For instance, it may be possible to connect the device to an SRAM configurable FPGA as supplied by vendors such as Altera and Xilinx. The FPGA device would normally be un-configured (i.e. have no defined function) at power-up. Application software on the PC could use Bit Bang Mode to download configuration data to the FPGA which would define it's hardware function, then after the FPGA device is configured the FT232BM can switch back into UART interface mode to allow the programmed FPGA device to communicate with the PC over USB. This approach allows a customer to create a "generic" USB peripheral who's hardware function can be defined under control of the application software. The FPGA based hardware can be easily upgraded or totally changed simply by changing the FPGA configuration data file. Application notes, software and development modules for this application area will be available from FTDI and other 3rd parties.

reScaler Divide By 1 Fix
he previous device had a problem when the iteger part of the divisor was set to 1. In the 2nd eneration device setting the prescaler value to 1 ives a baud rate of 2 million baud and setting it o zero gives a baud rate of 3 million baud. Non-iteger division is not supported with divisor values f 0 and 1.

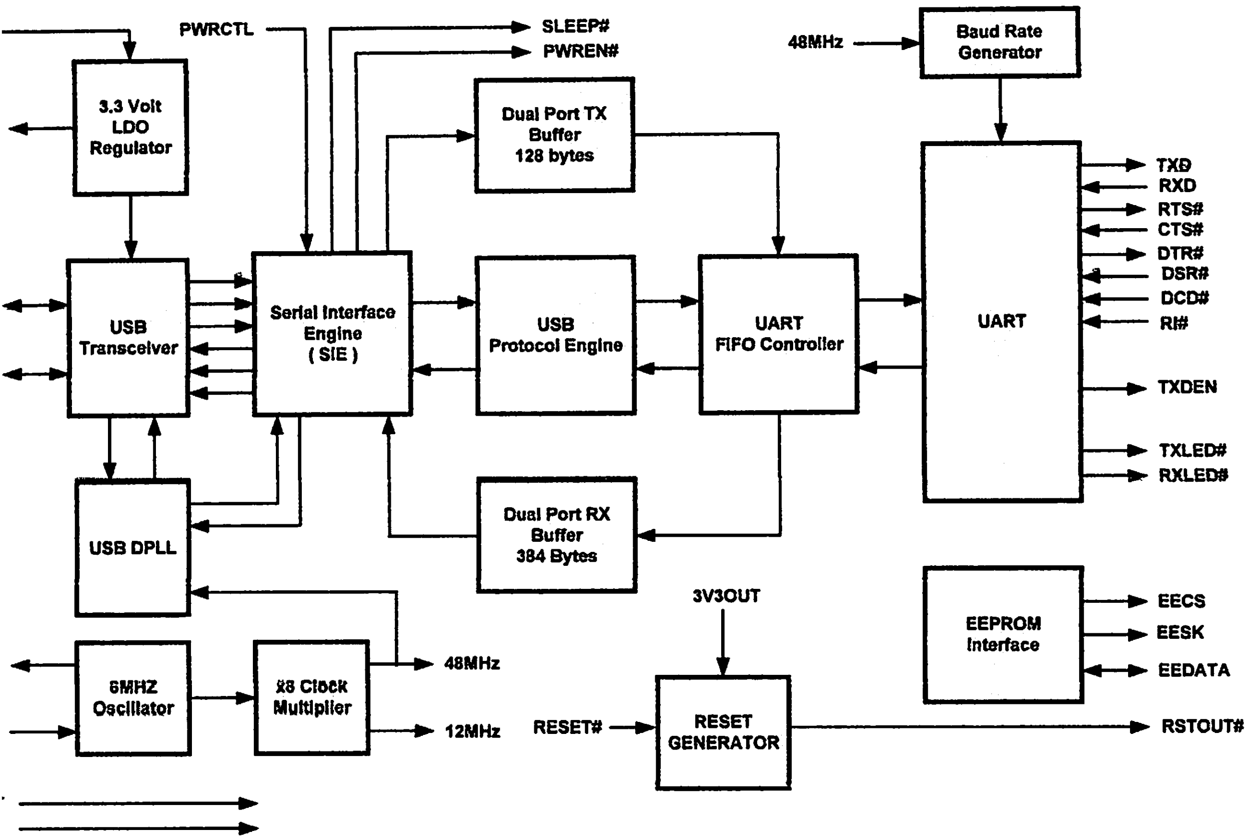
.ess External Support Components
s well as eliminating the RCCLK RC network, and or most applications the need for an external reset ircuit, we have also eliminated the requirement for 100k pull-up on EECS to select 6MHz operation. Vhen the FT232BM is being used without the onfiguration EEPROM, EECS, EESK and EDATA can now be left n/c. For circuits requiring long reset time (where the device is reset xternally using a reset generator i.c., or reset is ontrolled by the IO port of a MCU, FPGA or ASIC levice) an external transistor circuit is no longer equired as the 1k5 pull-up resistor on USB DP can e wired to the RESETO# pin instead of to 3.3v. Note : RESETO# drives out at 3.3v level, not at 5v /CC level. This is the preferred configuration for ew designs. In some other configurations, RSTO# an be used to reset external logic / MCU circuitry.

Extended EEROM Support
The previous generation of devices only supported EEPROM of type 93C46 (128 x 16 bit). The new levices will also work with EEPROM type 93C56 256 x 16 bit) and 93C66 (512 x 16 bit). The xtra space is not used by the device, however it s available for use by other external MCU / logic vhilst the FT232BM is being held in reset.

- **USB 2.0 (full speed option)**
  A new EEPROM based option allows the FT232BM to return a USB 2.0 device descriptor as opposed to USB 1.1. Note : The device would be a USB 2.0 Full Speed device (12Mb/s) as opposed to a USB 2.0 High Speed device (480Mb/s).
- **Multiple Device Support without EEPROM**
  When no EEPROM (or a blank or invalid EEPROM) is attached to the device, the FT232BM no longer gives a serial number as part of it's USB descriptor. This allows multiple devices to be simultaneously connected to the same PC. However, we still highly recommend that EEPROM is used, as without serial numbers a device can only be identified by which hub port in the USB tree it is connected to which can change if the end user re-plugs the device into a different port.



³lock Diagram (simplified)



⁼unctional Block Descriptions

V LDO Regulator

'he 3.3V LDO Regulator generates the 3.3 volt reference voltage for driving the USB transceiver cell output uffers. It requires an external decoupling capacitor to be attached to the 3V3OUT regulator output pin. It also rovides 3.3v power to the RSTOUT# pin. The main function of this block is to power the USB Transceiver and 1e Reset Generator Cells rather than to power external logic. However, external circuitry requiring 3.3v nominal t a current of not greater than 5mA could also draw it's power from the 3V3OUT pin if required.

B Transceiver

'he USB Transceiver Cell provides the USB 1.1 / USB 2.0 full-speed physical interface to the USB cable. The utput drivers provide 3.3 volt level slew rate control signalling, whilst a differential receiver and two single nded receivers provide USB data in, SEO and USB Reset condition detection.

B DPLL

'he USB DPLL cell locks on to the incoming NRZI USB data and provides separate recovered clock and data ignals to the SIE block.

Hz Oscillator

'he 6MHz Oscillator cell generates a 6MHz reference clock input to the X8 Clock multiplier from an external MHz crystal or ceramic resonator.

:lock Multiplier

he x8 Clock Multiplier takes the 6MHz input from the Oscillator cell and generates a 12MHz reference clock for ıe SIE, USB Protocol Engine and UART FIFO controller blocks. It also generates a 48MHz reference clock for ıe USB DPPL and the Baud Rate Generator blocks.

ial Interface Engine (SIE)

he Serial Interface Engine (SIE) block performs the Parallel to Serial and Serial to Parallel conversion of the ISB data. In accordance to the USB 1.1 specification, it performs bit stuffing / un-stuffing and CRC5 / CRC16 eneration / checking on the USB data stream.

3 Protocol Engine

'he USB Protocol Engine manages the data stream from the device USB control endpoint. It handles the ɔw level USB protocol (Chapter 9) requests generated by the USB host controller and the commands for ontrolling the functional parameters of the UART.

al Port TX Buffer (128 bytes)

)ata from the USB data out endpoint is stored in the Dual Port TX buffer and removed from the buffer to the IART transmit register under control of the UART FIFO controller.

al Port RX Buffer (384 bytes)

)ata from the UART receive register is stored in the Dual Port RX buffer prior to being removed by the SIE on a JSB request for data from the device data in endpoint.

RT FIFO Controller

'he UART FIFO controller handles the transfer of data between the Dual Port RX and TX buffers and the UART 'ansmit and receive registers.

RT

'he UART performs asynchronous 7 / 8 bit Parallel to Serial and Serial to Parallel conversion of the data on he RS232 (RS422 and RS485) interface. Control signals supported by the UART include RTS, CTS, DSR ,)TR, DCD and RI. The UART provides a transmitter enable control signal (TXDEN) to assist with interfacing ɔ RS485 transceivers. The UART supports RTS/CTS, DSR/DTR and X-On/X-Off handshaking options. Iandshaking, where required, is handled in hardware to ensure fast response times. The UART also supports he RS232 BREAK setting and detection conditions.

ud Rate Generator

'he Baud Rate Generator provides a x16 clock input to the UART from the 48MHz reference clock and consists ıf a 14 bit prescaler and 3 register bits which provide fine tuning of the baud rate (used to divide by a number ılus a fraction). This determines the Baud Rate of the UART which is programmable from 183 baud to 3 million ıaud.

SET Generator

The Reset Generator Cell provides a reliable power-on reset to the device internal circuitry on power up. An ıdditional RESET# input and RSTOUT# output are provided to allow other devices to reset the FT232BM or the FT232BM to reset other devices respectively. During reset, RSTOUT# is high-impedance otherwise it drives ıut at the 3.3v provided by the onboard regulator. RSTOUT# can be used to control the 1k5 pull-up on USB)P directly where delayed USB enumeration is required. It can also be used to reset other devices. RSTOUT# ıill stay high-impedance for approximately 5ms after VCC has risen above 3.5v <u>AND</u> the device oscillator is ınning <u>AND</u> RESET# is high. RESET# should be tied to VCC unless it is a requirement to reset the device 'om external logic or an external reset generator i.c.

'ROM Interface

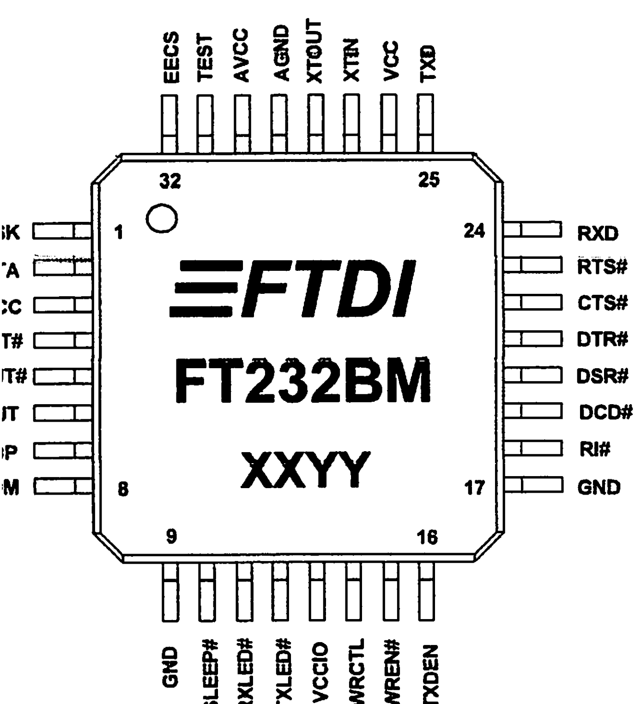
hough the FT232BM will work without the optional EEPROM, an external 93C46 (93C56 or 93C66) EEPROM ın be used to customise the USB VID, PID, Serial Number, Product Description Strings and Power Descriptor ılue of the FT232BM for OEM applications. The EEPROM is also required for applications where multiple T232BM's are connected to a single PC as the drivers rely on a unique serial number for each device to bind unique virtual COM port to each individual device. Other parameters controlled by the EEPROM include emote Wake Up, Isochronous Transfer Mode, Soft Pull Down on Power-Off and USB 2.0 descriptor modes. he EEPROM should be a 16 bit wide configuration such as a MicroChip 93LC46B or equivalent capable of 1Mb/s clock rate at VCC = 4.4v to 5.25v. The EEPROM is programmable-on board over USB using a utility vailable from FTDI's web site (http://www.ftdichip.com). This allows a blank part to be soldered onto the PCB nd programmed as part of the manufacturing and test process.

no EEPROM is connected (or the EEPROM is blank), the FT232BM will use it's built-in default VID, PID roduct Description and Power Descriptor Value. In this case, the device will not have a serial number as part f the USB descriptor.

)evice Pin-Out

Figure 1

ı-Out (LQFP-32 Package)



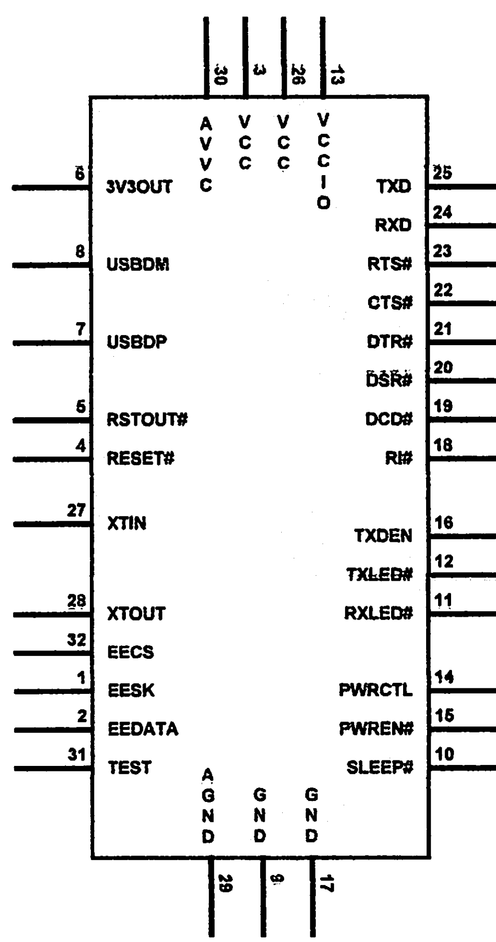


Figure 2

Pin-Out (Schematic Symbol)

;ignal Descriptions

1 - FT232BM - PINOUT DESCRIPTION

NTERFACE GROUP

Signal	Type	Description
TXD	OUT	Transmit Asynchronous Data Output
RXD	IN	Receive Asynchronous Data Input
RTS#	OUT	Request To Send Control Output / Handshake signal
CTS#	IN	Clear To Send Control Input / Handshake signal
DTR#	OUT	Data Terminal Ready Control Output / Handshake signal
DSR#	IN	Data Set Ready Control Input / Handshake signal
DCD#	IN	Data Carrier Detect Control Input
RI#	IN	Ring Indicator Control Input. When the Remote Wakeup option is enabled in the EEPROM, taking RI# low can be used to resume the PC USB Host controller from suspend.
TXDEN	OUT	Enable Transmit Data for RS485

ITERFACE GROUP

Signal	Type	Description
USBDP	I/O	USB Data Signal Plus (Requires 1.5k pull-up to 3V3OUT or RSTOUT#)
USBDM	I/O	USB Data Signal Minus

)M INTERFACE GROUP

Signal	Type	Description
EECS	I/O	EEPROM – Chip Select. For 48MHz operation pull EECS to GND using a 10k resistor. For 6MHz operation no resistor is required. Tri-State during device reset.
EESK	OUT	Clock signal to EEPROM. Tri-State during device reset, else drives out.
EEDATA	I/O	EEPROM – Data I/O Connect directly to Data-In of the EEPROM and to Data-Out of the EEPROM via a 2k2 resistor. Also pull Data-Out of the EEPROM to VCC via a 10k resistor for correct operation. Tri-State during device reset.

R CONTROL GROUP

Signal	Type	Description
SLEEP#	OUT	Goes Low during USB Suspend Mode. Typically used to power-down an external TTL to RS232 level converter i.c. in USB -> RS232 converter designs.
PWREN#	OUT	Goes Low after the device is configured via USB, then high during USB suspend. Can be used to control power to external logic using a P-Channel Logic Level MOSFET switch. Enable the Interface Pull-Down Option in EEPROM when using the PWREN# pin in this way.
PWRCTL	IN	Bus Powered – Tie Low / Self Powered – Tie High

.LANEOUS SIGNAL GROUP

Signal	Type	Description
RESET#	IN	Can be used by an external device to reset the FT232BM. If not required, tie to VCC.
RSTOUT#	OUT	Output of the internal Reset Generator. Stays high impedance for ~ 2ms after VCC > 3.5v and the internal clock starts up, then clamps it's output to the 3.3v output of the internal regulator. Taking RESET# low will also force RSTOUT# to go high impedance. RSTOUT# is NOT affected by a USB Bus Reset.
TXLED#	O.C.	LED Drive - Pulses Low when Transmitting Data via USB
RXLED#	O.C.	LED Drive - Pulses Low when Receiving Data via USB
XTIN	IN	Input to 6MHz Crystal Oscillator Cell. This pin can also be driven by an external 6MHz clock if required. Note : Switching threshold of this pin is VCC/2, so if driving from an external source, the source must be driving at 5V CMOS level or a.c. coupled to centre around VCC/2.
XTOUT	OUT	Output from 6MHz Crystal Oscillator Cell. XTOUT stops oscillating during USB suspend, so take care if using this signal to clock external logic.
TEST	IN	Puts device in i.c. test mode – must be tied to GND for normal operation.

? AND GND GROUP

Signal	Type	Description
3V3OUT	OUT	3.3 volt Output from the integrated L.D.O. regulator This pin should be decoupled to GND using a 33nF ceramic capacitor in close proximity to the device pin. It's prime purpose is to provide the internal 3.3v supply to the USB transceiver cell and the RSTOUT# pin. A small amount of current (<= 5mA) can be drawn from this pin to power external 3.3v logic if required.
VCC	PWR	+4.4 volt to +5.25 volt VCC to the device core, LDO and and none-UART interface pins.
VCCIO	PWR	+3.0 volt to +5.25 volt VCC to the UART interface pins 10..12, 14..16 and 18..25. When interfacing with 3.3v external logic connect VCCIO to the 3.3v supply of the external logic, otherwise connect to VCC to drive out at 5v CMOS level.
GND	PWR	Device– Ground Supply Pins
AVCC	PWR	Device - Analog Power Supply for the internal x8 clock multiplier
AGND	PWR	Device - Analog Ground Supply for the internal x8 clock multiplier

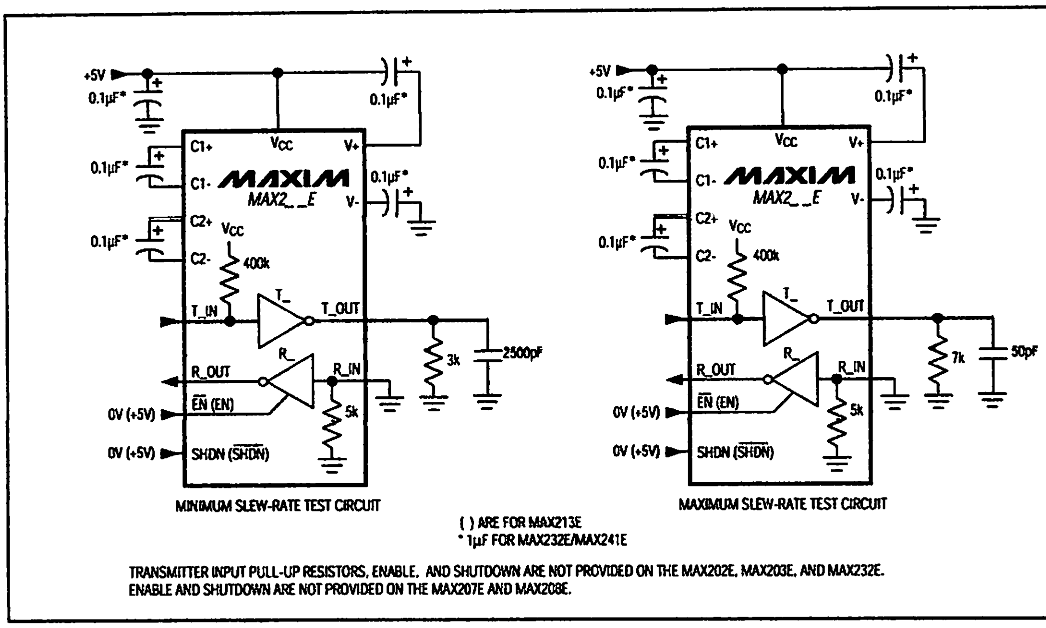


Figure 3. Transition Slew-Rate Circuit

Detailed Description

The MAX202E–MAX213E, MAX232E/MAX241E consist of three sections: charge-pump voltage converters, drivers (transmitters), and receivers. These E versions provide extra protection against ESD. They survive ±15kV discharges to the RS-232 inputs and outputs, tested using the Human Body Model. When tested according to IEC1000-4-2, they survive ±8kV contact-discharges and ±15kV air-gap discharges. The rugged E versions are intended for use in harsh environments or applications where the RS-232 connection is frequently changed (such as notebook computers). The standard (non-"E") MAX202, MAX203, MAX205–MAX208, MAX211, MAX213, MAX232, and MAX241 are recommended for applications where cost is critical.

+5V to ±10V Dual Charge-Pump Voltage Converter

The +5V to ±10V conversion is performed by dual charge-pump voltage converters (Figure 4). The first charge-pump converter uses capacitor C1 to double the +5V into +10V, storing the +10V on the output filter capacitor, C3. The second uses C2 to invert the +10V into -10V, storing the -10V on the V- output filter capacitor, C4.

In shutdown mode, V+ is internally connected to V_{CC} by a 1kΩ pull-down resistor, and V- is internally connected to ground by a 1kΩ pull-up resistor.

RS-232 Drivers

With V_{CC} = 5V, the typical driver output voltage swing is ±8V when loaded with a nominal 5kΩ RS-232 receiver. The output swing is guaranteed to meet EIA/TIA-232E and V.28 specifications that call for ±5V minimum output levels under worst-case conditions. These include a 3kΩ load, minimum V_{CC}, and maximum operating temperature. The open-circuit output voltage swings from (V+ - 0.6V) to V-.

Input thresholds are CMOS/TTL compatible. The unused drivers' inputs on the MAX205E–MAX208E, MAX211E, MAX213E, and MAX241E can be left unconnected because 400kΩ pull-up resistors to V_{CC} are included on-chip. Since all drivers invert, the pull-up resistors force the unused drivers' outputs low. The MAX202E, MAX203E, and MAX232E do not have pull-up resistors on the transmitter inputs.

When in low-power shutdown mode, the MAX205E/MAX206E/MAX211E/MAX213E/MAX241E driver outputs are turned off and draw only leakage currents—even if they are back-driven with voltages between 0V and 12V. Below -0.5V in shutdown, the transmitter output is diode-clamped to ground with a 1kΩ series impedance.

RS-232 Receivers

The receivers convert the RS-232 signals to CMOS-logic output levels. The guaranteed 0.8V and 2.4V receiver input thresholds are significantly tighter than the ±3V thresholds required by the EIA/TIA-232E specification. This allows the receiver inputs to respond to TTL/CMOS-logic levels, as well as RS-232 levels.

The guaranteed 0.8V input low threshold ensures that receivers shorted to ground have a logic 1 output. The 5kΩ input resistance to ground ensures that a receiver with its input left open will also have a logic 1 output.

Receiver inputs have approximately 0.5V hysteresis. This provides clean output transitions, even with slow rise/fall-time signals with moderate amounts of noise and ringing.

In shutdown, the MAX213E's R4 and R5 receivers have no hysteresis.

Shutdown and Enable Control (MAX205E/MAX206E/MAX211E/MAX213E/MAX241E)

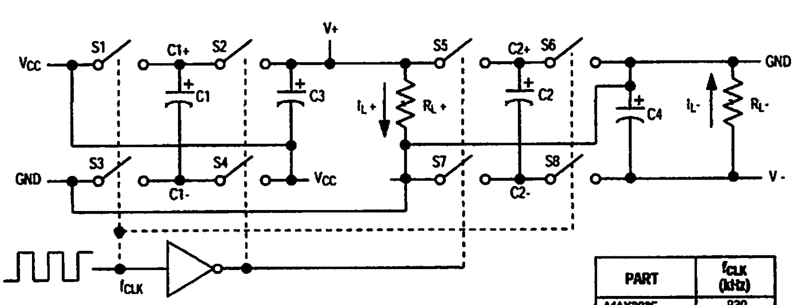
In shutdown mode, the charge pumps are turned off, V+ is pulled down to V_{CC}, V- is pulled to ground, and the transmitter outputs are disabled. This reduces supply current typically to 1µA (15µA for the MAX213E). The time required to exit shutdown is under 1ms, as shown in Figure 5.

Receivers

All MAX213E receivers, except R4 and R5, are put into a high-impedance state in shutdown mode (see Tables 1a and 1b). The MAX213E's R4 and R5 receivers still function in shutdown mode. These two awake-in-shutdown receivers can monitor external activity while maintaining minimal power consumption.

The enable control is used to put the receiver outputs into a high-impedance state, to allow wire-OR connection of two EIA/TIA-232E ports (or ports of different types) at the UART. It has no effect on the RS-232 drivers or the charge pumps.

Note: The enable control pin is active low for the MAX211E/MAX241E ($\overline{EN}$), but is active high for the MAX213E (EN). The shutdown control pin is active high for the MAX205E/MAX206E/MAX211E/MAX241E (SHDN), but is active low for the MAX213E ($\overline{SHDN}$).



PART	f_{CLK} (kHz)
MAX202E	230
MAX203E	230
MAX205E–206E	200
MAX211E/213E	200
MAX232E	140
MAX241E	30

Figure 4. Charge-Pump Diagram

The MAX213E's receiver propagation delay is typically 0.5µs in normal operation. In shutdown mode, propagation delay increases to 4µs for both rising and falling transitions. The MAX213E's receiver inputs have approximately 0.5V hysteresis, except in shutdown, when receivers R4 and R5 have no hysteresis.

When entering shutdown with receivers active, R4 and R5 are not valid until 80µs after SHDN is driven low. When coming out of shutdown, all receiver outputs are invalid until the charge pumps reach nominal voltage levels (less than 2ms when using 0.1µF capacitors).

±15kV ESD Protection

As with all Maxim devices, ESD-protection structures are incorporated on all pins to protect against electrostatic discharges encountered during handling and assembly. The driver outputs and receiver inputs have extra protection against static electricity. Maxim's engineers developed state-of-the-art structures to protect these pins against ESD of ±15kV without damage. The ESD structures withstand high ESD in all states: normal operation, shutdown, and powered down. After an ESD event, Maxim's E versions keep working without latchup, whereas competing RS-232 products can latch and must be powered down to remove latchup.

ESD protection can be tested in various ways; the transmitter outputs and receiver inputs of this product family are characterized for protection to the following limits:

1) ±15kV using the Human Body Model
2) ±8kV using the contact-discharge method specified in IEC1000-4-2
3) ±15kV using IEC1000-4-2's air-gap method.

ESD Test Conditions

ESD performance depends on a variety of conditions. Contact Maxim for a reliability report that documents test set-up, test methodology, and test results.

Human Body Model

Figure 6a shows the Human Body Model, and Figure 6b shows the current waveform it generates when discharged into a low impedance. This model consists of a 100pF capacitor charged to the ESD voltage of interest, which is then discharged into the test device through a 1.5kΩ resistor.

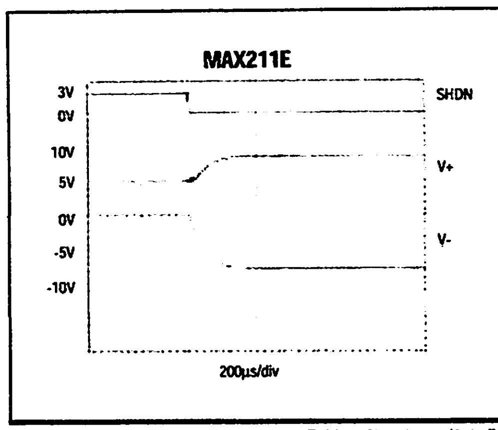


Figure 5. MAX211E V+ and V- when Exiting Shutdown (0.1µF capacitors)

Table 1a. MAX205E/MAX206E/MAX211E/MAX241E Control Pin Configurations

SHDN	EN	OPERATION STATUS	Tx	Rx
0	0	Normal Operation	All Active	All Active
0	1	Normal Operation	All Active	All High-Z
1	X	Shutdown	All High-Z	All High-Z

X = Don't Care

Table 1b. MAX213E Control Pin Configurations

SHDN	EN	OPERATION STATUS	Tx 1–4	Rx	
				1–3	4, 5
0	0	Shutdown	All High-Z	High-Z	High-Z
0	1	Shutdown	All High-Z	High-Z	Active*
1	0	Normal Operation	All Active	High-Z	High-Z
1	1	Normal Operation	All Active	Active	Active

**Active = active with reduced performance*

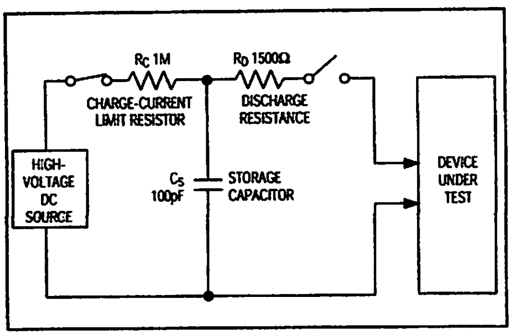


Figure 6a. Human Body ESD Test Model

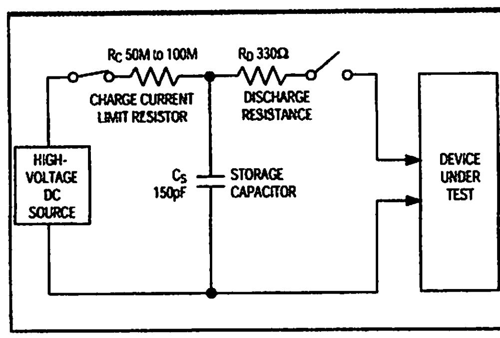


Figure 7a. IEC1000-4-2 ESD Test Model

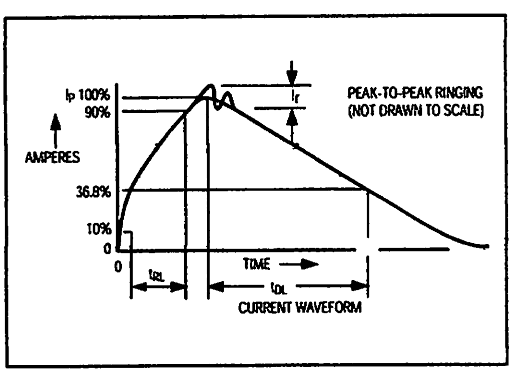


Figure 6b. Human Body Model Current Waveform

IEC1000-4-2

The IEC1000-4-2 standard covers ESD testing and performance of finished equipment; it does not specifically refer to integrated circuits. The MAX202E/MAX203E–MAX213E, MAX232E/MAX241E help you design equipment that meets level 4 (the highest level) of IEC1000-4-2, without the need for additional ESD-protection components.

The major difference between tests done using the Human Body Model and IEC1000-4-2 is higher peak current in IEC1000-4-2, because series resistance is lower in the IEC1000-4-2 model. Hence, the ESD withstand voltage measured to IEC1000-4-2 is generally lower than that measured using the Human Body Model. Figure 7b shows the current waveform for the 8kV IEC1000-4-2 level-four ESD contact-discharge test.

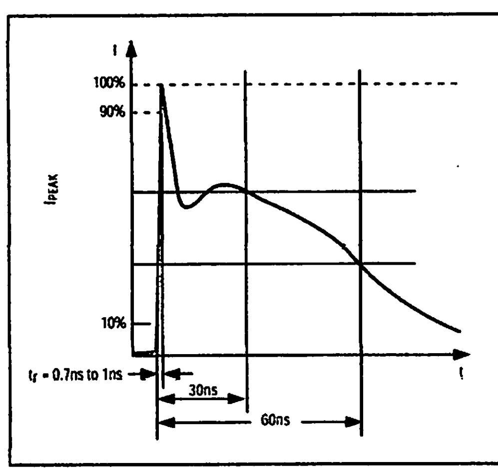


Figure 7b. IEC1000-4-2 ESD Generator Current Waveform

The air-gap test involves approaching the device with a charged probe. The contact-discharge method connects the probe to the device before the probe is energized.

Machine Model

The Machine Model for ESD tests all pins using a 200pF storage capacitor and zero discharge resistance. Its objective is to emulate the stress caused by contact that occurs with handling and assembly during manufacturing. Of course, all pins require this protection during manufacturing, not just RS-232 inputs and outputs. Therefore, after PC board assembly, the Machine Model is less relevant to I/O ports.

Pin Descriptions (continued)

MAX206E

PIN	NAME	FUNCTION
1, 2, 3, 24	T_OUT	RS-232 Driver Outputs
4, 16, 23	R_IN	RS-232 Receiver Inputs
5, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
6, 7, 18, 19	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.5V to +5.5V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	+2V_{CC} generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	-2V_{CC} generated by the charge pump
20	$\overline{EN}$	Receiver Enable—active low
21	SHDN	Shutdown Control—active high

MAX207E

PIN	NAME	FUNCTION
1, 2, 3, 20, 24	T_OUT	RS-232 Driver Outputs
4, 16, 23	R_IN	RS-232 Receiver Inputs
5, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
6, 7, 18, 19, 21	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.75V to +5.25V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	+2V_{CC} generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	-2V_{CC} generated by the charge pump

MAX208E

PIN	NAME	FUNCTION
1, 2, 20, 24	T_OUT	RS-232 Driver Outputs
3, 7, 16, 23	R_IN	RS-232 Receiver Inputs
4, 6, 17, 22	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
5, 18, 19, 21	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
8	GND	Ground
9	V_{CC}	+4.5V to +5.5V Supply Voltage
10, 12	C1+, C1-	Terminals for positive charge-pump capacitor
11	V+	+2V_{CC} generated by the charge pump
13, 14	C2+, C2-	Terminals for negative charge-pump capacitor
15	V-	-2V_{CC} generated by the charge pump

±15kV ESD-Protected, +5V RS-232 Transceivers

Pin Descriptions (continued)

MAX211E/MAX213E/MAX241E

PIN	NAME	FUNCTION
1, 2, 3, 28	T_OUT	RS-232 Driver Outputs
4, 9, 18, 23, 27	R_IN	RS-232 Receiver Inputs
5, 8, 19, 22, 26	R_OUT	TTL/CMOS Receiver Outputs. For the MAX213E, receivers R4 and R5 are active in shutdown mode when EN = 1. For the MAX211E and MAX241E, all receivers are inactive in shutdown.
6, 7, 20, 21	T_IN	TTL/CMOS Driver Inputs. Only the MAX211E, MAX213E, and MAX241E have internal pull-ups to V_{CC}.
10	GND	Ground
11	V_{CC}	+4.5V to +5.5V Supply Voltage
12, 14	C1+, C1-	Terminals for positive charge-pump capacitor
13	V+	+2V_{CC} voltage generated by the charge pump
15, 16	C2+, C2-	Terminals for negative charge-pump capacitor
17	V-	-2V_{CC} voltage generated by the charge pump
24	$\overline{EN}$	Receiver Enable—active low (MAX211E, MAX241E)
	EN	Receiver Enable—active high (MAX213E)
25	SHDN	Shutdown Control—active high (MAX211E, MAX241E)
	$\overline{SHDN}$	Shutdown Control—active low (MAX213E)

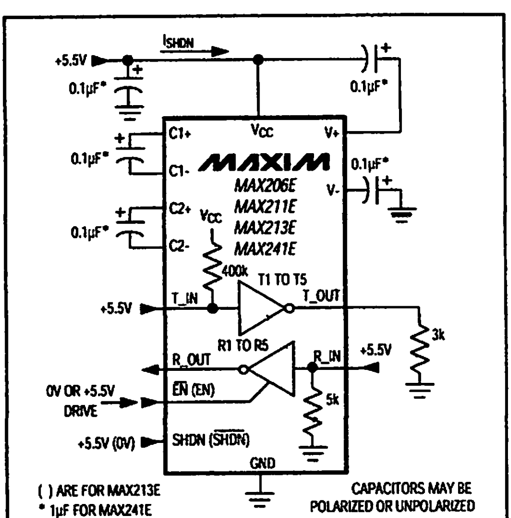


Figure 1. Shutdown-Current Test Circuit (MAX206E, MAX211E/MAX213E/MAX241E)

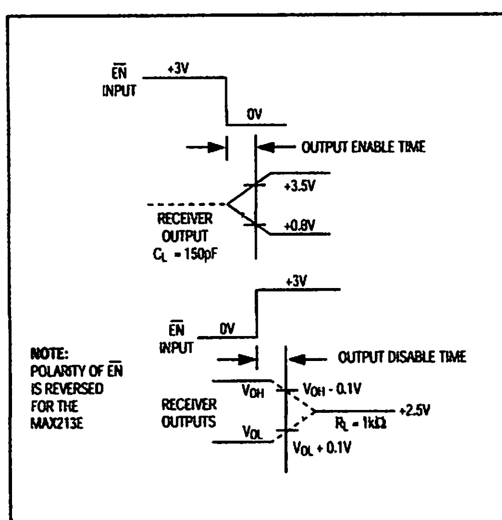


Figure 2. Receiver Output Enable and Disable Timing (MAX205E/MAX206E/MAX211E/MAX213E/MAX241E)

Supported Card Types

ACR30 can operate MCU card with T=0 and T=1 protocol. The table presented in Appendix A explains ch card type selection value must be specified for the various card types supported by the reader.

Memory-based smart cards (synchronous interface) (*)

'104' type EEPROM non-reloadable token counter cards, including:

Gemplus GPM103,

Siemens SLE 4406

Siemens SLE4436 and SLE5536 (firmware 2.10 onwards)

Cards following the I^2C bus protocol (free memory cards) with memory capacity up to 16 Kbit and minimum 4 bytes page write capability, including:

Atmel AT24C01/02/04/08/16

Gemplus GFM2K, GFM4K

SGS-Thomson ST14C02C,14C04C

Siemens SLE4432/4442 intelligent 256 bytes EEPROM with write protect function:

SLE 4432, SLE 4442

Siemens SLE 4418/4428 intelligent 1K bytes EEPROM with write-protect function:

SLE 4418, SLE 4428

IOTE (*) – SAM READER DOES NOT SUPPORT FOR MEMORY-BASED SMART CARD

Microcontroller-based smart cards (asynchronous interface)

ACR30 supports EEPROM microcontroller-based cards with internal programming voltage (VPP) ieration and the following programming parameters transmitted in the ATR:

PI1 = 0 or 5

I = 25 or 50

ACR30 performs the Protocol and Parameters Selection (PPS) procedure as specified *in ISO7816-997.*

ien the card ATR indicates the specific operation mode (TA_2 present; bit b5 of TA_2 must be 0) and that ticular mode is not supported by the ACR30, the reader will reset the card to set it to negotiable mode. If card cannot be set to negotiable mode, the reader will reject the card.

ien the card ATR indicates the negotiable mode (TA_2 not present) and communication parameters other in the default parameters, the ACR30 will execute the PPS and try to use the communication parameters

: the card suggested in its ATR. If the card does not accept the PPS, the reader will use the default ameters (F=372, D=1).
the meaning of the aforementioned parameters, please refer to *ISO7816, part 3*.

Smart Card Interface

ı interface between the ACR30 and the inserted smart card follows the specifications of *ISO7816-3* with tain restrictions or enhancements to increase the practical functionality of the ACR30.

' *Smart Card Power Supply VCC (C1)*

e current consumption of the inserted card must not be higher than **50mA**.

! *Programming Voltage VPP (C6)*

;ording to ISO 7816-3, the smart card contact C6 (VPP) supplies the programming voltage to the smart d. Since all common smart cards in the market are EEPROM based and do not require the provision of external programming voltage, the contact C6 (VPP) has been implemented as a normal control signal in ACR30. The electrical specifications of this contact are identical to those of the signal RST (at contact).

ł *Card Type Selection*

ɘ controlling PC has to always select the card type through the proper command sent to the ACR30 prior ıctivating the inserted card. This includes both the memory cards and MCU-based cards.
· MCU-based cards the reader allows to select the preferred protocol, T=0 or T=1. However, this ection is only accepted and carried out by the reader through the PPS when the card inserted in the ıder supports both protocol types. Whenever an MCU-based card supports only one protocol type, T=0 or 1, the reader automatically uses that protocol type, regardless of the protocol type selected by the ɔlication.

ł *Interface for Microcontroller-based Cards*

r microcontroller-based smart cards only the contacts C1 (VCC), C2 (RST), C3 (CLK), C5 (GND) and C7)) are used. A frequency of 3.6864 / 4 MHz is applied to the CLK signal (C3).

ʒ *Card Tearing Protection*

ɘ ACR30 provides a mechanism to protect the inserted card when it is suddenly withdrawn while it is wered up. The power supply to the card and the signal lines between the ACR30 and the card are nediately deactivated when the card is being removed. As a general rule, however, to avoid any electrical mage, **a card should only be removed from the reader while it is powered down.**

E - The ACR30 does never by itself switch on the power supply to the inserted card. This must explicitly be done by the olling computer through the proper command sent to the reader.

Power Supply

ACR30 requires a voltage of 5V DC, 100mA regulated power supply. The ACR30 gets the power supply n PC (through the cable supplied along with each type of reader).

tus LEDs

en LED on the front of the reader indicate the activation status of the smart card interface:

en LED - Indicates power supply to the smart card is switched on, i.e., the smart card is activated.

Serial Interface

: ACR30 is connected to a computer through a serial asynchronous interface following the RS-232 ndard.

Communication Parameters

: following communication parameters are used by the ACR30 and cannot be modified by the host nputer:

Transmission protocol	:	serial asynchronous
Parity	:	none
Data Bits	:	8
Stop Bits	:	1
Handshake	:	through CTS

: ACR30 provides two means to select the transmission speed (baud rate) used by the reader in the rmal operation, by hardware and/or by software.

.1 Hardware Baud Rate

: default hardware baud rate setting is 9600 bps.

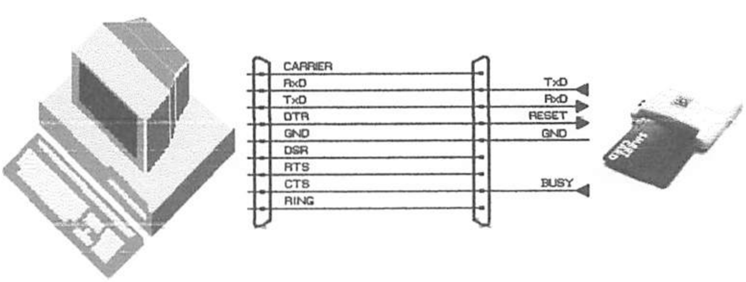
.2 Software Baud Rate Selection

: *SET_PROTOCOL* command allows setting the transmission speed (baud rate) and a delay time erted between the bytes transmitted by the reader to the PC.

ase note that the setting made with this command is volatile and will be lost when the reader is being et or powered up next time.

2 Interface Wiring

r the communication between the ACR30 and a computer, five lines of the RS-232 interface are used: D, TxD, CTS, DTR and GND.



·232 Interface Wiring

in	PC	Cyber-mouse	Function
2	RxD	TxD	Data transmitted from PC to ACR30.
3	TxD	RxD	Data transmitted from ACR30 to PC.
4	DTR	RESET	RESET input signal. Allows performing hardware reset of the reader module through the RS-232 interface. Applying a logic '1' signal (negative voltage according to the RS-232 convention) to this pin causes a hardware reset of the ACR30.
5	GND	GND	Reference voltage level for power supply and serial interface.
8	CTS	BUSY	CTS (Clear To Send) signal to the PC. Indicates to the PC whether the ACR30 is ready to receive the next command. A logic '0' signal (positive voltage according to the RS-232 convention) is applied to this pin while the ACR30 is executing a command. Only when a '1' signal (negative voltage according to the RS-232 convention) is present at this pin can the PC send a command to the ACR30.

TE - Communication problems between the ACR30 and a PC can occur if a 25 pin to 9 pin RS-232 adapter or a cable is used in :h not all 9 signal lines are connected. Adapters supplied with computer mouse frequently have not all lines connected. For the ect operation of the reader, use only a 9 pin to 25pin adapter and a serial interface cable in which all 9 signal lines are connected!

TE - To prevent any radio interference between the ACR30 and other electrical and electronic equipment, do <u>not</u> use an RS-232 le <u>longer than 3 meters</u>!

USB Interface

ACR30 is connected to a computer through a USB following the USB standard.

Communication Parameters

ACR30 is connected to a computer through USB as specified in the USB Specification.

ACR30 is working in low speed mode, i.e. 1.5 Mbps.

B Interface Wiring

	Signal	Function
	V_{BUS}	+5V power supply for the reader
	D-	Differential signal transmits data between ACR30 and PC.
	D+	Differential signal transmits data between ACR30 and PC.
	GND	Reference voltage level for power supply

TE - In order for the ACR30 functioning properly through USB interface, either ACS proprietary device drive or ACS PC/SC device er has to be installed. Please refer to the *Device Driver Installation Guide* for more detail.

Communication protocol

he normal operation, the ACR30 acts as a slave device with regard to the communication between a ıputer and the reader. The communication is carried out in the form of successive command-response hanges. The computer transmits a command to the reader and receives a response from the reader after command has been executed. A new command can be transmitted to the ACR30 only after the ponse to the previous command has been received.

ıre are two cases where the reader transmits data without having received a command from the nputer, namely, the Reset Message of the reader and the Card Status Message.

Command

.1 Normal Command (Length < 255 bytes)

command consists of four protocol bytes and a variable number of data bytes and has the following ıcture:

te	1	2	3	4 ... N+3 (0<N<255)	N+4
	Header	Instruction	Data length = N	Data	Checksum

ader Always 01_H to indicate the start of a command.

truction The instruction code of the command to be carried out by the ACR30

ta Length Number of subsequent data bytes. (0 < N < 255)

ta Data contents of the command.
For a READ command, for example, the data bytes would specify the start address and the number of bytes to be read. For a WRITE command, the data bytes would specify the start address and the data to be written to the card.
The data bytes can represent values to be written to a card and/or command parameters such as an address, a counter, etc.

ecksum The checksum is computed by XORing all command bytes including header, instruction, data length and all data bytes.

e following example shows the structure of a command with instruction code = 91_H and three data bytes h the values 11_H, 22_H and 33_H, respectively:

yte	1	2	3	4	5	6	7
	01_H	91_H	03_H	11_H	22_H	33_H	93_H

.2 Extended Command

:ommand consists of six protocol bytes and a variable number of data bytes and has the following ıcture:

te	1	2	3	4 5	6 ... N+5 (N>0)	N+6
	Header	Instruction	Data Length = N		Data	Checksum
			FF_H	Data Length N		

ıder Always 01_H to indicate the start of a command.

truction The instruction code of the command to be carried out by the ACR30.

:a Length Number of subsequent data bytes, and is encoded in 3 bytes. The first byte is FF_H. The second byte and the third byte represent data length N.

:a Data contents of the command.

For a READ command, for example, the data bytes would specify the start address and the number of bytes to be read. For a WRITE command, the data bytes would specify the start address and the data to be written to the card.

The data bytes can represent values to be written to a card and/or command parameters such as an address, a counter, etc.

ecksum The checksum is computed by XORing all command bytes including header, instruction, data length and all data bytes.

? *Response*

ə response from the ACR30 to any command depends if the command where received by the reader hout error (e.g., checksum error).

:.1 No transmission error with normal response (Length < 255 bytes)

ə response by the ACR30 to a correctly received command consists of three protocol bytes, two status :es and a variable number of data bytes and has the following structure:

te	1	2	3	4	5 ... N+4 (0<N<255)	N+5
	Header	SW1	SW2	Data length = N	Data	Checksum

ader Always 01_H to indicate the start of the response.

V1 Indicates the command execution status:

90_H = command successfully executed

60_H = error in command data; command cannot be executed

67_H = error detected in command execution

FF_H = status message initiated by the reader

/2 Further qualification of the command execution status.

A table listing the possible values of the status bytes SW1 and SW2 and the corresponding meaning is given in Appendix B.

a Length Number of subsequent data bytes (0 < N < 255)

a Data contents of the command.

For a *READ_DATA* command, for example, the data bytes would contain the contents of the memory addresses read from the card. The data bytes can represent values read from the card and/or status information.

ecksum The checksum is computed by XORing all response bytes including header, status bytes, data length and all data bytes.

following example shows the structure of the response to a command which has successfully been cuted and which returns three data bytes with the values 11_H, 22_H and 33_H, respectively:

byte	1	2	3	4	5	6	7	8
	01_H	90_H	00_H	03_H	11_H	22_H	33_H	92_H

.2 No transmission error with extended response

response by the ACR30 to a correctly received command consists of three protocol bytes, two status es and a variable number of data bytes and has the following structure:

e	1	2	3	4	5	6	7 ... N+6 (N>0)	N+7
	Header	SW1	SW2	Data length = N			Data	Checksum
				FF_H	Data Length N			

ader Always 01_H to indicate the start of the response.

/1 Indicates the command execution status:

90_H = command successfully executed

60_H = error in command data; command cannot be executed

67_H = error detected in command execution

FF_H = status message initiated by the reader

/2 Further qualification of the command execution status.

A table listing the possible values of the status bytes SW1 and SW2 and the corresponding meaning is given in Appendix B.

ta Length Number of subsequent data bytes, and is encoded in 3 bytes. The first byte is FF_H. The second byte and the third byte represent data length N.

ta Data contents of the command.

For a *READ_DATA* command, for example, the data bytes would contain the contents of the memory addresses read from the card. The data bytes can represent values read from the card and/or status information.

ɪcksum The checksum is computed by XORing all response bytes including header, status bytes, data length and all data bytes.

.3 Transmission error

ıe receiving party of a command (i.e., the ACR30) or a response (i.e., the computer) detects an error in data length or the checksum of a command, it disregards the received data and sends a "NOT KNOWLEDGE" message to the transmitting party upon completion of the faulty transmission. The "NOT KNOWLEDGE" message consists of two bytes:

byte	1	2
	05_H	05_H

ıe ACR30 responds with a 'NOT ACKNOWLEDGE' message to a command from the computer, the nputer would normally transmit the command again. If the computer detects a transmission error in a ponse from the ACR30, it can send the 'NOT ACKNOWLEDGE' to the reader upon which the reader will ısmit the most recent response again.

! *Reset Message*

eset of the reader occurs automatically whenever the reader is being powered up. A reset can also be uated through the RS-232/USB interface. In either case the reader transmits **one time** a Reset Message, ich has the same structure as the normal response to a command and the following contents:

te	1	2	3	4	5	6
	Header	**SW1**	**SW2**	**Data length**	**Data**	**Checksum**
	01_H	FF_H	00_H	01_H	BAUD=12_H	

.UD Indicates the hardware baud rate setting (default baud rate), which is set to 9600 bps (this is only valid in the RS232 reader).

ɘ reader does not expect an acknowledge signal from the computer. After transmitting the Reset ssage the reader is waiting for the first command from the computer.

! *Card Status Message*

ıen a card is being inserted into the reader or an inserted card is being removed from the reader while the ıder is idle, i.e., not executing a command, the reader transmits a Card Status Message to notify the host nputer of the change in the card insertion status.

ı system where these unsolicited messages from the reader to the computer are not desired, they can be abled with the *SET_NOTIFICATION* command. Please note that the setting made with this command is ıtile and will be lost with the next reader reset or power up. By default, the Card Status Message will be ısmitted by the reader after a reset.

Card Status Messages have the following structure and contents:

d Status Message for Card Insertion

e	1	2	3	4	5
	Header	SW1	SW2	Data length	Checksum
	01_H	FF_H	01_H	00_H	FF_H

d Status Message for Card Removal

e	1	2	3	4	5
	Header	SW1	SW2	Data length	Checksum
	01_H	FF_H	02_H	00_H	FC_H

ard status message is transmitted only **once** for every card insertion or removal event. The reader does expect an acknowledge signal from the computer. After transmitting a status message, the reader waits the next command from the computer.

ΓE - If the card is being removed from the reader **while a card command is being executed**, the reader will transmit a normal ıonse to the computer with the response status bytes indicating the card removal during command execution (see *Appendix B: ponse Status Codes*).

Transmission Protocol

ɜ start of a command (to the reader) or a response (from the reader, including the Reset Message and rd Status Messages) is indicated by the respective party through the transmission of the single byte Start-Γext (STX) character with the value 02_H.

ɜ end of a command or response is indicated through the single byte End-of-Text (ETX) character with value 03_H.

hin the command and response transmission only ASCII characters representing the hexadecimal (hex) its 0...F are used. Each byte of a command or response is splitted into its upper and lower halfbyte ɔble). For each halfbyte is transmitted the ASCII character representing the respective hex digit value. For ımple, to transmit the data byte $3A_H$, two bytes are actually sent on the interface, namely, 33_H (ASCII le for '3') followed by 41_H (ASCII code for 'A'):

Data byte value	$3A_H$	
Transmitted values	33_H = '3'	41_H = 'A'

following example shows the transmission of a command with instruction code $A2_H$ and one data byte the value $3D_H$. The command has the following structure:

byte	1	2	3	4	5
	Header	Instruction	Data length	Data	Checksum
	01_H	$A2_H$	01_H	$3D_H$	$9F_H$

s command is transmitted on the serial interface in 12 bytes as follows:

byte	1	2	3	4	5	6	7	8	9	10	11	12
	STX	'0'	'1'	'A'	'2'	'0'	'1'	'3'	'D'	'9'	'F'	ETX
	02_H	30_H	31_H	41_H	32_H	30_H	31_H	33_H	44_H	39_H	46_H	03_H

the representation of the hex halfbyte values as the corresponding ASCII characters in commands, the R30 accepts both upper case characters 'A' ... 'F' (41_H ... 46_H) and lower case characters 'a' ... 'f' (61_H ... ı):

byte	1	2	3	4	5	6	7	8	9	10	11	12
	STX	'0'	'1'	'A'	'2'	'0'	'1'	'3'	'D'	'9'	'F'	ETX
	02_H	30_H	31_H	41_H	32_H	30_H	31_H	33_H	44_H	39_H	46_H	03_H

... is equivalent to:

byte	1	2	3	4	5	6	7	8	9	10	11	12
	STX	'0'	'1'	'a'	'2'	'0'	'1'	'3'	'd'	'9'	'f'	ETX
	02_H	30_H	31_H	61_H	32_H	30_H	31_H	33_H	64_H	39_H	66_H	03_H

ts response messages, the ACR30 uses upper case characters 'A' ... 'F'.

Commands

e commands executed by the ACR30 can generally be divided into two categories, namely, Control mmands and Card Commands.

ntrol Commands control the internal operation of the ACR30. They do not directly affect the card inserted he reader and are therefore independent of the selected card type.

rd Commands are directed toward the card inserted in the ACR30. The structure of these commands and data transmitted in the commands and responses depend on the selected card type.

1 Control Commands

.1 GET_ACR_STAT

s command returns relevant information about the particular ACR30 model and the current operating tus, such as, the firmware revision number, the maximum data length of a command and response, the pported card types, and whether a card is inserted and powered up.

nmand format

Instruction Code	Data length
01_H	00_H

sponse data format

INTERNAL	MAX_C	MAX_R	C_TYPE		C_SEL	C_STAT

'ERNAL 10 bytes data for internal use only

.X_C The maximum number of command data bytes.

.X_R The maximum number of data bytes that can be requested to be transmitted in a response.

TYPE The card types supported by the ACR30. This data field is a bitmap with each bit representing a particular card type. A bit set to '1' means the corresponding card type is supported by the reader and can be selected with the *SELECT_CARD_TYPE* command. The bit assignment is as follows:

byte	1								2							
card type	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

See Appendix A for the correspondence between these bits and the respective card types.

SEL The currently selected card type as specified in a previous *SELECT_CARD_TYPE* command. A value of 00_H means that no card type has been selected.

STAT Indicates whether a card is physically inserted in the reader and whether the card is powered up:

00_H : no card inserted

01_H : card inserted, not powered up

03_H : card powered up

.2 SET_PROTOCOL

is command is used to control the line speed of the communication channel between ACR30 reader and st device. The line speed of the communication is controlled by two factors, namely, the Delay Factor and Baud Rate.

mmand format

Instruction Code	Data length	Data
		DELAY N
03_H	01_H	

change only the Delay Factor (for RS232 reader only), or

Instruction Code	Data length	Data	
		DELAY N	BAUD RATE
03_H	02_H		

hange the Delay Factor and the Baud Rate (for RS232 reader only).

.AY Determines the time delay inserted by the ACR30 between two consecutive bytes sent in order to adapt to slower host system speeds. The time delay is given by N * 0.1msec, with N ranging from 0 ... 255 (00 - FF_H). The default value is N = 0 (delay changes only valid on RS232 reader).

JD RATE Selects the baud rate (bps) of the serial interface between reader and host system. The default hardware baud rate is 9600 bps. (baud rate changes only valid on RS232 reader).

BAUD RATE	Serial baud rate (bps)
12_H	9600
11_H	19200
10_H	38400
03_H	14400
02_H	28800
01_H	57600
00_H	115200

sponse data format

No response data

: new protocol becomes effective by the completion of the *SET_PROTOCOL* command, immediately **er the ACR30 has sent out the response string to the *SET_PROTOCOL* command.**

.3 SELECT_CARD_TYPE

s command sets the required card type. The firmware in the ACR30 adjusts the communication protocol ween reader and the inserted card according to the selected card type.

nmand format

Instruction Code	Data length	Data
		TYPE
02_H	01_H	

PE See Appendix A for the value to be specified in this command for a particular card to be used.

sponse data format

No response data

pendix A: Supported Card Types

following table summarizes which values must be specified in the *SET_CARD_TYPE* command for a ticular card type to be used, and how the bits in the response to the *GET_ACR_STAT* command respond with the respective card types.

er-mouse card type code	Card Type
00_H	Auto-select T=0 or T=1 communication protocol
01_H (*)	GPM103, SLE4406 SLE4436, SLE5536 (firmware 2.10 onwards)
02_H (*)	I2C
05_H (*)	SLE4418, SLE4428
06_H (*)	SLE4432, SLE4442
$0C_H$	MCU-based cards with T=0 communication protocol
$0D_H$	MCU-based cards with T=1 communication protocol
$C0_H$ (**)	SAM cards with T=0 communication protocol (SAM Reader only)
$D0_H$ (**)	SAM cards with T=1 communication protocol (SAM Reader only)

te (*) – SAM Reader does not support for memory cards.

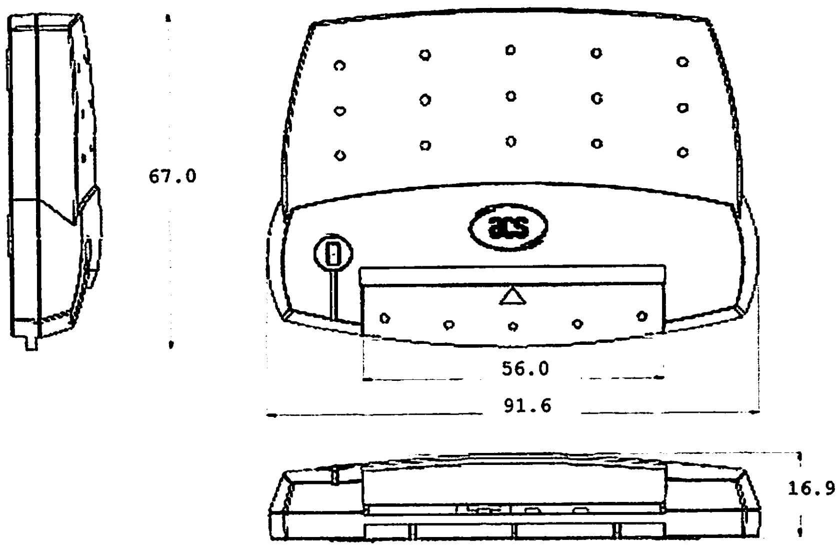
te (**) – SAM Reader only

ıpendix B: Response Status Codes

ꞁ following table summarizes the possible status code bytes SW1, SW2 returned by the ACR30:

ıW1	SW2	Status
90	00	OK – command successfully executed
90	01	OK – using T=1 protocol (only in response to the RESET command)
90	10	OK – synchronous protocol is used (only in response to the RESET command). The exact card type should be selected by using the SELECT_CARD_TYPE command.
60	01	No card type selected
60	02	No card in reader
60	03	Wrong card type specified
60	04	Card not powered up; This status code is also returned in a response if the card was temporarily removed during a card access.
60	05	Invalid Instruction Code
60	20	Card failure
60	22	Short circuit at card connector
62	01	Secret code verify failed
67	01	Command incompatible with card type
67	02	Card address error
67	03	Data length error
67	04	Invalid length of response (with READ command)
67	05	Secret code locked
67	12	APDU command aborted (only MCU-based card using T=1 protocol); the command abortion may be caused by a card internal failure.

ıpendix C: Technical Specifications



vice

R30 Smart Card Reader/Writer

wer supply

ıply voltage ....Regulated 5V DC
ıply current................................< 100mA (without smart card)

ial Communication Interface

ıe..RS-232C, five lines: RxD, TxD, CTS, DTR, GND
ver source.................................From PS/2 mouse interface
ıed ....9600 – 115200 bps

iversal Serial Bus Interface

wer source.................................From USB
ıed ....1.5 Mbps (Low Speed)

ıart Card Interface

ndard..ISO 7816 1/2/3, T=0 and T=1
ıply current................................max. 50mA
art card read / write speed.........9600 – 96000 bps
ırt circuit protection ....+5V / GND on all pins
The presence of the smart card power supply voltage is indicated through a green LED on the reader
K frequency...............................3.6864 / 4 MHz
rd connector..............................sliding contacts (8 contacts)
rd insertion cycles......................min. 100,000

ysical Specifications

ıensions....................................67.0mm (L) x 91.6mm (W) x 16.9mm (H)
lor...Transparent Blue
ight...85g (± 5g allowance for cable) – Cyberfrog casing
ble length, cord, connector.........1.5 meters, Fixed (non-detachable), USB A / RS-232C

erating Conditions

nperature..................................0 - 50° C
midity..40% - 90%

ındard/Certifications

V Level 1, ISO7816-1/2/3, PC/SC, CE, FCC, NETS, Microsoft WHQL 98, ME, 2K, NT 4.0 (Serial), XP

;

ıdows 98, ME, 2K, NT (Serial), XP

EM

:M-Logo possible, customer-specific colors, casing and card connector

MAX232

19-0175; Rev 3; 5/96

MAXIM

±15kV ESD-Protected, +5V RS-232 Transceivers

General Description

The MAX202E-MAX213E, MAX232E/MAX241E line drivers/receivers are designed for RS-232 and V.28 communications in harsh environments. Each transmitter output and receiver input is protected against ±15kV electrostatic discharge (ESD) shocks, without latchup. The various combinations of features are outlined in the *Selection Guide*. The drivers and receivers for all ten devices meet all EIA/TIA-232E and CCITT V.28 specifications at data rates up to 120kbps, when loaded in accordance with the EIA/TIA-232E specification.

The MAX211E/MAX213E/MAX241E are available in 28-pin SO packages, as well as a 28-pin SSOP that uses 60% less board space. The MAX202E/MAX232E come in 16-pin narrow SO, wide SO, and DIP packages. The MAX203E comes in a 20-pin DIP/SO package, and needs no external charge-pump capacitors. The MAX205E comes in a 24-pin wide DIP package, and also eliminates external charge-pump capacitors. The MAX206E/MAX207E/MAX208E come in 24-pin SO, SSOP, and narrow DIP packages. The MAX232E/MAX241E operate with four 1µF capacitors, while the MAX202E/MAX206E/MAX207E/MAX208E/MAX211E/MAX213E operate with four 0.1µF capacitors, further reducing cost and board space.

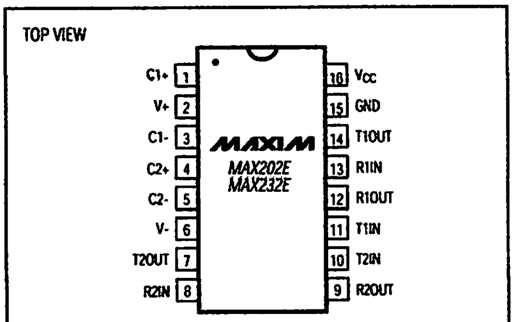
Applications

Notebook, Subnotebook, and Palmtop Computers
Battery-Powered Equipment
Hand-Held Equipment

Features

- **ESD Protection for RS-232 I/O Pins:**
  ±15kV—Human Body Model
  ±8kV—IEC1000-4-2, Contact Discharge
  ±15kV—IEC1000-4-2, Air-Gap Discharge
- **Latchup Free (unlike bipolar equivalents)**
- **Guaranteed 120kbps Data Rate—LapLink™ Compatible**
- **Guaranteed 3V/µs Min Slew Rate**
- **Operate from a Single +5V Power Supply**

Pin Configurations



DIP/SO

Pin Configurations and Typical Operating Circuits continued at end of data sheet.

Ordering Information appears at end of data sheet.

Selection Guide

PART	No. of RS-232 DRIVERS	No. of RS-232 RECEIVERS	RECEIVERS ACTIVE IN SHUTDOWN	No. of EXTERNAL CAPACITORS	LOW-POWER SHUTDOWN	TTL THREE-STATE
MAX202E	2	2	0	4 (0.1µF)	No	No
MAX203E	2	2	0	None	No	No
MAX205E	5	5	0	None	Yes	Yes
MAX206E	4	3	0	4 (0.1µF)	Yes	Yes
MAX207E	5	3	0	4 (0.1µF)	No	No
MAX208E	4	4	0	4 (0.1µF)	No	No
MAX211E	4	5	0	4 (0.1µF)	Yes	Yes
MAX213E	4	5	2	4 (0.1µF)	Yes	Yes
MAX232E	2	2	0	4 (1µF)	No	No
MAX241E	4	5	0	4 (1µF)	Yes	Yes

LapLink is a registered trademark of Traveling Software, Inc.

±15kV ESD-Protected, +5V RS-232 Transceivers

ABSOLUTE MAXIMUM RATINGS

V_{CC} .... -0.3V to +6V
V+ .... (V_{CC} - 0.3V) to +14V
V- .... -14V to +0.3V
Input Voltages
T_IN .... -0.3V to (V+ + 0.3V)
R_IN .... ±30V
Output Voltages
T_OUT .... (V- - 0.3V) to (V+ + 0.3V)
R_OUT .... -0.3V to (V_{CC} + 0.3V)
Short-Circuit Duration, T_OUT .... Continuous
Continuous Power Dissipation (T_A = +70°C)
16-Pin Plastic DIP (derate 10.53mW/°C above +70°C) 842mW
16-Pin Narrow SO (derate 8.70mW/°C above +70°C) 696mW
16-Pin Wide SO (derate 9.52mW/°C above +70°C) 762mW
20-Pin Plastic DIP (derate 11.11mW/°C above +70°C) ... 889mW
20-Pin SO (derate 10.00mW/°C above +70°C) 800mW
24-Pin Narrow Plastic DIP
(derate 13.33mW/°C above +70°C) 1.07W
24-Pin Wide Plastic DIP
(derate 14.29mW/°C above +70°C) 1.14W
24-Pin SO (derate 11.76mW/°C above +70°C) 941mW
24-Pin SSOP (derate 8.00mW/°C above +70°C) 640mW
28-Pin SO (derate 12.50mW/°C above +70°C) 1W
28-Pin SSOP (derate 9.52mW/°C above +70°C) 762mW
Operating Temperature Ranges
MAX2_ _EC_ _ .... 0°C to +70°C
MAX2_ _EE_ _ .... -40°C to +85°C
Storage Temperature Range .... -65°C to +165°C
Lead Temperature (soldering, 10sec) .... +300°C

Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of the specifications is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

ELECTRICAL CHARACTERISTICS

(V_{CC} = +5V ±10% for MAX202E/206E/208E/211E/213E/232E/241E; V_{CC} = +5V ±5% for MAX203E/205E/207E; C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E; C1–C4 = 1µF for MAX232E/241E; T_A = T_{MIN} to T_{MAX}; unless otherwise noted. Typical values are at T_A = +25°C.)

PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
DC CHARACTERISTICS							
V_{CC} Supply Current	I_{CC}	No load, T_A = +25°C	MAX202E/203E		8	15	mA
			MAX205E–208E		11	20	
			MAX211E/213E		14	20	
			MAX232E		5	10	
			MAX241E		7	15	
Shutdown Supply Current		T_A = +25°C, Figure 1	MAX205E/206E		1	10	µA
			MAX211E/241E		1	10	
			MAX213E		15	50	
LOGIC							
Input Pull-Up Current		T_IN = 0V (MAX205E–208E/211E/213E/241E)			15	200	µA
Input Leakage Current		T_IN = 0V to V_{CC} (MAX202E/203E/232E)				±10	µA
Input Threshold Low	V_{IL}	T_IN; EN, $\overline{SHDN}$ (MAX213E) or $\overline{EN}$, SHDN (MAX205E–208E/211E/241E)				0.8	V
Input Threshold High	V_{IH}	T_IN		2.0			V
		EN, $\overline{SHDN}$ (MAX213E) or $\overline{EN}$, SHDN (MAX205E–208E/211E/241E)		2.4			
Output Voltage Low	V_{OL}	R_OUT; I_{OUT} = 3.2mA (MAX202E/203E/232E) or I_{OUT} = 1.6mA (MAX205E/208E/211E/213E/241E)				0.4	V
Output Voltage High	V_{OH}	R_OUT; I_{OUT} = -1.0mA		3.5	V_{CC} - 0.4		V
Output Leakage Current		$\overline{EN}$ = V_{CC}, EN = 0V, 0V ≤ R_{OUT} ≤ V_{CC}, MAX205E–208E/211E/213E/241E outputs disabled			±0.05	±10	µA

ELECTRICAL CHARACTERISTICS (continued)

(V_{CC} = +5V ±10% for MAX202E/206E/208E/211E/213E/232E/241E; V_{CC} = +5V ±5% for MAX203E/205E/207E; C1–C4 = 0.1µF for MAX202E/206E/207E/208E/211E/213E; C1–C4 = 1µF for MAX232E/241E; T_A = T_{MIN} to T_{MAX}; unless otherwise noted. Typical values are at T_A = +25°C.)

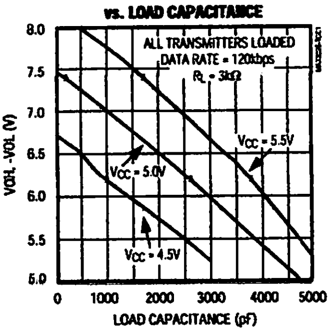
PARAMETER	SYMBOL	CONDITIONS		MIN	TYP	MAX	UNITS
EIA/TIA-232E RECEIVER INPUTS							
Input Voltage Range				-30		30	V
Input Threshold Low		T_A = +25°C, V_{CC} = 5V	All parts, normal operation	0.8	1.2		V
			MAX213E, $\overline{SHDN}$ = 0V, EN = V_{CC}	0.6	1.5		
Input Threshold High		T_A = +25°C, V_{CC} = 5V	All parts, normal operation		1.7	2.4	V
			MAX213E (R4, R5), $\overline{SHDN}$ = 0V, EN = V_{CC}		1.5	2.4	
Input Hysteresis		V_{CC} = 5V, no hysteresis in shutdown		0.2	0.5	1.0	V
Input Resistance		T_A = +25°C, V_{CC} = 5V		3	5	7	kΩ
EIA/TIA-232E TRANSMITTER OUTPUTS							
Output Voltage Swing		All drivers loaded with 3kΩ to ground (Note 1)		±5	±9		V
Output Resistance		V_{CC} = V+ = V- = 0V, V_{OUT} = ±2V		300			Ω
Output Short-Circuit Current					±10	±60	mA
TIMING CHARACTERISTICS							
Maximum Data Rate		R_L = 3kΩ to 7kΩ, C_L = 50pF to 1000pF, one transmitter switching		120			kbps
Receiver Propagation Delay	t_{PLHR}, t_{PHLR}	C_L = 150pF	All parts, normal operation		0.5	10	µs
			MAX213E (R4, R5), $\overline{SHDN}$ = 0V, EN = V_{CC}		4	40	
Receiver Output Enable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			600		ns
Receiver Output Disable Time		MAX205E/206E/211E/213E/241E normal operation, Figure 2			200		ns
Transmitter Propagation Delay	t_{PLHT}, t_{PHLT}	R_L = 3kΩ, C_L = 2500pF, all transmitters loaded			2		µs
Transition-Region Slew Rate		T_A = +25°C, V_{CC} = 5V, R_L = 3kΩ to 7kΩ, C_L = 50pF to 1000pF, measured from -3V to +3V or +3V to -3V, Figure 3		3	6	30	V/µs
ESD PERFORMANCE: TRANSMITTER OUTPUTS, RECEIVER INPUTS							
ESD-Protection Voltage		Human Body Model			±15		kV
		IEC1000-4-2, Contact Discharge			±8		
		IEC1000-4-2, Air-Gap Discharge			±15		

Note 1: MAX211EE_ _ tested with V_{CC} = +5V ±5%.

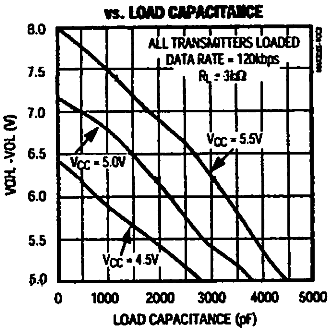
Typical Operating Characteristics

(Typical Operating Circuits, V_{CC} = +5V, T_A = +25°C, unless otherwise noted.)

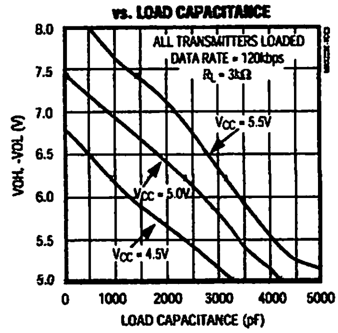
MAX232E TRANSMITTER OUTPUT VOLTAGE vs. LOAD CAPACITANCE



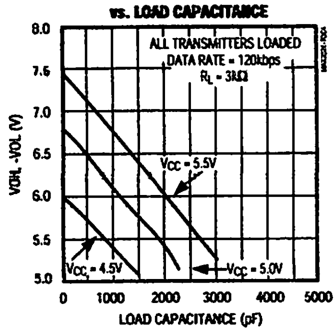
MAX202E/MAX203E TRANSMITTER OUTPUT VOLTAGE vs. LOAD CAPACITANCE



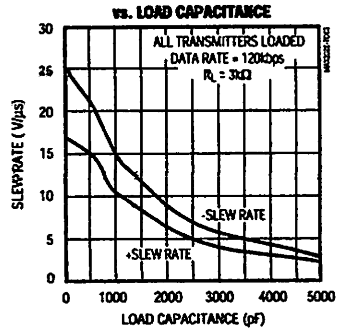
MAX241E TRANSMITTER OUTPUT VOLTAGE vs. LOAD CAPACITANCE



MAX211E/MAX213E TRANSMITTER OUTPUT VOLTAGE vs. LOAD CAPACITANCE

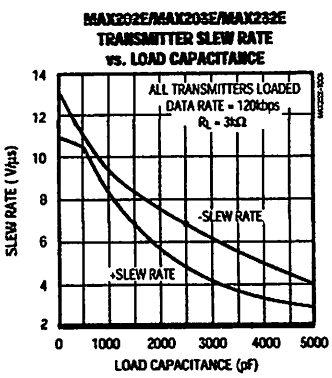


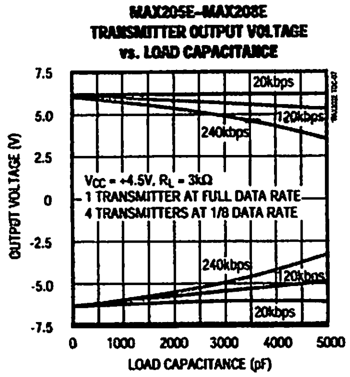
MAX211E/MAX213E/MAX241E TRANSMITTER SLEW RATE vs. LOAD CAPACITANCE

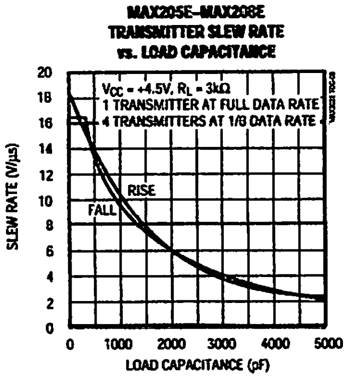


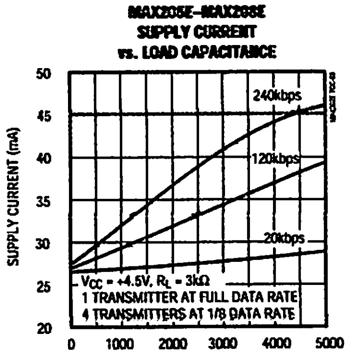
Typical Operating Characteristics (continued)

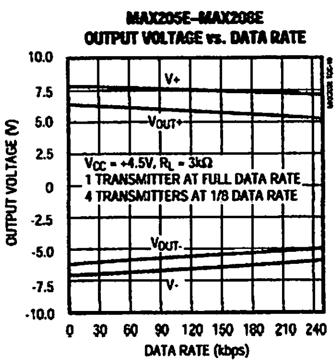
(Typical Operating Circuits, V_{CC} = +5V, T_A = +25°C, unless otherwise noted.)











Pin Descriptions

MAX202E/MAX232E

PIN		NAME	FUNCTION
DIP/SO	LCC		
1, 3	2, 4	C1+, C1-	Terminals for positive charge-pump capacitor
2	3	V+	+2V_{CC} voltage generated by the charge pump
4, 5	5, 7	C2+, C2-	Terminals for negative charge-pump capacitor
6	8	V-	-2V_{CC} voltage generated by the charge pump
7, 14	9, 18	T_OUT	RS-232 Driver Outputs
8, 13	10, 17	R_IN	RS-232 Receiver Outputs
9, 12	12, 15	R_OUT	RS-232 Receiver Outputs
10, 11	13, 14	T_IN	RS-232 Driver Inputs
15	19	GND	Ground
16	20	V_{CC}	+4.5V to +5.5V Supply-Voltage Input
—	1, 6, 11, 16	N.C.	No Connect—not internally connected.

MAX203E

PIN		NAME	FUNCTION
DIP	SO		
1, 2	1, 2	T_IN	RS-232 Driver Inputs
3, 20	3, 20	R_OUT	RS-232 Receiver Outputs
4,19	4, 19	R_IN	RS-232 Receiver Inputs
5,18	5, 18	T_OUT	RS-232 Transmitter Outputs
6, 9	6, 9	GND	Ground
7	7	V_{CC}	+4.5V to +5.5V Supply-Voltage Input
8	13	C1+	Make no connection to this pin.
10, 16	11, 16	C2-	Connect pins together.
12, 17	10, 17	V-	-2V_{CC} voltage generated by the charge pump. Connect pins together.
13	14	C1-	Make no connection to this pin.
14	8	V+	+2V_{CC} voltage generated by the charge pump
11, 15	12, 15	C2+	Connect pins together.

MAX205E

PIN	NAME	FUNCTION
1–4, 19	T_OUT	RS-232 Driver Outputs
5, 10, 13, 18, 24	R_IN	RS-232 Receiver Inputs
6, 9, 14, 17, 23	R_OUT	TTL/CMOS Receiver Outputs. All receivers are inactive in shutdown.
7, 8, 15, 16, 22	T_IN	TTL/CMOS Driver Inputs. Internal pull-ups to V_{CC}.
11	GND	Ground
12	V_{CC}	+4.75V to +5.25V Supply Voltage
20	$\overline{EN}$	Receiver Enable—active low
21	SHDN	Shutdown Control—active high