

3100098010142



**PERENCANAAN DAN PEMBUATAN  
SISTEM TELETEXT UNTUK PC  
DENGAN MENGGUNAKAN  
SISTEM KOMUNIKASI RADIO SIMPLEX**

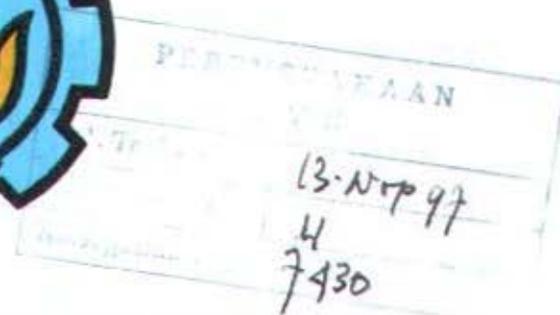
**TUGAS AKHIR**

**Disusun oleh :**

**TJIPTO PRAKOSA**

**NRP : 2291 100 031**

RSE  
621.398.1  
Pra  
p-1  
1997



**JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNOLOGI INDUSTRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER  
SURABAYA**

**1997**

**PERENCANAAN DAN PEMBUATAN  
SISTEM TELETEXT UNTUK PC  
DENGAN MENGGUNAKAN  
SISTEM KOMUNIKASI RADIO SIMPLEX**

**TUGAS AKHIR**

**Diajukan Guna Memenuhi Sebagian Persyaratan  
Untuk Memperoleh Gelar Sarjana Teknik Elektro  
Pada  
Bidang Studi Teknik Sistem Komputer  
Jurusan Teknik Elektro Fakultas Teknologi Industri  
Institut Teknologi Sepuluh Nopember Surabaya**

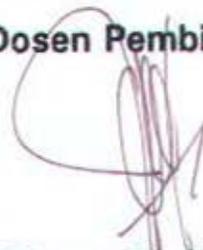
**Mengetahui/Menyetujui**

**Dosen Pembimbing I**



**Ir. Zainal Alim  
NIP. 131 532 037**

**Dosen Pembimbing II**



**Ir. Hanny Budinugroho  
NIP. 131 651 433**

**SURABAYA  
Oktober, 1997**

## ABSTRAK

Teknologi informasi yang sungguh pesat ini membuat sistem jaringan komputer pribadi PC semakin maju. Namun kemajuan ini harus dibayar mahal dengan pengadaan media komunikasi dan perlengkapannya.

Walaupun metode komunikasi searah (simplex) mempunyai beberapa kekurangan namun sebenarnya mempunyai potensial yang besar untuk dikembangkan sebagai sistem komunikasi data alternatif yang cukup murah sehingga kemajuan informasi dapat dinikmati oleh banyak orang.

Sistem komunikasi ini dirancang untuk dapat memancarkan (broadcast) data seperti layaknya pemancar radio niaga memancarkan sinyal audio. Sehingga data dapat dipancarkan ke banyak komputer penerima yang sedang menghidupkan penerimanya.

## **PRAKATA**

Atas berkat rahmat Allah S.W.T., maka penyusun dapat menyelesaikan Tugas Akhir yang berjudul :

### **PERENCANAAN DAN PEMBUATAN**

### **SISTEM TELETEX UNTUK PC**

### **DENGAN MENGGUNAKAN SISTEM KOMUNIKASI RADIO SIMPLEX**

Tugas akhir ini merupakan salah satu syarat yang harus ditempuh untuk meraih gelar kesarjanaan di Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh November Surabaya.

Dalam mengerjakan Tugas Akhir ini penulis banyak mendapat bantuan dan bimbingan dari berbagai pihak. Pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada :

- Bapak Ir. Zainal Alim, selaku Dosen Pembimbing I yang telah memberikan bimbingan dan pengarahan serta saran sehingga penulis dapat menyelesaikan Tugas Akhir ini.
- Bapak Ir. Hanny Budhinugroho, selaku Dosen II yang telah memberikan pengarahan dan semangat demi penyelesaian Tugas Akhir ini.
- Bapak Ir. Yoyon Kusnendar Suprpto, M.Sc, selaku Koordinator Bidang Studi Komputer, Jurusan Teknik Elektro, FTI ITS. yang ikut memberikan petunjuk dan arahnya dalam proses pembuatan Tugas Akhir.

- Para Bapak Dosen : Hariyadi ST, Eko M ST, Suryo S ST. yang ikut memberikan saran dan masukan yang positif.
- Rekan-rekan di LAB B201 : Kampress, Synam, Thekoez, Riss bin Mell, Asua, Ilad yang telah banyak memberi *Technical Supports* dan *Help Online*. Juga Litthel, Kaler, Inyong, Gendul, Kumbamz, FStone, Goang'k, Joko K, yang memberi banyak memberi semangat juang . Dan Semua Kru B201 yang belum tersebut.
- Buat Teman yang akan menemani Hidupku selanjutnya: Tri Kurniastuti yang telah memberi banyak warna dan arti hidupku yang baru.

Akhir kata, penyusun berharap semoga segala sesuatu yang telah dilaksanakan dalam Tugas Akhir ini dapat memberi manfaat bagi kita semua.

Surabaya, September 1997

Penyusun

Tjipto Prakosa

# DAFTAR ISI

ABSTRAK .....	i
PRAKATA .....	ii
DAFTAR ISI .....	iv
DAFTAR GAMBAR .....	vi
DAFTAR TABEL .....	vii
BAB I PENDAHULUAN .....	1
1.1 Latar Belakang .....	1
1.2 Permasalahan .....	2
1.3 Tujuan .....	2
1.4 Metodologi .....	2
1.5 Batasan Masalah .....	3
1.6 Sistematika Penulisan .....	3
BAB II TEORI PENUNJANG .....	5
7.1 Dasar Transmisi Data .....	5
7.1.1 Karakter Standart .....	5
7.1.2 Mode Arah Komunikasi .....	6
7.1.3 Komunikasi Data Serial .....	7
7.2 RS 232C .....	10
7.2.1 Konektor Serial Standart .....	10
7.2.2 Tegangan Logika .....	13
7.3 Modem .....	13
7.3.1 Modulasi Frekuensi .....	14
7.3.2 TCM 3105 .....	15
7.4 Protokol .....	18
7.4.1 Paket Data .....	18
7.4.2 Error Check .....	21
- Parity Redudancy Karakter .....	22
- Parity Redudancy Blok .....	23
- Checksum Redudancy Blok .....	24
- Cyclic Redudancy Check (CRC) .....	25
7.5 Pemrograman Windows 95 .....	30

7.5.1 Deklarasi Variabel .....	31
7.5.2 Struktur Kontrol .....	33
7.5.3 Struktur Perulangan (Loop) .....	33
7.5.4 Kontrol Serial Port .....	34
7.6 Data Dan Penampilan Data .....	36
7.7 Media Komunikasi .....	38
7.7.1 Pemancar FM .....	39
7.7.2 Radio Penerima FM .....	39
<b>BAB III PERENCANAAN DAN PEMBUATAN</b>	
<b>PERANGKAT KERAS .....</b>	<b>41</b>
8.1 Diagram Blok .....	41
8.2 Modem .....	42
8.3 Kalibrasi Modem .....	45
<b>BAB IV PERENCANAAN DAN PEMBUATAN</b>	
<b>PERANGKAT LUNAK .....</b>	<b>47</b>
9.1 Protokol Komunikasi .....	47
9.1.1 Paket .....	47
9.1.2 Cyclic Redudancy Check .....	49
9.2 Bagian Pengirim .....	52
9.3 Bagian Penerima .....	54
9.4 Contoh Program Aplikasi Database .....	58
9.5 Struktur Penampilan Html .....	59
<b>BAB V PEMASANGAN ALAT DAN HASIL AKHIR .....</b>	<b>61</b>
10.1 Pemasangan Modem .....	61
10.2 Pemasangan Software .....	61
10.2.1 Pemasangan Bagian Penerima .....	61
10.2.2 Pemasangan Bagian Pengirim .....	64
10.3 Pengukuran Jangkauan Pemancar .....	64
<b>BAB VI PENUTUP .....</b>	<b>67</b>
<b>DAFTAR PUSTAKA .....</b>	
<b>LAMPIRAN .....</b>	

## DAFTAR GAMBAR

Gambar 2.1 Mode Arah Komunikasi .....	7
Gambar 2.2 Contoh Data Asinkron 8E1 .....	10
Gambar 2.3 Tegangan Logika Rs 232 .....	13
Gambar 2.4 Pembagian Sistem TCM 3105 .....	15
Gambar 2.5 Blok Diagram Fungsional TCM 3105 .....	16
Gambar 2.6 Daerah Frekuensi Standar Bell 202 .....	18
Gambar 2.7 Paket Data Dengan Pembatas Karakter Kontrol .....	19
Gambar 2.8 Paket Dengan Field Penanda Panjang .....	20
Gambar 2.9 Paket Dengan Panjang Tetap (Xmodem) .....	21
Gambar 2.10 Tampilan Dari File HTML Pada MSIE 3.0 .....	38
Gambar 2.11 Diagram Blok Pemancar .....	39
Gambar 2.12 Diagram Radio Penerima Fm .....	40
Gambar 3.1 Blok Diagram Unit Pengirim .....	42
Gambar 3.2 Blok Diagram Bagian Penerima .....	42
Gambar 3.3 Rangkaian Modem .....	43
Gambar 3.4 Rangkaian Catu Daya .....	43
Gambar 3.5 Kalibrasi Modem .....	46
Gambar 4.1 Struktur Paket Data .....	48
Gambar 4.2 Urutan Paket .....	49
Gambar 4.3 Diagram Alir Program CRC .....	50
Gambar 4.4 Penambahan CRC Di Pengirim .....	52
Gambar 4.5 CRC Test Di Penerima .....	52
Gambar 4.6 Hubungan Antar Program Dan File .....	53
Gambar 4.7 Contoh Pengambilan Mode Frame .....	55
Gambar 4.8 Diagram Blok Penerimaan Menurut Mode Framenya .....	57
Gambar 4.9 Gambaran Lengkap Contoh Aplikasi .....	58
Gambar 5.1 Program Setting .....	62
Gambar 5.2 Diagram Pemasangan Pada Penerima .....	63
Gambar 5.3 Diagram Pemasangan Pada Pemancar .....	64

## DAFTAR TABEL

Tabel 2.1 Mode Parity Bit .....	9
Tabel 2.2 Fungsi Kaki Pin RS 232 .....	12
Tabel 2.3 Tabel Fungsi Dari Kaki Pin TCM 3105 .....	17
Tabel 2.4 Tipe Data Integer .....	31
Table 2.5 Tipe Data Real .....	32
Tabel 3.1 Perbedaan Tegangan Input Dan Output Antara RS 232 Dengan TCM 3105 .....	44
Tabel 4.1 Keterangan Isi Paket Header .....	48
Tabel 4.2 Tabel Mode Frame Dan Operasi Yang Dilakukan .....	56

# BAB I

## PENDAHULUAN

---

### 1.1 LATAR BELAKANG

Perkembangan teknologi informasi yang sangat pesat telah memberikan manfaat yang sangat besar bagi kehidupan kita. Hal ini telah kita rasakan dengan adanya era informasi jaringan komputer yang menglobal dan cepat seperti Internet, sehingga informasi yang kita inginkan bisa didapatkan dengan mudah dan langsung ke sumber informasi yang diinginkan. Namun perkembangan teknologi informasi ini masih dirasa sangat mahal bagi sebagian besar masyarakat, hal ini dikarenakan untuk berkoneksi dengan jaringan tersebut dibutuhkan media komunikasi yang mahal seperti line telepon atau perangkat komunikasi lain.

Untuk lebih memasyarakatkan teknologi informasi perlu dipikirkan alternatif sistem pengiriman data lain yang lebih murah dan efisien, mengingat dewasa ini komputer bukan lagi menjadi barang 'langka' bagi sebagian besar masyarakat. Salah satu alternatif yaitu komunikasi dengan sistem simplex (satu arah)/ broadcasting, walaupun beberapa aspek dalam komunikasi hilang seperti umpan balik informasi data namun dengan sistem komunikasi simplex ini aplikasi-aplikasi seperti koran elektronik, teletex dan informasi publik dapat ditampilkan lebih murah, cepat dan efisien.

Dengan melihat manfaat yang didapatkan seperti teletext pada TV maka kami mencoba memanfaatkan komputer untuk dapat menerima komunikasi data simplex

seperti pada sistem teletex televisi dengan tampilan yang tentu lebih baik dengan menambah perlengkapan modem dan radio penerima FM sebagai sarana / media komunikasi alternatif yang lebih murah, sehingga diharapkan akan menjadi model aplikasi koran elektronik yang akan menghemat kertas dan biaya pengirimannya.

## **1.2 PERMASALAHAN**

Untuk bisa mewujudkan hal tersebut diatas dibutuhkan perlengkapan dan suatu sistem komunikasi (protokol) dengan deteksi kesalahan yang handal untuk menangani sistem transmisi data dengan metode simplex sehingga data dapat terkirim dengan baik.

## **1.3 TUJUAN**

Membuat perlengkapan modem dan protokol komunikasi sehingga mampu menjadikan komunikasi simplex ini sebagai alternatif baru komunikasi data dan merancang aplikasi koran elektronik sebagai model pemanfaatannya.

## **1.4 METODOLOGI**

Langkah awal dalam membuat alat ini dengan mempelajari sistem protokol komunikasi data sehingga data dapat dikirimkan dalam model paket-paket dan kemudian ditranmisikan secara serial melalui RS 232 menuju modem FSK dan oleh modem sinyal serial dirubah menjadi isyarat analog yang ditranmisikan dengan pemancar FM sebagai media komunikasi. Setelah diterima oleh radio penerima FM sinyal tersebut diproses oleh demodulator dan dikirim lagi melalui RS 232 dan data paket tersebut diolah sehingga menjadi data semula.

## 1.5 BATASAN MASALAH

Membuat aplikasi dengan :

1. Sistem Operasi Windows 95
2. Mengirimkan file html untuk memperbaiki tampilan dan ditampilkan dengan browser internet yang sudah ada.
3. Pembuatan modem dengan menggunakan modul IC TCM 3105
4. Menggunakan pemancar dan penerima FM untuk model transmisinya.
5. Membuat model protokol komunikasi satu arah.
6. Menggunakan metode cek error CRC standar CCITT

## 1.6 SISTEMATIKA PENULISAN

Sistematika penulisan tugas akhir ini dibagi dalam beberapa bab, yaitu :

- BAB I : Pendahuluan, bab ini membahas tentang latar belakang permasalahan, tujuan dan sistematika penulisan
- BAB II : Teori Penunjang bab ini membahas tentang sistem komunikasi data dan sistem protokol
- BAB III : Perencanaan perangkat keras, bab ini membahas tentang pembuatan modem
- BAB IV : Perencanaan perangkat lunak, pada bab ini dibahas pembuatan program untuk sistem protokol pengiriman dan penerimaan

- BAB V : Pengujian alat, bab ini menguraikan tentang pengujian alat dan pemakaian softwarena
- BAB VI : Penutup, bab ini berisi kesimpulan dan saran-saran yang diperlukan untuk menyempurnakan alat tersebut.

## BAB II

### TEORI PENUNJANG

---

#### 2.1 DASAR TRANSMISI DATA

##### 2.1.1 Karakter Standart

Dalam memasukan data ke komputer melalui keyboard, tiap tombol yang mewakili karakter alfabetik atau karakter numerik dikodekan oleh perangkat keyboard dalam bentuk kode biner yang sesuai dengan kode standart untuk pertukaran informasi. Untuk mengkodekan semua huruf pada keyboard secara unik, dipakai 7 atau 8 digit. Jika menggunakan 7 bit berarti ada 128 macam karakter sedangkan untuk 8 bit berarti ada 256 macam karakter. Pola bit kode dari setiap karakter disebut *codewords*.

Dua kode standar yang umum dipakai adalah Extended Binary Coded Desimal Interchange Code (EBDIC) dan American Standards Committee for Information Interchange (ASCII). Kode EBDIC menggunakan 8 bit dan biasa diterapkan dalam peralatan yang diproduksi oleh IBM.

Kode ASCII adalah nama yang diberikan oleh CCITT dikenal dengan International Alphabet Number 5 atau IA5 dan juga dipakai oleh Standard Organization known as ISO 645.

Kedua sistem kode tersebut mewakili semua karakter alfabetik normal, numerik dan karakter tanda baca yang digolongkan dalam karakter cetak (printable

characters) ditambah dengan karakter kontrol tambahan dan digolongkan dalam karakter yang tidak tercetak (non-printable characters). Contoh dari kontrol karakter adalah ; karakter kontrol format - back space (BS), line feed (LF). Separator informasi - file separator (FS) dan record separator (RS). Karakter kontrol untuk transmisi - start of heading (SOH), start of text (STX), end of text (ETX), acknowledge (ACK), negative acknowledge (NAK) dan synchronous idle (SYN).

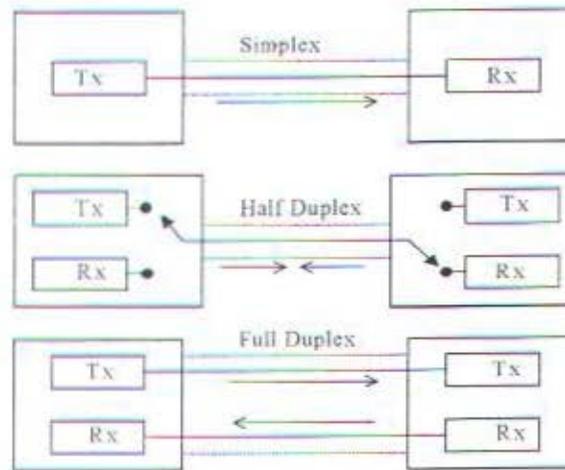
Dalam pemakaiannya data dalam komputer memakai satuan byte atau kelipatannya maka untuk dapat mentransmisikan semua data pada tugas akhir ini dipakai format 8 bit.

### 2.1.2 Mode Arah Komunikasi

Dalam komunikasi dikenal ada beberapa mode komunikasi data yang dapat dipakai :

1. Simplex : digunakan pada pengiriman data satu arah, misalnya sistem monitoring (telemetry) yang mengirim data ke stasiun kontrol secara kontinyu.
2. Half-Duplex : dapat mengirimkan data hanya satu arah setiap saat, tetapi arah komunikasi dapat diubah, misalnya pesawat komunikasi amatir yang bergantian dalam berbicara
3. Full Duplex / Duplex : dapat mengirimkan data dalam dua arah setiap saat, contoh seperti pada pesawat telepon dimana dapat dilakukn pembicaraan dan mendengar secara bersamaan

Gambaran fisik dari mode arah komunikasi dapat dilihat pada Gambar 2.1



Gambar 2.1 Mode arah komunikasi

Sebenarnya ada perbedaan antara operasi full duplex dengan operasi duplex, perbedaannya terletak pada kecepatan transmisi data pada kedua arah tidak sama pada operasi duplex. Hal tersebut sering diterapkan pada modem dewasa ini karena untuk meningkatkan performance kecepatan modem yang diakibatkan keterbatasan bandwidth line telepon.

### 2.1.3 Komunikasi Data Serial

Komunikasi data serial adalah mengirimkan data secara berurutan bit demi bit sehingga kecepatan lebih lambat dari komunikasi data paralel yang mengirimkan data bersamaan (misalnya 8 bit sekali kirim). Komunikasi data serial ini sangat cocok untuk sistem komunikasi jarak jauh karena lebih sedikit membutuhkan media (kabel) dalam transmisinya dibandingkan dengan sistem paralel.

Pada pengiriman data tak sinkron, setiap karakter dikirimkan sebagai satu kesatuan bebas, yang berarti bahwa waktu pengiriman bit terakhir dari sebuah karakter dengan bit pertama sebuah karakter berikutnya tidak selalu tetap.

Efisiensi sistem asinkron tidak begitu tinggi, karena hanya 8 bit dari 11 bit yang dikirimkan berisi informasi. Kecepatan transmisi dalam saluran biasanya diekspresikan dalam baud rate. Sebenarnya istilah baud menunjukkan jumlah sinyal yang ditransmisikan dinyatakan dalam keadaan biner. Untuk menunjukkan kecepatan informasi dipakai istilah bit per second (bps).

Format pengiriman asinkron pada umumnya diawali dengan start bit kemudian data bit dan diakhiri dengan parity dan stop bit.

- Start bit

Pada sistem komputer atau suatu peralatan penerima karakter asinkron yang dibaca adalah logika '0' dan logika '1', pada jalur received data akan memberikan kondisi awal logika '0' (start bit) selama bit time yang sudah ditentukan bila data word akan dikirimkan. Dengan adanya perubahan kondisi '1' ke '0', komputer akan menafsirkan bahwa suatu word data akan dikirimkan.

- Data bit

Proses transmisi harus mengikuti suatu aturan tertentu. Salah satu harus diperhatikan adalah kode karakter yang dipakai dalam bit data. Saat ini ada beberapa kode yang cukup populer antara lain : kode 5 bit

untuk Baudot Murray, kode 6 bit untuk IBM correspondence, kode 7 bit untuk ASCII dan kode 8 bit untuk EBCDIC.

Pada interface RS 232 jumlah data bit tergantung pada format data yang ditentukan oleh komputer jadi dapat diatur sesuai dengan kebutuhan yang ada.

#### - Parity bit

Setelah data bit akan diikuti dengan parity bit, yang digunakan untuk mendeteksi kesalahan yang mungkin terjadi pada saat proses pengiriman. Nilai dari bit parity tergantung pada mode dan data bitnya. Mode parity bit harus sama pada kedua terminal yaitu bila pada pengirim menggunakan bit check *even* maka pada penerima harus bit check *even* pula untuk lebih jelasnya lihat Tabel 2.1.

Tabel 2.1 Mode Parity Bit

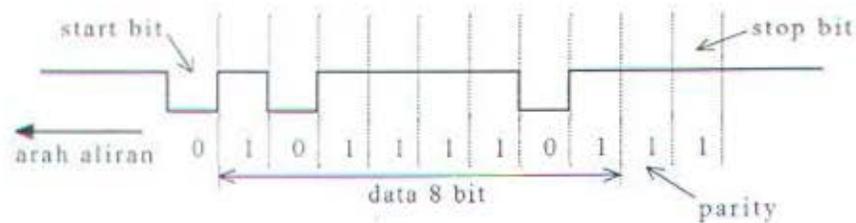
Mode	Keterangan
None (N)	Tanpa bit parity
Odd (O)	Bernilai '0' bila data bernilai ganjil
Even (E)	Bernilai '0' bila data bernilai genap
Mark (M)	Selalu bernilai '1'
Space (S)	Selalu bernilai '0'

#### - Stop bit

Bit yang terakhir adalah stop bit, bit ini menginformasikan akhir dari satu karakter data. Stop bit ini bernilai sama dengan '1' dan kondisi idle

pun akan tetap bernilai '1'. Pada umumnya jumlah stop bit ada dua macam yaitu 1 dan 2 yang menyatakan jumlah bit stopnya dalam satu frame bit data tersebut.

Contoh gambar dari format data asinkron dapat dilihat pada gambar 2.2.



Gambar 2.2 Contoh data Asinkron 8E1

## 2.2 RS 232C

Salah satu standart serial interface yang sering digunakan adalah RS232C. RS232C adalah interface standart Electric Industry Association (EIA). Prinsip kerja dari RS232C adalah mengubah data menjadi data serial standart dan juga sebaliknya. Metode pengiriman sinyal yang dipakai RS232C adalah metode asinkron.

### 2.2.1 Konektor Serial Standart

Untuk kebutuhan serial interface, RS 232 C memanfaatkan sebuah konektor dengan jumlah pin sebanyak 25. Konektor ini sering disebut sebagai DB-25 Connector. Pengendali dari interface tersebut adalah UART(Universal Asynchronous Receiver / Transmitter) yang dihubungkan dengan RS 232 C . Secara praktis untuk kebutuhan transfer data / komunikasi data cukup 9 pin yang digunakan. Adapun guna masing-masing pin adalah sebagai berikut :

- Protective Ground : pin 1

Pin ini berguna untuk menghindari kejutan karakteristik listrik karena kegagalan suatu catu daya. Dalam kasus tertentu pemakaian pin 1 ini bukan suatu keharusan

- Transmitted Data : pin 2. (TxD / TD)

Berguna sebagai output data serial dari Komputer (DTE) ke Modem (DCE).

- Received Data : pin 3 (RxD / RD)

Berguna sebagai input data yang dikirimkan oleh DCE masuk ke DTE

- Request To Send : pin 4 (RTS)

Berguna untuk memberitahukan DCE bahwa DTE akan mengirim data. RTS merupakan sebuah protokol perangkat keras yang mendahului pengiriman dari DTE ke DCE.

- Clear To Send : pin 5 (CTS)

Berguna untuk memberitahukan DTE bahwa DCE siap untuk menerima data. CTS merupakan sebuah protokol perangkat keras yang mendahului pengiriman dari DTE ke DCE.

- Data Set Ready : pin 6 (DSR)

Berguna untuk memberitahukan DTE bahwa DCE telah aktif dan siap bekerja.

- Signal Ground : pin 7 (G)

Berguna sebagai referensi semua tegangan interface

- Data Carrier Detect : pin 8 (DCD)

Berguna untuk memberi tahu DTE bahwa komunikasi sudah dibentuk sehingga siap mengirim dan menerima data.

- Data Terminal Ready : pin 20 (DTR)

Berguna untuk memberitahu DCE bahwa DTE telah aktif dan siap untuk bekerja.

- Ring Indicator : pin 22 (RI)

Berguna memberi informasi masuknya telepon.

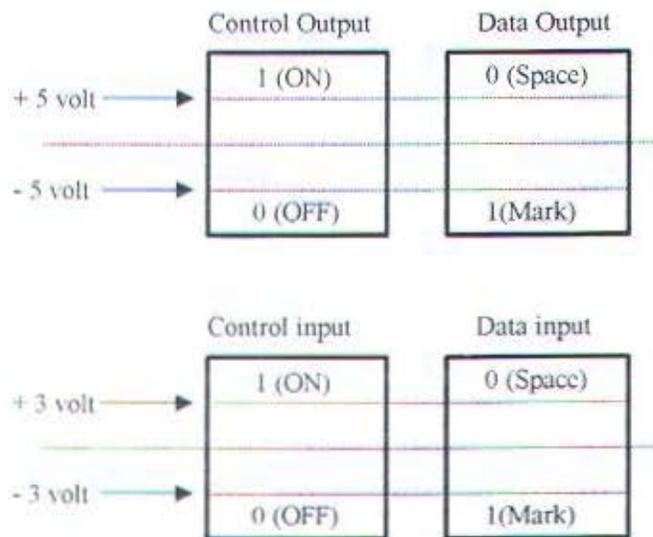
Untuk lebih jelasnya sinyal pada konektor RS 232 baik dengan DB 9 maupun DB 25 dapat dilihat pada Tabel 2.2 berikut :

Tabel 2.2 Fungsi kaki pin RS 232

9 pin	25 pin	Keterangan	IN / OUT
1	8	Carrier Detect	IN
2	3	Received Data	IN
3	2	Transmitted Data	OUT
4	20	Data Terminal Ready	OUT
5	7	Sinyal Ground	-
6	6	Data Set Ready	IN
7	4	Request To Send	OUT
8	5	Clear To Send	IN
9	22	Ring Indicator	IN

### 2.2.2 Tegangan Logika

Tegangan input data pada RS 232C berbeda dengan tegangan tegangan logika digital. Logika '0' pada sinyal digital (space menurut istilah RS-232 C) mempunyai nilai tegangan +3 sampai +15 Volt sedangkan logika '1' (mark) mempunyai nilai tegangan -3 sampai -15 Volt. Khusus tegangan output data dari RS 232C sekitar +5 sampai +15 volt atau -5 volt sampai -15 volt. Jadi bila tegangan berada pada +3 sampai -3 disebut tegangan invalid. Besar tegangan dari input atau ouput dapat dilihat pada Gambar 2.3.



Gambar 2.3 Tegangan logika RS 232

### 2.3 MODEM

Modem adalah singkatan dari Modulasi dan Demodulasi, yaitu suatu alat yang berfungsi memodulasikan sinyal digital pada sinyal analog, tujuan dari pemodulasian agar sinyal dapat ditranmisikan lebih jauh dengan bantuan sinyal carrier.

Modem pada umumnya banyak digunakan pada sistem line telepon dimana bandwidthnya dibatasi pada 300 Hz - 3400 Hz maka sinyal carrier juga harus disesuaikan dengan bandwidth tersebut.

Beberapa jenis sistem modulasi yang umum digunakan adalah :

- Modulasi Amplitudo
- Modulasi Sudut

Modulasi sudut dibagi menjadi dua jenis modulasi, yaitu modulasi frekuensi dan modulasi fasa. Pada tugas akhir ini digunakan modulasi sudut khususnya modulasi frekuensi. Oleh karena itu pembahasan akan ditekankan pada modulasi frekuensi.

### 2.3.1 Modulasi Frekuensi

Modulasi berarti mengatur suatu parameter dari gelombang pembawa frekuensi tinggi dengan pertolongan sinyal informasi yang frekuensinya lebih rendah. Gelombang pembawa berbentuk gelombang sinus sedangkan sinyal informasi dapat berupa gelombang sinus, kotak atau segitiga.

Pada sistem transmisi data yang memanfaatkan modulasi frekuensi, digunakan dua buah frekuensi untuk sinyal informasinya, sistem ini disebut *Frequency Shift Keying* (FSK). Metode ini hanya cocok pada modem dengan kecepatan rendah (300 - 1200 baud).

Pada prinsipnya FSK adalah mengubah logika biner menjadi sinyal dengan frekuensi tertentu, tetapi tidak mengalami perubahan pada sudut fasa dan amplitudonya. Sebagai contoh untuk logika '0' biner (mark) akan dihasilkan sinyal

dengan frekuensi 1300 Hz sedangkan untuk logika '1' biner (space) akan dihasilkan sinyal frekuensi 2100 Hz.

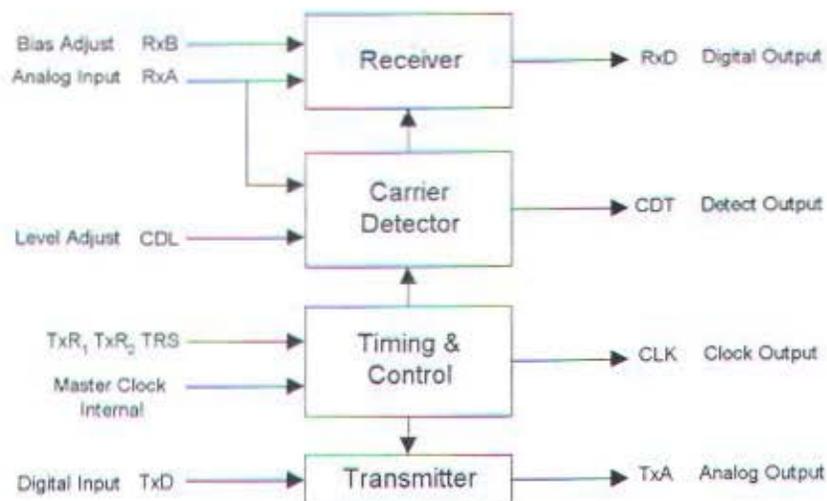
Karena keterbatasan bandwidth line telephone maka untuk sistem komunikasi full duplex biasanya mempunyai perbedaan kecepatan pada pengiriman dan penerimaan dan hal ini dimaksudkan untuk mengoptimalkan kecepatan transmisi dimana transmisi balik hanya berisi sedikit informasi (mengenai control error).

### 2.3.2 TCM 3105

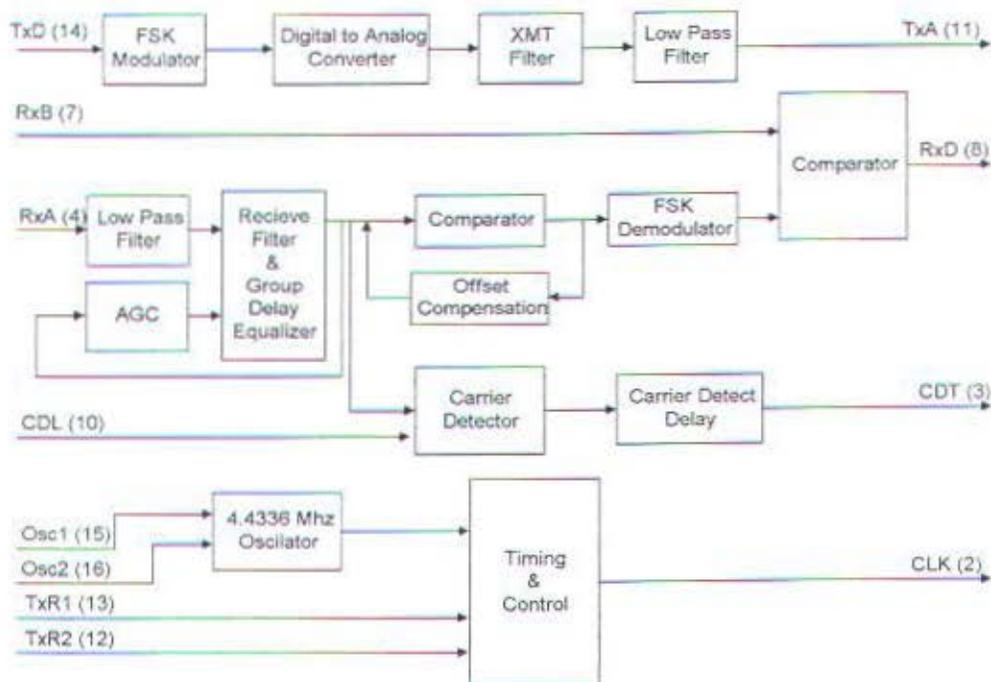
TCM 3105 adalah chip modem (modulator dan demodulator) buatan Texas Instrument dengan sistem FSK. modul ini telah mengikuti standar modem dari CCITT V.23 dan BELL 202 sehingga akan lebih mudah dalam standarisasinya.

Kecepatan dari modem ini dapat diatur yaitu :

- Transmit mode : 75, 150, 600, 1200 baud
- Receive mode : 5, 75, 150, 600, 1200 Baud



Gambar 2.4 Pembagian Sistem TCM 3105



Gambar 2.5 Blok Diagram Fungsional TCM 3105<sup>1</sup>

IC TCM 3105 merupakan modul modem dengan blok diagram dapat dilihat pada Gambar 2.4 dan 2.5 dimana untuk merancang modem diperlukan sedikit komponen tambahan. Yang perlu diperhatikan bahwa polaritas dari tegangan RS 232 tidak sama dan juga level tegangannya lebih tinggi sehingga diperlukan komponen tambahan untuk menyesuaikan dengan input dan output pada IC TCM 3105, Untuk itu diperlukan IC konverter seperti ICL232 , 74HC04, MC 1488, MC 1489. Fungsi dari tiap-tiap pin TCM 3105 dapat dilihat pada Tabel 2.3.

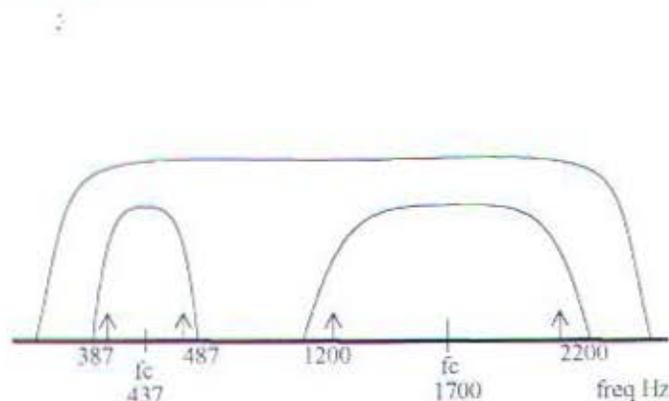
<sup>1</sup> Texas Instruments, DATA MANUAL : FSK MODEM ( Texas :Texas Instruments Inc, 1991), p. 2-99

Tabel 2.3 Tabel Fungsi dari kaki pin TCM 3105

PIN		I/O	Keterangan
Nama	No		
TxD	14	I	Transmit Digital, input digital yang akan dimodulasikan dg logika '1' - mark dan '0' - space. Baud rate dapat lebih kecil dari mode yang dipilih
RxA	4	I	Recieve Analog, input signal dikopling AC (dengan-kapasitor)
RxB	7	I	Recieve Bias Adjust, Mengatur referensi input untuk menghilangkan distorsi pada RxD
RxD	8	O	Receive Digital Output, Output hasil demodulasi dengan logika '1' untuk mark dan logika '0' untuk space, logika '1' jika tidak ada sinyal input
CDL	10	I	Carrier Detect Level, setting level pendeteksian carrier pd RxA
CDT	3	O	Carrier Detect, logika '1' menandakan ada carrier pada RxA
TRS	5	I	Transmit / Receive Standard Select Input, detail pada tabel mode
TxR1	13	I	Transmit Rate 1 dan 2, memilih kecepatan untuk detail pada tabel mode
TxR2	12		
CLK	2	O	Output clock signal 16 x dari sinyal transmit/Reccive tertinggi
OSC1	15	-	Koneksi ke kristal eksternal 4,4336 Mhz
OSC2	16		
Vdd	1	-	tegangan + 5 v dengan daya sekitar 40 mW

TCM 3105 dapat dioperasikan pada berbagai mode seperti mode full duplex atau mode half duplex. Pada operasi full duplex frekuensi transmit dari pengirim dan transmit untuk penerima berbeda agar tidak terjadi cross-talk dan kecepatan keduanya pun berbeda sebagai contoh pada standar BELL 202 pada transmit 1200 baud dan receive 150 baud maka frekuensi transmit mark : 1200 Hz dan space : 2200 Hz pada

frekuensi receive mark : 387 Hz dan space 487 Hz untuk gambar daerah frekuensi dari contoh diatas dapat dilihat pada Gambar 2.6.



Gambar 2.6 Daerah frekuensi Standar Bell 202

Untuk mode half duplex transmit dan receive kecepatan dan frekunsinya harus disamakan sebagai contoh transmit 1200 baud dan receive 1200 baud. Sedangkan pada Tugas Akhir ini kami menggunakan 1200 baud seperti mode half duplex karena dasar transmisinya sama.

## 2.4 PROTOKOL

Protokol jaringan adalah aturan yang mengatur bagaimana data dikirimkan dan diterima pada komputer jaringan tersebut. Salah satu bagian protokol adalah metode pengiriman paket - paket data.

### 2.4.1 Paket Data

Secara umum menurut pembatasan arus paket data yang dikirim ada 3 macam paket pada sistem protokol :

### 1. Paket yang dibatasi oleh karakter kontrol

Biasa digunakan untuk pengiriman teks biasa ASCII (plain text) yaitu alphanumeric karakter, ditambah 6 karakter kontrol (BS, HT, LF, VT, FF dan CR). Jadi untuk karakter kontrol lain tidak diperkenankan, hal ini karena karakter tersebut telah digunakan untuk sistem pengiriman data. Gambaran paket yang dibatasi oleh karakter kontrol dapat dilihat pada Gambar 2.7.



Gambar 2.7 Paket data dengan pembatas karakter kontrol

*SOH* (Start Of Header) merupakan kontrol karakter yang menandakan awal dari paket.

*No Paket* Adalah menunjukkan posisi dari paket yang dikirim

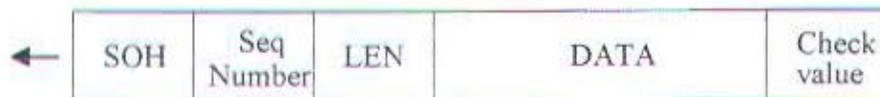
*STX* (Start Of Text) merupakan karakter kontrol yang menunjukkan bahwa karakter berikutnya adalah karakter data teks yang dikirim.

*DATA* Berisi data file teks.

*ETX* (End Of Text) merupakan karakter kontrol yang menandakan akhir dari data file teks yang dikirim.

## 2. Paket yang dibatasi dengan ukuran fieldnya

Bila komunikasi yang dilakukan menggunakan file binary biasanya data akan terdiri dari 8 bit tiap karakternya sehingga pengiriman dengan metode diatas tidak memungkinkan lagi, alternatif lain adalah dengan memberikan ukuran data yang akan dikirimkan (data 8 bit) tersebut. ukuran data (LEN) yang dikirim biasanya diubah dalam karakter 8 bit juga. Jadi bila LEN terdiri dari 1 byte maka maksimum panjang data adalah 256 byte. Gambaran paket yang dibatasi dengan ukuran fieldnya dapat dilihat pada Gambar 2.8.



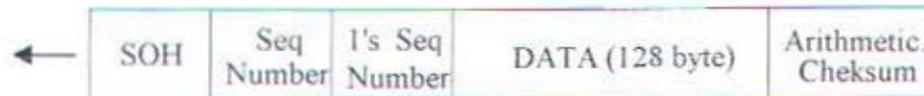
Gambar 2.8 Paket dengan field penanda panjang

## 3. Paket yang berukuran tetap

Paket ketiga ini lebih mudah direalisasikan yaitu karena tiap-tiap paket mempunyai panjang yang sama dan struktur yang hampir sama pula sehingga pola dari tiap paket dapat dengan mudah disusun kembali.

Kelemahan dari metode ini bila pada akhir pengiriman ukurannya lebih kecil dari ukuran paketnya maka akan menimbulkan karakter sampah, jadi diperlukan suatu program tambahan untuk menghilangkan data yang tidak berguna tadi. Gambar dari paket dengan ukuran tetap dapat dilihat pada Gambar 2.9. Protokol XMODEM juga menggunakan protokol type ini. Protokol ini dibuat pertama kali pada tahun 1977

oleh Ward Christensen dan sudah menjadi protokol yang memasyarakat.



Gambar 2.9 Paket Dengan Panjang Tetap (XMODEM)

*SOH* (Start Of Header) merupakan kontrol karakter yang menandakan awal dari paket.

*No Paket* Adalah menunjukkan posisi dari paket yang dikirim

*1's No Paket* Adalah 1's complement No Paket diatas

*DATA* Panjang data sebesar 128 byte dan bisa berisi data teks atau binary file

*Arithmetic Chechsum*

1 byte arithmetic sum hanya dari isi data yang ada modulo - 256.

#### 2.4.2 Error Check

Dalam mengecek kesalahan dibutuhkan informasi tambahan yang berisi data untuk menguji kebenaran isi dari data yang sering disebut *redudancy*. redudancy dapat berupa bit tambahan ataupun byte tambahan. berdasarkan macam redudancy error check dibagi menjadi beberapa type yaitu :

### 2.4.3 Parity Redudancy Karakter

Metode yang umum dipakai dan mudah diimplementasikan adalah metode redudancy karakter, yaitu pada karakter data ditambahkan bit sehingga selalu bernilai tetap; bila dipakai paritas genap maka nilai dari karakter data selalu genap dan paritas ganjil dimana karakter selalu bernilai ganjil.

Dengan demikian penerima dapat menentukan terjadinya kesalahan dengan menghitung nilai dari karakter data, jika paritas genap tetapi dihasilkan ganjil maka dapat dipastikan telah terjadi kesalahan.

Contoh Parity Redudancy Karakter : (tanda tebal sebagai petunjuk kesalahan).

Transmitted	Received	Kesimpulan untuk parity genap
01000001	01000001	bernilai genap : parity OK
01000001	01000011	(satu bit error) bernilai ganjil : error terdeteksi
01000001	00000011	(dua bit error) bernilai genap : error tak terdeteksi
01000001	010 <b>111</b> 01	(tiga bit error) bernilai ganjil : error terdeteksi

Dalam prakteknya IC UARTs telah memudahkan manipulasi dari data format asinkron tersebut tetapi banyak programmer tidak menggunakan metode parity redudancy character ini, karena metode ini sangat memboroskan waktu pengiriman data sebagai contoh : Untuk mengirimkan satu karakter yang terdiri dari 10 bit ( 1 start bit, 7 bit data, 1 bit parity, 1bit stop bit) maka akan menurunkan performance kecepatan sebesar 10 % padahal hasil ketelitian dalam mendeteksi error hanya 40 %<sup>2</sup>.

<sup>2</sup> Campbell, Joe. SERIAL COMMUNICATION : C PROGRAMMER'S GUIDE (Indiana : SAMS Publishing, Second Edition, 1994), p. 66.

#### 2.4.4 Parity Redudancy Blok

Ketika blok data (terdiri dari banyak karakter) dikirimkan, maka kemungkinan terjadinya kesalahan juga meningkat. Sehingga untuk meningkatkan mutu deteksi kesalahan, disamping setiap byte data ditambahkan satu bit paritas (transverse atau paritas baris) juga dilakukan kalkulasi untuk setiap posisi bit dari setiap bit blok data sehingga dihasilkan 1 byte data yang mewakili nilai paritas untuk setiap posisi bit data (longitudinal atau paritas baris)

Metode parity redudancy block ini menggunakan cara menata karakter menjadi array 2 dimensi kemudian menjumlahkan tiap-tiap kolom (bit) hingga akhir dari baris nilai ini yang disebut sebagai *block check character* (BCC) yang menunjukkan nilai genap / ganjil.

Contoh metode ini adalah sebagai berikut : (dengan sistem parity genap)

	Data	2 bit error dalam 1 baris	2 bit error dalam 2 baris
C	11000011	11000011	11000011
f	01100110	01100110	01100110
y	11111001	11100001	11100001
U	01010101	01010101	01001101
BCC	00001001	00001001	00001001
BCC penerima		000010001	00001001
Hasil		terdeteksi	tak terdeteksi

Dari contoh terlihat bahwa dengan metode parity block redundancy ini dapat mendeteksi bit error berurutan dalam satu karakter yang tak terdeteksi oleh redundancy karakter. Namun tidak dapat mendeteksi error bit yang berurutan tetapi dalam beberapa karakter.

#### 2.4.5Checksum Redudancy Blok

Pengenalan terhadap error check dengan sistem vertikal / longitudinal merupakan hal yang penting untuk mengecek error pada blok tiap karakter sendiri, namun untuk beberapa karakter yang lain masih kurang memuaskan. Untuk mengatasi hal tersebut ada teknik lain yaitu dengan menghitung secara aritmetika (arithmetic checksum). Nilai dari tiap karakter dijumlah kemudian disimpan sebagai BCC.

Contoh metode ini adalah sebagai berikut :

	Data	2 bit error dalam 1 baris	2 bit error dalam 2 baris	1 bit error dalam 2 baris
C	1000011	1000011	1000011	1000010
f	1100110	1100110	1100110	1100111
y	1111001	1100001	1100001	1111001
U	1010101	1010101	1001101	1010101
BCC	101110111	101110111	101110111	101110111
BCC penerima		101011111	101011111	101110111
Hasil		terdeteksi	terdeteksi	tak terdeteksi

Hasil dari metode ini cukup baik untuk mendeteksi error ganda pada tiap-tiap baris namun gagal pada pendeteksian bit tunggal pada tiap baris, dimana dengan sistem paritas sebelumnya dapat dideteksi.

#### 2.4.6 Cyclic Redundancy Check (CRC)

Deteksi error dengan metode paritas diatas akan sangat baik untuk mendeteksi kesalahan single bit (bit tunggal acak) yang terjadi. Tetapi akan buruk sekali jika diterapkan untuk mendeteksi kesalahan pada burst-error, dimana kesalahan terjadi pada beberapa bit secara berurutan. Sedangkan metode checksum lemah bila mendeteksi error bit tunggal acak seperti yang terdeteksi oleh error check metode parity tersebut.

Prinsip dasar dari metode cyclic redundancy check adalah dengan menambahkan data hasil sisa dari suatu hasil pembagian dari suatu standar bilangan tertentu sebagai parameternya (remainder). Sebagai contoh data 4 byte CfyU seperti contoh sebelumnya tetap dengan menggunakan format 8 bit :

C	11000011
f	01100110
y	11111001
U	01010101

Bila data diolah sebagai bilangan biner tunggal yang besar kita akan mendapatkan : 11000011011001101111100101010101 atau nilai dalam desimal adalah 3.278.305.621 maka kita misalkan bilangan standar pembagian adalah 65.540 (17 bit) maka :

$$\frac{3.278.305.621}{65.540} = 624.439 \text{ dengan sisa } 60.361$$

Dengan demikian nilai sisa  $60.361 = \text{EBC9} = 1110101111001001$  (16bit) dikirimkan sebagai kode cek error yang ditambahkan untuk per framenya. Bila panjang pembagi standar (divisor)  $n$  bit maka panjang dari kode cek tadi tidak akan melebihi dari panjang divisor atau maksimum  $n-1$  bit.

Permasalahan baru yang muncul dari metode ini adalah bagaimana cara membagi bila bilangan binernya sangat besar misalnya 256 byte maka bila digunakan cara konvensional maka komputer akan kesulitan menghitung bilangan biner dengan panjang 2048 bit ( $256 \text{ byte} \times 8 \text{ bit/byte}$ ) karena komputer tidak akan mampu menampilkan bilangan  $2^{2048}$ .

Karena yang dihitung hanyalah sisanya maka dengan bantuan pembagian aritmatik modulo 2 kesulitan tersebut dapat diatasi. Sifat dari modulo 2 tidak mempunyai carries atau borrows dan untuk operasinya digunakan XOR sifat lain adalah operasi penjumlahan dan pengurangan akan menghasilkan bilangan yang sama.

Metode pembagian dari modulo 2 adalah sebagai berikut : misal data 11100110 dibagi dengan 11001

$$\begin{array}{r}
 \begin{array}{l} 11001 \\ \uparrow \\ \text{divisor} \end{array} \quad \begin{array}{r}
 \overline{) 11100110} \\
 \oplus 11001 \\
 \hline
 \oplus 01011 \\
 \hline
 \oplus 00000 \\
 \hline
 \oplus 10111 \\
 \hline
 \oplus 11001 \\
 \hline
 \oplus 11100 \\
 \hline
 \oplus 11001 \\
 \hline
 \oplus 0101 \\
 \hline
 \text{remainder}
 \end{array}
 \end{array}$$

← quotient (diabaikan)  
← message

Dari perhitungan tersebut diatas maka dapat dirumuskan sebagai berikut

$$Message = (Quotient \times Divisor) + Remainder$$

Dengan menambahkan remainder pada kedua sisi didapatkan

$$Message + Remainder = (Quotient \times Divisor) + Remainder + Remainder$$

Pada modulo 2 kita ketahui bahwa operasi penjumlahan (+) sama dengan operasi XOR ( $\oplus$ ) sehingga persamaan menjadi :

$$Remainder + Remainder \Rightarrow Remainder \oplus Remainder \Rightarrow 0$$

Sekarang persamaan diatas akan menjadi :

$$Message \oplus Remainder = Quotient \times Divisor$$

$$\frac{Message \oplus Remainder}{Divisor} = Quotient + 0$$

Karena yang dihitung diatas adalah sisanya saja (remainder) maka hal tersebut dapat dipakai sebagai test dari data dan kode error yang dikirim dimana bila Remainder bernilai 0 maka tidak ada error sedangkan bila tidak bernilai 0 maka ada error pada data atau kode error.

Prinsip dari metode CRC adalah hampir sama dengan kondisi diatas namun ada sedikit perbedaan yaitu kode cek error dijadikan satu pada akhir data sehingga data dan CRC tersebut menjadi kesatuan data baru dimana data lama akan bertambah bitnya sebesar panjang kode cek errornya. Pada yang umum panjang redundancy tersebut adalah 16 bit dan 32 bit yang disebut *computed check digits*. Computed

check digits dipakai sebagai digit *frame check sequence (FCS)* atau *cyclic redundancy check (CRC)*

Sebagai contoh perhitungan CRC pada sisi pengirim dan sisi penerima adalah sebagai berikut :

**Sisi Pengirim :**

Message (data)	11100110	(8 bit)
Divisor (n bit)	11001	(5 bit)
Remainder (n-1 bit)	—————→	(4 bit)
Message B (8 bit + 4 bit)	111001100000	(12 bit)

	xxxxx ← quotient (diabaikan)
11001	111001100000 ← message B
↑	⊕ 11001
divisor	x01011
	00000
	⊕ x10111
	11001
	⊕ x11100
	11001
	⊕ x0101
	00000
	⊕ x10100
	11001
	⊕ x11010
	11001
	⊕ x00110
	00000
	⊕ x0110 ← remainder (FCS/CRC)

Message B	111001100000
Remainder	0110
Message yang dikirimkan	111001100110 ⊕

Sisi Penerima :

Message diterima (data) 111001100110 (12 bit)  
 Divisor (n bit) 11001 (5 bit)

$$\begin{array}{r}
 \text{xxxxx} \leftarrow \text{quotient (diabaikan)} \\
 \begin{array}{r}
 11001 \overline{) 111001100110} \leftarrow \text{message} \\
 \oplus \underline{11001} \\
 \text{x01011} \\
 \oplus \underline{00000} \\
 \text{x10111} \\
 \oplus \underline{11001} \\
 \text{x11100} \\
 \oplus \underline{11001} \\
 \text{x0101} \\
 \oplus \underline{00000} \\
 \text{x10101} \\
 \oplus \underline{11001} \\
 \text{x11001} \\
 \oplus \underline{11001} \\
 \text{x00000} \\
 \oplus \underline{00000} \\
 \text{x0000} \leftarrow \text{Remainder (Result)}
 \end{array}
 \end{array}$$

Remainder = 0  $\longrightarrow$  tidak ada error

*Cyclic redundancy check (CRC)* dengan panjang 16 bit dapat mendeteksi :

- Kesalahan bit tunggal (single-bit error) 100 %
- Kesalahan bit ganda (double-bit error). 100 %
- Kesalahan burst (error burst) < 16 bits 100 %
- Kesalahan burst (error burst) 17 bits 99,9969 %
- Semua burst error yang lain 99,9984 %

Standar dari nilai generator polinomial adalah penentuan divisor dengan posisi dari angka biner yang bernilai 1 sebagai pangkat dari X.

Berikut ini standar yang umum dikenal dan dipakai secara luas adalah<sup>3</sup> :

$$\text{CRC-16} = X^{16} + X^{15} + X^2 + 1$$

$$\text{CRC-CCITT} = X^{16} + X^{12} + X^5 + 1$$

$$\begin{aligned} \text{CRC-32} = & X^{32} + X^{26} + X^{23} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + \\ & X^4 + X^2 + X^1 + 1 \end{aligned}$$

Jadi, CRC-16 ekivalen dengan biner dalam bentuk : 1 1000 0000 0000 0101.

Dengan generator polinomial seperti diatas (17 bit), maka diperlukan 16 angka nol (biner) sebagai tempat FCS diakhir frame. CRC-16 dan CRC-CCITT dipakai secara luas di aplikasikan wide area network (WAN), sedangkan CRC-32 dipakai secara luas di local area network (LAN).

## 2.5 PEMROGRAMAN WINDOWS 95

Dalam sistem operasi windows 95 telah banyak software yang digunakan sebagai bahasa pemrograman, yang dalam perkembangannya software pembuat aplikasi ini telah menuju ke sistem visual dimana kita sudah tidak perlu report dalam membuat tampilan.

---

<sup>3</sup> Halsall, Fred. DATA COMMUNICATIONS COMPUTER NETWORKS AND OPEN SYSTEM (England : Addison-Wesley Publishing Company, Third Edition, 1993), p.110.

Salah satu dari software pemrograman adalah DELPHI yang dibuat oleh BORLAND Inc. dan menggunakan bahasa PASCAL. Dalam Tugas Akhir digunakan DELPHI ver 2.0 untuk membuat sistem perangkat lunaknya.

### 2.5.1 Deklarasi Variabel

Penulisan program dalam bahasa pascal dalam memakai variabel harus mematuhi aturan sebagai berikut :

- Panjang maksimum 63 karakter.
- Diawali dengan huruf atau garis bawah ( \_ ).
- Dapat disusul kemudian dengan huruf, angka dan ( \_ ).
- Tidak dibolehkan menggunakan karakter simbol seperti %, !, + .
- Tidak dibolehkan menggunakan kata yang telah dipakai oleh delphi seperti :  
program, do, begin dan sebagainya.

Ada 6 katagori data yaitu ; Integer, Real, Boolean, Char, String dan Pointer.

Untuk integer dan real dapat dibagi menjadi beberapa type yang ditunjukkan pada tabel 2.4 dan 2.5 berikut :

Tabel 2.4. Tipe Data Integer

Type	Range	Size (byte)	Status
ShortInt	-128 .. 127	1	Signed
Integer	-32768 .. 32767	2	Signed
LongInt	-2147483648 .. 2147483647	4	Signed
Byte	0 .. 255	1	Unsigned
Word	0 .. 65535	2	Unsigned

Table 2.5. Tipe Data Real

Type	Range	Size (byte)	Decimal point
Real	$\pm 2,9 \times 10^{-39} \dots 1,7 \times 10^{38}$	6	Floating
Single	$\pm 1,5 \times 10^{-45} \dots 3,4 \times 10^{38}$	4	Floating
Double	$\pm 5,0 \times 10^{-324} \dots 1,7 \times 10^{308}$	8	Floating
Extended	$\pm 3,4 \times 10^{-4932} \dots 1,1 \times 10^{-4932}$	10	Floating
Comp	$-2^{63} + 1 \dots 2^{63} - 1$	8	Fixed

Boolean menyatakan suatu nilai boolean (true atau false) yang digunakan untuk operasi logika atau operator relasi.

String merupakan data yang berisikan sederetan karakter dengan jumlah 1 sampai 255 karakter. Sedang pada delphi terdapat data type PCHAR merupakan data yang hampir sama dengan string, yang berbeda bila pada string panjang dibatasi pada 1byte pertama (String[0]) sedang pada PCHAR dibatasi pada akhirnya dengan NULL sehingga panjang pada PCHAR bisa lebih panjang dari 255 karakter. Pada Delphi 2.0 String dan PCHAR adalah sama sehingga untuk melihat panjang tidak bisa menggunakan string[0] tetapi harus menggunakan length.

Char menyatakan nilai (value) satu karakter pada tabel ASCII, nilai numeriknya adalah 0 sampai 255.

Pointer adalah suatu type data yang berisi suatu alamat memori tertentu. Isi memori yang ditunjuk oleh variabel bertipe pointer dapat bermacam-macam tipenya. Untuk mengambil variabel yang ditunjuk oleh pointer digunakan tanda (^).

### 2.5.2 Struktur Kontrol

Struktur kontrol adalah suatu metode untuk mengalihkan langkah eksekusi suatu proses sesuai dengan kondisi yang telah ditentukan contoh dari fungsi kontrol ini : `if .. then` dan `Case .. of`

Perintah `if` akan mengecek suatu kondisi dan menentukan apakah kondisi tersebut benar atau salah, kemudian melakukan suatu kegiatan sesuai dengan kondisi. Dimana statemen sesudah `then` dilaksanakan jika kondisi benar sedangkan kondisi false statemen sesudah `else` yang dilaksanakan

Perintah `case` berisi ungkapan / pemilih dan sederetan statemen yang masing-masing diawali dengan satu atau lebih konstanta (disebut konstanta `case`) atau dengan kata `else`. Pemilih harus bertipe `ordinal` yang mempunyai batas -32768 sampai 32767.

### 2.5.3 Struktur Perulangan (Loop)

Yang termasuk struktur ini adalah `Repeat..until`, `While..Do` dan `For..do`. Ketiganya mempunyai fungsi sebagai pengulangan suatu program. Perbedaan mendasar dari ketiganya adalah :

- `Repeat .. Until` : perulangan akan dilakukan sampai pada saat bila kondisi telah terpenuhi dimana statemen di test diakhir perintah.
- `While .. do` : Perulangan akan dilakukan bila kondisi terpenuhi dimana test kondisi diawal statemen perintah.
- `For .. do` : Perulangan dilakukan sebanyak yang ditentukan

#### 2.5.4 Kontrol Serial Port

Sistem operasi Windows 95 merupakan sistem operasi yang mengontrol sebagian besar hardware untuk digunakan secara bersama sehingga untuk mengakses hardware kita tidak perlu bersusah payah karena sudah ditangani oleh sistem windows sendiri. Contohnya dalam mengakses Serial Port (COM) akses yang digunakan sudah seperti kita mengakses file.

Perintah-perintah yang telah disediakan oleh windows 95 untuk pemrograman dari COM adalah :

```
function Createfile (devicename : PChar): THandle;
Setup Comm (FHandle : THandle, ReadBufferSize : integer, WriteBufferSize :
integer)
GetCommState (FHandle : THandle, Var FDCB : TDCB);
SetCommState (FHandle : THandle, FDCB : TDCB);
ClearCommErrors (FHandle : THandle, Errors : DWord, FCT : TFCT);
WaitCommEvent (FHandle : THandle, Event : Dword, Eventstatus :
TOverLapped);
CloseHandle (FHandle : THandle);
```

Untuk menggunakan device RS 232 terlebih dahulu kita membuka COM1 port yang hendak digunakan dengan perintah :

```
FHandle:=CreateFile ('COM1', GENERIC_READ or GENERIC_WRITE, 0,
nil, OPEN_EXISTING, FILE_FLAG_OVERLAPPED, 0);
```

Fungsi tersebut akan membuka COM1 untuk keperluan membaca dan menulis. Fungsi ini menghasilkan variabel FHandle yang bertipe THandle. FHandle akan dipakai selanjutnya sebagai pengenalan dari COM1 pada operasi berikutnya.

Untuk mengatur dari baudrate, parity, data bits serta stop bits pada RS 232 digunakan perintah

```
SetCommState(FHandle, FDCB);
```

FHandle adalah pengenalan device yang telah ditentukan semula, sedangkan FDCB merupakan variabel bertipe record yang berisi :

- FDCB.BaudRate
- FDCB.Parity
- FDCB.StopBits
- FDCB.ByteSize

Sedangkan perintah *GetCommState()* akan menghasilkan status dari FDCB pada handle tersebut. Untuk mendapatkan jumlah data yang ada dalam buffer tulis maupun buffer baca dapat dilakukan dengan perintah.

```
ClearCommError(FHandle, error, @FCT);
```

Panjang data yang berada dalam buffer akan dihasilkan dengan mengakses FCT yang merupakan type variabel record dan berisi : FCT.cbInQue sebagai panjang data yang berada dalam buffer baca sedangkan FCT.cbOutQue sebagai panjang data yang berada dalam buffer tulis.

Untuk mendapatkan event dari RS 232 digunakan :

```
WaitCommEvent (FHandle, Event, Eventstatus);
```

Event yang dihasilkan adalah bertipe dword dengan event -event antara lain :

- EV\_RLSD : Menunjukkan sinyal RLSD terdeteksi

- EV\_RING : Menunjukkan sinyal Ring (RI) terdeteksi
- EV\_RXCHAR : Menunjukkan data telah berada dibuffer dan siap dibaca
- EV\_ERR : Line terdapat error (Frame error, parity error, Overrun Error)
- EV\_TXEMPTY : Karakter terakhir telah terkirim.

Untuk menutup handle dari COM port tersebut diatas dilakukan dengan perintah Closehandle (FHandle);

## 2.6 DATA DAN PENAMPILAN DATA

Informasi dalam mode text biasa (ASCII Standar) telah menjadi standar dari pengiriman data text, Namun tampilan dari text ASCII menjadi monoton karena tidak bisa menampilkan font yang beragam (seperti terminal TELNET). Dari kekurangan tadi muncul berbagai format baru dalam penyajian data salah satunya adalah HTML (Hyper Text Markup Language) dimana HTML telah menjadi standar baru dalam perkembangan internet.

Dengan menggunakan HTML ini maka format ASCII dapat ditampilkan dalam berbagai macam font, warna, dan format (italic, bold, underline dan lainnya). Disamping itu keunggulan HTML adalah dapat dibuat link antar halaman sehingga dapat melompat ke halaman yang dikehendaki. Dalam perkembangannya link HTML telah dapat mengakses gambar dan bahkan Multimedia seperti suara, video.

Untuk menampilkan data dengan format HTML kita dapat membukanya dengan Browser Web internet yang sudah ada seperti Netscape, Microsoft Internet

Explorer, Mosaic dan lain sebagainya. contoh dari tampilan browser IE 3.0 dapat dilihat pada Gambar 2.10.

Sebagai contoh isi dari format HTML dan tampilannya pada MSIE 3.0 dapat dilihat sebagai berikut.

```
<HTML>
<Script Language="JavaScript">
function AutoLoad()
{location.HREF="coba.html"}
timerID=setTimeout("AutoLoad()",10000);
</Script>
<BODY>
<H1 ALIGN=center>Harga Sayur Mayur</H1>
<HR>
<H5 ALIGN=center>Last Update: 13/09/97</H5>
<H6 ALIGN=center><H6>Time:1:00:11</H6></H6>
<CENTER><TABLEBORDER=1>
<TR><TH><TR><TH>BARANG</TH><TH>TEMPAT</TH><TH>HARGA</TH></TR>
<TR><TD>CABE KERITING</TD><TD>SURABAYA</TD><TD>5000</TD></TR>
<TR><TD>CABE BESAR</TD><TD>SURABAYA</TD><TD>4000</TD></TR>
<TR><TD>KOL BULAT</TD><TD>SURABAYA</TD><TD>1000</TD></TR>
<TR><TD>TOMAT</TD><TD>SURABAYA</TD><TD>1500</TD></TR>
<TR><TD>WORTEL</TD><TD>SURABAYA</TD><TD>2000</TD></TR>
<TR><TD>CABE KERITING</TD><TD>JAKARTA</TD><TD>7500</TD></TR>
<TR><TD>CABE BESAR</TD><TD>JAKARTA</TD><TD>6000</TD></TR>
</TABLE></CENTER>
<FONT SIZE=1>Table:coba.dbf</FONT>
</BODY>
</HTML>
```

Pada HTML diatas digunakan java script untuk membuat auto refresh yang berguna membaca data pada selang waktu tertentu, hal ini akan berguna pada sistem refresh database yang dapat refresh dalam selang waktu pengiriman data berikutnya.

**Harga Sayur Mayur**

Last Update: 15/02/07  
Jenis: 1/00-11

BARANG	TEMPAT	HARGA
CABE KEJITING	SURABAYA	5000
CABE BESAR	SURABAYA	4000
KOL DITAT	SURABAYA	1000
TOMAT	SURABAYA	1500
WORTEL	SURADAYA	2000
CABE KEJITING	JAKARTA	7500
CABE BESAR	JAKARTA	6000

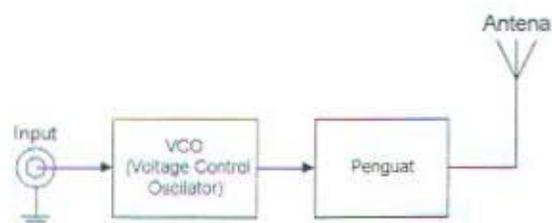
Gambar 2.10 Tampilan dari file HTML pada MSIE 3.0

## 2.7 MEDIA KOMUNIKASI

Untuk media komunikasi digunakan dengan sistem modulasi FM yang dapat diletakkan pada frekuensi tertentu yang belum digunakan stasiun komersial yang ada. Pemilihan sistem modulasi ini dikarena lebih kebal noise dan harganya juga relatif murah.

### 2.7.1 Pemancar FM

Rangkaian pemancar pada prinsipnya adalah berupa rangkaian penghasil gelombang sinus (oscilator) yang frekuensinya dapat berubah-ubah sesuai dengan sinyal inputnya. Gambar 2.9 memperlihatkan diagram blok pemancar FM Sederhana.

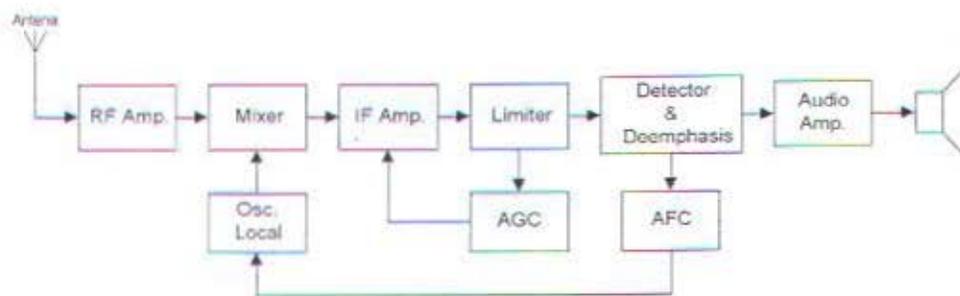


Gambar 2.11 Diagram Blok Pemancar

VCO merupakan oscilator yang frekuensinya dapat berubah-ubah sesuai dengan tegangan input, komponen yang sering digunakan adalah dioda varaktor yaitu dioda yang mempunyai kapasitansi berubah-ubah sesuai dengan tegangan biasnya. Jadi dengan memberikan sinyal input maka kapasitansi dari dioda tersebut akan berubah sehingga dengan perubahan kapasitansi maka frekuensi yang dibangkitkan dari rangkaian oscilator LC (kumparan dan kondensator) akan berubah pula. Untuk lebih menguatkan dan menyetabilkan frekuensi output ditambahkan penguat sehingga hasil sinyal oscilator dapat dipancarkan lebih jauh dan stabil.

### 2.7.2 Radio Penerima FM

Radio penerima ini berguna untuk mendapatkan kembali sinyal yang dibawa oleh sinyal FM. Diagram blok rangkaian penerima FM dapat diperlihatkan pada Gambar 2.12.



Gambar 2.12. Diagram Radio Penerima FM

Urutan kerjanya adalah sebagai berikut : RF amplifier merupakan penguat penerimaan dari sinyal FM yang diterima antena. Mixer berfungsi untuk mencampur frekuensi yang dihasilkan oscilator lokal dengan sinyal RF amplifier sehingga didapatkan frekuensi selisih sebesar 10,7 Mhz kemudian sinyal IF ini dikuatkan dan kemudian diratakan amplitudonya oleh limiter hasil dari bagian ini kemudian diumpankan ke detektor dan deemphasis sehingga sinyal IF tersebut dirubah menjadi sinyal audio. Sinyal audio yang akan diumpankan ke speaker dikuatkan dulu oleh penguat audio. Frekuensi yang dapat diterima oleh radio FM yang beredar dipasaran adalah diantara 88 Mhz sampai 108 Mhz

# BAB III

## PERENCANAAN DAN PEMBUATAN PERANGKAT KERAS

---

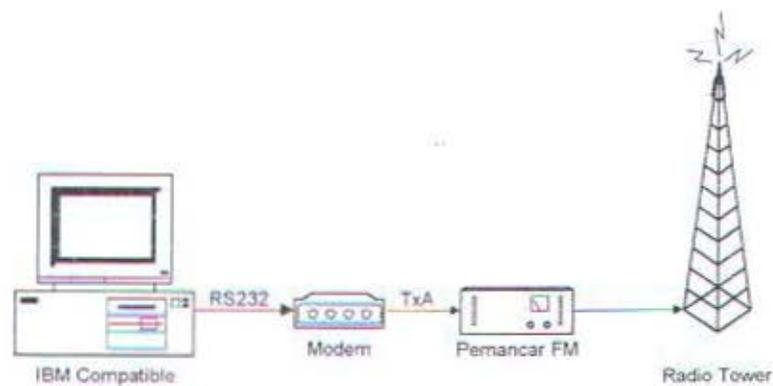
Pada pembuatan sebuah sistem, maka langkah awal yang dilakukan adalah tujuan dari pembuatan sistem tersebut. Hal ini dapat tercermin dari diagram blok yang dibuat. Penjelasan dimulai dari diagram blok secara umum, kemudian dilanjutkan dengan penjelasan setiap bagian blok secara lebih terperinci.

### 3.1 DIAGRAM BLOK

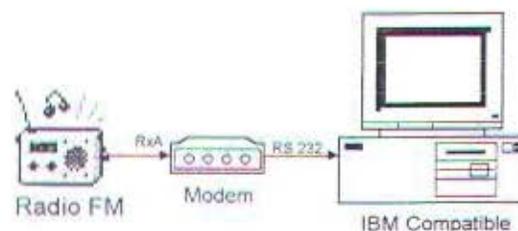
Blok diagram meliputi dua bagian besar yaitu; bagian pengirim dan bagian penerima. Pada bagian pemancar berisi komputer dengan perangkat lunak pengirim, modem FSK, unit pemancar FM. Bagian penerima terdiri dari unit penerima radio FM, modem dan komputer yang berisi perangkat lunak penerima. Blok diagram tersebut diperlihatkan pada gambar 3.1 dan 3.2

Rangkaian modem yang dibuat telah mempunyai bagian modulator dan demodulator, sehingga untuk bagian pengirim dan penerima rangkaian modem yang digunakan adalah sama yang berbeda adalah cara menghubungkannya pada rangkaian yaitu bila pada pengirim yang digunakan hanya bagian modulator sedangkan pada penerima bagian yang digunakan adalah bagian demodulator saja.

Secara umum pembuatan perangkat keras hanyalah pembuatan modem.



Gambar 3.1. Blok Diagram Unit Pengirim

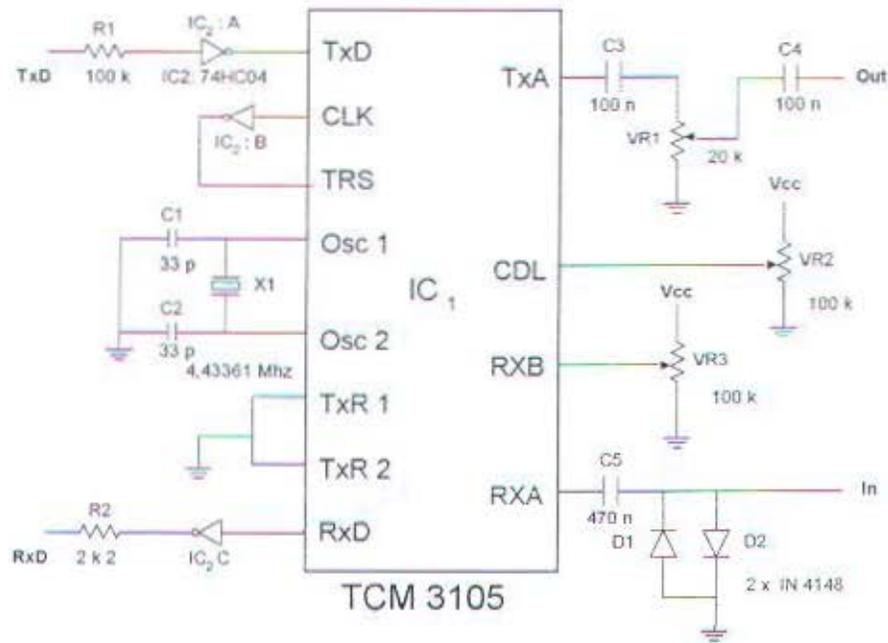


Gambar 3.2. Blok Diagram Bagian Penerima

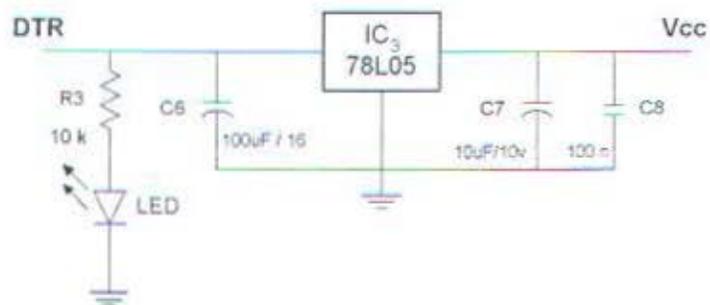
### 3.2 MODEM

Sebagai komponen utama dalam modem FSK ini adalah IC TCM 3105 buatan Texas Instruments Inc. yang sudah memiliki modulator dan demodulator menjadi satu. Modem ini dapat diset mode dan kecepatannya hingga pada kecepatan 1200 baud.

Pemilihan IC TCM 3105 karena dibutuhkan sedikit komponen tambahan dan sederhana dalam penyetalaan frekuensi (terdapat kristal standar osilator). Untuk tambahan komponen yang diperlukan yaitu : Catu daya, penyesuai dengan saluran RS232, Kristal osilator, Pengatur Tegangan (Level Adj.). Rangkaian modemnya dapat dilihat pada gambar 3.3.



Gambar 3.3. Rangkaian Modem



Gambar 3.4. Rangkaian Catu Daya

Untuk rangkaian catu daya kami sengaja mengambil sumber daya dari port RS232, hal ini bisa dimungkinkan karena TCM3105 ini sangat kecil komsumsi dayanya yaitu 40mW dengan tegangan searah 5 Volt sehingga tidak diperlukan lagi

tegangan dari luar. Untuk menurunkan dan menyetabilkan tegangan dari port RS 232 yang lebih tinggi dari 5 V maka ditambahkan IC regulator 78L05 sehingga tegangan yang dikeluarkan menjadi stabil 5 V. Pin yang kami pilih adalah DTR karena tidak terpakai dalam komunikasi ini, sehingga untuk menghidupkan modem maka DTR harus kita enable (ON) dahulu dengan perangkat lunak yang ada. Gambar dari rangkaian regulator dapat dilihat pada Gambar 3.4.

Untuk menyesuaikan input dan output data perlu ditambahkan komponen inverter karena kita ketahui untuk output data dengan logika '1' atau dalam istilah RS232 adalah 'Mark' mengeluarkan tegangan antara -5V hingga mendekati -15V sedangkan tegangan input pada TxD adalah +2V hingga mendekati  $V_{DD}$  demikian pula dengan sebaliknya. Perbedaan dari tegangan antara RS232 dengan karakteristik input /output TCM 3105 dapat dilihat pada Tabel 3.1.

Tabel 3.1 Perbedaan tegangan input dan output antara RS232 dengan TCM3105

	Kondisi	RS 232	TCM3105
TxD	'1' (mark)	-5 s/d -15	2 s/d 5
	'0' (space)	+5 s/d +15	0 s/d 0,8
RxD	'1' (mark)	-3 s/d -15	2,4 s/d 5
	'0' (space)	+3 s/d +15	0 s/d 0,4

Inverter yang digunakan adalah 74HC04 karena IC CMOS inverter ini mempunyai dioda pada inputnya untuk proteksi tegangan input yang terlalu besar.

Kaki TRS, TxR<sub>1</sub> dan TxR<sub>2</sub> untuk pengaturan mode dan kecepatan dari modem yaitu karena modem disesuaikan dengan mode Bell 202 dengan mode transmit 1200

baud dan receive 1200 baud (dengan frekuensi pengiriman dan penerimaan sama / half duplex) maka kaki TRS dihubungkan dengan -CLOCK kaki TXR<sub>1</sub> dan TXR<sub>2</sub> dihubungkan dengan Ground (low) .

Untuk input analog dari pesawat radio diberi pembatas tegangan dengan memberi sepasang dioda yang dipasang berlawanan karena hanya dibutuhkan tegangan sekitar 0,3 V (maksimum 0,7 V). Untuk pengaturan volume radio jangan terlalu besar karena malah dapat menimbulkan data cacat.

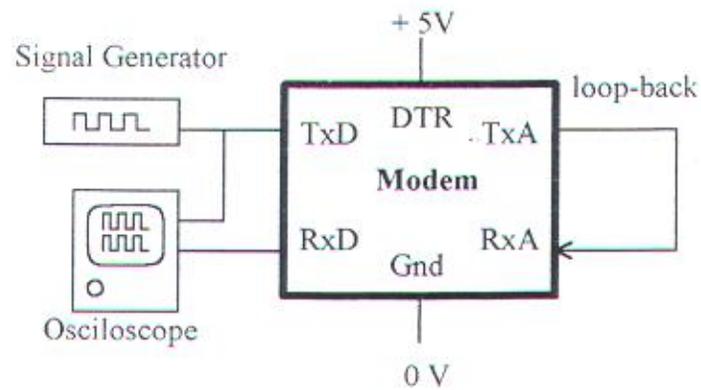
Untuk pengaturan tegangan level pada CDL disarankan pada data manual sekitar 3,3 V. Sedangkan untuk RxB diatur sekitar 2,7 V atau dapat dites dengan mengadakan loop-back pada TxA ke RxA sehingga didapatkan output RxD yang terbaik.

Penyambungan RS232 untuk penerima dan pengirim adalah sama yang berbeda terletak pada penyambungan RxA dan TxA. Pada pemancar yang dimasukkan pada pesawat pemancar adalah TxA. Sedangkan pada penerima yang dipasangkan pada pesawat penerima radio FM adalah RxA.

### **3.3 KALIBRASI MODEM**

Untuk kalibrasi yang persisi dibutuhkan Signal Generator dan Oscilloscope. Signal generator diset untuk menghasilkan gelombang persegi dengan frekuensi 600 Hz dan dimasukkan pada TxD. TxA dan RxA saling dihubungkan, kemudian output dari RxD dimasukkan pada input channel 1 oscilloscope dan output dari Signal generator dimasukkan pada channel 2 oscilloscope. Untuk kalibrasi dilakukan dengan

mengatur Variabel Resistor CDL dan Variabel Resistor RXB sampai didapatkan output kedua pola yang hampir sama dan sedikit cacat.



Gambar 3.5 Kalibrasi Modem

## BAB IV

### PERENCANAAN DAN PEMBUATAN PERANGKAT LUNAK

---

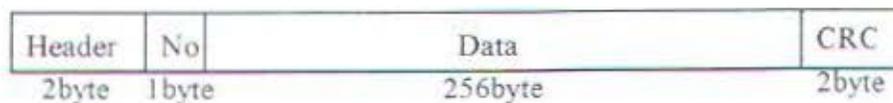
Pada pembuatan perangkat lunak penulis menggunakan Borland Delphi 2.0 Developer dengan sistem operasi windows 95. Perangkat lunak yang dibuat ada dua macam yaitu untuk pengirim dan untuk penerima penerima.

#### 4.1 PROTOKOL KOMUNIKASI

Dalam sistem komunikasi data perlu diatur protokol yang sesuai dengan data yang akan dikirimkan. Karena data yang dikirimkan berupa file yang tidak hanya teks maka penulis membuat dengan sistem paket dengan panjang data tetap. Hal ini diambil karena dengan sistem paket dengan panjang tetap ini pengembangan dan pembuatannya lebih mudah.

##### 4.1.1 Paket

Paket data yang direncanakan menggunakan panjang keseluruhan 261 byte dimana dibagi menjadi 4 bagian yaitu 'Header' dengan panjang 2 byte, 'No' dengan panjang 1 byte, 'Data' dengan panjang 256 byte dan 'CRC' panjang 2 byte seperti pada Gambar 4.1. dan keterangan header dapat dilihat pada Tabel 4.1.



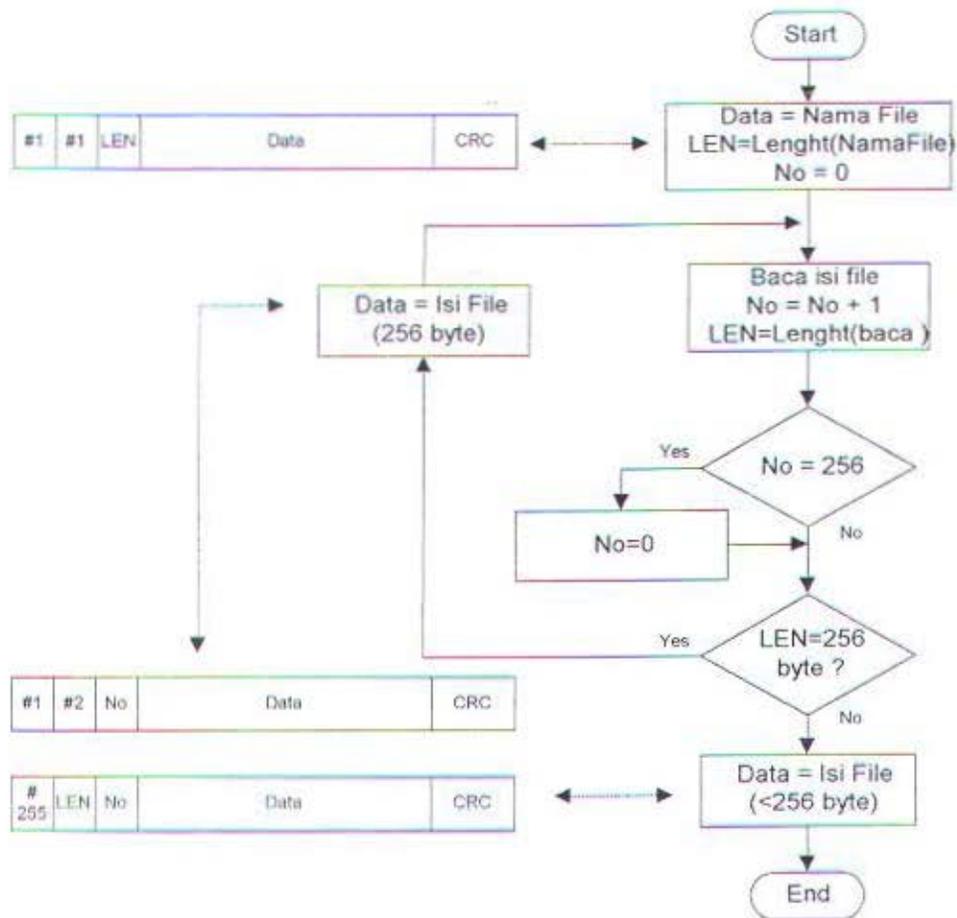
Gambar 4.1 Struktur Paket Data

Tabel 4.1 Keterangan Isi Paket Header

Header		No	Data	
1	2			
#1	#1	LEN	Nama file ( $\leq 254$ byte)	Total paket(2)
#1	#2	No urut frame (#1-#255)	Isi dari file (256 byte)	
#1	#2	Lanjutan frame (#0)	Isi dari file (256 byte)	
#255	LEN	No urut frame	Sisa isi file ( $\leq 256$ byte)	

LEN = panjang byte data

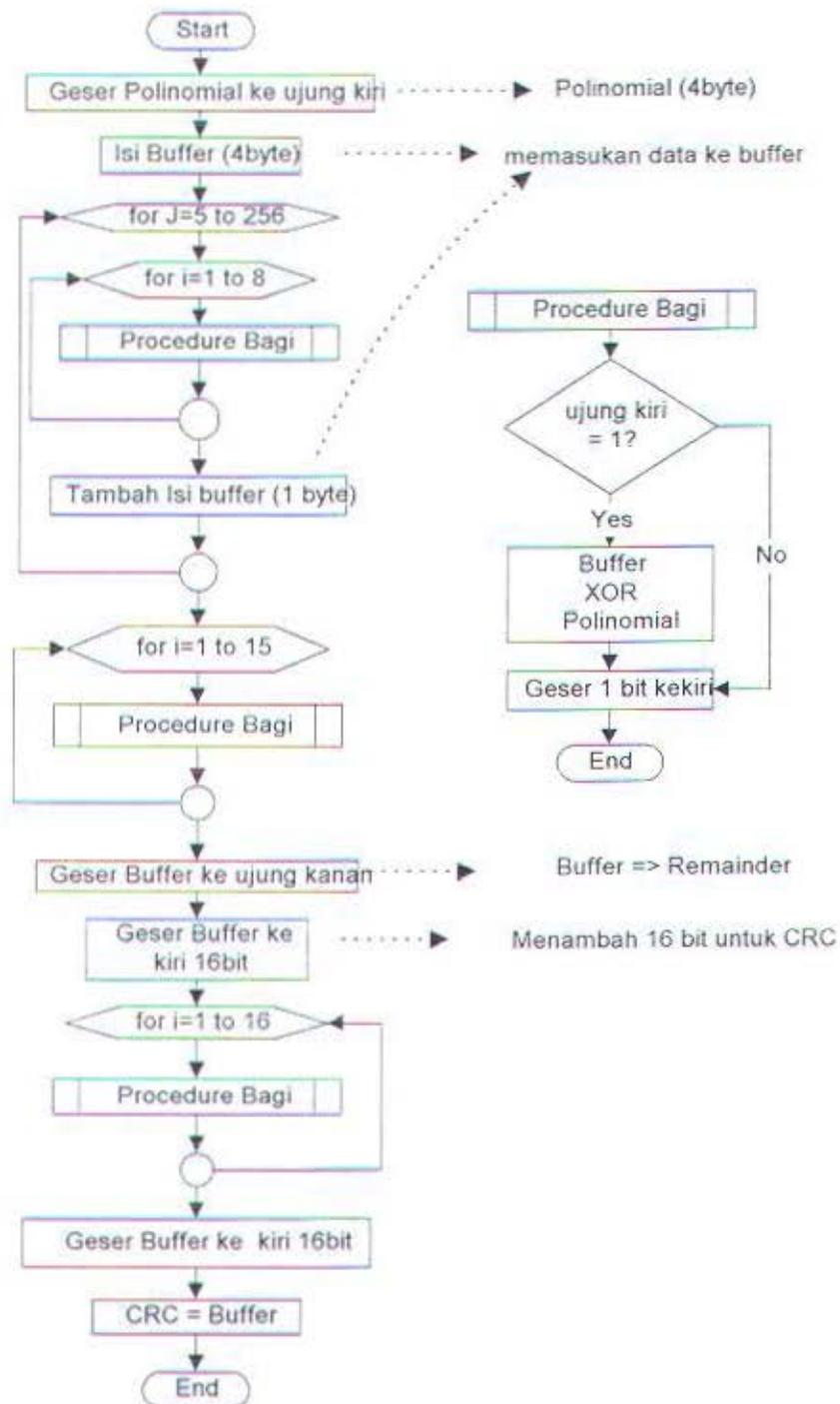
Untuk memulai pengiriman suatu file maka protokol akan mengirim nama dari file yang akan dikirim dengan cara 'Header' akan berisi karakter #1 (SOH) pada kedua byte tersebut. 'No' disini berfungsi sebagai penentu panjang dari isi data (panjang nama file) yang sesungguhnya. 'Data' berisi nama file dari data yang akan dikirimkan dan 'CRC' akan dihitung oleh bagian penghitung CRC sesuai dengan isi dari Data. Selanjutnya adalah mengirim isi dari file yaitu dengan mengirim 'Header' dengan karakter #1 dan #2, 'No' diisi dengan urutan Data yaitu karakter #1 untuk awal data, 'Data' diisikan dengan isi dari file dan 'CRC' akan dihitung pula. Bila sudah pada akhir dari isi file maka 'Header' akan diisi dengan karakter #255 sedangkan pada byte kedua akan diisi dengan panjang dari sisa data, 'Data' akan diisi dengan sisa file data, 'CRC' dihitung juga sesuai dengan isi 'data'. Urutan pengiriman file data dapat dilihat pada Gambar 4.2.



Gambar 4.2 Urutan Paket

#### 4.1.2 Cyclic Redudancy Check

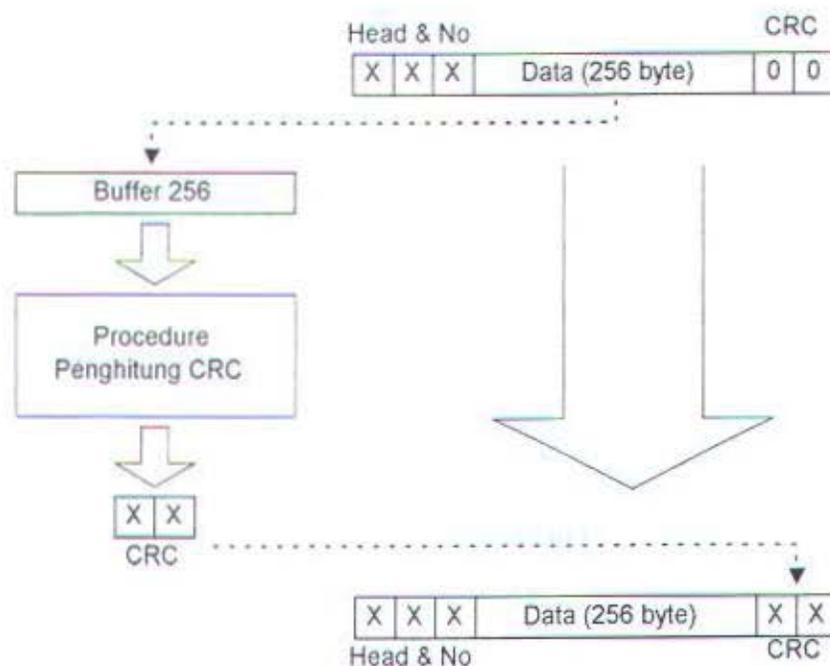
Seperti yang telah dijelaskan bahwa penghitungan CRC dilakukan dengan bantuan modulo 2 dengan operasi XOR pada penjumlahan maupun pengurangannya. Dari Algoritma tersebut maka dapat dibuat program sesuai diagram alir pada Gambar 4.3.



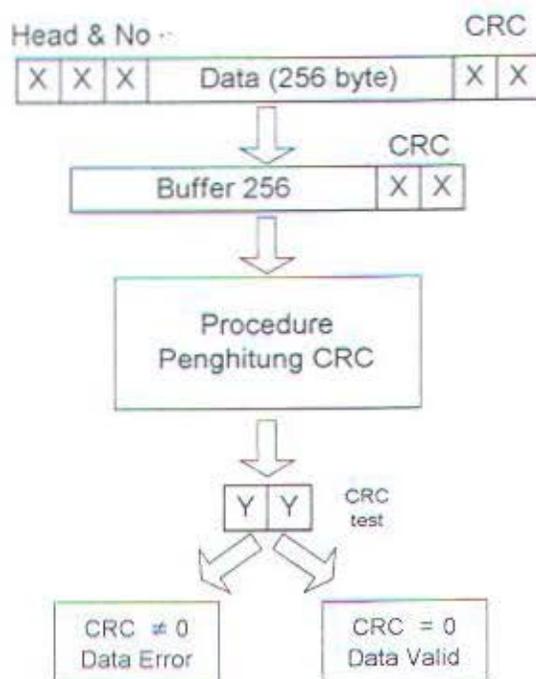
Gambar 4.3 Diagram Alir Program CRC

Penambahan procedure CRC dilakukan ketika data yang telah dimasukan pada paket akan dikirim yaitu dengan menambahkan 2 byte CRC pada akhir paket. Pada bagian penerima procedure CRC ditambahkan ketika paket telah diterima dan dideteksi kode pakatnya.

Terdapat sedikit perbedaan pada bagian penerimaan CRC dengan bagian pengiriman CRC yaitu data yang dimasukan ke buffer penghitung CRC pada bagian pengirim yaitu data saja (256 byte) dan outputnya dimasukan pada block CRC sebagai kode cek error, sedang pada bagian penerima data yang dimasukan pada buffer penghitung CRC ditambahkan dengan block CRC. Jadi panjang buffer pada penerima sebesar 258 byte (besar data ditambah besar CRC), sehingga outputnya berupa kondisi dari errornya yaitu bila bukan 0 maka ada error. Gambar 4.4 dan Gambar 4.5 menjelaskan bagan proses CRC pada penerima dan pengirim tersebut.



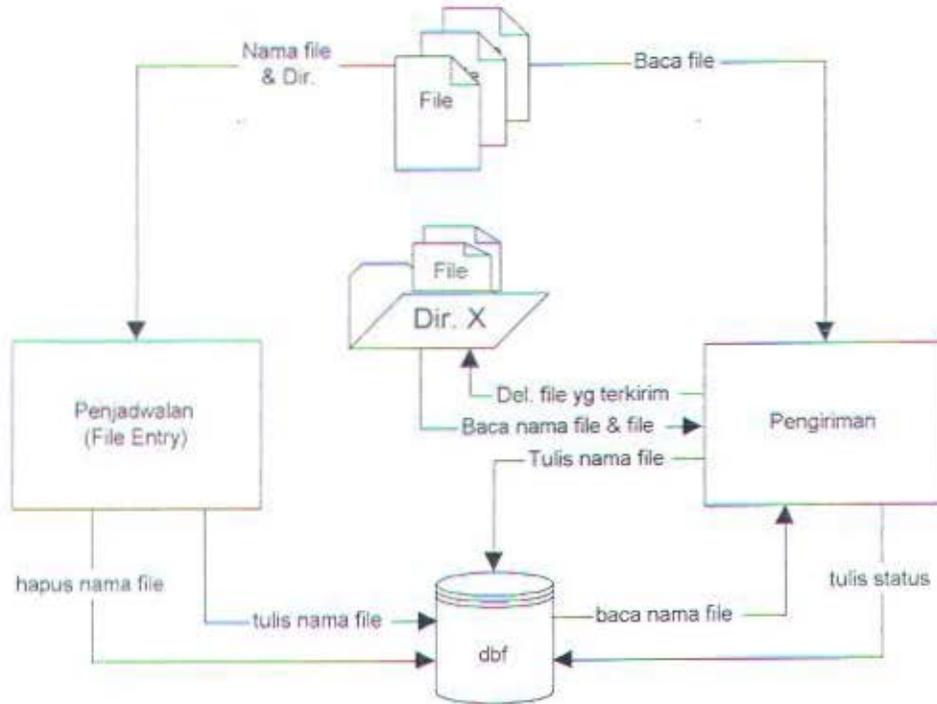
Gambar 4.4 Penambahan CRC di pengirim



Gambar 4.5 CRC test di penerima

## 4.2 BAGIAN PENGIRIM

Pada bagian pengirim terdapat dua aplikasi yaitu bagian bagian penjadwalan dan bagian pengiriman. Kedua bagian ini dihubungkan dengan sistem database dari nama file yang akan dikirim, bagian penjadwalan memasukan nama-nama file yang akan dikirim dan bagian pengiriman mengambil nama file tersebut satu persatu untuk dikirimkan melalui COM port.



Gambar 4.6 Hubungan antar Program dan File

Database berisi field yaitu : nama file & direktorinya, ukuran, status. Pertama kali file entry akan memasukan nama file, direktorinya dan ukuran dengan status mula queue (antrian) kemudian bila pengiriman dimulai maka bagian pengiriman akan membaca satu persatu tiap record dari awal, dan pengiriman akan membaca file yang berstatus queue saja dimana ketika ia baru membaca ia akan merubah status queue menjadi progress hal ini menunjukkan bahwa file tersebut dalam pengiriman. Apabila pengiriman telah selesai maka bagian pengiriman akan merubah status progress menjadi done. Demikian selanjutnya bagian pengiriman akan membaca lagi file dengan status queue yang lain. Sebagai penjelasannya secara bagan blok dapat dilihat pada Gambar 4.6.

Pada bagian pengiriman ini tiap file akan dipecah menurut paket-paket protokol yang telah disusun diatas dan pada tiap akhir paket akan ditambahkan CRC untuk deteksi error pada pengiriman.

Format data (1 byte pengiriman data) pada RS 232 dipilih yaitu 8N1 (8 bit data parity none dan stop bit 1). Bit parity pada RS 232 kami pilih none karena check parity bit efektifitasnya rendah (kemampuan mendeteksi error rendah) dan bila digunakan maka akan mengurangi kecepatan. Besar kerugian kecepatan yang terjadi bila bit parity ini digunakan adalah :

$$\frac{\text{bit parity}}{\text{total Bit}} * 100\% = \text{kerugian \%}$$

$$\frac{1}{(1\text{stopbit} + 8\text{databit} + 1\text{paritybit} + 1\text{stopbit})} * 100\% = 9\%$$

Dengan menggunakan CRC maka kemampuan mendeteksiannya tinggi dan kerugian akibat pemakaian CRC lebih rendah dibandingkan menggunakan bit check parity yaitu :

$$\frac{\text{byte CRC}}{\text{Total byte}} * 100 \% = \text{kerugian \%}$$

$$\frac{2}{(2 \text{ byte header} + 1 \text{ byte no} + 256 \text{ byte data} + 2\text{byteCRC})} * 100\% = 0,8\%$$

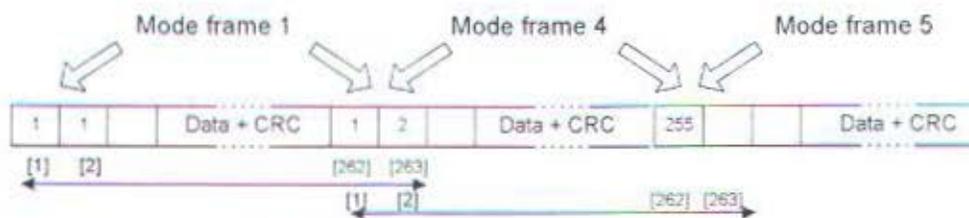
#### 4.3 BAGIAN PENERIMA

Pada bagian penerima, data yang diterima berupa paket-paket disusun kembali menjadi kesatuan file. Pengambilan paket pertama harus dikenali sebagai paket no 1

yaitu paket yang berisi nama file kemudian akan mengambil paket selanjutnya. Jadi bila paket no 1 tidak ditemukan maka penerima akan menunggu sampai paket no 1 diterima. Data yang diterima kemudian disimpan sementara dalam file sementara sampai ditemukan paket penutup. sehingga bila paket penutup ditemukan maka file sementara tersebut diganti dengan file dengan nama file yang ada pada paket no 1.

Untuk mengambil bentuk dari paket, kami menggunakan cara membandingkan juga dengan paket berikutnya. Mode frame didapat dari kombinasi paket yang sedang diterima dan paket header berikutnya.

Macam dari mode frame dan operasi yang dilakukan pada masing-masing mode frame dapat dilihat pada Tabel 4.2. sedang proses pengambilan mode frame dapat dilihat pada Gambar 4.7.



Gambar 4.7. Contoh Pengambilan mode frame

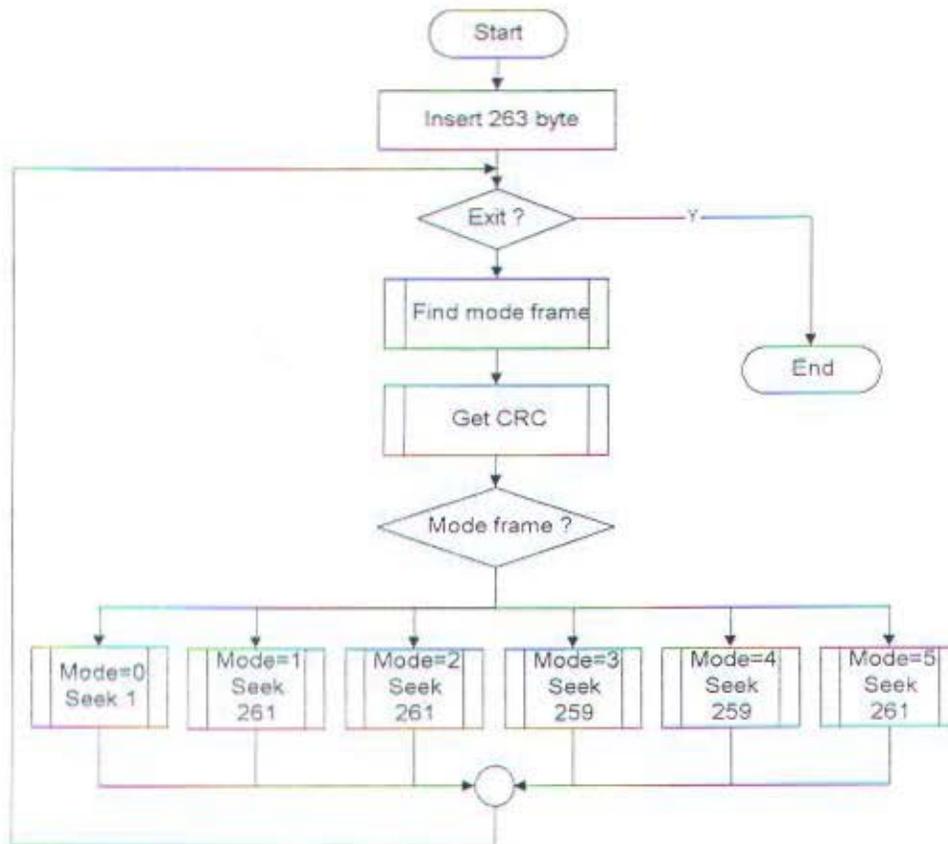
Dengan menggunakan metode diatas maka kemungkinan kesalahan pembacaan header dapat diperkecil dibanding hanya membaca header yang didepan saja. Untuk pengeseran pada mode 4 dan 5 sebesar 259 byte dilakukan untuk mencegah bilamana data yang diterima merupakan akhir dari pengiriman data

(seterusnya) maka data tidak akan menunggu terus data berikutnya yang kosong terus pada mode frame 5 (bilamana tidak maka file terakhir tidak akan pernah dihasilkan).

Tabel 4.2 Tabel mode frame dan Operasi yang dilakukan

Mode	Deteksi	Seek berikutnya	Operasi
1	[1]=#1 [2]=#1 [262]=#1 [263]=#2	261	- Write enable - Open file sementara - Simpan nama file asli
2	[1]=#1 [2]=#2 [262]=#1 [263]=#2	261	- Tulis data ke file sementara (merupakan paket type 2)
3	[1]=#1 [2]=#1 [262]=#255	259	-Seperti mode 1
4	[1]=#1 [2]=#2 [262]=#255	259	-Tulis data ke file sementara
5	[1]=#255	261	-Tulis data ke file sementara - Close file sementara - Rename file sementara dengan nama file asli - Write Disable
0	selain diatas	1	(error patern mode frame)

Diagram blok dari penerimaan dengan menggunakan mode frame dapat dilihat dalam diagram blok 4.8.



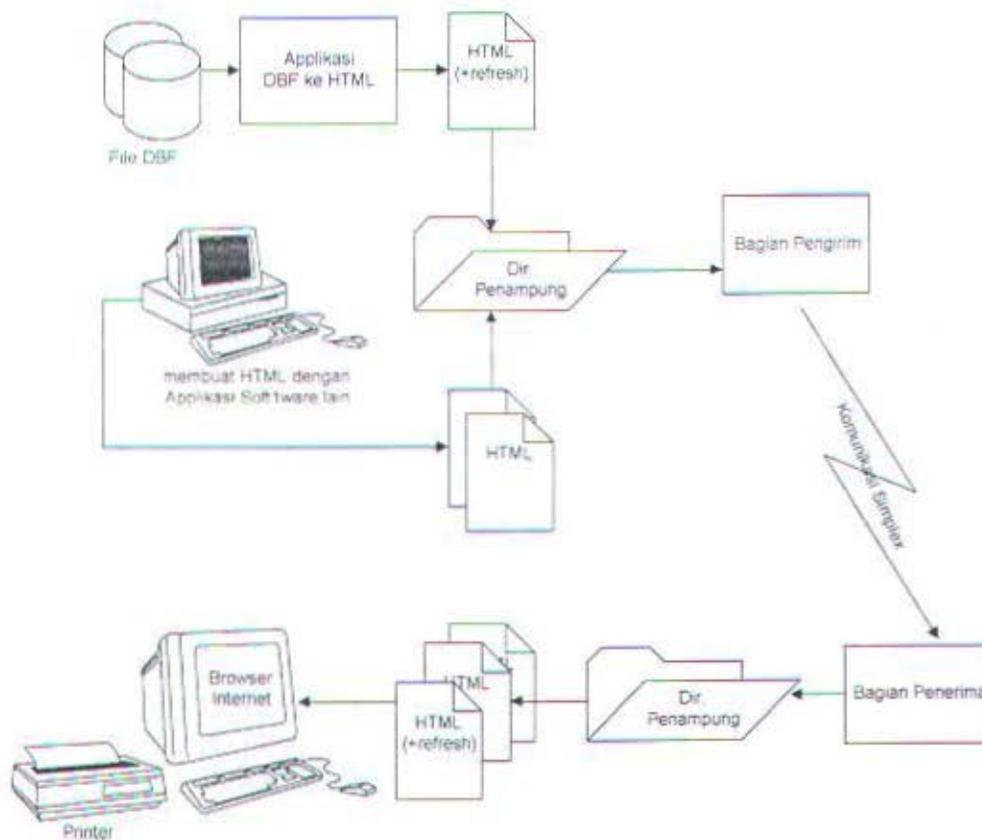
Gambar 4.8 Diagram Blok Penerimaan menurut mode framenya

File sementara digunakan sebagai buffer sementara pada saat penerimaan data demi data, karena kita ketahui bahwa operasi penulisan yang lama akan mengunci file yang mungkin akan dibaca oleh aplikasi lain. Dengan adanya file sementara maka waktu untuk mengunci semakin kecil (pada saat operasi rename lock file sangat cepat).

#### 4.4 CONTOH PROGRAM APLIKASI DATABASE

Sebagai contoh model aplikasi dibuat tampilan dari pembacaan suatu data base secara berkala dengan prinsip kerjanya adalah file DBF dibaca dan ditulis ke dalam bentuk format HTML secara berkala sehingga pada penerima data HTML tersebut dapat direfresh dalam periode tertentu.

Program ini dapat diterapkan pada sistem jadwal kedatangan dan keberangkatan pesawat terbang, daftar harga komediti, bursa efek dan lainnya tanpa harus campur tangan memasukan data tersebut secara berkala.



Gambar 4.9 : Gambaran Lengkap Contoh Aplikasi

#### 4.5 STRUKTUR PENAMPILAN HTML

Untuk mempercantik penampilan maka digunakan frame dimana jendela browser dibagi menjadi 3 frame yaitu frame atas dengan nama 'HEAD', frame bawah kiri dengan nama 'INDEX' dan frame bawah kanan dengan nama 'ISI FILE' seperti pada Gambar 4.10



Gambar 4.10 Pembagian Frame

Untuk bagian 'HEAD' digunakan sebagai space iklan dimana pada selang waktu 1 menit frame 'HEAD' ini akan menampilkan file secara berurutan dengan nama 1.txt, 2.txt, 3.txt , 4.txt dan 5.txt. Bagian 'INDEX' akan menampilkan file 'menuindex.txt' file ini dibuat oleh bagian program penerima (modem rx) dimana ia link index akan ditambahkan hanya bila file dengan extension html atau htm yang diterima. Pada bagian 'INDEX' ini secara otomatis file akan direfresh dalam waktu 30 detik sehingga bila ada perubahan pada link maka akan ditampilkan oleh proses refresh tersebut.

Untuk bagian 'ISI FILE' merupakan link yang dituju dari penekanan pada frame 'INDEX' sehingga isi dari frame 'ISI FILE' ditentukan oleh click pada link index.

Browser akan menerapkan frame tersebut bila pada awal menjalankan browser IE 3.0 diload file 'FIRSTMENU.TXT' setting ini dapat dilakukan pada setting RX seperti pada Lampiran 1 (Gambar 3.3) File tersebut yang digunakan untuk memperbaiki tampilan diatas. Isi dari file 'FIRSTMENU.TXT' dapat dilihat sebagai berikut :

```
<HTML>

<SCRIPT Language="JavaScript">

var i=0

function AutoLoad()

{i++

if (i==0) {WINDOW.OPEN("1.txt", "HEAD")};

if (i==1) {WINDOW.OPEN("2.txt", "HEAD")};

if (i==2) {WINDOW.OPEN("3.txt", "HEAD")};

if (i==3) {WINDOW.OPEN("4.txt", "HEAD")};

if (i==4) {WINDOW.OPEN("5.txt", "HEAD")};

if (i==5) {i = 0;WINDOW.OPEN("HEAD.TXT", "HEAD")};

timerID=setTimeout("AutoLoad()", 60000)}

timerID=setTimeout("AutoLoad()", 60000) </SCRIPT>

<FRAMESET ROWS="100,*"><FRAME SRC="HEAD.TXT" NAME="HEAD">

<FRAMESET COLS ="25%,75%"><FRAME SRC="menuindex.txt" NAME="INDEX">

<FRAME SRC="isifile.txt" NAME="ISIFILE"></FRAMESET></FRAMESET>

</HTML>
```

## BAB V

### PEMASANGAN ALAT DAN HASIL AKHIR

---

#### 5.1 PEMASANGAN MODEM

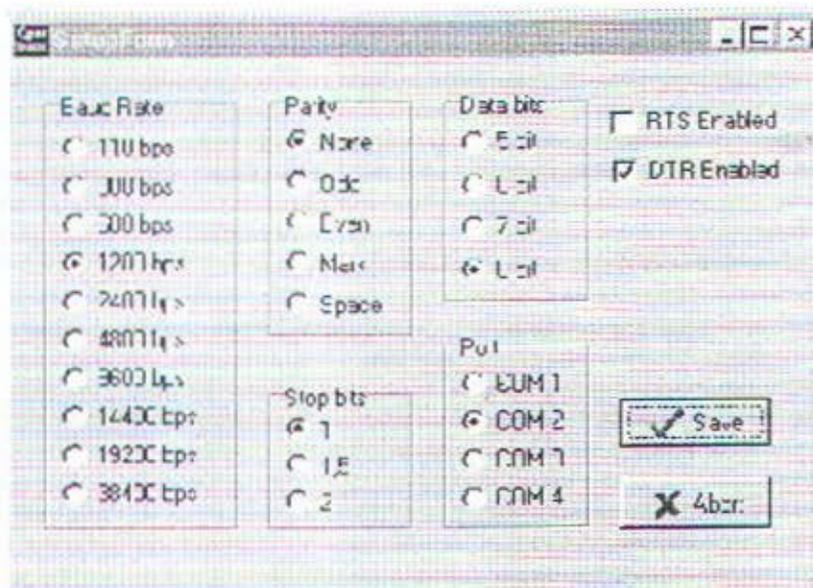
Dalam pemasangan modem ada beberapa hal yang perlu diperhatikan yaitu COM Port (Serial port) harus ada yang tersisa untuk dipakai oleh modem tersebut dan perlu diketahui COM Port yang dipasang adalah COM Port nomor berapa. Biasanya sebagian besar COM 1 telah digunakan mouse sehingga kemungkinan yang tersisa adalah COM 2 (sehingga COM 2 kami gunakan sebagai default setting)

Kemudian dilanjutkan dengan pemasangan pada pemancar atau pada penerima FM. Untuk bagian pemancar, modem yang dipasangkan ke input pemancar FM adalah TxA. Pada bagian penerima, modem dipasangkan pada output audio radio adalah RxA. Untuk bagian penerima, gelombang radio diset pada gelombang pemancar dan volume radio jangan terlalu besar.

#### 5.2 PEMASANGAN SOFTWARE

##### 5.2.1 Pemasangan Bagian Penerima

Pemasangan software yang terpenting adalah menyesuaikan setting dari modem dan konfigurasi pemancar. Setting tersebut dapat dilakukan pada form setup, pada form tersebut akan terdapat beberapa option yang harus disesuaikan yaitu :



Gambar 5.1. Program Setting

- Baud rate : merupakan penyesuaian setting kecepatan modem dan data yang dikirimkan. pada tugas akhir ini kecepatan yang digunakan adalah 1200 bps
- Parity : merupakan cek kontrol kesalahan dari bit cek parity, dipilih none.
- Stop bits : merupakan banyak dari bit stop, dipilih stop bit 1
- Data bits : merupakan banyaknya bit data, dipilih 8 bit
- Port : merupakan tempat COM yang dipasang modem, ini tergantung pada COM yang kosong yang diisi oleh modem.
- RTS : merupakan setting dari Pin RTS, tidak usah digunakan

- DTR : merupakan setting dari pin DTR, karena modem kita menggunakan tegangan dari DTR maka harus dipilih agar modem mendapat sumber tegangan.

Setelah setting telah benar maka selanjutnya adalah menyalakan program penerima bila pemasangan COM salah maka monitor open status akan mati dan juga modem tidak akan menyala. Jika sudah benar maka monitor RX akan menyala bila ada sinyal data masuk.

Selanjutnya adalah tinggal menghidupkan Browser Internet yang disini kami pilih adalah Microsoft Internet Explorer (MSIE) karena software ini sudah merupakan standar bagi sistem operasi Windows 95.

Bila pada awal penghidupan software penerimaan terdapat error pada beberapa saat (ditandai dengan hidupnya lampu monitor error) merupakan hal yang wajar karena masih menyesuaikan dengan bit frame. Namun bila sudah lama (lebih dari 1 menit) maka kemungkinan gelombang radio belum tepat atau terkena noise komputer. Agar lebih optimal kami menyarankan untuk menjauhkan antena radio dari komputer (lebih baik bila gunakan antena luar).

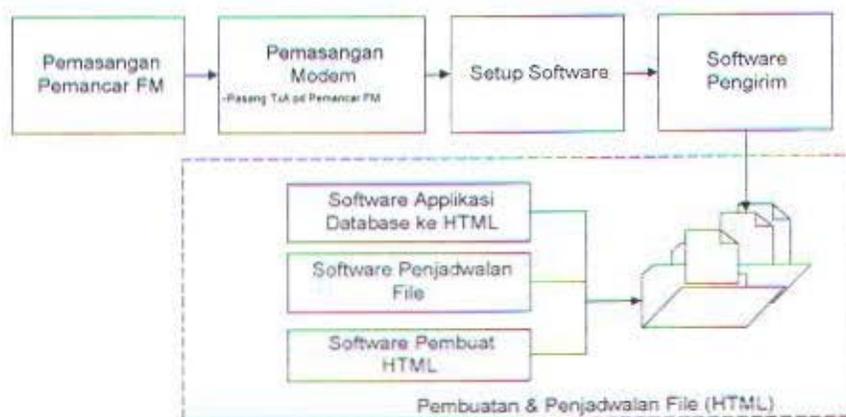


Gambar 5.2. Diagram pemasangan pada penerima

### 5.2.2 Pemasangan Bagian Pengirim

Pemasangan modem dan Setup Software pada pengirim hampir sama dengan pada penerima yang berbeda hanya pada pemasangan ke bagian pemancar FM. Output TxA dihubungkan pada input pemancar FM (pada bagian penerima yang digunakan RxA) kemudian bila sudah benar settingnya maka dilanjutkan pada software pengirim.

Pembuatan file HTML dapat dilakukan sebelum pengiriman dan hasilnya dimasukan pada direktori yang sudah ditentukan sehingga bila pengirim sudah siap maka semua file tersebut akan langsung terjadwal untuk dikirimkan.



Gambar 5.3 Diagram Pemasangan pada Pemancar

## 5.3 PENGUKURAN JANGKAUAN PEMANCAR

Jarak jangkauan antara pemancar dan penerima secara teoritis sangat dipengaruhi oleh berbagai kondisi yaitu :

- Daya Pemancar

- Kondisi alam sekitar (ada halangan atau tidak).
- Kondisi pesawat penerima.
- Gangguan / Noise dari luar.

Dalam pengujian jarak jangkauan ini penulis menggunakan spesifikasi peralatan pemancar dan penerima sebagai berikut :

1. Bagian Pengirim :

- Menggunakan Pemancar FM VCO sederhana
- Daya yang digunakan 50 mW
- Antena Teleskopik 75 cm terpasang langsung pada pemancar FM
- Catu daya 12 V

2. Bagian Penerima

- Menggunakan penerima radio saku FM Mono

Hasil dari pengujiannya dapat dilihat pada Tabel 5.1

Tabel 5.1. Pengujian jangkauan pemancar

No	Jarak	Kondisi	Suara	Hasil Pengukuran
1	± 10 m	tidak terhalang	bagus	Tanpa Error
2	± 20 m	terhalang tembok	bagus	Tanpa Error
3	± 30 m	terhalang tembok	Ada sedikit desah	Tanpa Error
4	± 40 m	terhalang tembok	Desah	Beberapa Error
5	± 50 m	terhalang tembok	Tak terdengar	Error

Untuk memperluas jarak jangkauan yang mungkin dilakukan adalah memperbesar daya pemancar dan mempertinggi antena pemancar dan penerima. Kesesuaian impedansi dari pemancar, kabel dan antena pemancar perlu diperhatikan bila daya pemancar besar. Bila tidak terjadi kesesuaian maka daya pemancar tidak bisa diradiasikan seluruhnya sehingga ada daya yang kembali pada pemancar ini menyebabkan pemancar akan menghasilkan panas yang berlebihan dan bisa merusak pemancar. Sehingga diperlukan Power SWR meter untuk melihat matching impedansi pada pemancar tersebut sehingga efisiensi daya yang diradiasikan besar.

## BAB VI

### PENUTUP

---

Setelah uji coba dilakukan, diperoleh beberapa kesimpulan dan saran yang berguna untuk pengembangan dan pemanfaatannya. Walaupun pada uji coba belum bisa memberikan bentuk pemanfaatan yang kongkret namun diperoleh beberapa kesimpulan berikut :

1. Pengiriman data dengan mode simplex memberikan peluang yang sangat lebar sebagai media baru dalam komunikasi data ataupun informasi yang sangat murah.
2. Dengan menggunakan IC modul TCM 3105 memudahkan dalam membuat modem FSK, dan dalam mengeset dan mengkalibrasi lebih mudah karena telah ada kristal standar.
3. Sistem operasi windows 95 banyak memberikan kemudahan dalam pemrograman dan pemakiannya.
4. Hasil keseluruhan dari pembuatan sistem komunikasi dengan metode simplek ini dapat bekerja dengan baik. namun utility pelengkap lain perlu dikembangkan dengan penyesuaian pemakaian yang diinginkan.

Berbagai saran untuk pengembangan dan perbaikan unjuk kerjanya adalah sebagai berikut :

1. Pembuatan sistem kompresi sangat dibutuhkan terlebih untuk pengiriman jenis file HTML ini rasio kompresinya sangat tinggi (dengan kompresi PKZIP bisa menaikkan kecepatan hingga kurang lebih 6 kali).
2. Perlu pengembangan format HTML dengan menggunakan Java Applet sehingga tampilan bisa lebih bervariasi dan fleksibel.
3. Dengan bekerja sama dengan pihak penyaji informasi berita maka sistem simplek ini merupakan ladang baru dalam bisnis informasi dan periklanan yang memasyarakat.
4. Perlu dipikirkan pembuatan modem yang lebih cepat agar informasi yang dikirimkan dapat lebih cepat dan beragam. Bahkan bila kecepatannya sudah bisa menyamai kecepatan modem yang beredar dipasaran maka siaran multimedia dapat dilangsungkan.
5. Dengan mengkoneksikan dengan internet maka beberapa aplikasi dapat di distribusikan dengan sistem simplex ini seperti Mailling List, News Server dan internet dapat sebagai penghubung dengan sumber berita (koran atau iklan).

Demikian kesimpulan dan saran yang dapat disampaikan semoga dapat bermanfaat bagi kita semua.

## DAFTAR PUSTAKA

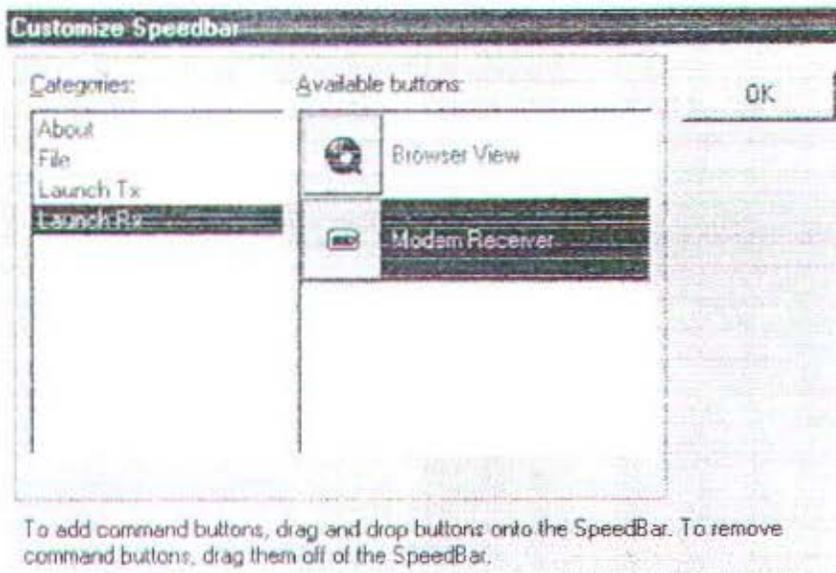
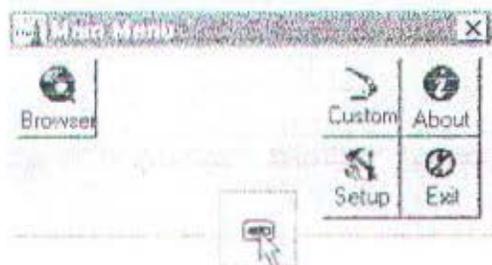
- Arianto Widyatmo dkk. BELAJAR MIKROPROSESOR - MIKRO KONTROLER MELALUI KOMPUTER PC. Jakarta : PT. Elex Media Komputindo Kelompok Gramedia, 1994
- Campbell, Joe. SERIAL COMMUNICATION : C PROGRAMMER'S GUIDE. Indiana : SAMS Publishing, Second Edition, 1994
- Halsall, Fred. DATA COMMUNICATIONS COMPUTER NETWORKS AND OPEN SYSTEM , England : Addison-Wesley Publishing Company, Third Edition, 1993.
- Sampurna, MEMBUAT HOME PAGE DENGAN HTML. Jakarta : PT. Elex Media Komputindo Kelompok Gramedia, 1997.
- Texas Instruments, DATA MANUAL : FSK MODEM , Texas : Texas Instruments Inc, 1991.
- Green, DC. KOMUNIKASI DATA ,Yogyakarta : Penerbit ANDI Yogyakarta, 1996.
- Wozniewicz, Andrew with Shanimas, Namir. BORLAND DELPHI IN 21 DAYS . Indiana : SAMS Publishing, First Edition, 1995.

## LAMPIRAN I

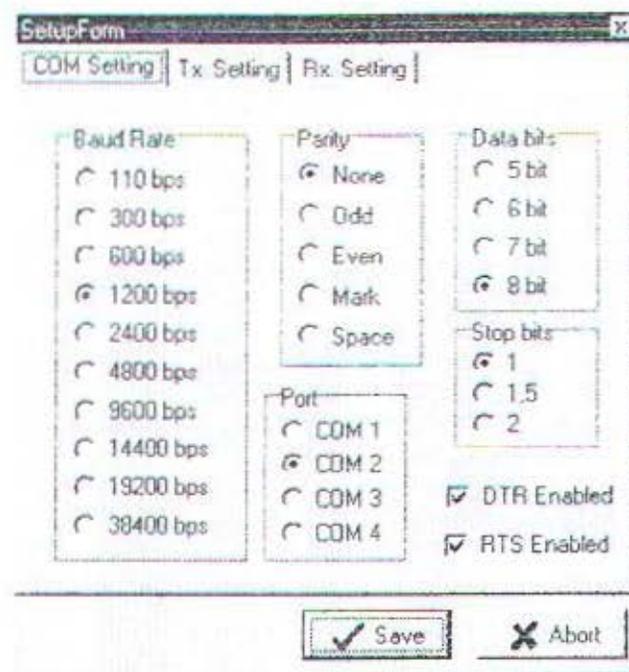
Hasil dan Tampilan Program TeleText for PC (TelePC)



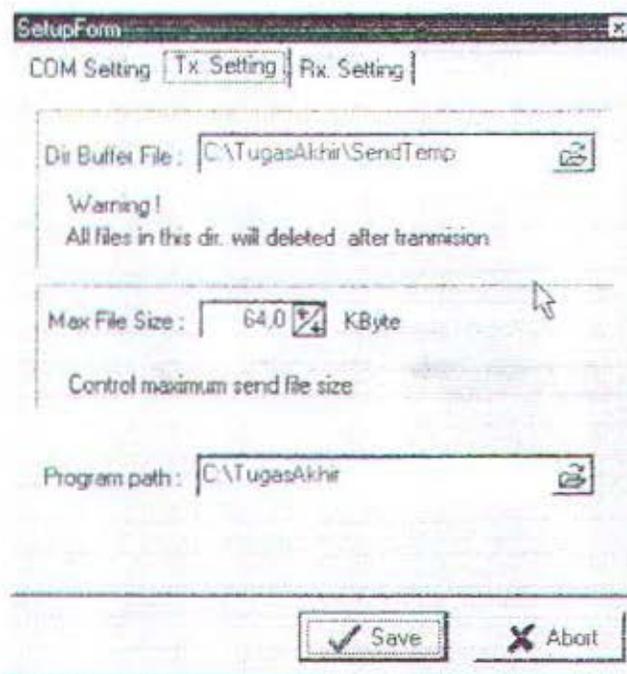
Gambar 1 : Tampilan Menu Utama



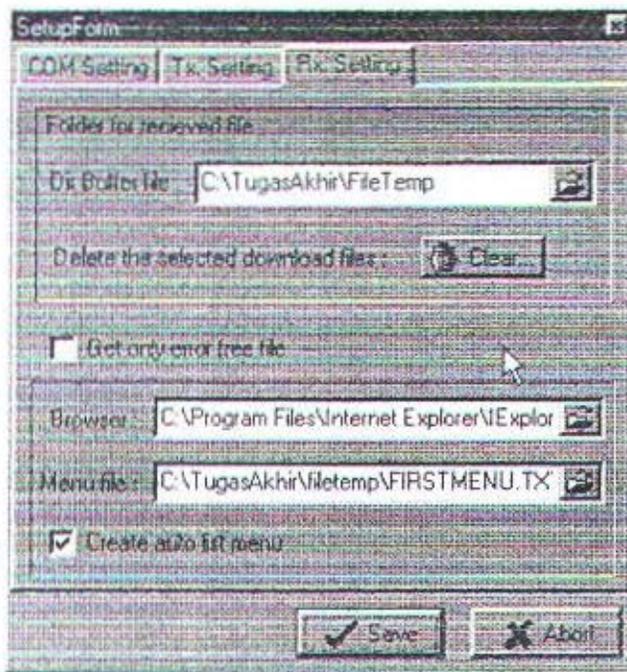
Gambar 2 : Tampilan Custom Menu



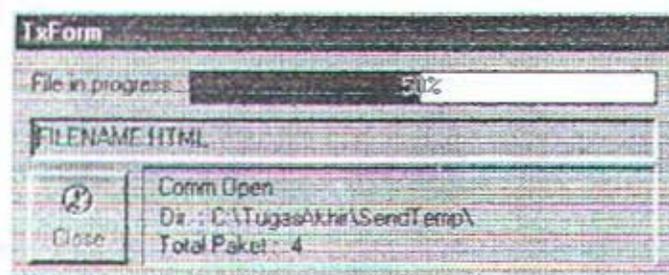
Gambar 3.1 : Tampilan Setup COM



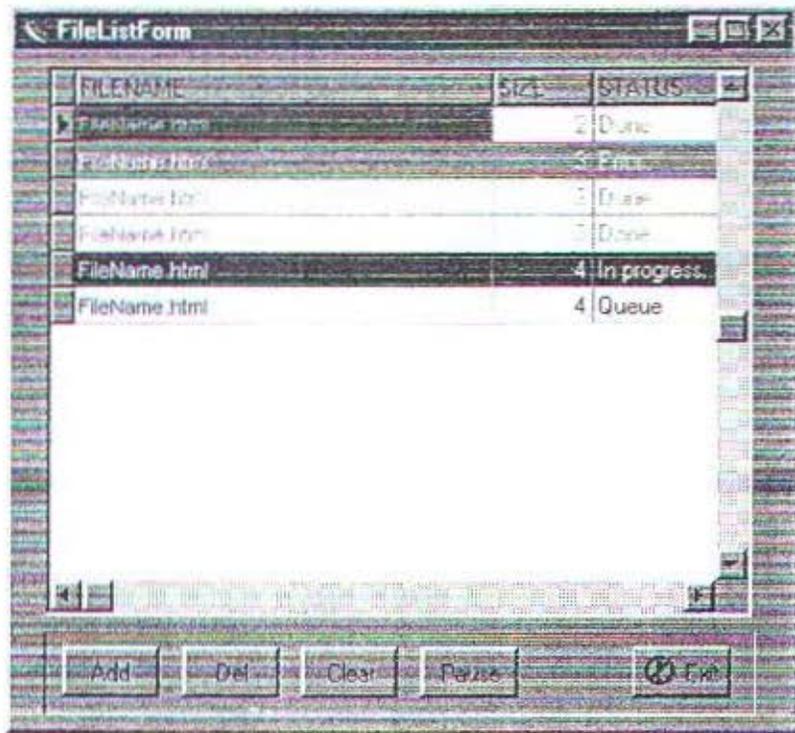
Gambar 3.2 : Tampilan Setup Transceiver (Tx)



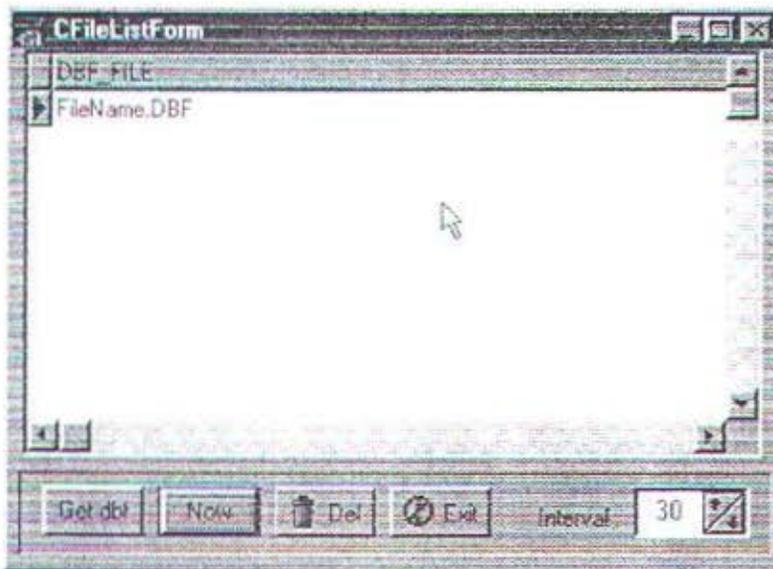
Gambar 3.3 : Tampilan Setup Receiver (Rx)



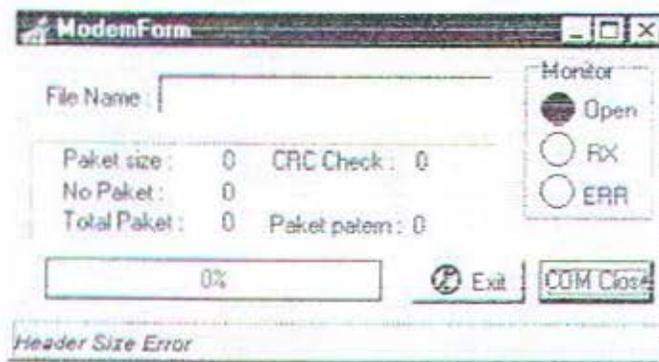
Gambar 4.1 : Tampilan Pengirim (Tx)



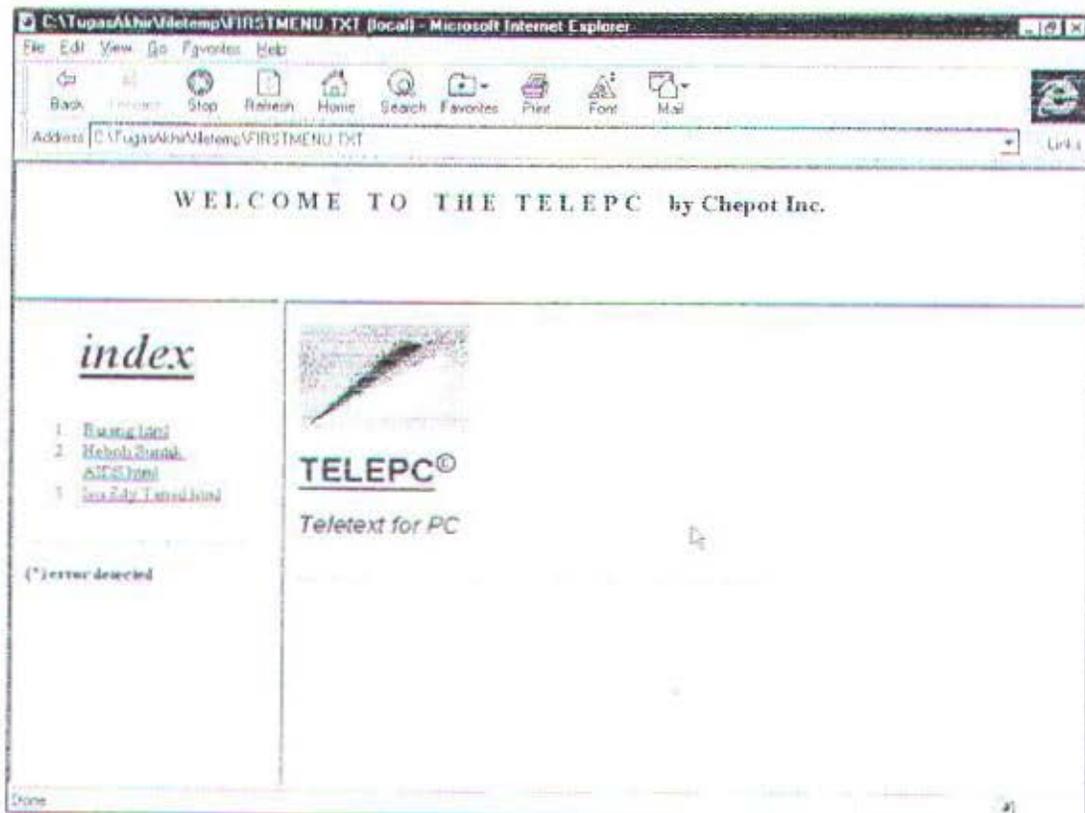
Gambar 4.2 : Tampilan List / Penjadualan (Tx)



Gambar 4.3 : Tampilan AutoCreate DBF to HTML (Tx)



Gambar 5 : Tampilan Penerima (Rx)



Gambar 6 : Tampilan pada Internet Explorer 3.0

20 SEP 1996

**JURUSAN TEKNIK ELEKTRO  
FAKULTAS TEKNOLOGI INDUSRI  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER**

**TUGAS AKHIR - EL 1799**

**Nama Mahasiswa :** Tjipto Prakosa  
**Nomor Pokok :** 2291100031  
**Bidang Studi :** Sistem Teknik Komputer  
**Tugas Diberikan :** 23 September 1996  
**Tugas Diselesaikan :**  
**Dosen Pembimbing :** 1. Ir. Zainal Alim  
2. Ir. Hanny Budinugroho  
**Judul Tugas Akhir :** PERENCANAAN DAN PEMBUATAN SISTEM  
TELETEXT UNTUK PC DENGAN MENGGUNAKAN  
SISTEM KOMUNIKASI RADIO SIMPLEX

**Uraian Tugas Akhir :**

Dalam tugas akhir ini akan dirancang suatu sistem perlengkapan pemancar dan penerima yang dihubungkan pada suatu PC sehingga dapat mengirimkan data text secara searah dari satu pemancar ke banyak penerima seperti pada sistem radio broadcast. Dengan bantuan modulator pada bagian pemancar dan demodulator pada bagian penerima maka jalur radio komunikasi simplex biasa dapat dijadikan jalur data searah dengan biaya operasional yang sangat murah sehingga dapat memasyarakat.

Surabaya, September 1996

Menyetujui

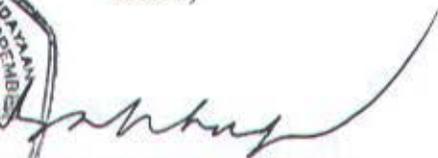
Dosen Pembimbing I,  
  
Ir. Zainal Alim  
NIP. 131532037

Dosen Pembimbing II,  
  
Ir. Hanny Budinugroho  
NIP. 131651433

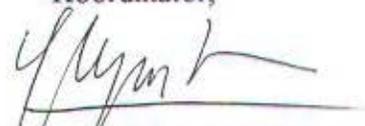
Mengetahui,

Jurusan Teknik Elektro  
Ketua,



  
Moch. Salehudin, MEng.Sc  
NIP. 130532026

Bidang Studi  
Sistem Teknik Komputer  
Koordinator,

  
Ir. Yoyon K. Suprpto, MSc  
NIP. 130687439

# USULAN TUGAS AKHIR

## 1. JUDUL TUGAS AKHIR :

PERENCANAAN DAN PEMBUATAN SISTEM  
TELETEXT UNTUK PC DENGAN MENGGUNAKAN  
SISTEM KOMUNIKASI RADIO SIMPLEX

## 2. BIDANG STUDI :

Teknik Sistem Komputer

## 3. RUANG LINGKUP :

- Teknik Interfacing
- Jaringan Komputer

## 4. LATAR BELAKANG :

Kemajuan teknologi informasi telah membawa manfaat yang sangat besar, akan tetapi keberadaan sistem teknologi informasi yang ada sekarang seperti internet, LAN dan WAN dirasa sangat mahal bagi sebagian besar masyarakat kita. Dengan mewujudkan teknologi teletext pada PC dengan sistem simplex (radio broadcast) maka sistem teknologi informasi dapat dimasyarakatkan dengan biaya murah sekaligus tepat guna.

## 5. TUJUAN :

Merencanakan peralatan guna mengirimkan informasi berupa text ke beberapa PC dari sebuah pemancar dengan metode komunikasi simplex (radio broadcast).

## 6. PENELAAHAN STUDI :

Data yang berada pada komputer pemancar ditransmisikan ke pemancar radio dalam bentuk sinyal yang telah dimodulasikan. dan diterima oleh penerima radio biasa kemudian di demodulasikan

kembali menjadi data pada komputer penerima. Proses tersebut hampir menyerupai sistem komunikasi paket radio half duplex, hanya saja dibuat agar menjadi simplex dengan pertimbangan tertentu (perubahan protokol dan paritas).

#### 7. RELEVANSI :

Dengan pembuatan alat ini diharapkan dapat menjadi alternatif sistem teknologi informasi baru dengan biaya murah dan efektif sehingga dapat lebih memasyarakat.

#### 8. LANGKAH LANGKAH :

1. Studi Literatur
2. Penyusunan blok diagram rangkaian dan flowchart program
3. Pembuatan alat dan program
4. Pengujian dan penyempurnaan alat
5. Penulisan naskah tugas akhir

#### 9. JADWAL KEGIATAN :

Seluruh kegiatan direncanakan dalam waktu 6 bulan dengan perincian sebagai berikut :

No	KEGIATAN	BULAN					
		I	II	III	IV	V	VI
1	Studi Literatur						
2	Penyusunan blok diagram dan flowchart program						
3	Pembuatan alat dan program						
4	Pengujian dan penyempurnaan alat						
5	Penulisan naskah tugas akhir						

## DAFTAR RIWAYAT HIDUP



Nama : Tjipto Prakosa

Tempat / Tgl lahir : Kediri / 24 Maret 1972

Agama : Islam

Nama Ayah : dr. Rusman

Nama Ibu : Adiarti

Alamat : Jln. Kalimaya 10 Malang

Penulis adalah putra kelima dari lima bersaudara

### **Riwayat Pendidikan**

- TK Santa Maria, Kediri (1976-1978)
- SDK St. Yosef, Surabaya (1978-1983)
- SDK Cor Jesu, Malang (1983-1984)
- SMP Negeri 3, Malang (1984-1987)
- SMA Negeri 1, Malang (1987-1990)
- Fak. Kedokteran Univ. Brawijaya, Malang (1990-1991)
- Jur. Teknik Elektro FTI - ITS  
bidang studi Teknik Sistem Komputer (1991-Sekarang)

### **Pengalaman Kemahasiswaan**

- Asisten Praktikum Prokom dan R. Logika periode 1994 - 1996.
- Ketua Umum Panitia Study Excursie 1994.
- Bendahara I Himpunan Mahasiswa Teknik Elektro 1995/1996.