

Karya Ilmiah

**PENERAPAN METODE FONEM PADA PERANCANGAN
PERANGKAT LUNAK PENGUBAHAN DATA TEKS SMS
KE AUDIO**

Oleh :

Purwadi, S.Kom., M.Kom.



**SEKOLAH TINGGI MANAJEMEN ILMU KOMPUTER
TRIGUNADARMA
MEDAN
2011**

BAB I

PENDAHULUAN

1.1. Latar Belakang

Semakin majunya perkembangan teknologi khususnya dalam telekomunikasi yang telah diciptakan oleh manusia membuat informasi dapat disampaikan dengan cepat. Diantara perkembangan teknologi telekomunikasi adalah adanya ponsel (telepon seluler), dengan adanya ponsel manusia dapat melakukan komunikasi tanpa terbatas oleh ruang dan waktu. Ponsel merupakan salah satu teknologi telekomunikasi tanpa kabel (*wireless*) sehingga memungkinkan untuk dibawa kemana saja asalkan sinyalnya masih terjangkau oleh jaringan operator yang digunakan.

Saat ini ponsel telah memegang peranan yang penting bagi kehidupan manusia. Bahkan sekarang ini banyak orang baik tua maupun muda menggunakan sarana komunikasi ponsel. Ini karena praktisnya media tersebut, ponsel memiliki dimensi yang relatif kecil sehingga mudah dibawa kemana-mana dan tetap dapat dihubungi. Kehadiran ponsel dengan berbagai fasilitas yang mendukung aplikasi SMS, bahkan mendukung untuk aplikasi WAP (*Wireless Application Protocol*) dapat dibeli dengan harga yang terjangkau sehingga menarik minat berbagai kalangan masyarakat untuk menggunakannya.

Murahnya harga ponsel menjadikannya barang dagangan yang dapat dicari di mana saja termasuk di pedagang kaki lima. Ini membuktikan ponsel sekarang menjadi media komunikasi kelas menengah dan sebagian kelas bawah. Hal ini terlihat dari jumlah pertumbuhan pengguna ponsel yang terus meningkat dari waktu ke waktu. Maraknya penggunaan ponsel juga tak lepas dari dukungan oleh penyedia layanan komunikasi bergerak (operator seluler) yang menjual kartu perdana (*staterpack/simcard*) dengan harga yang relatif murah.

Bahkan sekarang ini untuk meningkatkan jumlah pelanggan, biaya yang dikenakan untuk pengiriman sebuah SMS dengan panjang karakter 160 dibuat murah oleh penyedia layanan komunikasi bergerak. Oleh karena itu, SMS sebagai media alternatif penyampaian pesan akhir-akhir ini cukup diminati oleh masyarakat, dan

karenanya membuat para penyedia layanan tersebut berlomba-lomba meningkatkan pelayanannya dengan cara meningkatkan kecepatan pengiriman pesan.

SMS sebagai salah satu fasilitas layanan dasar penyedia layanan komunikasi bergerak memungkinkan pengguna mengirim dan menerima informasi dalam bentuk pesan singkat. Layanan ini sangat efektif untuk memberitahukan informasi dalam waktu yang singkat, tetapi pesan SMS yang berupa teks membutuhkan waktu untuk membacanya. Untuk itu perlu dikembangkan pemikiran untuk mengubah teks menjadi ucapan, sehingga manusia tidak perlu membaca namun cukup mendengar. Pengubahan teks menjadi ucapan bermanfaat bagi para penyandang tuna netra dan tuna aksara agar lebih mudah mendapat informasi dari suatu teks SMS

Pengubahan teks menjadi ucapan dengan pembangkitan ucapan dilakukan menggunakan berbagai algoritma pengolahan sinyal digital yang dijalankan dengan perangkat lunak. Berkembangnya teknologi digital menyebabkan perkembangan sistem tulis-ucap atau TTS (*Text to Speech*) dan melahirkan beberapa alternatif baru untuk menjalankan bagian pembangkit ucapan untuk mencari pendekatan yang menghasilkan ucapan yang lebih alami.

TTS adalah suatu sistem perangkat keras maupun perangkat lunak yang mempunyai kemampuan untuk mengubah tulisan menjadi ucapan. Salah satu teknik yang digunakan pada TTS adalah teknik penyambungan diphone. Teknik ini dapat menghasilkan bunyi ucapan dengan tingkat kealamian yang tinggi. Penggunaan sistem TTS diharapkan dapat memberikan kemudahan dalam membaca SMS pada ponsel dengan menggunakan media komputer. Dengan uraian diatas maka penulis membuat judul karya ilmiah ini “penerapan metode fonem pada perancangan perangkat lunak pengubahan data teks SMS ke audio”.

1.2. Tujuan Penulisan

1. Untuk mengetahui bagaimana membuat aplikasi yang dapat mengubah teks SMS pada ponsel menjadi ucapan.
3. Untuk mengetahui bagaimana menerapkan metode fonem pada perancangan perangkat lunak pengubahan data teks SMS ke audio.

BAB II

LANDASAN TEORI

2.1 *Text To Speech (TTS)*

Text to Speech adalah suatu sistem yang dapat mengubah deretan kata-kata sebagai masukan menjadi ucapan sebagai keluaran. Sistem TTS juga biasa disebut dengan Sistem pensintesis ucapan. Sistem ini dapat digunakan untuk berbagai aplikasi, misalnya sistem informasi tagihan telepon atau sistem informasi lainnya yang diucapkan secara lisan. TTS juga dapat digunakan untuk mengubah teks pesan SMS ke ucapan sehingga pesan SMS dapat didengar. Metode sintesis ucapan memungkinkan mesin dapat melewati perintah atau informasi kepada pengguna lewat “ucapan”. Proses ini melibatkan pemecahan kata menjadi fonem, menganalisis untuk penanganan khusus dari teks seperti angka, jumlah mata uang, perubahan nada suara, dan juga pemberian tanda-tanda baca (ArmanA.A,*september 2004*).

Pesan atau informasi yang dikirimkan lewat ucapan memiliki kelebihan antara lain :

1. Pengguna dapat dengan mudah memahami pesan atau informasi tanpa perlu intensitas konsentrasi tinggi.
2. Pesan atau informasi dapat diterima saat pengguna sedang terlibat dengan aktivitas lain, misalnya saat berjalan, menangani atau sedang melihat objek lain.

2.2. *Pengubahan Teks Menjadi Ucapan*

Sistem *Text to Speech* pada prinsipnya terdiri dari dua subsistem dasar, yaitu:

1. Subsistem konverter teks ke fonem

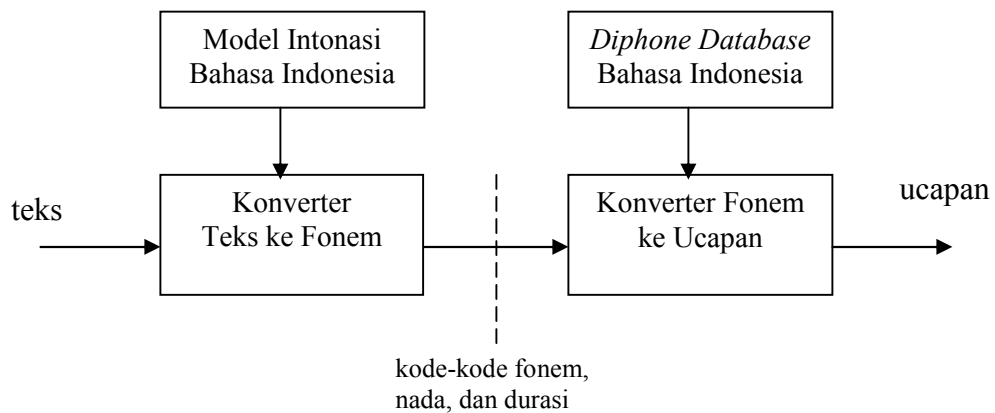
Subsistem konverter teks ke fonem yang memiliki dua fungsi utama. Pertama adalah mengambil kalimat masukan dalam suatu bahasa tertentu yang berbentuk barisan teks dan mengubah beberapa hal seperti nomor dan tanda ke dalam tulisan sesuai dengan bunyi yang seharusnya, sering disebut dengan normalisasi teks (*text normalization*). Kemudian menentukan kode fonetik (*phonetic transcriptions*)

untuk tiap kata beserta durasi dan nadanya. Kode fonem adalah kode yang merepresentasikan unit bunyi yang ingin diucapkan. Pengucapan kata atau kalimat pada prinsipnya adalah urutan bunyi atau secara simbolik adalah urutan kode fonem.

2. Subsistem konverter fonem ke ucapan

Subsistem konverter fonem ke ucapan yang akan menerima masukan kode-kode fonem serta nada dan durasi yang telah dihasilkan oleh bagian sebelumnya. Berdasarkan kode-kode tersebut bagian ini akan menghasilkan bunyi atau sinyal ucapan yang sesuai dengan kalimat yang ingin diucapkan. Ada beberapa alternatif teknik yang dapat digunakan untuk implementasi bagian ini. Salah satu teknik yang digunakan adalah penyambungan diphone.

Pada sistem yang menggunakan teknik penyambungan diphone, sistem harus didukung oleh suatu basis data diphone yang berisi rekaman segmen-segmen ucapan yang berupa diphone. Ucapan dari suatu bahasa dibentuk dari satu set bunyi mungkin berbeda untuk setiap bahasa. Gambar 2.1 menunjukkan diagram alir subsistem pensintesis ucapan.



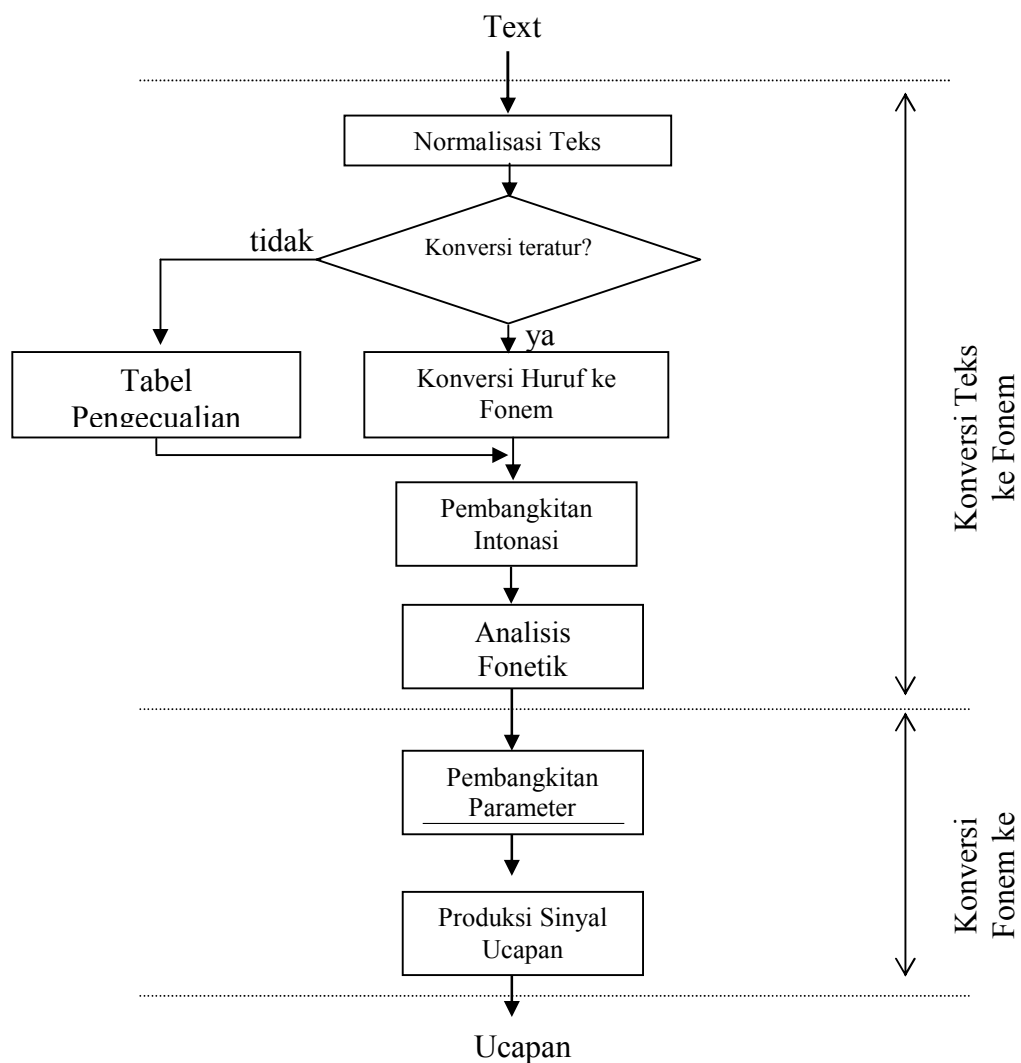
Gambar 2.1 Subsistem pensintesis ucapan.

Sumber : www.ppindo.com/download/tts.pdf

Setiap fonem dilengkapi dengan informasi durasi dan nada. Informasi durasi diperlukan untuk menentukan berapa lama suatu fonem diucapkan, sedangkan

informasi nada diperlukan untuk menentukan tinggi rendahnya nada pengucapan suatu fonem. Durasi dan nada bersama-sama akan membentuk intonasi suatu ucapan (Arman A.A., *september 2004*).

Setiap bahasa memiliki aturan cara pembacaan dan cara pengucapan teks yang sangat spesifik. Hal ini menyebabkan implementasi unit konverter teks ke fonem menjadi sangat spesifik terhadap suatu bahasa. Tahapan-tahapan utama konversi dari teks menjadi ucapan dapat dilihat pada Gambar 2.2.



Gambar 2.2 Urutan proses konversi dari teks ke ucapan.

Sumber : www.ppindo.com/download/tts.pdf

Tahapan-tahapan tersebut ialah normalisasi teks, konversi huruf ke fonem dan pengecualian, pembangkitan intonasi, analisis fonetik, dan generator sinyal ucapan MBROLA. Berikut ini adalah penjelasan dari masing-masing tahapan.

2.2.1. Normalisasi Teks

Tahap pertama dari sistem konversi teks ke ucapan adalah tahap normalisasi teks. Pada tahap ini semua teks kalimat diubah menjadi teks yang secara lengkap memperlihatkan cara pengucapannya. Normalisasi teks meliputi perubahan singkatan, akronim, angka, tanggal, waktu, karakter-karakter khusus, dan simbol-simbol dengan bentuk huruf alphabet lengkap sehingga tidak terjadi ambiguitas berkenaan dengan cara pengucapan (Dutoit T, *Dordrecht, 1997*).

Berkaitan dengan perubahan teks kalimat, tahapan normalisasi teks sendiri terdiri dari beberapa blok bagian meliputi :

1. *Konverter angka*, bagian konverter angka berfungsi mengubah angka menjadi deretan huruf alphabet yang menggambarkan cara pengucapannya, sebagai contoh angka 12550 akan diubah menjadi dua belas ribu lima ratus lima puluh.
2. Bagian *konverter akronim*, berfungsi mengubah akronim menjadi huruf tunggal atau deretan huruf alphabet yang menggambarkan cara pengucapannya, sebagai contoh singkatan “dlm” akan diubah menjadi “dalam”.
3. Bagian *konverter simbol dan karakter khusus* akan mengubah karakter dan simbol menjadi format teks sesuai dengan cara pengucapannya, sebagai contoh simbol “%” pada kalimat “5%” akan terbaca “lima persen”.

Subsistem ini harus memiliki pustaka setiap unit ucapan dari suatu bahasa. Perubahan angka, akronim, singkatan, simbol, dan karakter khusus sangat tergantung pada basisdata yang digunakan sistem. Ucapan dari suatu bahasa dibentuk dari satu set ucapan bunyi yang mungkin berbeda untuk setiap bahasa, oleh karena itu setiap bahasa harus dilengkapi dengan basisdata yang berbeda.

2.2.2. Konversi Huruf Ke Fonem Dan Pengecualian

Tahap selanjutnya setelah normalisasi teks adalah proses konversi huruf ke fonem. Tahapan ini bertujuan untuk mendapatkan ketentuan pengucapan dasar dari setiap kata teks yang telah dinormalisasi. Konversi huruf menjadi fonem biasanya dilakukan dengan dua pendekatan dasar yaitu :

1. Pendekatan berdasarkan kamus.

Pendekatan ini menggunakan suatu pendekatan yang menggunakan sebuah kamus besar dari sebuah bahasa yang terdiri dari semua kata, dan pengucapannya yang benar kemudian disimpan pada sebuah program. Pendekatan ini memiliki keuntungan lebih cepat dan teliti dalam proses perubahan huruf ke fonem tetapi akan berbeda jika kata yang diberikan tidak ada dalam kamus. Selain itu kamus memiliki perbendaharaan kata yang sangat banyak sehingga dibutuhkan ruang memori yang cukup besar untuk penggunaannya.

2. Pendekatan menurut aturan (*Rule-based approach*).

Pendekatan ini menggunakan aturan yang telah ditentukan untuk pengucapan kata berdasarkan ejaannya. Pendekatan ini bekerja pada masukan apa saja, tetapi kerumitan dari aturan berkembang sehingga membutuhkan ejaan dan pengucapan yang besarnya tidak tentu.

Pendekatan ini dapat diimplementasikan dengan tabel konversi yang berisi pasangan antara urutan huruf dan urutan fonem, bahkan mungkin hanya berisi satu huruf dan satu fonem. Aturan yang lebih sulit biasanya diimplementasikan dengan tabel konversi yang akan diterapkan jika kondisi rangkaian huruf tetangga kiri dan tetangga kanannya terpenuhi. Contoh bentuk aturan konversi huruf ke fonem yang memenuhi teknik tersebut adalah sebagai berikut:

teks-kiri[deretan huruf] teks-kanan = deretan fonem

Huruf tertentu yang ditunjuk dalam posisi [deretan huruf] akan diubah menjadi fonem dalam “deretan fonem” jika teks kiri dan teks kanan terpenuhi. Bahasa Inggris termasuk bahasa yang mempunyai keteraturan yang rendah untuk proses

konversi teks ke fonem. Suatu sistem TTS bahasa Inggris biasanya dilengkapi dengan suatu basis data yang berisi ribuan kata serta konversi padanan urutan fonemnya.

Bahasa Indonesia termasuk bahasa yang jelas aturan konversinya. Sebagian besar kata dalam bahasa Indonesia dapat diubah menjadi fonem dengan aturan yang jelas dan sederhana, walaupun tetap ada kondisi-kondisi yang tidak dapat ditemukan keteraturannya, sebagai contoh simbol huruf e dapat diucapkan sebagai e pepet atau e taling, artinya harus dikonversikan menjadi fonem yang berbeda untuk kondisi yang berbeda. Kondisi yang masih dapat ditangani oleh aturan diimplementasikan dengan blok konversi huruf ke fonem. Konversi yang tidak teratur ditangani oleh bagian tabel pengecualian.

2.2.3. Pembangkitan Intonasi

Gejala intonasi atau prosodi, mempunyai hubungan yang erat dengan struktur kalimat dan interelasi kalimat dalam suatu wacana. Karena sifatnya yang sangat subyektif, pembahasan intonasi lebih ditekankan pada aspek gramatikal dibandingkan aspek emosional.

Sebuah ucapan dilafalkan oleh penuturnya dalam pola melodi tertentu yang diterima pendengarnya sebagai sebuah deret nada dengan tinggi yang berbeda. Tingkat tinggi nada tidak dapat ditentukan secara pasti, sangat bervariasi tergantung dari jenis kelamin, emosi atau sikap. Prosodi adalah perubahan nilai-nilai nada atau frekuensi dasar selama pengucapan kalimat dilakukan atau nada sebagai satuan waktu. Pada prakteknya, informasi pembentuk prosodi berupa data nada serta durasi pengucapannya untuk setiap fonem yang dibangkitkan. Prosodi bersifat sangat spesifik untuk setiap bahasa, sehingga model yang diperlukan untuk membangkitkan data-data prosodi menjadi sangat spesifik untuk setiap bahasa. Beberapa metode pemodelan intonasi diantaranya adalah *Tone Sequence Model*, yang membangkitkan intonasi dengan cara merepresentasikan suatu kalimat dengan kuat lemahnya suku kata dan urutan tinggi rendahnya nada berkaitan dengan tekanan suku kata dan jeda prosodi.

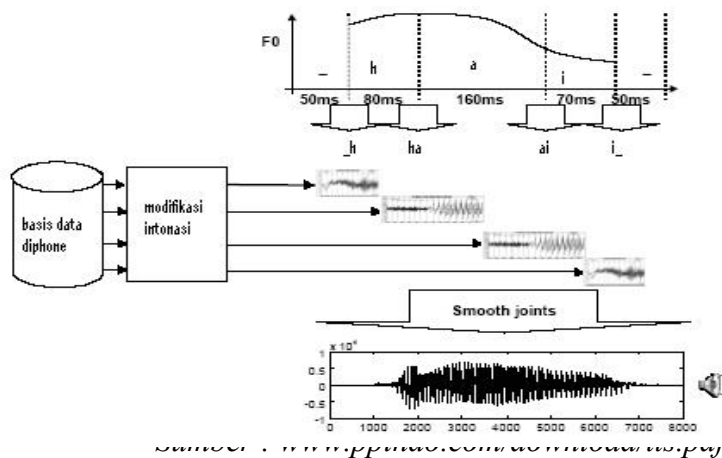
2.2.4. Analisis Fonetik

Satu tahap berikutnya yang masih sering dilakukan adalah analisis fonetik. Tahap ini dapat dikatakan sebagai tahap penyempurnaan, yaitu melakukan perbaikan di tingkat bunyi, sebagai contoh dalam bahasa Indonesia, fonem /k/ dalam kata “bapak” tidak pernah diucapkan secara tegas, atau adanya sisipan fonem /y/ dalam pengucapan kata “alamiah” antara fonem /i/ dan /a/.

2.2.5. Generator Sinyal Ucapan MBROLA

Salah satu contoh sistem pembangkit ucapan yang cukup populer adalah MBROLA yang dikembangkan oleh Faculte Polytechnique de Mons TCTS Lab, Belgia. MBROLA pada dasarnya merupakan sistem yang berbasis pada penyambungan *diphone*. MBROLA menerima masukan parameter berupa fonem dan informasi intonasi dengan format khusus.

Basisdata *diphone* merupakan komponen yang penting dalam MBROLA. Basisdata ini diproses dari segmen ucapan yang nantinya digunakan dalam proses pembangkitan ucapan. Basisdata *diphone* MBROLA telah tersedia untuk bahasa Portugis, Inggris, Belanda, Jerman, Prancis, dan juga Indonesia. Gambar 2.3 menunjukkan prinsip kerja generator sinyal ucapan MBROLA.



Gambar 2.3 Prinsip kerja generator sinyal ucapan

2.3. *Short Message Service (SMS)*

SMS merupakan aplikasi ponsel yang memberikan layanan untuk mengirim dan menerima pesan pendek berupa huruf dan angka. SMS adalah sebuah layanan pengiriman pesan singkat dari dan ke ponsel, mesin *fax*, dan atau sebuah alamat IP. SMS pertama dikirim dari komputer ke telepon bergerak pada jaringan Vodafone. SMS adalah ketentuan standar pesan dari ETSI (*European Telecommunication Standard Institute*).

Hal ini juga menentukan penerimaan standar GSM dan implementasinya melalui semua operator GSM . (*Febriyanto, Desember 1992*)

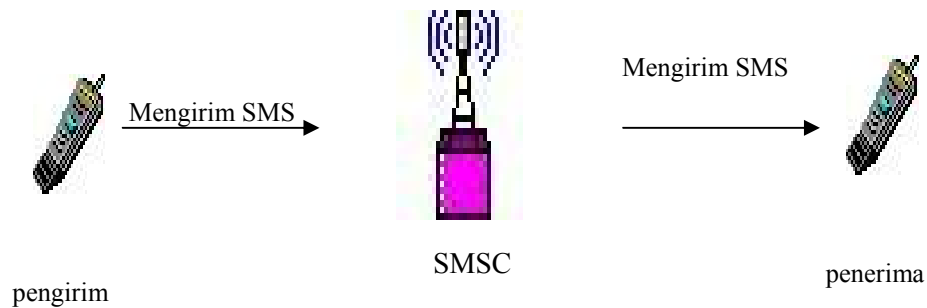
ETSI mendefinisikan 2 standar SMS yaitu :

1. Layanan yang diperuntukkan antara dua pihak dengan memerlukan penghubung perantara dari titik ke titik atau pengirim dan penerima antar ponsel dinamakan SMSPP (*SMS Point to Point*).
2. Layanan *broadcast* antar *network* melalui satu atau lebih *base station* dengan seluruh pengguna berada di dalam sel atau daerah layanan dinamakan *point to omni point* atau SMSCB (*SMS Cell Broadcast*).

SMS dikirim dari ponsel pengirim ke ponsel penerima melewati SMSC (*Short Message Service Centre*). SMSC adalah perangkat lunak yang berada di jaringan operator telepon seluler dan mengatur proses yang menyangkut pengiriman pesan SMS dengan prinsip *store* dan *forward*. Pesan SMS yang dikirim ke SMSC akan disimpan terlebih dahulu hingga masa validitas tertentu terpenuhi jika pada saat SMS dikirim ponsel yang dituju sedang tidak aktif ataupun di luar jangkauan operator.

Setelah nomor ponsel yang dituju sudah terdeteksi aktif atau berada dalam jangkauan operator maka SMS akan diteruskan oleh SMSC kepada penerima (jika *expired period* belum terlampaui). Apabila SMS yang tersimpan di SMSC sudah melewati masa validitas yang ditentukan, SMS tersebut akan dihapus dan tidak akan diteruskan ke nomor ponsel yang dituju.

Dengan adanya SMSC, pengirim juga dapat mengetahui status dari SMS yang dikirim, apakah SMS tersebut sudah diterima atau belum. Gambar 2.4 menunjukkan proses pengiriman SMS melewati SMSC.



Gambar 2.4 Pengiriman SMS melewati SMSC

Sumber : www.ppindo.com/download/smsgateway.pdf

Pada pengiriman dan penerimaan SMS ada 2 mode format SMS yang digunakan oleh operator. Mode pertama adalah mode PDU (Protocol Data Unit) yaitu format SMS dalam bentuk bilangan heksadesimal dengan panjang mencapai 160 (7 bit) atau 140 (8 bit) karakter. PDU berisi bilangan-bilangan heksadesimal yang mencerminkan bahasa I/O (kode). Mode yang kedua adalah mode teks yang menggunakan format SMS dalam bentuk teks asli. Tidak semua operator GSM di Indonesia mendukung format SMS mode teks dan kebanyakan menggunakan format SMS mode PDU.

Dalam proses pengiriman pesan, dikenal dua jenis *mobile* yaitu ponsel pengirim (*Mobile Originated*) dan ponsel penerima (*Mobile Terminated*). Pada ponsel pengirim, metode yang digunakan adalah *encodec*. Metode ini mengubah SMS dalam bentuk teks menjadi format PDU yang kemudian dikirimkan ke SMSC. Pada ponsel penerima, metode yang digunakan adalah *decodec* yang merupakan kebalikan dari proses *encodec*. Pada proses *decodec*, format PDU yang diterima dari SMSC akan diubah menjadi format teks.

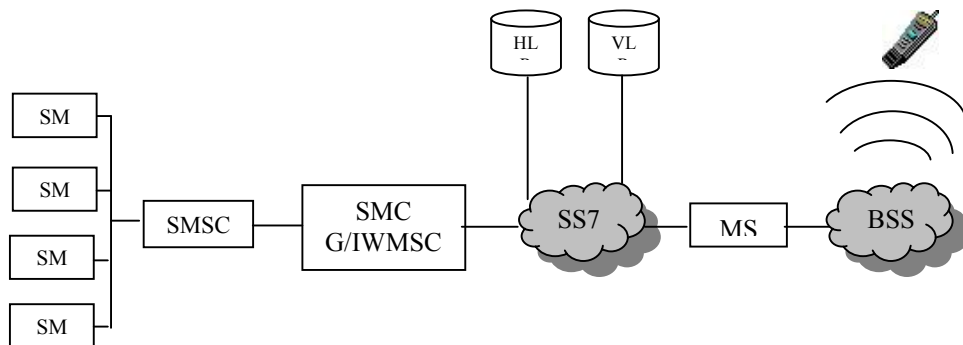
Pengiriman SMS dari dan ke PC perlu dilakukan terlebih dahulu koneksi ke SMSC. Koneksi PC ke SMSC dengan menggunakan terminal berupa GSM modem ataupun ponsel yang terhubung dengan PC. Dengan menggunakan ponsel GSM, SMS yang mengalir dari atau ke SMSC harus berbentuk PDU.

PDU sendiri terdiri atas beberapa bagian yang berbeda antara mengirim dan menerima SMS dari SMSC yang telah diatur dan distandarisasi oleh ETSI. Format

data PDU ini dikirimkan ke PC dalam bentuk teks (*string*) yang menunjukkan nilai heksadesimalnya. Jadi saat ponsel mengirim data heksadesimal F (0F h), maka yang diterima oleh PC adalah teks F.

2.3.1. Struktur Jaringan SMS

SMSC memiliki interkoneksi dengan SME (Short Messaging Entity) yang dapat berupa jaringan e-mail, web, dan voice e-mail. SMSC inilah yang akan mengatur manajemen pesan SMS, baik untuk pengiriman, pengaturan antrian SMS, dan penerimaan SMS. Gambar 2.5 menunjukkan elemen dan arsitektur jaringan SMS (Candera ,M.A.,*Jakarta,2004*).



Gambar 2.5 Elemen dan arsitektur jaringan SMS.

Sumber : www.ppindo.com/download/msggateway.pdf

Terdapat tujuh elemen pada jaringan SMS yaitu :

1. SME (*Short Messaging Entity*)

Adalah suatu piranti yang dapat menerima atau mengirim pesan singkat. SME berada dalam jaringan *fixed*, piranti bergerak, dan pusat layanan lainnya seperti VMS, *web*, dan *e-mail*.

2. SMSC (*Short Message Service Centre*)

SMSC merupakan perpaduan antara perangkat keras dan perangkat lunak yang bertugas untuk menerima, melakukan penyimpanan, dan meneruskan pesan pendek pada SME dan piranti bergerak.

3. SMS-GMSC (*SMS-Gateway Mobile Switching Centre*) dan SMS-IWMSC (*SMS-Interworking Mobile Switching Centre*)
SMS-GMSC merupakan suatu aplikasi yang dapat menerima pesan singkat dari SMSC, menginterogasi HLR untuk menginformasikan *routing*, dan mengirimkan SMS ke MSC dari ponsel yang akan dituju. Aplikasi MSC yang dapat menerima pesan pendek dari jaringan bergerak dan mengirimkannya ke SMSC yang tepat merupakan tugas dari SMS-IWMSC.
4. HLR (*Home Location Register*)
HLR pada elemen jaringan SMS merupakan suatu basisdata yang berfungsi sebagai penyimpanan tetap, pengelolaan data pelanggan, dan profil dari layanan. Setiap pelanggan bergantung pada sentral lokal pada jaringan tetap, tetapi pelanggan bergerak bergantung pada jaringan secara keseluruhan.
5. MSC (*Mobile Switching Centre*)
MSC melakukan fungsi pensaklaran sistem dan mengendalikan panggilan ke dan dari ponsel. MSC akan mengirimkan pesan pendek ke pelanggan tertentu melalui *base station* yang sesuai. MSC merupakan otak dari sistem radio selular. MSC juga berfungsi sebagai antarmuka antara jaringan GSM dengan *public voice* dan jaringan data.
6. VLR (*Visitor Location Register*)
VLR merupakan perangkat tempat penyimpanan data sementara (*database temporer*) dari pelanggan yang datang (*visitor*).
7. BSS (*Base Station System*)
Semua fungsi yang berkaitan dengan transmisi sinyal elektromagnetis radio antara MSC dan perangkat bergerak dilakukan di BSS. Perangkat bergerak merupakan terminal *wireless* yang mampu menerima dan mengirimkan pesan pendek. BSS terdiri dari BSC (*Base Station Controller*) dan BTS (*Base Transceiver Station*).
 - a. BSC merupakan penyaklaran dengan kapasitas tinggi yang digunakan untuk melakukan fungsi-fungsi yang ada kaitannya dengan *handover*, manajemen

jaringan radio, dan data konfigurasi dari sel. BSC mengontrol daya pancar radio baik pada *base station* ataupun ponsel.

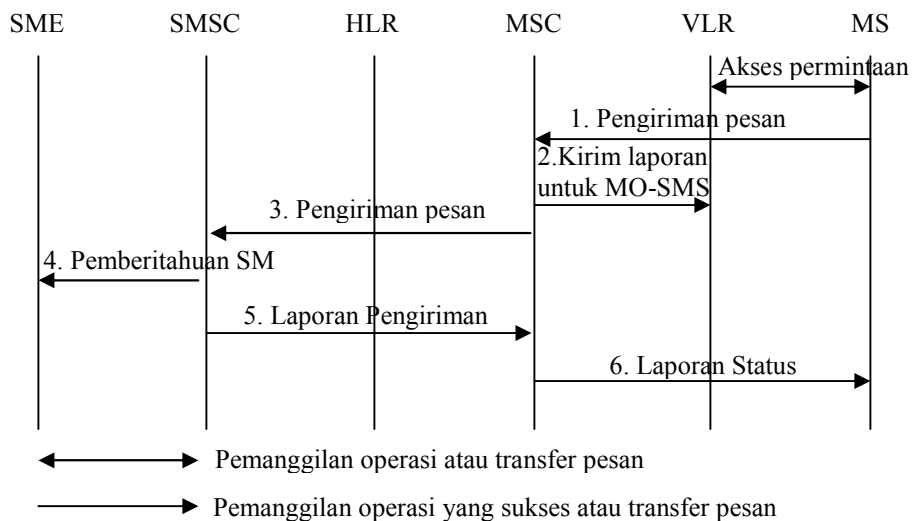
- b. BTS merupakan perangkat yang diperlukan oleh radio untuk mencakup satu atau beberapa sel serta menyediakan alur radio kirim dan terima.

2.3.2. Pengiriman dan Penerimaan SMS

Pada layanan SMSPP terhadap pelanggan, layanan SMS dibagi menjadi dua bagian yaitu:

1. MO-SM (*Mobile Originated Short Message*)

MO-SM dikirimkan dari *handset* yang MO-nya mampu menuju ke SMSC dan diterima ke pelanggan yang bergerak lainnya. Dalam layanan MO-SM selalu ada laporan yang dikirimkan ke *handset*, baik itu berupa konfirmasi pengiriman pesan pendek ke SMSC maupun konfirmasi kegagalan dalam pengiriman pesan dan pengidentifikasian penyebabnya.



Gambar 2.6 Skenario proses MO-SM.

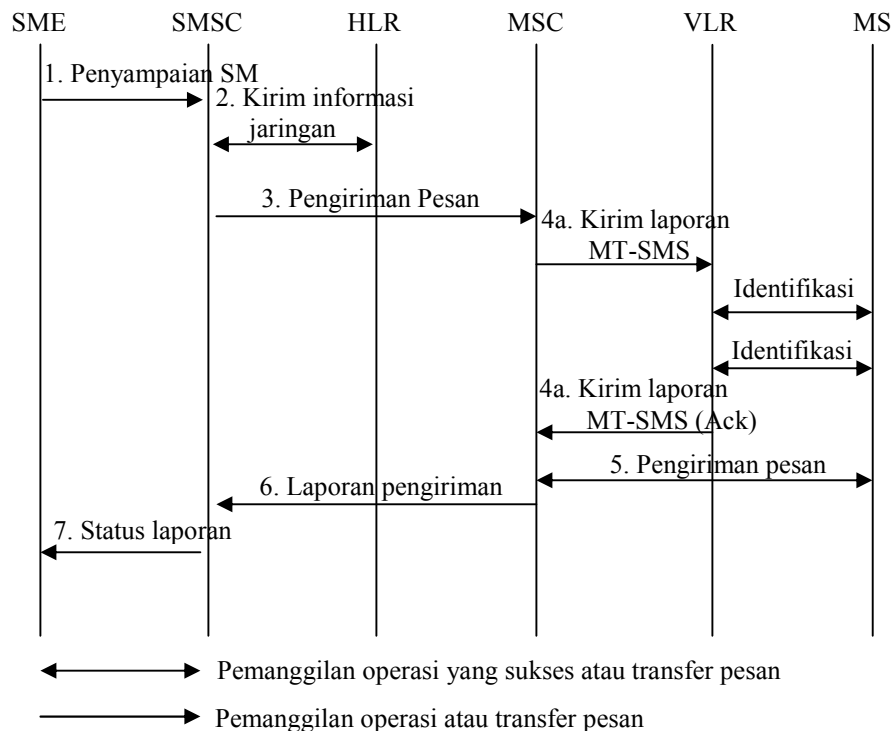
Sumber : www.ppindo.com/download/msgateway.pdf

Penjelasan dari proses MO-SM adalah sebagai berikut:

- a. MS akan mengirimkan pesan pendek ke MSC. Ketika proses ini terjadi, VLR akan melakukan akses permintaan informasi dari MS.
- b. MSC akan mengecek VLR bahwa pengiriman pesan tersebut tidak melanggar permintaan layanan tambahan atau layanan mengenai batasan yang telah ditentukan berupa laporan.
- c. MSC mengirim pesan pendek ke SMSC dengan menggunakan operasi kirim pesan pendek (*Forward Short Message*).
- d. SMSC akan menyampaikan pesan pendek tersebut ke SME.
- e. SMSC memberitahukan ke MSC dengan menggunakan laporan mengenai kesuksesan operasi kirim pesan pendek.
- f. MSC akan mengembalikan hasil dari operasi MO-SM ke MS.

2. MT- SM (*Mobile-Terminated Short Message*)

Kemampuan jaringan GSM mengirimkan SMS ke ponsel. Untuk laporan selalu dikirim kembali pada SMSC, salah satunya menegaskan konfirmasi penyampaian pesan pendek pada SMSC atau pemberitahuan pada SMSC apabila penyampaian pesan pendek mengalami kegagalan dan mengidentifikasi penyebab kegagalan tersebut. MT-SM dikirimkan dari SMSC ke *handset* dan dapat sampai ke pelanggan bergerak yang lain melalui MO-SM atau sumber lain seperti *voice-mail*. Gambar 2.7 menunjukkan proses MT-SM.



Gambar 2.7 Skenario proses MT-SM.

Sumber : www.ppindo.com/download/msggateway.pdf

Penjelasan proses MT-SM sebagai berikut:

- SME akan menyampaikan pesan singkat ke SMSC.
- SMSC akan memeriksa HLR dan menerima informasi tentang rute (jalur) bagi pelanggan yang bergerak setelah menyelesaikan pengolahan proses internal.

- c. SMSC mengirimkan pesan singkat ke MSC dengan menggunakan operasi pesan singkat.
- d. MSC mengambil kembali informasi pelanggan dari VLR. Proses ini dapat melibatkan prosedur *authentication* atau identifikasi. Bila telah teridentifikasi, VLR akan mengirimkan balik laporan *Acknowledgement*-nya ke MSC.
- e. MSC mengirimkan pesan singkat pada MS.
- f. MSC akan mengembalikan hasil dari operasi kirim pesan singkat ke SMSC.
- g. Jika SME meminta laporan status pengiriman, SMSC akan mengembalikan laporan status yang telah mengindikasikan pengiriman pesan singkat.

2.3.3. PDU untuk mengirim SMS ke SMSC

PDU untuk SMS yang dikirim ke SMSC terdiri atas delapan bagian. Gambar 2.8 menunjukkan skema PDU untuk mengirim SMS ke SMSC.

| | | | | | | | | |
|-----|----------|----|----|-----|-----|----|-----|----|
| SCA | Tipe SMS | MR | DA | PID | DCS | VP | UDL | UD |
|-----|----------|----|----|-----|-----|----|-----|----|

Gambar 2.8 Skema PDU untuk mengirim SMS ke SMSC.
 Sumber : www.ppindo.com/download/msgateway.pdf

Bagian-bagian tersebut adalah seperti berikut :

1. SCA (*Service Center Address*)

Bagian ini merupakan bagian yang pertama dari PDU yang merupakan informasi dari alamat (nomor) SMSC. SCA ini tersusun atas tiga bagian yaitu *len*, *type of number*, dan nomor SMSC (*Service Center Number*).

a. *Len*

Jumlah pasangan heksadesimal SMSC dalam bilangan heksadesimal.

b. Format nomor dari SMSC yang berisi kode nasional atau internasional SMSC (*type of number*)

- Kode nasional adalah 81

Kode ini bukan merupakan kode yang biasa dipakai di Indonesia. Ini digunakan jika pada ponsel pengirim dipasang menggunakan kode nasional.

- Kode internasional adalah 91

Kode internasional ini di Indonesia merupakan kode yang biasanya digunakan (*default setting*).

c. Nomor SMSC

Nomor SMSC dari operator pengirim pesan ini disusun sebagai pasangan heksadesimal dibolak-balik (heksadesimal disini sudah merupakan nomor SMSC itu sendiri yang sudah dalam bentuk *string*). Jika pada saat memasangkan tersisa satu angka heksadesimal yang tidak memiliki pasangan, angka tersebut akan dipasangkan dengan heksadesimal F didepannya.

Di Indonesia pada umumnya bagian SMSC menggunakan kode internasional, sebagai contoh untuk nomor SMSC dari Indosat-M3 akan dikodekan dengan kode internasional dengan langkah pengkodean adalah seperti berikut :

1. Nomor SMSC dari Indosat-M3 adalah +62855000000. Jika dipasangkan maka masih tersisa satu angka yaitu angka 0 yang tidak memiliki pasangan, angka 0 tersebut akan dipasangkan dengan heksadesimal didepannya, sehingga jika dibalik akan menjadi 26 58 05 00 00 F0.
2. Kode internasional adalah 91.
3. Penggabungan kode internasional dan nomor SMSC yang dibalik menghasilkan pasangan 91 26 58 05 00 00 F0 yang berjumlah 7 pasang (07 h).
4. Jadi bagian nomor SMSC berisi 07912658050000F0.

Beberapa nomor SMSC operator seluler di Indonesia ditunjukkan oleh Tabel 2.1.

Tabel 2.1 Daftar nomor SMSC dalam format PDU dengan kode internasional.

| No | Operator Seluler | SMSC | Kode PDU |
|----|-----------------------|-------------|------------------|
| 1 | Telkomsel (Simpati) | 6281100000 | 06912618010000 |
| 2 | Satelindo (Mentari) | 62816124 | 059126181624 |
| 3 | Excelcom (PRO-XL) | 62818445009 | 07912618485400F9 |
| 4 | Indosat-M3 | 62855000000 | 07912658050000F0 |

Sumber : www.ppindo.com/download/sistem_informasi.pdf

2. Tipe SMS

Pada saat mengirim SMS, tipe SMS bernilai 1. Jadi bilangan heksadesimalnya adalah 01 untuk tipe SMS kirim.

3. Nomor referensi SMS (*Message Reference*)

Nomor referensi SMS merupakan acuan dari pengaturan pesan SMS. Nomor referensi SMS dibiarkan kosong terlebih dahulu karena nanti akan diberikan sebuah nomor referensi otomatis oleh ponsel. Maka bilangan heksadesimalnya adalah 00.

4. Nomor ponsel penerima (*Destination Address*)

Bagian ini berisi alamat nomor tujuan. Bagian nomor ponsel penerima juga tersusun atas tiga bagian yaitu panjang nomor tujuan (*Len*), format nomor tujuan (*Type Number*), dan nomor tujuan (*Destination Number*).

5. PID (*Protocol Identifier*) atau bentuk SMS

Bagian ini berisi tipe dari cara pengiriman dimana biasanya diatur dari ponsel pengirim, misalnya tipe teks standar, *fax*, *telex*, dan lain-lain.

0 → 00 dikirim sebagai teks SMS

1 → 01 dikirim sebagai *telex*

2 → 02 dikirim sebagai *fax*

Untuk mengirim dalam bentuk teks akan digunakan kode heksadesimal 00.

6. DCS (*Data Coding Scheme*)

Bagian ini berisi skema *encoding* Data I/O.

Ada dua macam skema *encoding* yang digunakan yaitu :

- a. Skema 7 bit → ditandai dengan angka 00.
- b. Skema 8 bit → ditandai dengan angka yang lebih besar dari 0 lalu diubah ke heksadesimal.

Sebagian besar ponsel atau SMS *gateway* yang umum ada di pasaran menggunakan skema *encoding* 7 bit sehingga kode yang digunakan adalah 00.

7. VP (*Validity Period*)

Bagian ini berisi jangka waktu penyimpanan SMS di SMSC sebelum SMS *expired*. Jika bagian ini dikosongkan berarti tidak membatasi waktu berlakunya SMS sedangkan jika diisi dengan suatu bilangan integer yang kemudian diubah ke pasangan heksadesimal tertentu, bilangan tersebut akan mewakili jumlah waktu validitas SMS tersebut.

Tabel 2.2 Perhitungan waktu validitas SMS.

| Waktu VP | Nilai VP |
|------------------------------|--------------------------------------|
| 5 menit – 720 menit (12 jam) | $(\text{waktu VP}/5) - 1$ |
| 12,5 jam – 24 jam | $143 + ((\text{waktu VP} - 12) * 2)$ |
| 2 – 30 hari | $166 + \text{Waktu VP}$ |
| Lebih dari 4 minggu | $192 + \text{waktu VP}$ |

Sumber : www.ppindo.com/download/sistem_informasi.pdf

Sebaiknya waktu validitas dipasang maksimum, supaya SMS pasti terkirim ke ponsel penerima. Untuk jangka waktu validitas SMS adalah 5 hari maka nilai VP = $166 + 5 = 171$ d = AB h.

8. UDL (*User Data Length*)

Bagian ini berisi panjang SMS yang akan dikirim dalam bentuk teks standar. Misalkan contoh SMS yang dikirim adalah “Pesan singkat” yang mempunyai 13 karakter sehingga nilai heksadesimalnya adalah 0D h.

9. UD (*User Data*)

Bagian ini berisi isi SMS yang akan dikirimkan dan tentunya dalam format heksadesimal dengan skema pengkodean 7 bit. Skema 7 bit ini menunjukkan tabel konversi dari format bilangan 7 bit menjadi karakter yang diwakilinya dan sebaliknya. Format bilangan 7 bit ini kemudian diubah menjadi kode 8 bit untuk dikirimkan ke nomor tujuan. Skema 7 bit tersebut diperlihatkan pada Tabel 2.3.

Tabel 2.3 Skema 7 bit.

| | | | | | | | | | | | | |
|-------------|----|----|----|----|----|---|----|---|---|---|---|---|
| Skema 7 Bit | | | | B7 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| | | | | B6 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | | | | B5 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| B4 | b3 | B2 | b1 | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 | 0 | 0 | 0 | 0 | @ | Δ | SP | 0 | - | P | “ | P |
| 0 | 0 | 0 | 1 | 1 | £ | _ | ! | 1 | A | Q | a | Q |
| 0 | 0 | 1 | 0 | 2 | \$ | Φ | “ | 2 | B | R | b | r |
| 0 | 0 | 1 | 1 | 3 | ¥ | Γ | # | 3 | C | S | c | s |
| 0 | 1 | 0 | 0 | 4 | è | Λ | ↑ | 4 | D | T | d | t |

| | | | | | | | | | | | | |
|---|---|---|---|----|----|----|---|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 5 | é | Ω | % | 5 | E | U | e | u |
| 0 | 1 | 1 | 0 | 6 | ì | Π | & | 6 | F | V | f | v |
| 0 | 1 | 1 | 1 | 7 | ù | Ψ | ‘ | 7 | G | W | g | w |
| 1 | 0 | 0 | 0 | 8 | ò | Σ | (| 8 | H | X | h | x |
| 1 | 0 | 0 | 1 | 9 | Ç | Θ |) | 9 | I | Y | i | y |
| 1 | 0 | 1 | 0 | 10 | LF | Ξ | * | : | J | Z | j | z |
| 1 | 0 | 1 | 1 | 11 | Ø | 1) | + | ; | K | Ä | k | ä |
| 1 | 1 | 0 | 0 | 12 | ç | Æ | , | < | L | Ö | l | ö |
| 1 | 1 | 0 | 1 | 13 | CR | æ | - | = | M | Ñ | m | ñ |
| 1 | 1 | 1 | 0 | 14 | Å | B | . | > | N | Û | n | ü |
| 1 | 1 | 1 | 1 | 15 | å | É | \ | ? | O | § | o | à |

Sumber : www.ppindo.com/download/sistem_informasi.pdf

Berikut contoh pengkodean dengan skema pengkodean 7 bit untuk mengubah teks “Pesan singkat”, kemudian menjadi kode 8 bit untuk dikirimkan yang ditunjukkan Tabel 2.4.

Tabel 2.4 Skema pengkodean 7 bit untuk teks “Pesan singkat”.

| Nilai | Dec | Septet (7bit) | Oktet (8bit) | Hasil |
|-------|-----|---------------|--------------|-----------|
| P | 80 | 101 0000 | 1101 0000 | D0 |
| e | 101 | 110 0101 | 1111 0010 | F2 |
| s | 115 | 111 0011 | 0011 1100 | 3C |
| a | 97 | 110 0001 | 1110 1100 | EC |
| n | 110 | 110 1110 | 0000 0110 | 06 |
| spasi | 32 | 010 0000 | 1100 1101 | CD |
| s | 115 | 111 0011 | 1101 0011 | D3 |
| i | 105 | 110 1001 | | |
| n | 110 | 110 1110 | 1110 1110 | EE |
| g | 103 | 110 0111 | 1111 0011 | F3 |
| k | 107 | 110 1011 | 0011 1010 | 3A |
| a | 97 | 110 0001 | 0100 1100 | 4C |
| t | 116 | 111 0100 | 0000 0111 | 07 |

Sumber : www.ppindo.com/download/sistem_informasi.pdf

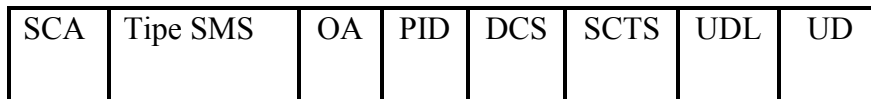
Data skema 7 bit dari teks SMS yang dikirim, dideret sedemikian rupa, kemudian mengambilnya tiap 8 bit. Jika pada akhir pengambilan bit yang tersisa kurang dari 8 bit, maka di bagian depan (MSB) ditambah dengan nol hingga

berjumlah 8 bit. Jadi PDU untuk teks “Pesan singkat” adalah D0F23CEC06CDD3EEF33A4C07 ditambah dengan panjang teks 13 (0D h). Jadi PDU isi SMS-nya adalah 0DD0F23CEC06CDD3EEF33A4C07.

Jadi pengiriman teks “Pesan singkat” ke nomor 628540787149 lewat SMSC Indosat-M3 tanpa ada jangka waktu *expired*, maka PDU lengkapnya adalah sebagai berikut
 07912658050000F001000C9126580487179400000DD0
 F23CEC06CDD3EEF33A4C07 (Candra,*Jakarta 2004*).

2.3.4. PDU SMS Yang Diterima Dari SMSC

PDU untuk SMS yang diterima dari SMSC terdiri atas delapan bagian yang sebagian besar hampir sama dengan PDU untuk SMS yang dikirim. Gambar 2.9 menunjukkan skema PDU untuk menerima SMS dari SMSC.



Gambar 2.9 Skema PDU untuk menerima SMS dari SMSC.
 Sumber : www.ppindo.com/download/smsgateway.pdf

Berikut adalah penjelasan dari kedelapan bagian tersebut:

1. SCA

Sama dengan untuk PDU SMS yang dikirim. SCA memiliki 3 bagian yaitu jumlah pasangan heksadesimal SMSC dalam bilangan heksadesimal, format nomor dari SMSC yang berisi kode nasional atau internasional SMSC, dan nomor SMSC dari operator pengirim untuk SMSC. Operator seluler Telkomsel dengan nomor SMSC 6281100000, maka PDU bagian SCA adalah 06912618010000.

2. Penanda bagian pesan

Tipe SMS untuk SMS terima adalah 04 h.

3. OA (*Originator Address*) atau nomor ponsel pengirim

OA berisi alamat nomor pengirim. Bagian nomor ponsel pengirim juga tersusun atas tiga bagian seperti pada PDU untuk nomor ponsel penerima yaitu panjang nomor pengirim, format nomor pengirim, dan nomor pengirim (*Originator Number*). Bentuk nomor ponsel pengirim sama dengan nomor ponsel penerima untuk PDU SMS yang dikirim.

4. PID atau bentuk SMS

Bagian ini berisi tipe dari cara pengiriman dimana biasanya diatur dari ponsel pengirim, misalnya tipe teks standar, *fax*, *telex*, dan lain-lain seperti halnya pada SMS PDU pengirim. Biasanya bagian ini memiliki nilai 00 yang menunjukkan bahwa bentuk SMS teks.

5. DCS atau skema *encoding*

Biasanya bagian ini memiliki nilai 00 yang menunjukkan bahwa skema pengkodean yang digunakan adalah skema 7 bit.

6. SCTS (*Service Centre Time Stamp*)

SCTS adalah tanggal dan waktu penerimaan SMS oleh SMSC penerima. Pada bagian ini memiliki 14 bilangan heksadesimal (7 pasangan), 6 pasangan yang berarti yy/mm/dd hh:mm:ss (tahun/ bulan/ tanggal jam: menit : detik) dan 1 pasangan yang menunjukkan kawasan waktu berdasarkan GMT, sebagai contoh data PDU SMS yang diterima pada bagian tanggal dan waktu SMS sampai SMSC 40807101010082 = 04/08/17 10:10:00 28 = 17 Agustus 2004, pukul 10:10:00 WIB.

7. Isi SMS

Bagian ini terdiri atas dua bagian yaitu :

a. UDL

UDL merupakan panjang isi pesan yang diterima dalam bentuk teks standar (jumlah huruf dari isi). Misalnya untuk kata “Pesan”, ada 5 huruf. Jadi pada bagian ini dituliskan 05.

b. UD

UD merupakan isi pesan yang diterima berupa pasangan bilangan heksa. Data ini menyerupai bilangan ASCII dari bilangan alphanumerik, namun beberapa karakter oleh forum SMS sedunia diadakan perubahan.

Cara mengubah data SMS ke dalam karakter yang diwakilinya adalah sebagai berikut: contoh data isi SMS adalah 05D0F23CEC06. Kode 05 menunjukkan jumlah karakter yang dikirim, data selanjutnya adalah kode dari karakter yang dikirimkan. Tabel 2.5 menunjukkan skema perubahan PDU ke teks untuk nilai D0F23CEC06.

Tabel 2.5 Skema perubahan PDU ke teks untuk nilai D0F23CEC06.

| Nilai | Oktet (8bit) | Septet (7 bit) | Hasil |
|-----------|------------------|------------------|----------|
| D0 | 1101 0000 | 101 0000 | P |
| F2 | 1111 0010 | 110 010 1 | e |
| 3C | 0011 1100 | 111 00 11 | s |
| EC | 1110 1100 | 110 00 01 | a |
| 06 | 0000 0110 | 110 1110 | n |

Sumber : www.ppindo.com/download/sistem_informasi.pdf

Dengan melihat data 7 bitnya dan melihat karakter alphanumeriknya pada Tabel 2.5 maka dapat diperoleh teks untuk PDU D0F23CEC06 adalah “Pesan”. Jika diterima PDU SMS seperti 06912618010000 04 0C91261822056896000040505101010 0820005D0F23CEC06 maka dapat dianalisis seperti berikut :

1. Kode 06912618010000 mempunyai arti bahwa pengirim menggunakan SMSC Telkomsel (Simpati).
2. Kode 04 menunjukkan bahwa PDU ini adalah PDU SMS terima.

3. Kode 0C91261822056896 menunjukkan bahwa nomor tujuan memiliki panjang 12 digit (0C) yaitu 628122508669, sedangkan 91 menunjukkan kode internasional sehingga perlu ditambah kode negara (Indonesia = 62).
4. Kode 0000 menunjukkan bentuk SMS adalah teks dan skema pengkodeannya adalah 7 bit.
5. Kode 40505101010082 menunjukkan SMS sampai di SMSC pada tanggal 15-05-04 pukul 10:10:00 dan 28 menunjukkan kawasan waktu Indonesia.
6. Kode 00 menunjukkan bahwa SMS tidak memiliki batas waktu validitas.
7. Kode 05D0F23CEC06 menunjukkan banyaknya karakter yang dikirim adalah 5 karakter yaitu kata “Pesan” yang dikodekan dengan D0F23CEC06.

2.4. Perintah AT (*AT Command*)

Perintah AT merupakan media komunikasi antara ponsel dengan PC. Perintah AT ini dapat digunakan untuk menulis, mengirim, membaca SMS, dan mengambil data yang ada pada ponsel maupun menjalankan aplikasi tertentu di ponsel. Antara ponsel dan komputer diperlukan kabel data untuk melakukan perintah AT. Perintah ini mengacu pada spesifikasi ETSI GSM 07.07 dan GSM 07.05 atau sesuai dengan spesifikasi yang diberikan oleh perusahaan pembuat ponsel (<http://www.etsi.org/juli2006>).

Perintah AT sebenarnya hampir sama dengan perintah *>* (*prompt*) pada DOS (*Disk Operating System*). Perintah-perintah yang dimasukkan ke *port* dimulai dengan kata AT, lalu diikuti oleh karakter lainnya yang mempunyai fungsi-fungsi unik. Beberapa contoh perintah AT yang penting untuk mengatur SMS dapat dilihat pada Tabel 2.6. Penjelasan secara lebih lengkap mengenai penjelasan fungsi-fungsi perintah AT ponsel Siemens M 35i dan cara penggunaannya dapat dilihat di lampiran laporan Tugas Akhir ini, sehingga dapat dicoba perintah AT yang diperlukan untuk aplikasi yang diinginkan.

Sebenarnya terdapat dua cara dalam pengiriman dan penerimaan pesan SMS yaitu mode teks dan mode PDU tergantung dari fasilitas yang ada pada ponsel yang

digunakan. Namun untuk ponsel Siemens seri M 35i, penerimaan dan pengiriman pesan SMS hanya bisa dilakukan dalam mode PDU. Pada mode PDU ini pesan dikirim dan diterima berisi PDU *string* yang tidak hanya berisi pesan saja tetapi juga beberapa bagian lain seperti SMSC. Keuntungan dari mode PDU yaitu dapat melakukan enkoding sendiri yang harus didukung pula oleh perangkat keras dan operator GSM, selain itu dengan mode PDU dapat melakukan kompresi data, menambahkan nada dering, dan gambar pada pesan yang akan dikirimkan. Tabel 2.6 menunjukkan beberapa perintah AT yang digunakan untuk mengatur SMS.

Tabel 2.6 Beberapa perintah AT untuk mengatur SMS.

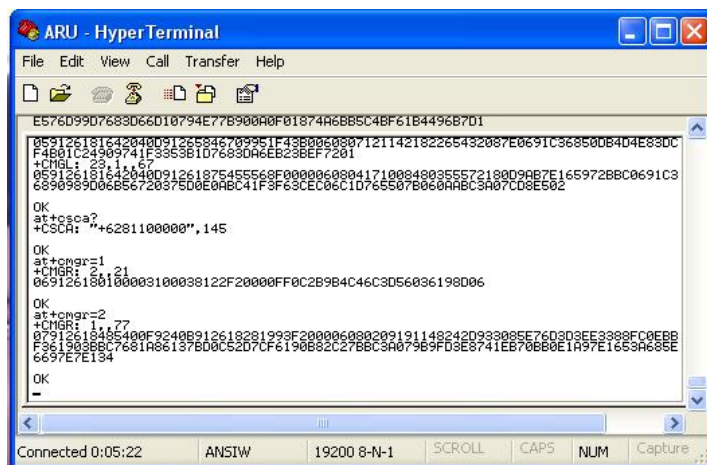
| Perintah AT | Keterangan |
|-----------------------|---|
| AT+CMGR=<mem>(Enter) | Digunakan untuk membaca SMS pada lokasi memori tertentu. Lokasi memori ini ditunjukkan oleh <mem>. Jadi AT+CMGR=1 menunjukkan sms yang dibaca adalah SMS pada lokasi memori 1 |
| AT+CMGL=<stat>(Enter) | Digunakan untuk membaca SMS dengan status tertentu yang ditunjukkan oleh <stat> dan ditampilkan dalam bentuk daftar. <stat> : 0 = SMS terima belum terbaca 1 = SMS terima sudah terbaca 2 = SMS tersimpan belum dikirim 3 = SMS tersimpan dan sudah terkirim 4 = semua SMS |
| AT+CMGD=<mem>(Enter) | Digunakan untuk menghapus SMS pada lokasi memori tertentu yang ditunjukkan oleh <mem> |

| | |
|----------------|---|
| AT+CMGF=<mode> | Digunakan untuk mengatur format SMS yang ditunjukkan oleh <mode>. <mode> : 0 = PDU mode 1 = Teks mode |
| AT+CSCA=? | Digunakan untuk mengetahui nomor SMSC |

Sumber : www.ppindo.com/download/sistem_informasi.pdf

Beberapa perintah AT yang terdapat pada Tabel 2.6 digunakan untuk menulis, mengirim, dan membaca SMS. Perintah AT+CMGR=1 digunakan untuk membaca SMS pada lokasi memori dengan indeks 1 pada *inbox* ponsel. Perintah AT+CMGL=0 digunakan untuk membaca SMS dengan status belum terbaca dan ditampilkan dalam bentuk daftar.

Untuk dapat mencoba penggunaan perintah AT dengan menggunakan perangkat lunak *Hyper Terminal* yang ada pada OS (*Operation System*) Windows. Gambar 2.10 menunjukkan penggunaan perintah AT dengan menggunakan perangkat lunak *Windows Hyper Terminal*.



Gambar 2.10 *Hyper Terminal*.

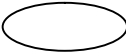
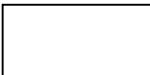
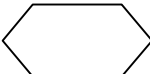
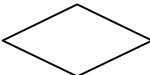
Sumber : www.ppindo.com/download/smsgateway.pdf


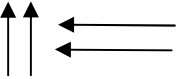


Hyper Terminal merupakan aplikasi yang dapat digunakan untuk mengirimkan perintah AT dari komputer ke ponsel dan menerima respon dari ponsel. Data pada ponsel dapat diketahui dengan mengirimkan perintah AT yang sesuai, sebagai contoh perintah AT+CMGR=1 digunakan melihat data PDU pada kotak masuk ponsel dengan indek 1. *Hyper Terminal* akan menerima respon “OK” dari ponsel apabila data pada ponsel dapat diambil.

2.5. Bagan Alir Program (Flowchart)

Suatu metode untuk menggambarkan tahap – tahap pemecahan masalah adalah dengan mempresentasikan simbol – simbol tertentu yang mudah di mengerti dan standart. Program flowchart adalah salah satu yang dapat didefinisikan sebagai bagan yang menjelaskan secara rinci langkah – langkah dari proses program (Jogiyanto, *HM : 2001 : 802*).

Tabel 2.7 Simbol – Simbol Flowchart Diagram

| Simbol | Nama simbol | Keterangan |
|---|-------------|--|
|  | Terminal | Untuk menggambarkan awal dan akhir proses aliran dokumen |
|  | Processing | Dipakai untuk pengolahan aritmatika dan pemindahan data |
|  | Preparation | Dipakai untuk memberikan nilai awal dari suatu variable atau counter |
|  | Decision | Dipakai untuk mewakili operasi perbandingan logika |

| | | |
|---|--------------------|---|
|  | Predefined process | Dipakai untuk proses yang detailnya dijelaskan secara terpisah , misalnya dalam bentuk subroutine |
|  | Garis alir | Dipakai untuk menunjukkan aliran dari program |
|  | Penghubung | Menunjukkan penghubung di halaman yang masih sama atau dihalaman lainnya |
|  | Input / Output | Mewakili data input atau output |

Sumber : www.ppindo.com/download/sistem_informasi.pdf

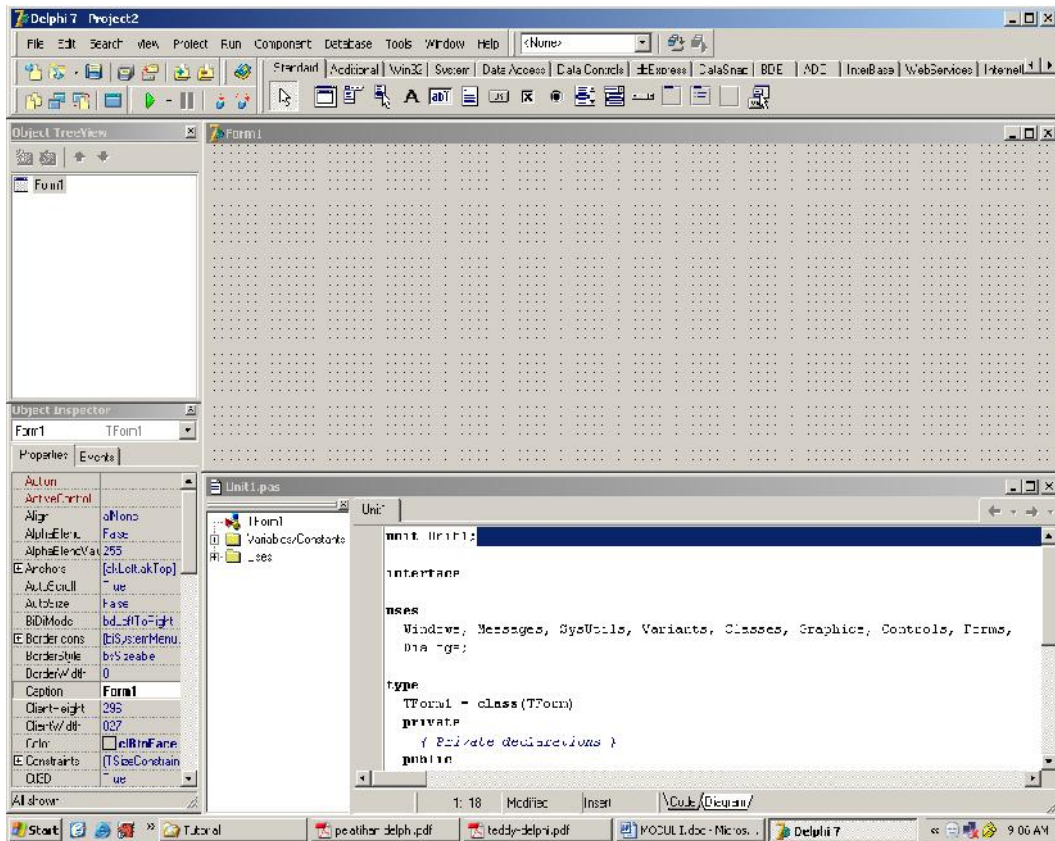
2.6 Borland Delphi

Delphi merupakan salah satu bahasa pemrograman yang bekerja di bawah lingkup sistem operasi windows, di mana Delphi memberikan fasilitas-fasilitas pembuatan aplikasi visual seperti Visual Basic. Dengan menggunakan Delphi dapat membuat aplikasi berbasis Windows dengan beberapa keunggulan, yaitu pada produktivitas, kualitas, pengembangan perangkat lunak, kecepatan kompiler, pola desain yang menarik serta diperkuat dengan bahasa pemrograman yang terstruktur dalam struktur bahasa pemrograman object Pascal.

Berbeda dengan pascal, pada Delphi kita akan mengenal OOP (Object oriented programming), jadi bila pada pascal kita akan melihat tampilan yang menjemukan, pada Delphi kita bisa mengatur tampilan kita semenarik mungkin pada form yang kita gunakan. Caranya cukup mudah, kita hanya menaruh komponen-komponen yang kita inginkan pada form tersebut, dan memfungsikan masing-masing komponen sesuai dengan yang kita inginkan.

2.6.1. IDE (Integrated Development Environment)

Merupakan sebuah lingkungan di mana semua tombol perintah yang diperlukan untuk mendesain, menjalankan dan menguji sebuah aplikasi disajikan dengan baik untuk memudahkan pengembangan program.



Gambar 2.11 : IDE Delphi

2.6.1.1 Main Window (Jendela Utama)

Jendela utama adalah bagian dari IDE yang mempunyai fungsi yang sama dengan semua fungsi utama dari program aplikasi windows lainnya. Jendela utama Delphi terbagi menjadi tiga bagian, yaitu Main Menu, Toolbar, dan Component Palette.



Gambar 2.12 : Main Window

2.6.1.2 Main Menu (Menu Utama)

Menu pada Delphi memiliki kegunaan yang sama seperti menu pada aplikasi windows. Dengan menggunakan fasilitas menu, Anda dapat memanggil atau menyimpan program, sehingga pada dasarnya semua perintah yang dikerjakan dapat ditemukan pada bagian menu utama. Nama-nama menu ditampilkan pada bagian menu bar yang terdapat pada bagian atas layar. Untuk bekerja dengan menggunakan menu, dapat menggunakan fasilitas mouse maupun keyboard.

2.6.1.3 Toolbar

Delphi memiliki beberapa toolbar yang masing-masing memiliki perbedaan fungsi dan setiap tombol pada bagian toolbar berfungsi untuk menggantikan suatu perintah menu yang sering digunakan. Toolbar terletak pada bagian bawah baris menu. Pada kondisi default, terdapat enam bagian toolbar, antara lain standard, view, debug, desktop, custom dan component palette.

2.6.2 Component palette

Berisi sekumpulan ikon yang melambangkan komponen-komponen yang terdapat pada VCL (Visual Component Library). Pada component palette terdapat beberapa page control seperti standard, additional, Win32, System, Data Access, dan lain-lain seperti gambar dibawah ini :



Gambar 2.13 : Componen Pallette

Komponen ini merupakan sebuah procedure/program yang sudah di compile dan langsung dapat digunakan, sesuai dengan fungsinya masing-masing. Untuk menggunakan komponen ini kita dapat meng-klik komponen yang diinginkan, kemudian kita klik di form, maka komponen tersebut akan muncul di form.

Kegunaan beberapa komponen :

a. Pointer

komponen khusus dan terdapat di setiap tab dalam component palette. Komponen pointer adalah komponen select yang digunakan untuk memilih komponen-komponen dalam form designer.

b. Frames

Sebuah kontainer yang digunakan untuk menampung komponen. Frame dapat diletakkan dalam form atau frame-frame yang lain.

c. Main menu

komponen yang digunakan untuk membuat menubar dan menu drop down, bersifat invisible.

d. PopupMenu

Popup Menu berfungsi sebagai perintah yang aktif bila kita meng-klik kanan mouse, Untuk mengaktifkannya kita harus mengaktifkan popup menu pada komponen yang diinginkan, caranya : ubah pada object inspector.

e. Label

Digunakan untuk menambah teks di dalam form.

f. Edit

Edit berfungsi sebagai masukan data (input) dalam bentuk string, dari bentuk string ini kita dapat mengolahnya menjadi bentuk integer atau bentuk lainnya. Yang kemudian dapat digunakan untuk operasi selanjutnya.

g. Memo

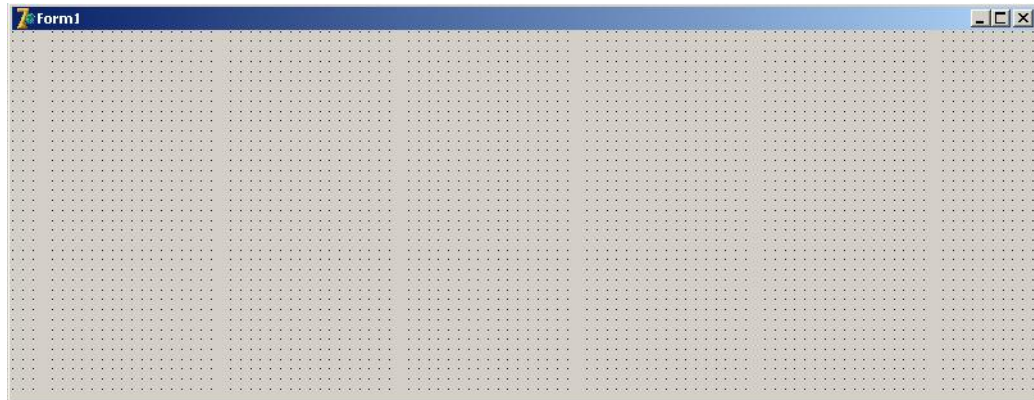
Dipakai untuk memasukkan atau menampilkan beberapa baris teks di dalam form.

- h. Button/ Bitbtn
Biasa digunakan sebagai tombol kendali. Perbedaan antara bitbtn dengan btn : pada bitbtn kita dapat menyisipkan warna pada tombol dan icon tertentu, lain halnya bila kita menggunakan btn.
- i. ComboBox
Combo Box berfungsi sebagai petunjuk untuk pemilihan berbagai masukan, di mana hanya ada satu pilihan yang dapat dipilih.
- j. CheckBox
Digunakan untuk memilih atau membatalkan suatu pilihan, yaitu dengan cara mengklik komponen.
- k. RadioButton
Prinsip kerjanya hampir sama dengan check box, cuma tampilannya saja yang berbeda.
- l. listbox
digunakan untuk membuat sebuah daftar option, di mana hanya ada satu option yang dapat dipilih.
- m. scrollbar
mempunyai fungsi sama seperti batang penggulung yang terdapat dalam program-program berbasis windows pada umumnya.
- n. groupbox
sebuah kontainer yang dapat digunakan untuk mengelompokkan komponen-komponen lain seperti radio button, checkbox, dan sebagainya.
- o. radiogroup
merupakan kombinasi dari groupbox yang didesain untuk membuat sekelompok radiobutton.
- p. panel
sebuah kontainer yang dapat digunakan untuk membuat status bar, toolbar, dan tool palette.

- q. **actionlist**
komponen yang berisi daftar action yang digunakan bersama-sama dengan komponen dan kontrol seperti item menu dan button.
- r. **Chart**
Data-data yang telah kita analisa, dapat kita tampilkan ke dalam grafik, sehingga memudahkan kita untuk menganalisanya.
- s. **Stringgrid**
Stringgrid berguna untuk menaruh data string kedalam bentuk kolom tabel, seperti pada Excel. Kita harus mengubah type data ke dalam bentuk string bila data yang ingin kita tampilkan data bukan string.
- t. **Media Player**
Biasa digunakan untuk menyalakan atau memainkan musik (format wav atau midi) dan menjalankan film (format avi).
- u. **Timer**
Timer berfungsi sebagai jam yang telah disediakan Delphi. Dengan timer kita juga dapat mendecode time, sehingga dapat terjadi akuisisi data.
- v. **Clientsocket**
Digunakan di client pada saat koneksi ke server.
- w. **Serversocket**
Digunakan di server, yang akan menyediakan layanan koneksi ke client.

2.6.3 Form Designer

Merupakan suatu objek yang dapat dipakai sebagai tempat untuk merancang program aplikasi. Form merupakan sebuah meja kerja yang dapat diisi dengan komponen-komponen yang diambil dari component palette. Pada saat memulai Delphi, Delphi akan memberikan sebuah form kosong disebut Form1.



Gambar 2.14 : Form Designer

Dalam sebuah form terdapat titik-titik yang disebut grid yang berguna untuk membantu pengaturan tata letak objek yang dimasukkan dalam form, dan gerakan pointer mouse akan disesuaikan dengan posisi titik-titik grid. Sebuah form mengandung unit yang berfungsi untuk mengendalikan form, komponen-komponen yang terletak dalam form dapat dikendalikan dengan menggunakan Object Inspector dan Code Editor.

Unit

Setiap perubahan pada form akan berakibat perubahan pada unit yang kita miliki, Untuk pindah dari form ke unit, anda bisa tekan F12. Berikut ini adalah bentuk unit yang diberikan Delphi saat pertama kita membuka sebuah form.

```
unit Unit1;  
interface  
uses  
Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs;  
type  
TForm1 = class(TForm)  
private  
{ Private declarations }
```

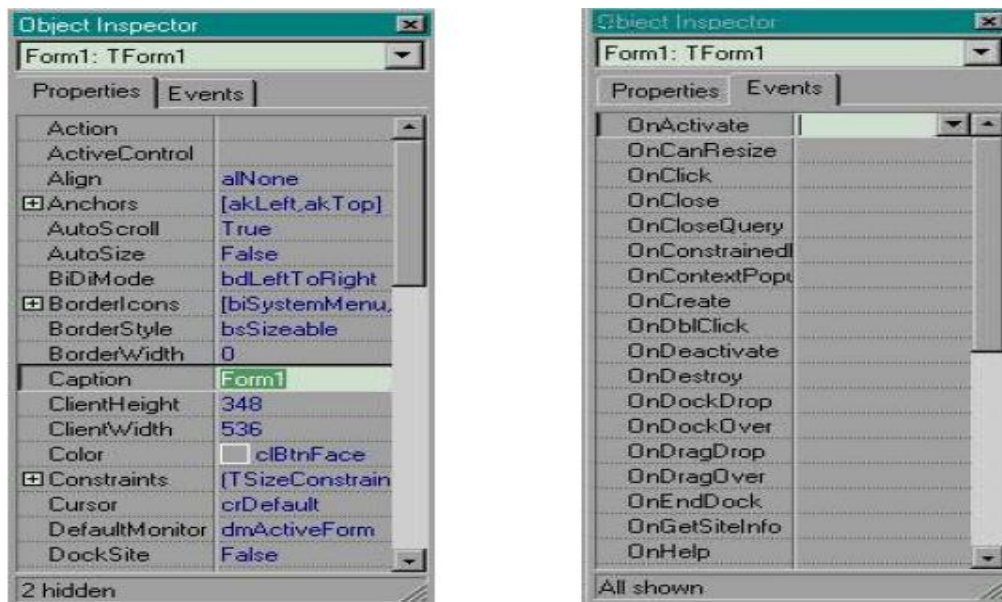
```

public
{ Public declarations }
end;
var
Form1: TForm1;
implementation
{$R *.DFM}
end

```

2.6.4 Object Inspector

Digunakan untuk mengubah properti atau karakteristik dari sebuah komponen, terdiri dari dua tab, yaitu properties dan events.



Gambar 2.15 : Object Inspector

1. Tab properties , digunakan untuk mengubah properti komponen. Properti dengan tanda + menunjukkan bahwa properti tersebut mempunyai subproperti. Klik pada posisi tanda + untuk membuka subproperti.

2. Tab events, bagian yang dapat diisi dengan kode program tertentu yang berfungsi untuk menangani kejadian-kejadian (berupa sebuah procedure) yang dapat direspon oleh sebuah komponen. Contoh, jika ingin suatu kejadian dikerjakan pada saat komponen tersebut diklik, maka dapat dituliskan kode program tersebut pada bagian OnClick.

BAB III

ANALISAN DAN PERANCANGAN

3.1. Analisa Teks Ke Suara

Konversi dari teks ke ucapan terdiri dari dua hal, yaitu :

1. mengubah dari teks ke fonem (text to fonem)
2. mengubah dari fonem ke ucapan (fonem to speech)

3.1.1. Teks Ke Fonem

Proses yang terjadi pada teks ke fonem adalah mengubah kalimat(teks) yang dimasukan dalam suatu bahasa tertentu yang berbentuk teks menjadi kode-kode bunyi yang biasanya diartikan menjadi kode fonem. Sedangkan kode fonem sendiri terdiri dari kode sampa, nilai duras dan nilai pitch (frekuensi dasar). Pada prinsipnya proses ini melakukan konversi dari symbol-simbol tekstual menjadi symbol-simbol fonetik yang mempresentasikan unit bunyi terkecil dalam bahasa, sedangkan setiap bahasa memiliki atuaran cara pembacaannya. Hal ini yang menyebabkan implementasi unit converter teks ke fonem menjadi sangat spesifik terhadap suatu bahasa. Karena setiap bahasa memiliki jumlah fonem yang berbeda sehingga mempunyai kode sampa yang berbeda pula. Maka, dibutuhkan diphone database yang berbeda untuk tiap bahasa. contoh : kata 'yo' dikodekan dengan kode fonem dengan kode sebagai berikut :

'y' dikodekan 'j 25 100 50 100'

'o' dikodekan 'Q 25 100 50 100'

Pada contoh diatas memiliki arti bahwa pada huruf 'y' dan 'o' pada durasi 25 ms, pitchnya senilai 100 hz. Jika kode tersebut kita masukan ke dalam engine MBROLA maka suara yang akan dihasilkan masih berupa suara yang datar tanpa intonasi karena memiliki durasi dan pitch yang sama.

3.1.2 Fonem Ke Ucapan

MBROLA adalah Speech syntheizer yang menggunakan teknik penggabungan segmen bunyi berdasarkan pangkalan diphone (diphone

concatenation), merupakan salah satu converter yang dibuat oleh TCTS Lab (TTS Research Team, Belgia), yang dapat beroperasi pada system operasi windows maupun yang lain, dan dapat menggunakan bahasa pemrograman delphi, java, visual basic dan bahasa pemrograman yang lain. Software ini digunakan untuk mengubah kode kode fonem menjadi suara/ ucapan.

Engine MBROLA hanya dapat membaca kode-kode fonem dalam file berextension '.pho'. Diphone database harus di masukan untuk mendefinisikan jenis suara berprosodi seperti apa yang akan dikeluarkan. Diphone adalah gabungan dari dua buah fonem, dan menggunakan teknik diphone concatenation yang bekerja dengan menggabungkan segmen-segmen bunyi yang telah direkam sebelumnya. dan setiap segmen merupakan gabungan dari dua buah fonem (diphone). Teknik ini digunakan agar dapat menghasilkan tingkat kenaturalan yang tinggi. Prosodi dataset, seperti yang kita ketahui bahwa prosodi adalah intonasi, dalam hal ini adalah intonasi suara yang dikeluarkan synthesizer MBROLA sesuai dengan pitch dan durasi yang tercantum pada kode fonemnya. Sedangkan model prosodi dataset adalah suatu model perbaikan ucapan yang dilakukan pada sistem text to speech dengan penambahan kosakata serta memasukkan parameter dari durasi dan pitch yang diubah-ubah berdasarkan langkah-langkah percobaan, sehingga menghasilkan sistem text to spech yang memiliki intonasi pada kata yang diucapkan.

Pada bagian teks ke fonem, teks dijabarkan dalam bentuk kode fonem yang kemudian kode-kode fonem itu akan dikonversikan menjadi kode sampa yang akan dimengerti oleh synthesizer MBROLA. Dalam synthesizer MBROLA yang akan terjadi adalah, kode-kode sampa yang telah berisikan pitch dan durasi akan disuarakan, pastinya sesuai dengan bahasa yang dimengerti synthesizer MBROLA. Oleh sebab itu dalam pembuatan sistem text to speech ini, sistem harus mengkonversi teks menjadi bahasa yang dikenali oleh MBROLA, sehingga dapat dibaca. lihat contoh berikut:

Pengucapan 'ratri sedang kerja'

3.2. Analisa SMS

Format SMS yang digunakan oleh ponsel/modem GSM adalah format PDU (Protocol Description Unit). Proses yang terjadi adalah format ASCII text yang kita ketikkan akan di-convert ke dalam format PDU, pada saat SMS diterima nomor tujuan, ponsel GSM akan mengubah kembali SMS format PDU tersebut menjadi format ASCII text kembali agar dapat terbaca dengan mudah oleh pengguna. Format PDU dituliskan dengan heksadesimal, terbagi atas 8 header, yaitu:

1. Nomor SMS Center.

Terdiri lagi dari 3 subheader yang memiliki aturan sendiri yaitu:

- a. Panjang atau jumlah pasangan digit dari nomor SMSC (service number) yang digunakan ditulis dalam bentuk heksa.
- b. Kode nasional/internasional. 81 untuk kode nasional, sedangkan 91 untuk kode internasional. Pada tulisan ini saya menggunakan kode internasional.
- c. Nomor SMS Center yang digunakan. Karena jumlah digit nomor SMS adalah ganjil, maka digit paling belakang dipasangkan dengan huruf F. Kalau diterjemahkan, nomor SMSC yang digunakan adalah +62855000000 (IM3). Jadi ditulis dalam bentuk PDU menjadi 2658050000F0.

Contoh: SMS Center dalam format PDU: 07912658050000F0 (IM-3),
07912618485400F9 (Excelcom), 059126181652 (Satelindo),
07912618010000F0 (Telkomsel).

2. Tipe SMS

Oktet pertama untuk PDU SMS untuk dikirim (SMS SUBMIT). Jadi bilangan heksanya adalah 01.

3. Nomor Referensi SMS

Nomor ini diisikan bilangan heksa 00. Yang artinya nanti akan diberikan sebuah nomor referensi otomatis oleh ponsel.

4. Nomor Ponsel Penerima

Header ini juga terdiri atas 3 bagian yang terdiri dari:

- a. Panjang digit nomor penerima yang dituliskan dalam bilangan heksa.
- b. Kode nasional/internasional. Sama dengan jenis nomor SMS Center.

- c. Nomor ponsel yang dituju. Dituliskan dalam pasangan heksa dibalik-balik. Apabila nomor tersebut ganjil, maka terlebih dahulu harus ditambahkan dengan F dibelakang nomor tersebut, baru kemudian dibalik-balik. Contoh: misal nomor yang dituju adalah 6285710155788. Karena ganjil maka ditambahkan F dibelakang nomor menjadi: 6285710155788F dan diubah menjadi:
 - a. 0D karena ada 13 angka.
 - b. 91 karena digunakan kode internasional.
 - c. 265817105587F8
 - d. Digabung menjadi: 0D91265817105587F8.
5. Pengenal Protokol
 - 0 → 00 → dikirim sebagai SMS.
 - 1 → 01 → dikirim sebagai telex.
 - 2 → 02 → dikirim sebagai telefax.
 6. Skema pengkodean SMS

Ada dua skema yaitu:

 - a. Skema 7 bit ditandai dengan angka 0, dan dituliskan sebagai 00.
 - b. Skema 8 bit ditandai dengan angka lebih besar dari 0 diubah ke heksa. Saat ini semua ponsel menggunakan skema 7 bit, sehingga dituliskan kode 00.
 7. Validitas Waktu

Menggunakan heksa FF yang berarti tidak membatasi waktu berlakunya SMS.
 8. Isi SMS

Dibagi menjadi 2 header yaitu:

 - a. Panjang dari pesan SMS. Misalkan kata “Halo” yang berarti ada 4 huruf, sehingga ditulis dalam bentuk heksa 04.
 - b. Isi SMS berupa pasangan heksa.

Langkah pertama: mengubah huruf menjadi kode 7 bit.

Langkah kedua: mengubah kode 7 bit menjadi 8 bit dengan cara menambahkan bit dari septet kedua sambil mengisi kekosongan bit dari septet kedua dengan bit dari septet ketiga dan seterusnya.

Contoh: untuk kata “Halo”

Langkah pertama:

H = 1001000

a = 1100001

l = 1101100

o = 1101111

Langkah kedua:

H = 11001000 : C8

a = 00110000 : 30

l = 11111011 : FB

o = 00001101 : 0D

Jadi kata “Halo” bila dikonversikan ke PDU akan menjadi C830FB0D. Sehingga apabila kata “Halo” dikirimkan ke ponsel dengan nomor 6285710155788 lewat SMS Center IM-3, tanpa batas waktu expired, maka PDU-nya adalah:

C830FB0D.

- c. Isi PDU dikonversi ke dalam karakter.

Langkah pertama: Pisahkan format PDU menjadi 2 karakter setiap kata

Langkah kedua :Ubah setiap 2 karakter tersebut ke bentuk heksa decimal

Langkah ketiga :Ubah Heksa decimal tersebut kedalam bentuk karakter

Contoh: untuk PDU “C830FB0D”

Langkah pertama:

C8 30 FB 0D

Langkah ke dua :

C8 = 1001000

30 = 1100001

FB= 1101100

0D = 1101111

Langkah ketiga:

11001000 =H

00110000 = a

11111011 = l

00001101 = o

Jadi kata “C830FB0D” bila dikonversikan ke Karakter akan menjadi “Helo”.

Sehingga apabila ingin mengirim kata “Halo” dengan Hyper Terminal melalui AT command adalah sebagai berikut:

1. Ketik “AT+CMGS=19”

Perintah AT+CMGS adalah AT command untuk mengirim pesan pada ponsel. Sedangkan 19 adalah jumlah pasangan heksa setelah nomor SMS Center.

2. Setelah itu tunggu respon dari ponsel. Apabila tidak terjadi kesalahan maka karakter yang dikembalikan adalah “>”.

Setelah diterima karakter “>” maka kita bisa menuliskan semua PDU yang kita dapat ke ponsel melalui Hyper Terminal dan diakhiri dengan penulisan kode ASCII 26 (CTRL+Z).

Jika sukses, maka yang dikembalikan adalah OK. Dan apabila terdapat error harus diperiksa kembali PDU beserta perintah AT command yang dimasukkan.

3.3. Arsitektur Sistem

Merancang arsitektur sebuah sistem merupakan langkah awal yang kritis. Arsitektur sistem menjadi landasan utama bagaimana nantinya sistem tersebut bekerja. Perancangan arsitektur sistem yang baik akan menghasilkan suatu sistem yang baik pula. Arsitektur sistem memberikan dasar-dasar untuk pengembangan sistem selanjutnya. Arsitektur sistem informasi disini merupakan gambaran skema arsitektur sistem secara keseluruhan dan bukan hanya skema arsitektur aplikasi saja. Skema ini menggambarkan bagaimana proses interaksi (aliran informasi) antara user dengan aplikasi sistem informasi dapat terjadi.

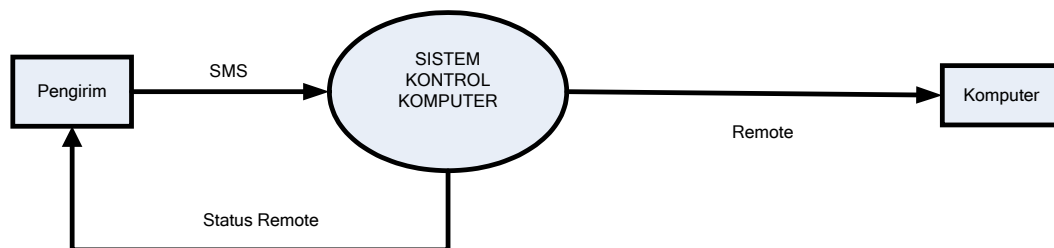
3.4. Alur Operasi Mengambil dan Mengirim Pesan SMS

Alur operasi sistem memberikan gambaran tentang keadaan-keadaan yang terdapat pada sistem serta kejadian-kejadian (events) yang menyebabkan terjadinya perubahan dari satu bagian ke lainnya. Alur operasi ini merupakan alat bantu yang sangat berguna dalam perancangan perangkat lunak. Dan merupakan representasi grafis yang menggambarkan tahapan-tahapan prosedural secara mendetail. Alur operasi untuk kasus sistem informasi ini yang melibatkan pengambilan dan pengiriman pesan SMS.

3.5. Data Flow Diagram

Data Flow Diagram adalah suatu diagram menggambarkan aliran informasi dan perubahan yang terjadi ketika data melalui proses input sampai menghasilkan output. Data Flow Diagram dapat digunakan untuk merepresentasikan sebuah sistem atau software pada setiap level abastaksinya. Data Flow Diagram dapat dipisahkan kedalam level yang merepresentasikan aliran data yang lebih mendetail

Untuk sistem ini penulis merancang Data Flow Diagram yang dapat dilihat dari gambar 3.1.



Gambar 3.1 Data Flow Diagram Level 0

3.6. Pembacaan SMS dan Perancangan

3.6.1. Metode Perancangan

Perancangan program aplikasi menggunakan pendekatan terstruktur dengan diagram alir yang menjelaskan urutan proses yang terjadi pada aplikasi,

disamping itu juga terdapat penjelasan mengenai algoritma program. Hasil yang diharapkan adalah berupa sebuah aplikasi yang dapat memberitahu pengguna jika ada pesan SMS yang masuk pada ponsel pengguna dan mengubah isi pesan SMS menjadi ucapan berbahasa Indonesia.

Perancangan sistem merupakan tahapan yang penting dalam pembuatan program. Perancangan bertujuan agar dalam pembuatannya dapat berjalan secara sistematis, terstruktur, dan rapi sehingga hasil program dapat optimal dan berjalan sesuai dengan apa yang dikehendaki. Aspek-aspek dalam perancangan yang diperhatikan meliputi kemungkinan pengembangan di masa depan, efektif dan efisien, kemampuan program, dan kemudahan untuk dipahami pengguna (*user friendly*) yang diwujudkan dalam tampilan grafis (*Graphical User Interface*).

Algoritma aplikasi *Text To Speech* berbahasa Indonesia sebagai pembaca pesan SMS adalah sebagai berikut:

1. Proses inialisasi basisdata yang digunakan untuk menyimpan pesan SMS dan *phonebook*, konfigurasi pengaturan ponsel yang ada pada *registry* dan koneksi ke ponsel.
2. Menghubungkan aplikasi dengan ponsel.
3. Aplikasi memeriksa isi dari kotak masuk ponsel, setelah terhubung.
4. Jika pada kotak masuk pada ponsel ada pesan SMS baru yang masuk atau SMS baru yang belum terbaca dan proses pemeriksaan aplikasi berjalan, pesan SMS baru tersebut akan diambil dan diproses.
5. Proses SMS baru yang masuk dilakukan dengan melakukan pemisahan bagian-bagian SMS seperti nama atau nomor ponsel pengirim, tanggal penerimaan SMS, isi SMS, lokasi SMS pada ponsel, dan jenis status SMS.
6. Aplikasi akan membacakan pesan SMS tersebut apabila program diatur secara otomatis membacakan pesan SMS dan menampilkan pesan dalam bentuk teks pada tabel pesan baru. Jika tidak diatur otomatis maka aplikasi hanya akan menampilkan pesan dalam bentuk teks pada tabel pesan baru.

7. Setelah pesan SMS dibacakan, aplikasi akan menyimpan pesan SMS tersebut dalam tabel basisdata dan menghapus pesan pada ponsel.

3.6.2. Struktur Pembacaan SMS

SMS Gateway akan menampung semua SMS yang masuk dalam suatu antrian. Bila antrian SMS Gateway belum ada SMS yang masuk maka string penampung SMS akan mengembalikan string kosong (empty string). Perintah SMS Gateway untuk mengembalikan string ini adalah *Retrieve_Message*. Jika SMS telah ada dalam antrian SMS Gateway dan program aplikasi telah terdaftar maka isi SMS akan dikembalikan ke string yang berisi field-field berikut:

Tabel 3.1 Struktur Field Pembacaan SMS

| Posisi Field | Panjang Field | Nama Field |
|--------------|---------------|------------|
| 1 | 4 | Status |
| 6 | 15 | Asal |
| 22 | 8 | Tanggal |
| 31 | 8 | Waktu |
| 40 | 0 sampai 160 | Teks pesan |

Perintah *Retrieve_Message* hanya mengambil satu buah SMS pada satu saat, sehingga untuk dapat mengambil semua SMS yang ada dalam antrian SMS Gateway diperlukan mekanisme perulangan dimana dalam hal ini digunakan sebuah *timer*. Timer di-set untuk melakukan perulangan setiap waktu (misalnya dalam interval 2 detik) yang ditentukan.

3.6.3. Pembacaan dan Penulisan Keyword

Keyword diperoleh dari SMS yang diterima oleh HP terminal. Pesan yang diperoleh diubah menjadi bertipe string. String SMS digunakan menjadi keyword pada pencarian data dan sekaligus keyword untuk memperoleh jawaban informasi dari sistem. Karakter Pesan untuk keyword dimulai dari posisi field 40. Setelah pembacaan dari data, pesan berubah menjadi string dan digunakan sebagai penulisan

keyword. Khusus untuk system remote komputer ini penulis merancang keyword yang dapat diakses oleh user dibatasi hanya pada keyword yang terekam di tabel keyword.

3.7. Perancangan Dialog (*User Interface*)

Rancangan Dialog (*user interface*) meliputi rancangan input dan output. Rancangan input meliputi rancangan dialog login sistem, menu utama, rancangan memperbaharui data dan rancangan output.

3.7.1. Desain Dialog Menu Utama

Desain dialog menu utama ini akan muncul setelah Program di jalankan . Menampilkan dan mengaktifkan seluruh menu dan toolbar.

| |
|--|
| PEMBACA SMS |
| [Aktifkan Modem] [SMS Masuk] [Kirim SMS] |
| |
| Design By : xxxx |

Gambar : 3.2 : Design Menu Utama dari Sistem

3.7.2. Form Aktifkan Handphone

Form ini dibuat untuk mengaktifkan handphone/modem yang digunakan untuk menerima SMS dari handphone lain.

| | | | |
|---|--|------------------------------|-------------------|
| PEMBACA SMS | | | |
| [Aktifkan Modem] [SMS Masuk] [Kirim SMS] | | | |
| <table border="1"><tr><td style="text-align: center;">Pilih Port : [= Pilihan =]</td></tr><tr><td style="text-align: center;">[Keluar] [OK]</td></tr></table> | | Pilih Port : [= Pilihan =] | [Keluar] [OK] |
| Pilih Port : [= Pilihan =] | | | |
| [Keluar] [OK] | | | |
| Design By : xxxx | | | |

Gambar : 3.3 : Design Form Aktifkan Handphone

3.7.3. Form SMS Masuk

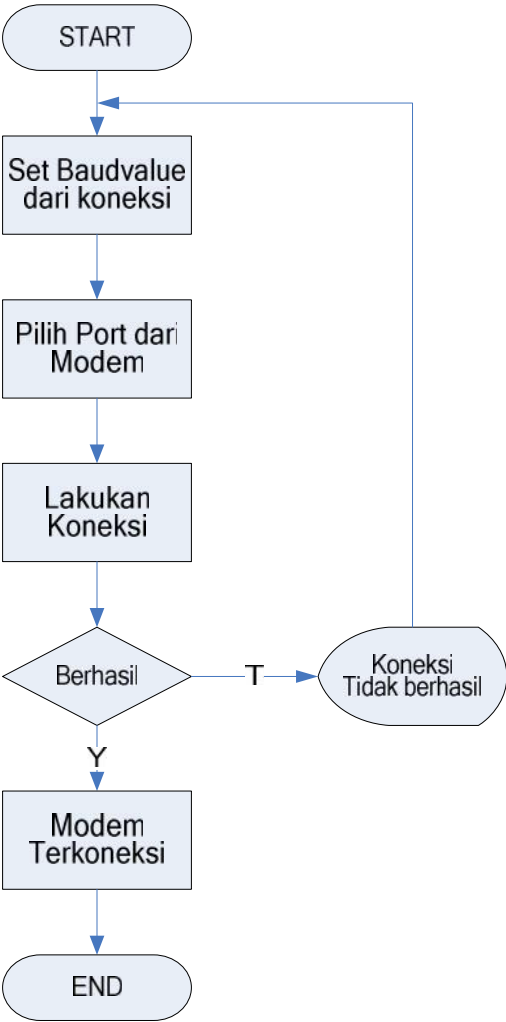
Form ini dibuat untuk menampilkan SMS yang masuk dari handphone lain. Yang dapat dilihat seperti rancangan dibawah ini :

| PEMBACA SMS | | | | |
|--|---------|-----|--------------|-------|
| [Aktifkan Modem] [SMS Masuk] [Kirim SMS] | | | | |
| DAFTAR SMS MASUK | | | | |
| No. | Tanggal | Jam | No. Pengirim | Pesan |
| | | | | |
| | | | | |
| | | | | |
| | | | | |
| | | | | |

Design By : xxxx

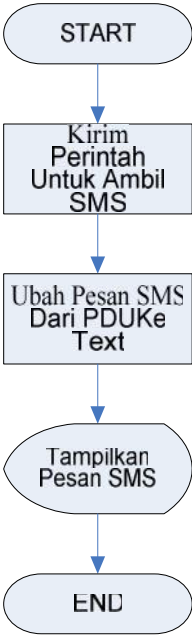
Gambar : 3.4 : Design Form Daftar SMS Masuk

3.8. Flowchart Koneksi Modem ke Komputer



Gambar 3.5 :Flowchar Koneksi Modem ke Komputer

3.8.1. Flowchar Terima SMS dari Modem



Gambar 3.6 : Flowchar Terima SMS dari Modem

BAB IV

PENUTUP

4.1. Kesimpulan

1. Program Rancangan dan Implementasi ini bertujuan untuk bisa membantu penderita tuna netra untuk mendengarkan SMS yang masuk.
2. Perangkat Lunak ini dapat diaplikasikan untuk penderita tuna netra
3. Dengan perangkat lunak ini diharapkan dapat membantu menyelesaikan kesulitan dalam pembacaan SMS yang masuk kedalam komputer.
4. Hasil Implementasi menunjukkan bahwa sistem ini dapat dijalankan dengan menerima input SMS langsung dari Handphone pengirim

4.2. Saran

Untuk kesempurnaan perangkat lunak ini dimasa yang akan datang penulis menyarankan :

1. Pembacaan SMS masih kurang baik, sehingga diharapkan perbaikan untuk mengatasi permasalahan tersebut
2. Ada Kosa Kata yang tidak bisa dibaca dengan baik, maka diharapkan untuk perbaikan selanjutnya.
3. Ponsel yang dapat digunakan hanya ponsel GSM, sehingga diharapkan perbaikan untuk mengatasi permasalahan tersebut.

DAFTAR PUSTAKA

Erma Suryani, 2006, “**Pemodelan dan Simulasi**”, PT Graha Ilmu.

Hartono Jogiyanto, 2000, “**Pengenalan Komputer**”, Andi Yogyakarta.

Kusumadewi.S, 2000, “**Sistem Operasi**”, Edisi Kedua, Graha Ilmu, Jakarta.

Raymond McLeod, Jr, 2007, “**Sistem Operasi Manajemen**”, Edisi Kesembilan, PT Indeks.

www.ppindo.com/download/tts.pdf

iww@stttelkom.ac.id

Tim Divisi Penelitian dan Pengembangan, Mahir dalam 7 hari pemograman Visual Basic, Madiun : Andi, 2005