

## OPTIMALISASI TEKNOLOGI INFORMASI BERBASIS SMS GATEWAY MEMANFAATKAN PERINTAH DASAR AT COMMAND

**Mohammad Ridwan, S.Kom., M.Kom**

*Dosen Jurusan Teknik Informatika, Universitas Islam Syekh Yusuf Tangerang  
Jl. Maulana Yusuf Kota Tangerang, Banten, Telp 021-5527061  
Email : [mridwan@unis.ac.id](mailto:mridwan@unis.ac.id)*

### Abstrak

*Perkembangan teknologi komunikasi yang semakin pesat, menghadirkan alat komunikasi yaitu telepon seluler (ponsel) yang menyediakan fasilitas pelayanan komunikasi lewat SMS yang bisa mengirimkan teks atau pesan singkat dengan cepat, fasilitas ini lebih praktis dan lebih ekonomis. Pemanfaatan SMS sebagai sarana layanan informasi dapat dibuat sebuah basis data yang bisa memberikan layanan informasi kepada seseorang atau sekelompok orang yang membutuhkannya. Dalam membangun Sistem berbasis Sms Gateway ada beberapa cara yang digunakan, bisa dengan tambahan aplikasi open source seperti Gammu dan ada juga yang memanfaatkan fitur Programming seperti GSM Visual, tetapi aplikasi tersebut masih membutuhkan base instruction under windows untuk mengakses kernel port output sms gateway, yaitu AT Command . AT Command adalah salah satu fitur windows yang menampung perintah-perintah yang digunakan dalam komunikasi dengan serial port. Dengan AT Command kita dapat mengetahui vendor dari Handphone yang digunakan, kekuatan sinyal, membaca pesan yang ada pada SIM Card, mengirim pesan, mendeteksi pesan SMS baru yang masuk secara otomatis, menghapus pesan pada SIM Card dan masih banyak lagi.*

*untuk memanfaatkan AT Command pada sistem berbasis Sms Gateway dibutuhkan Design Programming dalam decode ncodec pesan dan juga basis data sebagai basis penyimpanan data pesan (Message Box).*

**Kata kunci :** Technology, SMS, Gateway, AT Command, Programming

### I. PENDAHULUAN

#### 1.1 Latar Belakang

Seiring dengan perkembangan teknologi informasi dan komunikasi, sistem informasi memberikan peran yang sangat penting dalam dunia bisnis sehingga seringkali orang menggunakan keunggulan sistem informasi yang ia gunakan sebagai kunci strategi bisnis.

Sama halnya dengan perkembangan Teknologi Informasi, SMS (*Short Messaging Service*) yang merupakan salah satu fitur sistem seluler GSM (*Global System for Mobile Communication*), dimana dikembangkan dan distandarisasi oleh ETSI (*European Telecommunication Standard Institute*),

juga sebagai fasilitas yang banyak digunakan masyarakat sekarang karena SMS memiliki tarif yang sangat murah dibandingkan jika melakukan percakapan secara langsung dengan nomor yang dituju. SMS menjadi fenomena tersendiri, dalam waktu yang cukup singkat, tingkat pertumbuhannya sangat tinggi tanpa ada penurunan tarif yang berarti, bahkan dapat dikatakan tarifnya mengambil posisi *steady state*. Biasanya, bahkan dalam kasus layanan telepon bergerak, tarif akan turun seiring dengan meningkatnya penggunaan. Fakta lainnya adalah fasilitas SMS dalam telepon bergerak ternyata punya andil cukup, dalam menarik kaum muda masuk ke pasar telepon bergerak.

## 1.2 Tujuan dan Manfaat

Adapun tujuan dari penulisan ini adalah untuk mengidentifikasi perangkat /*module* apa saja yang dibutuhkan untuk memanfaatkan *AT Command* sebagai perangkat lunak utama sms gateway.

Manfaat dari penelitian ini adalah untuk mengembangkan *fiture* dan *function* *AT Command* yang selanjutnya akan berguna bagi efisiensi dan fleksibilitas kinerja Sms Gateway pada sistem informasi.

## 1.3 Rumusan Masalah

Bagaimana membangun perangkat lunak manajemen *AT Command* untuk mendukung rancang bangun sistem berbasis sms gateway.

## 1.4 Pembatasan Masalah

Pembatasan masalah dalam penelitian ini meliputi:

1. Hanya memaparkan manajemen *AT Command* tanpa membahas perancangan sistem sms gateway dan peripheral yang digunakannya.
2. Bahasa pemrograman yang digunakan adalah Visual Basic 6.0.

## II. TINJAUAN UMUM CARA KERJA SMS, AT COMMAND, DAN VB MSCOMM API

### 2.1 Cara Kerja SMS

SMS dikirim dari pengirim ke penerima melewati SMSC dengan prinsip *Store and Forward*, dimana pesan yang dikirim ke SMSC akan disimpan terlebih dahulu hingga masa validitas tertentu terpenuhi jika ponsel nomor yang dituju dalam keadaan mati ataupun diluar jangkauan operator, setelah ponsel nomor yang dituju sudah aktif atau berada dalam jangkauan operator maka pesan akan diteruskan oleh SMSC kepada penerima. Apabila pesan yang tersimpan di SMSC

sudah melewati masa validitas yang ditentukan, pesan tersebut akan dihapus dan tidak akan diteruskan kepada nomor yang dituju [7].



Gambar 2.1 Skema Pengiriman SMS.

Dengan adanya SMSC, pengirim SMS juga dapat mengetahui status dari pesan yang dikirim, apakah pesan tersebut sudah diterima atau belum.

Pada pengiriman dan penerimaan SMS, ada 2 mode format SMS yang digunakan oleh operator maupun terminal. Mode yang pertama adalah mode PDU (*Protocol Data Unit*), dimana format pesan dalam bentuk oktet heksadesimal dan oktet semidesimal dengan panjang mencapai 160 (7 bit) atau 140 (8 bit) karakter. Sedangkan mode yang kedua adalah mode teks, dimana pesan dalam bentuk teks asli. Akan tetapi, tidak semua operator GSM ataupun terminal di Indonesia mendukung format pesan mode teks. Oleh karena itu, format pesan yang digunakan dalam penyusunan tugas akhir ini adalah mode PDU.

Dalam proses pengiriman pesan, dikenal dua jenis *mobile*, yaitu Ponsel Pengirim (*Mobile Originated*) dan Ponsel Penerima (*Mobile Terminated*). Kedua *mobile* tersebut memiliki metode yang berbeda dalam pengolahan pesan. Pada Ponsel pengirim, metode yang digunakan adalah *encodec*, sedangkan pada Ponsel penerima, metode yang digunakan adalah *decodec*. Pada metode *encodec*, pesan dalam bentuk teks akan diubah dalam format PDU. Hal itu disebabkan SMSC hanya dapat menerima pesan dalam format PDU sehingga pesan yang akan dikirim harus diubah terlebih dahulu kedalam format PDU dan proses pengubahan ini berlangsung pada terminal atau ponsel. Sedangkan metode *decodec*

adalah kebalikan dari *encodec*, dimana pesan diterima dan SMSC dalam format PDU kemudian diubah menjadi format teks.

Selain metode pengolahan pesan yang berbeda, Ponsel Pengirim dan Ponsel Penerima juga memiliki skema format SMS PDU yang berbeda, dimana skema ini sudah diatur dan distandarisasi oleh ETSI. Pada makalah ini hanya akan dibahas mengenai SMS PDU Penerima karena jenis SMS inilah yang digunakan pada aplikasi yang dibangun. SMS PDU Penerima adalah pesan yang dikirim dari SMSC ke ponsel tujuan dalam format PDU. Pada aplikasi yang dibangun pada tugas akhir ini, SMS PDU Penerima inilah yang akan digunakan, untuk kemudian diubah kedalam bentuk teks agar dapat dibaca. Skema dari format SMS PDU Penerima adalah:

SCA	PDU Type	OA	PID	DCS	SCTS	UDL	UD
-----	----------	----	-----	-----	------	-----	----

Gambar 2.2 Skema Format SMS PDU Penerima.

## 2.2 AT Command

*AT Command* adalah perintah-perintah standar yang digunakan untuk melakukan komunikasi antara komputer dengan ponsel melalui serial port. Melalui *AT Command*, data-data yang ada di dalam ponsel dapat diketahui, mulai dari vendor ponsel, kekuatan sinyal, membaca pesan, mengirim pesan, dan lain-lain. Tabel 2.7 menunjukkan beberapa perintah *AT Command* yang sering digunakan[2].

AT Command	Keterangan
AT	Mengecek apakah ponsel telah
AT + CMGF	Menetapkan format mode dari
AT + CSCS	Menetapkan jenis
AT + CNMI	Mendeteksi pesan SMS baru masuk secara
AT + CMGL	Membuka daftar SMS yang ada pada kartu
AT + CMGS	Mengirim pesan SMS

AT + CMGR	Membaca pesan SMS
ST + CMGD	Menghapus pesan SMS

Tabel 2.1 Beberapa Perintah *AT Command*

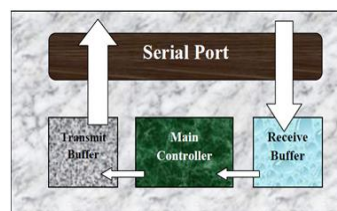
## 2.3 VB MSCComm API

MSComm menyediakan fasilitas komunikasi antara program aplikasi yang kita buat dengan port serial untuk mengirim atau menerima data melalui port serial [1]. Setiap MSComm hanya menangani satu port serial sehingga jika kita ingin menggunakan lebih dari satu port serial harus digunakan MSComm lain. Fungsi dari komponen ini adalah sebagai berikut :

- Mengadakan hubungan dengan *serial port* PC.
- Berhubungan dengan alat komunikasi lain ( contoh: *modem* )
- Melakukan pertukaran data
- Memonitor dan merespon *event* dan *error* yang terjadi pada hubungan *serial*.

Untuk mengadakan suatu komunikasi serial antara 2 peralatan, kita harus melakukan beberapa langkah.

- Membuka *serial port*
- Mengatur *serial device*
- Setting Receive and Transmit Buffer Properties*
- Managing Receive and Transmit Buffer*



Gambar 2.3 Arsitektur Serial.

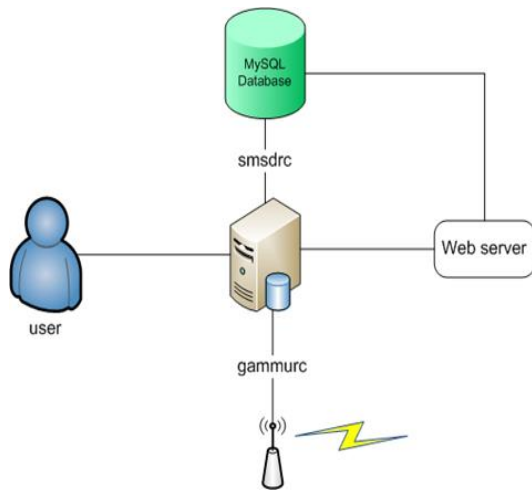
Pada komunikasi serial, bit-bit data yang masuk akan ditampung dulu di *receive buffer* sebelum dieksekusi oleh *main controller*. Demikian pula sebelum dikirimkan ke luar, data akan ditampung dulu di *transmit buffer*[1].

## III. PEMBAHASAN

### 3.1 AT Commands dan Software Sms Gateway

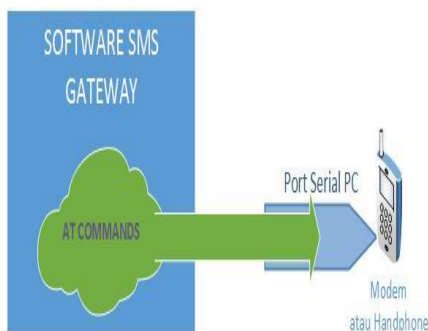
Di lingkungan IT, masih banyak orang yang belum mengerti bagaimana sebuah *software Sms*

Gateway seperti Gammu, GNoki, Kannel, Ozeki SMS dan lainnya mampu berkomunikasi secara dinamis dengan perangkat modem dan juga Handphone.



Gambar 3.1.1 Skema Design Gammu

*Software Sms Gateway* (SG) diciptakan untuk memudahkan pengembang aplikasi berbasis Sms gateway menggunakan interaksi komunikasi SMS sebagai perangkat efisien dalam dunia Teknologi Informasi. Sebagai contoh *software Gammu*, salah satu *Software SG* yang memerlukan *software* tambahan sebagai *counter* data yaitu **MySql** dan juga **PHP** untuk mengeksekusi *AT Commands*, Lihat gambar 3.1.1. *Software Sms Gateway* meneruskan intruksi-intruksi yang berkaitan dengan SMS untuk selanjutnya menjalankan *AT Commands* yang sudah tertanam di *Software SG* sebelumnya dan mengeksekusi intruksi-intruksi tersebut seperti gambar 3.1.2.



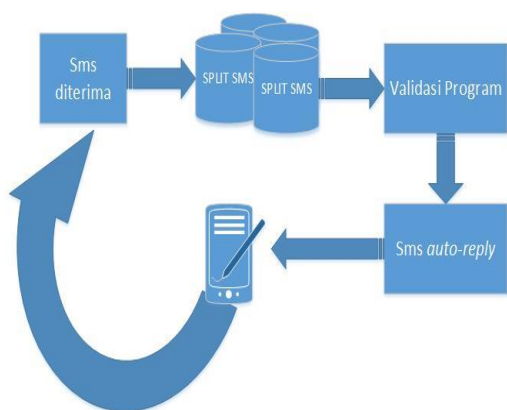
Gambar 3.1.2 *AT Commands* di *Software SG*

Dari beberapa gambar diatas terlihat bahwa *software SG* adalah *software Sms Gateway* yang sudah jadi dengan bahasa programming dan *software database* tertentu. Masalahnya jika ada pengembang sistem informasi berbasis SG yang menggunakan bahasa pemrograman berlainan dengan bahasa pemrograman yang dipakai *software SG*, maka mengakibatkan pengembang tersebut terpaksa harus mengkombinasikan program tersebut, atau bisa juga dengan *translating source code* dari bahasa pemrograman *Software SG* ke bahasa pemrograman yang akan dipakai dan itu membutuhkan waktu lama. Belum juga dengan *software database* yang dipakai, mengharuskan pengembang memakai *software database* yang sama dengan yang dipakai *Software SG*. Jadi *Software SG* Sangat tidak fleksibel pada kondisi seperti itu.

### 3.2 Implementasi *AT Commands* ke Bahasa Pemrograman Visual Basic 6

Disini penulis menggunakan bahasa pemrograman/*Compiler* Visual Basic 6.0 (VB6) untuk mengeksekusi perintah-perintah *AT Commands*, modem type huawei dan juga Windows 7 sebagai basic sistem operasi PC. Spesifikasi unit PC yang digunakan adalah CPU B960 2,20GHz dengan RAM 6GB serta hardisk 500GB.

Untuk menciptakan sistem informasi berbasis sms gateway yang dinamis, programmer harus memperhatikan siklus kebutuhan sistem yang nantinya akan menjadi inti dalam sistem informasi tersebut. Siklus tersebut menentukan bagaimana skema kerja sistem, seperti terlihat pada gambar dibawah ini :



Gambar 3.2.1 Skema Design Gammu

### 3.2.1 Sinkronisasi *Compiler* dengan *Port*

Dalam konfigurasi koneksi port yang harus di perhatikan adalah nomor port serial usb yang di gunakan modem, dengan nomor port tersebut program mampu me-root IO port dan memastikan validasi perangkat yang digunakan dan meminta sinkronisasi perangkat.

Pada tahap ini kita perlu memastikan bahwa pengaturan protokol komunikasi serial yang digunakan pada peralatan lain yang kita akses, sesuai dengan pengaturan pada komputer yang kita pakai.

Ada beberapa *property* dari *receive buffer* dan *transmit buffer* (*porperty* dari *MSComm*) yang perlu kita atur.

- InBufferSize* : mengatur ukuran *receive buffer*
- OutBuffer Size* : mengatur ukuran *transmit buffer*
- Rthreshold* : menentukan jumlah karakter yang diterima oleh *receive buffer* sebelum *OnComm event* dipicu
- Sthreshold* : menentukan jumlah karakter yang diterima oleh *transmit buffer* sebelum *OnComm event* dipicu. Nilai 0 berarti tidak pernah dipicu, sedangkan nilai 1 berarti dipicu setiap satu karakter.
- InputLen* : menentukan jumlah karakter yang dibaca CPU dari *receive buffer*
- InputMode* : menentukan tipe data input yang akan dibaca CPU. *comInputModeText* untuk data *string*/teks dan *comInputModeBinary* untuk data biner. Untuk contoh *source code* bisa dilihat pada *code* dibawah ini:

```

With MSComm1
.CommPort = 7
.Settings = "115200,N,8,1"
.Handshaking = comRTS
.RTSEnable = True
  
```

```

.DTREnable = True
.RThreshold = 1
.SThreshold = 1
.NullDiscard = True
.InputMode = comInputModeText
.InputLen = 0
.PortOpen = True
  
```

End With

Penulis memakai *comport 7* untuk menghubungkan perangkat modem dengan PC, dan selanjutnya memakai control API vb6 MScomm dengan nama MSComm1 dengan mencantumkan nomor port tersebut.

Untuk mengetahui status koneksi modem, perlu menambahkan *source code* pengecekan modem dengan perintah *AT Command* dibawah ini

```

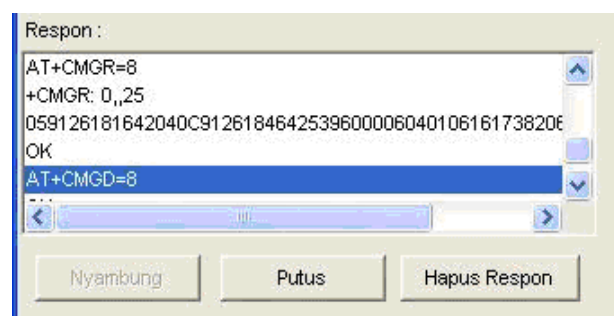
MSComm1.output = AT
  
```

Jika sudah terkoneksi dengan baik, MSComm.input akan memberi jawaban "OK".

### 3.2.2 Manajemen SMS

#### SMSmasuk

Seperti tampak pada Gambar 3.2.2.1 dibawah ini, saat ada pesan yang masuk ke ponsel penerima, kelas SMSmasuk langsung melakukan pembacaan terhadap pesan tersebut dan ditampilkan pada *list Respon..*



Gambar 3.2.2.1 Membaca pesan masuk.

Pesan yang ditampilkan ini tentunya masih dalam format PDU karena belum dilakukan pengubahan ke bentuk teks, untuk mengubahnya perlu *source code* :  
 MSComm1.output = AT+CMGF=mode  
 mode "0" = format PDU  
 mode "1" = format Text

Contoh:

`MSCComm1.output = AT+CMGF="1"`

Seperti halnya aktivitas mendengarkan pesan masuk pada kelas Port, untuk membaca pesan masuk juga dibutuhkan *AT Command*. Dalam hal ini, *AT Command* yang digunakan adalah AT+CMGR.

`MSCComm1.output = AT+CMGR=IndexSms`

Atau dengan menampilkan semua daftar sms yang ada di *SIM Card* tersebut :

`MSCComm1.output = AT+CMGL="ALL"`

### 3.2.3 Splitting Format SMS

Untuk pemecahan format sms, ada banyak cara untuk menyelesaikannya, dengan pengecekan satu persatu atau dengan intruksi kode yang sudah disediakan compiler. disini penulis memakai *code split* yang sudah disiapkan oleh vb6 untuk melakukan pembagian kata per kata pada sms.

```
split(<<kalimat>>,"<<string>>")(<<no split>>)
```

Misalkan penulis memakai pemisah <<spasi>> dalam penggabungan lebih dari satu informasi yang diperlukan yang berupa sebuah format SMS seperti : *Reg Start ridwan pria 1988*

maka *source code* yang di perlukan :

```
for i=1 to 5  
textsplit[i] = split("Start ridwan pria 1988", "  
") (i)  
next i
```

*text* yang sudah terpisah akan tersimpan dalam array data *textsplit* yang selanjutnya bisa ditampilkan dan dimanfaatkan untuk validasi program.

### 3.2.4 Auto-reply SMS

Secara dinamis sistem akan otomatis menganalisa data sms yang diterima dan telah di-*split* sebelumnya, dengan cara mencocokkan kebutuhan bisnis yang di-*request* oleh *client* untuk selanjutnya diteruskan ke *client* untuk dijadikan informasi yang bermanfaat bagi *client*.

Fasilitas sender *AT Command* diwakili oleh *code* AT+CMGS dan diikuti nomor telephone serta pesan sms yang akan

dikirim, dengan adanya proses ini admin program tidak perlu melakukan pengiriman sms balasan untuk pelanggan. skema *source code* nya seperti dibawah ini :

```
MSCComm1.Output = "AT+CMGS=" & "" &  
<<tSMSNum>> & "" & vbCr &  
<<tMessage>> & Chr(26))
```

tSMSNum adalah nomor telephone tujuan sedangkan tMessage adalah pesan yang akan dikirim.

## IV. KESIMPULAN

*AT Command* adalah perintah-perintah standar yang digunakan untuk melakukan komunikasi antara komputer dengan ponsel melalui serial port, dengan adanya itu programmer yang akan mengembangkan sistem informasi berbasis Sms Gateway ini tidak perlu menggunakan *software* jadi untuk mengaplikasikan Sms Gatewaynya, dan *AT Command* ini lebih fleksibel dalam pemilihan *software* database, berbeda dengan *Software* Sms Gateway yang databasenya harus sama dengan database *software*-nya karena *software*, dengan begitu optimalisasi Sistem Informasi Berbasis SMS Gateway akan terealisasikan.

Penulis menyadari untuk menjadikan *AT Command* menjadi sebuah alat yang penting dalam sistem informasi sms gateway, penulis tidak hanya memerlukan satu atau dua kali penelitian, penulis perlu berkali-kali percobaan dan pengetesan terutama pada masalah dukungan *AT Command* atas tipe-tipe modem yang bisa dikelola untuk membangun sistem informasi tersebut, selain itu penulis kedepannya ingin mengaplikasikan *AT Command* ini untuk *compiler-compiler* selain visual basic 6 yang akan memudahkan dunia programming dalam pemakaian fitur sms gateway tersebut.

## DAFTAR PUSTAKA

1. Telit Corporation., *AT Command Reference Guide, 80000ST10025a Rev. 0 - 04/08/06*
2. Ubenetic Ltd.,*GSM AT Command Set. Ubenetics 2001*
3. Ubox, *AT Command Example, Aplication Note, Revisi 22-Sep-2016*
4. sapta., *sekilas tentang sms gateway,*

anggoro, adheo., *komunikasi serial dalam visual basic* 6,  
<http://aodhio.blogspot.com/2012/05/komunikasi-serial-dalam-visual-basic-60.html>

5. Berda Setya A., Agung Budi P., *Aplikasi mp3 player Berdasarkan polling sms*