

## **II. TINJAUAN PUSTAKA**

### **A. Sistem Kendali**

#### **1. Pengertian Sistem Kendali**

Pertamkali untuk mengetahui pengertian sistem kendali diperlukan mendefinisikan pengertian sistem terlebih dahulu. Sistem adalah kombinasi dari beberapa komponen yang bekerja bersama-sama dan melakukan suatu sasaran tertentu. Sehingga sistem kendali adalah suatu sistem yang keluaran sistemnya dikendalikan pada suatu nilai tertentu atau untuk mengubah beberapa ketentuan yang telah ditetapkan oleh masukan ke sistem. Sebagai contoh adalah sebuah kendali suhu pada sistem pusat pemanasan di sebuah rumah, mempunyai masukan dari thermostat atau panel kendali yang telah ditentukan suhunya dan menghasilkan keluaran berupa suhu aktual. Suhu ini diatur dengan sistem kendali sehingga sesuai dengan nilai yang ditentukan oleh masukan pada sistem.

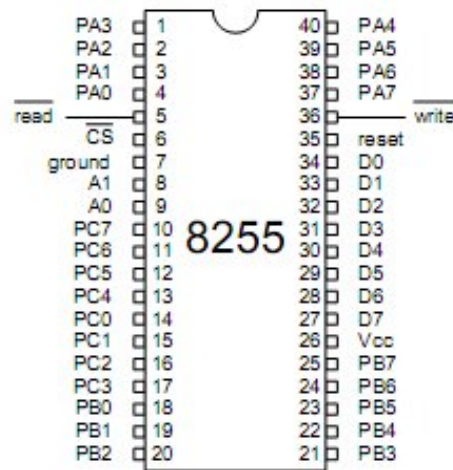
## **2. Sistem Kendali Kalang Terbuka dan Sistem Kendali Kalang Tertutup**

Sistem kendali kalang terbuka adalah sistem kendali yang keluarannya tidak berpengaruh pada aksi pengontrolannya. Jadi pada sistem kendali kalang terbuka, keluaran tidak diukur atau diumpan-balikkan untuk dibandingkan dengan masukan. Sedangkan sistem kendali kalang tertutup adalah sistem kendali yang sinyal keluarannya mempunyai pengaruh langsung pada aksi pengontrolan. Jadi sistem kendali kalang tertutup adalah sistem kendali berumpan balik yang menggunakan aksi umpan balik untuk memperkecil kesalahan sistem.

### **B. Komponen Pengendali**

#### **1. PPI 8255 (*PROGRAMMABLE PERIPHERAL INTERFACE 8255*)**

PPI 8255 adalah IC yang dirancang untuk membuat port masukan dan keluaran paralel, IC ini mempunyai 24 bit I/O yang terorganisasi menjadi tiga port 8 bit ( 24 jalur ) dengan nama Port A, Port B, dan Port C.



Gambar 2.1 Pin IC PPI 8255

Konfigurasi dari 24 jalur I/O ini bisa digunakan untuk masukan, keluaran, ataupun *bidirectional* (dua arah). Pada I/O, yang dikontrol secara *software* akan lebih mudah bila dibandingkan dengan pengontrolan secara *hardware*.

### 1.1. Deskripsi fungsi 8255

#### a. Data bus buffer

*Buffer bidirectional three state* ini digunakan untuk antarmuka 8255 ke sistem bus data, data dikirim dan diterima oleh *buffer* berdasarkan eksekusi *input* atau *output* dari CPU. *Control Word* dan *information status* juga dikirimkan melalui *buffer* data bus.

#### b. Read/Write dan kontrol logika.

Fungsi dari blok ini adalah untuk mengatur semua pengiriman baik internal maupun eksternal dari data dan *Control Word*. Blok ini menerima *input* dari

alamat CPU dan bus kontrol dan selanjutnya blok ini mengirimkan perintah ke kedua *group* kontrol.

*c. Chip Select*

*Chip Select*, jika *logika low* pada pin input ini maka komunikasi antara 8255 dan CPU akan *enable*.

*d. Read*

*Read*, jika *logika low* pada pin input ini maka 8255 akan mengirimkan data atau status informasi ke CPU pada bus data.

*e. Write*

Jika *Logika low* pada pin input ini maka CPU dapat menulis data atau kata kontrol ke 8255.

*f. A0 dan A1*

*Port select 0* dan *port select 1*, sinyal input ini berhubungan dengan input RD dan WR, mengontrol pemilihan satu dari tiga port atau register kontrol pin tersebut umumnya dihubungkan ke *least significant bit* dari *bus address* (A0 dan A1).

*g. Reset*

Logika *high* pada input pin ini akan menyebabkan *reset* pada *register control* dan semua port (A,B,C) akan berfungsi dalam mode *input*.

*h. Port A, B dan C*

8255 terdiri atas tiga buah port 8 bit (A, B dan C). semuanya dapat dikonfigurasi dalam berbagai variasi fungsi bergantung pada sistem software yang diberikan.

Port A (PA0-PA7). 8 bit data *Output latch buffer* dan 8 bit data *input latch*.

Port B (PB0-PB7). 8 bit data *Output latch buffer* dan 8 bit data *input latch*.

Port C (PC0-PC7). 8 bit data *Output latch buffer* dan 8 bit data *input latch*.

Tiap 4 bit port terdiri atas 4 bit *latch* dan dapat digunakan untuk sinyal output kontrol dan sinyal input status.

### *1.2. Control Group*

*Control group* dibagi menjadi dua *group*, yaitu *group A* dan *group B*. *Group* tersebut menerima Read/Write Control.

*Control group A* digunakan untuk :

- a. Mengatur port A yang bisa disetting sebagai *input/output latch buffer*
- b. Mengatur 4 upper bit (C4..C7), port C sebagai *input buffer* atau *output latch/buffer* jika bekerja pada mode 0.
- c. Mengatur 4 upper bit (C4..C7), port C sebagai *control group A* jika bekerja pada mode 1 atau 2.

*Control group B* digunakan untuk :

- a. Mengatur port B yang bisa di-setting sebagai *latch buffer input/output*
- b. Mengatur 4 lower bit (C0..C3), port C sebagai *input buffer* atau *output latch/ buffer* jika bekerja pada mode 0

- c. Mengatur 4 lower bit (C0..C3), port C sebagai *control group* B jika bekerja pada mode 1 atau 2

Tabel 2.1. Operasi dasar 8255

<b>A1</b>	<b>A0</b>	<b>RD</b>	<b>WR</b>	<b>CS</b>	<b>Operasi Read/Write</b>
0	0	0	1	0	Port A → Data Bus
0	1	0	1	0	Port B → Data Bus
1	0	0	1	0	Port C → Data Bus
0	0	1	0	0	Data Bus → Port A
0	1	1	0	0	Data Bus → Port B
1	1	1	0	0	Data Bus → Port C
X	X	X	X	1	Data Bus → Three State
1	1	0	1	0	Illegal Condition
X	X	1	1	0	Data Bus → Three State

### 1.3. Mode/Protokol komunikasi

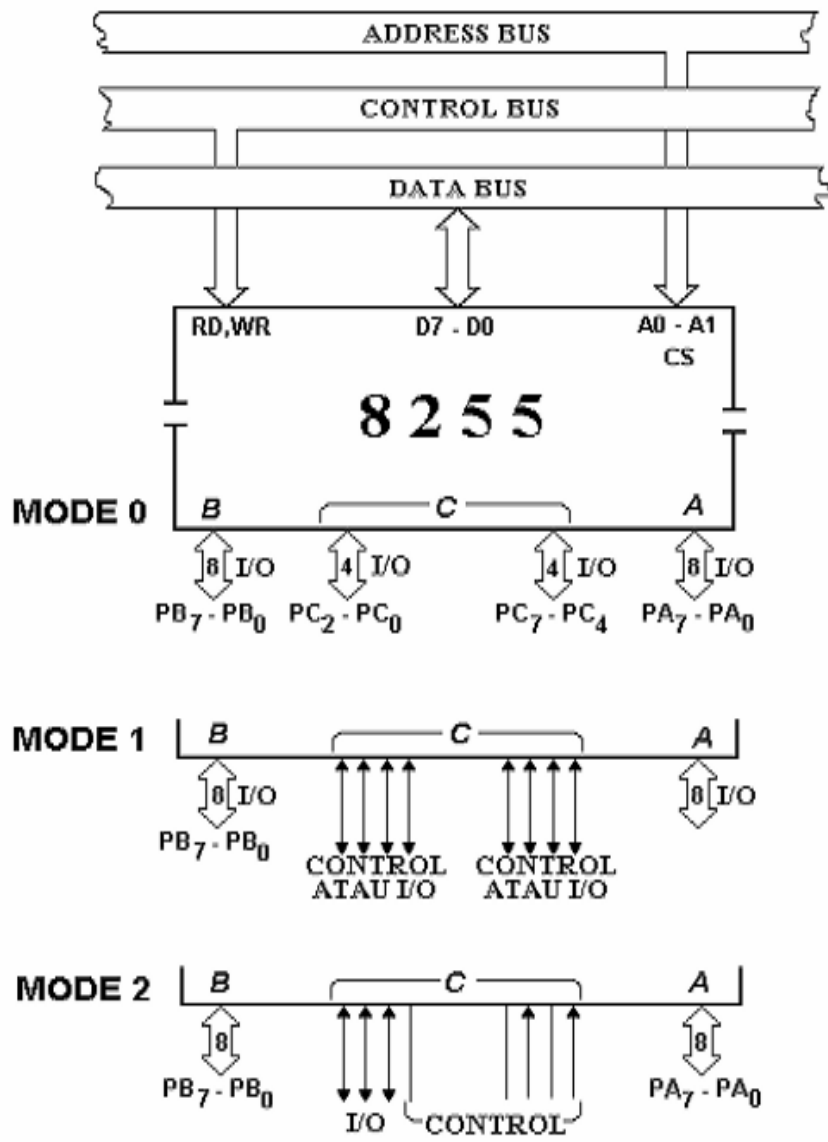
Transfer data pada PPI 8255 dibagi menjadi tiga protokol komunikasi:

1. Mode 0 (*Simple protocol*)/*Basic input-output*
2. Mode 1 (*Single handshaking protocol*)/*Strobed input-output*
3. Mode 2 (*Double handshaking protocol*)/*Bi-directional bus*

Transfer data pada mode 0 tidak memerlukan sinyal khusus yang menandakan apakah telah terjadi transfer data atau belum. Semua Port I/O dipakai sebagai Input dan Output. Tidak diperlukan sinyal “*Handshake*”. Data langsung ditulis atau dibaca dari port yang bersangkutan.

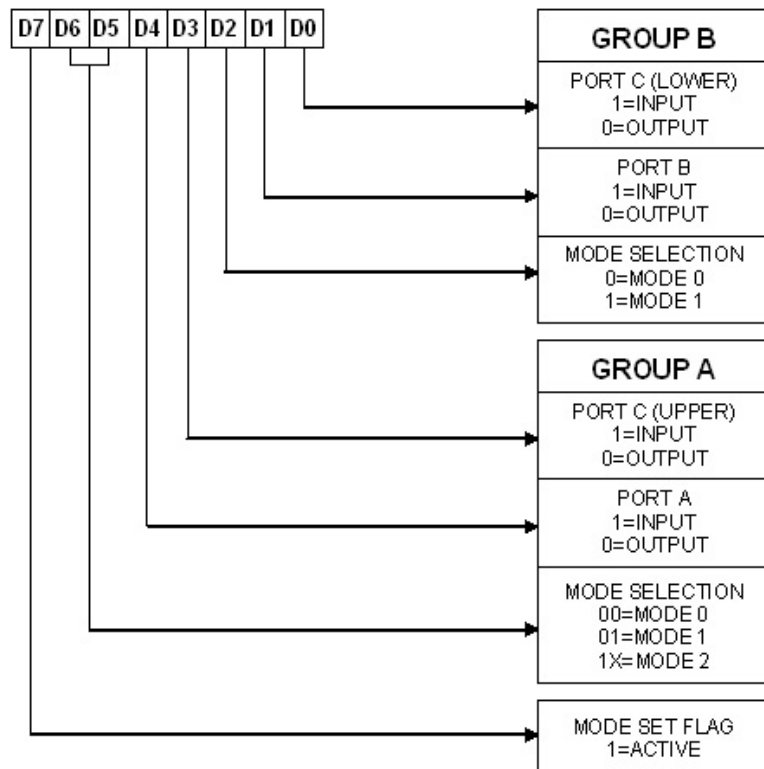
Fungsi dasar dari 8255 mode 0 adalah :

1. Dua Port-8 bit (Port A & B) serta 2 Port - 4 bit (Port C).
2. Setiap Port dapat dipakai sebagai *input* atau *output*.
3. Output di-*latch* (ditahan); input tidak di-*latch* (ditahan).
4. Menyediakan 16 kombinasi konfigurasi *input/output* pada mode ini.



Gambar 2.2. Group Control PPI 8255





Gambar 2.3. Format *Control Word* PPI 8255

*Control Word Port* digunakan untuk inisialisasi awal yang menentukan PPI 8255 bekerja pada mode 0, 1, 2 dan menentukan port-port mana saja yang digunakan sebagai *input* dan *output* serta sebagai sinyal pengendali.

#### 1.4. *Set/Reset Bit*

Pada PPI 8255 terdapat port untuk set dan reset sebuah bit, di mana jika terjadi Set atau Reset hanya salah satu port yang dipakai pada Port C.

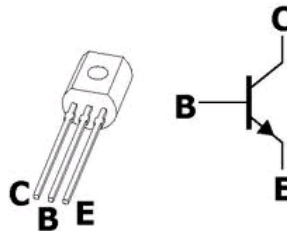
Contoh:

1. Jika Port C saat ini datanya adalah FFH (1111 1111), jika kita akan *reset* Port C5 (PC5) maka port C hasilnya adalah EFH (1101 1111).

2. Jika Port C saat ini datanya adalah 1FH (0001 1111), jika kita akan me-*set* Port C7 (PC7) maka Port C hasilnya adalah 9FH (1001 1111).

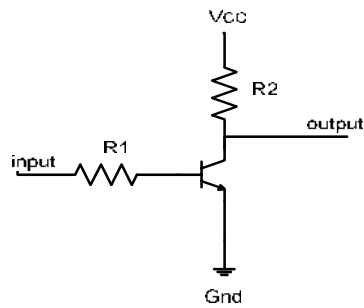
## 2. Transistor

Transistor adalah [alat semikonduktor](#) yang dipakai sebagai penguat, sebagai sirkuit pemutus dan penyambung (*switching*), stabilisasi tegangan, modulasi sinyal atau sebagai fungsi lainnya.



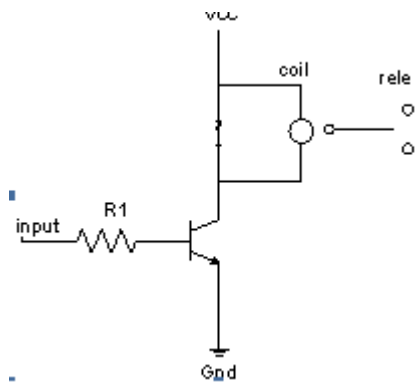
Gambar 2.4. Transistor

Untuk membentuk saklar dengan transistor, transistor hanya perlu dioperasikan pada dua keadaan, yaitu keadaan *on* dan *off*. Cara yang mudah untuk menggunakan sebuah transistor sebagai saklar adalah dengan mengoperasikan transistor pada salah satu keadaan saturasi atau *cutoff* tetapi tidak sepanjang garis beban. Jika sebuah transistor berada dalam keadaan saturasi, transistor tersebut seperti sebuah saklar yang tertutup dari kolektor ke emitor. Pada umumnya saklar dengan transistor bipolar menggunakan konfigurasi *common emitter*. Pada gambar 2. 12 ditunjukkan rangkaian transistor sebagai saklar.



Gambar 2.5. Transistor Sebagai Saklar

Transistor sebagai saklar dapat digunakan sebagai pengendali rele. Adpaun jenis aplikasinya ditunjukkan pada Gambar 2.13.



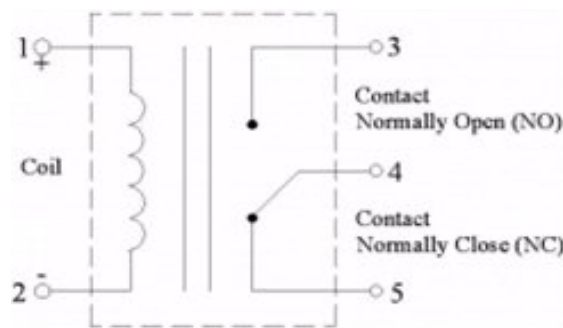
Gambar 2.6. Transistor Pengendali Rele

Apabila tegangan rangkaian sebelumnya berada dalam keadaan rendah (0 Volt), basis emitor mendapat bias mundur sehingga mengakibatkan transistor terputus dan tidak ada arus yang mengalir ke kumparan rele, sehingga kontak rele tidak diaktifkan. Transistor hidup (saturasi) bila tegangan basis emitor lebih besar dari 0,7 volt untuk transistor yang terbuat dari silicon dan 0,3 volt untuk transistor

yang terbuat dari germanium. Dalam keadaan ini, transistor terhubung dan rele menjadi aktif.

### 3. Rele

Rele adalah sebuah piranti elektro mekanik yang dioperasikan dengan listrik yang secara mekanis mengontrol penghubungan rangkaian listrik. Relay adalah bagian yang penting dari banyak sistem kontrol, bermanfaat untuk kontrol jarak jauh dan untuk pengontrolan alat tegangan dan arus tinggi dengan sinyal kontrol tegangan dan arus rendah. Ketika arus mengalir melalui elektromagnet pada relay kontrol elektromekanis, medan magnet yang menarik lengan besi dari jangkar pada inti terbentuk. Akibatnya, kontak pada jangkar dan kerangka relay terhubung. Relay dapat mempunyai kontak NO atau kontak NC atau kombinasi dari keduanya. Relay berfungsi untuk memutuskan atau mengalirkan arus listrik yang dikontrol dengan memberikan tegangan dan arus pada koilnya. Ada dua jenis relay berdasarkan tegangan untuk menggerakkan koilnya yaitu AC dan DC.



Gambar 2.7. Bagian Rele

( Sumber : <http://focus.ti.com/lit/ds/symlink/rele.pdf> )

#### **4. *Light Dependent Resistor (LDR)***

LDR terdiri dari sebuah cakram semikonduktor yang mempunyai dua buah elektroda pada permukaannya. Pada saat gelap maka LDR menjadi konduktor yang buruk karena bahan dari cakram tersebut menghasilkan elektron bebas dengan jumlah relatif kecil. Sehingga hanya sedikit elektron untuk mengangkut muatan elektrik

#### **5. Modem**

Modem berasal dari singkatan Modulator Demodulator. Modulator merupakan bagian yang mengubah sinyal informasi kedalam sinyal pembawa (Carrier) dan siap untuk dikirimkan, sedangkan Demodulator adalah bagian yang memisahkan sinyal informasi (yang berisi data atau pesan) dari sinyal pembawa (carrier) yang diterima sehingga informasi tersebut dapat diterima dengan baik.

Kebanyakan modem yang digunakan di PC atau laptop dewasa ini adalah dengan menggunakan teknik *asynchronous*. Data dari komputer yang berbentuk sinyal digital diberikan kepada modem untuk diubah menjadi sinyal analog. Sinyal analog tersebut dapat dikirimkan melalui beberapa media telekomunikasi seperti telepon dan radio. Setibanya di modem tujuan, sinyal analog tersebut diubah menjadi sinyal digital kembali dan dikirimkan kepada komputer. Modem yang akan digunakan dalam penelitian ini adalah modem wavecom supreme frashtrack 10.

### *5. 1. Modem Wavecom Supreme Frashtrack 10*

Modem wavecom supreme 10 dan wavecom supreme 20 bersifat diskrit, selular plug tidak rata dan menggunakan CPU tanpa kawat yang menawarkan penggunaan jaringan GSM/GPRS yang menghubungkan mesin ke mesin aplikasi.

Terbukti untuk dipercaya kemampuan jaringan wireless yang stabil di seluruh dunia, Wavecom generasi terakhir pada frashtrack supreme untuk memenuhi perkembangan pasar yang cepat.

Mempunyai ukuran yang hampir sama dengan generasi sebelumnya M1306B dan diperbaharui fitur yang baru, frashtrack supreme menawarkan penghubung *internal expansion socket* (IES) untuk digunakan pelanggan.

Frashtrack supreme 10 dan 20 menawarkan kemampuan GPRS kelas 10. Dan keduanya didukung penuh dengan menggunakan software AT Command.

## **C. Perangkat Lunak**

### **1. Visual Basic 6.0**

#### *1.1 Pengertian Visual Basic 6.0*

Visual Basic 6.0 merupakan salah satu bahasa pemrograman yang dapat digunakan untuk menyusun dan membuat program aplikasi pada lingkungan sistem operasi Windows. Dengan menggunakan Visual Basic 6.0, kemampuan Windows dapat dimanfaatkan secara optimal. Kecanggihannya yang dimiliki oleh Visual Basic 6.0 akan menjadikan betapa mudahnya

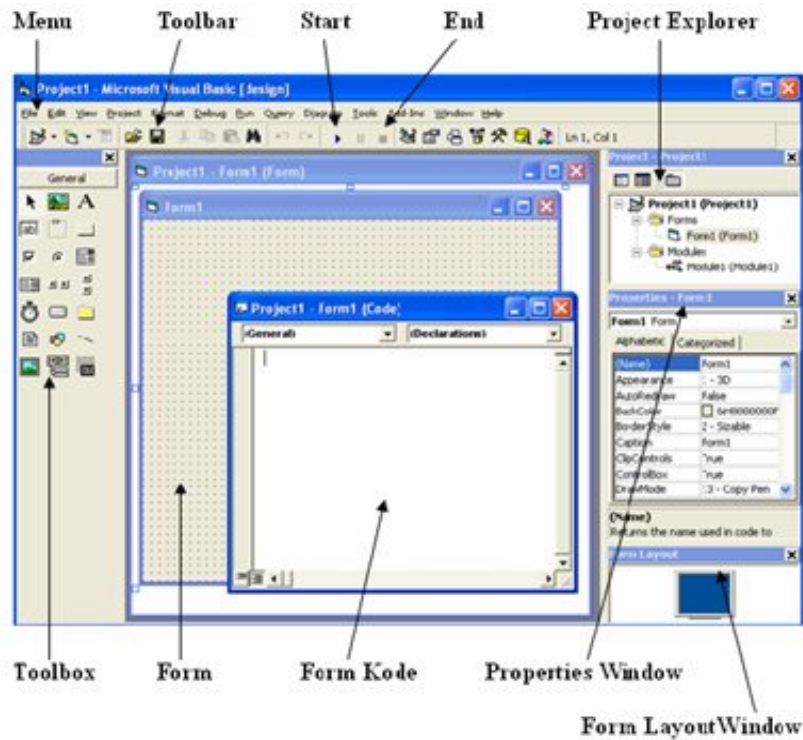
menyusun program aplikasi dengan tampilan grafis yang menawan dalam waktu yang relatif singkat. Program aplikasi dapat berupa program *database*, program grafis, program kendali, dan lain sebagainya. Di dalam Visual Basic 6.0 sudah terdapat komponen-komponen yang sangat membantu pembuatan program aplikasi.

Beberapa keuntungan menggunakan Visual Basic 6.0 daripada bahasa pemrograman yang lain di antaranya :

- 1) Tampilan grafis (*under Windows*) sehingga lebih “bersahabat”.
- 2) Cara pemrograman relatif lebih mudah sehingga cocok untuk segala tingkat programmer.
- 3) Hubungan dengan perangkat luar (*hardware*) tidak begitu rumit sehingga cukup mudah untuk meng-implementasikan sebagai pengendali peralatan elektronik.

### 1.2. IDE Visual Basic 6.0

Langkah pertama dalam membuat program aplikasi dengan Visual Basic 6.0 adalah membuat sebuah *project*. Pembuatan sebuah project dapat dilakukan dengan beberapa cara, di antaranya dengan meng-klik **Start | Program | Microsoft Visual Studio 6.0 | Microsoft Visual Basic 6.0**. Setelah itu akan terlihat tampilan pilihan jenis *New Project*, pilih *Standart EX* maka akan terlihat tampilan *IDE (Integrated Development Environment)* Visual Basic 6.0.



Gambar 2.8 Tampilan IDE Visual Basic 6.0

1) Menu

Visual Basic mempunyai tiga belas menu dan masing-masing menu mempunyai fungsi yang berbeda.

2) Toolbar

Toolbar mempunyai fungsi yang sama dengan menu, hanya saja berupa icon-icon gambar dan digunakan sebagai jalan pintas.

3) Toolbox

Toolbox merupakan tempat kontrol-kontrol yang akan digunakan untuk membantu pembuatan program aplikasi.

4) Project Explorer

Project Explorer merupakan tempat yang digunakan untuk



melihat daftar *forms, modules, class modules, dan designers*.

#### 5) Properties Window

Properties Window berfungsi untuk mengatur properti dari setiap objek kontrol atau form. Pada Properties Window semua objek kontrol dapat diatur karakteristiknya.

#### 6) Form Layout Window

Form layout window berfungsi untuk melihat atau mengetahui posisi tampilan form saat program dijalankan.

#### 7) Form Objek

Form objek digunakan untuk menempatkan atau meletakkan objek dari kontrol-kontrol yang akan digunakan untuk merancang dan membuat program aplikasi.

#### 8) Form Kode

Form kode digunakan sebagai tempat untuk menulis kode-kode program aplikasi.

## **2. Bahasa Pemrograman pada Visual Basic 6.0**

### *2.1. Variabel*

Setiap melakukan pemrograman, akan selalu memerlukan tempat penyimpanan data, misalnya untuk menampung data hasil perhitungan, menampung data hasil pembacaan register, atau lainnya. Tempat penyimpanan data itu dinamakan Variabel yang merupakan pointer yang

menunjuk pada alamat memori fisik tertentu di komputer.

Dalam penggunaannya variabel harus mempunyai nama dan tipe data tertentu. Nama variabel menunjuk pada suatu tempat pada memori komputer, sedangkan tipe data mengontrol besarnya memori yang disediakan untuk variabel tersebut. Berikut ini adalah tipe data pada Visual Basic beserta ukuran byte dan range tipe data tersebut.

Tabel 2.2 Tipe data pada *Visual Basic*

<b>Tipe Data</b>	<b>Ukuran byte</b>	<b>Range</b>
Integer	2 byte	-32768 s.d. 32767
Long	4 byte	-2.147.483.648 s.d. 2.147.483.647
Single	4 byte	Negatif: -3.402823E38 s.d. -1.401298E-45 Positif: 1.401298E-45 s.d. 3.402823E38
Double	8 byte	Negatif: -1.79769313486232E308 s.d. -4.94065645841247E-324 Positif: 4.94065645841247E-324 s.d. 1.79769313486232E308
Currency	8 byte	-922337203685477.5808 s.d. 922337203685477.5807
String	1 byte per karakter	0 s.d. 2 milyar karakter
Byte	1 byte	0 s.d. 255
Boolean	2 byte	True (Benar) atau False (salah)
Date	8 byte	1 Januari 100 s.d. 31 Desember 9999

Tabel 2.2 Lanjutan Tipe data pada *Visual Basic*

Object	4 byte	Referensi objek
Variant	16 byte + 1 byte per karakter	Null, error, dan seluruh tipe data yang lain (Boolean, numeric, string, objek, array)

Pada visual basic terdapat dua cara untuk mendeklarasikan sebuah variabel, yaitu dengan cara deklarasi eksplisit dan cara deklarasi implisit. Deklarasi eksplisit menggunakan pernyataan “Dim” diikuti nama dan tipe datanya, sedangkan deklarasi implisit menggunakan simbol dibelakang nama variabel yang mempresentasikan tipe data yang digunakan. Berikut ini adalah contoh deklarasi eksplisit :Dim Text As String

Contoh deklarasi implisit :

Tabel 2.3. Deklarasi Implisit

Tipe data	Simbol karakter	Contoh pemakaian
Integer	%	Angka% = 100
Long Integer	&	Angka& = 2147483647
Single	!	Angka! = 2147483647000
Double	#	Konstanta_Pi# = 3.1415926535
Currency	@	saldo@ = 1000.50
String	\$	Nama\$ = “Awan”

Pada Visual Basic juga terdapat Konstanta yang merupakan variabel tetapi nilainya tetap. Dengan konstanta, kode program yang dibuat akan lebih

mudah dibaca dan mencegah penulisan yang salah pada kode program yang dibuat. Visual Basic telah menyediakan konstanta-konstanta siap pakai yang dalam penamaannya diawali dengan karakter “vb”, contoh vbRed yang merupakan konstanta untuk warna merah.

#### a) Kontrol Program

Dengan kontrol program, alur eksekusi program dapat dikendalikan serta dapat menentukan keputusan apa yang harus dikerjakan oleh program pada kondisi tertentu. Kontrol program pada Visual Basic meliputi kontrol pertimbangan kondisi dan keputusan, kontrol pengulangan serta kontrol penyaluran alternatif. Beberapa kontrol program pada Visual Basic yang digunakan pada pemrograman ini :

##### 1) If ... Then

Pernyataan ini mengetes suatu kondisi berdasarkan syarat kondisi kemudian menentukan suatu tindakan jika kondisi tersebut dipenuhi yang berupa pernyataan.

```
If <syarat kondisi> Then <pernyataan>
```

```
End If
```

##### 2) If ... Then ... Else

Pernyataan ini hampir sama dengan If ... Then ..., yaitu digunakan untuk mengetes suatu kondisi tertentu. Hanya saja, jika suatu kondisi tidak terpenuhi, maka alur program akan mengeksekusi pernyataan yang

lain kemudian menentukan suatu tindakan jika salah satu kondisi tersebut terpenuhi.

```
If <syarat kondisi 1> Then <pernyataan pertama>  
    ElseIf <syarat kondisi 2> Then <pernyataan kedua>  
    -  
    ElseIf <syarat kondisi n> Then <pernyataan ke-n>  
Else <pernyataan>  
  
End If
```

### 3) Select ... Case

Pada dasarnya perintah ini sama dengan perintah If ... Then ... Else, yaitu akan mengeksekusi satu blok pernyataan dari beberapa pilihan blok pernyataan. Hanya saja penulisannya lebih ringkas dan lebih mudah dimengerti.

```
Select Case <kondisi yang diuji>  
  
Case <syarat kondisi 1>  
    <blok pernyataan pertama>  
  
Case <syarat kondisi 2>  
    <blok pernyataan kedua>  
  
-  
  
Case Else  
    <blok pernyataan ke-n>  
  
End Select
```

### 4) Do ... Loop

Perintah Do ... Loop digunakan untuk perulangan suatu blok pernyataan

sampai dipenuhinya syarat kondisi yang ditetapkan.

Do

<blok pernyataan>

Loop Until <syarat kondisi>

5) For ... Next

Perintah ini sama dengan melakukan perulangan seperti perintah Do ... Loop, tetapi dengan For ... Next bisa ditentukan nilai awal dan nilai akhir perulangan serta nilai kenaikannya.

For <nama\_variabel> = <nilai awal> To <nilai akhir>

<blok pernyataan>

Next <nama\_variabel>

b) Prosedur

Pembuatan program akan lebih mudah dengan memecah program menjadi blok-blok komponen yang lebih kecil yang disebut Prosedur. Pengaturan ini sangat berguna ketika ada bagian program yang sering melakukan tugas yang sama berulang-ulang atau bermaksud membagikannya pada program yang lain.

1) Sub Procedure

Salah satu jenis prosedur yang ada didalam Visual Basic adalah Sub Procedure. Sub Procedure adalah blok kode yang dijalankan sebagai tanggapan atas terbentuknya even, baik even itu merupakan even

pemanggilan dari prosedur lain maupun even yang terjadi dari pemakaian program, misal even penekanan tombol kiri mouse.

[Private/Public] [Static] Sub

<nama\_prosedur> (argumen)

<blok pernyataan> End Sub

Setiap kali prosedur dipanggil, blok pernyataan yang ada di antara Sub dan End Sub akan dijalankan.

## 2) Operator

Operator digunakan pada Visual Basic untuk memanipulasi data maupun untuk melakukan perhitungan. Operator pada Visual Basic dapat dikelompokkan menjadi tiga kelompok yaitu operator matematik, operator perbandingan dan operator logika.

### a) Operator Matematik

Operator matematik digunakan untuk melakukan perhitungan matematik.

Tabel 2.4 Operator matematik

Operator	Operasi	Contoh pemakaian
^	Pemangkatan	Nilai% = 2^2 ‘menghasilkan 4
-	Tanda negative	Nilai% = -5 ‘menghasilkan negatif 5
*, /	Perkalian dan pembagian	Nilai% = (2*3)/6 ‘menghasilkan 1
\	Pembagian integer	
Mod	Modulus (sisa Pembagian)	Nilai% = 10 Mod 5 ‘menghasilkan 2

Tabel 2.4 Lanjutan Operator matematik

+ , -	Penambahan dan Pengurangan	Nilai% = 6 + 2 - 4 ‘menghasilkan 4
&	Penggabungan string	Teks\$ = “aw” & “an” ‘menghasilkan “awan”

b) Operator Perbandingan

Operator perbandingan digunakan untuk membandingkan suatu ekspresi dengan ekspresi yang lain dan akan menghasilkan nilai Boolean (*False* atau *True*).

Tabel 2.5 Operator perbandingan

Operator	Operasi	Contoh pemakaian
=	Sama dengan	Nilai = (1 + 2) = 3 ‘menghasilkan True
<>	Tidak sama dengan	Nilai = (1 + 2) <> 3 ‘menghasilkan False
<	Lebih kecil	Nilai = 2 < 3 ‘menghasilkan True
>	Lebih besar	Nilai = 2 > 3 ‘menghasilkan False
<=	Lebih Kecil atau sama dengan	Nilai = 2 <= 3 ‘menghasilkan True
>=	Lebih besar atau sama dengan	Nilai = 2 >= 3 ‘menghasilkan False
Like	Mempunyai ciri yang sama	Nilai = “abba” Like “a*” ‘True Nilai = “abba” Like “a??a” ‘True Nilai = “abba” Like “a?a” ‘False Nilai = “a” Like “[a-z]” ‘True



Tabel 2.5 Lanjutan Operator perbandingan

Is	Mempunyai referensi obyek yang sama	Nilai = Command1 Is Label1 'menghasilkan Y
----	-------------------------------------	--

c) Operator Logika

Operator logika biasanya digunakan untuk mengekspresikan satu atau lebih ekspresi logika yang akan menghasilkan nilai Boolean.

Tabel 2.6. Operator logika

Operator	Keterangan	Tabel Kebenaran Operasi	
		Operan	Hasil
Not	Akan menghasilkan nilai kebalikan dari nilai operan	Not True	False
		Not False	True
And	Akan menghasilkan true jika kedua operan-nya berlogika true	True And True	True
		True And False	False
		False And True	False
		False And False	False
Or	Akan menghasilkan True jika salah satu operan-nya berlogika True	True Or True	True
		True And False	True
		False And True	True
		False And False	False

Tabel 2.6. Lanjutan Operator logika

Xor	Akan menghasilkan True jika operan-nya berlogika berbeda	True Or True	False
		True And False	True
		False And True	True
		False And False	False
Eqv	Akan menghasilkan True jika operan-nya berlogika sama	True Or True	True
		True Or False	False
		False Or True	False
		False Or False	True

#### D. SMS GATEWAY

##### 1. SMS



SMS singkatan dari *Short Message Service* adalah merupakan suatu teknologi yang memungkinkan untuk mengirim dan menerima pesan antar pengguna mobile phone. SMS pertama hadir di Eropa pada tahun 1992. Yang standar awalnya diterapkan dengan menggunakan GSM. Seperti namanya Short Message Service pesan yang dapat dikirim dengan SMS sangat terbatas. Satu pesan SMS

dapat berisi paling banyak 140 bytes (1120 bit) dari data, maka satu pesan SMS dapat berisi sampai:

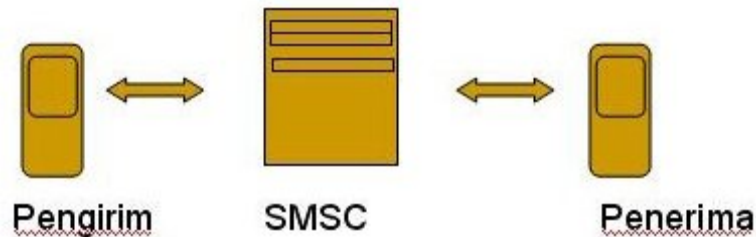
- a) 160 karakter, jika 7-bit character encoding digunakan (jika 7-bit character encoding adalah penggunaan karakter latin seperti karakter English).
- b) 70 characters, jika 16-bit Unicode UCS2 character encoding digunakan (pesan teks SMS berisi karakter bukan latin, seperti karakter Cina maka perlu menggunakan 16-bit character encoding).

### *1.1. SMS Center / SMSC*

SMS center/pusat (SMSC) bertugas untuk melakukan penanganan operasi SMS dari suatu jaringan wireless. Ketika suatu pesan SMS dikirim dari mobile phone, maka akan diterima oleh SMS pusat terlebih dahulu kemudian akan diteruskan ke nomor yang dituju.

Tugas pokok dari SMSC adalah untuk mengarahkan pesan SMS dan mengatur prosesnya. Jika penerima tidak tersedia (sebagai contoh ketika handphone dimatikan), SMSC akan menyimpan pesan SMS tersebut dan akan mengirimkan ke nomor tujuan apabila penerima sudah menghidupkan kembali handphonenya. SMSC bisa dikatakan sebagai gateway atau gerbang pusat untuk menghubungkan antara beberapa pengguna handphone. Pada umumnya suatu layanan jaringan mempunyai nomor pusat sendiri yang dapat digunkanan. Dan layanan nomor pusat ini dapat di atur dalam menu yang

tersedia pada handphone, yang secara default sudah diatur oleh operator jaringan kartu SIM itu sendiri.



### 1.2. Pengiriman dan Penerimaan SMS

Dalam proses pengiriman dan penerimaan SMS, data yang dikirim maupun diterima oleh stasiun bergerak atau handphone menggunakan salah satu dari 2 mode yaitu:

#### a. Mode Text

Mode ini adalah cara termudah untuk mengirim pesan. Pada mode teks pesan yang kita kirim tidak dilakukan konversi. Teks yang dikirim tetap dalam bentuk aslinya dengan panjang mencapai 160 (skema 7 bit) atau 140 karakter (skema 8 bit).

#### b. Mode *Protocol Data Unit* (Mode PDU)

Dalam PDU, pesan yang dikirim berupa informasi dalam bentuk data dengan beberapa *header-header* informasi. PDU tidak hanya berisi pesan teks saja, tetapi terdapat beberapa informasi lainnya seperti nomor pengirim, nomor SMS Center, dan sebagainya.

Data yang mengalir ke SMSC berbentuk PDU. PDU merupakan informasi yang terdiri dari header-header yang berisi bilangan heksadesimal. Bilangan heksadesimal adalah bilangan yang terdiri atas 1,2,3,4,5,6,7,8,9,A,B,C,D,E,F. Pada pengiriman SMS, PDU yang digunakan adalah PDU *Submit*. Sedangkan pada penerimaan SMS, PDU yang digunakan adalah PDU *Deliver*

### 1. Pengiriman SMS

PDU untuk mengirim SMS (PDU submit) terdiri atas sembilan header yaitu sebagai berikut:

#### a. SCA (*Service Center Address* )

Header pertama ini terbagi atas tiga subheader yaitu:

- Jumlah pasangan heksadesimal SMSC dalam bilangan heksa
- *National/International Code*
  1. Untuk nasional, kode subheadernya yaitu 81
  2. Untuk internatioanal, kode subheadernya yaitu 91
- Nomer SMSCnya sendiri, dalam pasangan heksa dibalik-balik. Jika tertinggal satu angka heksa yang tidak memiliki pasangan, angka tersebut akan dipasangkan dengan huruf F didepannya.

Tabel 2.7. Nomor SMSC Operator Seluler di Indonesia

Operator Seluler	SMS Center	Kode PDU
Telkomsel	6281100000	07912618010000F0
Satelindo	62816124	059126181652

Tabel 2.7.Lanjutan Nomor SMSC Operator Seluler di Indonesia

Excelcol	62818445009	07912618485400F9
Indosat-M3	62855000000	07912658050000F0

b. PDE *Type*

Untuk *Send* tipe SMS = 1. Jadi bilangan heksanya 01

c. MR (Message Reference)

Nomor referensi ini dibiarkan dulu 0, jadi bilangan heksanya adalah 00. Nanti akan diberikan sebuah nomor referensi otomatis oleh ponsel/alat SMS gateway.

d.DA (*Destination Address*)

Merupakan nomor ponsel penerima pesan. Sama seperti cara menulis PDU

Header untuk SMSC, header ini juga terbagi atas tiga bagian yaitu:

- Jumlah bilangan desimal nomor ponsel yang dituju dalam bilangan heksadesimal
- National/International Code  
Untuk nasional kode subheadernya 81 sedangkan untuk internasional kode subheadernya 91
- Nomor ponsel yang dituju dalam pasangan heksa dibalik-balik. Jika tertinggal satu angka heksa yang tidak memiliki pasangan maka angka tersebut dipasangkan dengan huruf F didepannya.

Contoh: Untuk nomor ponsel yang dituju =628129573337 dapat ditulis

dengan cara sebagai berikut

628129573337 diubah menjadi:

1. 0C : ada 12 angka

2. 91

3. 26-18-92-75-33-73

Digabung menjadi : 0C91261892753373

e. PID (*Protocol Identifier*) Bentuk SMS

Merupakan pesan dalam bentuk apa yang dikirimkan antara lain:

- 00 : dikirim sebagai SMS
- 01 : dikirim sebagai telex
- 02 : dikirim sebagai fax

Dalam hal ini untuk mengirim dalam bentuk SMS dipakai 00

f. DCS (*Data Coding Scheme*)

Menggunakan skema 7 bit dan ditandai dengan angka 00

g. VP (*Validity Period*)

Merupakan jangka waktu sebelum SMS expired. Agar SMS pasti terkirim ke ponsel penerima sebaiknya tidak diberi batasan waktu validnya. Bila pada batas waktu yang telah ditentukan pesan tidak juga sampai ke nomor tujuan maka pesan akan di gagalkan oleh SMSC.

h.UDL (*User D7ata Length*)

UDL menunjukkan panjang user data. Sebagai contoh untuk kata “hello” maka UDL-nya adalah 05

i. UD (*User Data*)

Header ini merupakan isi dari SMS yang dikirimkan. Isi SMS berupa pasangan bilangan heksa. Untuk ponsel/SMS *gateway* berskema *encoding* 7 bit jika kita mengetikkan suatu huruf dari keypad berarti kita telah membuat 7 angka I/O berurutan.

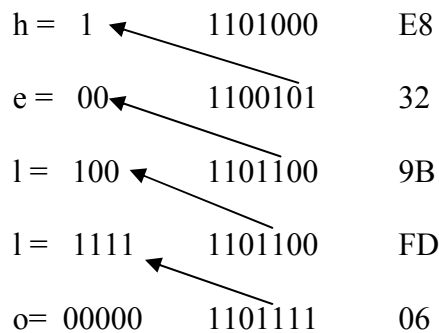
Ada dua langkah untuk mengkonversikan isi SMS yaitu:

- Mengubah karakter yang diketikkan menjadi kode 7 bit.
- Mengubah kode 7 bit menjadi 8 bit yang diwakili oleh pasangan heksa

Contoh : untuk kata “hello”

Langkah pertama:

Bit 7.....0



Oleh karena total 7bit x 5 huruf = 35 bit, sedangkan yang kita perlukan adalah 8 bit x 5 huruf = 40 bit, maka diperlukan 5 bit *dummy* yang diisi dengan



bilangan 0. Dengan demikian kata “hello” hasil konversinya menjadi E8329BFD06.

#### j. Menggabungkan Delapan Header

Masing-masing header maupun subheader untuk mengirim SMS di atas harus digabungkan menjadi sebuah PDU yang lengkap.

## **2. SMS Gateway**

Saat ini banyak sekali pengguna layanan pesan singkat atau SMS yang digunakan di beberapa lini bisnis, baik perusahaan besar, perusahaan kecil maupun perorangan. Dimana tujuannya pun beragam, seperti yang sering kita lihat bahkan menggunakannya. Sebagai contoh penggunaan yang sering kita lihat adalah dalam acara atau program televisi dimana semakin maraknya polling quiz yang menggunakan layanan SMS atau dapat kita lihat dalam penjualan pulsa elektronik. Dimana untuk mendukung proses tersebut diperlukan sebuah aplikasi penghubung yang dapat mengelola dan menangani pesan SMS yang masuk. Dapat dibayangkan apabila hal seperti ini ditangani secara manual tanpa adanya aplikasi pembantu untuk mengelolanya. Berdasarkan permasalahan diatas maka disini kita akan membahas bagaimana membuat aplikasi SMS gateway yang dapat membantu menangani pesan-pesan yang masuk dan mengelolanya sesuai dengan kebutuhan.

SMS Gateway adalah suatu platform yang menyediakan mekanisme untuk UEA menghantar dan menerima SMS dari peralatan mobile (HP, PDA phone, dll) melalui SMS Gateway s shortcode (sebagai contoh 9221). SMS Gateway membolehkan UEA untuk berkomunikasi dengan Telco SMSC (telkomsel, indosat, dll) atau SMS platform untuk menghantar dan menerima pesan SMS dengan sangat mudah, Karena SMS Gateway akan melakukan semua proses dan koneksi dengan Telco. SMS Gateway juga menyediakan UEA dengan interface yang mudah dan standar.

### *2.1. Keuntungan SMS Gateway*

SMS Gateway merupakan pintu gerbang bagi penyebaran Informasi dengan menggunakan SMS. Anda dapat menyebarkan pesan ke ratusan nomor secara otomatis dan cepat yang langsung terhubung dengan database nomor-nomor ponsel saja tanpa harus mengetik ratusan nomor dan pesan di ponsel anda karena semua nomor akan diambil secara otomatis dari database tersebut. Selain itu , dengan adanya SMS Gateway anda dapat mengustomisasi pesan-pesan yang ingin dikirim. Dengan menggunakan program tambahan yang dapat dibuat sendiri, pengirim pesan dapat lebih fleksibel dalam mengirim berita karena biasanya pesan yang ingin dikirim berbeda-beda untuk masing-masing penerimanya.

## *2.2. Pengiriman SMS dengan Komputer*

Secara umum ada beberapa cara untuk mengirimkan pesan SMS dari PC atau komputer.

- 1) Menghubungkan komputer dengan handphone atau GSM/GPRS modem. Selanjutnya tinggal menggunakannya perintah-perintah AT Command untuk mengirimkan pesan.
- 2) Menghubungkan komputer langsung dengan SMS Center (SMSC) atau SMS service provider. Untuk mengirimkan pesan SMS menggunakan beberapa protocol interface yang mendukung SMSC. Sebagai contoh protocol yang sering digunakan adalah HTTP atau HTTPS atau protocol lainnya.

Untuk mengirimkan pesan SMS dengan komputer dibutuhkan beberapa alat pendukung, seperti handphone, GSM atau GPRS modem dan sebuah kartu SIM. Ada beberapa cara untuk menghubungkan computer dengan handphone ataupun GSM modem yaitu dengan menggunakan Kabel Serial, kabel USB, Bluetooth atau dengan menggunakan Infrared. Ini semua tergantung dari fasilitas yang disediakan oleh handphone atau GSM modem tersebut. Setelah komputer terhubung dengan alat atau handphone/GSM modem, kita dapat mengendalikan handphone/GSM modem dengan cara mengirimkan perintah ke alat tersebut. Perintah-perintah yang dikirim ke handphone/GSM modem disebut dengan AT Command. Pada tabel

2.8 ada beberapa perintah dasar dari AT Command yang sering digunakan untuk menulis ataupun mengirim pesan SMS:

Tabel 2.8. AT Command untuk mengirim SMS

AT Command	Keterangan
+ CMGS	Send Message
+CMSS	Send Message From Storage
+CMGW	Write Message to Memory
+CMGD	Delete Message
+CMGC	Send Message
+CMMS	More Message to Send