# BAB II

# TINJAUAN PUSTAKA

Pada bab ini akan dijelaskan mengenai studi pustaka aplikasi brankas otomatis, teori-teori penunjang sistem brankas pengaman otomatis baik perangkat keras (*Hardware*), maupun perangkat lunak (*Software*), serta beberapa teori penunjang lainnya.

## 2.1 Studi Pustaka Aplikasi Brankas Otomatis

Perancangan dan pembuatan brankas otomatis pada umumnya memiliki beberapa perbedaan bila dilihat dari ukuran, badan/body, jenis pengunci, mekanisme pengunci, dan beberapa spesifikasi pendukung lainnya, namun pada dasarnya pembuatan brankas otomatis memiliki tujuan yang sama yaitu menciptakan suatu produk lemari penyimpanan yang handal dari segi fisik, serta memiliki kelebihan diantaranya dapat memberi layanan berupa pengamanan ekstra. Berdasarkan studi pustaka yang penulis lakukan, didapat beberapa jenis produk brankas otomatis yang telah dijual di pasaran sekarang ini. Berikut adalah uraian spesifikasi dan sistem kerja dari perangkat brankas otomatis tersebut.

1. Brankas Giant Safes dari Chubb Safes

   Brankas Giant adalah brankas yang dikembangkan oleh Chubbsafes (brankas produksi oleh PT. Chubb Safes Indonesia). Brankas Chubb Safes ini memberikan tingkat proteksi yang tinggi terhadap bahaya pembongkaran dan bahaya kebakaran dalam penyimpanan uang tunai dan barang-barang berharga.



Gambar 2.1 Brankas Giant Safes

Berikut adalah Spesifikasi Brankas Giant Safes.

- Pintu

  Tebal brankas keseluruhan 140 mm dengan lapisan formula Chubbsafes F42 setebal 75 mm, juga dilengkapi dengan pelat baja anti-bor di semua bagian-bagian penting.

- Badan/Body

  Konstruksi monolitik material pelindung dengan ketebalan 90 mm berupa formula konkret Miltonex N200 yang tahan terhadap karat dan pengeboran.

- Kunci

  Memakai sebuah kunci kombinasi S&G 6642 - 3 roda (satu juta kemungkinan) dan sebuah kunci Mauer President A 71113 - 8 lever (280.000 kemungkinan variasi anak kunci) sebagai pengaman standard.

- Mekanisme Penguncian

  Slot geser pada 3 sisi pintu berdiameter 32 mm dan slot bar pada sisi engsel, menjamin kesatuan antara pintu dan badan lemari guna memberikan perlindungan yang maksimal.

- Perlindungan Khusus

  Brankas ini dilengkapi dengan 2 pengunci otomatis (Automatic Relocking Device) dimana bila terjadi pembongkaran secara paksa atau peledakan, pintu akan terkunci secara otomatis.

- Sertifikasi

  Mendapatkan sertifikasi JIS S1037 - 1989 menurut standard industri Jepang untuk ketahanan terhadap kebakaran selama 60 menit.

- Warna

  RAL 9002 dengan tekstur (cat dasar Epoxy, lapisan akhir Polyurethane).

- Fitting

  Lemari besi ini dilengkapi dengan peralatan standar berupa rak baja pada bagian dalam yang penempatannya dapat disesuaikan dengan kebutuhan pemakai. Laci bagian dalam juga terbuat dari baja dan masing-masing dilengkapi oleh sebuah kunci dengan dua anak kunci.

  (Sumber : http://indolokbaktiutama.blogspot.com, 2011)

2. Brankas Ichiban HSX40

Brankas jenis Ichiban HSX40 adalah jenis brankas terbaru yang diproduksi oleh Ichiban dengan model desain yang cocok untuk setiap rumah modern dan juga untuk penggunaan di kantor. Diproduksi dengan standarisasi rekayasa keandalan tinggi Jepang menggunakan LSI kunci digital khusus yang memberi pengamanan lebih baik.



Gambar 2.2 Brankas Ichiban HSX40

Berikut diuraikan tentang spesifikasi dan fitur pada brankas Ichiban HSX40.

Tabel 2.1 Spesifikasi Brankas Ichiban HSX40

| Sisi luar | | | Sisi dalam | | | Berat bersih (Kgs/Lbs) | Kapasitas (Liter) | Standar (JIS-TS) |
|---|---|---|---|---|---|---|---|---|
| Lebar (mm) | Tebal (mm) | Tinggi (mm) | Lebar (mm) | Tebal (mm) | Tinggi (mm) | | | |
| 484 | 433 | 372 | 354 | 267 | 214 | 55/110 | 20 | 2 Jam |

Dibawah ini adalah penjelasan tentang fitur tambahan yang ada pada brankas Ichiban HSX40.

- Kunci

  Memiliki dua macam kunci yaitu kunci elektronik digital dan kunci manual,

- Mekanisme Penguncian

  Menggunakan 2-8 digit angka kombinasi rahasia yang dapat diubah oleh pengguna sesuai keinginan,

- Suplai Tegangan

  Suplai tegangan rendah sebesar 6 VDC menggunakan baterai dan ditempatkan diluar untuk keperluan pemeliharaan yang mudah dan aman.

  (Sumber : http://www.infoalatkantor.com, 2012)

## 2.2 Perangkat Keras (*Hardware*)

Perangkat keras brankas pengaman otomatis yang akan diuraikan dibawah ini adalah perangkat-perangkat fisik yang terdiri dari komponen-komponen elektronik penunjang sistem kerja brankas dimana dari setiap-komponen-komponen yang ada, akan dirangkai dan terintegrasi menjadi satu sistem kontrol perangkat keras brankas pengama otomatis berbasis mikrokontroler AT89S52.

### 2.2.1 Mikrokontroler AT89S52

Mikrokontroler adalah mikroprosessor yang dirancang khusus untuk aplikasi kontrol, dan dilengkapi dengan ROM, RAM dan fasilitas I/O pada satu chip. AT89S52 adalah salah satu anggota dari keluarga MCS-51/52 yang dilengkapi

dengan inte rnal 8 Kbyte *Flash* PEROM (*Programmable and Erasable Read Only Memory* ), yang memungkinkan memori program untuk dapat diprogram kembali. AT89S52 dirancang oleh Atmel sesuai dengan instruksi standar dan susunan pin 80C5.

Mikrokontroler AT89S52 memiliki spesifikasi diantaranya :

- sebuah *Central Processing Unit* (CPU) 8 Bit,
- 256 *byte Random Acces Memory* (RAM) internal,
- empat buah port I/O, yang masing masing terdiri dari 8 bit,
- osilator internal dan rangkaian pewaktu,
- dua buah *timer/counter* 16 bit,
- lima buah jalur interupsi ( 2 buah interupsi eksternal dan 3 interupsi internal),
- sebuah port serial dengan *full duplex-Universal Asynchronous Receiver Transmitter* (UART),
- mampu melaksanakan proses perkalian, pembagian, dan Boolean,
- 8 Kbyte EPROM untuk memori program,
- kecepatan maksimum pelaksanaan instruksi per siklus adalah 0,5 μs pada frekuensi *clock* 24MHz. Apabila frekuensi *clock* mikrokontroler yang digunakan adalah 12 MHz, maka kecepatan pelaksanaan instruksi adalah 1 μs.

#### 2.2.1.1 *Central Processing Unit* (CPU)

Bagian ini berfungsi mengendalikan seluruh operasi pada mikrokontroler. Unit ini terbagi atas dua bagian, yaitu unit pengendali atau *Control Unit* (CU ) dan unit aritmatika dan logika atau *Aritmetic logic Unit* (ALU). Fungsi utama unit

pengendali adalah mengambil instruksi dari memori (*fetch*) kemudian menterjemahkan susunan instruksi tersebut menjadi kumpulan proses kerja sederhana (*decode*), dan melaksanakan urutan instruksi sesuai dengan langkah-langkah yang telah ditentukan program (*execute*). Unit aritmatika dan logika merupakan bagian yang berurusan dengan operasi aritmatika seperti penjumlahan, pengurangan, serta manipulasi data secara logika seperti operasi AND, OR, dan perbandingan.

#### 2.2.1.2 Bagian Masukan/Keluaran (I/O)

Bagian ini berfungsi sebagai alat komunikasi serpih tunggal dengan piranti di luar sistem. Sesuai dengan namanya, perangkat I/O dapat menerima maupun memberi data dari atau ke serpih tunggal.

Ada dua macam piranti I/O yang digunakan, yaitu piranti untuk hubungan serial *Universal Asynchronous Receiver Transmitter* (UART) dan piranti untuk hubungan paralel yang disebut dengan *Paralel Input Output* (PIO). Kedua jenis I/O tersebut telah tersedia di dalam serpih tunggal AT89S52.

#### 2.2.1.3 Perangkat Lunak

Serpih tunggal keluarga MCS-51 memiliki bahasa pemrograman khusus yang tidak dipahami oleh jenis serpih tunggal yang lain. Bahasa pemrograman ini dikenal dengan nama bahasa *assembler* yang memiliki 256 perangkat instruksi. Namun saat ini pemrograman mikrokontroler dapat dilakukan dengan menggunakan bahasa C. Dengan bahasa C, pemrograman mikrokontroler menjadi lebih mudah, hal ini karena dengan format bahasa C akan secara otomatis diubah menjadi bahasa *assembler* dengan format file hexa.

Perangkat lunak pada mikrokontroler dapat dibagi menjadi lima kelompok sebagai berikut :

1. instruksi transfer data,

   instruksi ini berfungsi memindahkan data, yaitu antar register, dari memori ke memori, dari register ke memori dan lain lain.

2. instruksi aritmatika,

   instruksi ini melaksanakan operasi aritmatika yang meliputi penjuml ahan, pengurangan, penambahan satu (*increment*), pengurangan satu (*decrement*), perkalian dan pembagian.

3. instruksi logika dan manipulasi bit,

   berfungsi melaksanakan operasi logika AND, OR, XOR, perbandingan, penggeseran dan komplemen data.

4. instruksi percabangan,

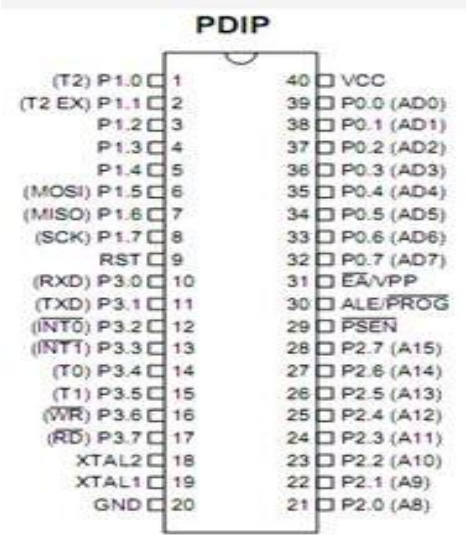
   berfungsi untuk mengubah urutan normal pelaksanaan suatu program. Dengan instruksi ini, program yang sedang dilaksanakan akan meloncat ke suatu alamat tertentu.

5. instruksi stack, I/O, dan kontrol,

   instruksi ini mengatur penggunaan stack, membaca/menulis port I/O, serta pengontrolan.

#### 2.2.1.4 Konfigurasi Pin

Mikrokontroler AT89S52 mempunyai 40 pin dengan catu daya tunggal 5 Volt. Ke-40 pin tersebut digambarkan sebagai berikut :




Gambar 2.3 Konfigurasi PIN Mikrokontroler AT89S52

Berikut adalah fungsi dari masing-masing pin AT89S52.

1. Pin 1 sampai 8 (Port 1) merupakan port 8 bit dua arah (*bidirectional*) yang dapat digunakan untuk berbagai keperluan (*general purpose*),
2. Pin 9 merupakan pin reset, reset aktif jika mendapat catuan tinggi,
3. Pin 10 sampai 17 (Port 3) adalah port 8 bit dua arah yang memiliki fungsi pengganti sebagai berikut :
    - P3.0 (10) : RXD (port serial penerima data),
    - P3.1 (11) : TXD (port serial pengirim data),
    - P3.2 (12) : INT0 (input interupsi eksternal 0, aktif *low*),
    - P3.3 (13) : INT1 (input interupsi ekstrernal 1, a ktif *low*),
    - P3.4 (14) : T0 (eksternal input *timer / counter* 0),
    - P3.5 (15) : T1 (eksternal input *timer / counter* 1),

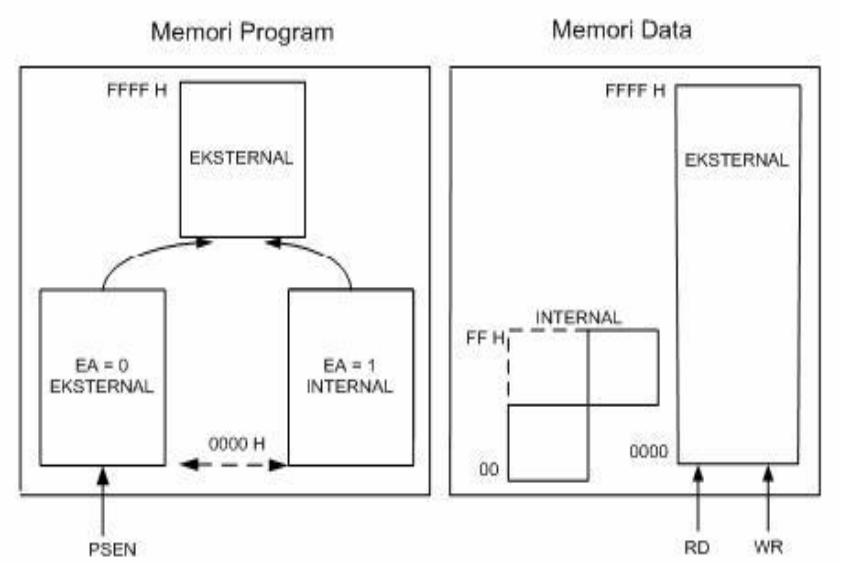
- P3.6 (16) : WR (*Write*, aktif *low*) sinyal kontrol penulisan data dari port 0 ke memori data dan input-output eksternal,
- P3.7 (17) : RD (*Read*, aktif *low*) sinyal kontrol pembacaan memori data input – output eksternal ke port 0.

4. Pin 18 sebagai XTAL 2, keluaran osilator yang terhubung pada kristal,
5. Pin 19 sebagai XTAL 1, masukan ke osilator berpenguatan tinggi, terhubung pada kristal,
6. Pin 20 sebagai Vss, terhubung ke ground pada rangkaian,
7. Pin 21 sampai 28 (Port 2) adalah port 8 bit dua arah. Port ini mengirim *byte* alamat bila pengaksesan dilakukan pada memori eksternal,
8. Pin 29 sebagai *Program Store Enable* (PSEN) adalah sinyal yang digunakan untuk membaca, memindahkan program memori eksternal (ROM/EPROM) ke mikrokontroler (aktif *low*),
9. Pin 30 sebagai *Address Latch Enable* (ALE) untuk menahan alamat bawah selama mengakses memori eksternal. Pin ini juga berfungsi sebagai PROG (aktif *low*) yang diaktifkan saat memprogram internal *flash* memori pada mikrokontroler (*on chip*),
10. Pin 31 sebagai *External Accesss* (EA) untuk memilih memori yang akan digunakan, memori program internal (EA = Vcc) atau memori program eksterna l (EA = Vss), juga berfungsi sebagai Vpp (*programming supply voltage*) pada saat memprogram internal *flash* memori pada mikrokontroler,

11. Pin 32 sampai 39 (Port 0) merupakan port 8 bit dua arah. Berfungsi sebagai alamat bawah yang dimultipleks dengan data untuk mengakses program dan data memori eksternal,
12. Pin 40 sebagai Vcc, terhubung ke +5 V sebagai catuan untuk mikrokontroler.

### 2.2.1.5 Organisasi Memori

Semua serpih tunggal dalam keluarga MCS-51 memiliki pembagian ruang alamat untuk program dan data. Pemisahan memori program dan memori data memperbolehkan memori data untuk diakses oleh alamat 8 bit. Sekalipun demikian, alamat data memori 16 bit dapat dihasilkan melalui register *Data Point Register* (DPTR). Memori program hanya bisa dibaca tidak bisa ditulis karena disimpan dalam EPROM. Dalam hal ini EPROM yang tersedia di dalam serpih tunggal AT89S52 sebesar 8 Kbyte .




Gambar 2.4 Organisasi Memori AT89S52

a. Memori Program

Pada EPROM 8 Kbyte, jika *External Access* (EA) bernilai tinggi, maka program akan menempati alamat 0000 H sampai 0FFF H secara internal. Jika EA bernilai rendah maka program akan menempati alamat 1000 H sampai FFFF H ke program eksternal.

b. Memori Data

Memori data internal dipetakan seperti pada gambar di bawah ini Ruang memorinya dibagi menjadi tiga blok yaitu bagian 128 bawah, 128 atas, dan ruang *Special Function Register* (SFR).



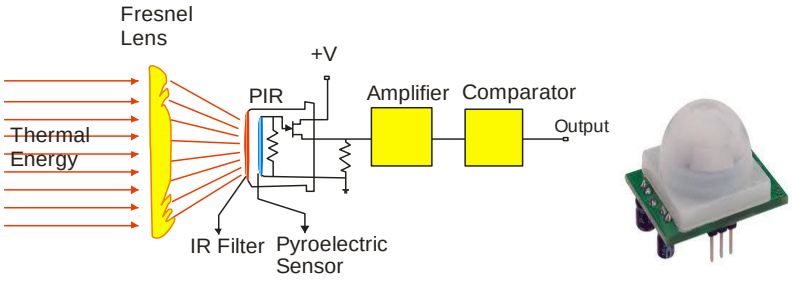

Gambar 2.5 Pemetaan Memori Data

Bagian RAM 128 *byte* bawah dipetakan menjadi 32 *byte* bawah dikelompokkan menjadi 4 bank dan 8 register (R0 sampai R7). Pada bagian 16 *byte* berikutnya, di atas bank-bank register, membentuk suatu blok ruang memori yang bisa teralamati per bit (*bit addressable*). Alamat-alamat bit ini adalah 00H hingga 7FH. Semua *byte* yang berada di dalam 128 bawah dapat diakses baik secara langsung maupun tidak langsung. Bagian 128 atas hanya dapat diakses dengan

pengalamatan tidak langsung. Bagian 128 atas dari RAM hanya ada di dalam piranti yang memiliki RAM 256 *byte*.

### 2.2.2 *Passive Infrared Receiver* (PIR)

*Passive Infrared Receiver* (PIR) merupakan sebuah sensor berbasiskan infrared. Akan tetapi, tidak seperti sensor infrared kebanyakan yang terdiri dari IR LED dan fototransistor. PIR tidak memancarkan apapun seperti IR LED. Sesuai dengan namanya *Passive*, sensor ini hanya merespon energi dari pancaran sinar inframerah pasif yang dimiliki oleh setiap objek yang terdeteksi olehnya. Objek yang bisa dideteksi oleh sensor ini biasanya adalah tubuh manusia. Bagian-bagian dari PIR dengan perannya masing-masing, yaitu *Fresnel Lens*, *IR Filter*, *Pyroelectric sensor*, *amplifier*, dan *comparator*.



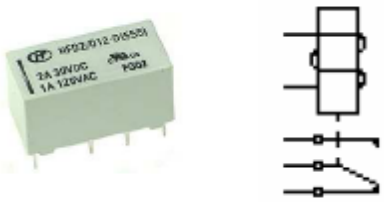

Gambar 2.6 Blok Diagram dan Bentuk Fisik *Passive InfraRed* (PIR)

Sensor PIR ini bekerja dengan menangkap energi panas yang dihasilkan dari pancaran sinar inframerah pasif yang dimiliki setiap benda dengan suhu benda di atas nol mutlak. Seperti tubuh manusia yang memiliki suhu tubuh kira-kira 32 derajat celcius, yang merupakan suhu panas yang khas yang terdapat pada lingkungan.

Pancaran sinar inframerah inilah yang kemudian ditangkap oleh *Pyroelectric* sensor yang merupakan inti dari sensor PIR ini sehingga menyebabkan *Pyroelectric* sensor yang terdiri dari *galium nitrida*, *caesium nitrat* dan *litium tantalate* menghasilkan arus listrik. Arus listrik bisa dihasilkan karena pancaran sinar inframerah pasif ini membawa energi panas. Prosesnya hampir sama seperti arus listrik yang terbentuk ketika sinar matahari mengenai *solar cell*.

### 2.2.3 *Relay*

*Relay* adalah sebuah saklar elektromagnetik yang prinsip kerjanya menggunakan azas kumparan listrik. *Relay* bekerja dengan prinsip apabila sebuah kumparan yang berintikan sebuah lempengan besi lunak dialiri aliran listrik, maka lempengan besi lunak tersebut akan menjadi magnet. Magnet tersebut menarik atau menolak pegas kontak sebuah alat penghubung dan akibatnya akan terjadi kontak dan lepas kontak dari alat penghubung tersebut. Prinsip seperti ini dapat dimanfaatkan sebagai dasar pembuatan saklar otomatis yang banyak dipergunakan dalam bidang elektronika. Bentuk kontak *relay* dan simbol *relay* dapat dilihat pada Gambar 2.7.



Gambar 2.7 Simbol dan Jenis *Relay*

### 2.2.4 *Electronic dazer*

*Electronic dazer* atau sering dikenal dengan nama listrik kejut adalah merupakan suatu aplikasi rangkaian elektronika yang banyak dipakai pada perangkat-perangkat elektronika untuk keperluan manusia. Aplikasi rangkaian *electronic dazer* pada perangkat elektronik yang umum dipakai oleh manusia adalah raket nyamuk yang dipakai untuk mengusir dan membunuh nyamuk. Melalui sengatan listrik yang berasal dari rangkaian *electronic dazer* yang dipasang pada bahan konduktor berupa besi pada bagian jaring raket, nyamuk yang terkena jaring raket saat raket diayunkan oleh manusia akan langsung mati akibat tersengat arus listrik yang cukup kuat. Hal ini bukan saja berbahaya bagi nyamuk, namun sengatan listriknya yang cukup kuat mampu mencederai manusia bila lalai dalam penggunaan.

Selain contoh aplikasi raket nyamuk, terdapat juga jenis *electronic dazer* yang dipakai sebagai alat pelindung dan memberi efek jera bagi pelaku tindak kriminal yang berpotensi membahayakan pengguna alat pelindung tersebut. Nama yang paling umum dari alat ini adalah stun gun. Dengan besar tegangan yang dihasilkan sampai ratusan bahkan  ribuan volt, alat ini sering digunakan oleh manusia untuk melindungi diri dan memberi efek jera bagi pelaku-pelaku tindak kriminal seperti perampokan, pencurian, dan beberapa tindak kriminal lainnya yang tidak diinginkan terjadi.



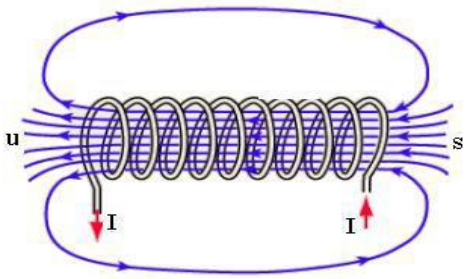
Gambar 2.8 Contoh Aplikasi Rangkaian *Electronic Dazer*

Berdasarkan aplikasi yang sudah banyak dipakai inilah, penulis menjadikannya sebagai acuan untuk menambahkan aplikasi pengaman pada brankas menggunakan rangkaian *electronic dazer* yang diharapkan dapat memberi efek jera bagi pelaku pencurian dan perampokan terhadap barang-barang berharga yang disimpan didalam brankas yang penulis buat.

### 2.2.5 *Solenoid*

*Solenoid* adalah peralatan yang dipakai untuk mengkonversi sinyal elektrik atau arus listrik menjadi gerak mekanik. *Solenoid* dibuat dari kumparan dan inti besi yang dapat digerakkan yang berfungsi sebagai aktuator untuk membuka kunci otomatis pada brankas.

Secara skematik bentuk dari *solenoid* dapat dilihat pada Gambar 2.6, dimana selonoid terdiri dari *n* buah lilitan kawat berarus listrik I, medan magnet yang dihasilkan memiliki arah seperti pada gambar dibawah, dimana kutub utara magnet mengikuti aturan tangan kanan.




Gambar 2.9 *Solenoid* Dengan Banyak Lilitan *n*

Kuat medan magnet pada selonoid dengan jumlah lilitan persatuan panjang *n* adalah dapat dihitung dengan menggunakan persamaan dibawah ini.

B = μ0 . *n* . I …… (2.1)

Dimana : $n = \frac{N}{l}$

Keterangan : B = Medan Magnet

μ0 = Konstanta Permeabilitas Udara

*n* = Jumlah Lilitan Persatuan Panjang

N = Jumlah Lilitan

*l* = Panjang Lilitan

I = Arus

Prinsip kerja dari *Solenoid* berdasarkan pada penghantar yang membawa arus kedalam kumparan sehingga kumparan akan menimbulkan medan magnet. Medan magnet ini dibuat sedemikian rupa sehingga keadaannya selalu tolak menolak antara medan magnet. Gambar 2.10 dibawah menunjukkan bentuk fisik *solenoid*.



Gambar 2.10 Bentuk Fisik *Solenoid*

### 2.2.6 Saklar Magnet

Saklar magnet merupakan suatu jenis saklar yang bekerja dengan cara mendetekasi keberadaan magnet. Terdiri atas dua bagian yang mana salah satu bagian

merupakan pendeteksi magnet dan berisi mekanik saklar, sedangkan bagian yang lain berfungsi sebagai magnet yang akan dideteksi.

Prinsip kerja dari saklar magnet ini sama dengan saklar jenis *push button switch*. Jika terdeteksi magnet, maka saklar akan berada dalam posisi tertutup (tombol saklar ON) dan jika tidak terdeteksi magnet maka saklar berada dalam posisi terbuka (tombol saklar OFF).



Gambar 2.11 Jenis Saklar Magnet

### 2.2.7 Modem GSM Serial wavecom 1306b

Modem GSM Serial Wavecom 1306b adalah salah satu jenis modem yang dirancang sedemikian rupa sehingga dapat dihubungkan dengan perangkat luar dengan koneksi serial. Modem ini memiliki kelebihan dalam pengiriman data ke mikrokontroler, tidak lagi menggunakan format *Protocol Data Unit* (PDU) yang rumit namun hanya menggunakan format pengiriman data serial biasa. Untuk komunikasi serial, digunakan baudrate 115200 sebagai pengaturan standar dari modem ini.




Gambar 2.12 Modem GSM Serial Wavecom 1306b

Secara umum modem GSM Serial Wavecom 1306b ini memiliki banyak fungsi diantaranya sebagai *SMS broadcast application*, *SMS Quiz application*, *SMS auto-reply, aplikasi server pulsa,* dan beberapa kegunaan lain, namun pada sistem brankas pengaman otomatis ini, modem GSM Wavecom ini digunakan sebagai penghubung perangkat kontrol brankas dengan *Handphone* (HP) *owner* untuk memberikan informasi melalui fasilitas *Short Message Service* (SMS). Agar dapat digunakan untuk mengirim dan menerima pesan ke *Handphone* (HP) pemilik/*owner*, modem GSM Serial Wavecom 1306b akan dilengkapi dengan GSM SIM Card yang operatornya akan disesuaikan dengan operator GSM yang digunakan pemilik/*owner*. Hal ini dilakukan agar lebih menghemat pengeluaran pulsa, serta pengiriman pesan bisa lebih lancar.

### 2.2.8 *Liquid Crystal Display* (LCD) *Character* 16x2

LCD adalah sebuah display dot matrix yang difungsikan untuk menampilkan tulisan berupa angka atau huruf sesuai dengan yang diinginkan (sesuai dengan program yang digunakan untuk mengontrolnya). LCD *character* memiliki banyak

jenis dilihat dari jumlah bitnya. LCD *character* yang digunakan pada tugas akhir ini adalah LCD dot matrix dengan karakter 16x2, dan memiliki kaki/pin berjumlah 16.

LCD sebagaimana output yang dapat menampilkan tulisan sehingga lebih mudah dimengerti, dibanding jika menggunakan LED saja. LCD *character* digunakan untuk menampilkan tulisan atau karakter saja.



Gambar 2.13 Modul LCD *Character* 16x2

Tampilan LCD terdiri dari dua bagian, yakni bagian panel LCD yang terdiri dari banyak “titik”. LCD dan sebuah mikrokontroler yang menempel dipanel dan berfungsi mengatur ‘titik-titik’ LCD tadi menjadi huruf atau angka yang terbaca.

Huruf atau angka yang akan ditampilkan dikirim ke LCD dalam bentuk kode ASCII. Kode ASCII ini diterima dan diolah oleh mikrokontroler di dalam LCD menjadi ‘titik-titik’ LCD yang terbaca sebagai huruf atau angka.

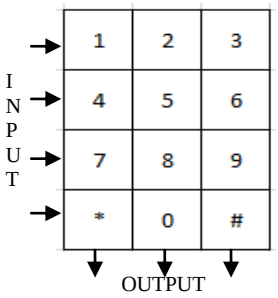
### 2.2.9 *Keypad* 3x4

*Keypad* adalah rangkaian tombol yang berfungsi untuk memberi sinyal pada suatu rangkaian dengan menghubungkan jalur-jalur tertentu. *Keypad* terdiri dari beberapa macam berdasarkan jumlah tombol dan fungsinya. Pada sistem pengontrolan dalam tugas akhir ini, digunakan *keypad* matriks 3x4 (12 saklar) dengan pin penghubung rangkaian berjumlah 7 buah.




Gambar 2.14 Tampilan Fisik *Keypad* 3x4

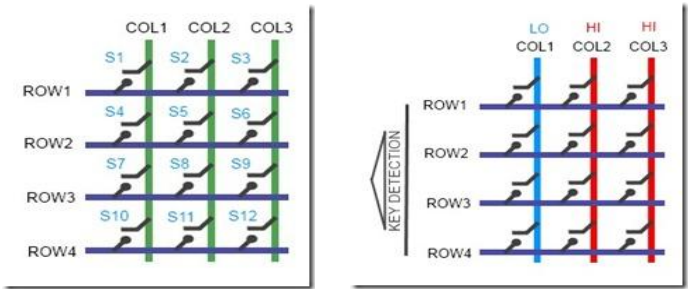
Ketujuh pin penghubung ini terbagi dua kelompok yaitu 4 buah pin sebagai input dan 3 buah lainnya sebagai ouput. Adapun maksud dari 7 pin I/O adalah untuk dijadikan kombinasi penghubungan pada rangkaian yang akan disambungkan dengan *keypad* ini. Dimana dalam setiap penekanan satu tombol/saklar *keypad* maka terjadi kombinasi antara dua buah pin dalam pembacaan sinyalnya.




Gambar 2.15 Skematik *Keypad* 3x4

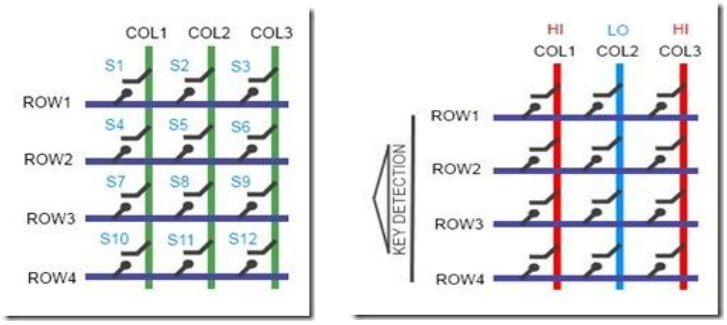
Ada beberapa teknik untuk membaca data dari matrix *keypad* tersebut, salah satunya adalah dengan teknik *scanning*, dimana baris atau kolom selalu dipindai untuk mendeteksi tombol yang ditekan. Caranya yaitu dengan memberikan status ‘0’ (*low*) pada pin colom secara bergantian, lalu pin row dideteksi apakah ada salah satunya yang berkondisi ‘0’ (*low*).

Untuk contoh penggunaan matrix 3x4, pertama C-1 (COL1) diberi logika ‘0’, kemudian status row dibaca apakah statusnya ‘1’ (*high*) semua atau ada salah satu yang *low*.




Gambar 2.16 Contoh Aplikasi *Scanning* Pada *Keypad* 3x4

Jika R-1 (ROW1) yang berlogika ‘0’ (*low*) berarti tombol S1 yang di tekan, jika R-2 (ROW2) yang berlogika ‘0’ (*low*), berarti tombol S4 yang ditekan, jika berlogika ‘1’ (*high*) semua, berarti tidak ada tombol yang ditekan dalam kolom ini, scanning dilanjutkan ke kolom berikutnya.




Gambar 2.17 Contoh Aplikasi Scanning Pada *Keypad* 3x4

Selanjutnya C-2 (COL2) yang diberi logika ‘0’ (*low*). Jika R-1 (ROW1) berlogika ‘0’ (*low*) berarti tombol S2 yang di tekan, namun jika R-2 yang berlogika ‘0’ (*low*),

berarti tombol S5 yang ditekan, jika berlogika ‘1’ (*high*) semua, berarti tidak ada tombol yang ditekan dalam kolom ini. Demikian selanjutnya untuk C3 (COL3).

Tabel 2.2 Status *Scanning Keypad* 3x4

| C-1 | C-2 | C-3 | R-1 | R-2 | R-3 | R-4 | CHAR | BIN |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 1 | 1 | 1 | 1 | none | 1111 |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0001 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 | 2 | 0010 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 3 | 0011 |
| 0 | 1 | 1 | 1 | 0 | 1 | 1 | 4 | 0100 |
| 1 | 0 | 1 | 1 | 0 | 1 | 1 | 5 | 0101 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 | 6 | 0110 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 | 7 | 0111 |
| 1 | 0 | 1 | 1 | 1 | 0 | 1 | 8 | 1000 |
| 1 | 1 | 0 | 1 | 1 | 0 | 1 | 9 | 1001 |
| 0 | 1 | 1 | 1 | 1 | 1 | 0 | * | 1010 |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0000 |
| 1 | 1 | 0 | 1 | 1 | 1 | 0 | # | 1011 |

### 2.2.10 Komunikasi Serial Port

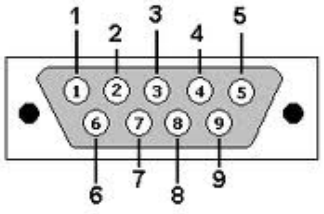
Komunikasi serial merupakan hal yang penting dalam sistem *embedded*, karena dengan komunikasi serial mikrokontroler dapat dengan mudah dihubungkan dengan alat lainnya. Port serial pada mikrokontroler terdiri atas dua pin yaitu RXD dan TXD, RXD berfungsi untuk menerima data dari komputer/perangkat lainnya, TXD berfungsi untuk mengirim data ke komputer /perangkat lainnya. Standar komunikasi serial untuk komputer ialah RS-232. RS-232 mempunyai standar tegangan yang berbeda dengan serial port mikrokontroler, sehingga untuk menghubungkan port serial mikrokontroler yang mempunyai level tegangan TTL (logika 1 dinyatakan sebagai 5 V dan logika 0 dinyatakan sebagai 0 V) dengan port

serial standar RS-232 maka dibutuhkan suatu rangkaian level converter yang berfungsi untuk mengubah level logika dari TTL ke RS-232 dan sebaliknya. Jenis integrated circuit (IC) yang digunakan bermacam-macam, tetapi yang paling mudah dan sering digunakan ialah IC MAX232.

Hardware pada komunikasi serial port dibagi menjadi 2 (dua) kelompok yaitu *Data Communication Equipment* (DCE) dan *Data Terminal Equipment* (DTE). Contoh dari DCE ialah modem, plotter, scanner dan lain lain sedangkan contoh dari DTE ialah terminal di komputer. Spesifikasi elektronik dari serial port merujuk pada *Electronic Industry Association* (EIA) :

1. "space" (logika 0) ialah tegangan antara + 3 hingga +25 V,
2. "mark" (logika 1) ialah tegangan antara –3 hingga –25 V,
3. daerah antara + 3V hingga –3V tidak didefinisikan /tidak terpakai,
4. tegangan *open circuit* tidak boleh melebihi 25 V,
5. arus hubungan singkat tidak boleh melebihi 500mA.

Konektor port serial yang paling umum dipakai adalah 9 pin (DB9) yang berpasangan (jantan dan betina).




Gambar 2.18 Serial Port DB9

Setiap kaki dari serial port DB9 ini mempunyai fungsi masing-masing yang dijelaskan dalam tabel di bawah ini.

Tabel 2.3 Keterangan Sinyal dan Kaki Serial Port

| Nama Sinyal | Arah Sinyal | Nomor Kaki Konektor |
|---|---|---|
| Data Carrier Detect (DCD) | Dari DCE | 1 |
| Received Data (RD) | Dari DCE | 2 |
| Transmitted Data (TD) | Dari DTE | 3 |
| DTE Ready (DTR) | Dari DTE | 4 |
| Ground | - | 5 |
| DCE Ready (DSR) | Dari DCE | 6 |
| Request to Send (RTS) | Dari DTE | 7 |
| Clear to Send (CTS) | Dari DCE | 8 |
| Ring Indicator (RI) | Dari DCE | 9 |

Berikut adalah keterangan mengenai fungsi saluran pada konektor DB9.

1. *Data Carrier Detect*, dengan saluran ini DCE memberitahukan ke DTE bahwa pada terminal masukkan ada data masuk,
2. *Receive Data*, digunakan DTE menerima data dari DCE,
3. *Transmitted Data*, digunakan DTE mengirimkan data ke DCE,
4. *Data Terminal Ready*, pada saluran ini DTE memberitahukan kesiapan terminalnya,
5. *Signal Ground*, saluran ground,
6. *DCE Ready*, sinyal aktif pada saluran ini menunjukkan bahwa DCE sudah siap,
7. *Request To Send*, dengan saluran ini DCE diminta mengirim data oleh DTE,
8. *Clear To Send*, dengan saluran ini DCE memberitahukan bahwa DTE boleh mulai mengirim data,

9. *Ring Indicator*, pada saluran ini DCE memberitahukan ke DTE bahwa sebuah stasiun menghendaki berhubungan dengannya.

## 2.3 Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan untuk mengontrol kinerja perangkat keras dari aplikasi brankas pengaman otomatis ini terbagi atas tiga bagian yaitu program *assembler*, AT-Command dan *Short Message Service* (SMS). Berikut adalah uraian tentang masing-masing perangkat lunak tersebut.

### 2.3.1 Program *Assembler*

Program sumber *assembly* merupakan program yang ditulis oleh pembuat program berupa kumpulan baris-baris perintah dan biasanya disimpan dengan extension .ASM. Program sumber *assembly* terdiri atas beberapa bagian yaitu *Label*, *Mnemonik Operand*, dan *Komentar*.

a. *Label*

Label sangat berguna dalam pemberian nama pada alamat-alamat yang dituju, karena pemberian label pada suatu alamat lebih bersifat relatif. Selain itu, label juga digunakan sebagai catatan diri alur program. Untuk membuat label, ada beberapa persyaratan yang harus dipenuhi, dimana persyaratan ini kadang-kadang juga bergantung pada program *assembler* yang digunakan, yaitu :

- harus diawali dengan huruf,
- tidak diperbolehkan adanya label yang sama dalam satu program *assembly*,

- maksimal 16 karakter,
- tidak diperbolehkan adanya karakter spasi dalam label.

b. *Mnemonic*

Mnemonic atau bisa juga disebut kode operasi adalah kode-kode yang akan dikerjakan oleh program *assembler* yang ada pada mikrokontroler merupakan perintah-perintah atau instruksi-instruksi yang sangat bergantung dengan jenis mikrokontroler yang digunakan. Contoh, untuk keluarga MCS51 digunakan MOVX, MOV, ADD dan lain-lain.

c. *Operand*

*Operand* merupakan pelengkap dari mnemonic, jumlah *operand* yang dibutuhkan oleh satu mnemonic tidak selalu sama, sebuah mnemonic dapat memiliki tiga, dua, satu atau bahkan tidak memiliki *operand* sama sekali.

d. Komentar

Bagian komentar tidak mutlak ada dalam sebuah program, namun bagian ini sangat berguna untuk menjelaskan proses-proses kerja ataupun catatan-catatan tertentu pada bagian-bagian program. Bahkan pembuat program seringkali membutuhkannya untuk mengingat kembali jalannya program rancanganya.

#### 2.3.1.1 Sistem Pengalamatan

Dalam sebuah program, terdapat beberapa sistem pengalamatan yang perlu diketahui, yaitu :

a. pengalamatan langsung

pengalamatan langsung terbagi atas dua bagian, yaitu :

1. *immediate* data

proses pengalamatan ini terjadi pada sebuah perintah ketika nilai *operand* merupan data yang akan diproses. Biasanya *operand* tersebut selalau diawali dengan tanda '#'. *Operand* yang digunakan pada immediate data juga dapat berupa bilangan bertanda mulai -256 hingga +256

contoh :

*Mov A,#-1 sama dengan Mov A,#0FFH*

bilangan 1 adalah sama dengan 0 dikurangi 1, dalam bentuk heksa bilangan 00H jika dikurangi 1, hasilnya adalah 0FFH. Dengan pengertian seperti ini, bilangan -1 dianggap sama dengan 0FFH.

2. pengalamatan data

proses pengalamatan ini terjadi pada sebuah proses perintah ketika nilai *operand* merupakan alamat dari data yang akan di isi, dipindahkan atau diproses.

b. pengalamatan tak langsung

proses pengalamatan ini terjadi pada sebuah perintah ketika salah satu *operand* merupakan *register* berisikan alamat dari data yang akan di isi atau dipindahkan. Pengalamatan jenis ini biasa digunakan untuk

melakukan penulisan, pemindahan, atau pembacaan beberapa data dalam lokasi memori yang mempunyai urutan beraturan.

c. pengalamatan kode

pengalamatan kode merupakan pengalamatan kerja *operand*, merupakan alamat dari instruksi jump dan call. Biasanya *operand* tersebut akan menuju ke suatu alamat yang telah diberi label sebelumnya.

d. pengalamatan bit

pengalamatan bit adalah pengalamatan ketika *operand* menunjuk ke alamat pada RAM internal ataupun *register* fungsi khusus yang mempunyai kemampuan pengalamatan secara bit.

berdasarkan penulisannya, pengalamatan ini terdiri dari beberapa macam yaitu :

1. langsung menuju ke alamat bit

   contoh : *Setb 0B0H*

   perintah ini memberikan logika 1 pada bit di alamat B0H dengan pengalamatan secara bit.

2. menggunakan operator titik

   contoh : *Setb P3.0*

   perintah ini memberikan logika 1 pada bit ke 0 dari port 3, bit tersebut terletak di alamat B0H dengan pengalamatan secara bit.

3. menggunakan lambang *assembler* secara standar

   contoh : *Setb RXD*

perintah ini memberikan logika 1 pada kaki RXD yang terletak pada bit ke 0 dari port 3.

4. menggunakan lambang *assembler* secara bebas

contoh : *Penerima Bit P30*

*Setb Penerima*

perintah ini memberikan logika 1 pada bit penerima yang sebelumnya didefinisikan sebagai bit P3.

Pengalamatan kode merupakan pengalamatan kerja *operand*, merupakan alamat dari instruksi jump dan call. Biasanya *operand* tersebut akan menuju ke suatu alamat yang telah diberi label sebelumnya.

### 2.3.2 *AT Command*

*AT Command* merupakan satu set perintah yang dapat digunakan untuk mengoperasikan GSM *mobile phone* melalui serial *interface* (kabel data). Menurut formatnya perintah-perintah yang diberikan harus selalu diawali dengan ’AT’ dan diakhiri dengan ’<CR>’ yang merupakan *Carriage Return*.

*AT Command* dapat digunakan antara lain untuk menginstruksikan perintah-perintah sebagai berikut :

1. mengirim dan menerima SMS,
2. mendapatkan informasi mengenai alat, misalnya nama manufaktur, nomor IMEI dan lain-lain,

3. mendapatkan status alat, misalnya status aktivitas, status registrasi *network*, kekuatan sinyal dan status baterai,
4. penulisan dan pencarian *phonebook*,
5. mengembalikan dan menyimpan konfigurasi.

Tidak semua alat mengimplementasikan *AT Command*, pada umumnya modem GSM lebih mendukung banyak *AT Command*. *AT Command* umumnya ditulis dengan huruf besar, tetapi banyak jenis modem GSM dan ponsel yang mengizinkan penulisan *AT Command* dalam huruf besar maupun huruf kecil.

Terdapat dua tipe *AT Command*, yaitu *basic command* dan *extended command*. *Basic command* adalah *AT Command* yang tidak menggunakan tanda ’+’, misalnya ATH, ATD dan ATA. Sebaliknya *AT Command* yang menggunakan tanda ’+’ merupakan *extended command*. Setiap *extended command* memiliki parameter atau perintah tes yaitu =?, yang akan mengembalikan informasi mengenai perintah tersebut.

Beberapa perintah *AT Command* yang digunakan untuk keperluan SMS (pengiriman, penerimaan) adalah sebagai berikut :

1. AT+CMGS

   perintah *AT Command* ini digunakan untuk mengirimkan SMS. Format yang digunakan adalah “AT+CMGS = <*length*> <CR> <PDU *is given*>”. Apabila pengiriman sukses dilakukan, format respon yang diterima adalah “+CMGS : <mr>”, dengan “<mr>” adalah *message reference* dari

SMSC. Sedangkan jika pengiriman gagal dilakukan, respon yang diterima adalah "+ *CMS error*".

2. AT+CMGR

   perintah ini digunakan untuk membaca sebuah SMS pada indeks tertentu. Format yang digunakan adalah "AT+CMGR = <*index*>". Apabila perintah ini berhasil diesekusi, format respon yang diterima adalah "+CMGR: <*stat*>,,<*length*><CR><LF><pdu>". "<*stat*>".

3. AT+CMGD

   perintah ini digunakan menghapus sebuah SMS pada *memory* SMS. Format yang digunakan adalah "AT=CMGD=<*index*>", respon yang diterima adalah "*OK/ERROR/*+CMS *ERROR"*

4. AT+CMGL

   perintah ini digunakan untuk membaca daftar SMS sesuai parameter tertentu. Format yang digunakan adalah "AT+CMGL [=<*stat*>]".

   Respon yang diterima adalah :

   "+CMGL: *<index>, <stat>, < oa/da>, [< alpha>],*

   *[<scts>][,<tooa/toda>, < length>]<CR> < LF> < da ta> [< CR> < LF>+CMGL:*

   *<index>, <stat>, <da/oa>, [<alpha>], [<scts>],[<tooa/toda>, <length>] <CR> <LF> <data>[.. j] OK"*

   atau "*+CMS ERROR : <err>*". "*[<alpha>]*", adalah deretan alfanumerik yang merepresentasikan nomor pengirim atau penerima.

Yang terpenting adalah mengetahui perintah *AT Command* untuk proses terima, kirim, dan delete SMS. Kemudian perintah tersebut dimasukkan ke dalam *coding* program yang dibuat.

Tabel 2.4 Daftar Perintah Dalam *AT Command*

| **Jenis *AT Command*** | ***AT Command*** | **Fungsi** |
|---|---|---|
| Untuk Kontrol | ATD | Melakukan dial |
| | ATH | Hang up panggilan masuk |
| | ATA | Menjawab panggilan |
| Bersifat Umum | AT+CGMI | Mengambil informasi manufaktur |
| | AT+CGMM | Mengambil informasi model |
| | AT+CGMR | Mengambil informasi revisi |
| | AT+CGSN | Mengambil informasi serial number alat |
| | AT+CSCS | Memilih set karakter |
| Untuk Layanan Jaringan | AT+CNUM | Nomor subscriber |
| | AT+CREG | Registrasi jaringan |
| | AT+COPS | Pemilihan operator |
| | AT+CLCK | Fasilitas lock |
| | AT+CPWD | Penggantian password |
| | AT+CCWA | Call waiting |
| Untuk Layanan Jaringan | AT+CPMS | Menentukan penyimpanan pesan |
| | AT+CMGF | Format pesan |
| | AT+CSCA | Nomor service provider |
| | AT+CNMI | Pengaktifan indikasi pesan baru |
| | AT+CMGL | Daftar pesan |
| | AT+CMGR | Membaca pesan |
| | AT+CMGS | Mengirim pesan |
| | AT+CMGD | Menghapus pesan |
| | AT+CMMS | Mengirimkan lebih banyak pesan |
| | +CMTI | Notifikasi delivery pesan |
| | +CDSI | Notifikasi status repot |

### 2.3.3 *Short Message Service* (SMS)

SMS merupakan salah satu layanan *data service* GSM yang menyediakan fasilitas untuk menyampaikan pesan singkat antara *Mobile Station* (MS) melalui *Service Centre* (SC) atau yang disebut dengan *Short Message Service Centre*

(SMSC). SMSC berfungsi menyampaikan pesan antara MS dan *Short Message Entity* (SME).

SMS adalah pesan yang terbatas besarnya (pesan singkat) yang dapat dikirim atau diterima oleh sebuah MS yang berupa data dalam bentuk *string* atau teks. Teks tersebut dapat terdiri dari kombinasi kata-kata, nomor-nomor atau penggabungan huruf dan angka, yang mempunyai panjang maksimum 160 karakter dengan menggunakan huruf latin.

Dalam mengirim dan menerima data sms, digunakan standar (European Telecommunication Standards Institute) ETSI-GSM 03.38, dimana data yang tertulis dilayar *handphone* berupa huruf atau angka yang kemudian diterjemahkan menjadi data 7-*bit* (*septet*). Data 7 bit tersebut menurut standar ETSI-GSM 03.38 diolah dulu menjadi data 8 bit (*octet*), baru dikirimkan, dimana besarnya data 7 bit maksimum adalah 160 karakter yang jika diolah menjadi data *octet* akan menjadi 140 *octet*.