

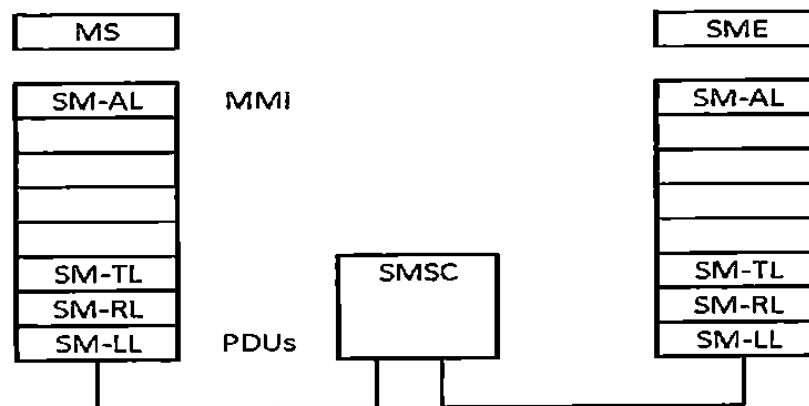
## BAB II

### DASAR TEORI

Dalam bab ini akan dibahas mengenai dasar teori dan prinsip kerja dari peralatan yang akan dipakai dalam perancangan *Sistem Update Running Text (Tulisan Berjalan) Berbasis SMS Dengan Mikrokontroler Atmega8535*.

#### 2.1 Teknologi SMS (Short Message Service)

SMS (short message service) merupakan salah satu layanan yang banyak diaplikasikan pada sistem komunikasi tanpa kabel, SMS memberikan fasilitas bagi pengguna untuk mengirimkan pesan sampai 169 karakter *alfanumerik* atau pesan teks yang dikembangkan dan distandarisasi oleh ETSI (*European Telecommunication Standarts Institute*) sebagai bagian dari pengembangan GSM phase 2 yang terdapat dalam dokumentasi GSM 03.04 dan GSM 03.38. Banyak sekali yang menggunakan bearer SMS antara lain: internet email, chat, remote. Dalam proses pengiriman dan penerimaan SMS, data yang dikirimkan maupun yang diterima menggunakan mode teks atau mode PDU (*Protocol Data Unit*).

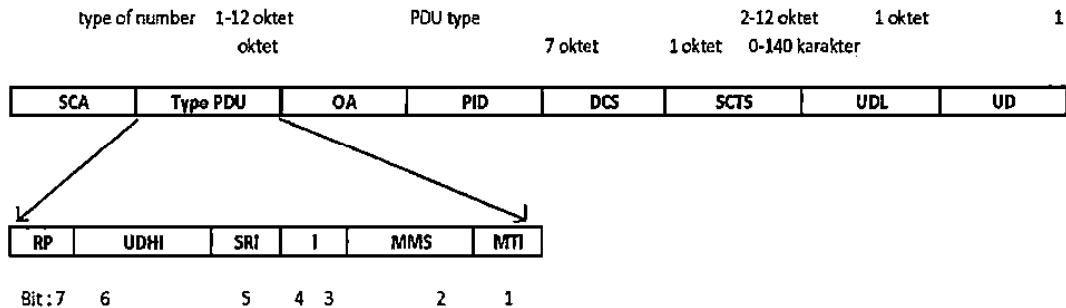


Bagan diatas menunjukkan susunan protokol SMS beserta piranti pendukung mulai dari MS (*Mobile Station*), SMSC (*Short Message Service Center*), dan SME (*Short Message Entity*). Sebelum menggunakan SMS kita terlebih dahulu harus mendeklarasikan nomor SMSC pada MS dengan menganggap bahwa *Handphone* kita mendukung SMS-MO (*Short Message Service-Mobile Originated*) dan hampir semua *Handphone* saat ini mendukung fitur ini. Proses diawali dari MMI (*Man Machine Interface*) yang terdapat dalam *Handphone*. MMI berhubungan langsung dengan pengguna dimana tiap perintah yang dieksekusi harus berdasarkan *AT+Cellular* atau yang lebih dikenal sebagai *AT Command Set* dan dapat dianggap sebagai suatu terminal. Kemudian perintah diteruskan ke SM-TL (*Short Message Transport Layer*). Yang menyediakan layanan untuk SM-AL (*Short Message Application*) dimana disini kita bebas memilih teks yang akan ditulis dan dikirim. Lapisan SM-TL memicu SM-AL untuk mengirim dan menerima SMS ke dan dari *peer entity*-nya, dan juga menerima laporan (*report*) mengenai permintaan pengiriman SMS yang pertama kan ditransfer. SM-TL berkomunikasi dengan peer entity dalam bentuk PDU (*Protokol Data Unit*) yaitu:

1. *SMS Deliver*, berisi SMS dari SMSC ke MS
2. *SMS Deliver Report*, berisi penyebab kesalahan (jika diperlukan)
3. *SMS Submit*, berisi SMS dari MS ke SMSC
4. *SMS Submit Report*, berisi status report dari SMSC keMS

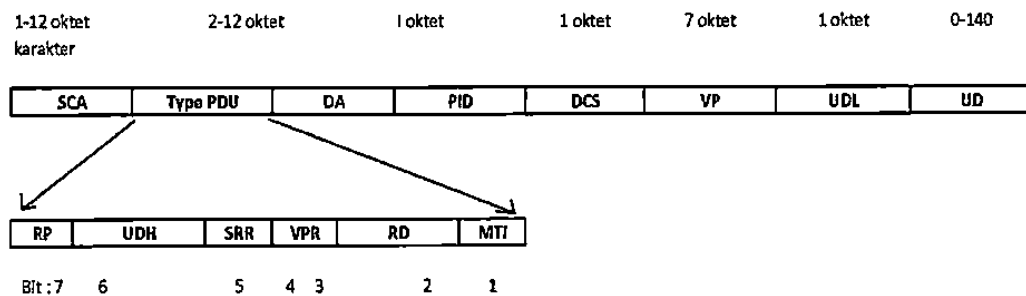
Dalam hal ini hanya akan dibahas dua bentuk yang umum dan paling banyak dipakai yaitu *PDU SMS Deliver* dan *PDU Submit*.

### 1. SMS Deliver



Gambar 2.2. SMS Deliver (Mobile Transmitted)

### 2. SMS Submit



Gambar 2.3. SMS SUBMIT (Mobile Originated)

## 2.2 AT Command sebagai bahasa Modem Handphone

Salah satu software yang digunakan untuk mencoba AT Command adalah Windows Hyperterminal, Hyperterminal tersedia dalam windows installer. AT Command hampir sama dengan perintah >(prompt) pada DOS.

### 2.2.1 AT Command untuk komunikasi dengan SMS Center.

adalah AT Command yang bertugas mengirim atau menerima data ke dan dari SMS center. AT Command tiap-tiap SMS bisa berbeda. Beberapa AT Command yang penting untuk SMS yaitu sebagai berikut:

a. AT + CPMS

Perintah ini digunakan untuk menyimpan pesan SMS yang disukai, seperti:

AT +CPMS=<mem1>

<mem1>

ME : Pesan akan disimpan pada memori *Handphone*

SM : Pesan akan disimpan pada *Simcard*

MT: Pesan akan disimpan pada penyimpanan yang berhubungan dengan *Handphone*

b. AT +CMGR

Perintah ini digunakan untuk membaca dalam sebuah SMS, seperti:

AT +CMGR=<index>

<index>

Urutan daftar pesan SMS dalam memori yang dipilih <mem1>

c. AT+CMGD

Perintah ini digunakan untuk membaca sebuah SMS dari memori

<index>

Urutan daftar pesan SMS dalam memori yang dipilih <mem1>

d. AT +CMGS

Perintah ini digunakan untuk mengirim pesan sebuah SMS, seperti:

AT +CMGS=<length>

<length>

Panjang format PDU

Disamping untuk SMS, sebenarnya masih ada banyak penggunaan *AT Command* yang lain.

2.2.2 PDU sebagai bahasa SMS dan bagian-bagiannya.

Data yang mengalir ke atau dari SMS center harus berbentuk PDU (*Protokol Data Unit*). PDU berisi bilangan-bilangan heksadesimal yang mencerminkan bahasa I/O. PDU terdiri dari beberapa header, Header untuk kirim SMS ke SMS Center berbeda dengan SMS yang diterima dari SMS Center. Maksud dari bilangan Heksadesimal adalah bilangan yang terdiri atas 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Sebagai contoh, untuk angka desimal 1000, bilangan heksadesimalnya adalah 3E8. Cara mengkonversinya :

1000	:	16 = 62 sisa 8 dalam heksadesimal 8
62	:	16 = 3 sisa 14 dalam heksadesimal adalah E
3	:	16 = 0 sisa 3 dalam heksadesimal adalah 3

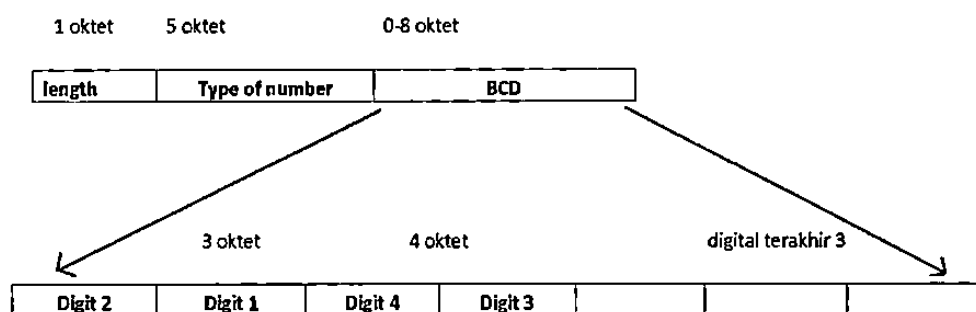
Jadi desimal 1000 adalah heksadesimal 8E3

### 2.2.3 PDU untuk terima SMS ke SMS Center

PDU (*Protokol Data Unit*) untuk mengirim SMS terdiri dari atas delapan header, sebagai berikut:

#### a. *Service Center Address Information Elemen (SCA info element)*

SCA (*Service Center Address*), berisi informasi dari nomor SMSC (*SMS Center*). Informasi yang ada dalam bagian ini adalah panjang nomor SMSC.



Gambar 2.4. Bagan Elemen SCA

#### ***Length:***

Nomor SMS Center, oktet pertama ini menunjukkan jumlah pasangan oktet dari seluruh Service Center, misalnya : 07 (berarti ada 7 pasang oktet dibelakang oktet pertama)

#### ***Type of number:***

National atau International Kode, pasangan oktet kedua ini menunjukkan jenis nomor *Handphone*.

**BCD digital:**

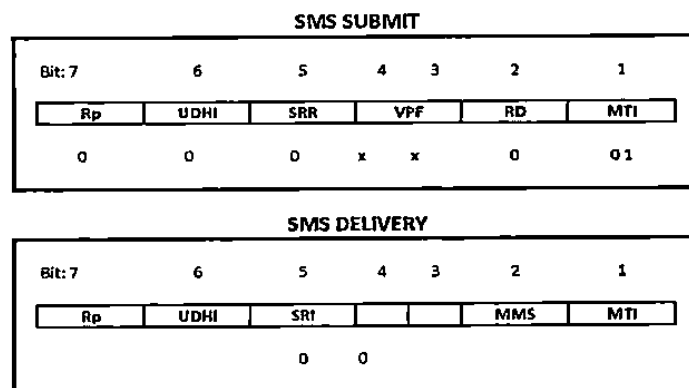
Berapa pasang oktet terakhir ini menunjukkan nomor dari SMSC yang dipakai oleh *Handphone*. Dalam pasangan heksadesimal dibalik-balik seperti:

06912618010000 diubah menjadi:

- 1). 06 : 6 pasang (length)
- 2). 91 : 1 pasang (type of number)
- 3). 628110000 : 5 pasang (BCD digital)

Digabung menjadi : 06912618010000

b. Protokol Data Unit Type (PDU)



Gambar 2.5. Bagan Elemen dari SMS Submit dan SMS Deliver

1. RP (*Relay Path*) :

RP (*Reply Path*) bit yang menandakan sms dapat di balas atau tidak,

... dan ... RP tidak dapat jika bit ini bernilai 0, begitupun sebaliknya

## 2. UDHI dan UDH:

UDHI (*User Data Header Indicator*), bit ini merupakan parameter untuk mengindikasikan bagian UD (*User Data*) terdapat header, UDHI dan UDH memiliki format yang sama yaitu bila bit ini bernilai 0 maka UD hanya berisi satu pesan saja, jika nilainya 1 maka awal UD berisi dengan header sebagai tambahan pesan.

## 3. SRR dan SRI:

SRI, (*Status Report Indication*), bit ini merupakan parameter indikator jika SME (*Short Message Entity*) telah meminta status *report*. SRR dan SRI hanya dibedakan pada peer entity-nya, jika bit ini bernilai 0 maka status report tidak akan diminta (MS) atau dikirim balik (SMSC) dan begitupula sebaliknya.

## 4. VPF:

Oktet ini mengindikasikan periode validasi SMS Submit.

Tabel 2.1. Validasi SMS Submit

Bit 4	Bit 4	Keterangan
0	0	Tidak ada Validasi
0	1	Reserverd
1	0	Periode Validasi dalam <i>format Integer</i>
1	1	Periode Validasi dalam <i>format semi oktet</i>

## 5. RD:

Bit ini menandakan penolakan atau penerimaan SMS oleh SMSC jika



pula, jika 1 berarti ditolak dan jika 0 berarti diterima.

#### 6. MMS:

MMS (*More Message to Send*), bit ini merupakan parameter yang mengindikasikan ada tidaknya pesan sms yang akan dikirim lagi oleh SMSC. Bit ini menandakan ada atau tidaknya pesan lagi yang ditunggu oleh SMSC, jika ada bernilai 0 dan begitupula sebaliknya.

#### 7. MTI:

MTI (*Message Type Indicator*), parameter yang digunakan untuk mengidentifikasi tipe pesan kirim atau terima. Dua bit ini berisi type pesan yang terkirim atau diterima, jika 00 maka tipe pesan adalah SMS header (SMS diterima) dan jika 01 maka tipenya adalah SMS Submit (SMS dikirim) seperti, PDU 04 diubah menjadi bilangan desimal:

RP	UDHI	SRI	MMS	MTI
0	0	0	0	0
0	0	0	1	0
0	0	0	0	0

Gambar 2.6. Elemen dari MTI

MTI : bit 1 bit type SMS

0 0 *SMS Deliver* (SMSC → MS)

0 0 *SMS Deliver Report*

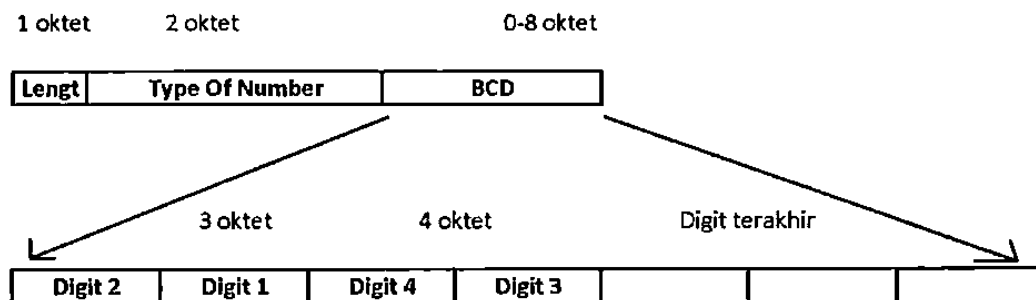
0 1 *SMS Submit*, berisi SMS dari MS ke SMSC

0 1 *SMS Submit Report*

1 0 *SMS Status Report* berisi status report dari SMSC ke MS

c. Original Address (OA) dan Destination address (DA)

OA (*Originator Address*), bagian ini memuat nomor pengirim. Runtun digit pada OA ini berupa digit BCD.



Gambar 2.7. Elemen dari OA dan DA

Original Address (OA) dan Destination Address (DA) memiliki format yang sama dengan elemen yang sama dengan SCA.

Misal :

OA dan SMS Deliver 0D91261823287889F1

8. 0D : *length*

9. 91 : *type of number*

10.6281328287981F : *BCD digital*

d. Protokol Identifier (PID)

PID (*Protocol Identifier*). PID berguna untuk melakukan hubungan dengan layer protokol yang lebih tinggi (*Application Layer*). Oktet ini mengidentifikasi proses tertentu yang dilakukan oleh telematik *handphone*, misalnya:

Bentuk SMS, antara lain:

0 → 00 → dikirim sebagai SMS

1 → 01 → dikirim sebagai Telex

2 → 02 → dikirim sebagai Fax

e. Data Coding Scheme (DCS)

DCS (*Data Coding Scheme*), bagian ini memberikan tanda tentang pengkodean yang dilakukan pada bagian UD (*User Data*), menggunakan 7-bit alfabet sehingga jumlah karakter sms maksimal 160 karakter atau menggunakan 8-bit heksa sehingga jumlah karakter sms maksimal menjadi 140 karakter.

Ada 2 (dua) skema, yaitu:

a) Skema 7 bit yaitu ditandai dengan angka 0 → 00

b) Skema 8 bit yaitu ditandai dengan angka lebih besar dari 0 → diubah ke heksadesimal.

Kebanyakan *handphone* atau SMS Gateway yang ada di pasaran sekarang menggunakan skema 7 bit sehingga kita menggunakan kode 00.

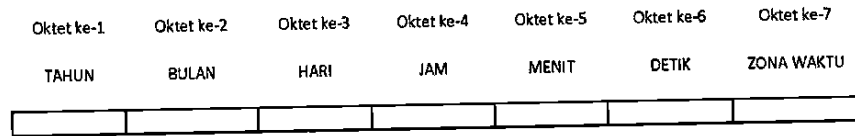
Bit : 7          6          5          4          3          2          1          0



Gambar 2.8. Elemen dari DCS

f. Service Center Time Stamp (SCTS)

SCTS (*Service Center Time Stamp*) bagian ini menerangkan waktu



Gambar 2.9. Elemen dari SCTS

Tujuan padang oktet SCTS menginformasikan tanggal dan waktu tiba diterimanya SMS pada transport layer SMSC. Pembacaan dari SMS ini dibalik-balik. Yaitu:

SCTS dan SMS delivey 50502231000580

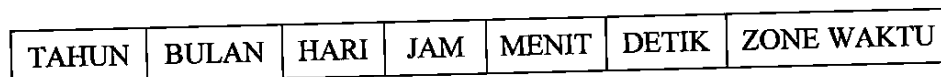
1. 50 → 05 : Menunjuk tahun
2. 50 → 05 : Menunjuk bulan
3. 22 → 22 : Menunjuk tanggal
4. 31 → 13 : Menunjuk Jam
5. 00 → 00 : Menunjuk Menit
6. 05 → 50 : Menunjuk detik
7. 80 → 08 : Menunjuk Zone waktu

Dimiliki oleh 12 bilangan heksadesimal (6 pasangan) yang berarti :

yy/mm/dd hh:mm:ss

50502231000580 → 05/05/22 13:00:50:08 → 22 Mei 2005 13:00:50 WIB

g. Validity Periode (VP)



Gambar 2.10. Elemen dari VPF

VPF mengindikasikan jangka waktu sebelum SMS Submit yang akan dikirim. Jika waktu telah ditentukan SMS tidak dibaca maka validitas waktunya akan lewat sehingga SMS tidak dapat dibaca lagi oleh penerima Jangka Waktu Sebelum SMS. Jika bagian ini di-skip, itu berarti kita tidak membatasi waktu berlakunya SMS. Sedangkan jika kita isi dengan suatu bilangan integer yang kemudian diubah ke pasangan heksadesimal tertentu, bilangan yang kita berikan tersebut akan mewakili jumlah waktu validitas SMS tersebut.

Rumus untuk menghitung jangka waktu validitas SMS adalah sebagai berikut:

Tabel 2.2 Validity Periode (VP)

Integer (INT)	Jangka Waktu Validitas SMS
0-143	$(INT + 1) \times 5$ menit (berarti: 5 menit s/d 12 jam)
144-167	12 jam + $((INT - 143) \times 30)$ menit
168-196	$(INT - 166) \times 1$ hari
197-255	$(INT - 192) \times 1$ minggu

Agar SMS kita pasti terkirim sampai terkirim sampai ke *handphone* penerima, kita tidak memberikan batasan waktu validnya.

h. User Data Length (UDL) dan User Data (UD)

1 oktet

1-140 karakter

UDL	UD
-----	----

UDL berisi tentang informasi banyaknya pasangan oktet pada UD, sedangkan UD berisi pesan atau SMS yang ditulis oleh pengirim berdasarkan format DCS.

Header ini terdiri dari 2 (dua) subheader yaitu:

- a) Panjang isi (jumlah huruf dari isi)

Seperti : untuk kata "hello" ada 5 huruf → 05

- b) Isi berupa pasangan bilangan heksadesimal.

Untuk *handphone* atau SMS Gateway berskema encoding 7 bit, jika kita mengetik suatu huruf dari keypad nya, berarti kita telah membuat 7 angka I/O berurutan.

Skema 7 (tujuh) Bit tersebut diperhatikan pada Tabel 2.4.

Ada 2 (dua) langkah yang harus kita lakukan untuk mengkonversikan isi SMS, yaitu:

- a) Langkah pertama : mengubah menjadi kode 7 bit.

Seperti kata "hello"

**Tabel 2.3 Kode 7 bit**

Bit	7	6	5	4	3	2	1
H	1	1	0	1	0	0	0
E	1	1	0	0	1	0	1
L	1	1	0	1	1	0	0
L	1	1	0	1	1	0	0
O	1	1	0	1	1	1	1

- b) Langkah kedua : mengubah kode 7 bit menjadi 8 bit, yang diwakili oleh pasangan heksadesimal.

H = 100 1000    48    1100 1000 → E8

L = 110 1100    6C    1001 1011 → 9B

L = 110 1100    6C    1111 1101 → FD

O = 110 1111    6F    0000 1101 → 06

Setiap 8 bit memiliki suatu pasangan heksadesimal. Tiap 4 bit mewakili suatu angka heksadesimal, tentu saja karena secara logika 2 pangkat 4 = 16 dengan demikian kata "hello" hasil konversinya





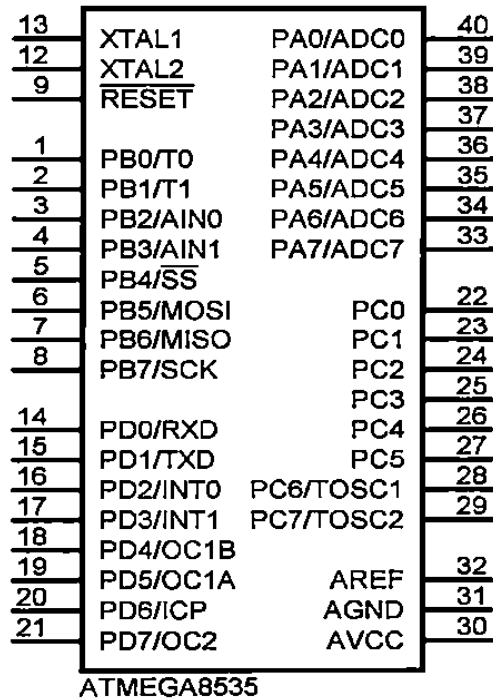
### 2.3 Mikrokontroler AVR ATmega 8535

Mikrokontroler AVR merupakan mikrokontroler berbasis arsitektur *RISC (Reduced Instruction Set Computing)* 8 bit. Berbeda dengan mikrokontroler keluarga 8051 yang mempunyai arsitektur *CISC (Complex Instruction Set Computing)*, AVR menjalankan sebuah instruksi tunggal dalam satu siklus dan memiliki struktur I/O yang cukup lengkap sehingga penggunaan komponen eksternal dapat dikurangi. Mikrokontroler AVR didesain menggunakan arsitektur *Harvard*, di mana ruang dan jalur bus bagi memori program dipisahkan dengan memori data. Memori program diakses dengan *single-level pipelining*, di mana ketika sebuah instruksi dijalankan, instruksi lain berikutnya akan di-*prefetch* dari memori program.

AVR ATmega 8535 memiliki bagian sebagai berikut:

1. Saluran I/O sebanyak 32 buah, yaitu port A, port B, port C, dan port D
2. CPU yang memiliki 32 buah register
3. ADC 10 bit sebanyak 8 saluran.
4. SRAM sebesar 512 byte
5. Flash memory sebesar 8kb
6. EEPROM sebesar 512 byte
7. Tiga buah *timer/counter* dengan kemampuan pembanding
8. *Two wire serial Interface*
9. Port antarmuka SPI
10. Antarmuka komparator analog.

## 12. Port USART untuk komunikasi serial



**Gambar 2.12. Deskripsi Pin ATMEGA 8535**

### 2.3.1 Fitur ATMEGA8535

Kapabilitas detail dari ATMEGA8535 adalah sebagai berikut:

1. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
2. Kapabilitas memori flash 8 KB, SRAM sebesar 512 byte, dan EEPROM sebesar 512 byte.
3. ADC internal dengan fidelitas 10 bit sebanyak 8 channel.
4. Portal komunikasi serial (USART) dengan kecepatan 2,5Mbps.
5. Enam pilihan mode sleep menghemat penggunaan daya listrik.

### 2.3.2 Konfigurasi Pin ATMEGA8535

1. Port A (PA7..PA0)

juga berfungsi sebagai suatu Port I/O 8-bit dua arah, jika A/D Konverter tidak digunakan. Pin-pin Port dapat menyediakan resistor *internal pull-up* (yang dipilih untuk masing-masing bit). Port A *output buffer* mempunyai karakteristik gerakan simetris dengan keduanya *sink* tinggi dan kemampuan sumber. Ketika pin PA0 ke PA7 digunakan sebagai *input* dan secara *eksternal* ditarikrendah, pin-pin akan memungkinkan arus sumber jika resistor *internal pull-up* diaktifkan. Pin Port A adalah *tri-stated* manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

## 2. Port B (PB7..PB0)

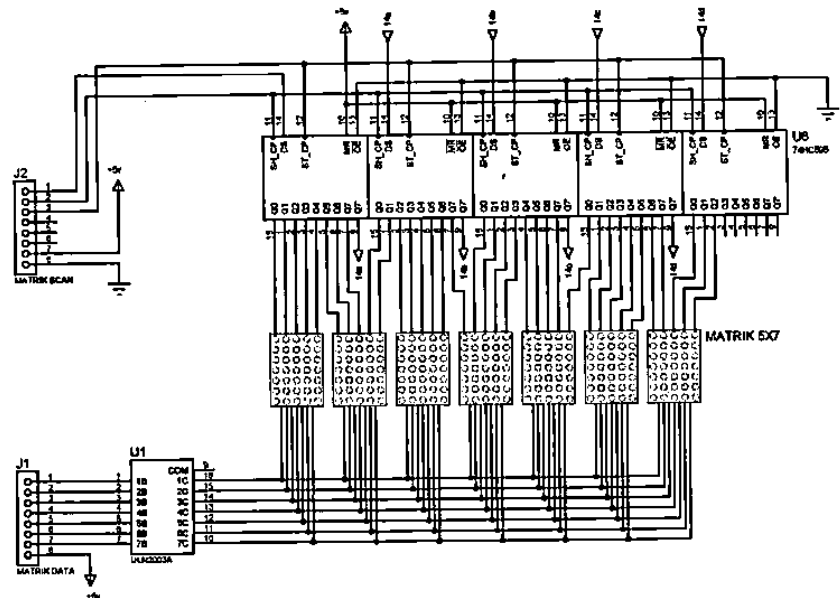
Port B adalah suatu Port I/O 8-bit dua arah dengan resistor internal pull-up (yang dipilih untuk beberapa bit). Port B *output buffer* mempunyai karakteristik gerakan simetris dengan keduanya *sink* tinggi dan kemampuan sumber. Sebagai input, pin port B yang secara *eksternal* ditarik rendah akan arus sumber jika resistor *pullup* diaktifkan. Pin Port B adalah *tri-stated* manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

## 3. Port C (PC7..PC0)

Port C adalah suatu Port I/O 8-bit dua arah dengan resistor internal pull-up (yang dipilih untuk beberapa bit). Port C *output buffer* mempunyai karakteristik gerakan simetris dengan keduanya *sink* tinggi dan kemampuan sumber. Sebagai input, pin port C yang secara *eksternal* ditarik rendah akan arus sumber jika resistor *pullup* diaktifkan. Pin Port C adalah *tri-stated* manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

### 3.1.1.2. Rangkaian *Running Text* Dot Matrik

Rangkaian ini menggunakan komponen shif register 74HC595 yang berfungsi menggeser data dari mikrokontroller setiap kali pin clock shif register ini diberi logika 1 maka data akan bergeser 1 bit.



Gambar 3.2 Skema rangkaian *running text*

Rangkaian *running text* atau penampil Dot matriks menggunakan led dot matrik 5x7 sehingga untuk menampilkan 1 karakter dibutuhkan 35 led. Pergeseran data pada led dotmatrik menggunakan IC 74HC595, kemudian outputnya dikuatkan menggunakan IC 74HC573.

### 3.1.1.3. Rangkaian Komunikasi Serial RS232

Rangkaian Level tegangan RS232 data output HP harus diubah menjadi level tegangan TTL ( transistor transistor logic) pada sistem minimum mikrokontroler AVR ATmega8535. IC Max232 digunakan untuk

waktu habis.

#### 4. Port D (PD7..PD0)

Port D adalah suatu Port I/O 8-bit dua arah dengan resistor internal pull-up (yang dipilih untuk beberapa bit). Port D *output buffer* mempunyai karakteristik gerakan simetris dengan keduanya *sink* tinggi dan kemampuan sumber. Sebagai input, pin port D yang secara *eksternal* ditarik rendah akan arus sumber jika resistor *pullup* diaktifkan. Pin Port D adalah *tri-stated* manakala suatu kondisi

1. reset menjadi aktif, sekalipun waktu habis.
2. RESET (*Reset input*)
3. XTAL1 (*Input Oscillator*)
4. XTAL2 (*Output Oscillator*) AVCC adalah pin penyedia tegangan untuk port A dan A/D Konverter
5. AREF adalah pin referensi analog untuk A/D konverter.
6. Vcc (power supply)
7. Ground

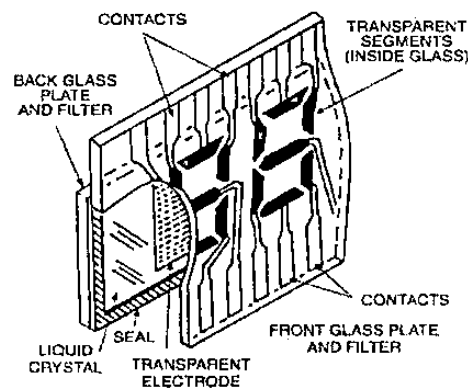
(Lingga Wardhana, 2006).

## 2.4 LCD

LCD dibuat dari kristal cair yang merespon adanya medan listrik. Kristal tersebut terdiri atas molekul seperti batang yang apabila terkena medan listrik akan menyusun diri agar melewatkan atau menahan cahaya yang mengenainya.

Gambar 2.4 menunjukkan gambar beberapa jenis layar tampilan LCD dapat terlihat

Lapisan film yang berisi kristal cair diletakkan diantara dua lempeng kaca yang telah ditanami elektroda logam transparan, seperti terlihat pada gambar 2. 10. Saat tegangan dicatukan pada beberapa pasang elektroda, molekul-molekul kristal cair akan menyusun diri agar cahaya yang mengenainya akan dipantulkan atau diserap. Dari hasil pemantulan atau penyerapan cahaya tersebut akan terbentuk pola huruf, angka, atau gambar sesuai bagian yang diaktifkan.



**Gambar 2. 13. Liquid Crystal Display**

LCD membutuhkan tegangan dan daya yang kecil sehingga sangat populer untuk aplikasi pada kalkulator, arloji digital, dan instrumen elektronik lain seperti Global Positioning System (GPS), bargraph display, dan multimeter digital. LCD umumnya dikemas dalam bentuk Dual In-line Package (DIP) dan mempunyai kemampuan untuk menampilkan beberapa kolom dan baris dalam satu panel. Untuk membentuk pola, baik karakter ataupun gambar, pada kolom dan baris secara bersamaan digunakan metode screening. Metode screening adalah mengaktifkan daerah perpotongan suatu kolom dan suatu baris secara bergantian dan cepat sehingga seolah-olah aktif semua. Penggunaan metode ini

LCD. Sekarang ini, berbagai jenis LCD telah dikembangkan. Dari jenis LCD biasa, Passive-Matrix LCD (PMLCD), hingga Thin-Film Transistor Active-Matrix LCD (TFT-AMLCD). Kemampuan LCD juga telah ditingkatkan, dari yang monokrom hingga yang mampu menampilkan ribuan warna (Syaiful Anwar, 2005).

## 2.5 Catu Daya

Rangkaian catu daya terdiri dari Transformator, IC Regulator, Dioda penyearah (*rectifier*) yang mengubah arus bolak-balik (AC) menjadi arus searah (DC) dan kapasitor yang berfungsi sebagai filter untuk mengurangi riak (*ripple*) pada tegangan keluaran.

### 1) Transformator

Transformator merupakan alat yang berfungsi untuk memindahkan daya dari satu rangkaian yang lain, secara induksi elektromagnetik dengan tidak mengubah harga frekwensinya (Sunyoto 1996). Transformator ada 2 (dua) jenis yaitu:

#### a) Transformator step-up (untuk menaikkan tegangan)

Tegangan pada sisi primer lebih kecil dibanding sisi sekunder.

#### b) Transformator step-down (untuk menurunkan tegangan)

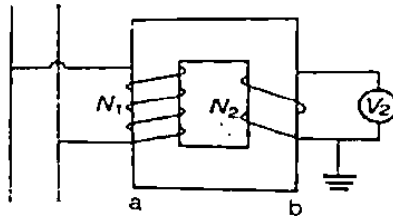
Tegangan pada sisi primer lebih besar dibanding sisi sekunder.

Transformator step-down banyak digunakan pada rangkaian elektronika,

transformator yang digunakan memiliki tegangan primer 220 Volt AC dan

tegangan sekunder 6 Volt AC hingga 12 Volt AC.

Berikut ini gambar dari Transformator:



**Gambar 2.14. Transformator**

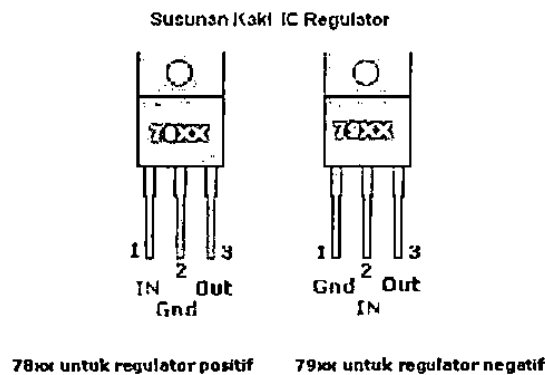
## 2) IC Regulator

IC regulator berfungsi untuk menstabilkan tegangan keluaran pada catu daya. Tegangan suatu rangkaian elektronika yang berubah-ubah besarnya, dapat mengakibatkan fungsi kerja rangkaian yang dicatunya rusak. Jika dari suatu rangkaian elektronika diharapkan suatu kinerja yang baik dan tahan lama, maka salah satu syaratnya adalah dengan menggunakan catu daya yang stabil dan mampu menekan riak (*ripple*) semaksimal mungkin. Catu daya yang stabil dan dapat diatur ini disebut *regulator*.

Regulator tegangan secara garis besar dibagi menjadi dua jenis yaitu regulator tegangan tetap dan regulator tegangan yang dapat diatur. IC regulator tegangan tetap yang sekarang ini banyak kita jumpai adalah seri 78XX untuk tegangan positif dan seri 79XX untuk tegangan negative.

Salah satu IC regulator adalah 7805V.





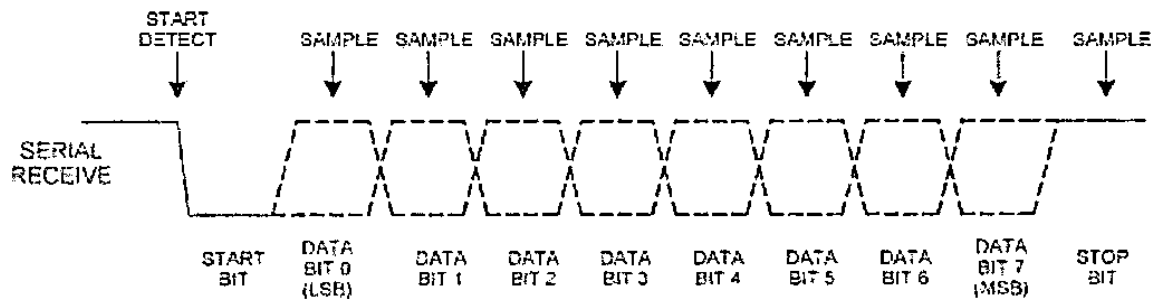
**Gambar 2.15. IC Regulator**

## 2.6 Komunikasi Data Serial UART (*Universal Asynchronous Receiver and Transmitter*)

Ada 2 (dua) cara komunikasi data secara serial. Kedua cara tersebut dibedakan oleh sinyal detak yang dipakai untuk mendorong data serial, jika detak dikirim bersama-sama dengan data serial, cara tersebut dikatakan sebagai transmisi data serial secara sinkron. Sedangkan dalam transmisi data serial secara asinkron, detak tidak dikirim secara bersama data serial dan rangkaian penerima harus membangkitkan sendiri detak pendorong data serial. (Agfianto, 2002)

Pengiriman data menggunakan UART (*Universal Asynchronous Receiver and Transmitter*) akan melibatkan pengoperasian 4 buah register UART, yaitu UDR, UCR, USR dan UBRR.

Pada pengiriman data menggunakan UART, receiver UART akan



**Gambar 2.16. Frame dari UART**

Komunikasi data yang digunakan pada *handphone* biasanya bersifat asinkron atau yang lebih dikenal dengan UART (*Universal Asynchronous Receiver and Transmitter*).

Pada komunikasi jenis ini data transfer per-bit persatuan waktu. Ketika jalur sedang menunggu maka logikanya ditahan tetap tinggi (*high* atau bernilai 1). Ketika data mulai ditransfer maka bit pertama yang dikirim atau disebut *Start bit* bernilai '0' atau low sehingga terjadi transmisi dari 1 ke 0 pada jalur pengiriman. Ini dapat digunakan oleh penerima sebagai pertandaan bahwa data akan dikirim.

Bit-bit setelahnya merupakan data yang akan ditransfer dengan urutan *least significant bit* (LSB) terlebih dahulu. Kemudian diakhiri oleh satu atau lebih *Stop bit* yang bernilai '1'. Dengan memberi logika '1' setelah *Stop bit* pada jalur menunggu maka proses pengiriman selesai. Paling tidak harus ada 1 *stop bit* sebelum tegangan jalur menunggu dicatu tinggi '1' sehingga transmisi '1' ke '0' (*awal penerima data*) dapat dideteksi oleh penerima yang berarti *start bit* baru.

Daftar Isi

Tabel 2.5 Pengeturan *BAUDRATE*

Baud Rate	3.2768 MHz	% Error	3.6864 MHz	% Error	4 MHz	% Error	4.608 MHz	% Error
2400	UBRR- 84	0.4	UBRR- 95	0.0	UBRR- 103	0.2	UBRR- 119	0.0
4800	UBRR- 42	0.8	UBRR- 47	0.0	UBRR- 51	0.2	UBRR- 59	0.0
9600	UBRR- 20	1.6	UBRR- 23	0.0	UBRR- 25	0.2	UBRR- 29	0.0
14400	UBRR- 13	1.6	UBRR- 15	0.0	UBRR- 16	2.1	UBRR- 19	0.0
19200	UBRR- 10	3.1	UBRR- 11	0.0	UBRR- 12	0.2	UBRR- 14	0.0
28800	UBRR- 6	1.6	UBRR- 7	0.0	UBRR- 8	3.7	UBRR- 9	0.0
38400	UBRR- 4	6.3	UBRR- 5	0.0	UBRR- 6	7.5	UBRR- 7	6.7
57600	UBRR- 3	12.5	UBRR- 3	0.0	UBRR- 3	7.8	UBRR- 4	0.0
76800	UBRR- 2	12.5	UBRR- 2	0.0	UBRR- 2	7.8	UBRR- 3	6.7
115200	UBRR- 1	12.5	UBRR- 1	0.0	UBRR- 1	7.8	UBRR- 29	20.0

Kecepatan *baudrate* ini juga bisa dicari dengan menggunakan rumus:

$$\text{BAUD} = \text{JCK} / 16 (\text{UBRR} + 1)$$

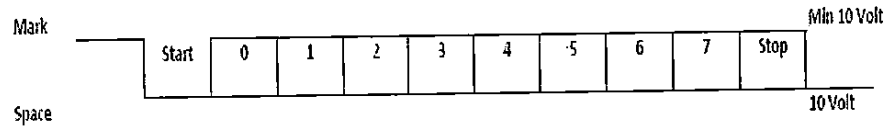
Komunikasi serial dengan computer memiliki tegangan kerja sekitar 11 Volt tanpa beban atau sekitar 8-9 Volt jika ada beban. Untuk berkomunikasi dengan *handphone* maka diperlukan penyetara tegangan logika dimana tegangan kerja *handphone* berkisar antara 2,5 Volt. Untuk mengatasi hal tersebut maka dipakai IC DC-DC *converter*, dimana dalam hal ini menggunakan *MAX 232 CPE* mengubah tegangan *TTL-CMOS* sekitar 5 Volt dan sekitarnya.

## 2.7 Antarmuka

### a. Format Data RS232

Komunikasi RS232 bersifat *asinkro (asynchronous)*, artinya sinyal *clock*

tidak diperlukan bersamaan dengan data. Berikut ini diagram struktur data

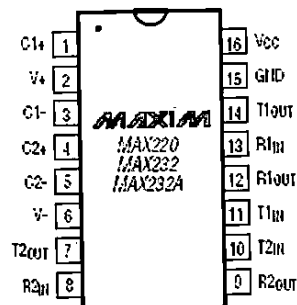


**Gambar 2.17** Struktur data 8N1 tingkat RS232

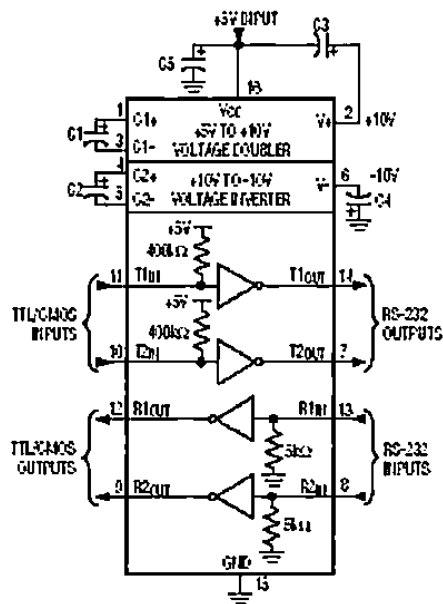
Gambar diatas berlaku untuk jalur kirim dan terima pada kanal RS232, jalur ini membawa data secara serial, sehingga dinamakan kanal serial.

*b. IC MAX 232*

Komunikasi dengan port serial sebagai pengujian data yang dikirim kemudian hasilnya dapat dilihat pada computer. Oleh karena itu dibutuhkan sebuah antarmuka RS232 sebagai perantara port serial pada Mikrokontroller Atmega8535 dengan PC atau computer. Piranti *IC MAX 232* sebagai antarmuka RS232 memiliki sebuah *charge pump* yang dapat menghasilkan tegangan + 10 Volt dan - 10 Volt dari catu daya tunggal 5 Volt. IC ini juga memiliki penerima dan pengirim ganda pada kemasan yang sama, sehingga tidak memerlukan dua IC dalam proses pengiriman dan penerimaan data.



**Gambar 2.18** Diagram pin *IC MAX 232*



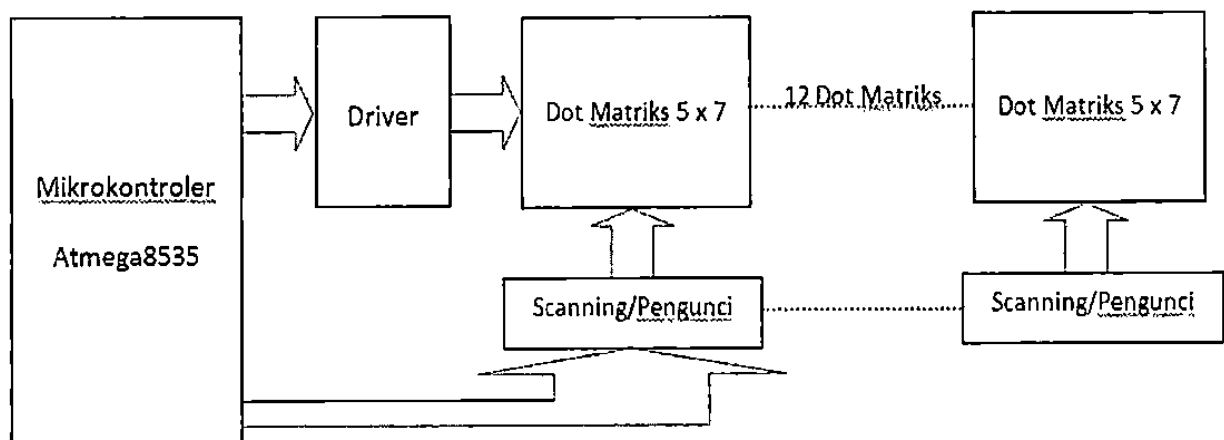
Gambar 2.19 Rangkaian umum IC MAX 232

## 2.8 Penampil Dot Matriks

Penampil Dot Matriks merupakan salah satu bentuk peralatan penampil yang digunakan untuk memberikan informasi-informasi terhadap penggunanya, dibentuk dengan beberapa komponen elektronika dan titik sumber cahaya yang berasal dari LED (Light Dipendent Resistor).

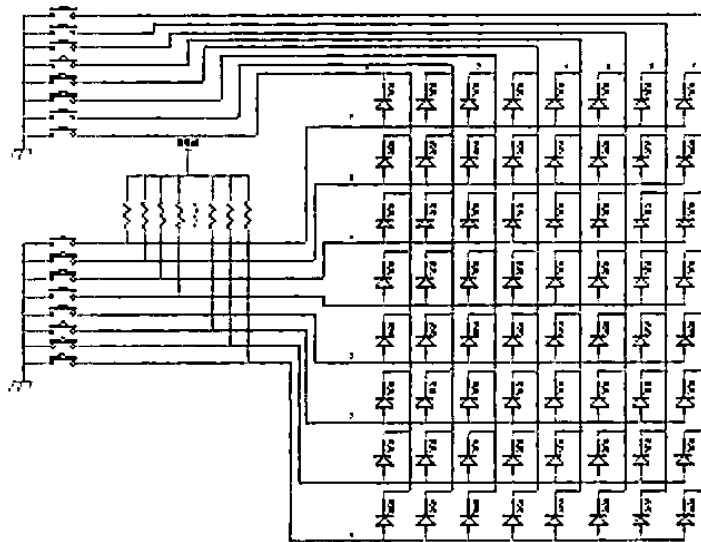
Kelebihan penampil Dot Matriks yaitu dapat menampilkan karakter-karakter seperti alphanumeric atau seperti standar simbol ASCII pada LCD dan dapat membentuk berbagai bentuk tampilan, sehingga Dot Matriks banyak digunakan dalam beberapa aplikasi seperti papan score, up counter, down counter, timer, moving sign display kurs valas, mesin antrian digital, industrial counter, ispu

Rangkaian perangkat-keras dapat dibagi menjadi beberapa bagian yaitu pengendali, *scanning*, sistem *switching* dan Dot Matriks. pembagian perangkat keras ini mengingat sistem kerja perangkat yang digunakan mempunyai spesifikasi dan fungsi sendiri-sendiri yang kemudian dipadu menjadi satu bagian yang hasilnya sistem pengendali penampil Dot Matriks



**Gambar 2.20** Blok Diagram Perangkat-Penampil Dot Matriks

Jenis Dot Matriks  $5 \times 7$  terdiri dari 5 kolom dan 7 baris, dengan arti 5 led kolom dan 7 led baris sehingga terdapat  $5 \times 7 = 35$  led dalam satu *pack* Dot Matriks. Sehingga apabila akan mengatur nyala led secara individu/ tidak bebarengan maka diperlukan 35 saklar. Hal ini mengakibatkan pengaturannya menjadi tidak praktis. Cara lain untuk mengatur nyala led adalah dengan menggunakan led yang berwujud



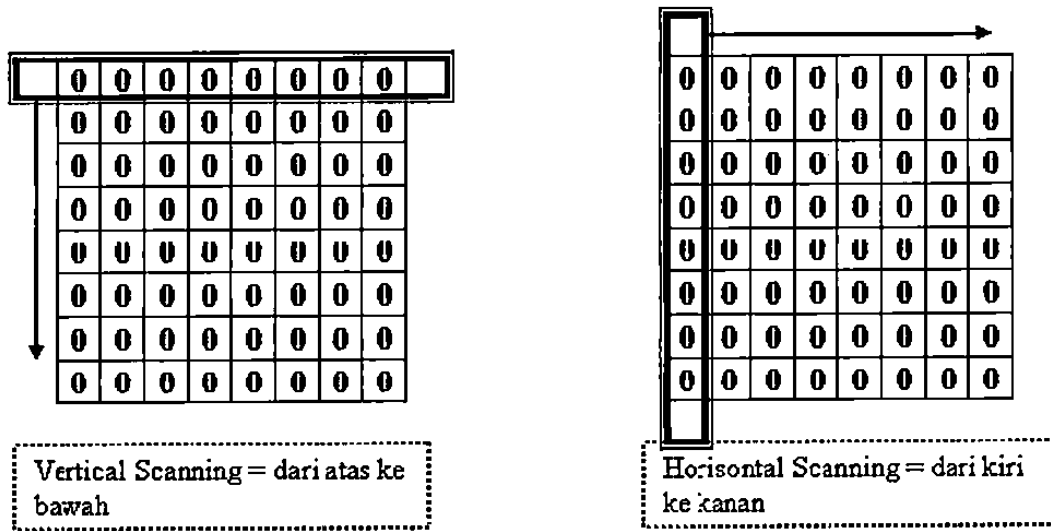
**Gambar 2.21** Susunan LED Matriks

Dengan penyusunan seperti diatas, maka saklar pengatur nyala atau mati led dapat dikurangi, hanya saja proses cara pengaturan nyala atau mati led tidak sama dengan saklar biasa.

Proses pembangkitan nyala led melibatkan proses *Scanning* baris/kolom. Memilih led yang tepat pada baris/kolom dan menyalakanya, proses ini diulangi untuk baris/kolom berikutnya dan seterusnya. Setelah semua baris/kolom dipilih dengan urutan, proses diatas diulangi mulai dari baris paling atas atau kolom pertama.

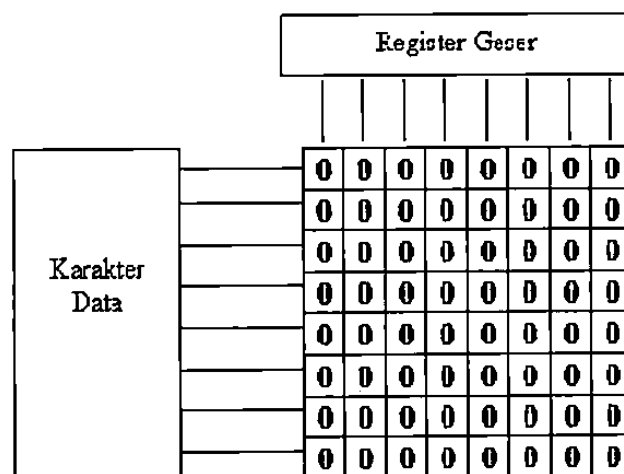
*Scanning* mode merupakan sistem pengiriman data kepada beberapa alamat secara bergantian. sistem *scanning* dimanfaatkan untuk mengirimkan data kepada setiap baris/kolom pada Dot Matriks, dengan mengatur penundaan waktu pengiriman

1. ... .. dan ... ..



Gambar 2.22 Sistem Scanning Dot Matriks

Jika matriks di-*scan* dari kiri ke kanan, kolom demi kolom, disebut horizontal *scanning*, jika dilakukan baris demi baris, disebut vertical *scanning*. Horizontal *scanning* cocok untuk tampilan sampai dengan empat karakter, sedangkan vertical *scanning* cocok untuk aplikasi yang memerlukan tampilan lebih dari lima karakter. Dengan karakteristik Dot Matriks seperti diatas, maka untuk menampilkan led maupun sebuah karakter pada Dot Matriks diperlukan sebuah register geser dan sumber data atau karakter



Gambar 2.23 Sistem Pengendali Dot matriks



Karakter Data merupakan papan masukan 8 bit yang berasal dari papan ketik atau dari piranti digital lain yang bisa berupa sembarang sandi biner. Dan Register geser digunakan untuk mengamati sandi pembangkit karakter Led yang menyediakan 8 bit register geser dan yang melakukan sistem *scanning* Dot Matriks. Apabila akan digunakan beberapa Dot Matriks, misal 12 Dot Matriks, maka diperlukan lagi piranti pengalamatan dan pewaktuan nyala setiap kolomnya, sehingga sekiranya menjadi tidak praktis dan efisien, belum lagi sumber pembangkitan karakter missal seperti karakter yang sesuai dengan sandi ASCII. Dengan dasar pemikiran tersebut diatas, maka digunakan mikrokontroler yang digunakan sebagai pembangkit karakter sandi ASCII atau simbol-simbol lainnya dan juga mikrokontroler sekaligus sebagai pengontrol pengalamatan Dot Matriks dan pergeseran kolom/baris atau kendali sistem *scanning*-nya.

(Sumber: <http://shatomedia.com/2009/02/penampil-dotmatiks-1>; Tanggal 20 Mei 2010 / Pukul 20.12 Wib)

## **2.9 Spesifikasi Awal**

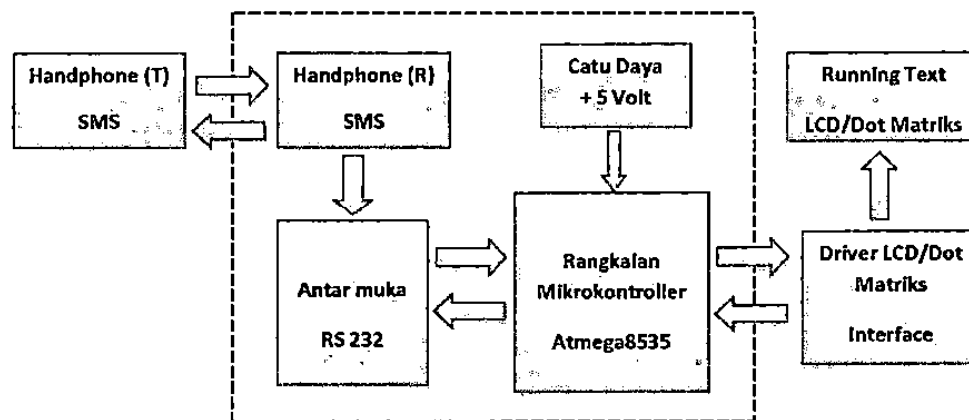
Sebagaimana yang telah dikemukakan di bagian-bagian sebelumnya dari bab ini, maka berikut ini dikemukakan spesifikasi awal dari alat yang akan dirancang dan dibuat dalam Tugas Akhir ini.

### **2.9.1 Perangkat Keras (*Hardware*)**

Perangkat keras yang dibutuhkan meliputi bagian elektronik sebagai unit pengirim dan penerima SMS (*Handphone* Sonyericsson T290), Bagian elektronik yang dibangun meliputi rangkaian catu daya,

antara lain RS232, mikrokontroler dan penampil pesan berupa dot Matriks

Diagram blok yang menunjukkan interaksi pada bagian elektronik tersebut dapat dilihat pada gambar 3.1 dibawah ini.



**Gambar 2.24** Diagram blok alat yang akan dibuat

**Keterangan gambar :**

1. *Handphone T (Transceiver)* adalah *handphone* pengirim SMS yang dipegang oleh operator yang kegunaannya untuk meng-update Running text (tulisan berjalan).
2. *Handphone R (receiver)* adalah *handphone* penerima SMS yang terhubung ke bagian pengendali mikrokontroler melalui serial RS232.
3. Antarmuka Komunikasi data Serial RS232, Komunikasi data yang digunakan adalah asinkron atau yang lebih dikenal dengan UART (*Universal Asynchronous Receiver and Transmitter*).
4. Rangkaian Catu Daya, berfungsi untuk mensuplai daya pada rangkaian, ada 2 tegangan yang dihasilkan dari rangkaian tersebut, 5 volt untuk rangkaian mikrokontroler dan 12 volt untuk mensuplai daya pada Dot Matrik.

5. MCU berfungsi untuk memolah input SMS dari Handphone R

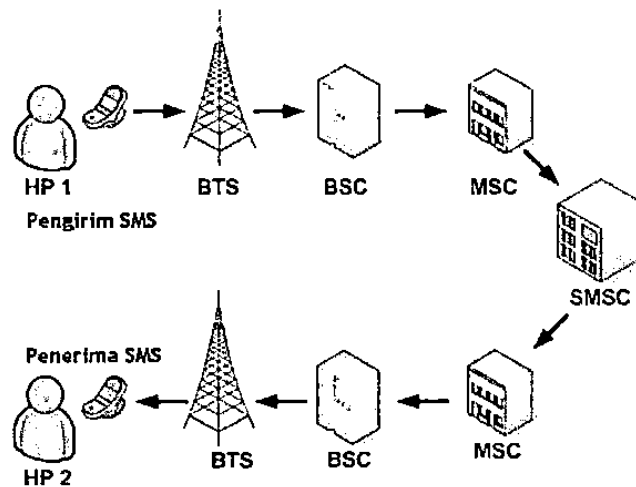
sehingga menghasilkan output pesan teks dari SMS yang sesuai dengan harapan.

6. Rangkaian Driver dot Matrik dan dot Matrik, berfungsi untuk menampilkan pesan SMS sesuai dengan apa yang dikirimkan oleh *Handphone T*.

### 2.9.2 Prinsip Kerja Alat

Short Message Service (SMS) merupakan layanan *handphone* dua arah yang mampu mengirimkan pesan singkat 160 karakter yang dapat disimpan dan di foward. Secara sederhana dapat dijelaskan, melalui SMS sebuah pesan dapat dikirim ke penerima dan sebuah tanda akan memberitahukan penerima bahwa ada pesan yang masuk. SMS juga dapat digunakan dalam mode cell broadcast untuk mengirim berita-berita terbaru dan pemberitahuan penting yang lainnya yang bersifat missal. Pesan SMS tersebut juga dapat disimpan didalam memori SIM Card dan memori *Handphone*.

Diagram ini adalah gambar ilustrasi umum dari pengiriman dan



**Gambar 2.25** Ilustrasi umum pengirim dan penerima SMS

Perancangan system update running text menggunakan SMS ini menggunakan *Handphone* Sonyericsson T290, ada dua *handphone* yang digunakan yaitu:

- a. *Handphone* T (transceiver) sebagai pengirim perintah berupa teks yang dipegang oleh operator.
- b. *Handphone* R (receiver) sebagai penerima perintah berupa teks yang terhubung oleh pengendali.

Secara umu prinsip kerjanya sebagai berikut:

*Handphone* T (*Transceiver*) akan mengirimkan pesan teks yang berisi update running teks ke *handphone* R (*Receiver*), kemudian *handphone* R akan memberikan pesan ke Mikrokontroler melalui komunikasi serial RS232, pesan yang diterima dari *handphone* masih berupa pesan *PDU* (Protocol data unit), pesan tersebut harus dirubah menjadi data *ASCII* (American Standard Code for Information Interchange) kemudian disimpan di RAM mikrokontroler, selanjutnya

dot matrik 5x7 ditambah 1 spasi, setelah data *patern* di konversi maka akan di tampilkan ke dot matrik.

### 2.9.3 Perangkat Lunak (Software)

Perangkat lunak dibutuhkan oleh mikrokontroler untuk menterjemahkan pesan yang diterima dan menampilkan pesan sesuai dengan yang dikirimkan. Berbagai jenis bahasa pemrograman dapat digunakan untuk membuat aplikasi pada mikrokontroler. Pada alat ini bahasa pemrograman yang akan digunakan adalah bahasa *Basic*. Penggunaan bahasa pemrograman *Basic* pada alat karena lebih mudah dipahami dan lebih cepat dipelajari dibanding dengan bahasa pemrograman *assembler* dan bahasa pemrograman *C*. Adapun *flowchart* dari program yang akan dibuat dapat dilihat pada gambar 2.26 di bawah



#### **2.9.4 Kriteria Teknis**

Kriteria teknis yang harus dipenuhi oleh alat tersebut adalah dapat meng-update running text dari jarak jauh, dengan menggunakan SMS dari handphone melalui operator GSM dengan nomor running text dan