

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Pada penelitian sebelumnya telah dibuat stetoskop yang mampu menampilkan grafik *ECG* pada *PC* oleh R. Arry Mustafa Yudhanegara dan Andri Okapriadi. Penelitian tersebut dirancang dengan sensor yang digunakan merupakan *mic electret* yang berukuran kecil sehingga dapat diletakan pada stetoskop. Rangkaian filter serta penguat yang digunakan dalam blok rangkaian menggunakan rangkaian *Op-amp 741. ADC* yang mengirimkan sinyal digital dari sinyal analog yang berasal dari penguat ke *PC*. Cara kerja dari alat tersebut dimulai dari sensor yang berupa *mic electret* mendeteksi getaran yang dihasilkan oleh detak jantung pada dada seseorang. Kemudian sinyal hasil pembacaan tersebut diperkuat beberapa kali dan dikeluarkan melalui *speaker (earphone)*. Selain itu, sinyal hasil penguatan tersebut diolah lagi oleh serial *ADC* untuk dikomunikasikan ke *PC* dan dibuat tampilan grafik *ECG* dari detak jantung yang dibaca. Kekurangan dari alat ini adalah alat yang digunakan masih berbasis digital dan tidak menggunakan *microcontroller*, sehingga kemungkinan *noise* yang dihasilkan cukup besar karena sinyal yang dideteksi oleh *mic electret* memiliki frekuensi yang berubah-ubah, selain itu alat ini juga belum dilengkapi dengan perhitungan denyut jantung per menit (*BPM*) [R. Arry Mustafa Yudhanegara, Stetoskop Digital Dengan Tampilan Grafik EKG pada PC, 2013].

Denok Santi Ardiningsih (2008) pada Tugas Akhir dengan judul “stetoskop elektronik berbasis *microcontroller* AT89S52 dengan kelebihan menampilkan

denyut jantung per menit (*heart beat*)” dengan kelebihan menampilkan denyut jantung per menit (*heart beat*). Prinsip kerja dari alat tersebut yaitu stetoskop yang didalamnya terdapat *mic condenser* mendeteksi bunyi jantung yang kemudian dikuatkan suaranya menggunakan rangkaian penguat awal, penguat audio dan penguat akhir serta dilengkapi rangkaian *filter* berupa *low pass filter*. Selain itu, terdapat rangkaian pengolah sinyal berupa rangkaian *integrator* dan komparator yang berfungsi sebagai pengubah sinyal kotak menjadi sinyal segitiga dan membandingkan sinyal *input high* atau *low* dengan sinyal referensi untuk masukan *microcontroller* AT89S52. Hasil keluaran dari rangkaian tersebut diteruskan ke *microcontroller* AT89S52 agar dapat mengolah sinyal denyut jantung yang masuk yang kemudian dilakukan perhitungan denyut jantung per menit (*heart beat*) dan hasilnya ditampilkan di *display seven segment*. Catu daya yang digunakan pada alat ini berupa baterai 9V sebanyak 2 buah. Pada penelitian ini masih terdapat kekurangan, yaitu alat ini memiliki banyak blok rangkaian yang rumit dan masih menggunakan baterai yang *non rechargable* sehingga kurang efisien apabila tegangan pada baterai telah habis. [Denok Santi Ardiningsih, Stetoskop Elektronik Berbasis Mikrokontroler AT89S52 Dengan Kelebihan Menampilkan Denyut Jantung Per Menit (*heart beat*), 2008].

2.2 Dasar Teori

2.2.1 Anatomi Jantung

Jantung berbentuk seperti pir/kerucut seperti piramida terbalik dengan *apeks* (superior-posterior:C-II) berada di bawah dan basis (anterior-inferior ICS-V) berada di atas. Pada basis jantung terdapat aorta, batang nadi paru, pembuluh

balik atas dan bawah dan pembuluh balik. Jantung sebagai pusat sistem kardiovaskular terletak di sebelah rongga dada (*cavum thoraks*) sebelah kiri yang terlindung oleh *costae* tepatnya pada mediastinum. Untuk mengetahui denyutan jantung, kita dapat memeriksa dibawah *papilla mammae* 2 jari setelahnya. Berat pada orang dewasa sekitar 250-350 gram. Hubungan jantung dengan alat sekitarnya yaitu:

- a. Dinding depan berhubungan dengan *sternum* dan kartilago *kostalis* setinggi *kosta III-I*.
- b. Samping berhubungan dengan paru dan *fascies mediastilais*.
- c. Atas setinggi *torakal IV* dan *servikal II* berhubungan dengan *aorta pulmonalis*, bronkus *dekstra* dan bronkus *sinistra*.
- d. Belakang alat-alat *mediastinum* posterior, esofagus, *aorta descendes*, vena *azigos*, dan *kolumna vetebrata torakalis*.
- e. Bagian bawah berhubungan dengan diafragma.

Jantung difiksasi pada tempatnya agar tidak mudah berpindah tempat. Penyokong jantung utama adalah paru yang menekan jantung dari samping, diafragma menyokong dari bawah, pembuluh darah yang keluar masuk dari jantung sehingga jantung tidak mudah berpindah. Faktor-faktor yang mempengaruhi letak dan kondisi jantung adalah:

- a. Umur

Pada usia lanjut, alat-alat dalam rongga *thoraks* termasuk jantung letaknya menjadi agak turun kebawah.

b. Bentuk rongga dada

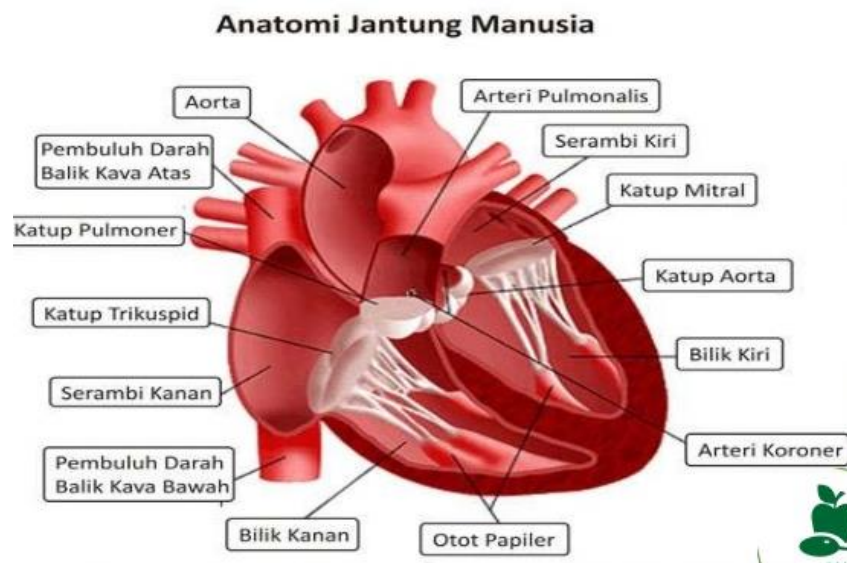
Perubahan bentuk *thoraks* yang menetap (TBC) menahun membuat letak batas jantung menurun sehingga pada asma, bentuk *thoraks* menjadi melebar dan membulat.

c. Letak diafragma

Jika terjadi penekanan diafragma keatas, maka akan mendorong bagian bawah jantung ke atas.

d. Perubahan posisi tubuh.

e. Proyeksi jantung normal di pengaruhi oleh posisi tubuh. Anatomi jantung ditunjukkan oleh Gambar 2.1.



Gambar 2.1 Anatomi jantung.

2.2.2 Otot Jantung

Otot jantung terdiri atas 3 lapisan, yaitu:

a. Luar/*pericardium*

Otot jantung bagian luar/*pericardium* berfungsi sebagai pelindung jantung atau merupakan kantong pembungkus jantung yang terletak di *mediastinum* minus dan di belakang *korpus sterni* dan rawan iga II- IV yang terdiri dari 2 lapisan *fibrosa* dan serosa yaitu lapisan *parietal* dan *viseral*. Diantara dua lapisan jantung ini terdapat lendir sebagai pelicin untuk menjaga agar gesekan *pericardium* tidak mengganggu jantung.

b. Tengah/ *miocardium*

Lapisan tengah otot jantung yaitu lapisan otot jantung yang menerima darah dari *arteri koronaria*. Susunan *miocardium*, yaitu:

1. Otot *atria*: sangat tipis dan kurang teratur, disusun oleh dua lapisan. Lapisan dalam mencakup serabut-serabut berbentuk lingkaran dan lapisan luar mencakup kedua *atria*.
2. Otot *ventrikular*: membentuk bilik jantung dimulai dari cincin *atrioventrikuler* sampai ke *apeks* jantung.
3. Otot *atrioventrikuler*: dinding pemisah antara serambi dan bilik (atrium dan ventrikel).

c. Dalam/*endocardium*

Dinding dalam atrium yang diliputi oleh membran yang mengilat yang terdiri dari jaringan *endotel* atau selaput lendir *endokardium* kecuali *aurikula* dan bagian depan *sinus vena kava*.

2.2.3 Bagian-Bagian Jantung

Jantung terdiri dari beberapa bagian. Bagian-bagian dari jantung adalah:

- a. Basis *kordis*: bagian jantung sebelah atas yang berhubungan dengan pembuluh darah besar dan dibentuk oleh atrium *sinistra* dan sebagian oleh atrium *dekstra*.
- b. *Apeks kordis* : bagian bawah jantung berbentuk puncak kerucut tumpul.

2.2.4 Fisiologis Jantung

Fungsi umum otot jantung yaitu:

- a. Sifat ritmisitas/otomatis: fungsi jantung secara potensial berkontraksi tanpa adanya rangsangan dari luar.
- b. Mengikuti hukum gagal atau tuntas: impuls dilepas mencapai ambang rangsang otot jantung maka seluruh jantung akan berkontraksi maksimal.
- c. Tidak dapat berkontraksi tetanik.
- d. Kekuatan kontraksi dipengaruhi panjang awal otot

2.2.5 Siklus Jantung

Siklus jantung terjadi di empat pompa yang terpisah, yaitu dua pompa primer atrium dan dua pompa tenaga ventrikel. Periode akhir kontraksi jantung sampai kontraksi berikutnya disebut siklus jantung.

2.2.6 Fungsi Jantung Sebagai Pompa

Lima fungsi jantung sebagai pompa, yaitu:

- a. Fungsi atrium sebagai pompa.
- b. Fungsi ventrikel sebagai pompa.

- c. Periode *ejeksi* saat sebuah pengukuran darah yang dipompa keluar dari *ventricular* terisi.
- d. *Diastole* satu periode relaksasi.
- e. Periode relaksasi *isometric* adalah periode terjadi kontraksi didalam *ventricular*, namun belum ada pengosongan.

Dua cara dasar pengaturan kerja pemompaan jantung, terdiri dari *autoregulasi intrinsic*, yaitu pemompaan akibat perubahan *volume* darah yang mengalir ke jantung dan *reflex*, yaitu mengawasi kecepatan dan kekuatan kontraksi jantung melalui saraf otonom.

Denyut jantung yang normal yakni 60-100 kali setiap menit, sedang denyut jantung lambat kurang dari 60 kali per menit dan yang cepat lebih dari 100 kali per menit. Jumlah denyut jantung yang normal berdasarkan usia seseorang, adalah:

- a. Bayi baru lahir :140 kali per menit
- b. Umur <1 bulan :110 kali per menit
- c. Umur 1-6 bulan :130 kali per menit
- d. Umur 6-12 bulan :115 kali per menit
- e. Umur 1-2 tahun : 110 kali per menit
- f. Umur 2-6 tahun : 105 kali per menit
- g. Umur 6-10 tahun : 95 kali per menit
- h. Umur 10-14 tahun : 85 kali per menit
- i. Umur 14-18 tahun : 82 kali per menit
- j. Umur >18 tahun : 60-100 kali per menit

- k. Usia lanjut : 60-70 kali per menit.

2.2.7 Curah jantung

Normalnya, jumlah darah yang dipompakan ventrikel kiri dan kanan sama besarnya. Jumlah darah yang dipompakan ventrikel selama satu menit disebut curah jantung (*cardiac output*). Faktor-faktor utama yang mempengaruhi otot jantung, yaitu:

- a. Beban awal.
- b. Kontraktilitas.
- c. Beban akhir.
- d. Frekuensi jantung.

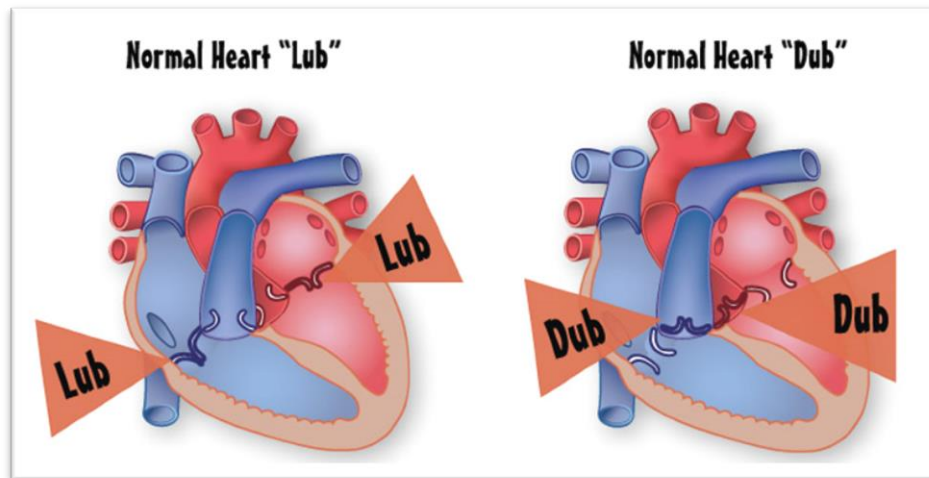
Periode pekerjaan jantung yaitu:

- a. Periode *systole*.
- b. Periode *diastole*.
- c. Periode istirahat.

2.2.8 Bunyi Jantung

Tahapan bunyi jantung:

- a. Bunyi pertama: lub.
- b. Bunyi kedua : dub.
- c. Bunyi ketiga: lemah dan rendah 1/3 jalan *diastolic* individu muda.
- d. Bunyi keempat: kadang-kadang dapat didengar segera sebelum bunyi pertama.



Gambar 2.2 Bagian Jantung Yang Bekerja Saat Bunyi “lub” dan “dub.

2.2.2 Stetoskop

a. Sejarah Stetoskop

Stetoskop berasal dari bahasa Yunani yaitu *stethos* yang berarti dada dan *skopeen* yang berarti memeriksa. Stetoskop ditemukan di Perancis pada tahun 1816 oleh ilmuwan Rene Theophile Hyacinthe Laennec. Pada awalnya stetoskop digunakan untuk *auskultasi*, yaitu pemeriksaan secara keseluruhan suara dalam tubuh dalam bidang medis. Laennec meletakkan langsung telinganya pada dada pasiennya. Cara seperti ini tentu mengganggu pasien. Selain itu, suara yang terdengar juga kurang jelas karena hanya menggunakan satu telinga. Kemudian berkembang menggunakan kertas yang digulung menyerupai mikrofon besar dan akhirnya berlanjut menggunakan tabung kayu kosong. Stetoskop pertama dibuat dari kayu berbentuk tabung yang berlubang di tengahnya.

Stetoskop adalah sebuah alat medis akustik untuk memeriksa suara dalam tubuh. Alat ini banyak digunakan untuk mendengar suara jantung dan pernafasan

serta untuk mendengar *intestine* dan aliran darah dalam arteri dan vena. Alat ini juga digunakan oleh mekanik untuk mengisolasi suara tertentu dari mesin untuk diagnosa.

2.2.3 Jenis Stetoskop

Saat ini dikenal dua jenis stetoskop, yaitu stetoskop akustik dan stetoskop elektronik.

a. Stetoskop Akustik

Stetoskop ini paling banyak digunakan dimana alat ini beroperasi dengan menyalurkan suara dari bagian dada, melalui tabung kosong berisi-udara, ke telinga pendengar. Bagian *chestpiece* biasanya terdiri dari dua sisi yang dapat diletakkan di badan pasien untuk memperjelas suara yaitu sebuah diafragma (disk plastik) atau bell (mangkok kosong). Bila diafragma diletakkan pada pasien, suara tubuh menggetarkan diafragma, menciptakan tekanan gelombang akustik yang berjalan sampai ke tube ke telinga pendengar. Bila bel diletakkan di tubuh pasien, getaran kulit secara langsung memproduksi gelombang tekanan akustik yang berjalan ke telinga pendengar. Bel menyalurkan suara frekuensi rendah, sedangkan diafragma menyalurkan frekuensi suara yang lebih tinggi. Stetoskop dua sisi ini diciptakan oleh Rappaport dan Sprague pada awal abad ke-20. Permasalahan dengan stetoskop akustik adalah tingkatan suara sangat rendah sehingga menyebabkan sulitnya membuat diagnosa. Pada stetoskop akustik, suara diperkeras "secara alami" dalam tabung kecil mirip kabel yang terbuat dari plastik. Bagian ujung stetoskop berupa membran yang akan bervibrasi jika

terkena gelombang suara. Adapun kedua ujungnya disalurkan ke dalam telinga.

b. Stetoskop Elektronik

Untuk mengatasi permasalahan dari stetoskop akustik tersebut maka terciptalah suatu stetoskop elektronik yang mengatasi tingkatan suara yang rendah dengan cara memperkuat suara tubuh menggunakan rangkaian penguat. Stetoskop elektronik lebih modern karena menggunakan energi listrik (baterai) untuk memperkuat gelombang suara.

2.2.4 Penguat Operasional (*Op-amp*)

a. Definisi Penguat Operasional (*Op-amp*)

Penguat operasional (*op-amp*) adalah sebuah penguat instan yang bisa langsung dipakai untuk banyak aplikasi penguatan. Sebuah *op-amp* biasanya berupa *IC* (*Integrated Circuit*). Pengemasan *op-amp* dalam *IC* bermacam-macam, ada yang berisi satu *op-amp* (contoh : 741), dua *op-amp* (4558, *LF356*), empat *op-amp* (contoh = *LM324*, *TL084*), dll.

Penguat operasional atau disingkat *op-amp* adalah merupakan satu penguat differensial berperolehan sangat tinggi yang terterkopel *DC* langsung yang dilengkapi dengan umpan. Oleh karena itu, penguat operasional lebih banyak digunakan dengan *loop* tertutup daripada dalam lingkaran terbuka.

Sebuah blok *op-amp* yang mempunyai berbagai tipe dalam bentuk *IC*. Dalam bentuk paket praktis *IC* seperti tipe 741 hanya berharga beberapa ribu rupiah. *Op-amp* memiliki masukan tak membalik v_+ (*non-inverting*),

masukan membalik v^- (*inverting*) dan keluaran v_o . Jika isyarat masukan dihubungkan dengan masukan membalik (v^-), maka pada daerah frekuensi tengah isyarat keluaran akan “berlawanan *fase*” (berlawanan tanda dengan isyarat masukan). Sebaliknya, jika isyarat masukan dihubungkan dengan masukan tak membalik (v^+), maka isyarat keluaran akan “*sefase*”. Sebuah *op-amp* biasanya memerlukan catu daya ± 15 V. Dalam menggambarkan rangkaian hubungan catu daya ini biasanya dihilangkan. Beberapa sifat ideal dari *op-amp* adalah sebagai berikut:

1. Penguat lingkaran terbuka tak berhingga atau $A_{v, \text{ib}} = \infty$
2. Hambatan keluaran lingkaran terbuka adalah nol atau $R_{o, \text{ib}} = 0$
3. Hambatan masukan lingkaran terbuka tak berhingga atau $R_{i, \text{ib}} = \infty$
4. Lebar pita tak berhingga atau $\Delta f = f_2 - f_1 = \infty$
5. Nisbah penolakan modulus bersama (*CMRR*) = ∞

b. Karakteristik *Op-amp*

Keuntungan dari pemakaian penguat operasional ini adalah karakteristiknya yang mendekati ideal sehingga dalam merancang rangkaian yang menggunakan penguat ini lebih mudah dan juga karena penguat ini bekerja pada tingkatan yang cukup dekat dengan karakteristik kerjanya secara teoritis. Dari sudut sinyal sebuah penguat operasional mempunyai tiga terminal, yaitu dua terminal masukan dan satu terminal keluaran.

Dapat dilihat bahwa terminal 1 dan 2 adalah terminal masukan dan terminal 3 adalah terminal keluaran. Kebanyakan penguat operasional membutuhkan catu daya *DC* dengan dua polaritas untuk dapat beroperasi.

Terminal 4 disambungkan ke tegangan positif (+V) dan terminal 5 disambungkan ke tegangan negatif (-V). Karakteristik utama sebuah penguat operasional yang ideal adalah:

1. Impedansi masukan tak terhingga. Penguat yang ideal diharapkan tidak menarik arus masukan, artinya tidak ada arus yang masuk kedalam terminal 1 maupun 2 ($I_1 = I_2 = 0$).
2. Impedansi keluaran sama dengan nol. Terminal 3 merupakan keluaran penguat operasional, idealnya diharapkan bertindak sebagai terminal keluaran sebuah sumber tegangan ideal. Tegangan antara terminal 3 dengan *ground* akan selalu sama dengan A, dimana A adalah faktor penguatan sebuah penguat operasional.
3. Penguatan *loop* terbuka tak terhingga. Apabila dioperasikan pada *loop* terbuka (tidak ada umpan balik dari keluaran ke masukan), maka sebuah penguat operasional ideal mempunyai *gain* (penguatan) yang besarnya tak terhingga..

2.2.7 *Mic/line*

Switch tekan ini untuk mengubah sirkuit *gain control*. Tergantung apakah yang menjadi input adalah *mic*, *effect return* atau *tape deck/CD*. Pada banyak *mixing console* terdapat terminal *input* yang terpisah antara *mic* dan *line input* pada *channel* yang sama. *Input mic* biasanya menggunakan tipe konektor *balance* 3 pin *XLR* atau kadang biasa disebut *jack canon*. Sedangkan *line input* menggunakan *jack* seperti yang biasa dipakai *jack* gitar. Hal ini memungkinkan untuk mencolokkan dua *input* yang berbeda dalam satu

channel dan *switch* ini untuk mengaktifkan salah satu *input* yang kita inginkan di antara keduanya. Sebagai contoh, dapat mencolokkan *effect return* dengan *gain* yang diset rendah pada *mic input* kemudian mencolokkan lagi *tape deck* pada *line input channel* yang sama. Pada saat *band* sedang *show* dan *tape deck* tidak dibutuhkan, tinggal men-*switch* tombol tersebut pada posisi *mic*. Kemudian pada saat *band* telah selesai dan butuh *playback* musik dari *tape deck/CD*, tinggal men-*switch*nya pada posisi *line*. Ini bisa dilakukan untuk menghemat *channel*, khususnya apabila *console* yang digunakan tidak terlalu besar.

2.2.8 Loudspeaker

Fungsi *speaker* adalah mengubah gelombang listrik menjadi getaran suara. Proses pengubahan gelombang listrik/elektromagnet menjadi gelombang suara terjadi karena adanya aliran listrik arus *AC* audio dari penguat audio kedalam kumparan yang menghasilkan gaya magnet sehingga akan menggerakkan membran, kuat lemahnya arus listrik yang diterima, akan mempengaruhi getaran pada membran, bergetarnya membran ini menghasilkan gelombang bunyi yang dapat kita dengar. Macam-macam *loudspeaker*, yaitu:

- a. *Full Range* yaitu *loudspeaker* yang memiliki kemampuan mengeluarkan semua frekuensi audio dari yang tertinggi sampai terendah.
- b. *Woofers* yaitu *loudspeaker* khusus untuk mengeluarkan frekuensi rendah.
- c. *Middle* atau *midrange* yaitu *loudspeaker* khusus untuk mengeluarkan frekuensi menengah.
- d. *Tweeter* yaitu *loudspeaker* khusus untuk mengeluarkan frekuensi tinggi.

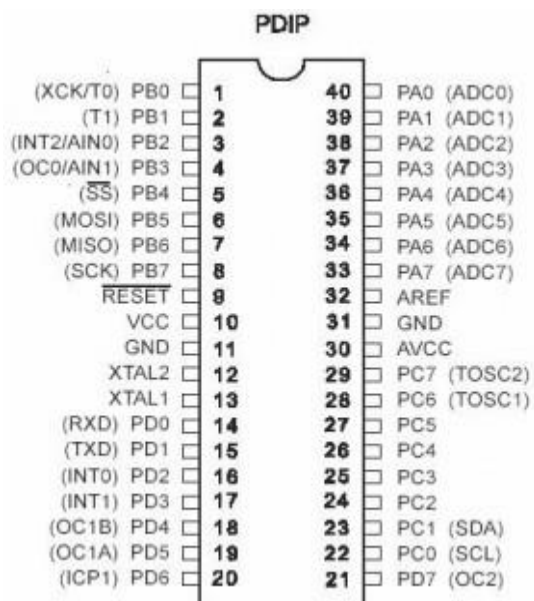
Nilai *loudspeaker* dinyatakan dalam 3 satuan yaitu sebagai berikut:

- Besarnya dinyatakan diameter (garis tengah) dinyatakan dalam *inchi*.
- Besarnya daya yang dapat dikeluarkan atau diterima dinyatakan dalam *watt*.
- Besarnya impedansi *spul* dinyatakan dalam *ohm*.

2.2.9 AT Mega8535

a. Konfigurasi Pin AT Mega8535

Konfigurasi pin AT Mega8535 ditunjukkan oleh Gambar 2.4.



Gambar 2.4 Konfigurasi Pin AT Mega8535

Secara umum konfigurasi dan fungsi pin AT Mega8535 dapat dijelaskan sebagai berikut:

- VCC*: Input sumber tegangan (+).
- GND*: Ground (-).

3. *Port A* (PA7 ... PA0) berfungsi sebagai *input* analog dari *ADC* (*Analog to Digital Converter*). *Port* ini juga berfungsi sebagai *port I/O* dua arah, jika *ADC* tidak digunakan.
4. *Port B* (PB7 ... PB0) berfungsi sebagai *port I/O* dua arah. *Port* PB5, PB6 dan PB7 juga berfungsi sebagai *MOSI*, *MISO* dan *SCK* yang dipergunakan pada proses *downloading*. Fungsi lain *port* ini selengkapnya bisa dibaca pada buku petunjuk "AVR AT Mega8535".
5. *Port C* (PC7 ... PC0) berfungsi sebagai *port I/O* dua arah. Fungsi lain *port* ini selengkapnya bisa dibaca pada buku petunjuk "AVR AT Mega8535".
6. *Port D* (PD7 ... PD0) berfungsi sebagai *port I/O* dua arah. *Port* PD0 dan PD1 juga berfungsi sebagai *RXD* dan *TXD*, yang dipergunakan untuk komunikasi serial. Fungsi lain *port* ini selengkapnya bisa dibaca pada buku petunjuk "AVR AT Mega8535".
7. *RESET*: *input reset*.
8. *XTAL1*: *input* ke *amplifier inverting osilator* dan *input* ke sirkuit *clock internal*.
9. *XTAL2*: *output* dari *amplifier inverting osilator*.
10. *AVCC*: *input* tegangan untuk *port A* dan *ADC*.
11. *AREF*: tegangan referensi untuk *ADC*.

b. Fitur Mikrokontroler AT Mega8535

Adapun kapabilitas detail dari AT Mega8535 adalah sebagai berikut:

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

c. Arsitektur AT Mega8535

Bagian-bagian AT Mega8535 adalah sebagai berikut :

1. Saluran *I/O* sebanyak 32 buah, yaitu *port A*, *port B*, *port C* dan *port D*.
2. *ADC* 8 *channel* 10 bit.
3. Tiga buah *timer/counter* dengan kemampuan pembanding.
4. *CPU* yang terdiri atas 32 buah *register*.
5. *Watchdog timer* dengan osilator *internal*.
6. *SRAM* sebesar 512 byte.
7. Memori *flash* sebesar 8 KB dengan kemampuan *read while write*.
8. *Interrupt internal* dan *eksternal*.
9. *Port* antarmuka *SPI* (*Serial Peripheral Interface*).
10. *EEPROM* sebesar 512 byte yang dapat diprogram saat operasi.
11. Antarmuka komparator analog.
12. *Port USART* untuk komunikasi serial.

2.2.10 LCD (*Liquid Crystal Display*)

LCD (Liquid Cristal Display) berfungsi untuk menampilkan karakter angka, huruf ataupun simbol dengan lebih baik dan dengan konsumsi arus yang rendah. Jenis *LCD* ada berbagai macam, dan yang paling sering digunakan adalah *LCD* karakter 16 x 2 yang maksudnya mempunyai lebar *display* 2 baris 16 kolom. Dengan 16 pin konektor, yang didefinisikan pada Gambar 2.5.:



Gambar 2.5. *LCD* karakter 2x16.

Tabel 2.1. Pin dan Fungsi *LCD* Karakter 2 x 16

<i>PIN</i>	<i>Name</i>	<i>Function</i>
1	VSS	Ground voltage
2	VCC	+5V
3	VEE	Contrast voltage
4	RS	Register Select 0 = Instruction Register 1 = Data Register
5	R/W	Read/ Write, to choose write or read mode 0 = write mode 1 = read mode
6	E	Enable 0 = start to lacht data to LCD character 1= disable
7	DB0	LSB
8	DB1	-
9	DB2	-
10	DB3	-
11	DB4	-
12	DB5	-
13	DB6	-

14	<i>DB7</i>	<i>MSB</i>
15	<i>BPL</i>	<i>Back Plane Light</i>
16	<i>GND</i>	<i>Ground voltage</i>

Display karakter pada *LCD* diatur oleh pin EN, RS dan RW. Jalur EN dinamakan *enable*. Jalur ini digunakan untuk memberitahu *LCD* bahwa anda sedang mengirimkan sebuah data. Untuk mengirimkan data ke *LCD* maka melalui program EN harus dibuat logika *low* “0” dan *set* pada dua jalur kontrol yang lain RS dan RW. Ketika dua jalur yang lain telah siap, set EN dengan logika “1” dan tunggu untuk sejumlah waktu tertentu (sesuai dengan *datasheet* dari *LCD* tersebut) dan berikutnya set EN ke logika *low* “0” lagi.

Jalur RS adalah jalur *register select*. Ketika RS berlogika *low* “0”, data akan dianggap sebagai sebuah perintah atau instruksi khusus (seperti *clear screen*, posisi kursor, dll). Ketika RS berlogika *high* “1”, data yang dikirim adalah data *text* yang akan ditampilkan pada *display LCD*. Sebagai contoh, untuk menampilkan huruf “T” pada layar *LCD* maka RS harus diubah logika *high* “1”.

Jalur RW adalah jalur kontrol *read/write*. Ketika RW berlogika *low* (0), maka informasi pada data akan dituliskan pada layar *LCD*. Ketika RW berlogika *high* “1”, maka program akan melakukan pembacaan memori dari *LCD*. Sedangkan pada aplikasi umum pin RW selalu diberi logika *low* “0”.

DDRAM (Display Data RAM) digunakan untuk menyimpan karakter yang akan ditampilkan. Semua teks yang kita tuliskan ke modul *LCD* adalah disimpan di dalam memori ini, dan modul *LCD* secara berurutan membaca memori ini untuk menampilkan teks ke modul *LCD* itu sendiri.

Display	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	...
Line 1	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	...
Line 2	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F	...

Gambar 2.8. Lokasi Memori *Display LCD* Karakter 2 x 16.

Pada peta memori tersebut, daerah yang berwarna biru (00 s/d 0F dan 40 s/d 4F) adalah *display* yang tampak. Sebagaimana yang anda lihat, jumlahnya sebanyak 16 karakter per baris dengan dua baris. Angka pada setiap kotak adalah alamat memori yang bersesuaian dengan posisi dari layar. Demikianlah karakter pertama di sudut kiri atas adalah menempati alamat 00h. Posisi karakter berikutnya adalah alamat 01h dan seterusnya.

Akan tetapi, karakter pertama dari baris ke-2 sebagaimana yang ditunjukkan pada peta memori adalah pada alamat 40h. Demikianlah kita perlu untuk mengirim sebuah perintah ke *LCD* untuk mengatur letak posisi kursor pada baris dan kolom tertentu. *Instruksi set* posisi kursor adalah 80h. Untuk ini kita perlu menambahkan alamat lokasi dimana kita berharap untuk menempatkan kursor. Sebagai contoh, kita ingin menampilkan kata "*World*" pada baris ke-2 pada posisi kolom ke-10. Sesuai peta memori, posisi karakter pada kolom 11 dari baris ke dua, mempunyai alamat 4Ah, sehingga sebelum kita tulis kata "*World*" pada *LCD*, kita harus mengirim instruksi set posisi kursor, dan perintah untuk instruksi ini adalah 80h ditambah dengan alamat $80h+4Ah = 0Cah$. Sehingga dengan mengirim perintah Cah ke *LCD*, akan menempatkan kursor pada baris kedua dan kolom ke 11 dari *DD*.