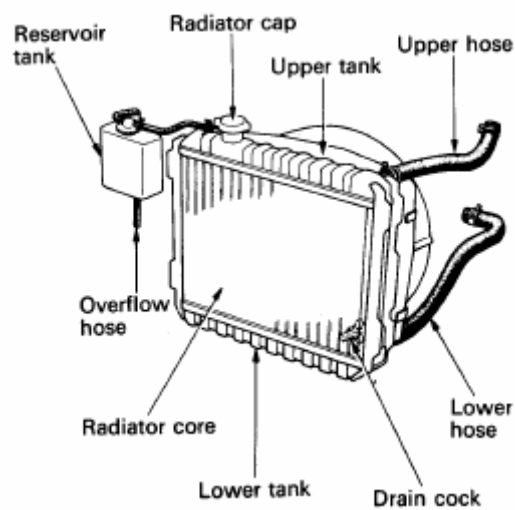


## BAB II TINJAUAN PUSTAKA

### 2.1 Radiator

Radiator adalah alat penukar panas yang digunakan untuk memindahkan energi panas dari satu media ke media lainnya yang tujuannya untuk mendinginkan maupun memanaskan dengan media berupa air. (H.M. Iwan Gayo, 2007: 39).



**Gambar 2.1 Radiator**

#### 2.1.1 Bagian – Bagian Radiator

##### 1. *Radiator cap* atau tutup radiator

Sesuai dengan namanya, fungsi pada bagian tersebut adalah sebagai penutup agar tekanan air serta udara yang berasal dari mesin utama tidak sampai keluar yang dapat menyebabkan berkurangnya air di dalam radiator.

##### 2. *Upper tank* atau tangki atas

*Upper tank* adalah bagian atas radiator yang terbuat dari 2 bahan. yang pertama berbahan dari kuningan dan yang kedua berbahan dari plastik composit. *Upper tank* ini berfungsi untuk transit air panas yang berasal dari blok mesin.

##### 3. *Lower tank* atau tangki bawah

*Lower tank* adalah bagian paling bawah dari radiator yang terbuat sama dengan *upper tank* yaitu kuningan dan plastik composit. *Lower tank* ini fungsinya

untuk menampung air yang sudah didinginkan dari core radiator dan tempat transitair yang akan masuk ke blok mesin.

#### **4. Radiator core atau bagian tengah radiator**

*Radiator Core* bisa disebut juga sarang radiator. *Radiator Core* ini terbuat dari 2 jenis juga, yaitu ada yang berbahan tembaga dan aluminium. *Radiator Core* ini yang memegang peranan penting saat proses pendinginan. *Radiator Core* ini terdiri dari susunan pipa pipih yang di rangkai dengan plat tipis yang berfungsi untuk melepas panas. fungsi dari *Radiator Core* adalah memecah volume air menjadi kecil-kecil melewati pipa pipih tadi agar air lebih mudah dingin saat terkena angin dari hembusan kipas pendingin.

#### **5. Selang radiator**

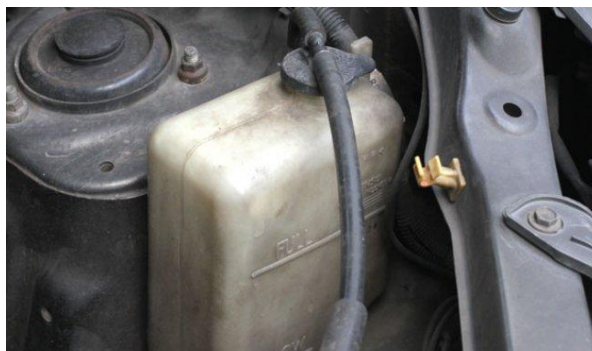
Selang radiator berfungsi sebagai penghubung antara radiator dan blok mesin. Ada dua selang di radiator, *Upper hose* berfungsi mengalirkan air panas dari mesin ke radiator. Sedangkan *lower hose* untuk menyalurkan air yang sudah didinginkan kembali ke mesin.

#### **6. Drain cock atau kran pembuang**

*Drain cock* atau kran pembuang berfungsi untuk membuang air yang ada di dalam radiator.

#### **7. Reservoir tank atau tangki cadangan**

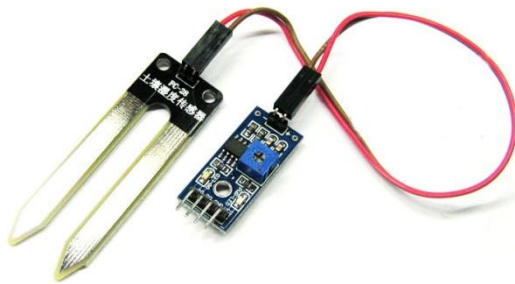
*Reservoir Tank* dihubungkan ke radiator melalui selang *overflow*. Reservoir Tank ini berfungsi untuk menjaga agar volume air pendingin selalu stabil.



**Gambar 2.2 Reservoir tank atau tanki cadangan**

## 2.2 Soil Moisture Sensor

Soil Moisture sensor adalah sensor yang dapat mendeteksi kelembaban tanah. Soil Moisture sensor adalah sensor yang dapat mendeteksi kelembaban tanah. (Riyanto Sigit, 2010: 27)



**Gambar 2.3 Soil Moisture Sensor**

Sensor ini terdiri dari dua probe untuk melewatkan arus melalui tanah, kemudian membaca resistansinya untuk mendapatkan tingkat kelembaban. Semakin banyak air membuat tanah lebih mudah menghantarkan listrik kepada sensor, sedangkan tanah yang kering sangat sulit menghantarkan listrik. Dari prinsip kerja sensor tersebut air radiator dapat diukur oleh sensor ini dengan tambahan elektroda di kedua probe. Elektroda adalah konduktor yang digunakan untuk bersentuhan dengan bagian atau media non-logam dari sebuah sirkuit (misal semikonduktor, elektrolit atau vakum).

## 2.3 Mikrokontroler

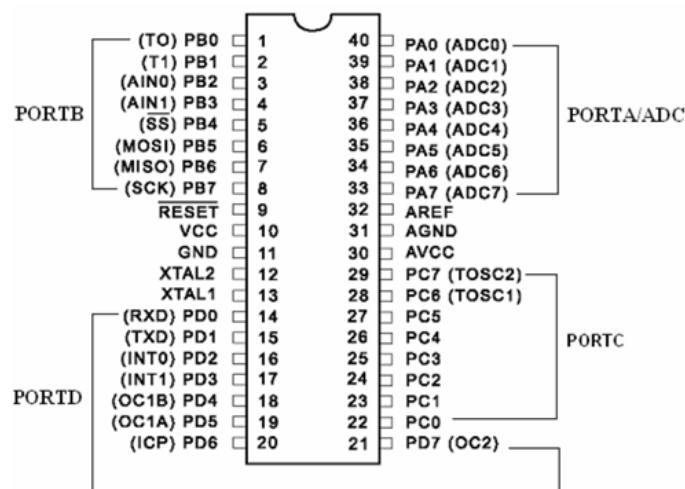
Mikrokontroler adalah mikroprosesor yang dirancang khusus untuk aplikasi kontrol, dan dilengkapi dengan ROM, RAM dan fasilitas I/O pada satu chip. Mikrokontroler *AVR* (*Alf and Vegard's Risc processor*) standar memiliki arsitektur 8 bit, dimana semua instruksi dikemas dalam kode 16-bit, dan sebagian besar instruksi dieksekusi dalam satu siklus clock. *AVR* berteknologi *RISC* (*Reduced Instruction Set Computing*), sedangkan *MCS51* berteknologi *CISC* (*Complex Instruction Set Computing*). (Budiarmo, 2009:5)

AVR dapat dikelompokkan menjadi empat kelas yaitu *ATTiny*, *AT90xx*, *ATMega*, dan *AT86RFxx*. Pada dasarnya yang membedakan masing – masing kelas adalah memori, peripheral, dan fungsinya.

### 2.3.1 Mikrokontroler ATmega8535

Mikrokontroler AVR Atmega 8535 AVR merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur *RISC (Reduced Instruction Set Computer)*. Hampir semua instruksi dieksekusi dalam satu siklus *clock*. AVR mempunyai 32 *register general-purpose*, *timer/counter* fleksibel dengan mode compare, interrupt internal dan eksternal, serial UART, *programmable Watchdog Timer*, dan *mode power saving*. Beberapa diantaranya mempunyai ADC dan PWM internal. AVR juga mempunyai *In- System Programmable Flash on-chip* yang memungkinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI. ATmega8535 adalah mikrokontroler CMOS 8-bit daya-rendah berbasis arsitektur RISC yang ditingkatkan. Kebanyakan instruksi dikerjakan pada satu siklus clock, ATmega8535 mempunyai throughput mendekati 1 MIPS per MHz membuat disainer sistem untuk mengoptimasi konsumsi daya versus kecepatan proses.

### 2.3.2 Arsitektur ATmega8535



Gambar 2.4 Konfigurasi Pin ATmega8535

Mikrokontroler AVR ATmega memiliki 40 *pin* dengan 32 *pin* diantaranya digunakan sebagai *port paralel*. Satu *port paralel* terdiri dari 8 *pin*, sehingga jumlah *port* pada mikrokontroler adalah 4 *port*, yaitu *port A*, *port B*, *port C* dan *port D*. Sebagai contoh adalah *port A* memiliki *pin* antara *port A.0* sampai dengan *port A.7*, demikian selanjutnya untuk *port B*, *port C*, *port D*. Diagram *pin* mikrokontroler dapat dilihat pada gambar berikut:

**Tabel 2.1 Penjelasan pin pada mikrokontroler ATmega8535**

Vcc	Tegangan suplai (5 volt)
GND	Ground
RESET	Input reset level rendah, pada <i>pin</i> ini selama lebih dari panjang pulsa <i>minimum</i> akan menghasilkan reset walaupun <i>clock</i> sedang berjalan. RST pada <i>pin</i> 9 merupakan reset dari AVR. Jika pada <i>pin</i> ini diberi masukan <i>low</i> selama minimal 2 <i>machine cycle</i> maka sistem akan di-reset
XTAL 1	Input penguat osilator inverting dan input pada rangkaian operasi <i>clock internal</i>
XTAL 2	Output dari penguat osilator inverting
Avcc	Pin tegangan suplai untuk <i>port A</i> dan ADC. Pin ini harus dihubungkan ke Vcc walaupun ADC tidak digunakan, maka <i>pin</i> ini harus dihubungkan ke Vcc melalui <i>low pass filter</i>
Aref	<i>pin</i> referensi tegangan analog untuk ADC
AGND	<i>pin</i> untuk analog ground. Hubungkan kaki ini ke GND, kecuali jika board memiliki analog ground yang terpisah

Berikut ini adalah penjelasan dari *pin* mikrokontroler ATmega8535 menurut *port*-nya masing-masing:

### 1. Port A

Pin33 sampai dengan *pin* 40 merupakan *pin* dari *port A*. Merupakan 8 bit *directional port I/O*. Setiap *pin*-nya dapat menyediakan *internal pull-up resistor*

(dapat diatur per *bit*). *Output buffer port A* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. *Data Direction Register port A* (DDRA) harus di-*setting* terlebih dahulu sebelum *port A* digunakan. *Bit-bit* DDRA diisi 0 jika ingin memfungsikan *pin-pin port A* yang disesuaikan sebagai *input*, atau diisi 1 jika sebagai *output*.

## **2. Port B**

*Pin 1* sampai dengan *pin 8* merupakan *pin* dari *port B*. Merupakan 8 bit *directional port I/O*. Setiap *pin*-nya dapat menyediakan *internal pull-up resistor* (dapat diatur per *bit*). *Output buffer port B* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. *Data Direction Register port B* (DDRB) harus di-*setting* terlebih dahulu sebelum *port B* digunakan. *Bit-bit* DDRB diisi 0 jika ingin memfungsikan *pin-pin port B* yang disesuaikan sebagai *input*, atau diisi 1 jika sebagai *output*.

---

## **3. Port C**

*Pin 22* sampai dengan *pin 29* merupakan *pin* dari *port C*. *Port C* sendiri merupakan *port input* atau *output*. Setiap *pin*-nya dapat menyediakan *internal pull-up resistor* (dapat diatur per *bit*). *Output buffer port C* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. *Data Direction Register port C* (DDRC) harus di-*setting* terlebih dahulu sebelum *port C* digunakan. *Bit-bit* DDRC diisi 0 jika ingin memfungsikan *pin-pin port C* yang disesuaikan sebagai *input*, atau diisi 1 jika sebagai *output*.

## **4. Port D**

*Pin 14* sampai dengan *pin 20* merupakan *pin* dari *port D*. Merupakan 8 bit *directional port I/O*. Setiap *pin*-nya dapat menyediakan *internal pull-up resistor* (dapat diatur per *bit*). *Output buffer port D* dapat memberi arus 20 mA dan dapat mengendalikan *display LED* secara langsung. *Data Direction Register port D* (DDRD) harus di-*setting* terlebih dahulu sebelum *port D* digunakan. *Bit-bit*

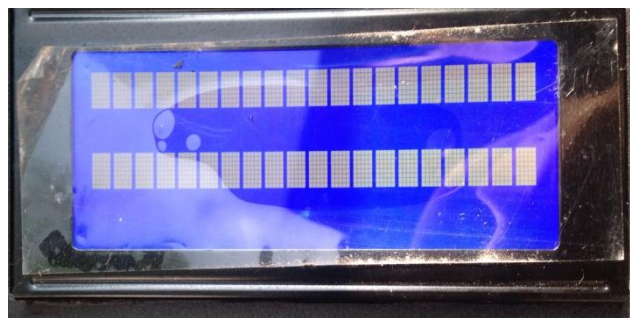
DDRD diisi 0 jika ingin memfungsikan *pin-pin port D* yang disesuaikan sebagai *input*, atau diisi 1 jika sebagai *output*.

## 2.6 Liquid Crystal Display

LCD (Liquid Crystal Display atau dapat di bahasa Indonesia-kan sebagai Tampilan Kristal Cair) adalah suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama. (Saputra, 2014:3)

LCD bisa memunculkan gambar atau tulisan dikarenakan terdapat banyak sekali titik cahaya (piksel) yang terdiri dari satu buah kristal cair sebagai sebuah titik cahaya. Walau disebut sebagai titik cahaya, namun kristal cair ini tidak memancarkan cahaya sendiri. Sumber cahaya di dalam sebuah perangkat LCD adalah lampu neon berwarna putih di bagian belakang susunan kristal cair tadi.

Titik cahaya yang jumlahnya puluhan ribu bahkan jutaan inilah yang membentuk tampilan citra. Kutub kristal cair yang dilewati arus listrik akan berubah karena pengaruh polarisasi medan magnetik yang timbul dan oleh karenanya akan hanya membiarkan beberapa warna diteruskan sedangkan warna lainnya tersaring. Menampilkan karakter untuk membantu menginformasikan proses dan kontrol yang terjadi dalam suatu program.



**Gambar 2.5 LCD 4x16**

LCD sudah dilengkapi perangkat kontrol sendiri yang menyatu dengan LCD, maka kita mengikuti aturan standar yang telah disimpan dalam pengontrolan tersebut.

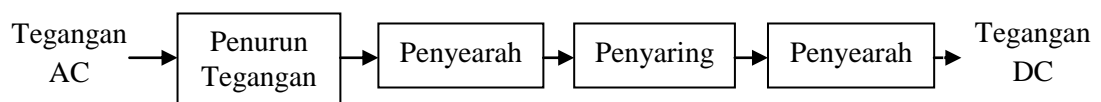
Fungsi dari masing-masing *Pin* LCD adalah sebagai berikut :

### **Tabel 2.2 Pin-pin pada LCD**

Pin	Nama	Fungsi
1	VSS	<i>Ground Voltage</i>
2	VCC	+5V
3	VEE	<i>Contrast Voltage</i>
4	RS	<i>Register Select</i> 0 = <i>Instruction Register</i> 1 = <i>Data Register</i>
5	R/W	<i>Read or Write mode</i> 0 = <i>Write mode</i> 1 = <i>Read mode</i>
6	E	<i>Enable</i> 0 = <i>start to lacht data to LCD character</i> 1 = <i>disable</i>
7	DB0	Data bit ke-0 (LSB)
8	DB1	Data bit ke-1
9	DB2	Data bit ke-2
10	DB3	Data bit ke-3
11	DB4	Data bit ke-4
12	DB5	Data bit ke-5
13	DB6	Data bit ke-6
14	DB7	Data bit ke-7
15	BPL	<i>Back Plane Light</i>
16	GND	<i>Ground Voltage</i>

## 2.5 Catu Daya

Catu daya diperlukan oleh setiap rangkaian elektronika untuk memenuhi kebutuhan arus dan tegangan sehingga rangkaian itu dapat berfungsi. Secara blok diagram, catu daya dapat ditunjukkan pada gambar 2.6 berikut.





### **Gambar 2.6 Diagram blok catu daya**

Tegangan bolak – balik sebesar 220 Volt yang berasal dari jala – jala PLN diturunkan tegangannya dengan menggunakan transformator. Tegangan bolak – balik ini kemudian disearahkan oleh rangkaian penyearah (penyearah gelombang penuh maupun penyearah setengah gelombang). Tegangan keluaran dari penyearah merupakan tegangan searah yang berdenyut atau masih terdapat riak. Riak dapat diperkecil dengan melewati tegangan DC hasil penyearahan ke rangkaian penyaring (filter), sehingga diperoleh tegangan searah murni. Regulasi tegangan diperlukan untuk menjaga besar tegangan keluaran tetap stabil, tidak terpengaruh oleh perubahan – perubahan yang terjadi. Misalnya perubahan besarnya beban yang terpasang atau perubahan tegangan yang terjadi pada jala – jala PLN.

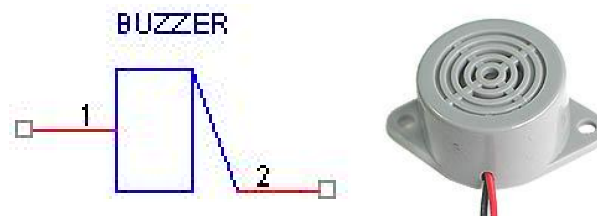
### **2.6 Relay**

Relay merupakan komponen output yang paling sering digunakan pada beberapa peralatan elektronika dan di berbagai bidang lainnya. Relay berfungsi untuk menghubungkan atau memutuskan aliran arus listrik yang dikontrol dengan memberikan tegangan dan arus tertentu pada koilnya. Ada 2 macam relay berdasarkan tegangan untuk menggerakkan koilnya, yaitu AC dan DC. Pada perangkat yang dibuat digunakan relay DC dengan tegangan koil 12VDC, arus yang diperlukan sekitar 20 sampai dengan 30 mA (Setiawan, 2011:21).

Pada dasarnya relay adalah sebuah kumparan yang terdiri arus listrik sehingga kumparan mempunyai sifat sebagai magnet. Magnet sementara tersebut digunakan untuk menggerakkan suatu system saklar yang terbuat dari logam sehingga pada saat relay dialiri arus listrik maka kumparan akan terjadi kemagnetan dan menarik logam tersebut, saat arus listrik diputuskan maka logam akan kembali pada posisi semula.

## 2.7 Buzzer

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Buzzer terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. Buzzer biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (alarm). (Setiawan, 2011:21).



**Gambar 2.7** (a) Simbol buzzer, (b). Bentuk Buzzer

## 2.8 Bahasa Pemrograman C

Bahasa pemrograman C merupakan salah satu bahasa pemrograman komputer. Dibuat pada tahun 1972 oleh Dennis Ritchie untuk Sistem Operasi Unix di Bell Telephone Laboratories. Bahasa C mempunyai kemampuan lebih dibanding dengan bahasa pemrograman lain. Bahasa C merupakan bahasa pemrograman yang bersifat portable, yaitu suatu program yang dibuat dengan bahasa C pada suatu komputer akan dapat dijalankan pada komputer lain dengan sedikit (atau tanpa) ada perubahan yang berarti. (Andrianto, 2013: 7)

Bahasa C merupakan bahasa yang biasa digunakan untuk keperluan pemrograman sistem, antara lain membuat :

1. Assembler
2. Interpreter
3. Compiler
4. Sistem Operasi

5. Program bantu (utility)
6. Editor
7. Paket program aplikasi

Bahasa C digolongkan sebagai bahasa tingkat menengah (medium level language). Penggolongan ini bukan berarti bahasa C kurang ampuh atau lebih sulit dibandingkan dengan bahasa tingkat tinggi (high level language – seperti Pascal, Basic, Fortran, Java, dan lain-lain). Namun untuk menegaskan bahwa bahasa C bukanlah bahasa yang berorientasi pada mesin, yang merupakan ciri dari bahasa tingkat rendah (low level language) yaitu bahasa mesin dan assembly. Pada kenyataannya, bahasa C mengkombinasikan elemen dalam bahasa tingkat tinggi dan bahasa tingkat rendah, yaitu kemudahan dalam membuat program yang ditawarkan pada bahasa tingkat tinggi dan kecepatan eksekusi dari bahasa tingkat rendah.

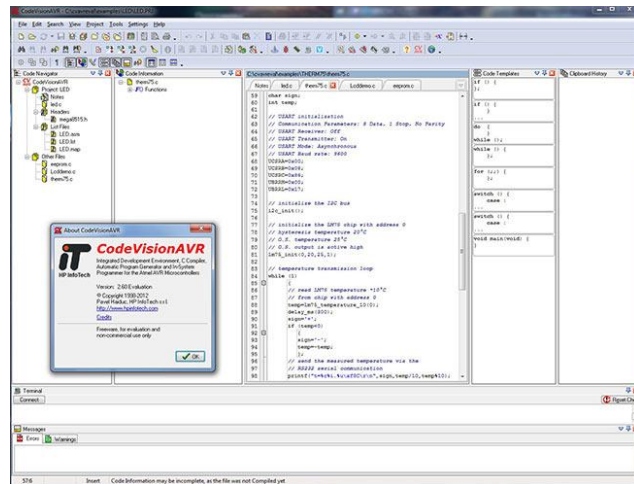
## **2.9 CodeVisionAVR**

Mikrokontroler sekarang ini telah umum dipergunakan. Mulai dari penggunaan untuk kontrol sederhana sampai kontrol yang cukup kompleks, mikrokontroler dapat berfungsi jika telah diisi sebuah program, pengisian program ini dapat dilakukan menggunakan compiler yang selanjutnya di download ke dalam mikrokontroler menggunakan downloader. Salah satu compiler program yang umum digunakan sekarang ini adalah CodeVisionAVR yang menggunakan bahasa pemrograman C.

CodeVisionAVR C Compiler adalah software yang digunakan untuk membuat program mikrokontroler AVR dalam bahasa C. Program tersebut kemudian diterjemahkan oleh CodeVisionAVR C menjadi kode heksadesimal yang akan didownload ke dalam chip mikrokontroler AVR. (Hendawan Soebhakti, 2009:3)

CodeVisionAVR mempunyai suatu keunggulan dari compiler lain, yaitu adanya codewizard, fasilitas ini memudahkan kita dalam inisialisasi mikrokontroler yang akan kita gunakan, codevision telah menyediakan konfigurasi yang bisa diatur pada masing – masing chip mikrokontroler yang akan

kita gunakan, sehingga kita tidak perlu melihat datasheet untuk sekedar mengkonfigurasi mikrokontroler. Tampilan utama CodeVisionAVR ditunjukkan pada gambar berikut.





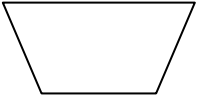
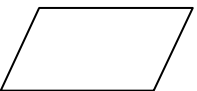
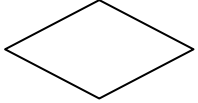
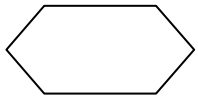
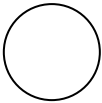
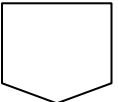
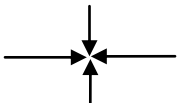
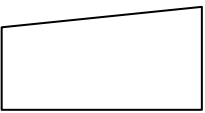
**Gambar 2.8 Tampilan utama CodeVisionAVR**



**2.10 Flowchart**

Hartono (662) mengatakan “Bagan alir program (program *flowchart*) adalah suatu bagan yang menggambarkan arus logika dari data yang akan diproses dalam suatu program dari awal sampai akhir”. Bagan alir program merupakan alat yang berguna bagi *programmer* untuk mempersiapkan program yang rumit. Bagan alir terdiri dari symbol-symbol yang mewakili fungsi-fungsi langkah program dan garis alir (*flow line*) menunjukkan urutan dari simbol-simbol yang akan dikerjakan.

**Tabel 2.3 Simbol - Simbol Flowchart**

No	Simbol	Fungsi
1	Terminal 	Simbol untuk memulai dan mengakhiri suatu program

2	<p>Proses</p> 	Simbol untuk menyatakan suatu tindakan (proses) yang dilakukan oleh komputer
3	<p>Manual Operator</p> 	Simbol untuk menyatakan suatu tindakan (proses) yang tidak dilakukan oleh komputer
4	<p>Input – Output</p> 	Simbol untuk menyatakan proses input atau output tanpa tergantung jenis peralatannya
5	<p>Decision</p> 	Simbol untuk menunjukkan suatu kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban : ya / tidak
6	<p>Predefined Process</p> 	Simbol untuk menyatakan penyediaan tempat penyimpanan suatu pengolahan didalam storage
7	<p>Connector</p> 	Simbol untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang sama
8	<p>Off Line Connector</p> 	Simbol untuk menyatakan sambungan dari proses ke proses lainnya dalam halaman yang berbeda
9	<p>Arus atau Flow</p> 	Garis untuk menghubungkan arah tujuan simbol flowchart yang satu dengan yang lainnya
10	<p>Manual Input</p> 	Simbol untuk memasukkan data secara manual dengan menggunakan on-line keyboard

11	<p>Punched Card</p> 	Simbol untuk menyatakan input berasal dari kartu atau output ditulis ke kartu
12	<p>Document</p> 	Simbol untuk mencetak keluaran dalam bentuk dokumen (melalui printer)