

BAB II

TINJAUAN PUSTAKA

2.1 *Cooling Pad*

Cooling Pad adalah sebuah alat untuk menjaga suhu laptop tetap dingin. *Cooling pad* ini mengandalkan kipas yang meniupkan udara segar di bagian bawah dan body luar sebuah notebook. Hembusan angin dari kipas ini difungsikan sebagai pendingin di sekitar laptop sehingga laptop dapat bekerja pada suhu lebih rendah.

Penemu dan perancang pertama *cooling pad* ini adalah Rakesh Bhatia. Orang yang berkebangsaan India ini merancang untuk Intel Corporation dan mengajukan paten pada tanggal 8 Januari 1999.



Gambar 2.1 *Cooling Pad*

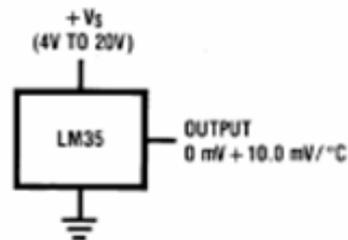
2.2 Suhu

(Esvandiari, 2006) Suhu adalah ukuran panas atau dinginnya suatu benda. Definisi yang lebih tepat menyatakan suhu adalah ukuran kelajuan gerak partikel-partikel dalam suatu benda atau ukuran kinetik rata-rata partikel dalam suatu benda.

Dalam kehidupan sehari-hari dalam mengukur suhu, masyarakat cenderung menggunakan indera peraba. Namun, dalam dunia modern saat ini, dalam mengukur suatu suhu sudah dapat dilakukan dengan cara yang mudah, yaitu dengan menggunakan sensor. Salah satu sensor suhu yang sering digunakan adalah IC LM35.

2.2.1 IC LM35

(AF. Nasution, 2011) Untuk mendeteksi suhu digunakan sebuah sensor suhu LM 35 yang dapat dikalibrasikan langsung. LM 35 ini difungsikan sebagai basic temperature sensor seperti pada gambar 2.2



Gambar 2.2 LM 35 Basic *Temperature* Sensor

Sensor LM35 memiliki 3 buah pin, yang masing-masing jika nampak dari depan memiliki fungsi yaitu pin 1 berfungsi sebagai sumber tegangan kerja dari sensor LM35, pin 2 atau tengah digunakan sebagai tegangan keluaran atau V_{out} dan pin 3 berfungsi sebagai ground. V_{out} pada sensor LM35 memiliki tegangan keluaran yang terskala linier terhadap suhu terukur, yaitu 10mV per 1°C . Jadi, jika $V_{out} = 530\text{mV}$, maka suhu terukur adalah 53°C .

Fitur LM35 adalah sebagai berikut :

1. *Centigrade* atau *celcius*
2. Sensitivitas $10\text{ mV}/^{\circ}\text{C}$
3. Akurasi $0,5\text{ }^{\circ}\text{C}$ pada suhu 25°C
4. Handal dalam pengukuran jarak jauh
5. Tegangan operasi 4 s/d 30 V
6. Konsumsi arus $60\text{ }\mu\text{A}$
7. *Self heating* rendah $0,08^{\circ}\text{C}$
8. *Non linieritas* $\pm 0,25^{\circ}\text{C}$
9. Impedansi keluaran $0,1\ \Omega$ untuk arus beban 1 mA.

2.3 LCD (*Lycuid Crystal Display*)

(Abdul Choir, 2014) LCD (*Lycuid Crystal Display*) adalah suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama, selain itu LCD juga dapat digunakan untuk menampilkan karakter ataupun gambar. Pada Gambar 2.5 merupakan bentuk dari LCD.



Gambar 2.3 LCD

Berikut ini merupakan tabel yang menjelaskan mengenai konfigurasi pin dari LCD 16x2 :

Tabel 2.1 Konfigurasi Pin LCD 16 x 2

| Pin | Simbol | Level | Tujuan | Fungsi |
|-----|-----------------|-------|--------------|--|
| 1 | V _{SS} | - | Power Supply | Ground |
| 2 | V _{DD} | - | Power Supply | Tegangan Supply |
| 3 | V _{LS} | - | Power Supply | Power Supply untuk mendrive LCD guna mengatur kontrasnya |
| 4 | RS | H/L | uC | H : Data; L : Instruction Code |
| 5 | R/W | H/L | uC | H : Read; L : Write |
| 6 | E | H/L | uC | Enable |
| 7 | DB0 | H/L | uC | Data Bus Line |
| 8 | DB1 | H/L | uC | |
| 9 | DB2 | H/L | uC | |
| 10 | DB3 | H/L | uC | |
| 11 | DB4 | H/L | uC | |
| 12 | DB5 | H/L | uC | |
| 13 | DB6 | H/L | uC | |
| 14 | DB7 | H/L | uC | |

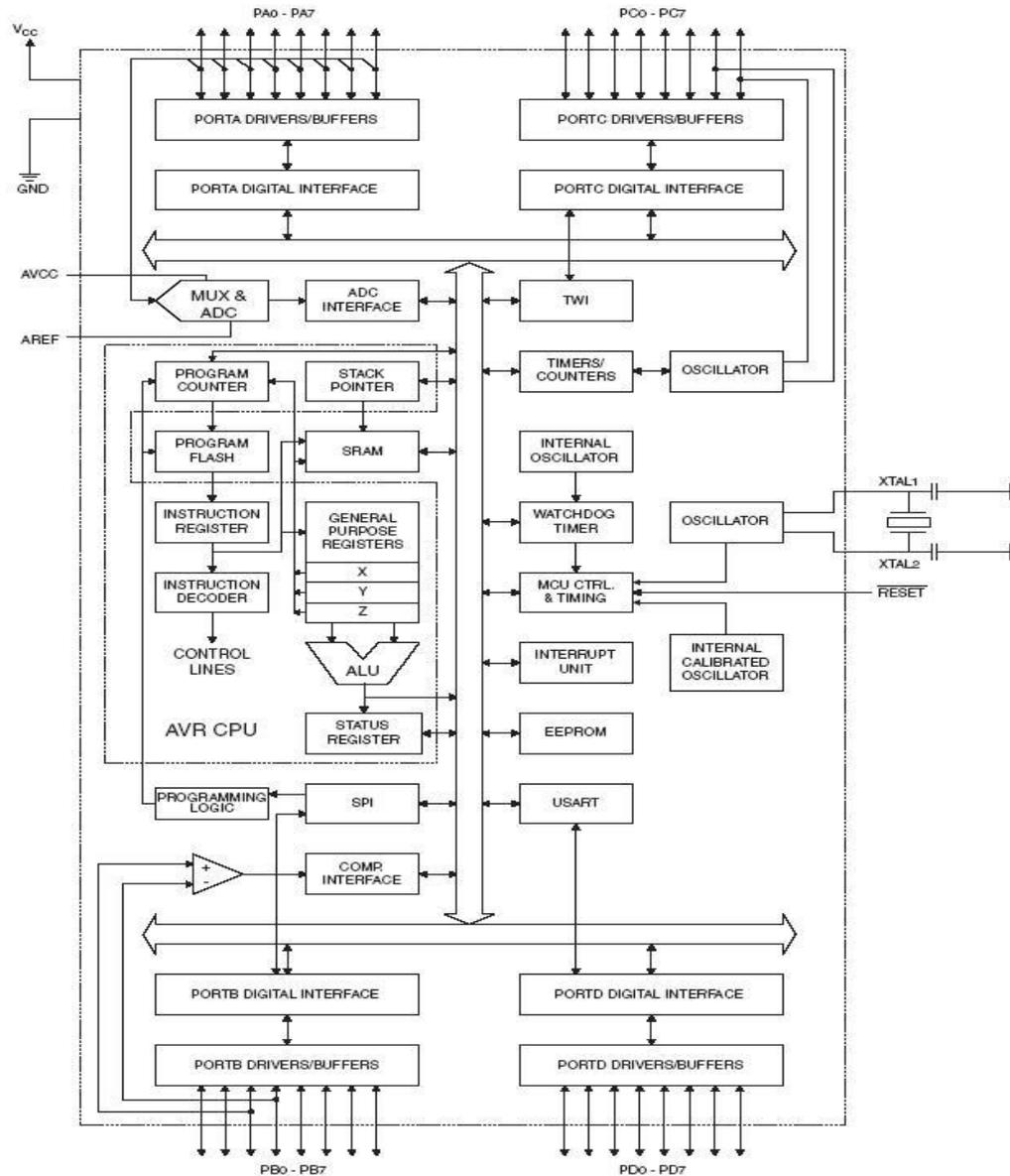
| | | | | |
|----|------|---|------------------|-----------------|
| 15 | V+BL | - | Back Ligh Supply | Tegangan Supply |
| 16 | V-BL | - | Back Ligh Supply | Ground |

2.4 Mikrokontroler AVR ATMega8535

(Agus Bejo, 2008) Mikrokontroler AVR memiliki arsitektur RISC 8 Bit, sehingga semua instruksi dikemas dalam kode 16-bit (16-bits *word*) dan sebagian besar instruksi dieksekusi dalam satu siklus instruksi *clock*. Dan ini sangat membedakan sekali dengan instruksi MCS-51 (Berarsitektur CISC) yang membutuhkan siklus 12 *clock*. Atmega 8535 merupakan *chip IC* keluaran ATMEL yang termasuk golongan *single chip microcontroller*, dimana semua rangkaian termasuk I/O dan memori tergabung dalam satu bentuk IC.

2.4.1 Diagram Blok ATMega 8535

(Agus Bejo, 2008) Ada 3 jenis tipe AVR yaitu AT Tiny, AVR klasik, AT Mega. Perbedaannya hanya pada fasilitas dan I/O yang tersedia serta fasilitas lain seperti ADC, EEPROM dan lain sebagainya. ATMega 8535 merupakan salah satu tipe AVR yang memiliki teknologi RISC dengan kecepatan maksimal 16 MHz membuat ATMega8535 lebih cepat bila dibandingkan dengan varian MCS 51. Dengan fasilitas yang lengkap tersebut menjadikan ATMega8535 sebagai mikrokontroler yang *powerfull*. Adapun blok diagramnya adalah sebagai berikut seperti terlihat pada gambar 2.6 yang diambil dari pustaka (W.N. Riantiningsih, 2009).



Gambar 2.4 Diagram Blok ATmega8535

1. Saluran I / O sebanyak 32 buah, yaitu Port A, Port B, Port C, dan Port D.
2. ADC 10 bit sebanyak 8 saluran.
3. Tiga buah Timer / Counter dengan kemampuan perbandingan.
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. Unit interupsi internal dan eksternal
9. Port antarmuka SPI.
10. EEPROM sebesar 512 byte yang dapat diprogram saat operasi.
11. Antarmuka komparator analog.
12. Port USART untuk komunikasi serial.

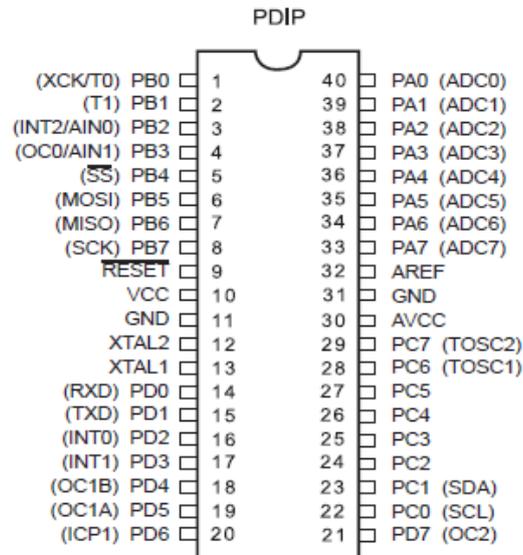
2.4.2 Fitur ATmega 8535

(Agus Bejo, 2008) Mikrokontroler ATmega8535 memiliki beberapa fitur diantaranya sebagai berikut :

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

2.4.3 Konfigurasi Pin ATmega8535

(Agus Bejo, 2008) IC Atmega 8535 ada 2 jenis yaitu jenis PDIP (berbentuk balok) dan jenis TQFP/MLF (berbentuk kotak) yang pada dasarnya memiliki fasilitas yang sama, hanya saja memiliki bentuk yang berbeda sehingga letak kaki-kaki IC berbeda mengikuti bentuknya. Gambar 2.7. berikut ini merupakan konfigurasi pin mikrokontroler ATmega8535 yang diambil dari pustaka (W.N. Riantiningsih, 2009).



Gambar 2.5 Konfigurasi Pin ATmega8535

Dari Gambar 2.7 diatas dapat dilihat ada 40 pin IC yang Secara fungsional konfigurasi pin tersebut sebagai berikut :

1. VCC merupakan pin yang berfungsi sebagai pin masukan catu daya.
2. GND merupakan pin ground.
3. Port A (PA0..PA7) merupakan pin I/O dua arah dan pin masukan ADC.
4. Port B (PB0.. PB7) merupakan pin I / O dua arah dan pin fungsi khusus, yaitu Timer/Counter, komparator analog, dan SPI.
5. Port C (PC0.. PC7) merupakan pin I / O dua arah dan pin fungsi khusus, yaitu TWI, komparator *analog* dan *timer Oscillator*.
6. Port D (PD0.. PD7) merupakan pin I / O dua arah dan pin fungsi khusus, yaitu komparator *analog*, interupsi eksternal, dan Komunikasi *serial*.
7. *RESET* merupakan pin yang digunakan untuk me-*reset* mikrokontroler.
8. XTAL1 dan XTAL2 merupakan pin masukan *clock* ekstenal.
9. AVCC merupakan pin masukan tegangan untuk ADC.
10. AREF merupakan pin masukan tegangan Referensi ADC.

2.5 Logika Fuzzy

Logika *fuzzy* merupakan suatu cara atau metode untuk memetakan suatu ruang input kedalam suatu ruang output. Titik awal dari konsep modern mengenai ketidakpastian adalah *paper* yang dibuat oleh Lotfi A Zadeh (1965) dimana Zadeh memperkenalkan teori yang memiliki objek-objek dari himpunan *fuzzy* yang memiliki batasan tidak presisi dan keanggotaan himpunan *fuzzy* dan bukan dalam bentuk logika benar (*true*) atau salah (*false*) tapi dinyatakan dalam derajat (*degree*). Konsep ini dikenal dengan *Fuzziness* dan teorinya dinamakan *Fuzzy set theory*. *Fuzziness* dapat didefinisikan sebagai logika kabur berkenaan dengan semantik dari suatu kejadian, fenomena atau pernyataan itu sendiri.

Contoh logika *fuzzy* yaitu sebagai berikut :

1. Manajer pergudangan mengatakan pada manajer produksi seberapa banyak persediaan barang pada akhir minggu ini, kemudian manajer produksi akan menetapkan jumlah barang yang harus diproduksi esok hari.
2. Pelayan restoran memberikan pelayanan terhadap tamu, kemudian tamu akan memberikan tip yang sesuai atas baik tidaknya pelayan yang diberikan;
3. Anda mengatakan pada saya seberapa sejuk ruangan yang anda inginkan, saya akan mengatur putaran kipas yang ada pada ruangan ini.
4. Penumpang taksi berkata pada sopir taksi seberapa cepat laju kendaraan yang diinginkan, sopir taksi akan mengatur pijakan gas taksinya.

Fuzzy system didasari atas konsep himpunan kabur yang memetakan domain *input* kedalam domain *output*. Perbedaan mendasar himpunan tegas dengan himpunan kabur adalah nilai keluarannya. Himpunan tegas hanya memiliki dua nilai yaitu 0 dan 1 sedangkan himpunan kabur memiliki banyak nilai keluaran yang dikenal dengan nilai derajat keanggotaan.

Logika *Fuzzy* adalah peningkatan dari logika *boolean* yang berhadapan dengan konsep kebenaran sebagian dimana logika klasik (*crisp*) menyatakan bahwa segala hal dapat diekspresikan dalam istilah *binary* (0 atau 1, ya atau tidak serta hitam atau putih). Logika *fuzzy* menggantikan kebenaran *boolean* dengan

tingkat kebenaran, logika *fuzzy* memungkinkan nilai 0 atau 1, tingkat keabuan dan juga hitam atau putih serta dalam bentuk *linguistic*, konsep tidak pasti seperti “sedikit”, “lumayan” dan “kuat”.

Ada beberapa alasan mengapa orang menggunakan logika *fuzzy* (Kusumadewi S, Purnomo H : 2010) antara lain :

1. Konsep logika *fuzzy* mudah dimengerti, konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. Logika *fuzzy* sangat fleksibel.
3. Logika *fuzzy* memiliki toleransi terhadap data-data yang tidak tepat.
4. Mampu memodelkan fungsi-fungsi *non linear* yang sangat kompleks.
5. Dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa melalui proses pelatihan.
6. Dapat bekerjasama dengan teknik-teknik kendali secara konvensional.
7. Didasarkan pada bahasa alami.

2.5.1 Metode Sugeno

Metode sugeno memiliki *output* (konsekuen) tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan linier. Ada dua model metode Sugeno yaitu model *fuzzy* sugeno orde nol dan model *fuzzy* sugeno orde satu. Dalam penulisan tugas akhir ini metode yang digunakan adalah metode sugeno orde (0) nol, metode ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985, sehingga metode ini sering juga dinamakan dengan metode TSK [3]. Metode *fuzzy* Sugeno orde-nol secara umum bentuk model *fuzzy* Sugeno orde-nol adalah :

$$R_i : \text{IF } \mathbf{x} \text{ is } A_i, \text{ THEN } y_i = f(x_i), \quad i=1, 2, \dots, n_R \quad \dots(2.1)$$

$$R_i : \text{IF } x_1 \text{ is } A_{i1} \text{ AND } x_2 \text{ is } A_{i2} \dots \text{ AND } x_r \text{ is } A_{ir}, \text{ THEN } y_i = f(x_i) \quad \dots(2.2)$$

Dimana,

\mathbf{x} adalah variabel *antecedent* atau disebut sebagai variabel input. Himpunan *fuzzy* variabel *antecedent* diberikan dalam bentuk vektor

$$\mathbf{x} = [x_1, x_2, \dots, x_r]^T \quad \dots(2.3)$$

y adalah variabel *consequent* atau variabel output. Model linear dari y diberikan pada persamaan :

$$y_i = \mathbf{a}_i^T \mathbf{x} + b_i \quad \dots(2.4)$$

Batasan *fuzzy* $\mu_A(\mathbf{x}): X_1 \times X_2 \times \dots \times X_r \rightarrow [0,1]$ merupakan derajat keanggotaan \mathbf{x} dalam A .

2.6 Relay

(Pambudi, 2000) Relay adalah sebuah saklar magnetik yang menggunakan medan magnet dan sebuah kumparan untuk membuka atau menutup satu atau beberapa kontak saklar pada saat relay dialiri arus. Pada dasarnya relay terdiri dari sebuah lilitan kawat yang terlilit pada suatu besi dari inti besi lunak yang selanjutnya berubah menjadi menjadi magnet yang menarik atau menolak suatu pegas sehingga kontak pun menutup atau membuka. Relay sering digunakan pada sistem elektronik sebagai sistem antarmuka antara sistem kendali dengan peralatan yang dikendalikan.



Gambar 2.6 Relay

2.6.1 Sifat-Sifat Relay

Apabila ingin menggunakan relay ada baiknya terlebih dahulu mengetahui sifat-sifat relay. Sifat-sifat relay yang perlu diketahui adalah besarnya tahanan. Tahanan ini ditentukan dengan tebal kawat yang digunakan dan banyaknya lilitan yang digunakan. Tahanan ini berharga antara 1 sampai dengan 50 k Ω .

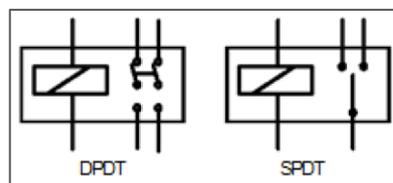
Adapun sifat-sifat relay adalah sebagai berikut:

1. Kuat arus yang digunakan guna pengoperasian relay ditentukan oleh pabrik pembuatnya. Relay dengan tahanan kecil memerlukan arus yang besar dan juga sebaliknya relay dengan tahanan besar memerlukan arus yang kecil.
2. Tegangan yang diperlukan untuk menggerakkan suatu relay akansama dengan kuat arus yang dikalikan dengan tahanan atau hambatan relay.

2.6.2 Konstruksi Relay

Simbol relay yang diperlihatkan pada gambar 2.9 adalah jenis DPDT (*Doublepole Double Throw*) dan SPDT (*Single Pole Double Throw*). *Pole* adalah kontak yang bergerak, sedangkan *throw* adalah kontak diam. NC (*Normally-Closed*) menunjukkan bahwa kontak tersebut pada keadaannya normal (*relay-off*) terhubung dengan *pole*. Sedangkan NO (*Normally-Opened*) pada keadaan normalnya tidak terhubung dengan *pole*.

Relay yang baik mempunyai resistansi isolasi yang tinggi, sehingga tegangan yang tinggi pada peralatan tidak mengganggu kerja dari rangkaian pengendali. Ada dua jenis relay yang bisa didapat yaitu inputnya bekerja pada arus searah dan yang bekerja pada arus bolak-balik. Pada umumnya relay yang digunakan pada rangkaian adalah yang bekerja pada tegangan DC. (Dedy, 2001)



Gambar 2.7 Konstruksi Relay

2.6.3 Prinsip Kerja Relay

Relay akan bekerja bila kontak-kontak yang terdapat pada relay tersebut dialiri arus pada kumparannya. Relay *normally open* kontak-kontaknya mempunyai posisi terbuka pada saat relay tidak bekerja dan akan menutup setelah ada arus yang mengalir. Sedangkan relay *normally close* kontak-kontaknya \

mempunyai posisi tertutup pada saat relay tidak bekerja dan akan membuka setelah ada arus yang mengalir.

Adapun dari jenis relay yang digunakan yaitu:

1. Relay Magnetik

Relay magnetik adalah relay yang bekerja berdasarkan magnet listrik untuk menggerakkan kontak-kontak mekaniknya. Jika kontak magnetiknya berada dalam keadaan NO maka akan berubah menjadi NC dan sebaliknya.

2. Relay Elektronik

Relay elektronik merupakan relay yang bekerja dengan menggunakan komponen-komponen elektronika pada saat pensuplaiannya.

Relay memiliki parameter antara lain sebagai berikut:

1. Resistansi kumparan

Dimana resistansi kumparan ini ditentukan oleh tebal kawat dari jumlah lilitan yang digunakan.

2. Arus driver

Arus driver adalah arus yang diperlukan untuk mengaktifkan relay. Besar arus inibiasanya sudah ditetapkan oleh pabrik produksinya.

3. Tegangan driver

Tegangan driver adalah tegangan yang digunakan untuk mengaktifkan relay

4. Daya driver

Daya driver adalah perkalian antara arus dan tegangan driver. Daya ini adalah yang digunakan untuk mengaktifkan relay. (Dedy, 2001)

2.7 Bahasa Pemrograman C

(Agus Bejo, 2008) Bahasa program adalah suatu bahasa ataupun suatu tata cara yang dapat digunakan oleh manusia (*programmer*) untuk berkomunikasi secara langsung dengan komputer. Bahasa C adalah bahasa pemrograman yang

dapat dikatakan berada di antara bahasa beraras rendah dan beraras tinggi. Bahasa beraras rendah artinya bahasa yang berorientasi pada mesin, sedangkan beraras tinggi berorientasi pada manusia. Bahasa beraras rendah misalnya assembler, ditulis dengan sandi yang hanya dimengerti oleh mesin sehingga hanya digunakan bagi yang memprogram mikroprosesor. Bahasa beraras tinggi relatif mudah digunakan karena ditulis dengan bahasa manusia sehingga mudah dimengerti dan tidak tergantung mesinnya. Bahasa beraras tinggi umumnya digunakan pada komputer.

Program Bahasa C tidak mengenal aturan penulisan dikolom tertentu sehingga bisa dimulai dari kolom manapun. Namun demikian untuk mempermudah pembacaan program dan keperluan dokumentasi sebaiknya penulisan bahasa C diatur sedemikian rupa sehingga mudah dan enak di baca. Program dalam bahasa C selalu berbentuk fungsi seperti ditunjukkan main (). Program yang dijalankan berada dalam tubuh program dan dimulai dengan tanda kurung buka { dan diakhiri dengan kurung tutup }. Semua yang tertulis di dalam tubuh program disebut blok.

Tanda () digunakan untuk mengapit *argument* suatu fungsi. Argumen adalah suatu nilai yang akan digunakan dalam fungsi tersebut. Dalam fungsi main tidak ada argument sehingga tak ada data dalam (). Yang merupakan perintah dan harus dikerjakan oleh prosesor. Setiap pernyataan diakhiri tanda titik koma ;. Baris pertama `#include <...>` bukanlah pernyataan sehingga tak diakhiri tanda titik koma (;). Baris tersebut meminta *compiler* untuk menyertakan file yang namanya ada di antara tanda <...> dalam proses kompilasi. File-file ini (berekstensi.h) berisi deklarasi fungsi ataupun *variable*. File ini disebut *header* dan digunakan semacam perpustakaan untuk pernyataan yang ada di tubuh program.

2.7.1 Tipe Data

Tipe data merupakan bagian program yang paling penting karena tipe data mempengaruhi setiap instruksi yang akan dilaksanakan oleh computer. Misalnya

saja 5 dibagi 2 bisa saja menghasilkan hasil yang berbeda tergantung tipe datanya. Jika 5 dan 2 bertipe integer maka akan menghasilkan nilai 2, namun jika keduanya bertipe float maka akan menghasilkan nilai 2.5000000. Pemilihan tipe data yang tepat akan membuat proses operasi data menjadi lebih efisien dan efektif.

Tabel 2.2 Tipe-tipe data

| Tipe Data | Ukuran | Jangkauan Nilai |
|-------------------|---------------|---|
| Bit | 1 bit | 0 atau 1 |
| Char | 1 byte | -128 s/d 225 |
| Unsigned Char | 1 byte | 0 s/d 225 |
| Signed Char | 1 byte | -128 s/d 127 |
| Int | 1 byte | -32.768 s/d 32.767 |
| Short Int | 2 byte | -32.768 s/d 32.767 |
| Unsigned Int | 2 byte | 0 s/d 65.535 |
| Signed Int | 2 byte | -32.768 s/d 32.767 |
| Long Int | 2 byte | -2.147.483.648 s/d 2.147.483.647 |
| Unsigned Long Int | 4 byte | 0 s/d 4.294.967.295 |
| Signed Long Int | 4 byte | -2.147.483.648 s/d 2.147.483.647 |
| Float | 4 byte | $1.2 \cdot 10^{-38}$ s/d $3.4 \cdot 10^{+38}$ |
| Double | 4 byte | $1.2 \cdot 10^{-38}$ s/d $3.4 \cdot 10^{+38}$ |

2.8 CodeVisionAVR

CodeVisionAVR merupakan sebuah cross-compiler C, Integrated Development Environment (IDE), dan Automatic Program Generator yang didesain untuk mikrokontroler buatan Atmel seri AVR. CodeVisionAVR dapat dijalankan pada sistem operasi Windows 95, 98, Me, NT4, 2000, dan XP.

Cross-compiler C mampu menerjemahkan hampir semua perintah dari bahasa ANSI C, sejauh yang diijinkan oleh arsitektur dari AVR, dengan tambahan

beberapa fitur untuk mengambil kelebihan khusus dari arsitektur AVR dan kebutuhan pada sistem embedded.

File object COFF hasil kompilasi dapat digunakan untuk keperluan debugging pada tingkatan C, dengan pengamatan variabel, menggunakan debugger Atmel AVR Studio.

IDE mempunyai fasilitas internal berupa software AVR Chip In-System Programmer yang memungkinkan Anda untuk melakukan transfer program kedalam chip mikrokontroler setelah sukses melakukan kompilasi/assembly secara otomatis. Software In-System Programmer didesain untuk bekerja dengan Atmel STK500/AVRISP/AVRProg, Kanda Systems STK200+/300, Dontronics DT006, Vogel Elektronik VTEC-ISP, Futurlec JRAVR dan MicroTronics ATCPU/Mega2000 programmers/development boards.

Untuk keperluan debugging sistem embedded, yang menggunakan komunikasi serial, IDE mempunyai fasilitas internal berupa sebuah Terminal. Selain library standar C, CodeVisionAVR juga mempunyai library tertentu untuk:

1. Modul LCD alphanumeric
2. Bus I2C dari Philips
3. Sensor Suhu LM75 dari National Semiconductor
4. Real-Time Clock: PCF8563, PCF8583 dari Philips, DS1302 dan DS1307 dari Maxim/Dallas Semiconductor
5. Protokol 1-Wire dari Maxim/Dallas Semiconductor
6. Sensor Suhu DS1820, DS18S20, dan DS18B20 dari Maxim/Dallas Semiconductor
7. Termometer/Termostat DS1621 dari Maxim/Dallas Semiconductor
8. EEPROM DS2430 dan DS2433 dari Maxim/Dallas Semiconductor
9. SPI
10. Power Management
11. Delay
12. Konversi ke Kode Gray

CodeVisionAVR juga mempunyai Automatic Program Generator bernama CodeWizardAVR, yang mengujinkan Anda untuk menulis, dalam hitungan menit, semua instruksi yang diperlukan untuk membuat fungsi-fungsi berikut:

1. Set-up akses memori eksternal
2. Identifikasi sumber reset untuk chip
3. Inisialisasi port input/output
4. Inisialisasi interupsi eksternal
5. Inisialisasi Timer/Counter
6. Inisialisasi Watchdog-Timer
7. Inisialisasi UART (USART) dan komunikasi serial berbasis buffer yang digerakkan oleh interupsi
8. Inisialisasi Pembanding Analog
9. Inisialisasi ADC
10. Inisialisasi Antarmuka SPI
11. Inisialisasi Antarmuka Two-Wire
12. Inisialisasi Antarmuka CAN
13. Inisialisasi Bus I2C, Sensor Suhu LM75, Thermometer/Thermostat DS1621 dan Real-Time Clock PCF8563, PCF8583, DS1302, dan DS1307
14. Inisialisasi Bus 1-Wire dan Sensor Suhu DS1820, DS18S20
15. Inisialisasi modul LCD