

II. TINJAUAN PUSTAKA

A. Tanaman Kelapa Sawit

Tanaman sawit (*Elaeis guineensis*) termasuk famili *Palmae*, subkelas *Monocotyledoneae*, kelas *Angiospermae*, subdivisi *Tracheophyta*. Nama *Enus* *Elaeis* berasal dari bahasa Yunani *Elaion* atau minyak, sedangkan nama *Guineensis* dari kata *Guins*, yaitu nama tempat dimana seseorang bernama Jacquin menemukan tanaman sawit pertama kali di pantai Guines Afrika Selatan

Sawit merupakan tanaman asli Afrika dan tumbuh secara alami di Afrika Selatan dan Afrika Barat. Tanaman ini dapat tumbuh baik pada daerah beriklim tropis dengan curah hujan 2000 mm/tahun (Ketaren, 2005).

B. Minyak Kelapa Sawit

Minyak kelapa sawit diperoleh dari pengolahan buah kelapa sawit (*Elaeis guineensis* JAC). Secara garis besar buah kelapa sawit terdiri dari serabut buah (pericarp) dan inti (kernel). Serabut buah kelapa sawit terdiri dari tiga lapis yaitu lapisan luar atau kulit buah yang disebut *pericarp*, lapisan sebelah dalam disebut *mesocarp* atau *pulp* dan lapisan paling dalam disebut *endocarp*. Inti kelapa sawit terdiri dari lapisan kulit biji (testa), endosperm dan embrio. *Mesocarp*

mengandung kadar minyak rata-rata sebanyak 56%, inti (kernel) mengandung minyak sebesar 44%, dan endocarp tidak mengandung minyak (Fauzi, 2006).

Minyak kelapa sawit seperti umumnya minyak nabati lainnya adalah merupakan senyawa yang tidak larut dalam air, sedangkan komponen penyusunnya yang utama adalah trigliserida dan nontrigliserida (Wijana, 2005).

C. Indeks Bias

Pada awalnya, oleh Newton, cahaya dipandang sebagai pancaran partikel-partikel halus dan pada tahap berikutnya, oleh Huygens, cahaya dipandang sebagai gelombang elastis longitudinal di dalam medium ether yang mengisi seluruh ruang kosong. Dari eksperimen yang dilakukan oleh Heinrich Hertz di sekitar pertengahan abad 19 yang lalu, cahaya diyakini sampai sekarang sebagai gelombang elektromagnetik yang menjalarnya tidak memerlukan medium. Sehingga teori-teori Newton dan Huygens maupun hipotesis adanya ether jadi runtuh, disanggah sepenuhnya dan terjabarkan berdasarkan teori gelombang elektromagnetik (Soedjojo, 2004).

Cahaya berjalan menempuh garis lurus pada berbagai keadaan. Anggapan yang masuk akal ini mengarah kemodel berkas dari cahaya. Model ini menganggap bahwa cahaya berjalan dalam lintasan yang berbentuk garis lurus yang disebut berkas cahaya. Sebenarnya berkas merupakan idealisasi, dimaksudkan untuk merepresentasikan sinar cahaya yang sangat sempit. Ketika kita melihat sebuah benda, menurut model berkas, cahaya mencapai mata kita dari setiap titik pada benda, walaupun berkas cahaya meninggalkan setiap titik dengan banyak arah,

biasanya hanya satu kumpulan kecil dari berkas-berkas ini yang dapat memasuki mata.

Pembiasan cahaya adalah pembelokan cahaya ketika berkas cahaya melewati bidang batas dua medium yang berbeda indeks biasnya. Indeks bias mutlak suatu bahan adalah perbandingan kecepatan cahaya di ruang hampa dengan kecepatan cahaya di bahan tersebut. Indeks bias relatif merupakan perbandingan indeks bias dua medium berbeda. Indeks bias relatif medium kedua terhadap medium pertama adalah perbandingan indeks bias antara medium kedua dengan indeks bias medium pertama.

Arah pembiasan cahaya dibedakan menjadi dua macam yaitu :

a. Mendekati garis normal

Cahaya dibiaskan mendekati garis normal jika cahaya merambat dari medium optik kurang rapat ke medium optik lebih rapat, contohnya cahaya merambat dari udara ke dalam air.

b. Menjauhi garis normal

Cahaya dibiaskan menjauhi garis normal jika cahaya merambat dari medium optik lebih rapat ke medium optik kurang rapat, contohnya cahaya merambat dari dalam air ke udara.

Pembiasan cahaya dapat terjadi dikarenakan perbedaan laju cahaya pada kedua medium. Laju cahaya pada medium yang rapat lebih kecil dibandingkan dengan laju cahaya pada medium yang kurang rapat. Menurut Christian Huygens (1629-1695) : "*Perbandingan laju cahaya dalam ruang hampa (c) dengan laju cahaya dalam suatu zat (v) dinamakan **indeks bias** (n).*"

$$n = \frac{c}{v} \quad (1)$$

Indeks bias tidak pernah lebih kecil dari 1, (artinya, $n = 1$), dan nilainya untuk beberapa zat ditampilkan pada Tabel 2.1 berikut ini :

Tabel 2.1. Indeks Bias Beberapa Zat

Medium	$n = c/v$
Udara hampa	1,0000
Udara (pada STP)	1,0003
Air	1,333
Es	1,31
Alkohol etil	1,36
Gliserol	1,48
Benzena	1,50
Kaca	
Kuarsa lebur	1,46
Kaca korona	1,52
Api cahaya/kaca flinta	1,58
Lucite atau plexiglass	1,51
Garam dapur (Natrium Klorida)	1,53
Berlian	2,42

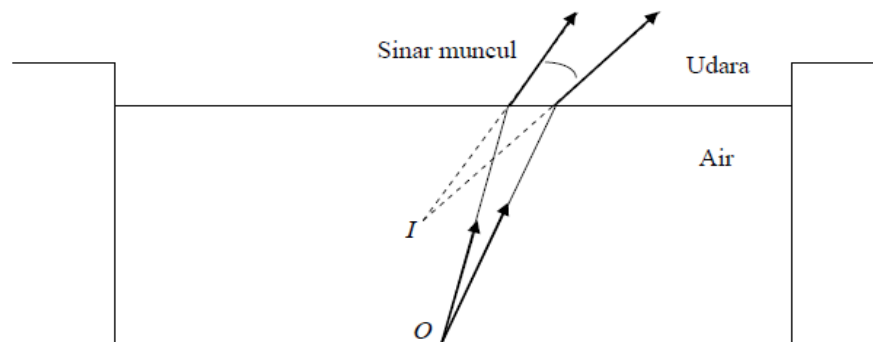
D. Hukum Snellius

Hukum-hukum snellius mendasari kaidah-kaidah optika geometris dalam alat-alat optik maupun sistem optik. Hukum-hukum itu tak lain ialah kaidah-kaidah yang berkenaan dengan pemantulan dan pembiasan sinar-sinar cahaya sebagai berikut:

1. Sinar datang, garis normal, sinar terpantul dan sinar terbias, semuanya terletak pada satu bidang datar.
2. Sudut pantul sama dengan sudut datang.
3. Perbandingan antara sinus sudut datang dan sinus sudut bias adalah tetap, tak tergantung besar sudut datang.

Adapun yang disebut sudut datang ialah sudut antara arah normal, yakni arah tegak lurus bidang batas dan arah sinar datang, dan sudut pantul ialah sudut antara arah normal dan arah sinar terpantul. Sedang yang dimaksud sudut bias ialah sudut antara arah normal dan arah sinar terbias (Soedjo, 2004).

Jika kita melihat ke bawah dasar kolam renang, air terlihat dangkal. Ini disebabkan cahaya dari titik pada bagian bawah kolam dibiaskan di permukaan dari air ke udara sehingga muncul ke arah yang baru seperti Gambar 2.1.



Gambar 2.1. Cahaya dari O di dasar kolam renang dibiaskan di permukaan air.

Gambar 2.1 menunjukkan apa yang terjadi ketika sinar melintasi permukaan air dari udara ke kaca. Sinar paralel tetap berada dalam balok sisi paralel, tapi air mengubah arah di persimpangan permukaan. Perubahan arah disebabkan oleh perubahan kecepatan cahaya ketika melewati batas dari udara ke gelas yang bergerak lebih lambat. Ada juga perubahan panjang gelombang ketika gelombang cahaya masuk ke kaca dari udara. Sebuah gelombang bergerak lebih lambat memasuki kaca mulai ditangkap oleh gelombang berikutnya yang masih di udara. Kemudian memasuki gelombang berikutnya ke kaca dan dua gelombang bergerak pada kecepatan baru tapi dengan pemisahan yang lebih kecil.

Ketika gelombang elektromagnetik menyentuh permukaan medium dielektrik dari suatu sudut, *leading edge* gelombang tersebut akan melambat sementara *trailing edgenya* tetap melaju normal. Penurunan kecepatan *leading edge* disebabkan karena interaksi dengan elektron dalam medium tersebut. Saat *leading edge* menumbuk elektron, energi gelombang tersebut akan diserap dan kemudian di radiasi kembali. Penyerapan dan re-radiasi ini menimbulkan keterlambatan sepanjang arah perambatan gelombang. Kedua hal tersebut menyebabkan perubahan arah rambat gelombang yang disebut refraksi atau pembiasan. Perubahan arah rambat gelombang cahaya dapat dihitung dari indeks bias berdasarkan hukum Snellius:

$$\frac{\sin\theta_1}{\sin\theta_2} = \frac{v_1}{v_2} = \frac{n_2}{n_1} \quad (2)$$

- θ_1 dan θ_2 adalah sudut antara normal dengan masing-masing sinar bias n_1 dan n_2 adalah indeks bias masing-masing medium
- v_1 dan v_2 adalah kecepatan gelombang cahaya dalam masing-masing medium.

E. Dioda Laser (*Light Amplification by Stimulated Emission of Radiation*)

Dioda laser merupakan dioda semikonduktor yang memancarkan cahaya karena mekanisme pancaran/emisi terstimulasi (*stimulated emission*). Cahaya yang dipancarkan oleh dioda laser bersifat koheren. Dioda laser memiliki lebar spektral yang lebih sempit (s/d 1 nm) jika dibandingkan dengan LED sehingga dispersi chromatic dapat ditekan. Dioda laser diterapkan untuk transmisi data dengan bit

rate tinggi. Daya keluaran optik dari dioda laser adalah $-12 \text{ s/d} + 3 \text{ dBm}$. Karakteristik arus kemudi daya optik dioda laser tidak linear. Kinerja (keluaran daya optik, panjang gelombang, umur) dari dioda laser sangat dipengaruhi oleh temperatur operasi. Berikut ini adalah salah satu gambar jenis laser.



Gambar 2.2. Laser Dioda (a) dan Batrai (b)

Efek laser (*light amplification by simulated emission of radiation*) telah diperoleh dengan menggunakan bermacam-macam jenis bahan yang berbeda, termasuk gas, cairan-cairan, dan benda-benda padat. Jenis laser yang digunakan untuk komunikasi fiber optik ialah laser semikonduktor.

Dioda Laser mempunyai berbagai kelebihan dibandingkan dengan LED antara lain :

1. Efisiensi kopling dioda laser injeksi lebih besar sehingga kebutuhan pengulang untuk kominikasi jarak jauh lebih sedikit.
2. Daya keluaran dioda laser injeksi lebih tinggi sehingga cocok untuk komunikasi jarak jauh.
3. Lebar bidang cahaya keluaran sangat sempit sehingga cahaya lebih koheren.

4. Tanggapan waktunya lebih cepat sehingga pesat modulasinya lebih tinggi.

F. Light Dependent Resistor (LDR)

Light Dependent Resistor atau LDR adalah jenis resistor yang nilainya berubah seiring intensitas cahaya yang diterima oleh komponen tersebut. LDR biasanya digunakan sebagai detektor cahaya atau pengukur besaran konversi cahaya.

Sensor yang digunakan pada penelitian ini yaitu *Light Dependent Resistor (LDR)*, merupakan resistor yang bergantung terhadap cahaya. Prinsip kerja LDR yaitu ketika ada energi foton dari cahaya sebesar hf , yang melebihi energi gap semikonduktor dan mengenai permukaan semikonduktor maka elektron yang berada di pita valensi mengalami transisi ke pita konduksi serta merta meningkatkan *hole* di pita valensi (Anonim, 2012).

Bahan sensor LDR umumnya terbuat dari *cadmium sulfide (CdS)*, *cadmium selenid (CdSe)* atau *cadmium sulfoselenid (CdSSe)* yang dilapiskan diatas substrat keramik dengan pola zigzag. Pola zigzag dimaksudkan agar daya tangkap permukaan sensor LDR maksimum menerima cahaya (Hani, 2010).



Gambar 2.3. Sensor *Light Dependent Resistor (LDR)*

G. Transistor

Transistor adalah alat semikonduktor yang dipakai sebagai penguat, sebagai sirkuit pemutus dan penyambung (switching), stabilisasi tegangan, modulasi sinyal atau sebagai fungsi lainnya. Transistor dapat berfungsi semacam kran listrik, dimana berdasarkan arus inputnya (BJT) atau tegangan inputnya (FET), memungkinkan pengaliran listrik yang sangat akurat dari sirkuit sumberlistriknnya.

Pada umumnya, transistor memiliki 3 terminal. Tegangan atau arus yang dipasang di satu terminalnya mengatur arus yang lebih besar yang melalui 2 terminal lainnya. Transistor adalah komponen yang sangat penting dalam dunia elektronik modern. Dalam rangkaian analog, transistor digunakan dalam amplifier (penguat). Rangkaian analog melingkupi pengeras suara, sumber listrik stabil, dan penguat sinyal radio. Dalam rangkaian2 digital, transistor digunakan sebagai saklar berkecepatan tinggi. Beberapa transistor juga dapat dirangkai sedemikian rupa sehingga berfungsi sebagai **logic gate**, memori, dan komponen-komponen lainnya. Bisa dilihat pada gambar 2.5 dibawah ini.



Gambar 2.4. Transistor

Cara Kerja Transistor

Dari banyak tipe-tipe transistor modern, pada awalnya ada dua tipe dasar transistor, bipolar junction transistor (*BJT* atau *transistor bipolar*) dan field-effect transistor (*FET*), yang masing-masing bekerja secara berbeda.

Transistor bipolar dinamakan demikian karena kanal konduksi utamanya menggunakan dua polaritas pembawa muatan: elektron dan lubang, untuk membawa arus listrik. Dalam BJT, arus listrik utama harus melewati satu daerah/lapisan pembatas dinamakan depletion zone, dan ketebalan lapisan ini dapat diatur dengan kecepatan tinggi dengan tujuan untuk mengatur aliran arus utama tersebut.

FET (*juga dinamakan transistor unipolar*) hanya menggunakan satu jenis pembawa muatan (*elektron atau hole, tergantung dari tipe FET*). Dalam FET, arus listrik utama mengalir dalam satu kanal konduksi sempit dengan depletion zone di kedua sisinya (*dibandingkan dengan transistor bipolar dimana daerah Basis memotong arah arus listrik utama*). Dan ketebalan dari daerah perbatasan ini dapat dirubah dengan perubahan tegangan yang diberikan, untuk mengubah ketebalan kanal konduksi tersebut.

H. Refraktometer

Refraktometer adalah alat yang digunakan untuk mengukur kadar/ konsentrasi bahan terlarut. Misalnya gula, garam, protein, dsb. Prinsip kerja dari refraktometer sesuai dengan namanya adalah memanfaatkan refraksi cahaya. Refraktometer

ditemukan oleh Dr. Ernest Abbe seorang ilmuwan dari German pada permulaan abad 20 (Anonim, 2010).

Indeks bias adalah perbandingan kecepatan cahaya dalam udara dengan kecepatan cahaya dalam zat tersebut. Indeks bias berfungsi untuk identifikasi zat kemurnian, suhu pengukuran dilakukan pada suhu 20°C dan suhu tersebut harus benar-benar diatur dan dipertahankan karena sangat mempengaruhi indeks bias. Harga indeks bias dinyatakan dalam farmakope Indonesia edisi empat dinyatakan garis (D) cahaya natrium pada panjang gelombang 589,0 nm dan 589,6 nm. Alat yang digunakan untuk mengukur indeks bias adalah refraktometer ABBE. Untuk mencapai kestabilan, alat harus dikalibrasi dengan menggunakan plat glass standart (Anonim, 2010).



Gambar 2.5. Refraktometer

Refraktometer Abbe adalah refraktometer untuk mengukur indeks bias cairan, padatan dalam cairan atau serbuk dengan indeks bias dari 1,300 sampai 1,700 dan persentase padatan 0 sampai 95%, alat untuk menentukan indeks bias minyak, lemak, gelas optis, larutan gula, dan sebagainya, indeks bias antara 1,300 dan 1,700 dapat dibaca langsung dengan ketelitian sampai 0,001 dan dapat diperkirakan sampai 0,0002 dari gelas skala di dalam.

Pengukurannya didasarkan atas prinsip bahwa cahaya yang masuk melalui prisma-cahaya hanya bisa melewati bidang batas antara cairan dan prisma kerja dengan suatu sudut yang terletak dalam batas-batas tertentu yang ditentukan oleh sudut batas antara cairan dan alas.

I. Mikrokontroler ATmega 8535

Mikrokontroler adalah IC yang dapat diprogram berulang kali, baik ditulis atau dihapus (Agus Bejo, 2007). Biasanya digunakan untuk pengontrolan otomatis dan manual pada perangkat elektronika.

Beberapa tahun terakhir, mikrokontroler sangat banyak digunakan terutama dalam pengontrolan robot. Seiring perkembangan elektronika, mikrokontroler dibuat semakin kompak dengan bahasa pemrograman yang juga ikut berubah. Salah satunya adalah mikrokontroler *Alf and Vegard's Risc processor* (AVR) ATmega8535 yang menggunakan teknologi *Reduce Instruction Set Computing* (RISC) dimana program berjalan lebih cepat karena hanya membutuhkan satu *siklus clock* untuk mengeksekusi satu instruksi program. Secara umum, AVR dapat dikelompokkan menjadi 4 kelas, yaitu kelas ATtiny, keluarga AT90Sxx, keluarga ATmega, dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah memori, peripheral, dan fungsinya. Dari segi arsitektur dan instruksi yang digunakan, mereka bisa dikatakan hampir sama.

Mikrokontroler AVR ATmega8535 memiliki fitur yang cukup lengkap. Mikrokontroler AVR ATmega8535 telah dilengkapi dengan ADC *internal*,

EEPROM internal, Timer/Counter, PWM, analog comparator, dll (M. Ary Heryanto).

Fitur-fitur yang dimiliki oleh mikrokontroler ATmega8535 adalah sebagai berikut:

1. Saluran I/O sebanyak 32 buah, yaitu port A, port B, port C, dan port D.
2. ADC internal sebanyak 8 saluran.
3. Tiga buah Timer/Counter dengan kemampuan perbandingan.
4. CPU yang terdiri atas 32 buah register.
5. SRAM sebesar 512 byte.
6. Memori Flash sebesar 8 kb dengan kemampuan Read While Write.
7. Port antarmuka SPI
8. EEPROM sebesar 512 byte yang dapat diprogram saat operasi.
9. Antarmuka komparator analog.
10. Port USART untuk komunikasi serial.
11. Sistem mikroprosesor 8 bit berbasis RISC dengan kecepatan maksimal 16 MHz.
12. Dan lain-lainnya.

a. Konstruksi ATmega8535

Mikrokontroler ATmega8535 memiliki 3 jenis memori, yaitu memori program, memori data dan memori EEPROM. Ketiganya memiliki ruang sendiri dan terpisah.

a. Memori program

ATmega8535 memiliki kapasitas memori program sebesar 8 Kbyte yang terpetakan dari alamat 0000h – 0FFFh dimana masing-masing alamat

memiliki lebar data 16 bit. Memori program ini terbagi menjadi 2 bagian yaitu bagian program *boot* dan bagian program aplikasi.

b. Memori data

ATmega8535 memiliki kapasitas memori data sebesar 608 byte yang terbagi menjadi 3 bagian yaitu register serba guna, register I/O dan SRAM. ATmega8535 memiliki 32 byte register serba guna, 64 byte register I/O yang dapat diakses sebagai bagian dari memori RAM (menggunakan instruksi LD atau ST) atau dapat juga diakses sebagai I/O (menggunakan instruksi IN atau OUT), dan 512 byte digunakan untuk memori data SRAM.

c. Memori EEPROM

ATmega8535 memiliki memori EEPROM sebesar 512 byte yang terpisah dari memori program maupun memori data. Memori EEPROM ini hanya dapat diakses dengan menggunakan register-register I/O yaitu register EEPROM *Address*, register EEPROM *Data*, dan register EEPROM *Control*. Untuk mengakses memori EEPROM ini diperlakukan seperti mengakses data eksternal, sehingga waktu eksekusinya relatif lebih lama bila dibandingkan dengan mengakses data dari SRAM.

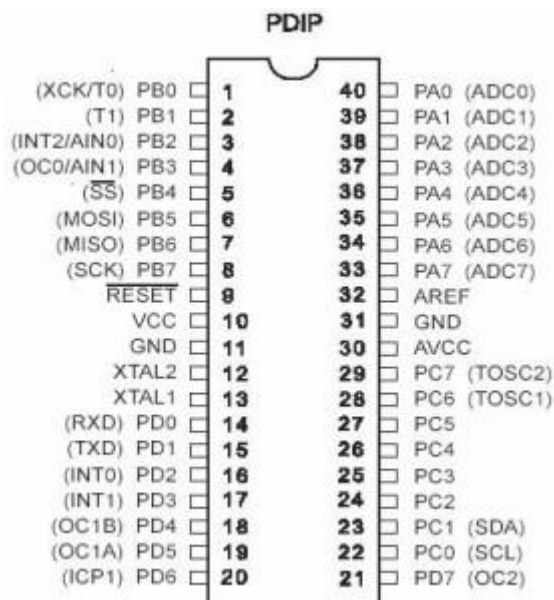
ATmega8535 merupakan tipe AVR yang telah dilengkapi dengan 8 saluran ADC internal dengan fidelitas 10 bit. Dalam mode operasinya, ADC ATmega8535 dapat dikonfigurasi, baik secara *single ended input* maupun *differential input*. Selain itu, ADC ATmega8535 memiliki konfigurasi pewaktuan, tegangan referensi, mode operasi, dan kemampuan filter derau yang amat fleksibel,

sehingga dengan mudah disesuaikan dengan kebutuhan ADC itu sendiri. Berikut ini adalah gambar mikrokontroler Atmega8535.



Gambar 2.6. Mikrokontroler ATmega8535

b. Konfigurasi Pin ATmega8535



Gambar 2.7. Konfigurasi Pin ATmega8535

Secara umum konfigurasi dan fungsi pin ATmega8535 dapat dijelaskan sebagai berikut :

1. **VCC** Input sumber tegangan (+)
2. **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 ATmega8535".
5. **Port C (PC7 ... PC0)** Berfungsi sebagai port I/O dua arah. Fungsi lain port ini selengkapnya bisa dibaca pada buku petunjuk "AVR ATmega8535".
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 ATmega8535".
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.

J. Pemrograman Bahasa C

Pada suatu pengontrolan alat, program yang digunakan adalah pemrograman bahasa C. Untuk itu diperlukan juga pemahaman tentang pemrograman tersebut. Bahasa C merupakan salah satu bahasa yang cukup populer dan handal untuk pemrograman mikrokontroler. Dalam melakukan pemrograman mikrokontroler diperlukan suatu *software* pemrograman, salah satunya yang mendukung bahasa C adalah *Code Vision AVR (CVAVR)*. CVAVR hanya dapat digunakan pada mikrokontroler keluarga AVR. CVAVR selain dapat digunakan

sebagai *software* pemrograman juga dapat digunakan sebagai *software downloader*. *Software downloader* akan men-*download*-kan file berekstensi “.hex” ke mikrokontroler. (Averroes, 2009).

Berikut adalah penjelasan dasar-dasar dari pemrograman bahasa C :

1. Tipe Data

Tipe-tipe data yang ada dalam bahasa C dan yang dikenali oleh *CodeVisionAVR* dijelaskan dalam Tabel 2.2.

Tabel 2.2. Tipe-tipe data dalam bahasa C

No	Tipe Data	Ukuran	Jangkauan Nilai
1	Bit	1 bit	0 atau 1
2	Char	1 byte	-128 s/d 225
3	Unsigned Char	1 byte	0 s/d 225
4	Signed Char	1 byte	-128 s/d 127
5	Int	2 byte	-32.768 s/d 32.767
6	Short Int	2 byte	-32.768 s/d 32.767
7	Unsigned Int	2 byte	0 s/d 65.535
8	Signed Int	2 byte	-32.768 s/d 32.767
9	Long Int	4 byte	-2.147.483.648 s/d 2.147.483.647
10	Unsigned Long Int	4 byte	0 s/d 4.294.967.295
11	Signed Long Int	4 byte	-2.147.483.648 s/d 2.147.483.647
12	Float	4 byte	$1.2 \cdot 10^{-38}$ s/d $3.4 \cdot 10^{+38}$
13	Double	4 byte	$1.2 \cdot 10^{-38}$ s/d $3.4 \cdot 10^{+38}$

2. Konstanta dan Variabel

Konstanta dan variabel merupakan sebuah tempat untuk menyimpan data yang berada di dalam memori. Konstanta berisi data yang nilainya tetap dan tidak dapat diubah selama program dijalankan, sedangkan variabel berisi data yang bisa berubah nilainya saat program dijalankan. Untuk membuat sebuah konstanta atau variabel maka kita harus mendeklarasikannya lebih dahulu, yaitu dengan sintaks berikut : **Const** [tipe_data][nama_konstanta]=[nilai]

Contoh :

```
Const char konstantaku=0x10;
```

Deklarasi variabel : [tipe_data][nama_variabel]=[nilai_awal]

Contoh :

```
Char variabelku;
```

```
Char variabelku=0x20;
```

```
Bit variabel_bit;
```

```
Bit variabel_bit=1;
```

Pada deklarasi variabel, [nilai_awal] bersifat operasional sehingga boleh diisi dan boleh tidak diisi. Nilai_awal merupakan nilai default variabel tersebut dan jika tidak diisi maka nilai defaultnya adalah 0 (nol). Beberapa variabel dengan tipe yang sama dapat dideklarasikan dalam satu baris seperti contoh berikut :

```
Char data_a, data_b, data_c;
```

3. Komentar

Komentar adalah tulisan yang tidak dianggap sebagai bagian dari tubuh program. Komentar digunakan untuk memberikan penjelasan, informasi ataupun keterangan-keterangan yang dapat membantu mempermudah dalam memahami kode program baik bagi pembuat program maupun bagi orang lain yang membacanya. Komentar yang hanya satu baris ditulis dengan diawali `'//'` sedangkan komentar yang lebih dari satu baris diawali dengan `'/*'` dan diakhiri dengan `'*/'`. Selain digunakan untuk memberikan keterangan program, komentar juga dapat digunakan untuk membantu dalam pengujian program yaitu dengan menon-aktifkan dan mengaktifkan kembali bagian program tertentu selama proses pengujian.

4. Pengarah Preprocessor

Pengarah preprocessor digunakan untuk mendefinisikan prosesor yang digunakan, dalam hal ini adalah untuk mendefinisikan jenis mikrokontroler yang digunakan. Dengan pengarah preprosesor ini maka pendeklarasian register-register dan penamaannya dilakukan pada file yang lain yang disisipkan dalam program utama dengan sintaks sebagai berikut :

```
# include <nama_preprocessor>
```

Contoh :

```
# include <mega8535.h>
```

5. Pernyataan

Pernyataan adalah satu buah instruksi lengkap berdiri sendiri.

```
PORTC = 0x0F;
```

Pernyataan diatas merupakan sebuah instruksi untuk mengeluarkan data 0x0F ke

Port C. Contoh sebuah blok pernyataan :

```
{
  PORTA=0x00; // pernyataan_1
  PORTB=0x0F; // pernyataan_2
  PORTC=0xFF; // pernyataan_3
}
```

6. Operator Aritmatika

Operator aritmatika adalah beberapa operator yang digunakan untuk melakukan perhitungan aritmatika, seperti terlihat pada Tabel 2.3.

Tabel 2.3. Operator aritmatika

No	Operator	Keterangan
1	+	Operator untuk operasi penjumlahan
2	-	Operator untuk operasi pengurangan
3	*	Operator untuk operasi perkalian
4	/	Operator untuk operasi pembagian
5	%	Operator untuk operasi sisa pembagian

7. Pernyataan If

Pernyataan If digunakan untuk melakukan pengambilan keputusan terhadap dua buah kemungkinan yaitu mengerjakan suatu blok pernyataan atau tidak. Bentuk pernyataan If adalah :

```

if (kondisi)
{
// blok pernyataan yang akan dikerjakan
// jika kondisi if terpenuhi
}

```

Contoh :

```

if (PINA>0x80)
{
Dataku=PINA;
PORTC=0xFF;
}

```

Pernyataan if diatas akan mengecek apakah data yang terbaca pada Port A (PINA) nilainya lebih dari 0x80 atau tidak, jika ya maka variable dataku diisi dengan nilai PINA dan data 0xFF dikeluarkan ke PORT C.(Agus Bejo, 2008).

Dalam penulisan program C sebenarnya terbagi dua kategori yaitu:

1. Kategori deklarasi (declaration)

Deklarasi adalah membuat dan memberitahu kepada compiler tentang sesuatu yang digunakan nanti dalam penulisan program agar digunakan semestinya dan tidak dianggap *error* atau asing.

2. Kategori pernyataan (statement)

Pernyataan adalah membuat instruksi-instruksi program dengan menggunakan *keyword* seperti instruksi operasi aritmatika, logika, operasi bit, atau instruksi percabangan dan loping, atau pembuatan fungsi (Winoto, 2010).

K. Visual basic

Visual Basic merupakan sebuah *software* untuk membangun program atau aplikasi komputer yang dikembangkan dari bahasa BASIC dimana di dalamnya sudah berisi *statemen*, *fungsi* dan *keyword*. Konsep dasar dari *Visual Basic* adalah komponen pemrograman yang berorientasi pada visual. Kelebihan *Visual Basic* dibanding dengan bahasa pemrograman yang lain adalah mampu menambahkan sendiri sebagian kode program secara otomatis ke dalam program (Setyadi, 2000).

Dengan menggunakan *Visual Basic 6.0* dapat menghasilkan berbagai macam jenis program. Aplikasi yang dibuat dapat diintegrasikan dengan *database*, *hardware* lain (*interface*) dan sebagainya. *Visual Basic 6.0* merupakan salah satu *tool* untuk pengembangan aplikasi yang banyak diminati oleh orang. Di sini *Visual Basic 6.0* menawarkan kemudahan dalam pembuatan aplikasi dan dapat menggunakan komponen-komponen yang telah disediakan. Untuk memulai *Visual Basic 6.0*

perlu dipasang *software Visual Basic 6.0* pada komputer. Konsep dasar dari *Visual Basic* adalah komponen pemrograman yang berorientasi pada visual. Berikut adalah gambar tampilan awal *Visual Basic 6.0*.

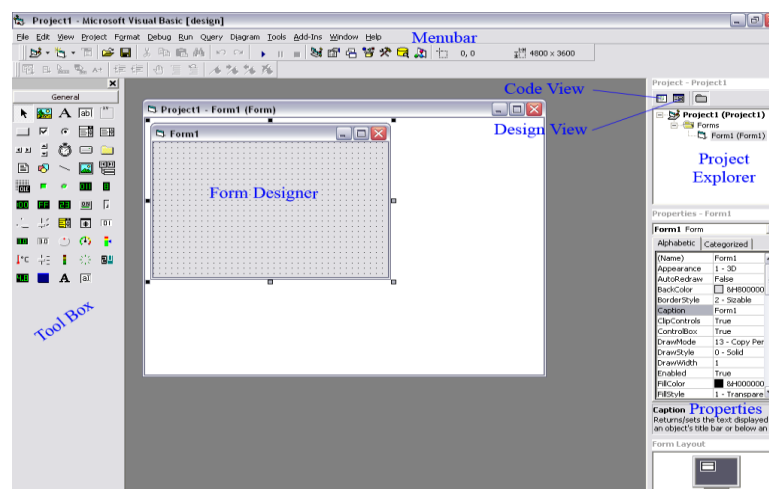


Gambar 2.8. Tampilan awal Visual Basic 6.0

Visual Basic 6.0 menyediakan banyak jenis modul aplikasi. Untuk memulai program standar pilihlah **Standar EXE**, kemudian klik open. Setelah itu akan muncul tampilan seperti berikut ini, yang menunjukkan bagian-bagian dari IDE (*Integrated Development Environment*) yang akan digunakan, antara lain:

1. **Form Designer** merupakan tempat perancangan *user interface* (antar muka pemakai). Untuk menampilkan layar ini, klik **Design View** atau dengan menekan **shift + F7**. Sedangkan untuk menampilkan layar *coding* dapat menekan **F7** atau klik pada **Coding View**.
2. **Menu** merupakan menu standar pada *Windows*, digunakan untuk menyimpan *project*, menjalankan *project*, membuka *project* baru dan sebagainya.

3. **Toolbox** merupakan tempat komponen-komponen yang disediakan untuk merancang *user interface*. Setiap komponen memiliki ciri dan kegunaan yang berbeda, disesuaikan dengan kebutuhan pengguna.
4. **Project Explorer** merupakan struktur *project* yang sedang dikerjakan, suatu *project* dapat terdiri dari beberapa *form*.
5. **Properties** menampilkan bagian-bagian dari komponen yang sedang aktif. Setiap komponen mempunyai karakteristik yang berbeda, bergantung pada kegunaan. Berikut adalah tampilan lembar kerja pada *Visual Basic 6.0*



Gambar 2.9. Tampilan lembar kerja *Visual Basic 6.0*