

BAB II

TINAUAN PUSTAKA

Dari penelitian Laporan Akhir sebelumnya yang dibuat pada tahun 2007 oleh Devi Ratnawati tentang Kajian Variasi Kadar Glukosa Dan Derajat Keasaman (Ph) Pada Pembuatan Nata De Citrus Dari Jeruk Asam, di mana alat yang digunakan untuk mengukur kadar *pH* adalah *pH* meter. (Ratnawati, 2007)

Pada tahun 2008, berdasarkan Laporan Akhir yang dibuat oleh Mahfudz Shidiq dan Panca M Rahardjo tentang Pengukur Suhu dan *pH* air Tambak terintegrasi dengan Data Logger, di mana alat dibuat menggunakan Sensor *pH* sebagai alat pendeteksi kadar *pH*. (Shidiq, 2008)

Pada tahun 2011, berdasarkan Laporan Akhir yang dibuat oleh Yudi Ariansyah yang berjudul Rancang Bangun Alat Pendeteksi Kadar *pH* Dalam Air Dengan *Output LCD* Berbasis Mikrokontroler, di mana untuk pengukuran kadar *pH* air menggunakan *pH* meter dengan *output* tampilan pada *LCD*. (Ariansyah, 2012).

2.1 Antarmuka

Antarmuka adalah salah satu layanan yang disediakan sistem operasi sebagai sarana interaksi antara pengguna (*user*) dengan sistem operasi. Antarmuka dikenal juga dengan istilah *interface*. Antarmuka merupakan komponen dari suatu sistem operasi yang bersentuhan langsung dengan pengguna atau pemakai. Antarmuka pemakai (*user interface*) merupakan mekanisme komunikasi antara pengguna (*user*) dengan sistem. Antarmuka pemakai dapat menerima informasi dari pengguna (*user*) dan memberikan informasi kepada pengguna (*user*) untuk membantu mengarahkan alur penelusuran masalah sampai ditemukan suatu solusi.

2.2 Monitoring

Monitoring adalah penilaian secara terus menerus terhadap fungsi kegiatan-kegiatan suatu program-program dalam hal jadwal penggunaan *input*/masukan data oleh kelompok sasaran berkaitan dengan harapan-harapan yang telah direncanakan. Secara umum *monitoring* digunakan untuk mengetahui ketepatan antara pelaksanaan dengan perencanaan program. *Monitoring* selain

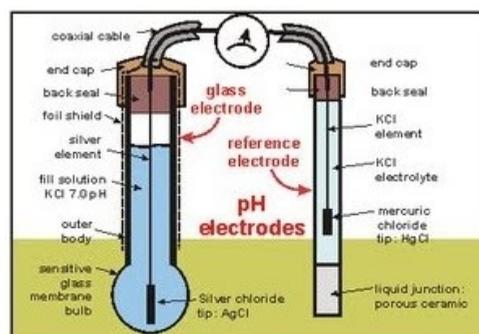
berkaitan dengan pengawasan dan supervisi, serta mempunyai hubungan erat dengan penilaian program.

2.3 Teori Dasar pH

pH adalah derajat keasaman yang digunakan untuk menyatakan tingkat keasaman atau kebasaan yang dimiliki oleh suatu larutan. Kadar pH yang normal bernilai 7, jika $pH < 7$ menunjukkan tingkat keasaman dari larutan sedangkan jika $pH > 7$ menunjukkan larutan yang bersifat basa. Derajat keasaman tertinggi suatu larutan bernilai 0 sementara derajat kebasaan tertinggi bernilai 14.

Ada 2 indikator yang dapat digunakan dalam pengukuran kadar pH , yaitu kertas lakmus dan pH meter. Indikator sederhana untuk mengukur kadar pH adalah kertas lakmus. Kertas lakmus adalah indikator pengukuran kadar pH yang jika ditetesi oleh suatu larutan yang memiliki derajat keasaman yang tinggi akan berwarna merah namun kertas lakmus akan berwarna biru jika ditetesi oleh suatu larutan yang memiliki derajat keasaman yang rendah.

Indikator pengukuran kadar pH lainnya adalah pH meter. pH meter tersebut bekerja dengan berdasarkan pada prinsip elektrolit atau konduktivitas suatu larutan. Istilah pH tersebut berasal dari huruf “ p ” dan “ H ”. Huruf “ p ” pada istilah pH yaitu lambang matematika dari negatif logaritma dan huruf “ H ” merupakan lambing kimia untuk unsur *hydrogen*. Sehingga pH merupakan *negative logaritma* dari aktivitas *ion hydrogen* (*power of hydrogen*).



Gambar 2.1 Skema elektroda sensor pH

2.3.1 Dasar Pengukuran pH

Asam dan basa adalah besaran yang sering digunakan untuk pengelolaan suatu zat. Pada prinsipnya pengukuran suatu pH adalah didasarkan pada potensial

elektro kimia yang terjadi antara larutan yang terdapat di dalam elektroda membran gelas yang telah diketahui, dengan larutan yang terdapat di luar elektroda membran gelas yang tidak diketahui. Hal tersebut dikarenakan lapisan tipis pada gelembung kaca akan berinteraksi dengan *ion hydrogen* yang berukuran relatif aktif dan kecil. Elektroda gelas tersebut akan mengukur potensial elektro kimia dari *ion hydrogen*.

Asam secara umum merupakan senyawa kimia yang jika dilarutkan ke dalam air maka akan menghasilkan larutan dengan *pH* dari 0 hingga 7. Dalam menerima aksi penetralan untuk membentuk garam, maka suatu asam akan bereaksi dengan basa. Pada umumnya suatu asam akan berasa asam. Sifat-sifat suatu asam di antaranya :

- Berasa asam jika dilarutkan ke dalam air
- Jika disentuh, asam akan terasa menyengat (terutama asam kuat)
- Asam bereaksi hebat dengan logam, seperti korosif terhadap logam

Selain itu, basa adalah senyawa kimia yang menyerap *ion hydronium* ketika dilarutkan ke dalam air. Basa ditujukan untuk senyawa kimia yang memiliki *pH* sekitar 7 hingga 14. Jika dilarutkan ke dalam air, maka akan menghasilkan *ion -OH*. Basa memiliki sifat-sifat yang berlawanan dengan asam. Sifat-sifat suatu basa di antaranya sebagai berikut :

- Tidak berasa asam jika dilarutkan ke dalam air
- Jika disentuh, basa tidak akan terasa menyengat
- Kebanyakan basa tidak bereaksi dengan logam

2.4 Air

Air adalah zat cair yang tidak mempunyai rasa, warna dan bau, yang terdiri dari *hydrogen* dan oksigen (H_2O). Karena air merupakan larutan yang hampir-hampir bersifat *universal*, maka zat-zat yang paling alamiah maupun buatan manusia hingga tingkat tertentu terlarut di dalamnya. Dengan demikian, air mengandung zat-zat terlarut. Zat-zat ini sering disebut pencemar yang terdapat di dalam air.

Sebagian besar tubuh semua organisme yang hidup terdiri dari air. Sekitar 70 atau 90 persen bahan organik terdiri dari air. Reaksi kimia yang mendukung

kehidupan di semua tumbuhan dan hewan berlangsung di dalam sebuah *medium* air. Air tidak hanya menyediakan media yang menjadi tempat dimungkinkannya reaksi yang menyokong kehidupan, tapi air sendiri sering menjadi produk atau reaktan yang penting dari reaksi-reaksi itu. Singkat kata, alkimia kehidupan ditemukan di dalam kimia air.

2.5 Komputer (*Laptop*)

Komputer adalah suatu alat yang memiliki kemampuan dalam melakukan beberapa tugasnya, seperti menerima dan memproses suatu *input*, menyimpan perintah-perintah, serta menyediakan suatu *output* dalam bentuk informasi. Komputer juga mampu mengoperasikan suatu program berupa perhitungan dan operasi logika.

2.6 Mikrokontroler

Mikrokontroler adalah sebuah sistem *microprocessor* yang di dalamnya sudah terdapat *CPU*, *ROM*, *RAM*, *I/O*, *Clock* dan peralatan *internal* lainnya yang sudah saling terhubung dan terorganisasi (teralamat) dengan baik oleh pabrik pembuatnya dan dikemas dalam satu *chip* yang siap pakai. Sehingga kita tinggal memprogram isi *ROM* sesuai aturan penggunaan oleh pabrik yang membuatnya.

Mikrokontroler merupakan sistem komputer yang mempunyai satu atau beberapa tugas yang sangat spesifik, berbeda dengan *personal computer* yang memiliki berbagai macam fungsi. (Winoto, 2008)

Berikut ini adalah peralatan *internal* pada mikrokontroler yang terdiri dari :

a. *CPU (Central Processing Unit)*

CPU adalah suatu *unit* pengolah pusat yang terdiri dari 2 bagian, yaitu *unit* pengendali (*control unit*) dan *unit* logika (*logic unit*). Selain itu, *CPU* juga mempunyai beberapa simpanan yang berukuran kecil yang disebut sebagai *register*. Fungsi utama dari *unit* pengendali (*control unit*) yaitu mengatur dan mengendalikan semua peralatan yang ada pada sistem komputer serta dapat mengatur kapan alat alat *input* menerima data dan kapan data diolah dan ditampilkan pada suatu alat *output*. Sedangkan *unit* logika (*logic unit*) berfungsi dalam melakukan semua perhitungan aritmatika yang terjadi sesuai dengan instruksi program serta melakukan pengambilan keputusan

dari operasi logika sesuai dengan instruksi yang diberikan kepadanya tersebut.

b. *Bus Data*

Bus data merupakan sejumlah lintasan saluran keluar masuknya data dalam suatu mikrokontroler. Pada umumnya saluran data yang masuk sama dengan saluran data yang keluar

c. *Bus Alamat*

Bus alamat berfungsi sebagai sejumlah lintasan saluran pengalamatan antara alamat dengan sebuah komputer. Pengalamatan tersebut harus ditentukan terlebih dahulu terhindar dari terjadinya kesalahan dalam pengiriman sebuah instruksi dan terjadinya bentrok antara 2 buah alat yang bekerja dalam waktu bersamaan.

d. *Bus Kontrol*

Bus Kontrol atau dikenal dengan *Bus Kendali* berfungsi dalam menyerempakkan operasi mikrokontroler dengan operasi rangkaian luar.

e. *Memori*

Di dalam sebuah mikrokontroler terdapat memori yang berfungsi untuk menyimpan data atau program. Ada beberapa jenis memori, yaitu *RAM (Random Access Memory)* dan *ROM (Read Only Memory)*. *Register internal* adalah memori yang terdapat di dalam *ALU (Arithmetic Logic Unit)*. Memori utama adalah memori yang ada pada suatu sistem, waktu aksesnya lebih lambat dibandingkan *register internal*. Sedangkan memori dipakai untuk penyimpanan dengan kapasitas tinggi, biasanya berbentuk disket, pita magnetik atau kaset.

f. *RAM (Random Access Memory)*

RAM adalah memori yang dapat dibaca atau ditulis. Data dalam *RAM* bersifat *volatile* di mana isinya akan hilang begitu *IC* kehilangan catu daya, karena sifat yang demikian *RAM* hanya digunakan untuk menyimpan data pada saat program bekerja.

g. *ROM (Read Only Memory)*

ROM merupakan memori yang hanya dapat dibaca, di mana isinya tidak dapat berubah apabila *IC* telah kehilangan catu daya. *ROM* dipakai untuk

menyimpan program. Pada saat di-*reset* maka mikrokontroler akan langsung bekerja dengan program yang terdapat di dalam *ROM* tersebut. Ada berbagai jenis *ROM* antara lain *ROM* murni, yaitu *PROM* (*Programmable Read Only Memory*) dan *EPROM* (*Eraseable Programmable Read Only Memory*). Namun yang paling banyak digunakan di antara tipe-tipe di atas adalah *EPROM* yang dapat diprogram ulang dan dapat juga dihapus sinar *ultraviolet*.

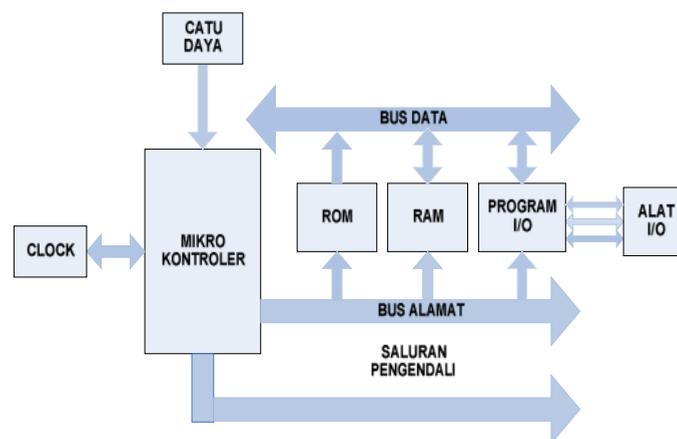
h. *Input/Output*

Setiap sistem komputer memerlukan sistem *input* dan *output* yang merupakan media keluar masuk data dari dan ke komputer. Contoh peralatan *I/O* yang umum yaitu terhubung dengan sebuah komputer seperti *keyboard*, *mouse*, *monitor*, *sensor*, *printer*, *LED*, dan sebagainya.

i. *Clock*

Clock atau pewaktuuan berfungsi untuk memberikan referensi waktu dan sinkronisasi antar elemen.

Secara umum blok diagram pada mikrokontroler seperti pada gambar berikut ini :



Gambar 2.2 Blok diagram mikrokontroler

2.6.1 Mikrokontroler *ATmega 16*

Salah satu mikrokontroler yang banyak digunakan saat ini yaitu mikrokontroler *AVR*. *AVR* adalah mikrokontroler *RISC* (*Reduce Instruction Set*

Compute) 8 bit berdasarkan arsitektur *Harvard*. Secara umum mikrokontroler AVR dapat dikelompokkan menjadi 3 kelompok, yaitu keluarga *AT90Sxx*, *ATmega*, *ATtiny*. Pada dasarnya yang membedakan masing-masing kelas adalah memori, *peripheral*, dan fiturnya.

Seperti mikroprosesor pada umumnya, secara *internal* mikrokontroler *ATmega 16* terdiri atas *unit-unit* fungsionalnya *Arithmetic and Logical Unit (ALU)*, himpunan *register* kerja, *register* dan dekoder instruksi, dan pewaktu beserta komponen kendali lainnya. Berbeda dengan mikroprosesor, mikrokontroler menyediakan memori dalam serpih yang sama dengan prosesornya (*in chip*).

Secara teknis hanya ada 2 macam mikrokontroler. Pembagian ini didasarkan pada kompleksitas instruksi-instruksi yang dapat diterapkan pada mikrokontroler tersebut. Pembagian itu yaitu *RISC* dan *CISC*. *RISC* merupakan kependekan dari *Reduced Instruction Set Compute*. Instruksi yang dimiliki terbatas, tetapi memiliki fasilitas yang lebih banyak. Sebaliknya, *CISC* kependekan dari *Complex Instruction Set Compute*. Instruksi bisa dikatakan lebih lengkap tapi dengan fasilitas secukupnya.

2.6.1.1 Arsitektur ATmega 16

Mikrokontroler ini menggunakan arsitektur *Harvard* yang memisahkan memori program dari memori data, baik *bus* alamat maupun *bus* data, sehingga pengaksesan program dan data dapat dilakukan secara bersamaan (*concurrent*).

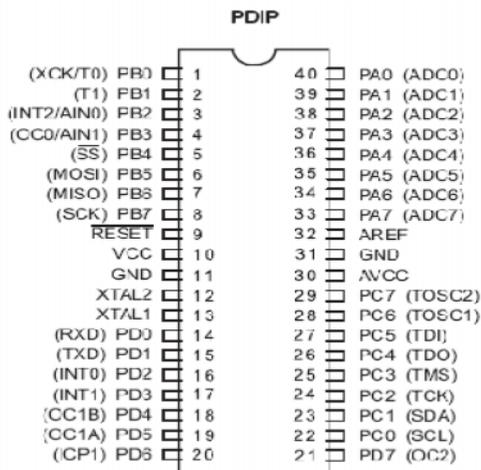
Secara garis besar mikrokontroler *ATmega 16* terdiri dari :

- a. Arsitektur *RISC* dengan *throughput* mencapai 16 *MIPS* pada frekuensi 16Mhz.
- b. Memiliki kapasitas *Flash memori* 16 *Kbyte*, *EEPROM* 512 *Byte*, dan *SRAM* 1 *Kbyte*
- c. Saluran *I/O* 32 buah, yaitu *port A*, *port B*, *port C*, dan *port D*.
- d. *CPU* yang terdiri dari 32 buah *register*.
- e. *User* interupsi *internal* dan eksternal
- f. *Port* antarmuka *SPI* dan *Port USART* sebagai komunikasi *serial*
- g. Fitur *Peripheral*

- Dua buah 8 bit timer/counter dengan prescaler terpisah dan mode compare
- Satu buah 16 bit timer/counter dengan prescaler terpisah, mode compare, dan mode capture
- Real time counter dengan Oscillator tersendiri
- Empat kanal PWM dan Antarmuka komparator analog
- 8 kanal, 10 bit ADC
- Byte-oriented Two-wire Serial Interface Watchdog timer dengan Oscillator internal. (Winoto, 2008:7)

2.6.1.2 Konfigurasi Pin ATmega 16

ATmega 16 memiliki 32 pin saluran I/O yang digunakan untuk input/output. Pin-pin tersebut terdiri dari 8 pin masing-masing pada Port A, Port B, Port C, dan Port D. Dalam komunikasi serial, maka hanya Port D yang dapat digunakan karena fungsi khusus yang dimilikinya.



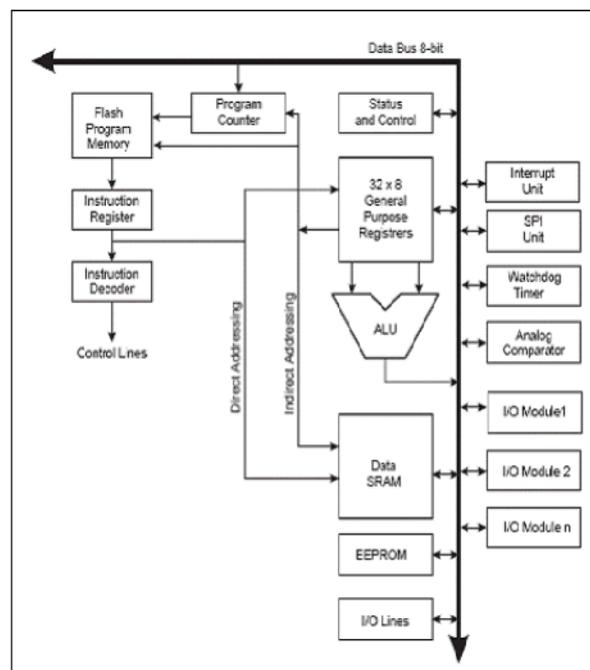
Gambar 2.3 Konfigurasi pin atmega 16

Sumber: Prawiroedjo, 2008

Berikut ini adalah penjelasan fungsi dasar dari pin-pin pada mikrokontroler ATmega 16, yaitu :

1. VCC merupakan pin yang berfungsi sebagai masukan catu daya
2. GND merupakan pin ground

3. *Port A (PA0 – PA7)* merupakan *pin input/output 2 arah (full duplex)* dan selain itu merupakan *pin masukan ADC*
4. *Port B (PB – PB7)* merupakan *pin input/output 2 arah (full duplex)* dan selain itu merupakan *pin khusus*
5. *Port C (PC0 – PC7)* merupakan *pin input/output 2 arah (full duplex)* dan selain itu merupakan *pin khusus*
6. *Port D (PD0 – PD7)* merupakan *pin input/output 2 arah (full duplex)* dan selain itu merupakan *pin khusus*
7. *RESET* merupakan *pin* yang digunakan untuk me-*reset* mikrokontroler
8. *XTAL1* dan *XTAL2* merupakan *pin masukan external Clock*
9. *AVCC* merupakan *pin masukan tegangan untuk ADC*
10. *AREF* merupakan *pin masukan tegangan referensi untuk ADC*.



Gambar 2.4 Arsitektur mikrokontroler *atmega 16*

Pada mikrokontroler *ATmega 16* memiliki 4 buah *port*, yaitu *port A*, *port B*, *port C*, *port D*. *Port A* merupakan *port* yang digunakan sebagai *input ADC* mikrokontroler. Sedangkan *port B*, *port C*, *port D* memiliki fungsi khusus di antaranya :

Tabel 2.1 Fungsi khusus port B

Pin	Fungsi Khusus
PB0	XCK (USART External Clock Input/Output) T0 (Timer/Counter0 External Counter Input)
PB1	T1 (Timer/Counter1 External Counter Input)
PB2	INT2 (External Interupt 2 Input) AIN0 (Analaog Comparator Negative Input)
PB3	OC0 (Timer/Counter0 Output Compare Macth Output) AIN1 (Analaog Comparator Negative Input)
PB4	SPI Slave Select Input)
PB5	MOSI (SPI Bus Master Output /Slave Input)
PB6	MISO (SPI Bus Master Input/Slave Output)
PB7	SCK (SPI Bus Serial Clock)

Tabel 2.2 Fungsi khusus port C

Pin	Fungsi Khusus
PC0	SCL (Two-wire Serial Bus Clock Line)
PC1	SDA (Two-wire Serial BusData Input/Output Line)
PC2	TCK (Joint Test Action Group Test Clock)
PC3	TMS (JTAG Test Mode Select)
PC4	TDO (JTAG Data Out)
PC5	TDI (JTAG Test Data In)
PC6	TOSC1 (Timer Oscillator pin 1)
PC7	TOSC2 (Timer Oscillator pin 2)

Tabel 2.3 Fungsi khusus port D

Pin	Fungsi Khusus
PD0	RXD (USART Input Pin)
PD1	TXD (USART Output Pin)
PD2	INT0 (External Interupt 0 Input)
PD3	INT1 (External Interupt 1 Input)
PD4	OC1B (Timer/Counter1 Output Compare B Macth Output)
PD5	OC1A (Timer/Counter1 Output Compare A Macth Output)
PD6	ICP (Timer/Counter1 Input Capture Pin)
PD7	OC2 (Timer/Counter2 Output Compare Macth Output)

2.6.2 Mikrokontroler ATtiny 2313

Menurut Winoto (2006) Mikrokontroler ATtiny 2313 merupakan mikrokontroler 8 bit AVR dengan kapasitas memori maksimum sebesar 2 Kbytes yang tersimpan di dalam Memory Flash-nya. ATtiny 2313 merupakan chip IC produksi ATMEL yang termasuk golongan single chip microcontroller, di mana semua rangkaian termasuk memori dan I/O tergabung dalam satu paket IC. Dalam

pemrogramannya mikrontroler ini dapat dijalankan menggunakan 2 bahasa yaitu bahasa *Assembly* atau bahasa *C*. Sehingga memungkinkan pengguna dapat meng-*optimal*-kan kinerja sistem yang dibuat secara fleksibel.

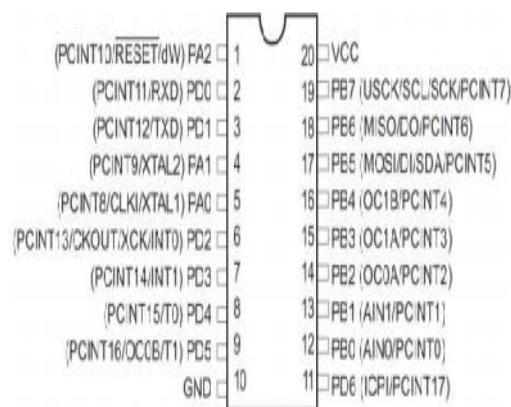


Gambar 2.5 Mikrokontroler *attiny 2313*

Sumber : Anto, B. 2011

2.6.2.1 Konfigurasi *Pin ATtiny 2313*

IC ATtiny 2313 ada 2 jenis yaitu jenis *PDIP/SOIC* (berbentuk prisma segi empat) dan jenis *VQFN/MLF* (berbentuk kotak) yang pada dasarnya memiliki fungsi yang sama, hanya saja memiliki bentuk yang berbeda.



Gambar 2.6 Konfigurasi *pin attiny 2313*

Sumber : Anto, B. 2011

Secara keseluruhan, mikrokontroler *ATtiny 2313* memiliki total *pin* yaitu 20 *pin*. Berikut ini adalah penjelasan dari konfigurasi *pin-pin* tersebut :

1. *VCC* merupakan *pin* yang berfungsi sebagai masukan catu daya sebesar 5 *Volt*
2. *GND* merupakan *pin ground*. Referensi nol suplai tegangan

3. *Port A (PA0 – PA2)*

Port A hanya terdapat 3 buah *pin* atau 3 *bit pin I/O.*, di mana *pin-pin* tersebut digunakan untuk keperluan membuat sistem minimum. *PA0* dan *PA1* digunakan untuk *input Clock* Kristal serta *PA2* digunakan sebagai *input* tombol *reset*.

4. *Port B (PB0 – PB7)*

Port B terdapat 8 buah *pin* atau 8 *bit pin I/O*. Pada *port B* terdapat *port SPI* (*Serial Peripheral Interface*) yaitu *pin* komunikasi untuk men-*download* program secara *serial synchronous* dari komputer ke mikrokontroler. *Pin-pin* tersebut adalah *MOSI* (*Port B5*), *MISO* (*Port B6*), dan *SCK* (*Port B7*).

5. *Port D (PD0 – PD6)*

Port D memiliki 7 buah *pin* atau 7 *bit pin I/O*

6. *Reset*

Reset berfungsi untuk menyusun ulang *routing* program dari awal. *Reset* biasanya bersifat *Active Low* (Aktif Rendah) yaitu aktif saat logika bernilai 0. Sinyal *Low* pada *pin* dengan lebar minimum 1,5 mikrodetik ini akan membawa mikrokontroler ke kondisi *Reset* meskipun *Clock* tidak *running*. Sinyal dengan lebar kurang dari 1,5 mikrodetik tidak akan menjamin terjadinya kondisi *Reset*.

7. *XTAL1* merupakan masukan ke *inverting Oscillator amplifier* dan *input* ke *internal Clock operating circuit*

8. *XTAL2* merupakan *output* dari *inverting Oscillator amplifier*.

(Winoto, 2006:8)

2.7 Sensor *pH*

Sensor *pH* merupakan suatu alat yang digunakan untuk mengukur kadar asam dan basa dalam suatu larutan. Prinsip kerja dari alat ini yaitu didasarkan pada potensial elektro kimia yang terjadi antara larutan yang terdapat di dalam elektroda gelas (membran gelas) dengan larutan yang terdapat di luar elektroda gelas. Hal ini dikarenakan lapisan tipis dari gelembung kaca akan berinteraksi dengan *ion hydrogen* yang ukurannya relatif aktif dan kecil, elektroda gelas tersebut akan mengukur potensial elektro kimia dari *ion hydrogen*.



Gambar 2.7 Sensor *ph*

2.8 LCD (*Liquid Crystal Display*)

LCD (Liquid Crystal Display) adalah satu layar bagian dari modul peraga yang menampilkan karakter yang diinginkan. Layar *LCD* menggunakan 2 buah bahan yang dapat mempolarasikan dan Kristal cair di antara kedua lembaran tersebut. Arus listrik yang melewati cairan menyebabkan Kristal merata sehingga cahaya tidak dapat melalui setiap Kristal, seperti pengaturan cahaya, menentukan apakah cahaya dapat melewati apa tidak. Sehingga dapat mengubah bentuk Kristal cairnya membentuk tampilan angka atau huruf pada layar.

Banyak kegunaan *LCD* dalam perancangan sistem atau alat dengan mikrokontroler. Dalam pembuatan alat ini, *LCD* berfungsi sebagai *output* yang dapat menampilkan nilai sensor dan menampilkan teks. Ukuran *LCD* yang biasanya digunakan adalah *LCD* 16x2, seperti pada gambar berikut ini.



Gambar 2.8 Konfigurasi *pin lcd*

Pada konfigurasi *pin LCD*, *LCD* memiliki 16 *pin* yang masing-masing memiliki fungsi yang berbeda. Berikut ini adalah table fungsi *pin LCD*.

Tabel 2.4 Fungsi *pin lcd*

<i>Pin</i>	Simbol	Logika	Keterangan
1	<i>V_{ss}</i>	-	Catu Daya 0 Volt (<i>Ground</i>)
2	<i>V_{cc}</i>	-	Catu Daya 5 Volt
3	<i>V_{ee}</i>	-	Catu Daya untuk <i>LCD</i>
4	<i>RS</i>	<i>H/L</i>	H: Masukan Data, L: Masukan Instruksi
5	<i>R/W</i>	<i>H/L</i>	H: Baca (<i>Read</i>), L: Tulis (<i>Write</i>)
6	<i>E</i>	<i>H/L (L)</i>	<i>Enable Signal</i>
7	<i>DB0</i>	<i>H/L</i>	Data Bit 0
8	<i>DB1</i>	<i>H/L</i>	Data Bit 1
9	<i>DB2</i>	<i>H/L</i>	Data Bit 2
10	<i>DB3</i>	<i>H/L</i>	Data Bit 3
11	<i>DB4</i>	<i>H/L</i>	Data Bit 4
12	<i>DB5</i>	<i>H/L</i>	Data Bit 5
13	<i>DB6</i>	<i>H/L</i>	Data Bit 6
14	<i>DB7</i>	<i>H/L</i>	Data Bit 7
15	<i>V + BL</i>	-	<i>Backlight</i> 4-4, 2 Volt ; 50-200 mA
16	<i>V - BL</i>	-	<i>Backlight</i> 0 Volt (<i>Ground</i>)

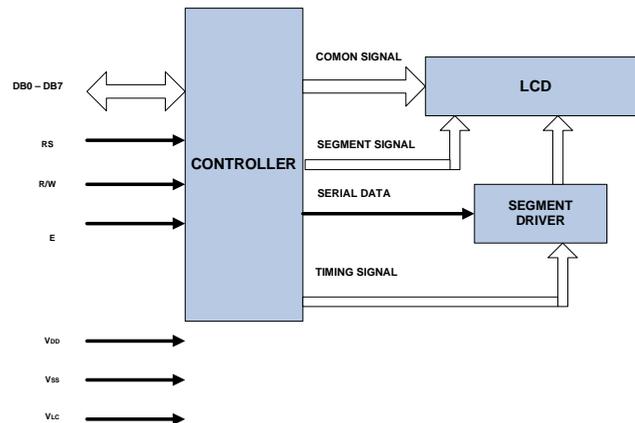
2.8.1 *LCD (Liquid Crystal Display) MI632*

LCD MI632 terdiri dari 2 bagian utama, yaitu bagian pertama merupakan *panel LCD* sebagai media penampil informasi dalam bentuk huruf, angka, dan baris yang pada masing-masing baris mampu menampung 16 huruf atau angka. Pada bagian kedua, yaitu sistem yang dibentuk oleh mikrokontroler yang ditempel di balik *panel LCD* serta berfungsi sebagai pengatur tampilan informasi dan komunikasi *LCD* dengan mikrokontroler.

LCD MI632 memiliki karakteristik seperti berikut ini :

1. Memiliki 16 karakter, yaitu 2 baris tampilan Kristal cair (*LCD*) dari matriks titik
2. *Duty Ratio*, yaitu 1/16
3. Memiliki *ROM* pembangkit karakter untuk 192 tipe karakter (bentuk karakternya 5x7 matriks titik)
4. Memiliki 2 jenis *RAM*, yaitu *RAM* pembangkit karakter dan *RAM* data tampilan
5. *RAM* pembangkit karakter untuk 8 tipe karakter program tulis dengan bentuk 5x7 matriks titik
6. Ram data tampilan berbentuk 80 x 8 matriks titik (maksimum 80 karakter)
7. Memiliki pembangkit *Clock internal*

8. Memiliki sumber tegangan tunggal +5 volt
9. Memiliki rangkaian otomatis *Reset* saat daya dinyalakan
10. Memiliki jangkauan suhu pengoperasian 0 sampai 50 derajat



Gambar 2.9 Diagram blok pengendali *lcd*

Pada gambar di atas, data *input* pada *LCD* yang berupa 8 bit data (*D0 – D7*) diterima terlebih oleh mikrokontroler di dalam *LCD* yang berfungsi sebagai pengatur data *input* sebelum ditampilkan oleh *LCD* serta dilengkapi dengan *input* *E*, *R/W*, dan *RS* yang digunakan sebagai pengendali mikrokontroler. Pada proses pengiriman data yaitu $R/W=1$ dan proses pengambilan data yaitu $R/W=0$.

Selain itu, *pin RS* pada *LCD* digunakan untuk membedakan jenis data yang dikirim. Jika $RS=0$ maka data yang dikirim adalah berupa perintah untuk mengatur kerja modul *LCD*, sedangkan jika $RS=1$ maka data yang dikirim adalah berupa kode *ASCII* yang ditampilkan. Demikian juga pada saat pengambilan data, jika $RS=0$ maka data yang diambil dari modul merupakan data status yang mewakili aktivitas modul *LCD*, sedangkan pada saat $RS=1$ maka data yang diambil merupakan kode *ASCII* dari data yang ditampilkan.

2.9 Baterai

Baterai adalah alat listrik-kimiawi yang menyimpan energi dan mengeluarkan tenaganya dalam bentuk listrik. Baterai ada yang berbentuk tabung atau kotak. Ada juga yang dinamakan *rechargeable battery* yaitu baterai yang dapat diisi ulang, seperti yang biasa terdapat pada telepon genggam. Baterai sekali

pakai disebut juga dengan baterai primer, sedangkan baterai isi ulang disebut dengan baterai sekunder.



Gambar 2.10 Baterai

2.10 Bahasa Pemrograman

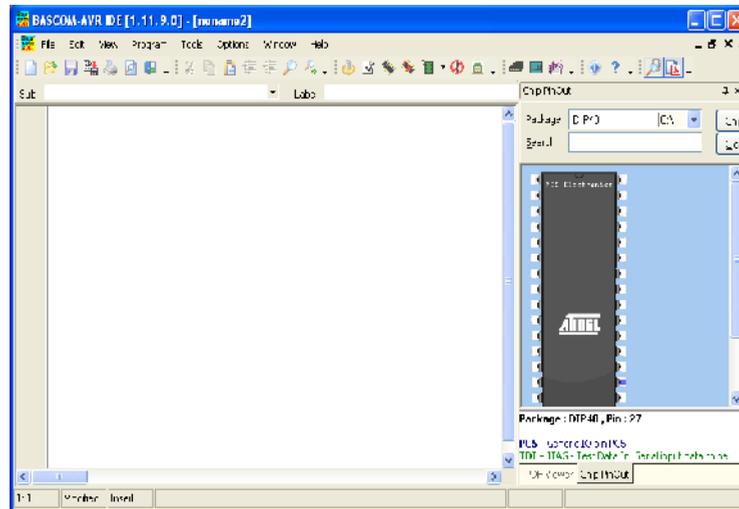
Secara umum bahasa pemrograman mikrokontroler adalah bahasa tingkat rendah yaitu bahasa *assembler*, di mana setiap mikrokontroler memiliki bahasa pemrograman yang berbeda-beda. Karena banyaknya hambatan dalam penggunaan bahasa *assembler*, maka mulai dikembangkan *compiler* atau penerjemah untuk bahasa tingkat tinggi. (Winoto, 2008)

Pada keluarga *ATmega* atau mikrokontroler AVR, bahasa tingkat tinggi yang banyak dikembangkan antara lain *Basic*, *Pascal*, dan bahasa *C*. Dalam melakukan pemrograman, banyak *software* yang digunakan antara lain *AVR Studio-4* dan *BASCOM-AVR*. Dalam pembuatan antarmuka *monitoring* kadar *pH* air penulis menggunakan bahasa pemrogramana *Basic* dengan *software BASCOM-AVR* dan *Visual Basic 6*.

2.10.1 Basic Compiler AVR (BASCOM-AVR)

BASCOM-AVR adalah program *Basic Compiler* berbasis *Windows* untuk mikrokontroler keluarga AVR. *BASCOM-AVR* merupakan pemrograman dengan bahasa tingkat tinggi “*BASIC*” yang dikembangkan dan diluncurkan oleh *MCS Electronics* sehingga mudah dimengerti atau diterjemahkan. Dalam program *BASCOM-AVR* terdapat beberapa kemudahan dalam membuat program *software ATmega 16*, seperti program simulasi yang sangat berguna untuk melihat simulasi hasil dari program yang telah dibuat sebelum program di-*download* ke IC atau mikrokontroler.

Berikut ini tampilan pada jendela program *BASCOM-AVR* sebelum program *BASCOM* dijalankan.



Gambar 2.11 Tampilan jendela program *bascom-avr*

Berikut ini adalah tabel 2.5 yang menunjukkan kelebihan dan kekurangan *BASCOM-AVR*.

Tabel 2.5 Kelebihan dan kekurangan *bascom-avr*

Kelebihan	Kekurangan
Merupakan bahasa tingkat tinggi	Merupakan <i>software proprietary</i> (tertutup)
Bersifat <i>user friendly</i>	
Memiliki <i>simulator</i>	Lisensi nya berbayar
Logika yang digunakan manusiawi	

2.10.1.1 Menu Bar *BASCOM-AVR*

Menu pada *menu bar* hampir sama dengan *software* lain yang berbasis *Windows* sebagai *operating system*. *Menu bar* pada *BASCOM-AVR* terdiri dari *sub menu* *New*, *Open*, *Save* dan *Close*.



Gambar 2.12 *Menu bar bascom-avr*

2.10.1.2 Tool Bar BASCOM-AVR

Ikon *command* atau perintah pada *tool bar* adalah ikon yang mewakili perintah-perintah *sub menu* dan terdapat pada *menu bar*. Perintah *sub menu* yang ditampilkan di *tool bar* dalam bentuk ikon berguna untuk mempercepat akses melalui *mouse*. Pada *tool bar BASCOM-AVR* tersebut Anda bisa menambahkan dan mengurangi ikon-ikon *command* yang diperlukan untuk mempercepat penggunaan.



Gambar 2.13 Tool bar bascom-avr

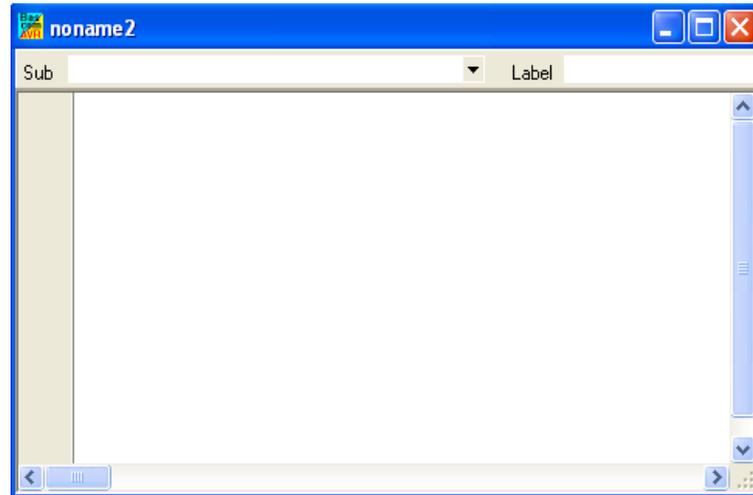
Berikut ini adalah penjelasan ikon-ikon *tool bar BASCOM-AVR*.

Tabel 2.6 Ikon tool bar bascom-avr

Ikon	Nama	Fungsi	Shortcut
	<i>File New</i>	Membuat <i>file</i> baru	<i>Ctrl + N</i>
	<i>Open</i>	Membuka <i>file</i>	<i>Ctrl + O</i>
	<i>Save</i>	Menyimpan <i>file</i>	<i>Ctrl + S</i>
	<i>Save As</i>	Menyimpan <i>file</i> dengan nama lain	-
	<i>Print</i>	Mencetak dokumen	<i>Ctrl + P</i>
	<i>Print Preview</i>	Melihat tampilan dokumen sebelum dicetak	-
	<i>Exit</i>	Keluar dari program	-
	<i>Compile Program</i>	Mengkompilasi program yang telah dibuat	<i>F7</i>
	<i>Check Syntax</i>	Memeriksa kesalahan sintaks	<i>Ctrl + F7</i>
	<i>Show Result</i>	Menampilkan hasil kompilasi	<i>Ctrl + W</i>

2.10.1.3 Text Editor

Text editor adalah tempat menulis atau membuat program. Setelah *text editor* ditampilkan maka ketikkan program.

Gambar 2.14 Jendela *text editor*

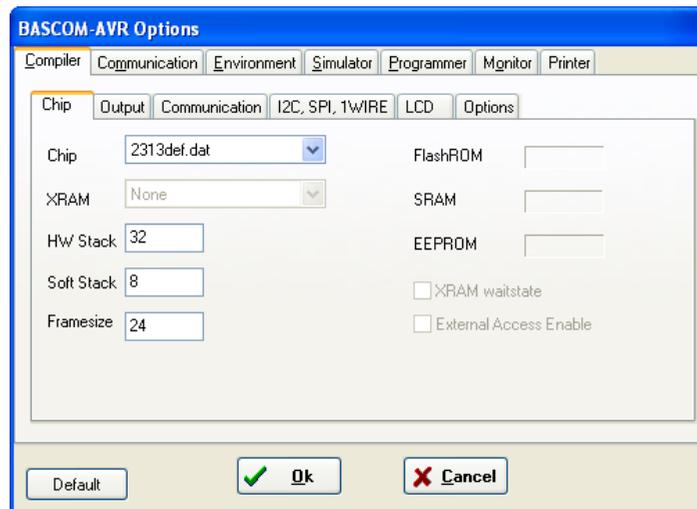
Berikut ini adalah informasi yang akan ditampilkan dari *menu show result* yaitu:

Tabel 2.7 Informasi *menu show result*

Informasi	Keterangan
<i>Compiler</i>	Versi dari <i>compiler</i> yang digunakan
<i>Processor</i>	<i>Processor</i> yang dipilih
<i>Date and Time</i>	Tanggal dan waktu kompilasi
<i>Flash Used</i>	Persentase <i>flash ROM</i> yang terisi program
<i>Baud rate and Error</i>	<i>Baud rate</i> yang dipilih dan Kristal yang <i>error</i> . Nilai <i>baud</i> yang di- <i>set</i> dengan nilai yang sebenarnya.
<i>Stock Start</i>	Lokasi awal <i>stack pointer</i> memori
<i>RAM Start</i>	Lokasi awal <i>external RAM</i>
<i>LCD Mode</i>	<i>Mode LCD</i> yang digunakan 4 bit atau 8 bit

2.10.1.4 *Compiler*

BASCOM-AVR menyediakan pilihan *menu compiler* untuk memodifikasi pilihan-pilihan pada saat kompilasi. Dengan memilih *menu compiler* tersebut maka akan tampil jendela *BASCOM-AVR* sebagai berikut.

Gambar 2.15 Jendela *options bascom-avr*

Keterangan dari gambar di atas dapat dilihat pada tabel *menu* pilihan *BASCOM-AVR* berikut ini.

Tabel 2.8 *Menu* pilihan *bascom-avr*

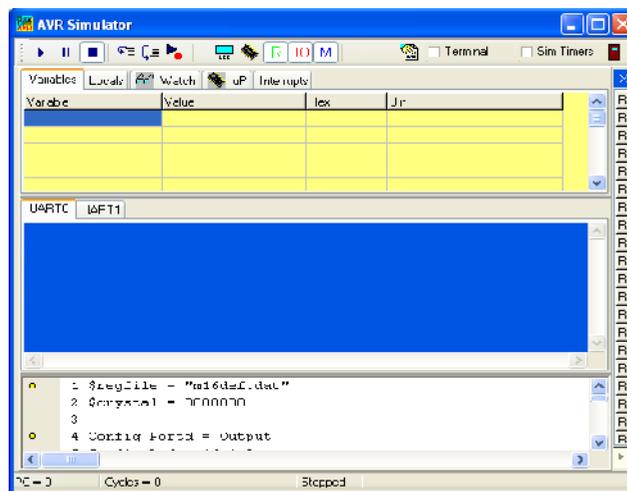
<i>Tab Menu</i>	<i>Options</i>	<i>Keterangan</i>
<i>Chip</i>	<i>Chip</i>	Mikrokontroler yang digunakan sebagai contoh <i>2313def.dat</i> untuk <i>ATtiny 2313</i>
	<i>XRAM</i>	Jika menggunakan <i>external RAM</i> nilai ini bisa ditampilkan
	<i>HW Stack</i>	<i>Stack memory hardware</i> , setiap <i>Go sub</i> membutuhkan 2 <i>byte</i> . Jika menggunakan interupsi maka naikan nilainya
	<i>Soft Stack</i>	<i>Stack software</i> , nilai <i>default</i> -nya 8
	<i>Flash ROM</i>	Nilai <i>flash ROM chip</i> , <i>chip</i> yang dipilih
	<i>SRAM</i>	Nilai <i>RAM internal chip</i> yang dipilih
	<i>EEPROM</i>	Nilai <i>EEPROM chip</i> yang dipilih
<i>Output</i>		<i>File output</i> yang akan dihasilkan dalam proses kompilasi
<i>Communication</i>	<i>Baud rate 0</i>	Nilai <i>baud rate</i> yang digunakan dalam komunikasi <i>serial</i>
	<i>Frekuensi</i>	Nilai osilator yang digunakan
	<i>Error</i>	<i>Error</i> antara <i>baud rate</i> yang dipilih dengan nilai sebenarnya. Hal ini tergantung pada osilator yang dipilih
<i>SDA, SCL, SPI, I Wire</i>	<i>SDA</i>	<i>Pin</i> yang berfungsi untuk data <i>serial</i> dalam komunikasi <i>I2C</i>
	<i>SCL</i>	<i>Pin</i> yang berfungsi untuk data <i>Clock</i> dalam komunikasi <i>I2C</i>
	<i>SPI</i>	<i>Pin</i> yang berfungsi untuk komunikasi <i>I Wire</i>

	<i>I Wire</i>	<i>Pin</i> untuk komunikasi <i>serial</i> sinkron
<i>LCD</i>		Pemilihan <i>port</i> yang digunakan untuk tampilan <i>LCD</i> , jenis <i>LCD</i>

2.10.1.5 Program Simulasi

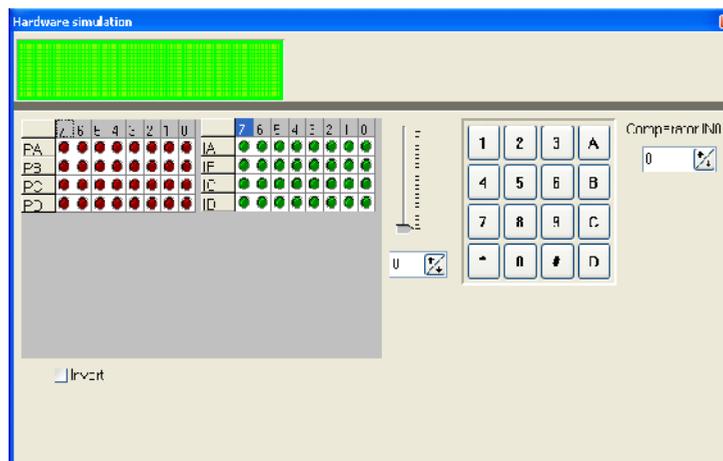
BASCOM-AVR menyediakan pilihan yang dapat mensimulasikan program. Agar Anda dapat menjalankan *simulator* ini, *file DBG* dan *OBJ* harus dipilih pada *menu Options Compiler Output*. Program simulasi bertujuan untuk menguji suatu aplikasi yang dibuat dengan melihat pergerakan *LED* atau *LCD* sebagai indikator program mikrokontroler.

Berikut ini adalah gambar yang menunjukkan tampilan dari program simulasi, yaitu :



Gambar 2.16 Tampilan program simulasi

Tekan tombol  untuk memulai simulasi dan untuk memberhentikan simulasi atau menahan proses simulasi gunakan tombol di sebelahnya. Untuk dapat melihat perubahan data pada setiap *port* atau ketika kita ingin memberikan *input* pada *pin-pin* tertentu dari mikrokontroler, maka gunakan tombol  untuk menampilkan jendela sebagai berikut.

Gambar 2.17 Jendela simulasi *lcd*

Untuk memberikan *input* pada *pin-pin* tertentu kita tinggal menekan *LED* yang kita inginkan, maka program simulasi ini akan melakukan program sedang disimulasikan. Misalkan kita menggunakan *port PI.7* sebagai *input* maka *LED* pada kolom 7 dan baris *PI* kita tekan ketika program telah dijalankan.

2.10.1.6 Tipe Data *BASCOM-AVR*

Setiap variabel dalam *BASCOM* memiliki tipe data yang menunjukkan daya tampung variabel tersebut, hal ini berhubungan dengan penggunaan memori dari mikrokontroler.

Tabel 2.9 Tipe data *bascom-avr*

Tipe	Ukuran	Range
<i>Bit</i>	1/8	-
<i>Byte</i>	1	0 – 255
<i>Integer</i>	2	-32,768 – 32,767
<i>Word</i>	2	0 – 65535
<i>Long</i>	4	-2147483648
<i>Single</i>	4	-
<i>String</i>	s/d 254	-

2.10.1.7 Variabel

Variabel dalam sebuah program berfungsi sebagai tempat penyimpanan data atau penampung data sementara, misalnya menampung hasil perhitungan, menampung data hasil pembacaan *register* dan lain-lain. Variabel

merupakan *pointer* yang menunjuk pada alamat memori fisik di mikrokontroler.

Dalam *BASCOM* ada beberapa aturan dalam penamaan sebuah variabel, yaitu :

- Nama variabel maksimum terdiri atas 32 karakter
- Karakter bisa berupa angka atau huruf
- Nama variabel harus dimulai dengan huruf
- Variabel tidak boleh menggunakan kata-kata yang digunakan oleh *BASCOM* sebagai perintah, pernyataan, *internal register* dan nama *operator* (*AND*, *OR*, *DIM*, dan sebagainya).

Sebelum variabel digunakan maka variabel tersebut harus dideklarasikan terlebih dahulu. Dalam *BASCOM*, ada beberapa cara untuk mendeklarasikan sebuah variabel. Cara pertama adalah menggunakan pernyataan '*DIM*' diikuti nama dan tipe datanya.

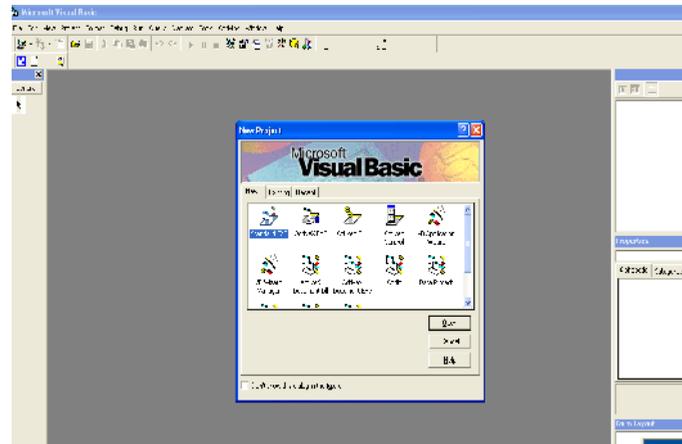
2.10.2 Microsoft Visual Basic 6.0

Menurut Retna dan Catur (2004: 3) *Visual Basic* adalah perangkat lunak untuk menyusun program aplikasi yang bekerja dalam lingkungan sistem operasi *Windows*. Dengan *Visual Basic* kita bisa memanfaatkan kemampuan *Windows* secara *optimal*. Dengan kecanggihan yang ditawarkan oleh *Visual Basic* kita akan merasakan begitu mudahnya menyusun program aplikasi dengan tampilan grafis yang menawan dalam waktu yang relatif singkat.

Bahasa pemrograman *Visual Basic* yang dikembangkan oleh *Microsoft* sejak tahun 1991 merupakan pengembangan dari pendahulunya yaitu bahasa pemrograman *BASIC* (*Beginner's All-purpose Symbolic Instruction Code*) yang dikembangkan pada era 1950-an. *Visual Basic* merupakan salah satu *Development Tool* yaitu alat bantu untuk membuat berbagai macam program komputer, khususnya yang menggunakan sistem operasi *Windows*.

Microsoft Visual Basic 6.0 merupakan salah satu *tool* untuk pengembangan aplikasi yang banyak diminati oleh orang. Di sini *Microsoft Visual Basic 6.0* menawarkan kemudahan dalam pembuatan aplikasi dan dapat menggunakan komponen-komponen yang telah disediakan. Dengan menggunakan *Microsoft Visual Basic 6.0* kita bisa menghasilkan berbagai macam jenis program.

Dari aplikasi yang mengintegrasikan *database*, jaringan, *office automation*, dan *web application*.



Gambar 2.18 Tampilan utama *microsoft visual basic 6.0*

Adapun beberapa keuntungan mengapa menggunakan *Microsoft Visual Basic 6.0* daripada bahasa pemrograman yang lain di antaranya :

1. Tampilan grafis (*under Windows*) sehingga lebih “bersahabat”.
2. Cara pemrograman relatif lebih mudah sehingga cocok untuk segala tingkat *programmer*.
3. Hubungan dengan perangkat luar (*hardware*) tidak begitu rumit sehingga cukup mudah untuk mengimplementasikan sebagai pengendali peralatan elektronik.

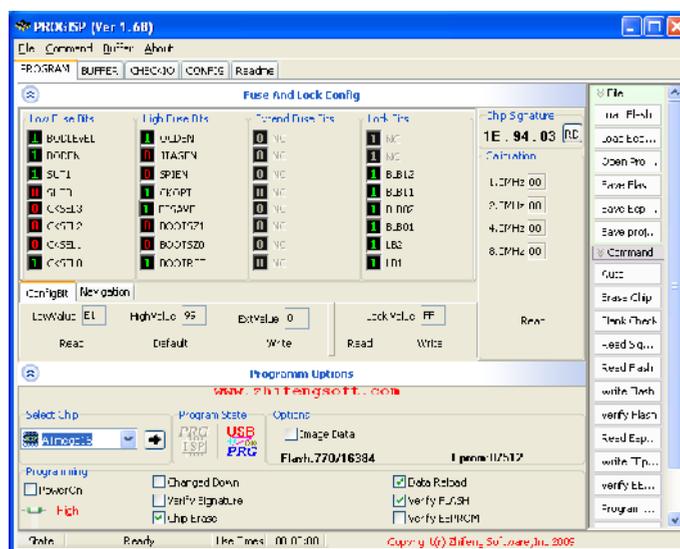
Istilah-istilah pada *Microsoft Visual Basic 6.0* di antaranya yaitu :

1. *Even-Driven*, artinya setiap aplikasi yang ada digerakkan oleh *user*
2. *GUI (Graphical Unit Interface)* adalah pemrograman yang menggunakan tampilan grafik sebagai alat komunikasi dengan pemakainya.
3. *DAO (Data Access Object)* dukungan untuk memudahkan akses terhadap *database* lain.
4. *OLE (Object Linking and Embedding)* kemudahan menghubungkan beberapa aplikasi dalam *Windows*.
5. *OCX*, dukungan untuk menggunakan *tools* tambahan.

6. *DLL (Dynamic Link Library)* adalah kumpulan *library* untuk menggabungkan program dalam *Microsoft Visual Basic 6.0* dengan bahasa pemrograman lainnya.
7. *DDE (Dynamic Data Exchange)*, dukungan kemudahan pertukaran dalam membangun program.

2.11 Progisp Versi 1.68

Cara pengisian program pada *IC* mikrokontroler yaitu dengan menggunakan *software downloader*. Salah satu *software downloader* yaitu *progisp*. *Software* ini digunakan dalam pemrograman mikrokontroler khususnya saat melakukan *download file *.HEX* ke dalam memori mikrokontroler *ATmega 16* Untuk bisa menggunakan *software* ini tentunya Anda harus mempunyai *soft copy software* tersebut, karena *software* ini bersifat *portable* jadi tidak perlu diinstal terlebih dahulu.



Gambar 2.19 *Progisp* versi 1.68

2.12 Kamus Data (*Data Dictionary*)

Kamus data adalah sekumpulan elemen atau simbol yang biasanya digunakan untuk membantu dalam proses penggambaran dan pengidentifikasian setiap *field* dan *file* di dalam suatu sistem.

Tabel 2.10 Simbol-simbol dalam kamus data

No	Simbol	Arti
1	=	Terdiri dari
2	+	Dan (<i>And</i>)
3	()	Pilihan (<i>Optional</i>)
4	[]	Memilih salah satu alternatif
5	**	Komentar, keterangan, catatan
6	@	Kunci (<i>Key Field</i>)
7	/	Pemisahan alternatif simbol

2.13 Konsep Dasar *Database*

Database merupakan seluruh data disimpan dalam basis data pada masing-masing tabel atau *file* sesuai dengan fungsinya, sehingga dengan mudah dapat melakukan penelusuran data yang diinginkan, sehingga akan cepat mendapatkan informasi yang diperlukan.

2.13.1 *Database*

Database merupakan kumpulan *file* yang berisi data yang saling berhubungan dan terorganisir, terpadu, diatur dan disimpan menurut suatu cara tertentu yang memudahkan proses pengambilan kembali.

“Secara praktis basis data (*database*) dapat dianggap sebagai penyusun data yang terstruktur yang disimpan dalam media pengikat (*hard disk*) yang tujuannya adalah agar data tersebut dapat diakses dengan mudah dan cepat.” (Kadir, 2008: 3)

2.14 *Microsoft Access 2007*

Microsoft Access 2007 adalah sebuah *software* yang digunakan untuk *manage database* agar data dapat terorganisasi dengan baik. 4 *object* terpenting dalam *Access*, adalah :

2.14.1 *Komponen Utama (Object)*

Microsoft Access 2007 memiliki komponen utama (*object*) dalam pembuatan *database* yang terdiri dari :

1. *Table*

Table adalah objek utama dalam *database* yang digunakan untuk

menyimpan sekumpulan data sejenis dalam sebuah objek. *Table* terdiri atas :

- a. *Field Name*, merupakan atribut dari sebuah *table* yang menempati bagian kolom.
- b. *Record*, merupakan isi dari *field* atau atribut yang saling berhubungan yang menempati bagian baris.

2. *Form*

Form digunakan untuk mengontrol proses masukan data (*input*), menampilkan data yang tersimpan dalam *table* (*output*), memeriksa dan memperbaharui data.

3. *Query (SQL / Structured Query Language)*

Query merupakan satu dari dua sumber data selain *table* yang berguna untuk menampilkan, mengubah dan menganalisis data. *Query* dibedakan menjadi 2, yaitu :

- a. *DDL (Data Definition Language)* digunakan untuk membuat atau mendefinisikan obyek-obyek *database* seperti membuat tabel, relasi antar tabel dan sebagainya.
- b. *DML (Data Manipulation Language)* digunakan untuk manipulasi *database*, seperti : menambah, mengubah atau menghapus data serta mengambil informasi yang diperlukan dari *database*.

4. *Report*

Report berfungsi untuk menampilkan data yang dirangkum dan mencetak data secara efektif.

2.14.2 Tipe Data

Field - field dalam sebuah tabel harus ditentukan tipe datanya. Ada beberapa tipe data dalam *Microsoft Access 2007*, di antaranya:

1. *Text*

Text digunakan untuk *field alphanumeric* (misal : nama, alamat, kode pos, nomor telp), tiap *field*-nya terdiri dari sekitar 255 karakter.

2. *Memo*

Memo dapat menampung 64000 karakter untuk tiap *field*-nya, tapi tidak bisa diurutkan/diindeks.

3. *Number* (data numerik)

Number digunakan untuk menyimpan data numerik yang akan digunakan untuk proses perhitungan matematis.

4. *Date/time* (nilai tanggal dan waktu)

5. *Currency* (mata uang)

6. *Auto Number* (angka urut dan angka acak)

7. *Yes/No* (menampung 2 jawaban pasti)

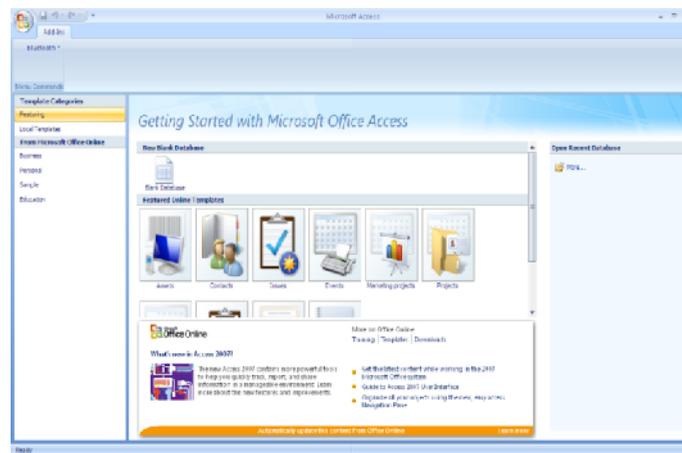
8. *OLE Object* (membuat eksternal *object*)

OLE Object digunakan untuk eksternal objek, seperti *bitmap* atau *file* suara.

9. *Hyperlink* (gabungan dari teks dan bilangan)

10. *Lookup Wizard*

Jika menggunakan tipe data ini untuk sebuah *field*, maka bisa memilih sebuah nilai dari tabel lain atau dari sebuah daftar nilai yang ditampilkan dalam *combo box*.



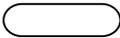
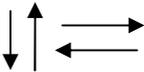
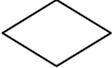
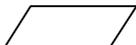
Gambar 2.20 Tampilan utama *microsoft access 2007*

2.15 *Flowchart*

Flowchart adalah penyajian yang sistematis tentang proses dan logika dari kegiatan penanganan informasi atau penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. *Flowchart* menolong analisis

dan *programmer* untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian.

Tabel 2.11 Simbol-simbol *flowchart*

Simbol	Nama	Fungsi
	<i>Terminal</i>	Menyatakan permulaan atau akhir suatu program
	Garis Alir (<i>Flow Line</i>)	Menyatakan jalannya arus suatu proses
	<i>Decision</i>	Perbandingkan pernyataan, penyeleksian data yang memberikan pilihan untuk langkah selanjutnya
	<i>On Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman (halaman yang sama)
	<i>Off Page Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman yang berbeda
	<i>Preparation</i>	Proses inisialisasi atau pemberian harga awal.
	<i>Process</i>	Menyatakan perhitungan / pengolahan data
	<i>Input/Output</i>	Menyatakan proses <i>input</i> atau <i>output</i> tanpa tergantung jenis peralatannya
	<i>Predefined Process (Sub Program)</i>	Menyatakan permulaan <i>sub</i> program / proses menjalankan <i>sub</i> program.