

BAB II

TINJAUAN PUSTAKA

II.1. Pengenalan Mikrokontroller AVR

Sebuah komputer mikro memiliki tiga komponen utama: unit pengolah pusat (CPU=central processing unit), memori dan sistem input/output (I/O) untuk dihubungkan dengan perangkat luar. CPU, yang mengatur sistem kerja komputer mikro, dibangun oleh sebuah mikroprosesor. Memori terdiri atas EEPROM untuk menyimpan program dan RAM untuk menyimpan data. Sistem I/O bisa dihubungkan dengan perangkat luar misalnya sebuah keyboard dan sebuah monitor, bergantung pada aplikasinya. Apabila CPU, memori dan sistem I/O dibuat dalam sebuah chip semikonduktor, maka inilah yang dinamakan mikrokontroller (Usman : 2008).

Secara fisik, kerja dari sebuah mikrokontroler dapat dijelaskan sebagai siklus pembacaan instruksi yang tersimpan di dalam memori. Mikrokontroler menentukan alamat dari memori program yang akan dibaca, dan melakukan proses baca data di memori. Data yang dibaca diinterpretasikan sebagai instruksi. Alamat instruksi disimpan oleh mikrokontroler di register, yang dikenal sebagai *program counter*. (Widodo dan Sigit : 2005)

Mikrokontroler adalah sebuah sistem mikroprosesor dimana didalamnya 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 (Winoto : 2010).

Atmel, salah satu vendor yang bergerak di bidang mikroelektronika, telah mengembangkan AVR (Alf and Vegard's Risc processor) sekitar tahun 1997. Berbeda dengan mikrokontroler MCS51, AVR menggunakan arsitektur RISC (Reduce Instruction Set Computer) yang mempunyai lebar bus data 8 bit. Perbedaan ini bisa dilihat dari frekuensi kerjanya. MCS51 memiliki frekuensi kerja seperdua belas kali frekuensi osilator sedangkan frekuensi kerja AVR sama dengan frekuensi osilator. Jadi dengan frekuensi osilator yang sama, kecepatan AVR dua belas kali lebih cepat dibandingkan dengan kecepatan MCS51. Secara umum AVR dibagi menjadi 4 kelas yaitu ATtiny, AT90Sxx, ATmega, AT86RFxx. Perbedaan antar tipe AVR terletak pada fitur-fitur yang ditawarkan, sementara dari segi arsitektur dan set instruksi yang digunakan hampir sama. (Heryanto & Adi : 2008).

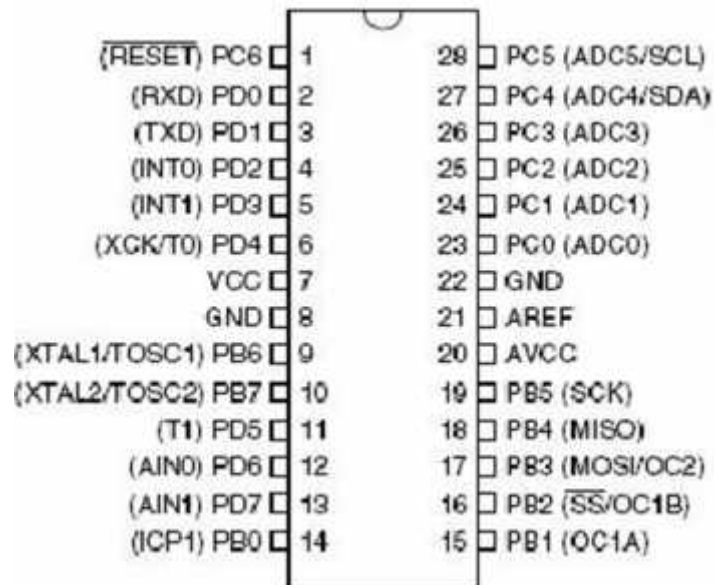
Sedangkan Arduino didefinisikan sebuah platform elektronik yang open source, berbasis pada software dan hardware yang fleksibel dan mudah digunakan, yang ditujukan untuk para seniman, desainer dan setiap orang yang tertarik dalam membuat objek atau lingkungan yang interaktif (Artanto : 2006).

II.1.1. Konfigurasi Pin Mikrokontroler ATMEGA 8

AVR merupakan salah satu jenis mikrokontroler yang ada di dalamnya terdapat berbagai macam fungsi. Perbedaannya dengan mikro yang pada umumnya digunakan seperti MCS51 adalah pada AVR tidak perlu menggunakan oscillator eksternal karena di dalamnya sudah terdapat oscillator internal. Selain itu kelebihan dari AVR adalah melakukan reset. Untuk

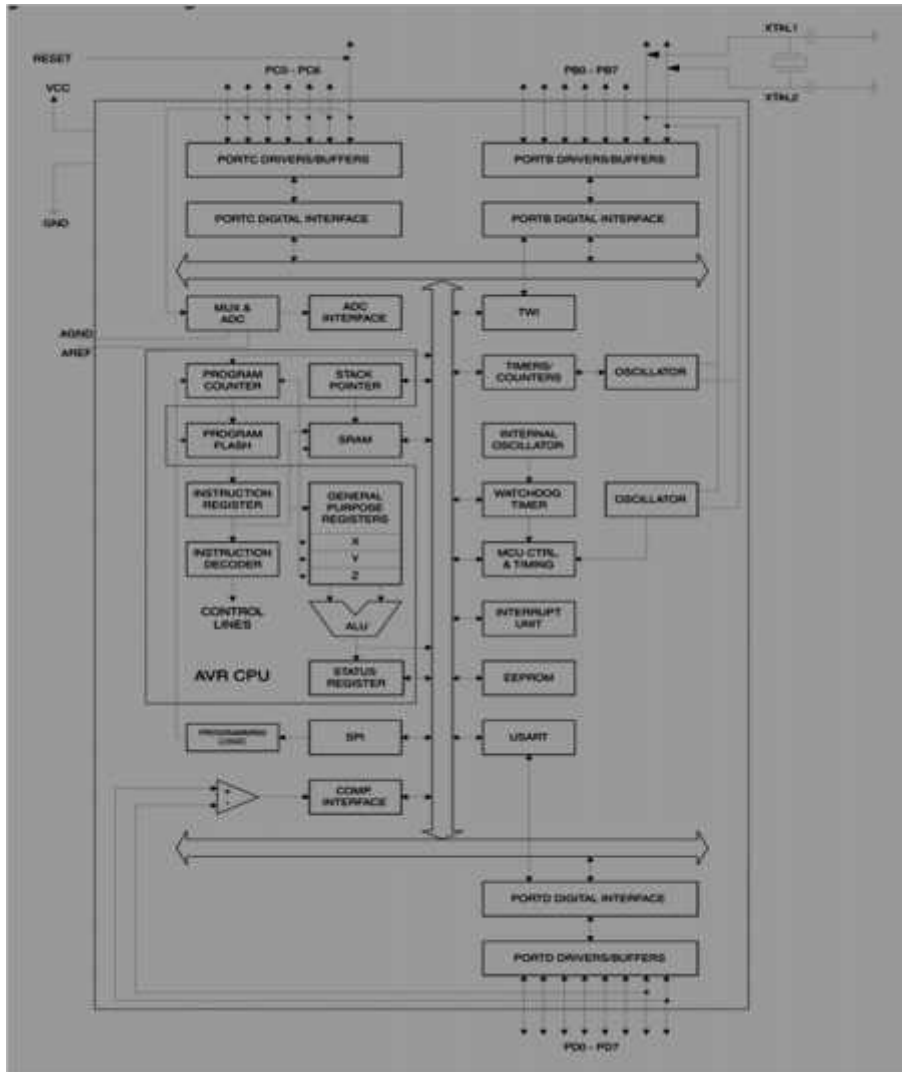
beberapa jenis AVR terdapat beberapa fungsi khusus seperti ADC, EEPROM sekitar 128 bytes sampai dengan 512 bytes.

Dalam hal ini yang digunakan adalah AVR ATmega 8, perbedaannya dengan AVR ATmega 8L hanyalah terletak pada besarnya tegangan yang diperlukan untuk bekerja. Untuk ATmega 8 tipe L dapat bekerja pada tegangan antara 2,7 – 5,5 V sedangkan untuk ATmega 8 hanya dapat bekerja pada tegangan 4,5 – 5,5 V.



Gambar II.1. Konfigurasi PIN Atmega 8

(Sumber:Ardi Winoto:40)



Gambar II.2. Rangkaian Blok Diagram ATMEGA 8

(Sumber: <http://digilib.petra.ac.id/viewer.php?page=1&submit.x=22&submit.y=16&qual=high&submitval=next&fname=%2Fjiunkpe%2Fs1%2Felkt%2F2008%2Fjiunkpe-ns-s1-2008-23403022-10720-parkir-chapter2.pdf>)

ATmega 8 memiliki 28 pin yang masing-masing pin-nya memiliki fungsi yang berbeda-beda baik sebagai port ataupun sebagai fungsi yang lainnya. Berikut akan dijelaskan tentang kegunaan dari kaki pada ATmega 8 (Winoto : 2010) yaitu:

1. VCC

Merupakan supply tegangan untuk digital.

2. GND

Merupakan ground untuk semua komponen yang membutuhkan grounding.

3. Port B

Di dalam port B terdapat XTAL1, XTAL2, TOSC1, TOSC2. Jumlah port B adalah 8 buah pin B 0 sampai dengan pin B 7. Tiap pin dapat digunakan sebagai input dan juga output. Port B merupakan sebuah 8 bit bi-directional I/O port dengan internal pull-up resistor. Sebagai input, pin0pin yang terdapat pada port B yang secara eksternal diturunkan, maka akan mengeluarkan arus jika pull-up resistor diaktifkan. Jika ingin menggunakan tambahan kristal, maka cukup dengan menghubungkan kaki dari kristal ke kaki pada pin port B. Namun jika tidak digunakan, maka cukup dibiarkan saja. Penggunaan kegunaan dari masing-masing kaki ditentukan dari clock fuse setting-nya.

4. Port C

Port C merupakan sebuah 7 bit bi-directional I/O port yang didalam masing-masing pin terdapat pull-up resistor. Jumlah pin-nya hanya 7 buah mulai dari pin C 0 sampai dengan pin C 6. Sebagai keluaran, port C memiliki karakteristik yang sama dalam hal kemampuan menyerap arus ataupun mengeluarkan arus.

5. Reset / PC 6

Jika RSTDISBL Fuse deprogram, maka PC6 akan berfungsi sebagai pin I/O. Untuk diperhatikan juga bahwa pin ini memiliki karakteristik yang berbeda dengan pin-pin yang terdapat pada port C. Namun jika RSTDISBL Fuse tidak diprogram, maka pin ini akan berfungsi sebagai input reset. Dan jika level tegangan yang masuk ke pin ini rendah dan pulsa yang ada lebih pendek dari pulsa minimum, maka akan menghasilkan suatu kondisi reset meskipun clocknya tidak bekerja.

6. Port D

Port D merupakan 8 bit bi-directional I/O dengan internal pull-up resistor. Fungsi dari port ini sama dengan port-port yang lain. Hanya saja pada port ini tidak terdapat kegunaan-kegunaan yang lain. Pada port ini hanya berfungsi sebagai masukan dan keluaran saja atau biasa disebut dengan I/O.

7. AVCC

Pada pin ini memiliki fungsi sebagai supply tegangan untuk ADC. Untuk pin ini harus dihubungkan secara terpisah dengan VCC karena pin ini digunakan untuk analog saja. Bahkan jika ADC pada AVR tidak digunakan, tetap saja disarankan untuk menghubungkan secara terpisah dengan VCC. Cara menghubungkan AVCC adalah melewati low pass filter setelah itu dihubungkan dengan VCC.

8. AREF

Merupakan pin referensi analog jika menggunakan ADC.

II.2. Sensor

Sensor merupakan peralatan atau komponen yang mempunyai peranan penting dalam sebuah sistem pengaturan otomatis. Ketepatan dan kesesuaian dalam memilih sebuah sensor akan sangat menentukan kinerja dari sistem pengaturan secara otomatis.

Sensor adalah suatu peralatan yang berfungsi untuk mendeteksi gejala-gejala atau sinyal-sinyal yang berasal dari perubahan suatu energi seperti energi listrik, energi fisika, energi kimia, energi biologi, energi mekanik dan sebagainya. Contoh: sensor ultrasonik, sensor LDR (*light dependent resistance*) sebagai sensor cahaya, dan lain-lain.

(Sumber:<http://repository.usu.ac.id/bitstream/123456789/26400/4/Chapter%20II.pdf>)

II.2.1. Klasifikasi Sensor

Secara umum berdasarkan fungsi dan penggunaannya sensor dapat dikelompokkan menjadi 3 bagian yaitu:

- a. Sensor thermal (panas)
- b. Sensor mekanis
- c. Sensor optik (cahaya)
- c. Sensor ultrasonik (gelombang)

Sensor thermal adalah sensor yang digunakan untuk mendeteksi gejala perubahan panas/temperature/suhu pada suatu dimensi benda atau dimensi ruang tertentu. Contohnya: *bimetal, termistor, termokopel, RTD, photo transistor, photo diode, photo multiplier, photovoltaic, infrared pyrometer, hygrometer, dsb*

Sensor mekanis adalah sensor yang mendeteksi perubahan gerak mekanis, seperti perpindahan atau pergeseran atau posisi, gerak lurus dan melingkar, tekanan, aliran, level dsb. Contohnya: *strain gauge, linear variable deferential transformer (LVDT), proximity, potensiometer, load cell, bourdon tube, dsb.*

Sensor optik atau cahaya adalah sensor yang mendeteksi perubahan cahaya dari sumber cahaya, pantulan cahaya ataupun bias cahaya yang mengenai benda atau ruangan. Contohnya: *photo cell, photo transistor, photo diode, photo voltaic, photo multiplier, pyrometer optic, dsb.*

Sensor ultrasonik adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek tertentu di depannya, frekuensi kerjanya pada daerah diatas gelombang suara dari 40 KHz hingga 400 KHz.

II.3. Perangkat Keras Pendukung (Support Hardware)

II.3.1. Sensor Ultrasonik

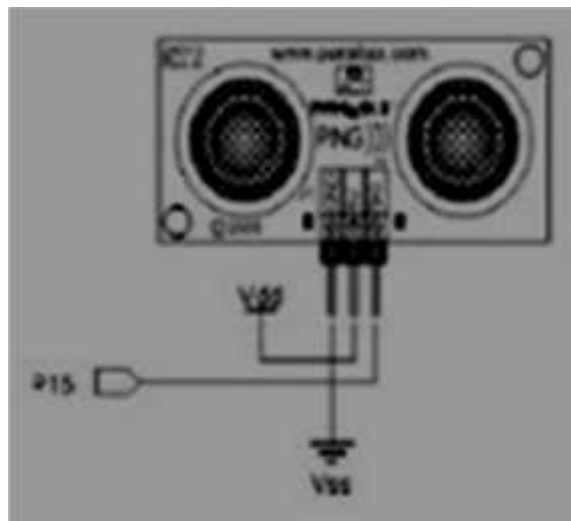
Sensor ultrasonik adalah sensor yang bekerja berdasarkan prinsip pantulan gelombang suara dan digunakan untuk mendeteksi keberadaan suatu objek tertentu di depannya, frekuensi kerjanya pada daerah diatas gelombang suara dari 40 KHz hingga 400 KHz.

Sensor ultrasonik terdiri dari dari dua unit, yaitu unit pemancar dan unit penerima. Struktur unit pemancar dan penerima sangatlah sederhana, sebuah kristal *piezoelectric* dihubungkan dengan mekanik jangkar dan hanya dihubungkan dengan diafragma penggetar. Tegangan bolak-balik yang memiliki frekuensi kerja 40 KHz – 400 KHz diberikan pada plat logam. Struktur atom dari kristal *piezoelectric* akan berkontraksi (mengikat), mengembang atau menyusut terhadap polaritas tegangan yang diberikan, dan ini disebut dengan efek *piezoelectric*. Kontraksi yang terjadi diteruskan ke diafragma penggetar sehingga terjadi gelombang ultrasonik yang dipancarkan ke udara (tempat sekitarnya), dan pantulan gelombang ultrasonik akan terjadi bila ada objek tertentu, dan pantulan gelombang ultrasonik akan diterima kembali oleh oleh unit sensor penerima. Selanjutnya unit sensor penerima akan menyebabkan diafragma penggetar akan bergetar dan efek *piezoelectric* menghasilkan sebuah tegangan bolak-balik dengan frekuensi yang sama.

Besar amplitudo sinyal elektrik yang dihasilkan unit sensor penerima tergantung dari jauh dekatnya objek yang dideteksi serta kualitas dari sensor pemancar dan sensor penerima. Proses *sensing* yang dilakukan pada sensor ini menggunakan metode pantulan untuk menghitung jarak antara sensor dengan obyek sasaran. Jarak antara sensor tersebut dihitung dengan cara mengalikan setengah waktu yang digunakan oleh sinyal ultrasonik dalam perjalanannya dari rangkaian Tx sampai diterima oleh rangkaian Rx, dengan kecepatan rambat dari sinyal ultrasonik tersebut pada media rambat yang digunakannya. Waktu di hitung ketika pemancar aktif dan sampai ada input dari rangkaian penerima dan bila pada melebihi batas waktu tertentu rangkaian

penerima tidak ada sinyal input maka dianggap tidak ada halangan didepannya. Komponen penting yang dipakai dalam membuat alat pengukur volume residu ini adalah sensor ultrasonik, yang berfungsi sebagai pengindra atau pembaca. Sensor itu sendiri berfungsi untuk mengubah resistansi atau hambatan yang mengakibatkan tegangan pembanding (0/1) berubah terhadap tegangan referensi atau tegangan acuan. Sensor tersebut yang menentukan besar kecilnya tegangan yang masuk pada ADC yang kemudian dikonversi menjadi kode-kode biner.

Pemancar sensor ultrasonik ini bekerja apabila diberi trigger dari control MCU (mikrokontroler unit). Trigger berfungsi untuk menghitung (T) atau waktu kapan pantulan dari transmister (Tx) diterima oleh receiver (Rx). Pada saat Rx menerima sinyal dimasukkan dalam rangkaian penguat dan dikonversikan menjadi tegangan kemudian di kirim ke mikrokontroller.



Gambar II.3. Sensor Ultrasonik

(Sumber:<http://repository.usu.ac.id/bitstream/123456789/26400/4/Chapter%20II.pdf>)

II.3.2. LCD (Liquid Crystal Display)

LCD (Liquid Crystal Display atau dapat di bahasa Indonesia-kan sebagai tampilan kristal cair) adalah suatu jenis media tampilan yang menggunakan kristal cair sebagai penampil utama.

LCD dapat memunculkan gambar atau dikarenakan terdapat banyak sekali titik cahaya (piksel) yang terdiri dari satu buah kristal cair sebagai sebuah titik cahaya. Walau disebut sebagai titik cahaya, namun kristal cair ini tidak memancarkan cahaya sendiri. Sumber cahaya di dalam sebuah perangkat LCD adalah lampu neon berwarna putih di bagian belakang susunan kristal cair tadi. Titik cahaya yang jumlahnya puluhan ribu bahkan jutaan inilah yang membentuk tampilan citra. Kutub kristal cair yang dilewati arus listrik akan berubah karena pengaruh polarisasi medan magnetik yang timbul dan oleh karenanya akan hanya membiarkan beberapa warna diteruskan sedangkan warna lainnya tersaring. Dalam menampilkan karakter untuk membantu menginformasikan proses dan *control* yang terjadi dalam suatu program robot kita sering menggunakan LCD juga. LCD yang digunakan untuk pembuatan skripsi ini adalah LCD dengan banyak karakter 16x2, maksudnya 16 menyatakan kolom dan 2 menyatakan baris.



Gambar II.4. LCD 16x2

(Sumber:Ardi Winoto:40)

II.3.3. Buzzer

Buzzer adalah suatu alat yang dapat mengubah sinyal listrik menjadi sinyal suara. Pada umumnya buzzer digunakan untuk alarm, karena penggunaannya cukup mudah yaitu dengan memberikan tegangan input maka buzzer akan mengeluarkan bunyi. Frekuensi suara yang di keluarkan oleh buzzer yaitu antara 1-5 KHz.



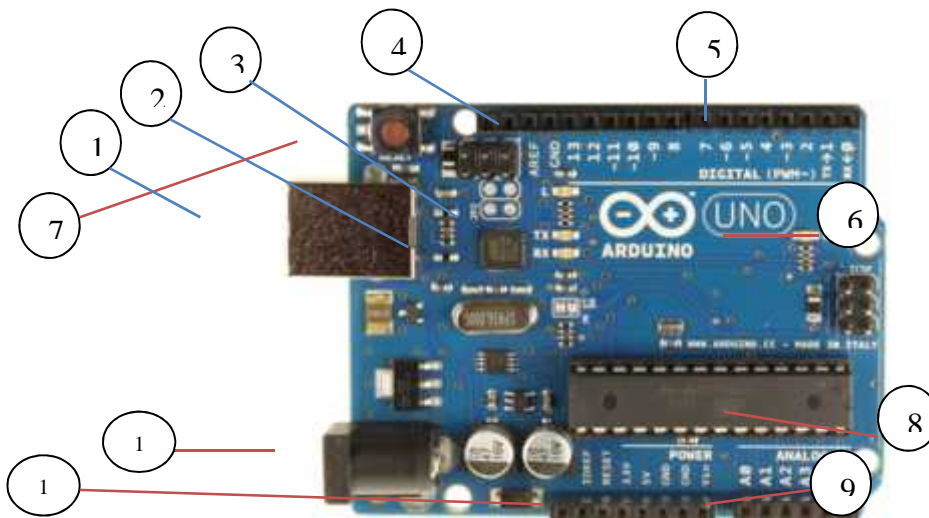
Gambar II.5. Buzzer

(Sumber: <http://www.futurlec.com.au/Buzzers.jsp>)

II.3.4. Minimum Sistem Arduino Uno R3

Beberapa versi yang cukup banyak tersedia di pasaran lokal, diantaranya adalah versi Arduino Duemilanove yang disebut sebagai Arduino 2009, versi Arduino Uno yang disebut sebagai Arduino 2010, dan Arduino Mega yang memiliki kapasitas memori dan kaki I/O yang lebih banyak.

Pada dasarnya, bagian-bagian komponen dari Arduino Duemilanove dan Arduino Uno sama, hanya berbeda pada IC konverter USB ke serialnya.



Gambar II.6. Bagian-bagian Arduino Uno

(Sumber: Dian Artanto:10)

Keterangan:

1. Port USB
2. IC Konverter Serial-USB (ATmega 8 U2)
3. Led untuk test output kaki D13
4. Kaki-kaki input Output Digital (D8 – D13)
5. Kaki-kaki input Output Digital (D0 – D7)
6. LED Indikator catu daya
7. Tombol Reset
8. Mikrokontroler ATmega 8
9. Kaki-kaki input analog (A0 – A5)
10. Kaki-kaki catu daya (5V, GND)
11. Terminal Catudaya (6 – 9V)

II.3.5. Residu (Main Fuel Pump)

Menurut Jeffri Syani selaku seorang Enginer Pembangkit di PT.PLN bahwa residu merupakan bahan bakar cair berat yang mempunyai sifat penguapan yang relatif rendah, kekentalan tinggi dan digunakan sebagai bahan bakar mesin industri *boiler*. Residu dibuat dari *fraksi residu* yang berasal dari *distilasi atmosferik*. Kualitas residu harus diperhatikan agar dalam pemakaiannya aman dan tidak menyebabkan kerusakan / buntu pada *nozzle burner* atau *combustion chamber*.

Digunakannya residu sebagai bahan bakar yang lebih banyak digunakan daripada batubara karena mempunyai keuntungan sebagai berikut:

- Nilai kalori yang cukup tinggi
- Harga per kalori lebih murah

- Biaya operasi dan penanganannya lebih murah
- Tidak mengandung logam berat
- Kadar abu lebih rendah
- Tidak rusak pada penyimpanan dalam waktu yang lama
- Memiliki efisiensi yang tinggi pada sistem pembakaran

Sifat-sifat fisika residu :

1. Sifat umum. Sifat umum adalah sifat yang hasil ujinya dapat menggambarkan atau memperkirakan kualitas suatu produk dan kaitannya dengan pengujian sifat lainnya dengan segera.
2. Sifat pembakaran. Pada dasarnya jika kita membeli bahan bakar, yang kita beli adalah kalori atau panasnya. Panas pembakaran adalah jumlah panas yang dihasilkan dalam satuan btu/lb, kcal/kg atau btu/usg. Jika panas yang dihasilkan dapat diketahui maka jumlah residu yang akan dipakai dapat diketahui.
3. Sifat kemudahan mengalir. Sifat kemudahan mengalir dari residu sangat berpengaruh kepada sistem pemompaan baik pada waktu penyaluran dengan sistem perpipaan maupun dalam pemakaiannya dan pembentukan kabut (*atomizing*) dalam ruang bakar.
4. Sifat kebersihan. Pembakaran dapat mengakibatkan kontaminasi atau tingginya kadar *deposisi karbon* pada sisa pembakaran dan tingginya kadar air. Tingginya *kadar deposisi karbon* dapat menimbulkan kerak pada lubang *nozzle combustion*. Sedangkan tingginya kadar air dapat membuat residu sulit untuk dinyalakan pertama kali dan api yang terbentuk tidak akan stabil atau mudah mati dan panas pembakaran yang dihasilkan akan rendah karena energi yang

dihasilkan dari pembakaran residu akan terserap untuk penguapan air (*endotermis*). Selain itu tingginya kadar air dapat mengakibatkan tekanan yang berlebihan dalam ruang bakar.

5. Sifat keselamatan. Untuk menjamin keselamatan pemakaian dan penyimpanan residu perlu diperhatikan titik nyala (*flash point*) residu.

Sifat-sifat kimia residu:

1. Pencemaran udara. Gas buang hasil pembakaran residu yang mengandung kadar belerang tinggi akan menghasilkan gas SO_2 . Gas SO_2 yang terbentuk akan *teroksidasi* menjadi SO_3 dan bila bereaksi dengan uap air dalam jumlah besar akan menyebabkan hujan asam.
2. Kerusakan pada ruang bakar. Dengan kondisi sama seperti diatas jika H_2SO_4 encer yang korosif tersebut *terkondensasi* dalam ruang bakar saat *combustion* dimatikan dapat mengakibatkan karat.

II.4. Perangkat Lunak (Software)

II.4.1. Pemrograman C

Pada suatu pengontrolan alat, program yang digunakan adalah pemrograman bahasa C. Untuk itu diperlukan juga pemahaman tentang pemrograman tersebut. Berikut adalah penjelasan dasar-dasar dari pemrograman bahasa C :

1. Tipe Data

Berikut ini adalah tipe-tipe data yang ada dalam bahasa C dan yang dikenali oleh Arduino 0023 :

Tabel II.1. 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 | 2 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 | 4 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. 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 si 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 `'*/'`.

Contoh :

```
// Ini adalah komentar satu baris
```

```
/* Sedangkan yang ini adalah komentar yang lebih dari satu baris*/
```

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 prosessor yang digunakan, dalam hal ini adalah untuk mendefinisikan jenis mikrokontroller 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.

Tabel II.2.Operator Aritmatika

| Operator | Keterangan |
|----------|---------------------------------------|
| + | Operator untuk operasi penjumlahan |
| - | Operator untuk operasi pengurangan |
| * | Operator untuk operasi perkalian |
| / | Operator untuk operasi pembagian |
| % | Operator untuk operasi sisa pembagian |

7. Operator Logika

Operator logika digunakan untuk membentuk suatu logika atas dua buah kondisi atau lebih.

Berikut ini adalah tabel operator logika :

Tabel II.3.Operator Logika

| Operator | Keterangan |
|----------|---------------------------|
| && | Operator untuk logika AND |
| | Operator untuk logika OR |
| ! | Operator untuk logika NOT |

Contoh :

```
if ( (a==b) && (c!=d) ) = 0xFF;
```

Pernyataan diatas terdiri dari 2 buah kondisi yaitu $a==b$ dan $c!=d$ yang keduanya dihubungkan dengan logika && (AND). Jika logika yang dihasilkan benar maka perintah $PORTC = 0xFF$ akan dikerjakan dan jika salah tidak akan dikerjakan.

8. Operator Penambahan dan Pengurangan

Operator penambahan dan pengurangan adalah operator yang digunakan untuk menaikkan atau menurunkan nilai sebuah variabel dengan selisih 1.

Tabel II.4.Operator Penambahan dan Pengurangan

| Operator | Keterangan |
|----------|---|
| ++ | Operator untuk penambahan nilai variabel dengan 1 |
| -- | Operator pengurangan nilai variabel dengan 1 |

Contoh :

```
a = 1; b = 5;
```

```
a++; b--;
```

Maka operator $a++$ akan mengubah variable a dari 1 menjadi 2 dan operator $b--$ akan mengubah variable b dari 5 menjadi 4.

9. 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 data di isi dengan nilai PINA dan data 0xFF dikeluarkan ke PORT C. Apabila dalam blok pernyataan hanya terdapat satu pernyataan saja maka tanda { dan } dapat dihilangkan seperti :

```
if (PINA>0x80) PORTC=0xFF;
```

(Sumber:<http://repository.usu.ac.id/bitstream/123456789/26400/4/Chapter%20II.pdf>)

II.4.2. Arduino 0023

Arduino 0023 merupakan sebuah *cross-compiler C, Integrated Development Environment (IDE), dan Automatic Program Generator* yang didesain untuk mikrokontroler maupun minimum sistem arduino. Arduino 0023 dapat dijalankan pada system operasi Windows

95, 98, 2000, XP, 7 dan Linux. Arduino 0023 mampu menerjemahkan hampir semua perintah dari bahasa C, sejauh yang diijinkan oleh arsitektur dari AVR dan arduino, dengan tambahan beberapa fitur untuk mengambil kelebihan khusus dari arsitektur AVR dan arduino dan kebutuhan pada sistem embedded. *File object* COFF hasil kompilasi dapat digunakan untuk keperluan debugging pada tingkatan C, dengan pengamatan variabel, menggunakan *serial monitor*. Dari beberapa software kompiler C yang pernah digunakan, *Arduino software* merupakan yang terbaik jika dibandingkan dengan kompiler-kompiler yang lain karena memiliki beberapa kelebihan yang dimiliki oleh Arduino 0023 antara lain :

1. Menggunakan IDE yang multiplatform (Integrated Development Environment)
2. Fasilitas yang disediakan lengkap (mengedit program, mengkompilasi program, mendownload program) serta tampilannya terlihat menarik dan mudah dimengerti
3. Mampu membangkitkan kode program secara otomatis dengan menggunakan fasilitas dari Arduino 0023.
4. Memiliki fasilitas untuk mendownload program langsung tanpa bantuan USBasp.
6. Memiliki komunikasi serial monitor yang terintegrasi dalam arduino 0023 sehingga dapat digunakan untuk membantu pengecekan *output* program yang telah dibuat khususnya yang menggunakan fasilitas komunikasi serial USART.

