

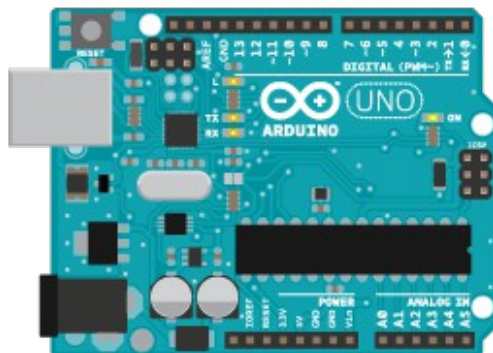
BAB II TINJAUAN PUSTAKA

2.1 Arduino

Arduino adalah merupakan sebuah *board minimum system mikrokontroler* yang bersifat *open source*. Didalam rangkaian board arduino terdapat mikrokontroler AVR seri ATmega 328 yang merupakan produk dari Atmel.

Arduino uno menggunakan board mikrokontroler yang didasarkan pada ATmega328, mempunyai 14 pin digital input dan output (6 diantaranya sebagai *output PWM*), 6 *input* analog yang merupakan osilator kristal 16Mhz, koneksi USB, *power jack*, ICSP *header*, dan tombol *reset*.

Arduinio uno dapat disuplai melalui koneksi USB atau dengan sebuah power suplai eksternal. Sumber daya dipilih secara otomatis. Suplai eksternal (non-USB) dapat diperoleh dari sebuah adaptor AC ke DC atau battery. Adaptor dapat dihubungkan dengan mencolokkan sebuah center-positive plug yang panjangnya 2,1 mm ke power jack dari board. Kabel lead dari sebuah battery dapat dimasukkan dalam header/kepala pin Ground (Gnd) dan pin Vin dari konektor POWER. Memory arduino, ATmega328 mempunyai 32 KB (dengan 0,5 KB digunakan untuk bootloader). ATmega 328 juga mempunyai 2 KB SRAM dan 1 KB EEPROM (yang dapat dibaca dan ditulis (RW/*read and written*) dengan EEPROM *library*. (djuandi,Feri. 2011)



Gambar 2.1 Board Arduino uno

(Sumber: <http://www.arduino.cc>)

Diakses pada Kamis, 14 Juli 2016 Pukul 11.47

Adapun deskripsi dari Arduino uno dapat ditunjukkan pada tabel berikut :

Tabel 2.1 Deskripsi Arduino Uno

Mikrokontroler	Atmega328
Tegangan pengoperasian	5V
Tegangan input yang disarankan	7-12V
Batas tegangan input	6-20V
Jumlah pin I/O digital	14(6 diantaranya output PWM)
Jumlah pin input analog	6
Arus DC tiap pin I/O	40 mA
Arus DC tiap pin I/O	50 mA
Memori Flash	32 KB(Atmega328), 0.5kb bootloader
SRAM	2 KB (Atmega328)
EEPROM	1 KB (Atmega328)
Clock Speed	16 Mhz

2.1.1 Pin Masukan dan Keluaran Arduino Uno

Masing-masing dari 14 pin digital arduino uno dapat digunakan sebagai masukan atau keluaran menggunakan fungsi `pinMode()`, `digitalWrite()` dan `digitalRead()`. Setiap pin beroperasi pada tegangan 5 volt. Setiap pin mampu menerima atau menghasilkan arus maksimum sebesar 40 mA dan memiliki resistor pull-up internal (diputus secara default) sebesar 20-30 KOhm. Sebagai tambahan, beberapa pin masukan digital memiliki kegunaan khusus yaitu:

- a. Komunikasi serial: pin 0 (RX) dan pin 1 (TX), digunakan untuk menerima(RX) dan mengirim(TX) data secara serial.
- b. External Interrupt: pin 2 dan pin 3, pin ini dapat dikonfigurasi untuk memicu sebuah interrupt pada nilai rendah, sisi naik atau turun, atau pada saat terjadi perubahan nilai.
- c. Pulse-width modulation (PWM): pin 3,5,6,9,10 dan 11, menyediakan keluaran PWM 8-bit dengan menggunakan fungsi `analogWrite()`.
- d. Serial Peripheral Interface (SPI): pin 10 (SS), 11 (MOSI), 12 (MISO) dan 13 (SCK), pin ini mendukung komunikasi SPI dengan menggunakan SPI library.

- e. LED: pin 13, terdapat built-in LED yang terhubung ke pin digital 13. Ketika pin bernilai HIGH maka LED menyala, sebaliknya ketika pin bernilai LOW maka LED akan padam.

Arduino UNO mempunyai 6 input analog, diberi label A0 sampai A5, setiapnya memberikan 10 bit resolusi (contohnya 1024 nilai yang berbeda). Secara default, 6 input analog tersebut mengukur dari ground sampai tegangan 5 Volt, dengan itu mungkin untuk mengganti batas atas dari rangnya dengan menggunakan pin AREF dan fungsi `analogReference()`. Di sisi lain, beberapa pin mempunyai fungsi spesial:

- a. TWI: pin A4 atau SDA dan pin A5 atau SCL. Mendukung komunikasi TWI dengan menggunakan Wire library
- b. AREF. Referensi tegangan untuk input analog. Digunakan dengan `analogReference()`.
- c. Reset. Membawa saluran ini LOW untuk mereset mikrokontroler. Secara khusus, digunakan untuk menambahkan sebuah tombol reset untuk melindungi yang memblock sesuatu pada board. (Angin, B Perangin. 2013)

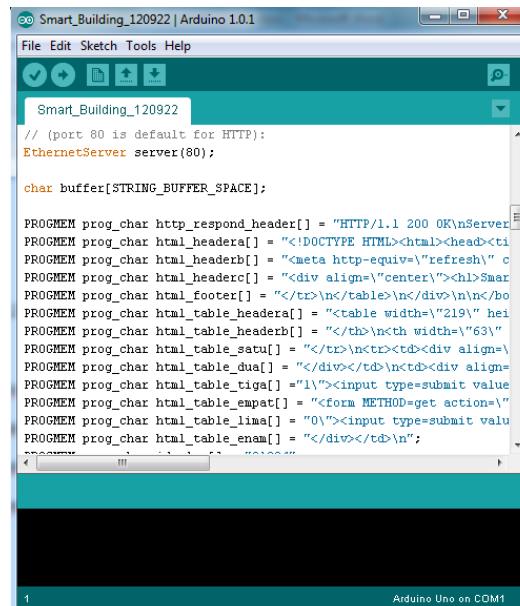
2.2 Bahasa Pemrograman Arduino

Arduino *board* merupakan perangkat yang berbasis mikrokontroler. Perangkat lunak (*software*) merupakan komponen yang membuat sebuah mikrokontroler dapat bekerja. Arduino *board* akan bekerja sesuai dengan perintah yang ada dalam perangkat lunak yang ditanamkan padanya.

Bahasa pemrograman Arduino adalah bahasa pemrograman utama yang digunakan untuk membuat program untuk arduino *board*. Bahasa pemrograman arduino menggunakan bahasa pemrograman C sebagai dasarnya. Karena menggunakan bahasa pemrograman C sebagai dasarnya, bahasa pemrograman arduino memiliki banyak sekali kemiripan, walaupun beberapa hal telah berubah.

Arduino *Development Environment* terdiri dari editor teks untuk menulis kode, sebuah area pesan, sebuah konsol, sebuah *toolbar* dengan tombol-tombol untuk fungsi yang umum dan beberapa menu. Arduino *Development Environment* terhubung ke arduino board untuk meng-*upload* program dan juga untuk berkomunikasi dengan arduino board.

Perangkat lunak yang ditulis menggunakan Arduino *Development Environment* disebut *sketch*. *Sketch* ditulis pada editor teks. *Sketch* disimpan dengan file berekstensi (.ino). Area pesan memberikan informasi dan pesan error ketika kita menyimpan atau membuka *sketch*. Konsol menampilkan output teks dari Arduino *Development Environment* dan juga menampilkan pesan error ketika kita mengkompilasi *sketch*. Pada sudut kanan bawah dari jendela Arduino *Development Environment* menunjukkan jenis board dan port serial yang sedang digunakan. Tombol *toolbar* digunakan untuk mengecek dan mengupload *sketch*, membuat, membuka atau menyimpan *sketch*, dan menampilkan *serial monitor*.



```

Smart_Building_120922
// (port 80 is default for HTTP)
EthernetServer server(80);

char buffer[STRING_BUFFER_SPACE];

PROGMEM prog_char http_respond_header[] = "HTTP/1.1 200 OK\r\nServer:
PROGMEM prog_char html_headera[] = "<!DOCTYPE HTML><html><head><ti
PROGMEM prog_char html_headerb[] = "<meta http-equiv=\"refresh\" c
PROGMEM prog_char html_headerc[] = "<div align=\"center\"><h1>Smar
PROGMEM prog_char html_footer[] = "</tr>\n</table>\n</div>\n\n</bo
PROGMEM prog_char html_table_headera[] = "<table width=\"219\" hei
PROGMEM prog_char html_table_headerb[] = "</th>\n<th width=\"63\"
PROGMEM prog_char html_table_satu[] = "</tr>\n<tr><td><div align=\
PROGMEM prog_char html_table_dua[] = "</div></td>\n<td><div align=\
PROGMEM prog_char html_table_tiga[] = "1\"><input type=submit value
PROGMEM prog_char html_table_empat[] = "<form METHOD=get action=\
PROGMEM prog_char html_table_lima[] = "0\"><input type=submit valu
PROGMEM prog_char html_table_enam[] = "</div></td>\n";

```

Gambar 2.2 Tampilan *Software* Arduino

(Sumber: <http://repository.usu.ac.id>)

Diakses pada Sabtu, 16 Juli 2016 Pukul 17.34

Berikut ini adalah tombol-tombol *toolbar* serta fungsinya:



Verify berfungsi untuk mengecek error pada kode program



Upload berfungsi untuk meng-*compile* dan meng-*upload* program ke Arduino board.



New berfungsi untuk membuat *sketch* baru



Open berfungsi untuk menampilkan sebuah menu dari seluruh *sketch* yang berada di dalam *sketchbook*.



Save berfungsi untuk menyimpan *sketch*



Serial Monitor berfungsi untuk membuka *serial monitor*.

Dalam lingkungan arduino digunakan sebuah konsep yang disebut *sketchbook*, yaitu tempat standar untuk menampung program (*sketch*). *Sketch* yang ada pada *sketchbook* dapat dibuka dari menu *File > Sketchbook* atau dari tombol *open* pada *toolbar*. Ketika pertama kali menjalankan arduino development environment, sebuah direktori akan dibuat secara otomatis untuk tempat penyimpana *sketchbook*. Kita dapat melihat atau mengganti lokasi dari direktori tersebut dari menu *File > Preferences*.

Serial monitor menampilkan data serial yang sedang dikirim dari arduino board. Untuk mengirim data ke board, masukkan teks dan klik tombol *send* atau tekan *enter* pada *keyboard*. Sebelum meng-*upload* program, kita perlu mensetting jenis board dan port serial yang sedang kita gunakan melalui menu *Tools > Board* dan *Tools > Serial Port*. Pemilihan board berguna untuk mengeset parameter (contohnya: kecepatan mikrokontroler dan *baud rate*) yang digunakan ketika meng-*compile* dan meng-*upload* *sketch*. Setelah memilih board dan port serial yang tepat, tekan tombol *upload* pada *toolbar* atau pilih menu *File > Upload*.

Arduino board akan me-*reset* secara otomatis dan proses *upload* akan dimulai. Pada kebanyakan board, LED RX dan TX akan berkedip ketika program sedang di-*upload*. *Arduino development environment* akan menampilkan pesan ketika proses upload telah selesai, atau menampilkan pesan error.

Ketika sedang meng-*upload* program, arduino bootloader sedang digunakan, Arduino bootloader adalah sebuah program kecil yang telah ditanamkan pada mikrokontroler yang berada pada arduino board. Bootloader ini memungkinkan kita meng-*upload* program tanpa menggunakan perangkat keras tambahan. (Simanjuntak, Maratur Gabe. 2013)

2.2.1 Struktur

Setiap program dalam arduino board terdiri dari dua fungsi utama yaitu `setup()` dan `loop()`. Instruksi yang berada dalam fungsi `setup()` dieksekusi hanya sekali, yaitu ketika arduino board pertama kali dihidupkan. Biasanya instruksi yang berada pada fungsi `setup()` merupakan konfigurasi dan inisialisasi dari arduino board. Fungsi `loop()` merupakan tugas utama dari arduino board. Jadi setiap program yang menggunakan bahasa pemrograman arduino memiliki struktur sebagai berikut:

```
void setup()
{
  // perintah-perintah untuk konfigurasi dan inisialisasi arduino board
}

void loop()
{
  //perintah-perintah utama arduino board
}
```

Gambar 2.3 Struktur Dasar *Software* Arduino

(Sumber: <http://repository.usu.ac.id>)

Diakses pada Sabtu, 16 Juli 2016 Pukul 17.34

2.2.2 Konstanta

Konstanta adalah variable yang sudah ditetapkan sebelumnya dalam bahasa pemrograman arduino. Konstanta digunakan agar program lebih mudah untuk dibaca dan dimengerti. Konstanta dibagi menjadi 3 kelompok yaitu:

- a. Konstanta yang digunakan untuk menunjukkan tingkat logika (konstanta Boolean), yaitu *true* dan *false*
- b. Konstanta untuk menunjukkan keadaan pin, yaitu HIGH dan LOW
- c. Konstanta untuk menunjukkan fungsi pin, yaitu INPUT, INPUT_PULLUP dan OUTPUT

Konstanta yang digunakan untuk menunjukkan benar atau salah dalam bahasa pemrograman arduino adalah *true* dan *false*. *False* lebih mudah didefinisikan daripada *true*. *False* didefinisikan sebagai 0 (nol). *True* sering didefinisikan sebagai 1 (satu), yang mana hal ini benar, tetapi *true* memiliki definisi yang lebih luas. Setiap integer yang bukan nol adalah *true* dalam pengertian Boolean. Jadi -2, 3 dan -100 semuanya didefinisikan sebagai *true*, juga dalam pengertian Boolean. Tidak seperti konstanta yang lain *true* dan *false* diketik dengan menggunakan huruf kecil.

Ketika membaca atau menulis ke sebuah pin digital, terdapat hanya dua nilai yang dapat diberikan atau diterima, yaitu HIGH dan LOW. HIGH memiliki arti yang berbeda tergantung apakah sebuah pin dikonfigurasi menjadi masukan atau keluaran. Ketika pin dikonfigurasi sebagai masukan dengan fungsi `pinMode()`, lalu kemudian dibaca dengan fungsi `digitalRead()`, mikrokontroler akan melaporkan nilai HIGH jika tegangan yang ada pada pin tersebut berada pada tegangan 3 volt atau lebih.

Ketika sebuah pin dikonfigurasi sebagai masukan, dan kemudian dibuat bernilai HIGH dengan fungsi `digitalWrite()`, maka resistor *pull-up* internal dari chip ATmega akan aktif, yang akan membawa pin masukan ke nilai HIGH kecuali pin tersebut ditarik (pull-down) ke nilai LOW oleh sirkuit dari luar.

Ketika pin dikonfigurasi sebagai keluaran dengan fungsi `pinMode()`, dan diset ke nilai HIGH dengan fungsi `digitalWrite()`, maka pin berada pada tegangan 5 volt. Dalam keadaan ini, pin tersebut dapat memberikan arus, sebagai contoh,

untuk menhidupkan LED yang terhubung seri dengan resistor dan ground, atau pin lain yang dikonfigurasi sebagai keluaran dan diberi nilai LOW.

Sama seperti HIGH, LOW juga memiliki arti yang berbeda bergantung pada konfigurasi pin. Ketika pin dikonfigurasi sebagai masukan, maka mikrokontroler akan melaporkan nilai LOW jika tegangan yang terdapat pada pin berada pada tegangan 2 volt atau kurang. Ketika pin dikonfigurasi sebagai keluaran dan diberi nilai LOW maka pin berada pada tegangan 0 volt.

Setiap pin pada arduino dapat dikonfigurasi sebagai masukan, masukan dengan resistor *pull-up* atau keluaran. Untuk mengkonfigurasi fungsi pin pada arduino digunakan konstanta INPUT, INPUT_PULLUP dan OUTPUT. Pin arduino yang dikonfigurasi sebagai masukan dengan fungsi pinMode() dikatakan berada dalam kondisi berimpedansi tinggi. Pin yang dikonfigurasi sebagai masukan memiliki permintaan yang sangat kecil kepada sirkuit yang di-sampling-nya. Hal ini membuat pin tersebut berguna untuk membaca sensor, tetapi tidak untuk menhidupkan sebuah LED.

Cip ATmega pada arduino memiliki resistor *pull-up* internal (resistor yang terhubung ke sumber tegangan secara internal) yang dapat digunakan. Untuk menggunakan resistor *pull-up* internal ini kita menggunakan konstanta INPUT_PULLUP pada fungsi pinMode().

Pin yang dikonfigurasi menjadi sebuah keluaran dikatakan berada dalam kondisi berimpedansi rendah. Hal ini berarti pin tersebut dapat menyediakan sejumlah besar arus ke sirkuit yang lain. Pin pada ATmega mampu menyediakan arus hingga 40 mA.

2.2.3 Fungsi Masukan dan Keluaran Digital

Arduino memiliki 3 fungsi untuk masukan dan keluaran digital pada arduino board, yaitu pinMode(), digitalWrite() dan digitalRead().

Fungsi pinMode() mengkonfigurasi pin tertentu untuk berfungsi sebagai masukan atau keluaran. Sintaksis untuk fungsi pinMode() adalah sebagai berikut:

```
pinMode(pin, mode)
```

Parameter: pin = angka dari pin digital yang akan dikonfigurasi

mode = konfigurasi yang diinginkan (INPUT, INPUT_PULLUP dan OUTPUT).

Fungsi `digitalWrite()` berfungsi untuk memberikan nilai HIGH atau LOW suatu digital pin. Sintaksis untuk fungsi `digitalWrite()` adalah sebagai berikut:

`digitalWrite(pin, value)`

Parameter: pin = angka dari pin digital yang akan dikonfigurasi

value = nilai yang diinginkan (HIGH atau LOW).

Fungsi `digitalRead()` bertujuan untuk membaca nilai yang ada pada pin arduino uno. Sintaksis untuk fungsi `digitalRead()` adalah sebagai berikut:

`digitalRead(pin)`

Parameter: pin = angka dari pin digital yang akan dibaca

Berikut ini adalah contoh penggunaan fungsi masukan dan keluaran digital dalam sebuah program:

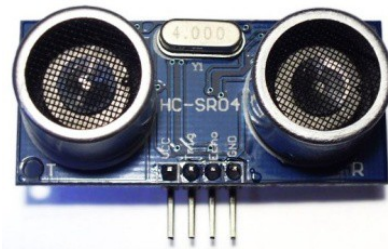
```
int ledPin = 13; // LED terhubung ke pin digital 13
int inPin = 7; // pushbutton terhubung ke pin digital 7
int val = 0; // variable untuk menyimpan sebuah nilai
void setup()
{
pinMode(ledPin, OUTPUT); // set pin digital 13 sebagai keluaran
pinMode(inPin, INPUT); // set pin digital 13 sebagai masukan
}

void loop()
{
val = digitalRead(inPin); // baca nilai pin input
digitalWrite(ledPin, val); // sets LED sesuai dengan nilai val
```

2.3 Sensor Yang Digunakan

2.3.1 Sensor Jarak Ultrasonik

Sensor ultrasonik adalah komponen yang kerjanya didasarkan prinsip dari pantulan suatu gelombang suara sehingga dapat dipakai untuk menafsirkan eksistensi sebuah benda spesifik yang ada dalam frekuensinya.



Gambar 2.4 Sensor Jarak Ultrasonik

(Sumber: <https://stab-iitb.org>)

Diakses pada Kamis, 14 Juli 2016 Pukul 11.34

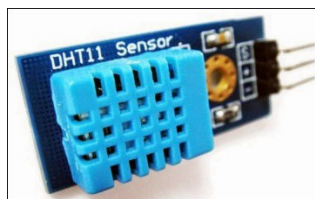
Modul sensor ultrasonik ini dapat mengukur jarak antara 3cm sampai 300cm. Keluaran dari modul sensor ultrasonik ini berupa pulsa yang lebarnya merepresentasikan jarak. Lebar pulsanya yang dihasilkan modul sensor ultrasonik ini bervariasi dari $115\mu\text{S}$ sampai 18.5mS . Secara prinsip modul sensor ultrasonik ini terdiri dari sebuah pembangkit sinyal 40KHz, sebuah speaker ultrasonik dan sebuah mikropon ultrasonik. Speaker ultrasonik mengubah sinyal 40KHz menjadi suara, sementara mikropon ultrasonik berfungsi untuk mendeteksi pantulan suaranya.

Sinyal output sensor ultrasonik dapat langsung dihubungkan dengan arduino tanpa tambahan komponen apapun. Modul sensor ultrasonik hanya akan mengirimkan suara ultrasonik ketika ada pulsa trigger dari arduino (Pulsa high selama $5\mu\text{S}$). Suara ultrasonik dengan frekuensi sebesar 40KHz akan dipancarkan selama $200\mu\text{S}$ oleh modul sensor ultrasonik ini. Suara ini akan merambat di udara dengan kecepatan 344.434m/s yang kemudian mengenai objek dan

dipantulkan kembali ke model sensor ultrasonik tersebut. Selama menunggu pantulan sinyal ultrasonik dari bagian transmitter, modul sensor ultrasonik dari bagian transmitter, modul ini akan menghasilkan sebuah pulsa. Pulsa ini akan berhenti ketika suara pantulan terdeteksi oleh modul sensor ultrasonik. Oleh karena itulah lebar pulsa tersebut dapat merepresentasikan jarak antara modul sensor ultrasonik dengan objek. (Arief, U Meidiaty. 2012)

2.3.2 Sensor Suhu DHT11

DHT11 adalah sensor Suhu dan Kelembaban udara, DHT11 memiliki keluaran sinyal digital yang dikalibrasi dengan sensor suhu dan kelembaban yang kompleks. Teknologi ini memastikan keandalan tinggi dan sangat baik stabilitasnya dalam jangka panjang. Sensor ini termasuk elemen resistif dan perangkat pengukur suhu. Memiliki kualitas yang baik, respon cepat, dan kemampuan anti-gangguan. Dengan menggunakan teknik digital-signal eksklusif dan suhu dan teknologi penginderaan kelembaban, memastikan keandalan yang tinggi dan stabilitas jangka panjang yang sangat baik. Setiap sensor DHT11 memiliki fitur kalibrasi sangat akurat dari kelembaban. Koefisien kalibrasi yang disimpan dalam memori program OTP, sensor internal mendeteksi sinyal dalam proses, yang biasa disebut koefisien kalibrasi. Sistem antarmuka tunggal-kabel serial terintegrasi untuk menjadi cepat dan mudah. Kecil ukuran, daya rendah, sinyal transmisi jarak hingga 20 meter. (Amri, M. Faisal Afrido. 2015)



Gambar 2.5 Sensor DHT11

(Sumber: <http://ilmu-otomasi.blogspot.co.id>)

Diakses pada Kamis, 14 Juli 2016 Pukul 11.47

Spesifikasi sensor suhu DHT11 sebagai berikut:

- a. Pasokan Voltage: 5 V
- b. Rentang temperatur: 0-50 ° C kesalahan ± 2 ° C
- c. Kelembaban: 20-90% RH $\pm 5\%$ RH *error*
- d. *Interface*: Digital

2.4 Modem GSM

Modem adalah sebuah alat yang dapat membuat komputer terkoneksi dengan internet melalui line telepon standar. Modem banyak digunakan komputer rumah dan jaringan sederhana untuk dapat berkomunikasi dengan jutaan komputer lain dalam lalu lintas internet. Kata Modem itu sendiri merupakan kependekan dari Modulator Demodulator. Ini berarti Modem bekerja dengan cara mengubah informasi digital dari komputer pengirim ke dalam bentuk sinyal analog yang ditransmisikan melalui line telepon.

Selanjutnya Modem pada komputer penerima akan mengubah ulang sinyal analog ke sinyal digital. Modem GSM adalah sebuah perangkat Modem Wireless Plug and Play dengan konektivitas GSM/GPRS untuk aplikasi-aplikasi machine to machine. GSM Modul atau Modem GSM adalah jenis khusus dari modem yang menerima kartu SIM, dan mengoperasikan selama berlangganan ke operator mobile, seperti ponsel. Modem GSM dihubungkan dengan suatu interface yang memungkinkan aplikasi seperti SMS untuk mengirim dan menerima pesan melalui Modem. Beberapa fungsi kegunaan modem ini di masyarakat adalah antara lain:

- a. SMS Broadcast application
- b. SMS Quiz application
- c. SMS Polling
- d. SMS auto-reply
- e. M2M integration
- f. Aplikasi Server Pulsa
- g. Telemetri
- h. Payment Point Data (Hermawan, Reyfaldi. 2015)



Gambar 2.6 Modem GSM Wavecom

(Sumber: <http://www.musbikhin.com>)

Diakses pada Minggu, 24 Juli 2016 Pukul 1.15

2.4.1 AT-Command

AT-Command adalah singkatan dari *Attention Command*. AT Command adalah perintah yang digunakan dalam komunikasi dengan serial port. Pada awalnya standar perintah ini untuk modem-modem telepon PSTN, akan tetapi perintah ini sekarang dikembangkan juga untuk modem-modem GSM.

Perintah AT-Command dapat diberikan kepada handphone atau GSM/CDMA modem untuk melakukan sesuatu hal, termasuk untuk mengirim dan menerima SMS. Dengan memberikan perintah ini di dalam komputer/mikrokontroler maka perangkat kita dapat melakukan pengiriman atau penerimaan SMS secara otomatis untuk mencapai tujuan tertentu. Untuk memulai suatu perintah AT-Command, diperlukan prefiks “AT” atau “at” dalam setiap perintah AT-Command.

Tabel 2.2 Tabel Set AT-Command

AT-Command	Keterangan
AT	Mengecek koneksi ke PC
AT+CMGF	Menetapkan format mode
AT+CSCS	Menetapkan jenis encoding
AT+CPMS	Mendeteksi SMS baru otomatis
AT+CMGL	Membuka daftar SMS di SIM
AT+CMGS	Mengirim pesan SMS
AT+CMGR	Membaca pesan SMS
AT+CMGD	Menghapus pesan SMS
AT+CNMI	Mendeteksi Kode HP/Modem

2.5 *Short Message Service (SMS)*

Short Message Service (SMS) adalah layanan untuk mengirim dan menerima pesan tertulis (teks) dari maupun kepada perangkat bergerak (*Mobile Device*). Pesan teks yang dimaksud tersusun dari huruf, angka, atau karakter alfanumerik. Pesan teks dikemas dalam satu paket/*frame* yang berkapasitas maksimal 160 byte yang dapat direpresentasikan berupa 160 karakter huruf latin atau 70 karakter alfabet non-latin seperti alfabet Arab atau Cina.

SMS pada awal diciptakan adalah bagian dari layanan pada sistem GSM yang dikembangkan dan distandarisasi. SMS semula hanyalah merupakan layanan yang bersifat komplementer terhadap dua layanan utama sistem GSM (atau sistem 2G pada umumnya) yaitu layanan *suara* dan *switched data*. Namun karena

keberhasila SMS yang tidak terduga, dengan ledakan pelanggan yang mempergunakannya, menjadikan SMS sebagai bagian integral dari layanan sistem.

SMS adalah data tipe pesan *asinkron* yang pengiriman datanya dilakukan dengan mekanisme protokol *store and forward*. Hal ini berarti bahwa pengirim dan penerima SMS tidak perlu berada dalam status berhubungan (*connected/online*) satu sama lain ketika akan saling bertukar pesan.

Keterbatasan SMS adalah pada ukuran pesan yang dapat dikirimkan, yaitu maksimal hanya sebesar 160 byte. Keterbatasan ini disebabkan karena mekanisme transmisi SMS itu sendiri. Pada awalnya, SMS merupakan sebuah layanan yang ditambahkan pada sistem GSM yang digunakan untuk mengirimkan data mengenai konfigurasi dari handset pelanggan GSM. SMS dikirimkan menggunakan *signalling frame* pada kanal frekuensi atau *time slot frame* GSM yang biasanya digunakan untuk mengirimkan pesan untuk kontrol dan sinyal setup panggilan telepon, seperti pesan singkat tentang kesibukan jaringan atau pesan CLI (*Caller Line identification*). *Frame* ini bersifat khusus dan ada pada setiap panggilan telepon serta tidak dapat digunakan untuk membawa suara atau data dari pelanggan melainkan hanya berupa pesan saja. Ukuran *frame* pada sistem GSM sendiri adalah sebesar 1250 bit (kurang lebih sama dengan 160 byte). Karena hanya menggunakan satu *frame* inilah pengiriman pesan SMS menjadi sangat murah dan terjangkau, karena beban biaya hanya dihitung dari penggunaan satu frame melalui kanal frekuensi.

Pengiriman SMS menggunakan frame pada kanal frekuensi adalah berarti SMS dikirim oleh pengirim ke nomor telepon tertentu yang bertindak sebagai SMSC (*SMS-Center*) dan kemudian SMSC bertugas untuk meneruskannya ke penerima. Pengiriman SMS berlangsung cepat karena, SMSC selain terhubung ke LAN aplikasi juga terhubung ke MSC (*Mobile Switching Network*) melalui SS7 (*Signaling System 7*) yang merupakan jaringan khusus untuk menangkap frame kontrol dan sinyal. Mekanisme pengiriman pesan singkat SMS yang serupa juga ditemukan dalam sistem jaringan lain seperti TDMA, PDC, dan cdmaOne. Beda antara sistem jaringan satu dengan yang lainnya adalah ukuran dari pesan SMS itu

sendiri yang bergantung pada ukuran paket yang digunakan pada masing-masing sistem. Pada sistem TDMA dan PDC ukuran pesan SMS sama dengan sistem GSM, yaitu 160 byte, dan pada cdma-One ukuran pesan SMS sebesar 256 byte. (Chandra, Budiman. 2010)

2.6 *Power Supply Unit*

Power Supply adalah sebagai alat atau perangkat keras yang mampu menyuplai tenaga atau tegangan listrik secara langsung dari sumber tegangan listrik ke tegangan listrik yang lainnya. *Power supply* memiliki input dari tegangan yang berarus *alternating current* (AC) dan mengubahnya menjadi arus *direct current* (DC) lalu menyalurkannya ke berbagai perangkat keras elektronika. Karena memang arus (DC) yang dibutuhkan untuk perangkat keras agar dapat beroperasi dengan baik, *direct current* biasa disebut juga sebagai arus yang searah sedangkan *alternating current* merupakan arus yang berlawanan.



Gambar 2.7 *Power Supply Unit*

(Sumber: <https://www.circuitspecialists.com>)

Diakses pada Sabtu, 16 Juli 2016 Pukul 22.32

Baterai atau accu adalah sumber catu daya DC yang paling baik. Namun untuk aplikasi yang membutuhkan catu daya lebih besar, sumber dari baterai tidak cukup. Sumber catu daya yang besar adalah sumber AC (*alternating current*) dari pembangkit tenaga listrik. Untuk itu diperlukan suatu perangkat catu daya yang dapat mengubah arus AC menjadi DC. Rangkaian penyearah sudah cukup bagus jika tegangan ripple-nya kecil, namun ada masalah stabilitas. Jika tegangan PLN

naik/turun, maka tegangan outputnya juga akan naik/turun. Seperti rangkaian penyearah, jika arus semakin besar ternyata tegangan dc keluarnya juga ikut turun. Untuk beberapa aplikasi perubahan tegangan ini cukup mengganggu, sehingga diperlukan komponen aktif yang dapat meregulasi tegangan keluaran ini menjadi stabil. (Nur, Miskah. 2013)

2.7 Pompa Akuarium

Pompa adalah mesin atau peralatan mekanis yang digunakan untuk menaikkan cairan dari dataran rendah ke dataran tinggi atau untuk mengalirkan cairan dari daerah bertekanan rendah ke daerah yang bertekanan tinggi dan juga sebagai penguat laju aliran pada suatu sistem jaringan perpipaan. Hal ini dicapai dengan membuat suatu tekanan yang rendah pada sisi masuk atau *suction* dan tekanan yang tinggi pada sisi keluar atau *discharge* dari pompa.

Pada prinsipnya, pompa mengubah energi mekanik motor menjadi energi aliran fluida. Energi yang diterima oleh fluida akan digunakan untuk menaikkan tekanan dan mengatasi tahanan-tahanan yang terdapat pada saluran yang dilalui. Pompa juga dapat digunakan pada proses-proses yang membutuhkan tekanan hidrolis yang besar. Hal ini bisa dijumpai antara lain pada peralatan-peralatan berat. Dalam operasi, mesin-mesin peralatan berat membutuhkan tekanan *discharge* yang besar dan tekanan isap yang rendah. Akibat tekanan yang rendah pada sisi isap pompa maka fluida akan naik dari kedalaman tertentu, sedangkan akibat tekanan yang tinggi pada sisi *discharge* akan memaksa fluida untuk naik sampai pada ketinggian yang diinginkan.



Gambar 2.8 Contoh Pompa Akuarium

(Sumber: <http://sanfordlegenda.blogspot.co.id>)

Diakses pada Sabtu, 16 Juli 2016 Pukul 22.32

2.8 Elemen Pemanas (*Heater*)

Elemen pemanas atau *heater* berfungsi untuk mengubah arus listrik menjadi panas. *Heater* merupakan elemen pemanas listrik yang banyak dipakai dalam kehidupan sehari-hari, di dalam rumah tangga ataupun peralatan mesin industri. *Heater* bekerja sangat sederhana, tidak seperti konduktor. *Heater* terbuat dari logam nilai resistansinya yang tinggi, bahan logam yang biasa digunakan adalah: *mild stell*, *stainless stell*, tembaga, kuningan serta paduan *nikel-chrome* yang disebut *nichrome*. Jika arus mengalir melalui elemen dengan resistansi yang tinggi, aliran yang bekerja pada elemen ini akan menghasilkan panas. Jika arus mati, elemen secara perlahan menjadi dingin. Bentuk dan tipe dari *heater* ini bermacam-macam disesuaikan dengan fungsi, tempat pemasangan dan media yang akan dipanaskan.

Panas yang dihasilkan oleh elemen pemanas listrik ini bersumber dari kawat ataupun pita bertahanan listrik tinggi (*Resistance Wire*) yang dialiri arus listrik pada kedua ujungnya dan dilapisi oleh isolator listrik yang mampu meneruskan panas dengan baik hingga aman jika digunakan. (Suryana, Cahya. 2012)



Gambar 2.9 Elemen Pemanas (*Heater*)

(Sumber: Dokumen pribadi)

2.9 Relay

Relay berfungsi untuk menghubungkan atau memutus aliran arus listrik yang dikontrol dengan memberikan tegangan dan arus tertentu pada koilnya. Relay biasanya hanya mempunyai satu kumparan tetapi relay dapat mempunyai beberapa kontak. Dalam memutus atau menghubungkan kontak digerakkan oleh fluksi yang ditimbulkan dari adanya medan magnet listrik yang dihasilkan oleh kumparan yang melilit pada besi lunak.



Gambar 2.10 Relay

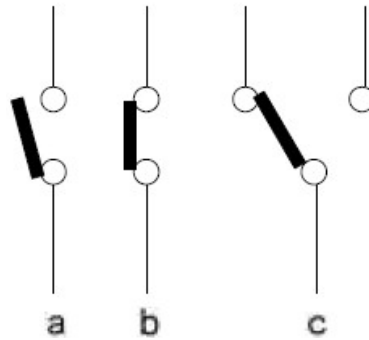
(Sumber: <http://teknikelektronika.com>)

Diakses pada Kamis, 14 Juli 2016 Pukul 11.47

Pada dasarnya, konstruksi dari relay terdiri dari lilitan kawat (koil) yang dililitkan pada inti besi lunak. Jika lilitan kawat mendapat aliran arus, inti besi lunak kontak menghasilkan medan magnet dan menarik switch kontak. Switch kontak mengalami gaya listrik magnet sehingga berpindah posisi ke kutub lain atau terlepas dari kutub asalnya. Keadaan ini akan bertahan selama arus mengalir pada kumparan relay. Dan relay akan kembali ke posisi semula (normal), bila tidak ada lagi arus yang mengalir padanya. Posisi normal relay tergantung pada jenis relay yang digunakan. Beberapa susunan kontak relay yang umum dipakai, semuanya terisolasi secara listrik dari rangkaian kumparan :

- a. Normal terbuka (Normally Open, NO) : kontak-kontak tertutup bila relay diberi arus listrik.

- b. Normal tertutup (Normally Close, NC) : kontak-kontak terbuka bila dialiri arus listrik.
- c. Tukar sambung (Change Over, CO) : relay ini mempunyai kontak tengah yang normalnya tertutup tetapi akan terbuka bila diberi arus listrik.



Gambar 2.11 Simbol Relay

a) Normally Open , b) Normally Close , c) Change Over

(Sumber: <http://teknikelektronika.com>)

Diakses pada Kamis, 14 Juli 2016 Pukul 11.47

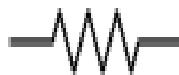
Relay-relay ini terdapat dalam dua jenis penggunaan arus , AC dan DC. Relay AC menggunakan daya jaringan listrik umum yaitu 220 Volt dan 110 Volt. Sedangkan relay DC menggunakan tegangan antar 5 Volt sampai 24 Volt dan membutuhkan arus kerja sekitar 100 mA. (Napitupulu, Chandra M. 2011)

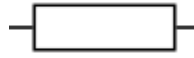
2.10 Resistor

Resistor adalah suatu komponen elektronika yang fungsinya untuk menghambat arus listrik. Resistor dapat dibagi menjadi dua, yaitu :

a.a. Resistor Tetap

Resistor tetap adalah resistor yang memiliki nilai hambatan yang tetap. Resistor memiliki batas kemampuan daya misalnya : 1/16 watt, 1/8 watt, ¼ watt, ½ watt dan sebagainya. Artinya resistor hanya dapat dioperasikan dengan daya maksimal sesuai dengan kemampuan dayanya.





Gambar 2.12 Simbol Resistor

(Sumber: <http://zoniaelektro.net>)

Diakses pada Kamis, 14 Juli 2016 Pukul 12.05

Resistor ini biasanya berbentuk silinder dengan pita-pita warna yang melingkar di badan resistor. Pita-pita warna dikenal sebagai kode resistor. Dengan mengetahui kode *resistor* kita dapat mengetahui nilai resistansi resistor, toleransi, koefisiensi temperatur dan reliabilitas resistor tersebut.

Resistor dengan kode warna terdiri dari 3 macam yaitu :

1. Resistor 4 pita warna dengan 1 pita warna untuk toleransi.
2. Resistor 5 pita warna dengan 1 pita warna untuk toleransi.
3. Resistor 5 pita warna dengan 1 pita warna untuk toleransi dan 1 pita warna untuk reliabilitas.

Dibawah ini merupakan table kode warna dari resistor:

Tabel 2.3 Kode Warna Resistor

Warna	Pita pertama	Pita kedua	Pita ketiga (pengali)	Pita keempat (toleransi)	Pita kelima (koefisien suhu)
Hitam	0	0	$\times 10^0$		
Cokelat	1	1	$\times 10^1$	$\pm 1\%$ (F)	100 ppm
Merah	2	2	$\times 10^2$	$\pm 2\%$ (G)	50 ppm
Oranye	3	3	$\times 10^3$		15 ppm
Kuning	4	4	$\times 10^4$		25 ppm
Hijau	5	5	$\times 10^5$	$\pm 0.5\%$ (D)	
Biru	6	6	$\times 10^6$	$\pm 0.25\%$ (C)	
Ungu	7	7	$\times 10^7$	$\pm 0.1\%$ (B)	
Abu-abu	8	8	$\times 10^8$	$\pm 0.05\%$ (A)	
Putih	9	9	$\times 10^9$		
Emas			$\times 10^{-1}$	$\pm 5\%$ (J)	
Perak			$\times 10^{-2}$	$\pm 10\%$ (K)	
Kosong				$\pm 20\%$ (M)	

b. Resistor Tidak Tetap

Resistor tidak tetap ialah resistor yang nilai hambatannya atau resistansinya dapat diubah-ubah. Jenisnya antara lain : hambatan geser, trimpot dan potensiometer. Yang banyak digunakan ialah trimpot dan potensiometer.

1. Potensiometer

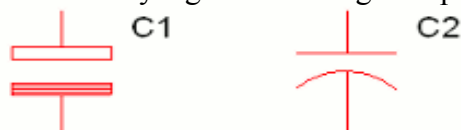
Potensiometer adalah resistor yang nilai resistansinya dapat diubah-ubah dengan memutar poros yang telah tersedia. Potensiometer pada dasarnya sama dengan trimpot secara fungsional.

2. Trimpot

Trimpot adalah resistor yang nilai resistansinya dapat diubah-ubah dengan cara memutar porosnya dengan menggunakan obeng. Untuk mengetahui nilai hambatan dari suatu trimpot dapat dilihat dari angka yang tercantum pada badan trimpot tersebut.

2.11 Kapasitor

Kapasitor adalah komponen elektronika yang mampu menyimpan muatan listrik. Kapasitor terbuat dari dua buah keping logam yang dipisahkan oleh bahan dielektrik, seperti keramik, gelas, vakum, dan lain-lain. Muatan positif dan negatif akan berkumpul pada kedua ujung berlainan tersebut apabila kedua ujung metal (elektroda) dihubungkan dengan sumber tegangan. Kapasitor berfungsi untuk menyimpan muatan listrik/elektron yang disebut dengan kapasitansi.



Gambar 2.13 Simbol Kapasitor

(Sumber: <http://zoniaelektro.net>)

Diakses pada Kamis, 14 Juli 2016 Pukul 12.32

Adapun jenis jenis kapasitor berdasarkan isolatornya adalah sebagai berikut :

- a. Kapasitor Elektrolit / ELCO (kapasitor yang memiliki polaritas, kaki + dan kaki -)
- b. Kapasitor Keramik
- c. Kapasitor Mylar
- d. Kapasitor Mika
- e. Kapasitor Kertas