

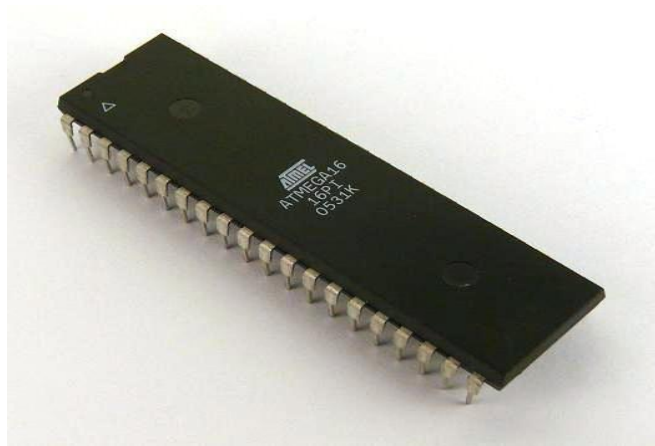
BAB II

LANDASAN TEORI

2.1 Mikrokontroler ATmega 16

Mikrokontroler adalah sebuah sistem komputer lengkap dalam satu serpih (*chip*). Mikrokontroler lebih dari sekedar sebuah mikroprosesor karena sudah terdapat atau berisikan ROM (*Read-Only Memory*), RAM (*Read-Write Memory*), beberapa port masukan maupun keluaran, dan beberapa *peripheral* seperti pencacah/pewaktu, ADC (*Analog to Digital converter*), DAC (*Digital to Analog converter*), PWM (*pulse width modulation*) dan serial komunikasi.

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 dan ATtiny. Pada dasarnya yang membedakan masing-masing kelas adalah memori, *peripheral*, dan fiturnya. Seperti mikroprosesor pada umumnya, secara internal mikrokontroler ATmega16 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*). Bentuk fisik IC mikrokontroler ATmega16 di tunjukkan pada gambar 2.1.



Gambar 2.1 Bentuk fisik ATmega16¹

2.1.1 Arsitektur ATmega16

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 ATmega16 terdiri dari:

Mikrokontroler AVR 8-bit yang memiliki kemampuan tinggi, dengan daya rendah

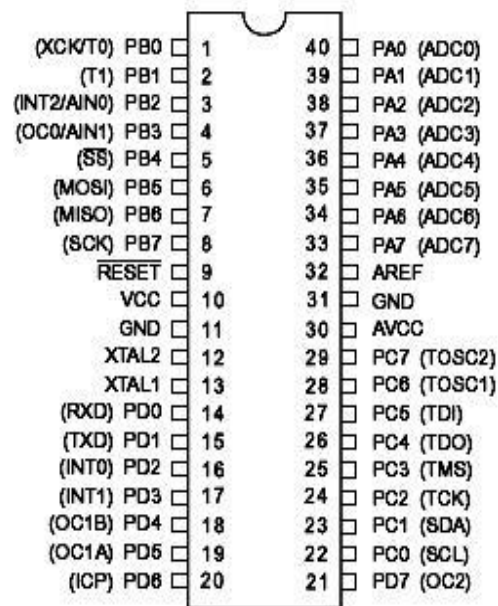
1. Arsitektur RISC dengan throughput mencapai 16 MIPS pada frekuensi 16Mhz.
2. Memiliki kapasitas Flash memori 16Kbyte, EEPROM 512 Byte, dan SRAM 1Kbyte
3. Saluran I/O 32 buah, yaitu Port A, Port B, Port C, dan Port D.
4. CPU yang terdiri dari 32 buah register.
5. User interupsi internal dan eksternal

¹ <http://www.hallroad.org/microcontrollers/38-atmega16.html>, diakses tanggal: 7 Januari 2017

6. Port antarmuka SPI dan Port USART sebagai komunikasi serial
7. 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 osilator tersendiri
 - Empat kanal PWM dan Antarmuka komparator analog
 - 8 kanal, 10bit ADC
 - *Byte-oriented Two-wire Serial Interface*
 - *Watchdog timer* dengan osilator internal

2.1.2 Konfigurasi Pin ATmega16

Konfigurasi *pin* mikrokontroler ATmega16 dengan kemasan 40 pin dapat dilihat pada Gambar 2.1 Dari gambar tersebut dapat terlihat ATmega16 memiliki 8 pin untuk masing-masing *Port A*, *Port B*, *Port C*, dan *Port D*. gambar pin ATmega16 ditunjukkan pada gambar 2.2.



Gambar 2.2 Port-port ATmega16²

2.1.3 Deskripsi Mikrokontroler ATmega16

- VCC (*Power Supply*) dan GND (*Ground*)

VCC merupakan pin yang berfungsi sebagai masukan catu daya

- GND (*Ground*)

GND merupakan pin *ground*

- Port A (PA7...PA0)

Port A berfungsi sebagai *input* analog pada konverter A/D. Port A juga sebagai suatu port I/O 8-bit dua arah, jika A/D konverter tidak digunakan. Pin-pin port dapat menyediakan resistor *internal pull-up* (yang dipilih untuk masing-masing bit). Port A *output* buffer mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Ketika pin PA0 ke PA7 digunakan sebagai input dan secara eksternal ditarik rendah, pin-pin akan memungkinkan arus

² <http://www.atmel.com/images/doc2466.pdf>, diakses tanggal: 7 Januari 2017

sumber jika resistor *internal pull-up* diaktifkan. Pin port A adalah tri-stated manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

- Port B (PB7...PB0)

Port B adalah suatu port I/O 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). Port B output buffer mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Sebagai input, pin Port B yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pin port B adalah tri-stated manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

- Port C (PC7...PC0)

Port C adalah suatu port I/O 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). Port C output buffer mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Sebagai input, pin port C yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pin port C adalah tri-stated manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

- Port D (PD7...PD0)

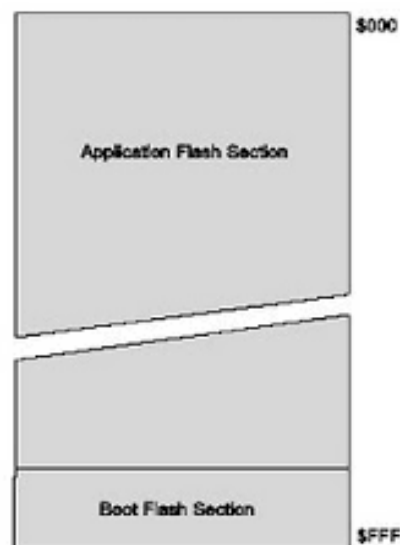
Port D adalah suatu port I/O 8-bit dua arah dengan resistor *internal pull-up* (yang dipilih untuk beberapa bit). Port D output buffer mempunyai karakteristik gerakan simetris dengan keduanya sink tinggi dan kemampuan sumber. Sebagai input, pin port D yang secara eksternal ditarik rendah akan arus sumber jika resistor *pull-up* diaktifkan. Pin port D adalah tri-stated manakala suatu kondisi reset menjadi aktif, sekalipun waktu habis.

- RESET (*Reset input*)
- XTAL1 (*Input Oscillator*)
- XTAL2 (*Output Oscillator*)
- AVCC adalah pena penyedia tegangan untuk bandar A dan Konverter A/D.
- AREF adalah pena referensi analog untuk konverter A/D.

2.1.4 Struktur Memori ATmega16

2.1.4.1 Memori Program

Arsitektur ATmega16 mempunyai dua memori utama, yaitu memori data dan memori program. Selain itu, ATmega16 memiliki memori EEPROM untuk menyimpan data. ATmega16 memiliki 16Kbyte *On-chip In-System Reprogrammable Flash Memory* untuk menyimpan program. Instruksi ATmega16 semuanya memiliki format 16 atau 32 bit, maka memori *flash* diatur dalam 8K x 16 bit. Memori *flash* dibagi kedalam dua bagian, yaitu bagian program *boot* dan aplikasi.



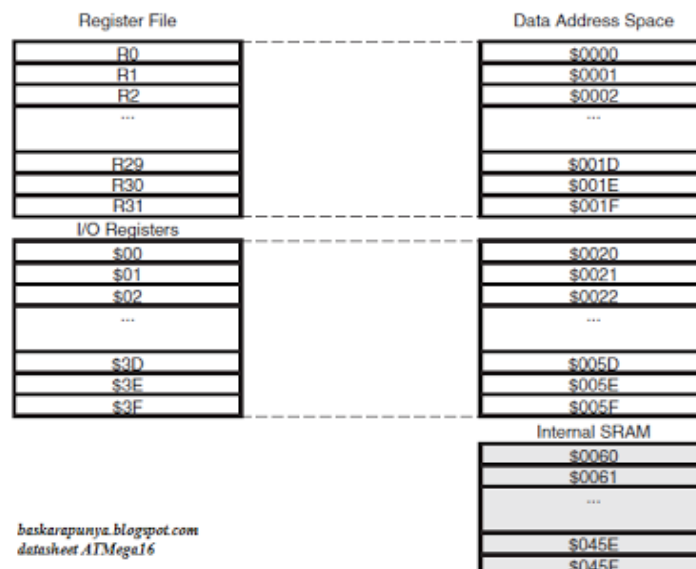
Gambar 2.3 Peta Memori Program AVR ATmega16³

³ <http://www.atmel.com/images/doc2466.pdf>, diakses tanggal: 7 Januari 2017

Bootloader adalah program kecil yang bekerja pada saat sistem dimulai yang dapat memasukkan seluruh program aplikasi ke dalam memori prosesor. Gambar peta memori program ATmega16 AVR ditunjukkan pada gambar 2.3.

2.1.4.2 Memori Data (SRAM)

Memori data AVR ATmega16 terbagi menjadi 3 bagian, yaitu 32 register umum, 64 buah register I/O dan 1 Kbyte SRAM internal. *General purpose Register* menempati alamat data terbawah, yaitu \$00 sampai \$1F. Sedangkan memori I/O menempati 64 alamat berikutnya mulai dari \$20 hingga \$5F. Memori I/O merupakan register yang khusus digunakan untuk mengatur fungsi terhadap berbagai fitur mikrokontroler seperti kontrol register, *timer/counter*, fungsi-fungsi I/O, dan sebagainya. 1024 alamat berikutnya mulai dari \$60 hingga \$45F digunakan untuk SRAM internal. Gambar peta memori data ATmega16 ditunjukkan pada gambar 2.4.



Gambar 2.4 Peta Memori Data AVR ATmega16⁴

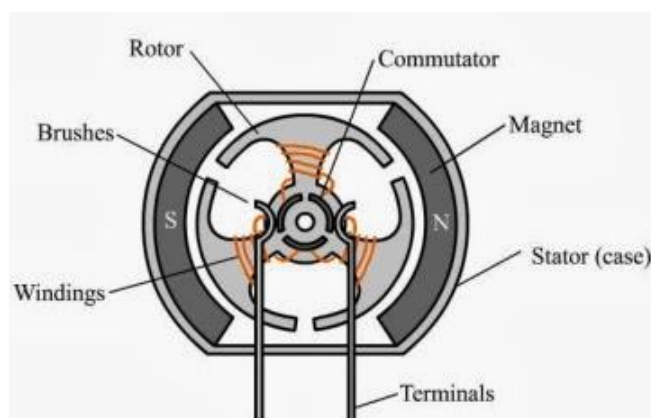
⁴ <http://www.atmel.com/images/doc2466.pdf>, diakses tanggal: 7 Januari 2017

2.1.4.3 Memori Data EEPROM

ATmega16 terdiri dari 512 byte memori data EEPROM 8 bit, data dapat tulis/baca dari memori ini, ketika catu daya dimatikan, data terakhir yang ditulis pada memori EEPROM masih tersimpan pada memori ini, atau dengan kata lain memori EEPROM bersifat *nonvolatile*. Alamat EEPROM mulai dari \$000 sampai \$1FF=.

2.2 Motor DC

Motor DC merupakan jenis motor yang menggunakan tegangan searah sebagai sumber tenaganya. Dengan memberikan beda tegangan pada kedua terminal tersebut, motor akan berputar pada satu arah, dan bila polaritas dari tegangan tersebut dibalik maka arah putaran motor akan terbalik pula seperti pada gambar 2.5. Polaritas dari tegangan yang diberikan pada dua terminal menentukan arah putaran motor sedangkan besar dari beda tegangan pada kedua terminal menentukan kecepatan motor.



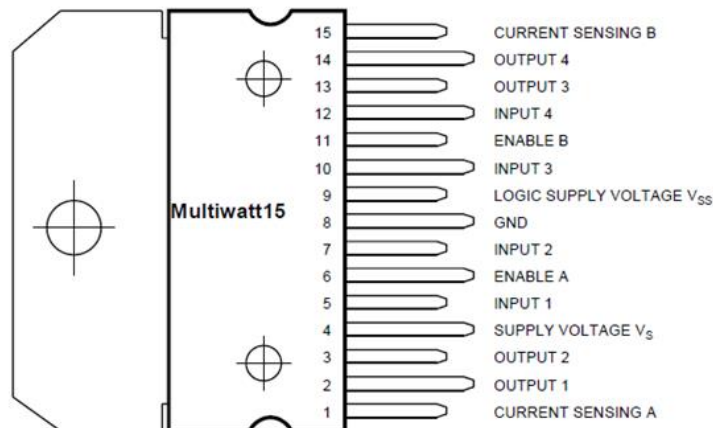
Gambar 2.5 Motor DC⁵

⁵<http://wahyu-umiq.blogspot.co.id/2013/10/motor-dc.html>, diakses tanggal: 11 Januari 2017

Konstruksi motor DC pada dasarnya memiliki 2 bagian utama. Bagian yang tetap/stasioner yang disebut stator. Stator ini menghasilkan medan magnet, baik yang dibangkitkan dari sebuah koil (elektro magnet) ataupun magnet permanen. Bagian yang berputar disebut rotor, ini berupa sebuah koil dimana arus listrik mengalir. Gaya elektromagnet pada motor DC timbul saat ada arus yang mengalir pada penghantar yang berada dalam medan magnet. Medan magnet itu sendiri ditimbulkan oleh magnet permanen. Garis-garis gaya magnet mengalir diantara dua kutub magnet dari kutub N ke kutub S. Menurut hukum gaya Lorentz, arus yang mengalir pada penghantar yang terletak dalam medan magnet akan menimbulkan gaya. Gaya F , timbul tergantung pada arah arus I , dan arah medan magnet B .

2.3 Driver Motor DC L298

L298 adalah integrated circuit (IC) yang dapat digunakan sebagai *driver* motor DC. IC ini menggunakan prinsip kerja *H-Bridge*. Tiap *H-Bridge* dikontrol menggunakan level tegangan TTL yang berasal dari *output* mikrokontroler. L298 dapat mengontrol 2 buah motor DC. Tegangan yang dapat digunakan untuk mengendalikan motor bisa mencapai tegangan $46V_{DC}$ dan arus $2A$ untuk setiap kanalnya. Berikut ini bentuk IC L298 (gambar 2.6) yang digunakan sebagai motor *driver*.



Gambar 2.6 Driver motor DC L298⁶

Pengaturan kecepatan motor dilakukan dengan cara pengontrolan lama pulsa aktif (mode PWM – *Pulse width Modulation*) yang dikirimkan ke rangkaian *driver* motor oleh pengendali (mikrokontroler). *Duty cycle* PWM yang dikirimkan menentukan kecepatan putar motor DC. PWM (*Pulse width Modulation*), adalah sebuah metode untuk pengaturan kecepatan perputaran, dalam hal ini adalah motor DC. PWM dapat dihasilkan oleh empat metode, yaitu metode analog, metode digital, IC diskrit, dan mikrokontroller.

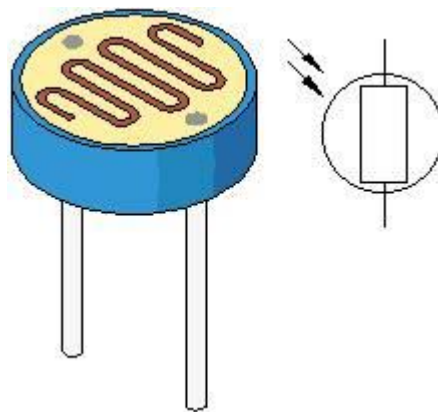
Pada alat ini, metode PWM akan dikerjakan oleh mikrokontroller. Metode PWM ini akan mengatur lebar atau sempitnya periode pulsa aktif yang dikirimkan oleh mikrokontroler ke driver motor. Secara *analog* besaran PWM dihitung dalam prosentase, nilai ini didapat dari perbandingan: $T_{high} / (T_{high} + T_{low}) \times 100\%$. Dimana T adalah periode atau waktu tempuh untuk sebuah pulsa, yang terbagi menjadi bagian puncak positif (*T_{high}*) dan puncak negatif (*T_{low}*).

⁶ <http://www.robotics-university.com/2015/01/driver-motor-dcmp-menggunakan-ic-l298.html>, diakses tanggal: 11 Januari 2017

2.4 Light Dependent Resistor (LDR)

Light Dependent Resistor (LDR) merupakan suatu sensor yang tahanannya akan berubah seiring intensitas cahaya yang diterimanya. Light Dependent Resistor merupakan sebuah cakram semikonduktor yang permukaannya mempunyai dua buah elektroda. Pada saat cahaya redup atau gelap, bahan dari semikonduktor tersebut menghasilkan elektron bebas dengan jumlah relatif kecil, sehingga hanya sedikit elektron untuk mengangkut muatan elektrik.

Hal ini menjadikan LDR sebagai konduktor yang buruk atau mempunyai resistansi tinggi pada saat cahaya redup atau gelap. Pada saat cahaya terang bahan semikonduktor tersebut menghasilkan lebih banyak elektron sehingga lebih mudah mengangkut muatan elektrik. Hal ini menjadikan LDR sebagai konduktor yang baik atau mempunyai resistansi yang rendah pada saat cahaya terang. LDR ditunjukkan pada gambar 2.7.



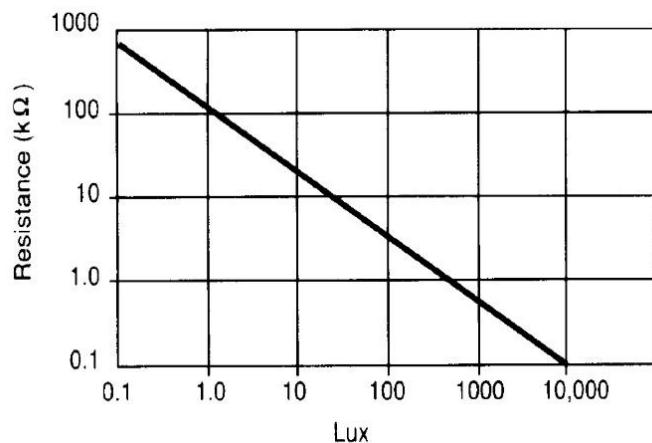
Gambar 2.7 LDR (Light Dependent Resistor)⁷

⁷ <http://elektronika-dasar.web.id/sensor-cahaya-ldr-light-dependent-resistor>, diakses tanggal: 11 Januari 2017

2.4.1 Karakteristik Light Dependent Resistor (LDR)

A. Laju Recovery

Nilai resistansi LDR tidak akan segera berubah apabila dibawa dari suatu ruangan dengan level kekuatan cahaya tertentu ke dalam suatu ruangan yang gelap. Resistansi LDR tersebut hanya akan mencapai harga dikegelapan setelah mengalami selang waktu tertentu. Laju recovery merupakan suatu ukuran praktis atau suatu kenaikan nilai resistansi dalam waktu tertentu. Nilai ini ditulis dalam $K\Omega/\text{detik}$, untuk LDR tipe arus nilainya lebih besar dari $200K\Omega/\text{detik}$ (selama 20 menit pertama mulai dari level cahaya 100 lux), kecepatan tersebut akan lebih tinggi pada arah sebaliknya, yaitu pindah dari tempat gelap ke tempat terang yang memerlukan waktu kurang dari 10 ms untuk mencapai resistansi yang sesuai dengan level cahaya 400 lux.

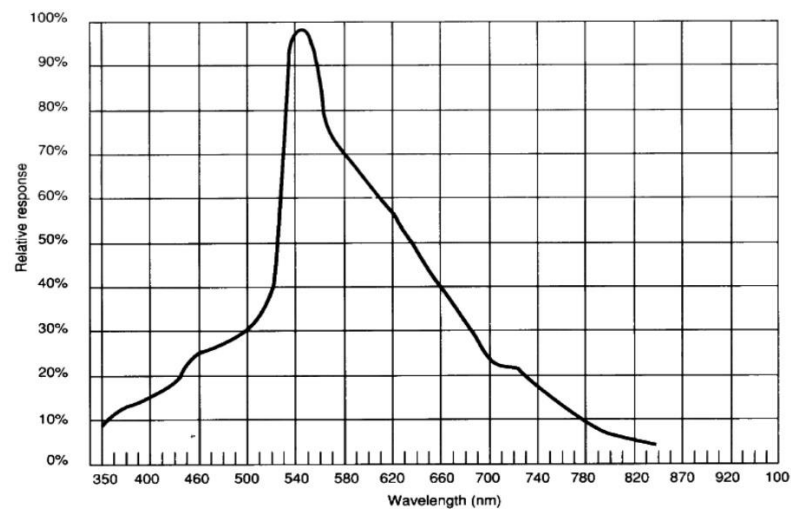


Gambar 2.8 Karakteristik tahanan terhadap intensitas cahaya pada LDR⁸

^{8,9} <http://kennarar.vma.is/thor/v2011/vgr402/ldr.pdf>, diakses tanggal: 3 Maret 2017

B. Respon Spektral

Seperti mata manusia, LDR mempunyai kepekaan yang relatif pada setiap panjang gelombang cahaya yang diterimanya (warna). Setiap jenis bahan (material) fotokonduktor memiliki keunikan tersendiri. Gambar 2.9 kurva relatif respon fotocell terhadap panjang gelombang cahaya.

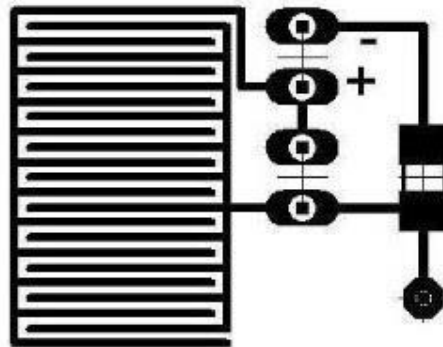


Gambar 2.9 Respon spektral⁹

2.5 Sensor Hujan

Rangkaian sensor hujan ini dirancang untuk mendeteksi air pada saat turun hujan tetapi juga dapat digunakan untuk mendeteksi level air dan lain-lainnya. Rangkaian ini menggunakan komponen resistor sebagai komponen utama dan elektroda sebagai pendeteksi air. Dari gambar 2.10 dapat dilihat ketika air menyentuh kedua elektroda (tembaga) maka tegangan 5V akan terhubung dengan output dan sebagian tegangan akan berkurang karena air berfungsi sebagai

penghambat. Tegangan keluarannya sebesar 3v sampai 4.5v dengan jarak antara kedua elektroda + 2mm dan resistor yang digunakan sebesar 10k ohm sampai 100k ohm. Untuk mendeteksi air hujan dengan kawasan yang besar maka elektroda dibuat berliku-liku, sebagai contoh dapat dilihat seperti gambar di bawah ini.

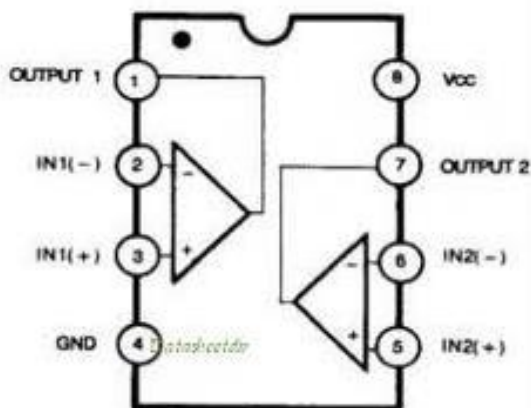


Gambar 2.10 Rangkaian elektroda sensor hujan

Dengan metode berliku-liku seperti itu akan mengurangi hambatan dari air hujan dan tegangan keluar setara dengan logika 1. Untuk menghindari karat atau tertutup kotoran yang menyebabkan sensor tidak bekerja, jalur tersebut harus dilapisi timah atau apa saja yang dapat menyatu dengan jalur tersebut dan dapat mengantarkan arus listrik.

2.6 Komparator LM393

Komparator LM393 adalah suatu perangkat elektronik yang sering digunakan dalam rangkaian sistem kontrol. Komparator LM393 memiliki fungsi sebagai alat untuk membandingkan dua buah nilai *input*. Komparator ini memiliki 2 buah input dan sebuah output. Inputnya yaitu input (+) dan input (-). Komparator ini berfungsi saat sinyal *analog* digunakan untuk memicu suatu kejadian.



Gambar 2.11 Skema IC LM393¹⁰

Skema IC LM393 ditunjukkan pada gambar 2.11. IC LM393 memiliki dua buah komparator atau pembanding yang akurat dan tidak saling berhubungan di dalamnya. Komparator ini memiliki dua cara kerja yaitu *inverting* (-) dan *non-inverting* (+). Komparator ini juga memiliki *output* yaitu *low* atau *high*. Cara kerja komparator memiliki ketentuan sebagai berikut:

1. Inverting: $V_- > V_+ = \text{Low}$ $V_- \leq V_+ = \text{High}$
2. Non-inverting: $V_+ > V_- = \text{High}$ $V_+ \leq V_- = \text{Low}$

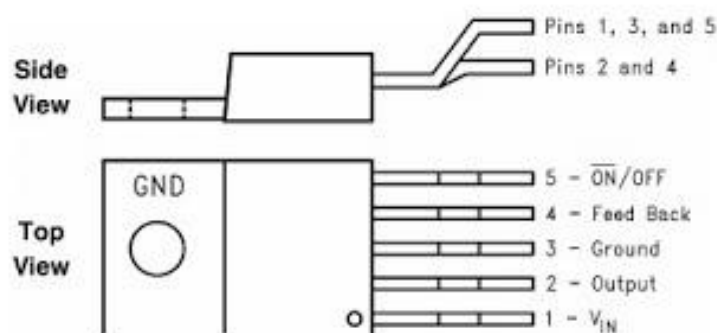
Dalam aplikasinya salah satu pin input dari komparator sebagai tegangan referensi sedangkan pin input lainnya sebagai tegangan yang akan dibandingkan.

2.7 Regulator LM2596

Regulator LM2596 adalah sebuah sirkuit terpadu (IC) yang berfungsi sebagai penurun (step-down) tegangan DC dengan beban maksimal 3A. Komponen ini tersedia dalam versi keluaran (output) yang dapat diatur. LM2596 mempunyai kompensasi internal untuk meminimalkan jumlah komponen eksternal untuk

¹⁰ <http://himatro.ee.unila.ac.id/komparator-lm393>, diakses tanggal: 5 Mei 2017

menyederhanakan desain catu daya. Regulator LM2596 sangat efisien dalam menangani voltase masukan yang lebih tinggi. LM2596 beroperasi pada frekuensi switching 150 kHz yang memungkinkan komponen filter berukuran lebih kecil dari yang dibutuhkan dengan regulator pengubah frekuensi rendah. Skema dari Regulator LM2596 ditunjukkan pada gambar 2.12.



Gambar 2.12 Skema Regulator LM2596¹¹

Fitur lainnya berupa toleransi sebesar 4% yang terjamin pada output tegangan pada kondisi tegangan input dan kondisi beban output tertentu, dan 15% pada frekuensi osilator. Fitur perlindungan diri termasuk batas arus, peralihan arus per siklus untuk saklar keluaran, seperti termal shutdown untuk perlindungan lengkap karena kesalahan kondisi.

2.8 Short Message Service (SMS)

Short message service adalah salah satu tipe Instant Messaging (IM) yang memungkinkan user untuk bertukar pesan singkat kapanpun, walaupun user sedang melakukan sambungan data/suara. SMS dihantarkan pada channel signal GSM

¹¹ <http://www.ti.com/lit/ds/symlink/lm2596.pdf>, diakses tanggal: 5 Mei 2017

(Global System for Mobile) spesifikasi teknis ETSI (European Telecommunication Standards Institute). SMS diaktifkan oleh ETSI dan dijalankan di scope 3GPP. SMS menjamin pengiriman pesan oleh jaringan, jika terjadi kegagalan pesan akan disimpan dahulu di jaringan, pengiriman paket SMS bersifat out of band dan menggunakan bandwidth rendah.

Teknologi SMS merupakan suatu teknologi yang tidak asing lagi dalam kehidupan masyarakat. Teknologi SMS ini banyak digemari oleh masyarakat karena teknologi ini bersifat praktis, murah dan mudah untuk digunakan. SMS merupakan suatu sistem pengiriman pesan sederhana yang disediakan oleh jaringan telepon selular. Fitur SMS ini didukung oleh GSM (Global System for Mobile), TDMA (Time Multiple Digital Access), CDMA (Code Multiple Digital Access). Semakin pesatnya perkembangan teknologi SMS ini, didukung oleh beberapa faktor, antara lain adalah semakin terjangkau harga perangkat keras yang digunakan (telepon selular). Selain itu, faktor lain yang mempengaruhi adalah banyaknya provider penyedia jasa telekomunikasi yang menawarkan jasanya dengan harga yang cukup terjangkau oleh masyarakat saat ini.

2.8.1 AT Command

Perintah AT (AT Command) digunakan untuk berkomunikasi dengan terminal melalui serial port pada computer. AT Command merupakan inti dari sebuah SMS gateway karena berisi perintah-perintah yang memungkinkan sebuah komputer untuk memberikan instruksi ke terminal. Umumnya AT Command digunakan pada Hyper Terminal.

Secara umum AT Command terdiri dari tiga bagian, yang dimulai dengan AT, diikuti dengan perintah dan diakhiri dengan karakter penghentian baris. Pengecualian pada A/ dan +++ yang dikirim tanpa menggunakan AT dan umpan balik. Respon dari masing-masing perintah bervariasi dan di dokumentasikan dalam teks pengikut. Secara umum format dapat dituliskan “AT+perintah=karakter” //karakter pengikut. AT command dapat digunakan untuk menginstruksikan perintah-perintah sebagai berikut:

1. Mengirim dan menerima pesan SMS atau faks.
2. Mendapatkan informasi mengenai device, misalnya nama manufaktur, nomor IMEI, dan lain-lain.
3. Mendapatkan status device, misalnya status aktivitas, status registrasi network, kekuatan sinyal, ataupun status baterai.
4. Penulisan dan pencarian phonebook.
5. Dapat mengaktifkan fasilitas lock dan mengubah password
6. Menyimpan dan mengembalikan konfigurasi.

Tabel 2.1 Daftar AT Command¹²

AT Command	Keterangan
AT + CMGS	Mengirim pesan
AT + CMGL	Menampilkan daftar SMS dalam inbox
AT + CMGR	Membaca sms
AT + CMGD	Menghapus SMS
AT + CMGF	Mengoperasikan alat dalam format text
AT + CSCA	Informasi nomor Service Center
AT + CGSN	Mengetahui nomor IMEI
AT + COPS	Mengetahui operator jaringan
AT + GMM	Mengetahui informasi modem

¹² <http://tutordaninfogoblog.blogspot.co.id/2013/02/informasi-perintah-at-command.html>, diakses tanggal: 13 Maret 2017

2.8.2 Modul GSM SIM800L

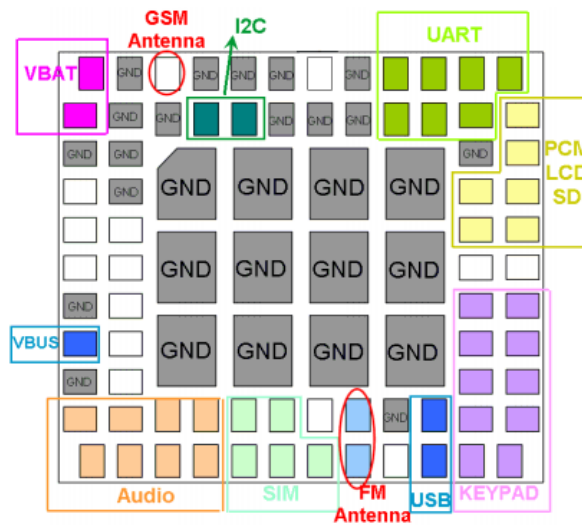
SIM800L adalah quad band GSM/GPRS Modul yang dapat kita gunakan bersama mikrokontroller baik untuk fitur SMS, telepon ataupun data GPRS. Modul ini bekerja pada frekuensi GSM850Mhz, EGSM900Mhz, DCS1800Mhz, dan PCS1900Mhz sehingga fleksibel untuk digunakan bersama kartu SIM dari berbagai operator telepon seluler di Indonesia. SIM800L memiliki fitur GPRS multi-slot class 12/class 10 dan mendukung skema pengkodean GPRS CS-1, CS-2, CS-3 dan CS-4. Bentuk fisik SIM800L ditunjukkan pada gambar 2.13.



Gambar 2.13 SIM800L¹³

SIM800L memiliki 88pin (gambar 2.14) yang disediakan untuk dukungan ke semua perangkat keras antara modul dan pembuat board. SIM800L di desain dengan teknik penghemat energi yang hanya mengonsumsi daya kurang dari 0,7mA pada saat mode tidur. Modul ini menggunakan port TTL serial port, sehingga dapat langsung diakses menggunakan mikrokontroler tanpa perlu memerlukan MAX232.

^{13,14} <http://www.datasheetcafe.com/sim800l-datasheet-simcom/>, diakses tanggal 13 Maret 2017



Gambar 2.14 Pin-pin Modul GSM SIM800L¹⁴

2.9 Bahasa Pemrograman Mikrokontroler AVR ATmega16

Pengembangan sebuah sistem menggunakan mikrokontroler AVR buatan ATMEL menggunakan *software* AVR STUDIO dan CodeVision AVR. AVR STUDIO merupakan *software* khusus untuk bahasa *assembly* yang mempunyai fungsi sangat lengkap, yaitu digunakan untuk menulis program, kompilasi, simulasi dan *download* program ke IC mikrokontroler AVR. Sedangkan CodeVision AVR merupakan *software C-cross compiler*, dimana program dapat ditulis dalam bahasa C.

CodeVision memiliki IDE (Integrated Development Environment) yang lengkap, dimana penulisan program, *compile*, *link*, pembuatan kode mesin (*assembler*) dan *download* program ke *chip* AVR dapat dilakukan pada CodeVision, selain itu ada fasilitas terminal, yaitu untuk melakukan komunikasi serial dengan mikrokontroler yang sudah diprogram. Proses *download* program ke

IC mikrokontroler AVR dapat menggunakan *downloader* secara ISP (*In-System Programming*). *In-System Programming flash on-chip* mengizinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI.

2.9.1 Bahasa C

Berikut ini penjelasan aturan penulisan program dalam bahasa C. Untuk seterusnya, pemrograman mikrokontroler AVR menggunakan bahasa C dengan penjelasan sebagai berikut:

Penulisan program dalam bahasa C

```
#include<mega16.h>

#include <delay.h>

#define      IRsensor      PINA.0

#define      pompa        PORTB.0

//variable global

unsigned int I,j;

void main(void)

{

//variable local

Chart data_rx;

DDRA=0x00;

PORTA=0xFF;

DDRB=0xFF;

PORTB=0x00;

....
```

```

....

While(1)

{

.....

.....

};

}

```

Penjelasan:

Preprocessor (#): digunakan untuk memasukan (*include*) *text* dan *file* lain, mendefinisikan *macro* yang dapat mengurangi beban kerja pemrograman dan meningkatkan *legibility source code* (mudah dibaca).

#define: digunakan untuk mendefinisikan *macro*

Contoh	<i>#define</i>	ALFA	0xff
	<i>#define</i>	SUM(a,b)	a+b
	<i>#define</i>	sensor	PINA.2
	<i>#define</i>	pompa	PORTB.0

Komentar :

penulisan komentar untuk beberapa baris komentar sekaligus

```
/*....komentar....*/
```

Penulisan untuk satu baris saja

```
//....komentar....
```

2.9.1.1 Identifiers

Identifier adalah nama yang diberikan pada *variable*, fungsi, label, atau objek lain. *Identifier* dapat mengandung huruf (A ...Z, a ...z) dan angka (0 ... 9) dan karakter (_). *Identifiers* bersifat *Case sensitive*. *Identifier* dapat mencapai maksimal 32 karakter.

2.9.1.2 Konstanta

Konstanta *integer* dan *long integer* ditulis dalam bentuk *decimal* (1234), dalam bentuk biner (0b101001), *hexadecimal* mempunyai awalan 0x (0xff), atau dalam *octal* dengan awalan 0 (0777). *Unsigned integer* mempunyai akhiran U(10000U). *Long integer* mempunyai akhiran L(99L). *Unsigned long integer* mempunyai akhiran UL(99UL). *Floating point* mempunyai akhiran F(1.234F). Konstanta karakter harus di lingkungi oleh tanda kutip ('a')

2.9.1.3 Program Control

a. Percabangan

Perintah *if* dan *if ... else ...*

Perintah *if* dan *if ... else ...* digunakan untuk melakukan operasi percabangan bersyarat. Fungsi-fungsi untuk menetapkan kondisi dapat dilihat dalam tabel. Sintaks penulisan *if* dapat ditulis sebagai berikut:

```
if(<expression>)<statement>;
```

Sintak perintah *if ... else ...* dapat ditulis sebagai berikut:

```
if(<expression>)<statement>;
```

```
else<statement2>;
```

jika hasil *testing expression* memberikan hasil tidak nol *statement1* akan dilaksanakan. Pada keadaan sebaliknya *statement2* yang akan dilaksanakan. Sebaiknya pemanfaatan perintah *if* untuk beberapa kondisi dilakukan dengan menggunakan blok-blok.

b. Looping (Pengulangan)

Looping adalah pengulangan satu atau beberapa perintah sampai mencapai keadaan tertentu. Ada tiga perintah *looping*, yaitu: *for...*, *while...*, dan *do...while...* Syarat_loop adalah pernyataan rasional yang menyatakan syarat berhentinya pengulangan, biasanya berkaitan dengan *variable control*, *nama_variable++* dan *nama_variable--*, menyatakan proses *increment* dan proses *decrement* pada *variable* kontrol.

Perintah *while* dapat melakukan *looping* apabila persyaratannya benar. Perintah *while* terlebih dahulu melakukan pengujian persyaratan sebelum melakukan *looping*. Kadang-kadang hal ini menimbulkan keropotan-keropotan yang tidak perlu, misalnya inisialisasi *variable control*. Salah satu solusi adalah dengan menggunakan loop *do ...while*.

2.9.1.4 Array

Array adalah deretan *variable* yang berjenis sama dan mempunyai nama yang sama. Setiap anggota deretan (elemen) diberi nomor yang disebut indeks, dimulai dari indeks nol. *Array* diatur agar mempunyai lokasi memori yang bersebelahan dengan alamat terkecil menuju elemen *array* pertama dan alamat terbesar menunjukkan elemen terakhir. Elemen *array* dapat diakses dengan menggunakan indeksnya. Bentuk deklarasi *array* adalah:


```
Tipe nama_array[ukuran]
```

```
Int nilai[100];
```

```
Nilai[1]=10;
```

```
Niali[2]=3;
```

2.9.1.5 Fungsi

Sebuah program yang besar dapat dipecah menjadi beberapa subprogram yang terpisah yang melakukan fungsi tertentu. Subprogram yang seperti itu disebut fungsi. Sebagai contoh, sebuah program yang melakukan proses pengisian data berulang kali dapat dilengkapi dengan sebuah *fungsi* yang bertugas untuk melakukan proses pengisian data. Apabila program hendak melakukan proses pengisian data, program dapat melakukan pemanggilan *fungsi* tersebut.

Fungsi adalah sebuah blok yang melingkupi beberapa perintah. Parameter dalam *fungsi* ada 2 yaitu tipe dan argumen. Tipe adalah nilai yang dihasilkan oleh *fungsi*, jika tidak dinyatakan, hasil *fungsi* dianggap bertipe integer. Deklarasi tipe *void* dapat dimanfaatkan untuk menghindari terjadinya nilai balik. Argumen adalah deklarasi variable apa saja yang dibutuhkan *fungsi* dan bersifat *optional*. Deklarasi *fungsi* dapat dilakukan dengan cara:

```
Tipe nama_fungsi(argumen)
```