

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Sensor**

Sensor adalah sesuatu yang digunakan untuk mendeteksi adanya perubahan lingkungan fisik atau kimia. Variabel keluaran dari sensor yang diubah menjadi besaran listrik disebut transduser. Pada saat ini, sensor tersebut telah dibuat dengan ukuran sangat kecil dengan orde nanometer. Ukuran yang sangat kecil ini sangat memudahkan pemakaian dan menghemat energi.

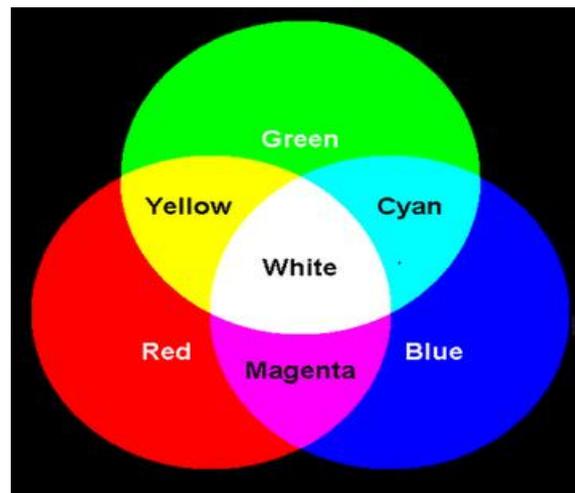
Sensor adalah suatu peralatan yang berfungsi untuk mendeteksi gejala-gejala atau sinyal-sinyal yang berasal dari perubahan suatu energi seperti energi listrik, energi kimia, energi fisika, energi biologi, energi mekanik dan sebagainya.

##### **2.1.1 Sensor Citra**

Citra adalah suatu representasi (gambaran), kemiripan, atau inisiasi dari suatu objek. Citra sebagai keluaran suatu sistem perekaman data yang dapat bersifat optik berupa foto, bersifat analog berupa sinyal-sinyal video seperti gambar pada monitor televisi, atau bersifat digital yang dapat langsung disimpan pada suatu media penyimpanan (Indra Agustian, 2012).

Pengolahan citra digital menggunakan teknologi *Computer Vision* saat ini banyak digunakan sebagai objek penelitian. Bagian dari pengolahan citra adalah dengan menggunakan pengolahan berdasarkan warna. Analisis warna dalam pengenalan citra digital ini ada beberapa model diantaranya, model RGB, CMY, HIS, HSV, dan *normalized RGB*.

Pembentukan warna pada gambar fotografi sebenarnya hanya terdiri dari tiga warna yaitu merah, hijau, dan biru atau disebut *additive color system*. Apabila ketiga warna ini digabungkan dengan intensitas yang sama akan membentuk warna putih. Penggabungan dua warna dengan intensitas yang sama akan menghasilkan warna baru : *Red-Green = Yellow, Green-Blue = Cyan, Blue-Red = Magenta*.



**Gambar 2.1 Additive Color System**

(Sumber : [http://facweb.cs.depaul.edu/sgrais/color\\_additive%20system.htm](http://facweb.cs.depaul.edu/sgrais/color_additive%20system.htm))

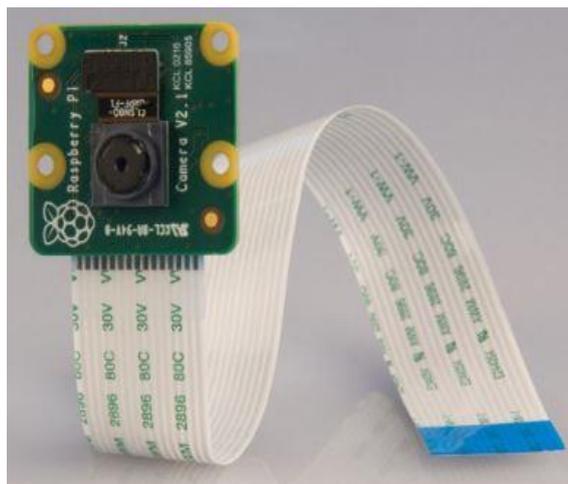
Gambar 2.1 tersebut merupakan proses penyaringan warna, setiap cahaya yang ditangkap oleh piksel *photosites* akan tersaring menurut warna yang digunakan untuk proses penyaringan sesuai dengan intensitasnya.

#### **2.1.1.1 Raspberry Pi Camera**

*Raspberry Pi Camera* merupakan sebuah modul sensor *camera* yang berisi sensor gambar CMOS 5 MP dari OmniVision yang dirancang dengan arsitektur piksel iluminasi sisi belakang 1,4 mikron yang menghadirkan fotografi 5 *megapixel* dan rekaman video *high-definition* (HD) dengan laju bingkai tinggi 720p/60. Modul ini berantarmuka dengan *Raspberry Pi* melalui konektor kamera yang ada, dengan menggunakan CSI untuk data dan I2C untuk kontrol. Ini memungkinkan pengguna untuk merekam 720p dan 1080p dengan laju 30 bingkai per detik dalam format video H264. *Raspberry Pi Camera* sendiri biasanya digunakan untuk keperluan konferensi jarak jauh atau juga sebagai kamera pemantau.

*Raspberry Pi Camera* adalah sebuah periferal berupa kamera sebagai pengambil citra/gambar yang dikendalikan oleh sebuah komputer atau juga oleh jaringan komputer. Gambar yang diambil oleh *Raspberry Pi Camera* ditampilkan ke layar monitor dan dikendalikan dengan *interface* atau *port* yang digunakan untuk menghubungkan *Raspberry Pi Camera* dengan komputer atau jaringan.

*Raspberry Pi Camera* hampir sama dengan *webcam* sebagai *web pages camera*, karena dengan menggunakan *webcam* untuk mengambil gambar video secara aktual bisa langsung diupload bila komputer yang mengendalikan terkoneksi dengan internet, dan *Raspberry Pi Camera* juga demikian. *Raspberry Pi Camera* atau *webcam* sangat bermanfaat dalam bidang telekomunikasi, keamanan, dan industri. Sebagai contoh *webcam* digunakan untuk *video call chatting*, *surveillance camera*, dan juga sebagai *video conference* oleh beberapa *user*. Gambar 2.2 berikut adalah gambar fisik *Raspberry Pi Camera*.



**Gambar 2.2 Modul *Raspberry Pi Camera***

(Sumber : <https://www.raspberrypi.org/blog/new-8-megapixel-camera-board/>)

### **2.1.1.2 Cara Kerja *Raspberry Pi Camera***

Sebuah *Raspberry Pi Camera* yang sederhana terdiri dari sebuah lensa standar dan dipasang di sebuah papan sirkuit untuk menangkap sinyal gambar, *casing*, termasuk *casing* depan dan *casing* samping untuk menutupi lensa standar dan memiliki sebuah lubang lensa di *casing* depan yang berguna untuk memasukkan gambar kabel *support* yang dibuat dari bahan fleksibel, salah satu ujungnya dihubungkan dengan papan sirkuit dan ujung satunya lagi memiliki *connector*, kabel ini dikontrol untuk menyesuaikan ketinggian, arah, dan sudut pandang *Raspberry Pi Camera*.

Sebuah *Raspberry Pi Camera* biasanya dilengkapi *software*. *Software* ini mengambil gambar-gambar dari kamera digital secara terus menerus ataupun

dalam interval waktu tertentu dan menyiarkannya. Ada beberapa metode penyiaran, metode yang paling umum adalah *hardware* mengubah gambar ke dalam bentuk file JPG. *Frame rate* ini mengindikasikan jumlah gambar sebuah *software* dapat diambil dan transfer dalam satu detik. Untuk *streaming* video, dibutuhkan minimal 15 *frame per second* (fps) atau idealnya 30 fps. Untuk mendapatkan *frame rate* yang tinggi, dibutuhkan koneksi internet yang tinggi kecepatannya. Sebuah *Raspberry Pi Camera* atau *webcam* tidak harus selalu terhubung dengan komputer, ada yang memiliki *software webcam* dan *web server built in*, sehingga yang diperlukan hanyalah koneksi internet.

#### **2.1.1.3 Metode *Image Processing***

*Image processing* merupakan salah satu jenis keilmuan untuk menyelesaikan masalah mengenai pemrosesan gambar, serta untuk meningkatkan kualitas penampakan gambar agar lebih mudah diinterpretasikan oleh sistem penglihatan manusia, baik dengan melakukan manipulasi dan juga penganalisisan terhadap gambar.

Teknik manipulasi citra secara digital yang khususnya menggunakan komputer, menjadi citra lain yang sesuai untuk digunakan dalam aplikasi tertentu. Agar mudah diinterpretasi oleh manusia dan komputer, *image processing* harus dilakukan dengan berbagai macam metode untuk mencapai citra sesuai yang diinginkan. Operasi *Image processing* digital umumnya dilakukan dengan tujuan memperbaiki kualitas suatu gambar sehingga dapat dengan mudah diinterpretasikan oleh mata manusia dan mengolah informasi yang ada pada suatu gambar untuk kebutuhan identifikasi objek secara otomatis.

#### **2.1.1.4 Aplikasi *Raspberry Pi Camera* Sebagai Sensor Citra**

*Raspberry Pi Camera* sebagai sensor citra tentunya diaplikasikan untuk mengambil gambar atau video, namun *Raspberry Pi Camera* ini digunakan untuk mengambil video bukan untuk disimpan dalam suatu *memory* melainkan untuk diproses ulang oleh sebuah mikroprosesor. *Raspberry Pi Camera* yang berantar mukakan *CSI Camera*, dikoneksikan ke *port CSI Camera* sebuah mini komputer dalam hal ini adalah sebuah *Raspberry Pi*, dan mikroprosesor ini akan menerima

*Raspberry Pi Camera* sebagai *input device* dan langsung siap digunakan untuk pemrosesan *input data*.

#### 2.1.1.5 Aplikasi *Raspberry Pi* Sebagai *Controller* Untuk *Raspberry Pi Camera*

Untuk *Raspberry Pi Camera* dapat digunakan dalam aplikasi ini, *Raspberry Pi* digunakan sebagai pengontrol *Raspberry Pi Camera* untuk mendapat acuan posisi suatu objek dari apa yang dilihat oleh *Raspberry Pi Camera* dan memprosesnya untuk *input trigger* mikrokontroler pengontrol *driver motor dc*. Dengan memanfaatkan fungsi kamera untuk diambil data koordinat posisi tertengah suatu daerah cakupan sekitar satu *pixel* yang tertengah pada suatu warna dasar dari hasil konversi warna RGB ke HSV oleh mikroprosesor.

#### 2.1.2 Sensor Jarak

Sensor jarak adalah sebuah sensor yang mampu mendeteksi keberadaan benda didekatnya tanpa kontak fisik. Sensor ini bekerja berdasarkan prinsip kerja pantulan gelombang suara, dimana sensor menghasilkan gelombang suara yang kemudian menangkap kembali dengan perbedaan waktu sebagai dasar pengindra. Perbedaan waktu antara gelombang suara yang dipancarkan dan diterima kembali adalah berbanding lurus dengan jarak atau tinggi objek yang memantulkannya. Jenis objek yang dapat diindranya adalah zat padat dan zat cair. Sensor jarak sering memancarkan elektromagnetik atau berkas radiasi elektromagnetik (*infrared*) dan mencari perubahan pada sinyal kembali.

##### 2.1.2.1 Sensor Ultrasonik HC-SR04

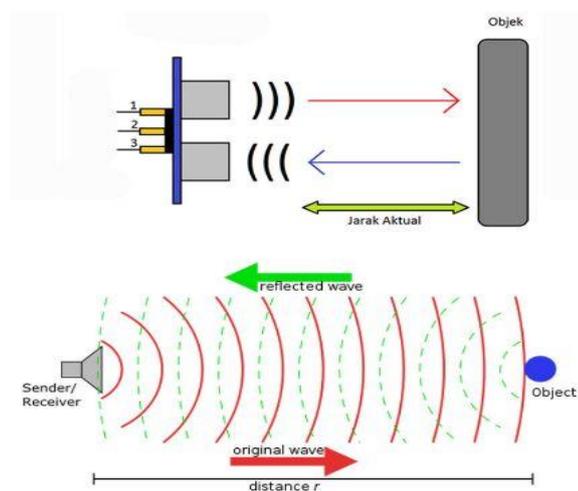
HC-SR04 merupakan sensor ultrasonik yang dapat digunakan untuk mengukur jarak antara penghalang dan sensor. Tampilan sensor HC-SR04 diperlihatkan pada Gambar 2.3 berikut ini.



**Gambar 2.3 Tampilan Sensor Ultrasonik HC-SR04**

(Sumber : <https://www.makerlab-electronics.com/ultrasonic-sensor-hc-sr04/>)

HC-SR04 memiliki 2 komponen utama sebagai penyusunnya, yaitu *ultrasonic transmitter* dan *ultrasonic receiver*. Fungsi dari *ultrasonic transmitter* adalah memancarkan gelombang ultrasonik dengan frekuensi 40 KHz kemudian *ultrasonic receiver* menangkap hasil pantulan gelombang ultrasonik yang mengenai suatu objek. Waktu tempuh gelombang ultrasonik dari pemancar hingga sampai ke penerima sebanding dengan 2 kali jarak sensor dan bidang pantul seperti yang diperlihatkan pada Gambar 2.4 berikut ini.



**Gambar 2.4 Prinsip Kerja HC-SR04**

(Sumber : <http://www.elangsakti.com/2015/05/sensor-ultrasonik.html>)

Prinsip pengukuran jarak menggunakan sensor ultrasonik HC-SR04 adalah ketika pulsa *trigger* diberikan pada sensor, *transmitter* akan memulai memancarkan gelombang ultrasonik, pada saat yang sama sensor akan menghasilkan *output* TTL. Transisi naik menandakan sensor mulai menghitung waktu pengukuran, setelah *receiver* menerima pantulan yang dihasilkan oleh suatu objek maka pengukuran waktu akan dihentikan dengan menghasilkan transisi turun *output* TTL. Jika waktu pengukuran adalah  $t$  dan kecepatan suara adalah 340 m/s, maka jarak antara sensor dengan objek dapat dihitung dengan menggunakan Persamaan 2.1 berikut ini.

$$s = t \times \frac{340 \text{ m/s}}{2} \dots\dots\dots (2.1)$$

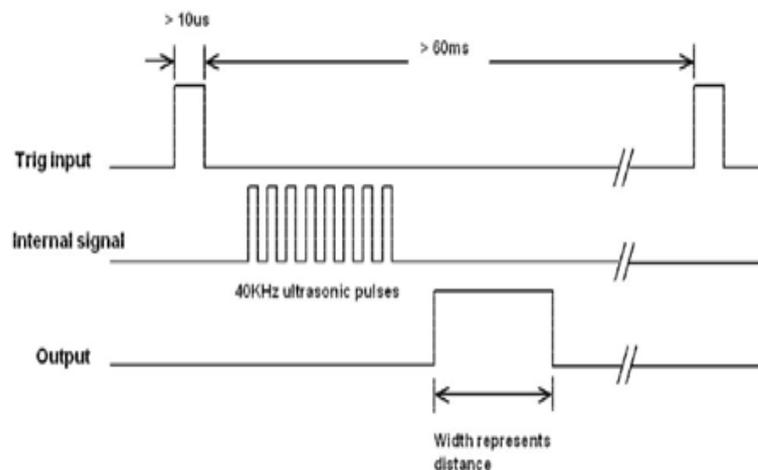
Dimana :

$s$  = Jarak antara sensor dengan objek (m)

$t$  = Waktu tempuh gelombang ultrasonik dari *transmitter* ke *receiver* (s)

Pemilihan HC-SR04 sebagai sensor jarak memiliki fitur sebagai berikut : kinerja yang stabil, pengukuran jarak yang akurat dengan ketelitian 0,3 cm, pengukuran maksimum dapat mencapai 4 meter dengan jarak minimum 2 cm, ukuran yang ringkas dan dapat beroperasi pada level tegangan TTL.

Prinsip pengoperasian sensor ultrasonik HC-SR04 adalah sebagai berikut : diawali dengan memberikan pulsa *Low* (0) ketika modul mulai dioperasikan, kemudian diberikan pulsa *High* (1) pada *trigger* selama 10  $\mu$ s sehingga modul mulai memancarkan 8 gelombang kotak dengan frekuensi 40 KHz, tunggu hingga transisi naik terjadi pada *output* dan mulai perhitungan waktu hingga transisi turun terjadi, setelah itu gunakan Persamaan 2.1 untuk mengukur jarak antara sensor dengan objek. *Timing Diagram* pengoperasian sensor HC-SR04 diperlihatkan pada Gambar 2.5 berikut ini.



**Gambar 2.5 Timing Diagram Pengoperasian Sensor HC-SR04**

(Sumber : <https://wisuda.unud.ac.id/pdf/1108205006-3-4%20BAB%20II.pdf>)

## 2.2 Mikrokontroler

Mikrokontroler adalah sistem mikroprosesor lengkap yang terkandung di dalam sebuah *chip*. Mikrokontroler berbeda dari mikroprosesor serba guna yang digunakan dalam sebuah PC, karena di dalam sebuah mikrokontroler umumnya juga berisi komponen pendukung sistem minimal mikroprosesor, yakni *memory* dan antarmuka I/O, sedangkan didalam mikroprosesor umumnya hanya berisi CPU saja. Sebuah *chip* yang berfungsi sebagai pengontrol rangkaian elektronik dan umumnya dapat menyimpan program di MCS51 ialah mikrokomputer CMOS 8 bit dengan 4 KB *Flash PEROM (Programmable and Erasable Only Memory)* yang dapat dihapus dan ditulis sebanyak ribuan kali.

### 2.2.1 Arduino Uno

*Arduino Uno* adalah pengendali mikro *single-board* yang bersifat *open-source*, yang dirancang untuk memudahkan penggunaan elektronik dalam berbagai bidang. *Hardware* (perangkat kerasnya) memiliki prosesor Atmel AVR dan *software* (perangkat lunaknya) memiliki bahasa pemrograman sendiri. *Open source IDE* yang digunakan untuk membuat aplikasi mikrokontroler yang berbasis *platform Arduino*. Mikrokontroler *single-board* yang bersifat *open source hardware* dikembangkan untuk arsitektur mikrokontroler AVR 8 bit dan ARM 32 bit.

Dari pengertian diatas, dapat disimpulkan bahwa *Arduino* adalah kit atau *board* rangkaian elektronik *open source* yang di dalamnya terdapat komponen utama yaitu sebuah *chip* mikrokontroler dengan jenis AVR. Mikrokontroler ini sendiri yaitu sebuah *chip* atau IC (*Integrated Circuit*) yang bisa diprogram menggunakan komputer. Tujuan menanamkan program pada mikrokontroler adalah agar rangkaian elektronik dapat membaca *input* dan memproses *input* tersebut menghasilkan *output* seperti yang diinginkan. Jadi mikrokontroler bertugas sebagai otak yang mengendalikan *input*, proses, dan *output* sebuah rangkaian elektronik.

Mikrokontroler terdapat pada perangkat elektronik sekelilingnya, misalnya *Handphone*, *MP3 Player*, *DVD*, *Televisi*, *AC*, dan lain-lain. Mikrokontroler juga dapat mengendalikan robot, baik robot mainan maupun industri. Karena

komponen utama *Arduino* adalah mikrokontroler, maka *Arduino* dapat diprogram menggunakan komponen sesuai kebutuhan.

*Arduino* dikembangkan oleh sebuah tim yang beranggotakan orang-orang dari berbagai belahan dunia. Anggota inti dari tim ini adalah Massimo Banzi (Milano, Italia), David Cuartielles (Malmoe, Swedia), Tom Igoe (USA), Gianluca Martino (Torino, Italia) dan David A. Mellis (USA).

Kelebihan *Arduino*, antara lain :

1. Tidak perlu perangkat *chip programmer* karena didalamnya sudah ada *bootloader* yang akan menangani *upload* program dari komputer.
2. Sudah memiliki sarana komunikasi USB, sehingga pengguna laptop yang tidak memiliki *port* serial / RS232 bisa menggunakannya.
3. Memiliki modul siap pakai (*shield*) yang bisa ditancapkan pada *board Arduino*. Contohnya *shield* GPS, *Ethernet*, dan lain-lain.

#### **2.2.1.1 Socket USB**

*Socket* USB adalah soket kabel USB yang disambungkan ke komputer atau laptop, yang berfungsi untuk mengirimkan program ke *Arduino* dan juga sebagai *port* komunikasi serial.

#### **2.2.1.2 Input/Output Digital dan Input Analog**

*Input/Output digital* atau *digital pin* adalah *pin-pin* untuk menghubungkan *Arduino* dengan komponen atau rangkaian *digital*, contohnya jika ingin membuat LED berkedip, LED tersebut bisa dipasang pada salah satu *pin input* atau *output digital* dan *ground* komponen lain yang menghasilkan *output digital* atau menerima *input digital* bisa disambungkan ke *pin* ini.

*Input analog* atau *analog pin* adalah *pin-pin* yang berfungsi untuk menerima sinyal dari komponen atau rangkaian *analog*, contohnya potensiometer, sensor suhu, sensor cahaya, dan lain-lain.

#### **2.2.1.3 Catu Daya**

*Pin* catu daya adalah *pin* yang memberikan tegangan untuk komponen atau rangkaian yang dihubungkan dengan *Arduino*. Pada bagian catu daya ini *pin VInput* dan *Reset*. *VInput* digunakan untuk memberikan tegangan langsung kepada *Arduino* tanpa melalui tegangan pada USB atau *adaptor*, sedangkan *Reset*

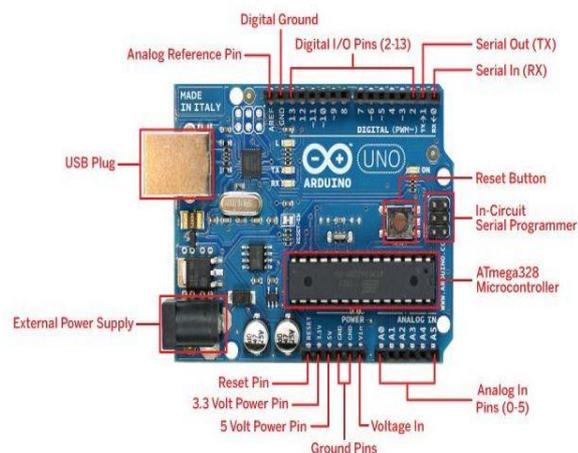
adalah *pin* untuk memberikan sinyal *reset* melalui tombol atau rangkaian eksternal.

#### 2.2.1.4 Baterai/Adaptor

Soket baterai atau *adaptor* digunakan untuk menyuplai *Arduino* dengan tegangan dari baterai atau *adaptor* 9 Volt pada saat *Arduino* sedang tidak disambungkan ke komputer. Jika *Arduino* mendapatkan suplai tegangan dari USB, maka tidak perlu memasang baterai atau *adaptor* pada saat memprogram *Arduino*.

#### 2.2.1.5 Bagian-Bagian Board Arduino Uno

Dengan mengambil contoh sebuah *Board Arduino Uno* tipe USB, bagiannya dapat dijelaskan pada Gambar 2.6 sebagai berikut.



**Gambar 2.6 Board Arduino Uno**

(Sumber : <https://www.robomart.com/arduino-uno-online-india>)

#### a) 14 *pin input output digital* (0-13)

Berfungsi sebagai *input* atau *output*, dapat diatur oleh program. Khusus untuk 6 buah *pin* 3, 5, 6, 9, 10, dan 11, dapat juga berfungsi sebagai *pin analog output* di mana tegangan *output*nya dapat diatur. Nilai sebuah *pin output analog* dengan diprogram antara 0 – 255, dimana hal itu mewakili nilai tegangan 0 – 5 Volt.

b) USB

Berfungsi untuk memuat program dari komputer ke dalam *board arduino*, komunikasi serial antara *board* dan komputer, dan memberi daya listrik kepada *board arduino*.

c) Sambungan SV1

Sambungan atau *jumper* untuk memilih sumber daya *board*, apakah dari sumber eksternal atau menggunakan USB. Sambungan ini tidak diperlukan lagi pada *board Arduino* versi terakhir karena pemilihan sumber daya eksternal atau USB dilakukan secara otomatis.

d) Q1 = Kristal (*quartz crystal oscillator*)

Jika mikrokontroler dianggap sebagai sebuah otak, maka kristal adalah jantungnya karena komponen ini menghasilkan detak-detak yang dikirim kepada mikrokontroler agar melakukan sebuah operasi untuk setiap detaknya. Kristal ini dipilih yang berdetak 16 juta kali per detik (16 MHz)

e) Tombol *Reset* S1

Untuk *mereset board* sehingga program akan mulai lagi dari awal. Perhatikan bahwa tombol *reset* ini bukan untuk menghapus program atau mengosongkan mikrokontroler.

f) *In Circuit Serial Programming* (ICSP)

*Port ICSP* memungkinkan pengguna untuk memprogram mikrokontroler secara langsung, tanpa melalui *bootloader*. Umumnya pengguna *Arduino* tidak melakukan ini sehingga ICSP tidak terlalu dipakai walaupun disediakan.

g) IC 1 – Mikrokontroler ATmega

Komponen utama dari *board Arduino*, didalamnya terdapat CPU, ROM, dan RAM.

h) X1 = Sumber Daya Eksternal

Jika hendak disuplai dengan sumber daya eksternal, *board Arduino* dapat diberikan tegangan DC antara 9 – 12 Volt.

i) 6 Pin Input Analog (0 – 5)

*Pin* ini sangat berguna untuk membaca tegangan yang dihasilkan oleh sensor *analog*, seperti sensor suhu. Program dapat membaca nilai sebuah *pin input* antara 0 – 1023, dimana hal itu mewakili nilai tegangan 0 – 5V.

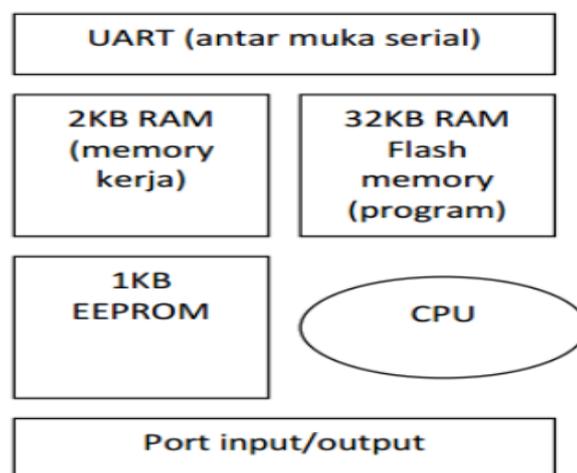
### 2.2.1.6 Aplikasi Arduino Sebagai Kontroler Motor

*Arduino Uno* berperan untuk menerima data koordinat yang dikirim oleh *Raspberry Pi* melalui sambungan kabel USB dengan komunikasi serial secara langsung. Dari data yang diterima ini digunakan *Arduino Uno* untuk menentukan operasi pergerakan dari *mobile robot*.

### 2.2.2 Mikrokontroler ATmega 328

Mikrokontroler ATmega328 digunakan pada *Arduino Uno* sebagai otak untuk mengendalikan perangkat elektronik yang akan dirancang. ATmega328 itu sendiri diproduksi oleh ATMEL yang mempunyai arsitektur RISC (*Reduced Instruction Set Computing*) dimana arsitektur RISC ini adalah suatu arsitektur yang memiliki instruksi yang sederhana namun memiliki banyak fasilitas tambahan.

Untuk memberikan gambaran apa saja yang terdapat di dalam sebuah mikrokontroler, gambar 2.7 memperlihatkan contoh diagram blok sederhana dari mikrokontroler ATmega 328 (dipakai pada *Arduino Uno*)



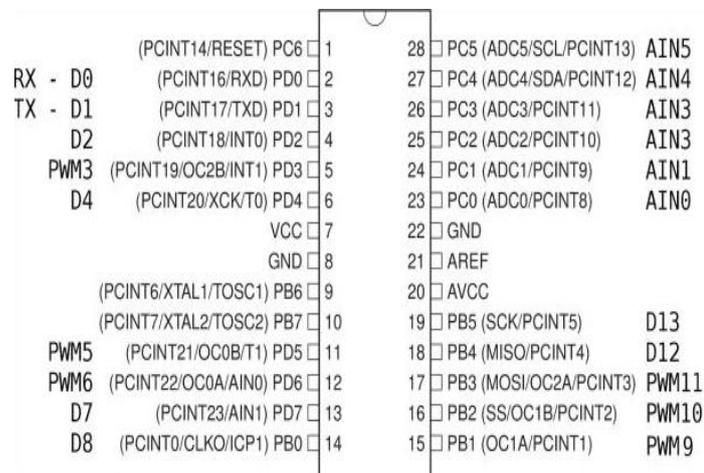
**Gambar 2.7 Diagram Sederhana Mikrokontroler ATmega 328**

(Sumber : <https://widuri.raharja.info/index.php/KP1133465645>)

Blok-blok gambar 2.7 dijelaskan sebagai berikut :

1. *Universal Asynchronous Receiver/Transmitter* (UART) adalah antarmuka yang digunakan untuk komunikasi serial seperti pada RS-232, RS-442, dan RS-485.
2. 2KB RAM pada *memory* kerja bersifat *volatile* (hilang saat daya dimatikan), digunakan oleh variabel-variabelnya di dalam program.
3. 32KB RAM *flash memory* bersifat *non-volatile*, digunakan untuk menyimpan program yang dimulai dari komputer. Selain program, *flash memory* juga menyimpan *bootloader*. *Bootloader* adalah program inisiasi yang ukurannya kecil, dijalankan oleh CPU saat daya dihidupkan. Setelah *bootloader* selesai dijalankan, berikutnya program di dalam RAM akan dieksekusi.
4. 1KB EEPROM bersifat *non-volatile*, digunakan untuk menyimpan data yang tidak boleh hilang saat daya dimatikan. Tidak digunakan pada *board Arduino*.
5. *Central Processing Unit* (CPU), bagian dari mikrokontroler untuk menjalankan setiap instruksi dari program.
6. *Port input / output, pin-pin* untuk menerima data (*input*) *digital* atau *analog*, dan mengeluarkan data (*output*) *digital* atau *analog*.

ATmega328 memiliki 3 buah PORT utama yaitu PORTB, PORTC, dan PORTD dengan total *pin input/output* sebanyak 23 *pin*. PORT tersebut dapat difungsikan sebagai *input/output digital* atau difungsikan sebagai alternatif lainnya. Gambar 2.8 berikut adalah *pin-pin* dari mikrokontroler ATmega 328.



**Gambar 2.8 Pin Mikrokontroler ATmega 328**

(Sumber : [http://www.gravitech.com/Atmega328\\_datasheet](http://www.gravitech.com/Atmega328_datasheet))

### 2.2.2.1 Port B

Port B merupakan jalur data 8 bit yang dapat difungsikan sebagai *input/output*. Selain itu PORTB juga dapat memiliki fungsi alternatif seperti di bawah ini.

- a. ICP1 (PB0), berfungsi sebagai *Timer Counter 1 input capture pin*.
- b. OC1A (PB1), OC1B (PB2) dan OC2 (PB3) dapat difungsikan sebagai keluaran *PWM (Pulse Width Modulation)*.
- c. MOSI (PB3), MISO (PB4), SCK (PB5), SS (PB2) merupakan jalur komunikasi SPI.
- d. Selain itu pin ini juga berfungsi sebagai jalur pemrograman serial (ISP).
- e. TOSC1 (PB6) dan TOSC2 (PB7) dapat difungsikan sebagai sumber *clock external* untuk *timer*.
- f. XTAL1 (PB6) dan XTAL2 (PB7) merupakan sumber *clock* utama mikrokontroler.

### 2.2.2.2 Port C

Port C merupakan jalur data 7 bit yang dapat difungsikan sebagai *input/output digital*. Fungsi alternatif PORTC antara lain sebagai berikut.

- a. ADC 6 *channel* (PC0, PC1, PC2, PC3, PC4, PC5) dengan resolusi sebesar 10 bit. ADC dapat kita gunakan untuk mengubah *input* yang berupa tegangan *analog* menjadi data *digital*.
- b. I<sup>2</sup>C (SDA dan SDL) merupakan salah satu fitur yang terdapat pada PORTC. I<sup>2</sup>C digunakan untuk komunikasi dengan sensor atau *device* lain yang memiliki komunikasi data tipe I<sup>2</sup>C seperti sensor kompas, *accelerometer nunchuck*.

### 2.2.2.3 Port D

Port D merupakan jalur data 8 bit yang masing-masing *pin*-nya juga dapat difungsikan sebagai *input/output*. Sama seperti Port B dan Port C, Port D juga memiliki fungsi alternatif dibawah ini.

- a. USART (TXD dan RXD) merupakan jalur data komunikasi serial dengan level sinyal TTL. Pin TXD berfungsi untuk mengirimkan data serial,

sedangkan RXD kebalikannya yaitu sebagai *pin* yang berfungsi untuk menerima data serial.

- b. *Interrupt* (INT0 dan INT1) merupakan pin dengan fungsi khusus sebagai interupsi *hardware*. Interupsi biasanya digunakan sebagai selaan dari program, misalkan pada saat program berjalan kemudian terjadi interupsi *hardware/software* maka program utama akan berhenti dan akan menjalankan program interupsi.
- c. XCK dapat difungsikan sebagai sumber *clock external* untuk USART, namun kita juga dapat memanfaatkan *clock* dari CPU, sehingga tidak perlu membutuhkan *external clock*.
- d. T0 dan T1 berfungsi sebagai masukan *counter external* untuk *timer* 1 dan *timer* 0.
- e. AIN0 dan AIN1 keduanya merupakan masukan *input* untuk *analog comparator*.

### 2.2.3 *Raspberry Pi*

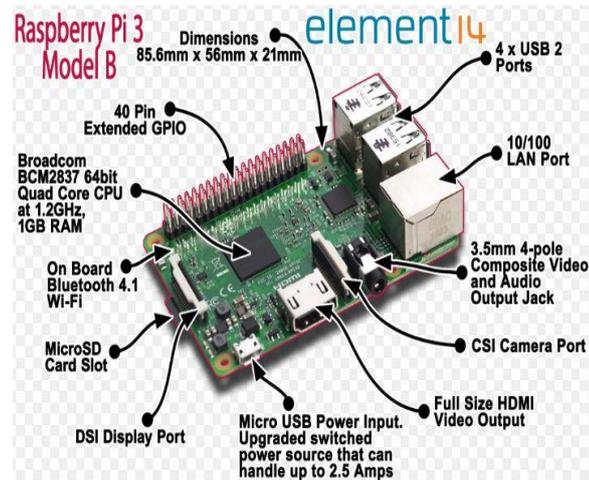
*Raspberry Pi* atau *Raspi* adalah komputer kecil seukuran sebuah kartu kredit, *Raspberry Pi* memiliki *processor*, RAM dan *port hardware* yang khas yang bisa anda temukan pada banyak komputer. Ini berarti, anda dapat melakukan banyak hal seperti pada sebuah komputer *desktop*. Anda dapat melakukan seperti mengedit dokumen, memutar video HD, bermain *game*, *coding*, dan banyak lagi.

*Raspberry Pi* adalah sebuah SBC (*Single Board Computer*) seukuran kartu kredit yang dikembangkan oleh Yayasan *Raspberry Pi* di Inggris (UK) dengan maksud untuk memicu pengajaran ilmu komputer dasar di sekolah-sekolah. *Raspberry Pi* tidak akan memiliki kekuatan seperti *desktop* PC, tapi harganya jauh lebih murah.

Sistem operasi utama untuk *Pi* adalah *Raspbian OS* dan didasarkan dari *Debian* (*based on debian*). Ini adalah distribusi *Linux* sehingga anda mungkin akan merasa sedikit berbeda jika anda sering menggunakan komputer *Windows*. *Raspberry Pi* menggunakan *system on chip* (SoC) dari Broadcom BCM2835, juga sudah termasuk *processor* ARM1176JZF-S 700 MHz, GPU VideoCore IV dan RAM sebesar 256 (untuk Rev.B), tidak menggunakan *harddisk* namun

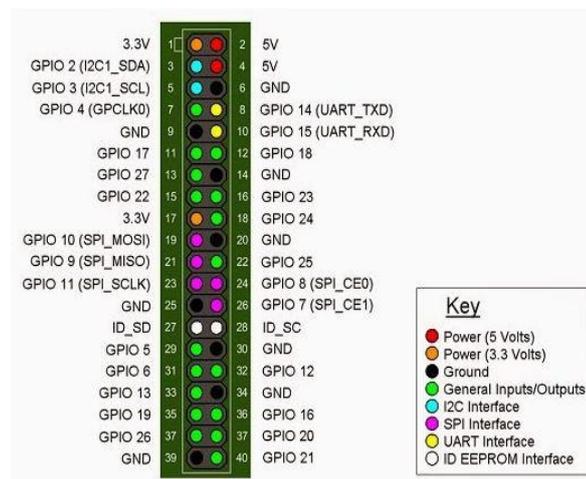
menggunakan *SD Card* untuk proses *booting* dan penyimpanan data jangka panjang.

### 2.2.3.1 Bagian-Bagian *Board Raspberry Pi*



**Gambar 2.9** *Board Raspberry Pi*

(Sumber : <https://www.element14.com/raspberry-pi-3-specifications>)



**Gambar 2.10** *Konfigurasi Pin GPIO Raspberry Pi*

(Sumber : [https://widuri.raharja.info/index.php/Pengguna:Hendri\\_mei\\_subagya](https://widuri.raharja.info/index.php/Pengguna:Hendri_mei_subagya))

## 2.3 Sistem Penggerak (*Actuator*)

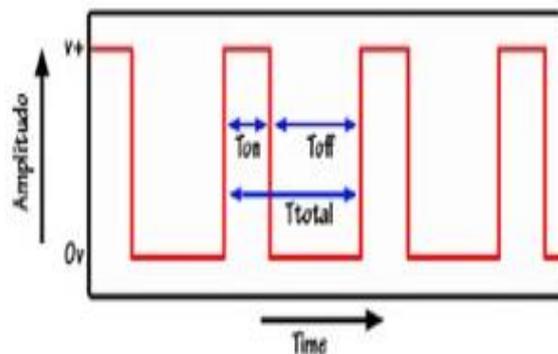
Penggerak adalah sebuah peralatan mekanis untuk menggerakkan atau mengontrol sebuah mekanisme atau sistem. Penggerak diaktifkan dengan menggunakan lengan mekanis yang biasanya digerakkan oleh motor listrik, yang

dikendalikan oleh media pengontrol otomatis yang terprogram diantaranya mikrokontroler. Penggerak adalah elemen mengkonversikan besaran listrik analog menjadi besaran lainnya misalnya kecepatan putaran dan merupakan perangkat elektromagnetik yang menghasilkan daya gerakan sehingga dapat menghasilkan gerakan pada robot.

Untuk meningkatkan tenaga mekanik penggerak ini dapat dipasang sistem *gearbox*. Penggerak dapat melakukan hal tertentu setelah mendapat perintah dari kontroler, misalnya pada suatu robot pengikut bola, jika terdapat target bola, maka sensor akan memberikan informasi pada kontroler yang kemudian akan memerintah pada penggerak untuk mendekati arah target.

### 2.3.1 *Pulse Width Modulation (PWM)*

*Pulse Width Modulation (PWM)* secara umum adalah sebuah cara memanipulasi lebar sinyal yang dinyatakan dengan pulsa dalam satu periode, untuk mendapatkan tegangan rata-rata yang berbeda. Beberapa contoh aplikasi PWM adalah pemodulasian data untuk telekomunikasi, pengontrolan daya atau tegangan yang masuk ke beban, regulator tegangan, *audio effect* dan penguatan, serta aplikasi-aplikasi lainnya. Gambar 2.11 berikut adalah sinyal *Pulse Width Modulation*.



**Gambar 2.11 *Pulse Width Modulation***

(Sumber : <https://depokinstruments.com/pwm-pulse-width-modulation/>)

Aplikasi PWM berbasis mikrokontroler biasanya berupa pengendalian kecepatan motor DC, pengendalian motor servo, dan pengaturan nyala terang LED. Oleh karena itu diperlukan pemahaman terhadap konsep PWM itu sendiri.

Sinyal PWM pada umumnya memiliki *amplitude* dan frekuensi dasar yang tetap, namun memiliki lebar pulsa yang bervariasi. Lebar pulsa PWM berbanding lurus dengan *amplitude* sinyal asli yang belum termodulasi. Artinya, sinyal PWM memiliki frekuensi gelombang yang tetap namun *duty cycle* bervariasi antara 0% hingga 100%. Untuk menghitung nilai *duty cycle*, tegangan keluaran dan waktu total tersebut menggunakan persamaan-persamaan berikut ini.

$$T_{total} = T_{on} + T_{off} \dots\dots\dots (2.2)$$

$$D = \frac{T_{on}}{T_{total}} \times 100\% \dots\dots\dots (2.3)$$

$$PWM = PWM_{max} \times D \dots\dots\dots (2.4)$$

$$V_{out} = \frac{T_{on}}{T_{total}} \times V_{in} \dots\dots\dots (2.5)$$

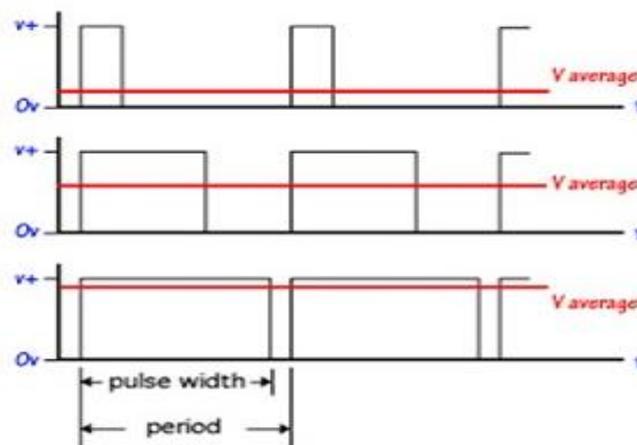
Keterangan :

$T_{on}$  = Waktu pulsa “High”.

$T_{off}$  = Waktu pulsa ‘Low”.

$D$  = *Duty cycle* adalah lamanya pulsa *High* dalam satu perioda.

Dari persamaan diatas, diketahui bahwa perubahan *duty cycle* akan merubah tegangan *output* atau tegangan rata-rata seperti gambar 2.12 dibawah ini.

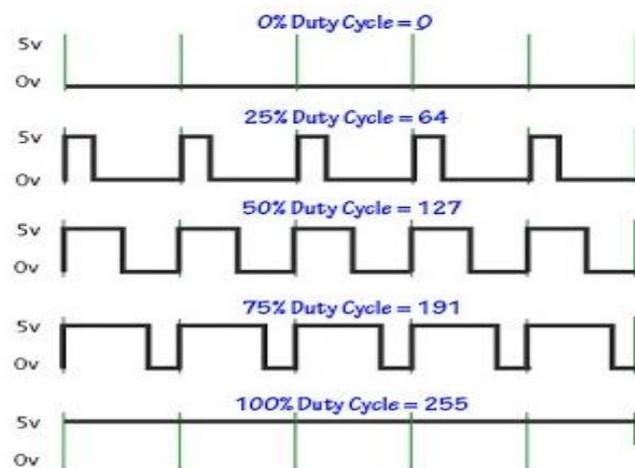


**Gambar 2.12 Tegangan Rata-Rata Perubahan *Duty Cycle***

(Sumber : <https://depokinstruments.com/pwm-pulse-width-modulation/>)

PWM merupakan salah satu teknik untuk mendapatkan sinyal *analog* dari sebuah piranti *digital*. Sebenarnya sinyal PWM dapat dibangkitkan dengan

banyak cara, secara *analog* menggunakan IC *Op-Amp* atau secara *digital*. Secara *analog* setiap perubahan PWM-nya sangat halus, sedangkan secara *digital* setiap perubahan PWM dipengaruhi oleh resolusi PWM itu sendiri. Resolusi adalah jumlah variasi perubahan nilai dalam PWM tersebut. Misalkan suatu PWM memiliki resolusi 8 bit, berarti PWM ini memiliki variasi perubahan nilai sebanyak 256 variasi mulai dari 0 – 255 perubahan nilai yang mewakili *duty cycle* 0 % – 100 % dari keluaran PWM tersebut. Gambar 2.13 berikut adalah sinyal *duty cycle* dan tabel 2.1 adalah linearitas PWM tersebut.



**Gambar 2.13 Sinyal *Duty Cycle***

(Sumber : <https://depokinstruments.com/pwm-pulse-width-modulation/>)

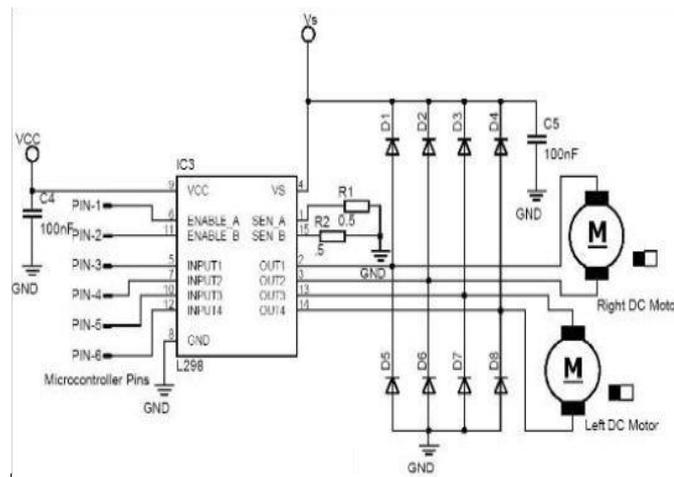
**Tabel 2.1 Linearitas PWM**

Nilai <i>Analog Arduino</i>	<i>Duty Cycle</i>	<i>VOut Driver Motor</i>
0	0 %	0 Volt
64	25 %	1,25 Volt
127	50 %	2,5 Volt
191	75 %	3,75 Volt
255	100 %	5 Volt

### 2.3.2 *Driver Motor DC L298N*

IC *H-Bridge driver* motor DC L298N memiliki dua buah rangkaian *H-Bridge* didalamnya, sehingga dapat digunakan untuk men-*drive* dua buah motor

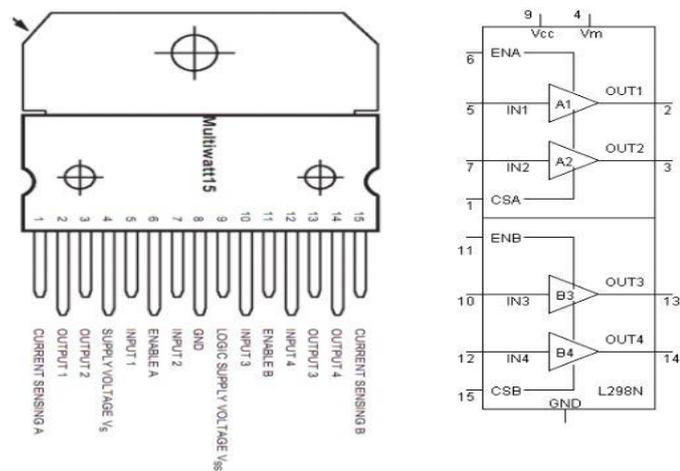
DC. *H-Bridge driver* motor DC L298N masing-masing dapat mengantarkan arus hingga 1A. Namun dalam penggunaannya, *H-Bridge driver* motor DC L298N dapat digunakan secara paralel, sehingga kemampuan menghantarkan dari *H-Bridge driver* motor DC L298N arusnya menjadi 2A. Gambar 2.14 berikut adalah rangkaian *driver* motor L298N.



**Gambar 2.14 Rangkaian *Driver* Motor DC L298N**

(Sumber : <http://www.robot-id.com/merancang-driver-motor-robot-line.html>)

Prinsip kerja IC L298N ini memiliki empat *channel* masukan yang didesain untuk dapat menerima masukan logika TTL. Masing-masing *channel* masukan ini memiliki *channel* keluaran yang bersesuaian. Gambar 2.15 memperlihatkan penampang IC L298N, dengan memberi tegangan 5 volt pada pin *enable A* dan *enable B*, masing-masing *channel output* akan menghasilkan logika *high* (1) atau *low* (0) sesuai dengan *input* pada *channel* masukan. Untuk lebih jelasnya prinsip kerja IC L298N dapat dilihat pada tabel 2.1.



**Gambar 2.15** Penampang IC L298N

(Sumber : <http://www.bristolwatch.com/L298N/>)

**Tabel 2.2** Logika Prinsip Kerja IC L298N

Input M1		Input M2		Enable		M1	M2	Ket.
IN1	IN2	IN3	IN4	ENA	ENB			
0	0	0	0	1	1	Off	Off	
1	0	1	0	1	1	On (CW)	On (CW)	
0	1	0	1	1	1	On (CCW)	On (CCW)	
1	0	1	0	0	0	Off	Off	
0	1	0	1	0	0	Off	Off	
1	1	1	1	1	1	-	-	Hindari

(Sumber : <http://www.robotics-university.com/driver-motor-dcmp-ic-l298.html>)

### 2.3.3 Smart RC Motor DC 6 Volt

Motor DC ini memiliki prinsip kerja yang sama seperti motor DC pada umumnya, yang membedakan adalah sumbu putarnya langsung terhubung dengan sebuah *gearbox* khusus yang dapat memutar roda posisi pemasangannya bukan langsung disumbu motor DC tapi disamping motor DC tersebut. Motor DC ini dapat digerakkan dengan tegangan minimum sekitar 3 Volt hingga maksimal 6 Volt. Motor DC ini didesain khusus untuk *prototyping robot mobile* dan biasanya

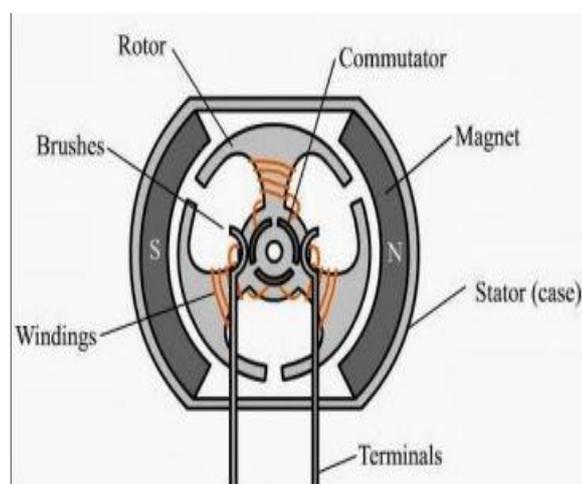
dijual berpasangan dengan sasisnya. Gambar 2.16 berikut adalah gambar fisik motor DC tersebut.



**Gambar 2.16 Smart RC Motor DC dengan Gearbox**

(Sumber : <http://www.ebay.com/itm/Gear-Motor-DC-3-12V-Robot-Tire-Wheel-/>)

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 2.17. Polaritas dari tegangan yang diberikan pada dua terminal menentukan arah putaran motor sedangkan besar dari beda tegangan pada kedua terminal menentukan kecepatan motor.



**Gambar 2.17 Konstruksi Motor DC**

(Sumber : <http://wahyu-umiq.wordpress.co.id/2013/10/motor-dc.html>)

Konstruksi motor DC pada gambar 2.17 memiliki 2 bagian dasar, yaitu :

1. Bagian tetap atau stasioner yang disebut *stator*. *Stator* ini menghasilkan medan magnet, baik yang dibangkitkan dari sebuah koil (*electromagnetic*) ataupun magnet permanen.
2. Bagian yang berputar disebut *rotor*. *Rotor* ini berupa sebuah koil dimana arus listrik mengalir.

Gaya *electromagnetic* 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 di antara dua kutub magnet dari kutub utara ke kutub selatan.