

## BAB II

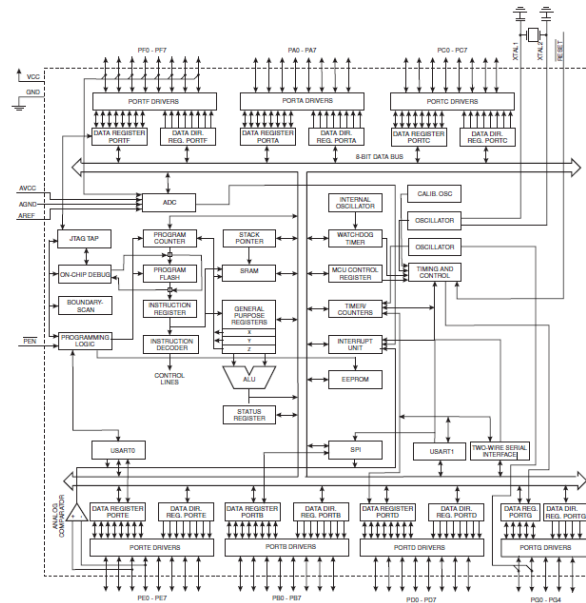
### LANDASAN TEORI

#### 2.1. Mikrokontroler ATmega 128

Mikrokontroler ATmega 128 merupakan mikrokontroler keluarga AVR yang mempunyai kapasitas flash memori 128KB. AVR (*Alf and Vegard's Risc Processor*) merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Secara umum, AVR dapat terbagi menjadi 4 kelas, yaitu keluarga ATtiny, keluarga AT90Sxx, keluarga AT-Mega, dan AT86RFxx. Pada dasarnya yang membedakan masing-masing kelas adalah memori, *peripheral*, dan fungsinya. Dari segi arsitektur dan instruksi yang digunakan, bisa dikatakan hampir sama. Semua jenis AVR dilengkapi dengan *flash* memori sebagai memori program. Kapasitas dari *flash* memori ini berbeda antara *chip* yang satu dengan *chip* yang lain, tergantung dari jenis IC yang digunakan. Untuk *flash* memori yang paling kecil adalah 1 kbytes (ATtiny11, ATtiny12, dan ATtiny15) dan paling besar adalah 128 kbytes (AT-Mega128).

##### 2.1.1. Blog Diagram ATmega128

Blog diagram mikrokontroler ATmega128 dapat dilihat pada gambar 2.1:



Gambar 2.1 Blog diagram ATmega128. (Atmel Corporation, 2007)

Core AVR menggabungkan banyak instruksi dengan 32 general register. 32 register secara langsung terhubung ke Arithmetic Logic Unit (ALU), yang memungkinkan dua register independen untuk diakses dalam satu instruksi tunggal pada satu siklus. Arsitektur yang dihasilkan lebih efisien sementara mencapai *throughput* hingga sepuluh kali lebih cepat daripada mikrokontroler CISC konvensional. ATmega128 menyediakan fitur berikut : byte 128K *In-System Programmable Flash* dengan kemampuan *READ - WHILE - Write*, 4K byte EEPROM, 4K byte SRAM, 53 general purpose I/O line, 32 *r general purpose working registers*, *Real Time Counter* (RTC), empat fleksibel Timer/Counter, PWM, 2 USARTs, 10-bit ADC, pemrograman Watchdog Timer dengan osilator internal, sebuah port serial SPI dan pemrograman dan perangkat lunak enam mode yang hemat energy.

Sementara itu SRAM, *Timer / Counter*, port SPI, memberikan instruksi kepada sistem untuk terus berfungsi. Mode Power-down menyimpan register tapi

membekukan *oscillator*, menonaktifkan semua fungsi chip yang lain sampai interupsi berikutnya atau me-reset Hardware. Dalam mode *power save*, *timer asynchronous* terus berjalan, yang memungkinkan pengguna untuk mempertahankan time base sedangkan perangkat sedang dalam kondisi sleep. ADC modus noise reduction memberhentikan CPU dan semua modul I/O kecuali Timer Asynchronous dan ADC, untuk meminimalkan switching noise selama ADC konversi. Dalam modus siaga, crystal/resonator oscillator berjalan sebagian perangkat dalam keadaan sleep. Hal ini memungkinkan sangat cepat nya start-up dengan konsumsi daya yang rendah. Dalam *extended standby mode*, baik *oscillator* utama dan *timer asynchronous* terus berjalan . Perangkat ini diproduksi menggunakan teknologi memori *nonvolatile high-density*.

Onchip ISP Flash memungkinkan memori program untuk diprogram melalui interface serial SPI, oleh memori nonvolatile konvensional, atau dengan program boot on-chip berjalan pada core AVR. Program boot dapat menggunakan interface apapun untuk mengunduh program aplikasi dalam memori flash. Software di bagian flash boot akan terus berjalan sedangkan bagian flash aplikasi diperbarui, memberikan operasi read-while -write. Dengan menggabungkan 8-bit RISC CPU dan In-System Self- Programmable Flash pada chip monolitik, ATmega128 adalah mikrokontroler kuat yang memberikan solusi yang efektif, dan sangat fleksibel

### **2.1.2. Fitur ATmega128**

Beberapa fitur yang dimiliki ATmega128 adalah sebagai berikut:

- a. Mempunyai kinerja tinggi dengan konsumsi daya yang rendah

- b. Arsitektur RISC yang telah maju
  - 1. 130 instruksi kuat – *Most Single Clock Cycle Execution*
  - 2. 32 x 8 Register kerja multifungsi + *peripheral control registers*
  - 3. Operasi statis penuh
  - 4. *Throughput* hingga 16 MIPS pada 16 MHz
  - 5. *Multiplier 2-cycle on-chip*
- c. Daya tahan tinggi memori *non volatile*
  - 1. 128K bytes *In-System Self-programmable Flash program memory*
  - 2. 4K bytes EEPROM
  - 3. 100,000 EEPROM dengan kemampuan 10.000 *wirte/erase cycle*
  - 4. 4K bytes RAM internal
  - 5. Memori eksternal hingga 64K Bytes
  - 6. Penguncian program untuk keamanan sistem
  - 7. *SPI Interface* untuk *In-System Programming*
- d. JTAG (IEEE std. 1149.1 *Compliant*) *Interface*
  - 1. Kemampuan batas scan mengikuti standar JTAG
  - 2. *Extensive On-chip Debug Support*
  - 3. Pemrograman *Flash*, EEPROM, *Fuses* dan *Lock Bits* melalui antarmuka JTAG
- e. I/O dan paket
  - 1. 53 *programmable I/O lines*
  - 2. 64-lead TQFP dan 64-pad QFN/MLF
- f. Tingkat kecepatan mulai dari 0 hingga 16MHz
- g. Bekerja pada tegangan 4,5 – 5,5 Volt

h. Fitur spesial mikrokontroler

1. *Power-on reset* dan deteksi *programmable brown-out*
2. Osilator RC kalibrasi internal
3. *Interrupt source* external dan internal
4. Enam mode Sleep: *Idle*, *ADC noise reduction*, *Power-save*, *Power-down*, *Stand-by*, dan *Extended Stand-by*
5. *Software Selectable Clock Frequency*
6. ATmega103 dapat berkerja sesuai mode yang dipilih pada *fuses*
7. *Global Pull-up Disable*

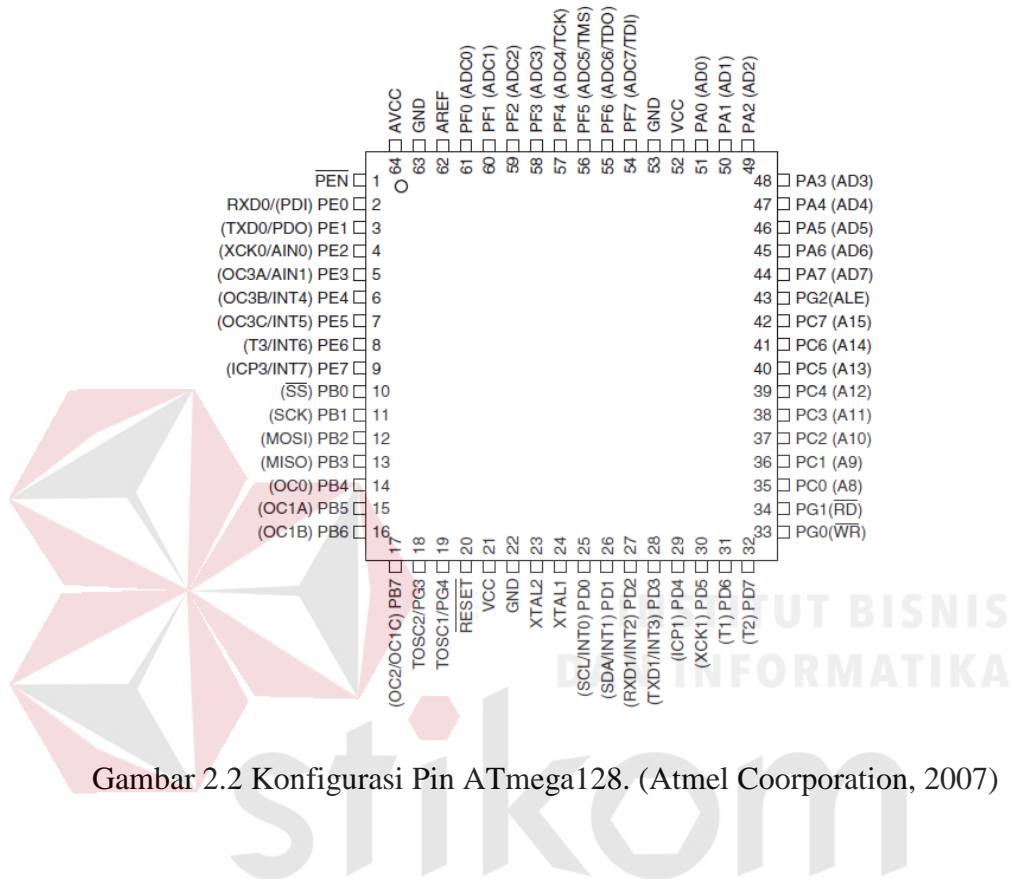
i. Fitur *pheripheral*

1. Dua *timer/counter* 8-bit dengan *separate prescalers* dan *compare modes*
2. Dua *timer/counter* 16-bit dengan *separate prescalers*, *compare modes*, dan *capture modes*
3. *Counter real time* dengan osilator terpisah
4. Dua channels 8-bit PWM
5. 8 *channel*, 10-bit ADC
6. *Byte-oriented two-wire serial interface*
7. *Dual Programmable* serial USART
8. *Master/slave* SPI serial *interface*
9. *Programmable Watchdog Timer* dengan Osilator *On-chip*
10. *On-chip analog comparator*

(Atmel Cooperation, 2007)

### 2.1.3. Konfigurasi Pin ATmega128

Konfigurasi pin pada mikrokontroler ATmega128 dengan 64 pin dapat dilihat pada gambar 2.2:



Gambar 2.2 Konfigurasi Pin ATmega128. (Atmel Cooperation, 2007)

Dari gambar di atas dapat dijelaskan fungsi masing-masing pin pada mikrokontroler ATmega128 sebagai berikut:

1. VCC merupakan pin yang berfungsi sebagai masukan catu daya
2. GND merupakan pin *ground*
3. Port A (PortA0...PortA7) merupakan pin *input/output* dua arah
4. Port B (PortB0...PortB7) merupakan pin *input/output* dua arah dan pin fungsi khusus, seperti dapat dilihat pada tabel 2.3.

Tabel 2.1 Fungsi Khusus Port B

<b>Pin</b>	<b>Fungsi Khusus</b>
PB7	OC2 ( <i>Timer/Counter Output Compare Match Output</i> ) OC1C ( <i>Timer/Counter1 Output Compare C Match Output</i> )
PB6	OC1B ( <i>Timer/Counter1 Output Compare B Match Output</i> )
PB5	OC1A ( <i>Timer/Counter1 Output Compare A Match Output</i> )
PB4	OC0 ( <i>Timer/Counter0 Output Compare Match Output</i> )
PB3	MISO ( <i>SPI Bus Master Input/ Slave Output</i> )
PB2	MOSI ( <i>SPI Bus Master Output/ Slave Input</i> )
PB1	SCK ( <i>SPI Bus Serial Clock</i> )
PB0	SS ( <i>SPI Slave Select Input</i> )

5. Port C (PortC0...PortC7) merupakan pin *input/output* dua arah
6. Port D (PortD0...PortD7) merupakan pin *input/output* dua arah dan pin fungsi khusus, seperti dapat dilihat pada tabel 2.2.

Tabel 2.2 Fungsi Khusus Port D

<b>Pin</b>	<b>Fungsi Khusus</b>
PD7	T2 ( <i>Timer/ Counter2 External Counter Input</i> )
PD6	T1 ( <i>Timer/ Counter1 External Counter Input</i> )
PD5	XCK1 ( <i>USART External Clock Input/Output</i> )
PD4	ICP1 ( <i>Timer/Counter1 Input Capture Pin</i> )
PD3	TXD1 ( <i>USART1 Output Pin</i> )

	INT3 ( <i>External Interrupt 3 Input</i> )
PD2	RXD1 ( <i>USART1 Input Pin</i> ) INT0 ( <i>External Interrupt 0 Input</i> )
PD1	SDA ( <i>Two-wire Serial Bus Data Input/Output Line</i> )
PD0	SCL ( <i>Two-wire Serial Bus Clock Line</i> )

7. Port E (PortE0...PortE7) merupakan pin *input/output* dua arah dan pin fungsi khusus, seperti yang terlihat pada tabel 2.3.

Tabel 2.3 Fungsi Khusus Port E

Pin	Fungsi Khusus
PE7	ICP3 ( <i>Timer/Counter3 Input Capture Pin</i> ) INT7 ( <i>External Interrupt 7 Input</i> )
PE6	T3 ( <i>Timer/ Counter3 External Counter Input</i> ) INT6 ( <i>External Interrupt 6 Input</i> )
PE5	OC3C ( <i>Timer/Counter3 Output Compare C Match Output</i> ) INT5 ( <i>External Interrupt 5 Input</i> )
PE4	OC3B ( <i>Timer/Counter3 Output Compare B Match Output</i> ) INT4 ( <i>External Interrupt 4 Input</i> )
PE3	OC3A ( <i>Timer/Counter3 Output Compare A Match Output</i> ) AIN1 ( <i>Analog Comparator Negative Input</i> )
PE2	XCK0 ( <i>USART0 External Clock Input/Output</i> ) AIN0 ( <i>Analog Comparator Positive Input</i> )



PE1	TXD0(USART0 <i>Output Pin</i> )
PE0	RXD0 (USART0 <i>Input Pin</i> )

8. Port F (PortF0...PortF7) merupakan pin *input/output* dua arah dan pin masukan ADC serta pin khusus seperti yang terlihat pada tabel 2.4.

Tabel 2.4 Fungsi Khusus Port F

Pin	Fungsi Khusus
PF7	ADC ( <i>Analog to Digital Converter</i> ) TDI (JTAG <i>Test Data Out</i> )
PF6	ADC ( <i>Analog to Digital Converter</i> ) TDO (JTAG <i>Test Data In</i> )
PF5	ADC ( <i>Analog to Digital Converter</i> ) TMS (JTAG <i>Test Mode Select</i> )
PF4	ADC ( <i>Analog to Digital Converter</i> ) TCK (JTAG <i>Test Clock</i> )
PF3	ADC ( <i>Analog to Digital Converter</i> )
PF2	ADC ( <i>Analog to Digital Converter</i> )
PF1	ADC ( <i>Analog to Digital Converter</i> )
PF0	ADC ( <i>Analog to Digital Converter</i> )

9. Port G (PortG0...PortG4) merupakan pin *input/output* dua arah dan pin fungsi khusus, seperti yang terlihat pada tabel 2.5.

Tabel 2.5 Fungsi Khusus Port G

Pin	Fungsi Khusus
PG4	TOSC1 ( <i>Timer Oscillator Pin1</i> )
PG3	TOSC2 ( <i>Timer Oscillator Pin2</i> )
PG2	ALE ( <i>Adreess Latch Enable</i> )
PG1	RD ( <i>External Data Memory Read Strobe</i> )
PG0	WR ( <i>External Data Memory Write Strobe</i> )

10. RESET merupakan pin yang digunakan untuk me-reset mikrokontroler

11. PEN berfungsi mengaktifkan pin untuk mode Pemrograman Serial SPI

12. XTAL1 dan XTAL2 merupakan pin masukan clock eksternal

13. AVCC merupakan pin masukan tegangan untuk ADC

14. AREFF merupakan pin masukan tegangan referensi ADC

## 2.2. DT-AVR ATMEGA1280 CPU MODULE

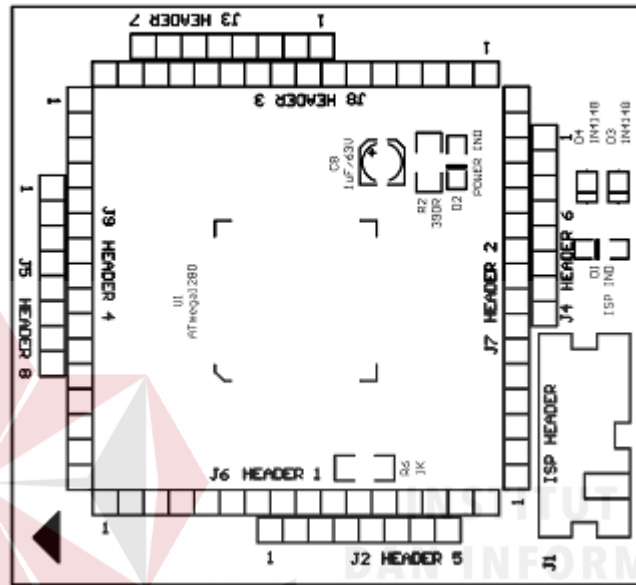
DT-AVR ATMEGA1280 CPU MODULE merupakan modul mikrokontroler berbasis mikrokontroller AVR ATmega1280 yang memiliki kemampuan dan konektor untuk *in-System Programming* (ISP).

### Spesifikasi

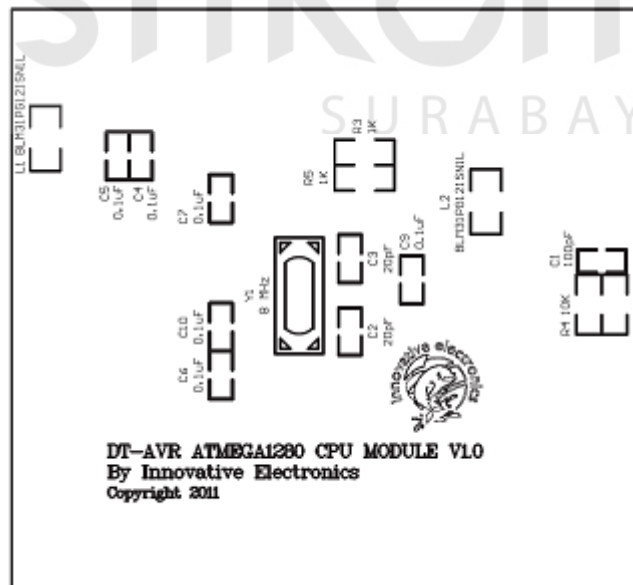
1. Berbasis ATmega 1280 (128 KB Flash Memory, 4 KB EEPROM, dan 8 KB Internal SRAM).
2. Memiliki jalur *input/output* hingga 86 pin, termasuk 2 *timer/counter* 8 bit, 12 kanal PWM 2-16 bit, 16 kanal ADC 10 bit, 4 serial USART, *watchdog timer*, dan *analog comparator*.
3. Tersedia *crystal oscillator* berfrekuensi 8 MHz.

4. Sebuah port untuk pemrograman secara ISP.
5. LED indicator pemrograman.
6. Catu daya (VCC) 2,7 – 5,5 Volt DC.
7. Kompatibel dengan DT-COMBO BASE BOARD series.

#### Tata Letak Komponen

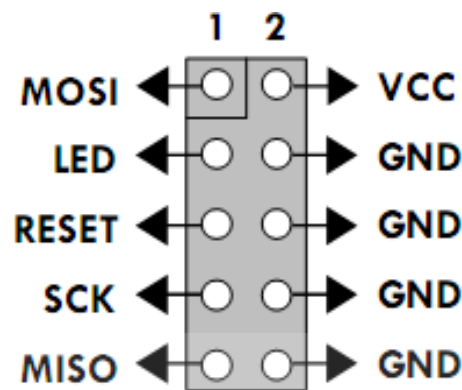


Gambar 2.3 Tata letak komponen minimum sistem



Gambar 2.4 Tata letak komponen minimum sistem 2

Konektor ISP *HEADER* (J1) adalah konektor untuk pemrograman secara ISP.

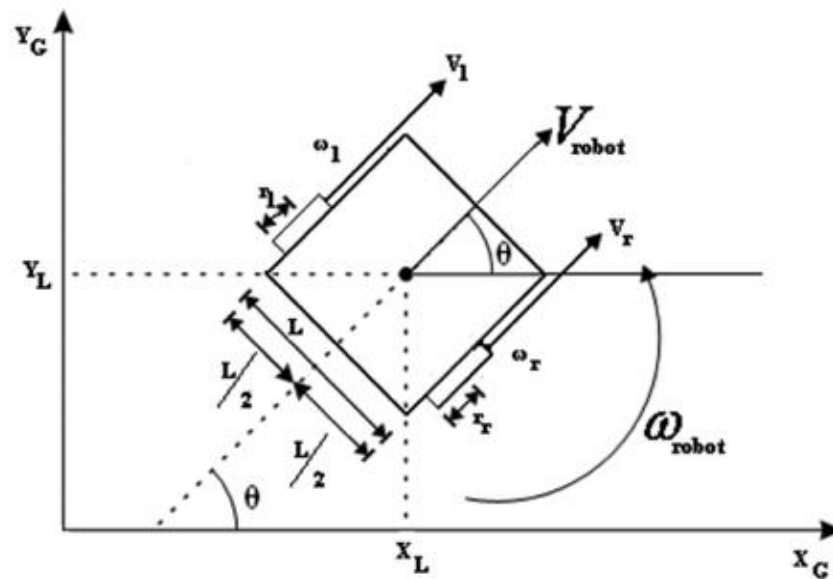


Gambar 2.5 Pin pada konektor ISP *HEADER* (J1)

### 2.3. Penggerak Differensial Robot

Salah satu jenis *mobile robot* yang umum digunakan, terutama untuk dioperasikan dalam ruangan adalah dengan pengemudian atau sistem penggerak diferensial (*differential drive*). Alasan utamanya karena relative dan lebih fleksibel dalam melakukan manuver serta kemudahan dalam pengontrolannya.

Arsitektur dari *differential drive* dapat dilihat pada gambar 2.6.



Gambar 2.6 Posisi dan orientasi *mobile robot* dalam sistem koordinat cartesian. Kecepatan linier *mobile robot* pada masing-masing roda kanan dan kiri berturut-turut adalah  $V_R$  dan  $V_L$ . Kecepatan rotasi masing-masing roda dengan jari-jari  $r$  adalah  $\omega_R$  dan  $\omega_L$  sesuai dengan persamaan 2.1 dan 2.2 berikut:

$$2. \quad \omega_R(t) = \frac{V_R(t)}{r} \dots\dots\dots (2.1)$$

$$3. \quad \omega_L(t) = \frac{V_L(t)}{r} \dots\dots\dots (2.2)$$

Ketika robot melakukan gerakan memutar (berotasi) sesaat dengan panjang jari-jari  $R$  diukur dari pusat rotasi dan titik pusat kedua titik maka kecepatan rotasi disetiap titik robot tersebut selali sama (robot adalah sistem mekanis yang rigid), sehingga persamaan 2.3 dan 2.4 berikut ini berlaku untuk menghitung kecepatan rotasi dari robot tersebut:

$$4. \quad \omega(t) = \frac{V_R}{R+L/2} \dots\dots\dots (2.3)$$

$$5. \omega(t) = \frac{V_L}{R-L/2} \dots\dots\dots (2.4)$$

Berdasarkan persamaan 2.3 dan 2.4 kecepatan rotasi robot tersebut dapat dihitung hanya berdasarkan informasi dari kedua kecepatan linier roda robot tersebut:

$$6. \omega(t) = \frac{V_R(t)-V_L(t)}{L} \dots\dots\dots (2.5)$$

Sedangkan jari-jari dapat dicari dengan:

$$7. R = \frac{L(V_R+V_L)}{2(V_R-V_L)} \dots\dots\dots (2.6)$$

Dari persamaan 2.6, jari-jari lintasan lingkaran sesaat berbanding terbalik dengan selisih kedua kecepatan roda robot. Semakin kecil selisih kedua kecepatan roda maka jari-jari lingkaran sesaat yang dibentuk oleh lintasan robot tersebut semakin panjang dan sebaliknya. Sedangkan jika kecepatan linier roda kiri maka  $R = \infty$ , atau secara praktis robot akan bergerak membentuk lintasan yang lurus. Agar robot dapat berotasi pada pusat sumbunya ( $R=0$ ) maka berdasarkan persamaan 2.6, kecepatan kedua roda tersebut harus berlawanan.

Berdasarkan persamaan 2.5 dan 2.6, maka kecepatan linier robot dapat dihitung dengan menggunakan persamaan 2.7 berikut:

$$8. R = \frac{V_R(t)+V_L(t)}{2} \dots\dots\dots (2.7)$$

Agar lebih sederhana, persamaan 2.5 dan 2.7 dapat dikumpulkan dalam bentuk persamaan matrik vektor sebagaimana berikut:

$$9. \begin{bmatrix} V(t) \\ \omega(t) \end{bmatrix} = \begin{bmatrix} 1/2 & 1/2 \\ 1/L & -1/L \end{bmatrix} \begin{bmatrix} V_R(t) \\ V_L(t) \end{bmatrix} \dots\dots\dots (2.8)$$

Persamaan 2.8 pada dasarnya memperlihatkan relasi antara kecepatan linier roda-roda robot terhadap kecepatan linier dan angular robot, sedangkan persamaan 2.9 berikut akan memperlihatkan relasi sebaliknya.

$$10. \begin{bmatrix} V_R(t) \\ V_L(t) \end{bmatrix} = \begin{bmatrix} 1/2 & 1/2 \\ 1/L & -1/L \end{bmatrix} \begin{bmatrix} V(t) \\ \omega(t) \end{bmatrix} \dots\dots\dots (2.9)$$

Dengan mengetahui kecepatan linier dan angular robot setiap saat, maka kecepatan pada setiap sumbu kartesian dapat dicari dengan cara memproyeksikan vektor kecepatan robot pada sumbu-sumbu tersebut. (Utomo, 2007).

*Mobile robot* tipe penggerak diferensial memiliki 2 buah roda penggerak yang terpisah (kanan dan kiri). Kedua roda ini digerakkan oleh motor DC yang ditempatkan pada satu sumbu secara terpisah. Sehingga kedua roda ini berfungsi sebagai penggerak sekaligus sebagai kemudi *mobile robot*. Sehingga tingkat keluwesan robot dan kemampuan manuver *mobile robot* tipe penggerak diferensial jauh lebih baik.

Jika kedua roda didorong dalam arah yang sama dan kecepatan sama pula, robot akan bergerak dalam garis lurus. Jika tidak, tergantung pada kecepatan rotasi dan arahnya, pusat rotasi bisa jatuh di mana saja di garis yang menghubungkan dua roda. Karena arah robot tergantung pada kecepatan dan arah putaran dari dua roda yang digerakkan. Jika kedua roda berputar dengan kecepatan yang sama dalam arah yang berlawanan, robot akan berputar di titik pusat sumbu. (Hartanti, 2011).

#### 2.4. Dagu Rover 5 4WD

Rover 5 adalah generasi baru *chassis* robot yang dirancang khusus untuk kepentingan penelitian dan hobi. Tidak seperti *chassis* yang konvensional, sudut kaki-kaki dapat disesuaikan dengan memutar gearbox secara bertahap yang mempunyai 5 tingkatan. Elastisitas pada tapak karet mempertahankan ketegangan pada saat sudut kaki dinaikkan.

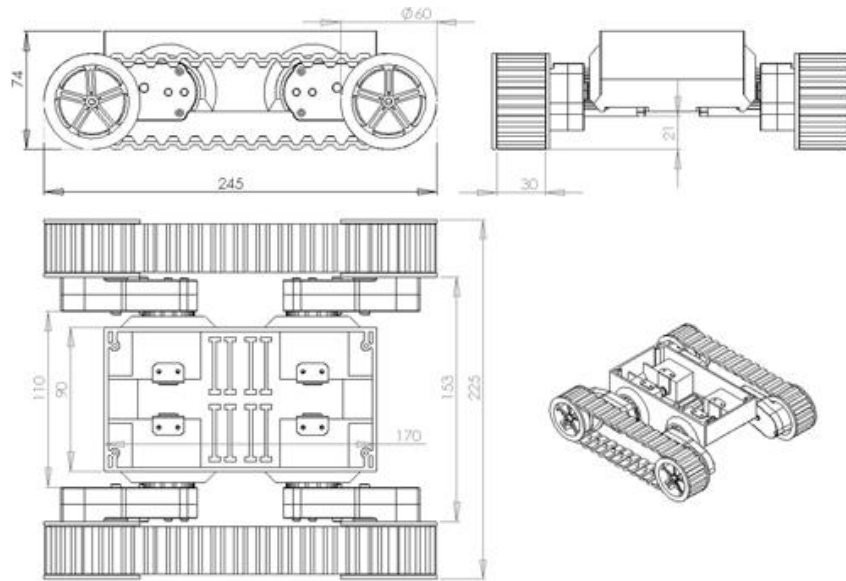
Setiap gearbox memiliki rasio 87:1 yang memiliki encoder quadrature optik yang memberikan 1000 pulsa dari tiap 3 putaran dari poros output. Chassis dapat ditingkatkan dengan menggunakan empat motor dan encoder sehingga ideal untuk roda mecanum.

Di dalam chassis ini terdapat dudukan baterai yang menerima 6x AA baterai. Disarankan untuk menggunakan baterai NiMH karena lebih tahan lama dan memiliki output arus yang lebih tinggi daripada baterai Alkaline.

spesifikasi:

1. Tegangan motor: 7.2V
2. Arus motor maksimal: 2.5A
3. Poros keluaran torsi stall: 10Kg/cm
4. Rasio Gearbox: 86.8:1
5. Jenis encoder: Quadrature
6. Resolusi Encoder: 1000 perubahan pulsa per 3 rotasi roda
7. Kecepatan: 1Km/hr
8. Dimensi:





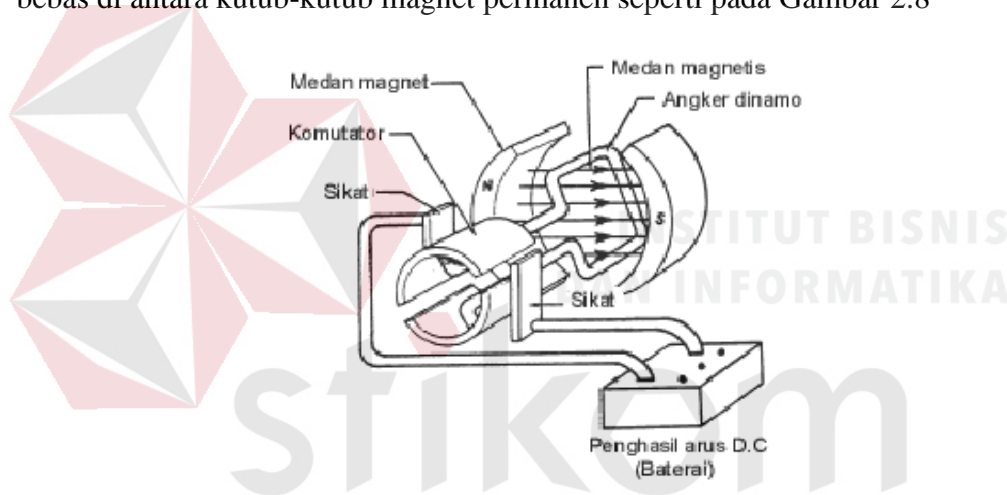
Gambar 2.7 Dagu Rover 5 4WD

## 2.5. Motor DC

Motor DC merupakan motor arus searah yang berfungsi mengubah tenaga listrik arus searah (listrik DC) menjadi tenaga gerak atau mekanik (Sumanto, 1984). Motor DC banyak ditemukan pada peralatan rumah tangga seperti mixer, blender, pemutar tape, dll. Selain itu motor DC juga digunakan dalam industri misalkan untuk mengaduk adonan, memutar konveyor, dll.

Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Kumparan medan pada motor dc

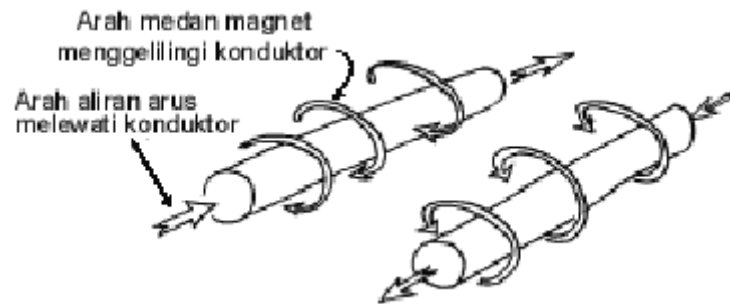
disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Jika terjadi putaran pada kumparan jangkar dalam pada medan magnet, maka akan timbul tegangan (GGL) yang berubah-ubah arah pada setiap setengah putaran, sehingga merupakan tegangan bolak-balik. Prinsip kerja dari arus searah adalah membalik fasa tegangan dari gelombang yang mempunyai nilai positif dengan menggunakan komutator, dengan demikian arus yang berbalik arah dengan kumparan jangkar yang berputar dalam medan magnet. Bentuk motor paling sederhana memiliki kumparan satu lilitan yang bisa berputar bebas di antara kutub-kutub magnet permanen seperti pada Gambar 2.8



Gambar 2.8 Motor DC Sederhana (Sumanto, 1994)

Catu tegangan DC dari baterai menuju ke lilitan melalui sikat yang menyentuh komutator, dua segmen yang terhubung dengan dua ujung lilitan. Kumparan satu lilitan pada gambar di atas disebut rotor dinamo. Rotor dynamo adalah sebutan untuk komponen yang berputar di antara medan magnet.

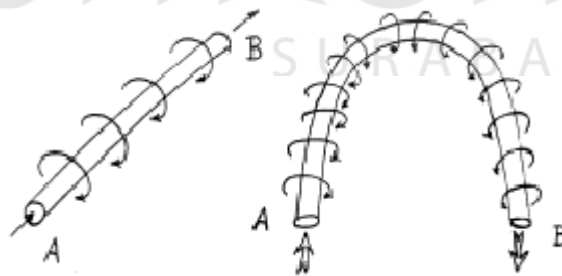
Jika arus lewat pada suatu konduktor, timbul medan magnet di sekitar konduktor seperti pada Gambar 2.9. Arah medan magnet ditentukan oleh arah aliran arus pada konduktor.



Gambar 2.9 Medan magnet yang membawa arus mengelilingi konduktor

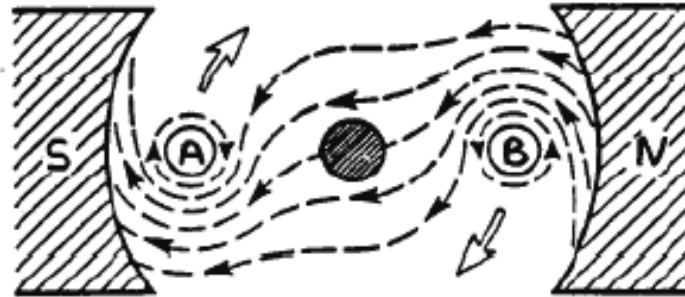
(Djati, 2010)

Aturan Genggaman Tangan Kanan bisa dipakai untuk menentukan arah garis fluks di sekitar konduktor. Genggam konduktor dengan tangan kanan dengan jempol mengarah pada arah aliran arus, maka jari-jari anda akan menunjukkan arah garis fluks. Gambar 2.10 menunjukkan medan magnet yang terbentuk di sekitar konduktor berubah arah karena bentuk U. Pada motor listrik konduktor berbentuk U disebut angker dinamo.



Gambar 2.10 Konduktor Berbentuk U (Djati, 2010)

Jika konduktor berbentuk U (angker dinamo) diletakkan di antara kutub utara dan selatan yang kuat medan magnet konduktor akan berinteraksi dengan medan magnet kutub seperti yang ditunjukkan pada gambar 2.11.



Gambar 2.11 Reaksi garis Fluks (Djati, 2010)

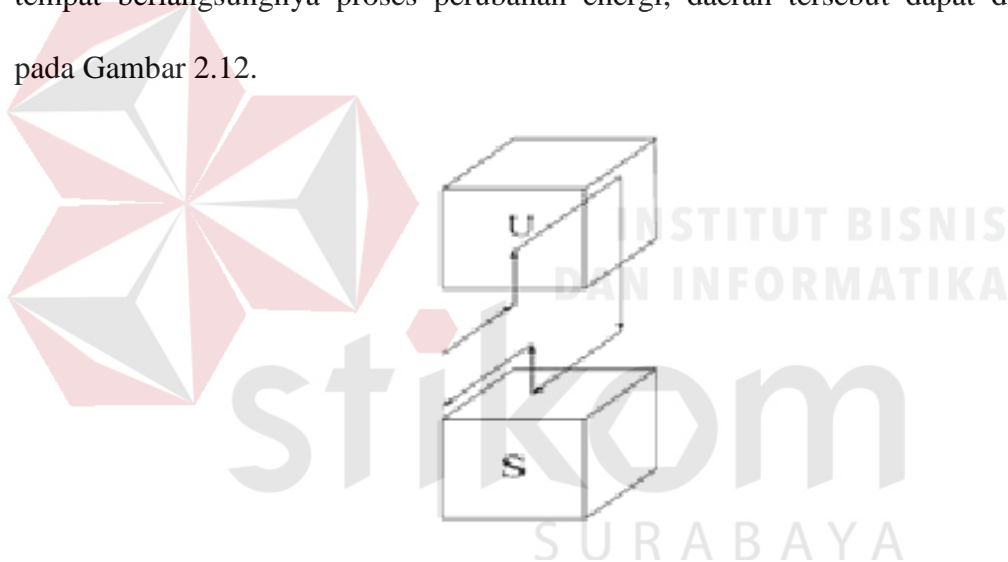
Lingkaran bertanda A dan B pada gambar 2.13 merupakan ujung konduktor yang dilengkungkan (*looped conductor*). Arus mengalir masuk melalui ujung A dan keluar melalui ujung B. Medan konduktor A yang searah jarum jam akan menambah medan pada kutub dan menimbulkan medan yang kuat di bawah konduktor. Konduktor akan berusaha bergerak ke atas untuk keluar dari medan kuat ini. Medan konduktor B yang berlawanan arah jarum jam akan menambah medan pada kutub dan menimbulkan medan yang kuat di atas konduktor. Konduktor akan berusaha untuk bergerak turun agar keluar dari medan yang kuat tersebut. Gaya-gaya tersebut akan membuat angker dinamo berputar searah jarum jam.

Mekanisme kerja untuk seluruh jenis motor secara umum :

- a. Arus listrik dalam medan magnet akan memberikan gaya.
- b. Jika kawat yang membawa arus dibengkokkan menjadi sebuah lingkaran / loop, maka kedua sisi loop, yaitu pada sudut kanan medan magnet, akan mendapatkan gaya pada arah yang berlawanan.
- c. Pasangan gaya menghasilkan tenaga putar / torque untuk memutar kumparan.

d. Motor-motor memiliki beberapa loop pada dinamonya untuk memberikan tenaga putaran yang lebih seragam dan medan magnetnya dihasilkan oleh susunan elektromagnetik yang disebut kumparan medan.

Pada motor DC, daerah kumparan medan yang dialiri arus listrik akan menghasilkan medan magnet yang melingkupi kumparan jangkar dengan arah tertentu. Konversi dari energi listrik menjadi energi mekanik (motor) maupun sebaliknya berlangsung melalui medan magnet, dengan demikian medan magnet disini selain berfungsi sebagai tempat untuk menyimpan energi, sekaligus sebagai tempat berlangsungnya proses perubahan energi, daerah tersebut dapat dilihat pada Gambar 2.12.



Gambar 2.12 Prinsip kerja motor DC (Djati, 2010)

Agar proses perubahan energi mekanik dapat berlangsung secara sempurna, maka tegangan sumber harus lebih besar daripada tegangan gerak yang disebabkan reaksi lawan. Dengan memberi arus pada kumparan jangkar yang dilindungi oleh medan maka menimbulkan perputaran pada motor. Dalam memahami sebuah motor, penting untuk mengerti apa yang dimaksud dengan beban motor. Beban dalam hal ini mengacu kepada keluaran tenaga putar / torque

sesuai dengan kecepatan yang diperlukan. Beban umumnya dapat dikategorikan ke dalam tiga kelompok :

- a. Beban torque konstan adalah beban dimana permintaan keluaran energinya bervariasi dengan kecepatan operasinya namun torquanya tidak bervariasi. Contoh beban dengan torque konstan adalah conveyors, rotary kilns, dan pompa displacement konstan.
- b. Beban dengan variabel torque adalah beban dengan torque yang bervariasi dengan kecepatan operasi. Contoh beban dengan variabel torque adalah pompa sentrifugal dan fan (torque bervariasi sebagai kuadrat kecepatan).
- c. Beban dengan energi konstan adalah beban dengan permintaan torque yang berubah dan berbanding terbalik dengan kecepatan. Contoh untuk beban dengan daya konstan adalah peralatan-peralatan mesin.

Untuk menentukan arah putaran motor digunakan kaedah Fleming tangan kiri. Kutub-kutub magnet akan menghasilkan medan magnet dengan arah dari kutub utara ke kutub selatan. Jika medan magnet memotong sebuah kawat penghantar yang dialiri arus searah dengan empat jari, maka akan timbul gerak searah ibu jari. Gaya ini disebut gaya Lorentz, yang besarnya sama dengan  $F$ .

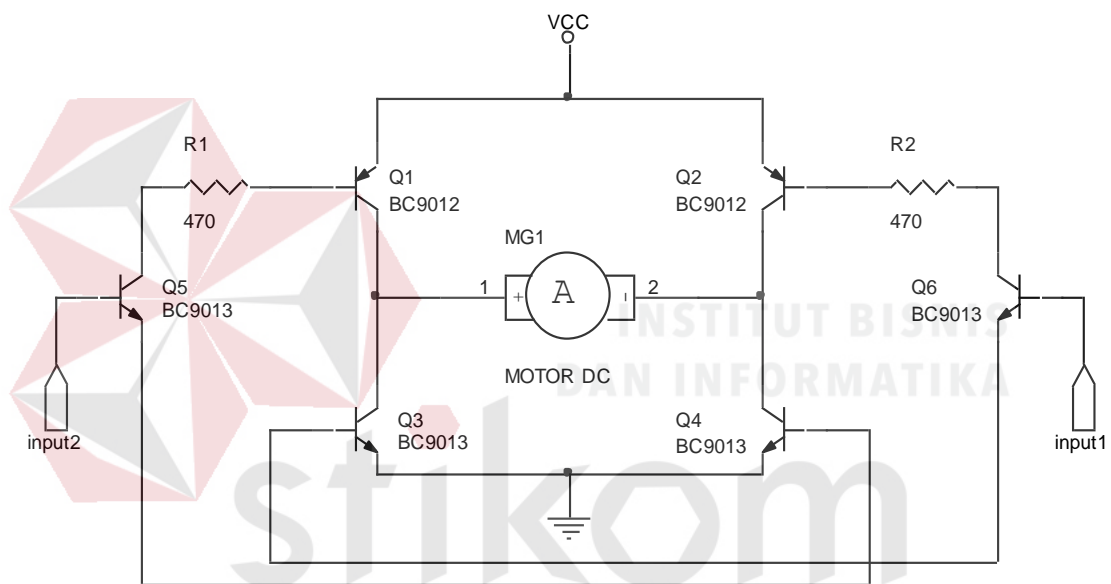
Prinsip motor adalah aliran arus di dalam penghantar yang berada di dalam pengaruh medan magnet akan menghasilkan gerakan. Besarnya gaya pada penghantar akan bertambah besar jika arus yang melalui penghantar bertambah besar.

## **2.6. Pengendali Motor**

Pengendali motor (*motor driver*) adalah rangkaian komponen yang dapat mengatur kinerja putaran motor. Dengan menggunakan motor driver,

pengendalian motor meliputi *men-start*, *men-stop*, dan memilih arah perputaran. Selain itu dapat juga mengatur kecepatan putar motor dan mencegah terjadinya kelebihan beban arus pada motor. Penggunaan *motor driver* disesuaikan dengan jenis dan besar arus dari motor yang akan dikendalikan.

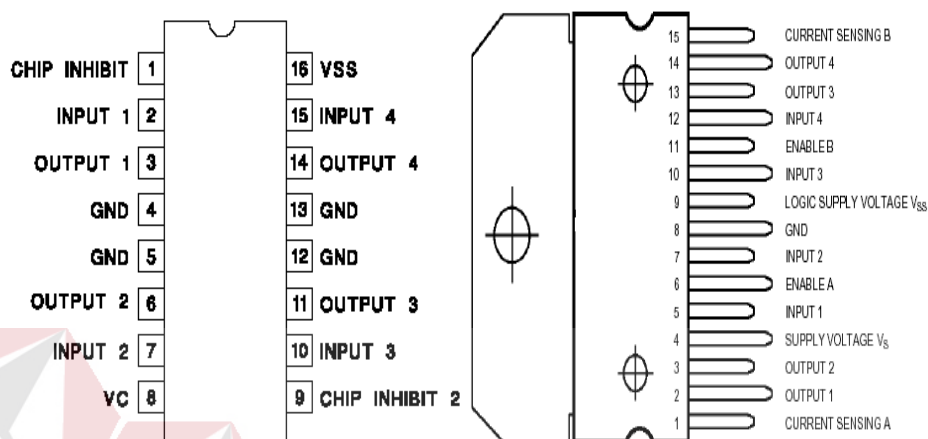
H-Bridge adalah salah satu contoh rangkaian *motor driver*. Susunan dasar dari rangkaian tersebut adalah beberapa transistor yang terangkai menyerupai huruf 'H'. Oleh karena itu, rangkaian tersebut dinamakan H-Bridge.



Gambar 2.13 Rangkaian dasar *H-Bridge motor driver*

Sesuai dengan gambar 2.13, motor DC akan berputar jika salah satu transistor NPN (Q3 atau Q4) dan salah satu transistor PNP (Q1 atau Q2) aktif bersamaan. Untuk membuat hal tersebut, maka perlu diaktifkannya salah satu transistor Q5 atau Q6, yaitu dengan memberikan tegangan masukan pada kaki basis (*input 1* atau *input 2*).

Saat ini rangkaian H-Bridge *motor driver* telah dikemas dalam bentuk IC, seperti IC LM293D dan IC LM298. Masing-masing IC tersebut memiliki kemampuan menghantarkan sejumlah arus yang berbeda.

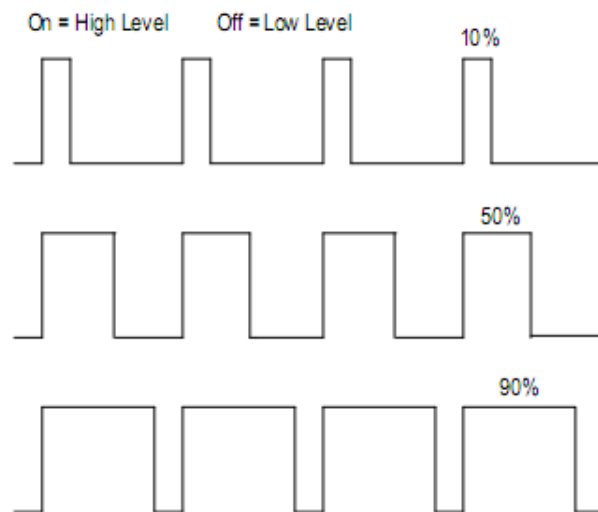


Gambar 2.14 (a) L293D Motor driver, (b) L298 Motor driver

## 2.7. Pulse Width Modulation (PWM)

Metode *Pulse Width Modulation* (PWM) adalah metode yang cukup efektif untuk mengendalikan kecepatan motor DC. PWM ini bekerja dengan cara membuat gelombang persegi yang memiliki perbandingan pulsa *high* terhadap pulsa *low* yang telah tertentu biasanya diskalakan dari 0 hingga 100 % gelombang persegi ini memiliki frekuensi tetap (biasanya max 10 KHz) namun lebar pulsa *high* dan *low* dalam 1 periode yang akan diatur. Perbandingan pulsa *high* terhadap *low* ini akan menentukan jumlah daya yang diberikan kemotor DC. Pada gambar 2.17 kita dapat melihat bagaimana pengendalian dengan *Pulse Width Modulation* (PWM).





Gambar 2.15 Pengendalian dengan *Pulse Width Modulation* (PWM)

## 2.8. *Liquid Cristal Display* (LCD)

*Display* elektronik adalah salah satu komponen elektronika yang berfungsi sebagai tampilan suatu data, baik karakter, huruf ataupun grafik. LCD (*Liquid Cristal Display*) adalah salah satu jenis *display* elektronik yang dibuat dengan teknologi CMOS *logic* yang bekerja dengan tidak menghasilkan cahaya tetapi memantulkan cahaya yang ada di sekelilingnya terhadap *front-lit* atau mentransmisikan cahaya dari *back-lit*. LCD (*Liquid Cristal Display*) berfungsi sebagai penampil data baik dalam bentuk karakter, huruf, angka ataupun grafik.



Gambar 2.16 LCD (*Liquid Cristal Display*)

LCD adalah lapisan dari campuran organik antara lapisan kaca bening dengan elektroda transparan indium oksida dalam bentuk tampilan *seven-segment* dan lapisan elektroda pada kaca belakang. Ketika elektroda diaktifkan dengan medan listrik (tegangan), molekul organik yang panjang dan silindris menyesuaikan diri dengan elektroda dari segmen. Lapisan *sandwich* memiliki *polarizer* cahaya vertikal depan dan *polarizer* cahaya horisontal belakang yang diikuti dengan lapisan reflektor. Cahaya yang dipantulkan tidak dapat melewati molekul-molekul yang telah menyesuaikan diri dan segmen yang diaktifkan terlihat menjadi gelap dan membentuk karakter data yang ingin ditampilkan.

Dalam modul LCD (*Liquid Cristal Display*) terdapat mikrokontroler yang berfungsi sebagai pengendali tampilan karakter LCD (*Liquid Cristal Display*). mikrokontroler pada suatu LCD (*Liquid Cristal Display*) dilengkapi dengan memori dan register. Memori yang digunakan mikrokontroler internal LCD adalah:

- a. DDRAM (*Display Data Random Access Memory*) merupakan memori tempat karakter yang akan ditampilkan berada.
- b. CGRAM (*Character Generator Random Access Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai dengan keinginan.
- c. CGROM (*Character Generator Read Only Memory*) merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut merupakan karakter dasar yang sudah ditentukan secara permanen oleh pabrikan pembuat LCD (*Liquid Cristal Display*) tersebut sehingga pengguna tinggal

mangambilnya sesuai alamat memorinya dan tidak dapat merubah karakter dasar yang ada dalam CGROM.

Register control yang terdapat dalam suatu LCD diantaranya adalah :

- a. Register perintah yaitu register yang berisi perintah-perintah dari mikrokontroler ke panel LCD (*Liquid Cristal Display*) pada saat proses penulisan data atau tempat status dari panel LCD (*Liquid Cristal Display*) dapat dibaca pada saat pembacaan data.
- b. Register data yaitu register untuk menuliskan atau membaca data dari atau keDDRAM. Penulisan data pada register akan menempatkan data tersebut keDDRAM sesuai dengan alamat yang telah diatur sebelumnya.

Pin, kaki atau jalur input dan kontrol dalam suatu LCD (*Liquid Cristal Display*) diantaranya adalah :

- a. Pin data adalah jalur untuk memberikan data karakter yang ingin ditampilkan menggunakan LCD (*Liquid Cristal Display*) dapat dihubungkan dengan bus data dari rangkaian lain seperti mikrokontroler dengan lebar data 8 bit.
- b. Pin RS (*Register Select*) berfungsi sebagai indikator atau yang menentukan jenis data yang masuk, apakah data atau perintah. Logika *low* menunjukkan yang masuk adalah perintah, sedangkan logika *high* menunjukkan data.
- c. Pin R/W (*Read Write*) berfungsi sebagai instruksi pada modul jika *low* tulis data, sedangkan *high* baca data.
- d. Pin E (*Enable*) digunakan untuk memegang data baik masuk atau keluar.
- e. Pin VLCD berfungsi mengatur kecerahan tampilan (kontras) dimana pin ini dihubungkan dengan trimpot 5 Kohm, jika tidak digunakan dihubungkan ke *ground*, sedangkan tegangan catu daya ke LCD sebesar 5 Volt.

## 2.9. Rotary Encoder

*rotary encoder* adalah perangkat elektromekanik yang dapat memonitor gerakan dan posisi. *Rotary encoder* pada umumnya menggunakan sensor optik untuk menghasilkan serial pulsa yang dapat diartikan menjadi gerakan, posisi, dan arah. Sehingga posisi sudut suatu poros benda berputar dapat diolah menjadi informasi berupa kode digital oleh *rotary encoder* untuk diteruskan oleh rangkaian kendali. Rotary encoder tersusun dari suatu piringan tipis yang memiliki lubang-lubang pada bagian lingkaran piringan.

LED ditempatkan pada salah satu sisi piringan sehingga cahaya akan menuju ke piringan. Di sisi yang lain suatu photo-transistor diletakkan sehingga *photo-transistor* ini dapat mendeteksi cahaya dari LED yang berseberangan. Apabila posisi piringan mengakibatkan cahaya dari LED dapat mencapai *photo-transistor* melalui lubang-lubang yang ada, maka *photo-transistor* akan mengalami saturasi dan akan menghasilkan suatu pulsa gelombang persegi. Semakin banyak deretan pulsa yang dihasilkan pada satu putaran menentukan akurasi *rotary encoder* tersebut. (rizqiawan, 2009)



Gambar 2.17 Rotary encoder

## 2.10. Sensor Ultrasonik PING Parallax

Ping))) *Ultrasonic Range Finder*, adalah modul pengukur jarak dengan ultrasonik buatan Parallax Inc. yang didesain khusus untuk teknologi robotika. Dengan ukurannya yang cukup kecil (2,1cm x 4,5cm), sensor seharga 350 ribu rupiah ini dapat mengukur jarak antara 3 cm sampai 300 cm. Keluaran dari Ping))) berupa pulsa yang lebarnya merepresentasikan jarak. Lebar pulsanya bervariasi dari 115 uS sampai 18,5 mS.

Pada dasarnya, Ping))) terdiri dari sebuah chip pembangkit sinyal 40KHz, sebuah speaker ultrasonik dan sebuah mikropon ultrasonik. Speaker ultrasonik mengubah sinyal 40 KHz menjadi suara sementara mikropon ultrasonik berfungsi untuk mendeteksi pantulan suaranya. Pada modul Ping))) terdapat 3 pin yang digunakan untuk jalur power supply (+5V), ground dan signal. Pin signal dapat langsung dihubungkan dengan mikrokontroler tanpa tambahan komponen apapun.

Ping))) mendeteksi objek dengan cara mengirimkan suara ultrasonik dan kemudian “mendengarkan” pantulan suara tersebut. Ping))) hanya akan mengirimkan suara ultrasonik ketika ada pulsa trigger dari mikrokontroler (Pulsa high selama 5uS). Suara ultrasonik dengan frekuensi sebesar 40KHz akan dipancarkan selama 200uS. Suara ini akan merambat di udara dengan kecepatan 344.424m/detik (atau 1cm setiap 29.034uS), mengenai objek untuk kemudian terpantul kembali ke Ping))). Selama menunggu pantulan, Ping))) akan menghasilkan sebuah pulsa. Pulsa ini akan berhenti (low) ketika suara pantulan terdeteksi oleh Ping))). Oleh karena itulah lebar pulsa tersebut dapat merepresentasikan jarak antara Ping))) dengan objek.

Berdasarkan Datasheet :

Jarak = (Lebar Pulsa/29.034uS)/2 (dalam cm)

Jarak = (Lebar Pulsa x 0.034442)/2 (dalam cm)

Karena  $1/29.034 = 0.034442$

(Sanjaya, 2013)



Gambar 2.18 Sensor Ultrasonik PING Parallax

