BAB II
TEORI PENUNJANG

2.1 Mikrokontroler

Mikrokontroler adalah sebuah sistem komputer fungsional dalam sebuah chip. Di dalamnya terkandung sebuah inti prosesor, memori (sejumlah kecil RAM, memori program, atau keduanya), dan perlengkapan *input output*. Dengan kata lain, mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus, cara kerja mikrokontroler sebenarnya membaca dan menulis data.

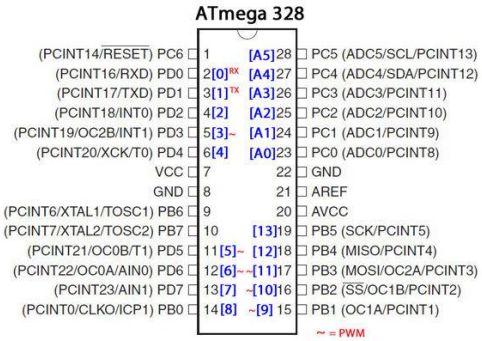
Mikrokontroler digunakan dalam produk dan alat yang dikendalikan secara automatis, seperti sistem kontrol mesin, *remote controls*, mesin kantor, peralatan rumah tangga, alat berat, dan mainan.

2.1.1 Arduino Uno

Arduino Uno adalah *board* mikrokontroler berbasis ATmega328.Arduino Uno memiliki beberapa fitur antara lain:

a. Mikrokontroler ATmega328.
b. Catu Daya 5V.
c. Teganan *Input* (rekomendasi) 7-12V.
d. Teganan *Input* (batasan) 6-20V.
e. Pin I/O Digital 14 (*of which* 6 *provide* PWM *output*).
f. Pin *Input Analog* 6.
g. Arus DC per Pin I/O 40 mA.
h. Arus DC per Pin I/O untuk Pin 3.3V 50 mA.
i. *Flash Memory* 32 KB (ATmega328) dimana 0.5 KB digunakan oleh *bootloader*.
j. SRAM 2 KB (ATmega328).
k. EEPROM 1 KB (ATmega328).
l. *Clock Speed* 16 MHz.

2.1.2 Konfigurasi PIN Arduino Uno



Gambar 2.1 Konfigurasi Pin ATmega 328 Pada Arduino Uno

Pada gambar 2.1 konfigurasi pin Atmega328 akan dijelaskan pada table 2.1 dibawah ini.



Gambar 2.2 Bentuk Fisik Arduino Uno

Pada gambar 2.2 merupakan gambar Arduino Uno yang akan digunakan pada perancangan dan implementasi pada plotter.

Tabel 2.1: Konfigurasi *Port* B

Port Pin	***Alternate Functions***
PB7	XTAL2(*Chip Clock Oscillator* Pin 2) TOSC2 (*Timer Oscillator*Pin 2) PCINT7 (Pin *Change Interrupt* 7)
PB6	XTAL1 (*Chip Clock Oscillator* Pin 1 *or External Clock input*) TOSC1(*Timer Oscillator*Pin 1) PCINT6 (Pin *Change Interrupt* 6)
PB5	Sck(SPI *Bus Master clock input*) PCINT5(Pin *Change Interrupt* 4)

Port Pin	*Alternate Functions*
PB4	MISO (SPI *Bus Master Input*/*Slave Output*) PCINT4(Pin *Change Interrupt* 4)
PB3	MOSI (SPI *Bus Master Slave Select*) OC2A (*Timer/Counter1 Output Compare Match A Output*) PCINT3 (Pin *Change Interrupt*)
PB2	SS (SPI *Bus Master Slave Select*) OC2A (*Timer/CounterOutput Compare Match A Output*) PCINT2 (Pin *Change Interrupt*)
PB1	OC2A (*Timer/Counter1 Output Compare Match A Output*) PCINT1 (Pin *Change Interrupt*)
PB1	ICP1 (*Timer/Counter 1 Input Capture Input*) CLKO (*Divided System Clock Output*) PCINT0 (Pi*n Change Interrupt* 0)

Table 2.2: Konfigurasi *Port* C

Port Pin	*Alternate Functions*
PC6	RESET (*Rese*t pin) PCINT14 (Pin *Change Interrupt* 14)
PC5	ADC5 (ADC *Input Channel* 5) SCL (2 *Wire Serial Bus Data Input/Output*) PCINT13 (Pin *Change Interrupt* 13)
PC4	ADC4 (ADC *Input Channel* 4) SCL (2 *Wire Serial Bus Data Input/Output*) PCINT12 (Pin *Change Interrupt* 12)
PC3	ADC3 (ADC *Input Channel* 3) PCINT11 (Pin *Change Interrupt* 11)
PC2	ADC2 (ADC *Input Channel* 2) PCINT10 (Pin *Change Interrupt* 10)
PC1	ADC1 (ADC *Input Channel* 1) PCINT9 (Pin *Change Interrupt* 9)
PC0	ADC0 (ADC *Input Channel* 0) PCINT8 (Pin *Change Interrupt* 8)

Tabel 2.3: Konfigurasi *Port* D

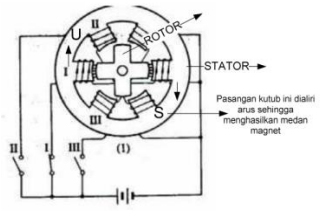
Port Pin	***Alternate Functions***
PD7	AIN1 (*Analog Comparator Negative Input*) PCINT23 (Pin *Change Interrupt* 23)
PD6	AIN0 (*Analog Comparator Positive Input*) OC0A (*Timer/Counter0 Output Compare Match A Output*) PCINT22(Pin *Change Interrupt* 22)
PD5	T1 (*Timer/Counter1 External Counter Input*) OCDB (*Timer/Counter0 Output Compare Match B Output*) PCINT21(Pin *Change Interrupt* 21)
PD4	XCK (USART *External Clock Input/Output*) T0 (*Timer/Counter 0 Output Compare Match B Output*) PCINT20 (Pin *Change Interrupt* 20)
PD3	INT1 (*External Interrupt1input*) T0 (*Timer/Counter 2 Output Compare Match B Output*) PCINT19 (Pin *Change Interrupt* 19)
PD2	INT0 (*External Interrupt 0 Inpu*t) PCINT18 (Pin *Change Interrupt* 18)
PD1	TXD (USART *Output* Pin) PCINT17 (Pin *Change Interrupt* 17)
PD0	RXD (USART Input Pin) PCINT16 (Pin *Change Interrupt* 16)

2.2 *Motor stepper*

*Motor stepper*adalah suatu alat penggerak yang memanfaatkan gaya tarik magnet. Motor stepper adalah perangkat elektromekanis yang bekerja dengan mengubah pulsa elektronis menjadi gerakan mekanis diskrit.*Rotor* berhenti pada posisi kutub yang dieksitasi oleh arus yang mengalir pada lilitan. *Rotor* pada *motor* biasanya berputar secara kontinyu jika *motor* dieksitasi, tetapi *rotor* pada *motor stepper*berubah dari posisi diam dengan mengubah eksitasi kutub. Arus yang mengalir pada setiap lilitan hanyasesaat sehingga bentukarusnyaberupa pulsa. *Rotor* berputar karena pulsayang bergantian. Kecepatan putaran *rotor* ditentukan oleh kecepatan perpindahan pulsa dansudut putaran sebanding dengan banyaknya pulsa yang diberikan. Apabila satu pulsa inputmenghasilkan

perputaran sejauh 1,8 derajat, sehingga 20 pulsa akan menghasilkan perputaran penuh sebesar 36 derajat dan untuk mendapatkan satu putaran penuh 360 derajat dibutuhkan 200 pulsa.

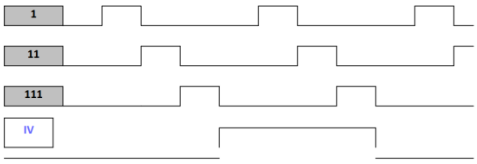
Rotor yang digunakan terbuat dari baja lunak dan memiliki sejumlah gigi yang jumlahnya kurang dari jumlah kutub pada *stator*. *Stator*memiliki beberapa pasangkutubdimana setiap pasang kutub diaktifkan melalui prinsip elektromagnetik oleh arus yang mengalirmelalui kumparan yang dililitkan pada masing – masing kutub. Pada sepasang kutub dalam keadaan aktif sehingga akan timbul medan magnet yang kemudian menarik pasangan gigi *rotor* terdekat, sehingga gigi akan bergerak ke posisi segaris dengan kutub. Untuk menggerakkan sebuah *motor stepper*setiap pasang kumparan *stator* harus disambungkan dengan aliran listrik dan diputuskan secara bergantian dalam urutan yang benar. Dengan demikian, input ke *motor* berupa deretan pulsa yang menghasilkan outputke setiap pasang kumparan *stator*. Sistem penggerakyang biasa digunakan terdiri daridua blok utama yaitu pengatur urutan logika dimana menerima pulsa-pulsa input dan menghasilkan pulsa–pulsaoutput dalam urutan sebagai mana yang dibutuhkan untuk mengontrol penggerakagar menghasilkan pulsa output dengan amplitudo yang sesuai. *Motorstepper* banyak digunakan dalam berbagai aplikasi, dipergunakan apabila dikehendaki jumlah putaran yang tepat atau diperlukan sebagian dari putaran *motor*. Aplikasi penggunaan *motor* langkah dapat juga dijumpai dalam bidang industri atau untuk jenis *motor* langkah kecil dapat di gunakan dalam perancangan suatu alat mekatronik atau robot. Pada gambar 2.1 berikut ditunjukkan dasar susunan sebuah *motor stepper*.



Gambar 2.3 Diagram Motor Stepper

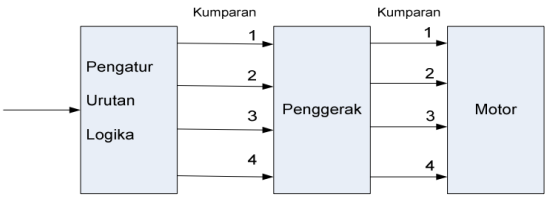
Magnet permanen berputar kearah medan magnet yang aktif. Apabila kumparan *stator* dialiri arus sedemikian rupa, sehingga akan timbul medan magnet dan *rotor*akan berputar mengikuti medan magnet tersebut. Setiap

pengalihan arus ke kumparan berikutnya menyebabkan medan magnet berputar menurut suatu sudut tertentu, biasanya informasi besar sudut putar tertulis pada badan *motor stepper* yang bersangkutan. Jumlah keseluruhan pengalihan menentukan sudut perputaran *motor*. Jika pengalihan arus di tentukan, sehingga *rotor* akan berhenti pada posisi terakhir. Jika kecepatan pengalihan tidak terlalu tinggi, sehingga slip akan dapat dihindari. Memerlukan umpan balik (*feedback*) pada pengendalian *motor stepper*. *Motor stepper* yang akan di gunakan memiliki 4 fasa (pole atau kutub), pengiriman pulsa dari mikrokontroler ke rangkaian *motor* langkah dilakukan secara bergantian, masing-masing 3 data(sesuai dengan jumlahfasa-nya), sebagian di tunjukkan pada gambar 2.2 berikut.



Gambar 2.4Pemberian Data/Pulsa Pada Motor Stepper

Pada saat yang sama, untuk tiap *motor stepper*, tidak boleh ada 2 (dua) masukan ataulebih yang mengandung pulsasama dengan 1(*high*), atau dengan kata lain, pada suatu saat hanya sebuah masukan yang bernilai 1 (satu) sedangkan lainnya bernilai 0 (nol). Sistem penggerak yang biasa digunakan terdiri dari dua blok utama yaitu pengaturan urutan logika dan sebuah penggerak ditunjukkan pada gambar 2.5.



Gambar 2.5Sistem Penggerak Motor Stepper

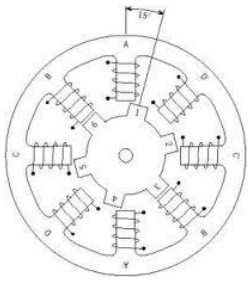
Pengatur urutan logika menerima pulsa–pulsa input dan menghasilkan pulsaoutput dalam urutan sebagaimana yang dibutuhkan untuk mengontrol penggerak agar menghasilkan pulsa *output* dengan *amplitude* yang sesuai.

Motor stepper bergerak berdasarkan urutan pulsa yang diberikan kepada motor. Karena itu, untuk menggerakkan *motor stepper* diperlukan pengendali

motor stepper yang membangkitkan pulsa-pulsa periodik. Pada dasar-nya terdapat 3 tipe motor stepper yaitu:

2.2.1. *Motor Stepper* Tipe *Variable Reluctance* (VR)

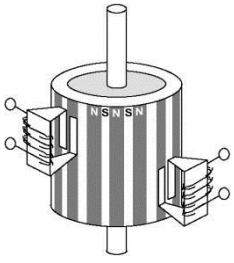
Motor stepper jenis ini telah lama ada dan merupakan jenis motor yang secara struktural paling mudah untuk dipahami. Motor ini terdiri atas sebuah rotor besi lunak dengan beberapa gerigi dan sebuah lilitan *stator*. Ketika lilitan *stator* diberi energi dengan arus DC, kutub-kutubnya menjadi termagnetasi. Perputaran terjadi ketika gigi-gigi *rotor* tertarik oleh kutub-kutub *stator*. Berikut ini adalah penampang melintang dari *motor stepper* tipe *variable reluctance* (VR):



Gambar 2.6Penampang Melintang Motor Stepper Variable Reluctance (VR)

2.2.2. *Motor Stepper* Tipe *Permanent Magnet* (PM)

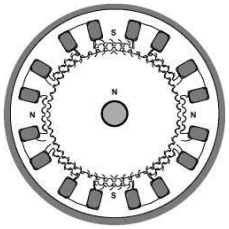
Motor stepper jenis ini memiliki *rotor* yang berbentuk seperti kaleng bundar (*tin can*) yang terdiri atas lapisan magnet permanen yang diselang-seling dengan kutub yang berlawanan. Dengan adanya magnet permanen, maka intensitas *fluks* magnet dalam motor ini akan meningkat sehingga dapat menghasilkan torsi yang lebih besar. Motor jenis ini biasanya memiliki resolusi langkah (step) yang rendah yaitu antara 7,5° hingga 15° per langkah atau 48 hingga 24 langkah setiap putarannya. Berikut ini adalah ilustrasi sederhana dari motor stepper tipe permanent magnet:



Gambar 2.7 Ilustrasi Motor Stepper Permanent Magnet (PM)

2.2.3. *Motor Stepper* Tipe *Hybrid* (HB)

Motor stepper tipe hibrid memiliki struktur yang merupakan kombinasi dari kedua tipe motor stepper sebelumnya. *Motor stepper* tipe hibrid memiliki gerigi seperti pada motor tipe VR dan juga memiliki magnet permanen yang tersusun secara aksial pada batang porosnya seperti *motor* tipe PM. *Motor* tipe ini paling banyak digunkan dalam berbagai aplikasi karena kinerja lebih baik. *Motor* tipe hibrid dapat menghasilkan resolusi langkah yang tinggi yaitu antara 3,6° hingga 0,9° per langkah atau 100-400 langkah setiap putarannya. Berikut ini adalah penampang melintang dari *motor stepper* tipe hibrid:

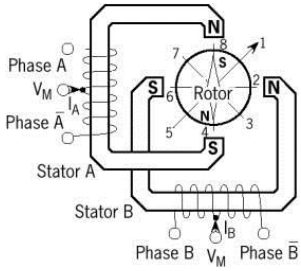


Gambar 2.8 Penampang Melintang Motor Stepper Hibrid

Berdasarkan metode perancangan rangkain pengendali motor stepper, motor stepper dapat dibagi menjadi jenis unipolar dan bipolar.

2.2.4. *Motor Stepper Unipolar*

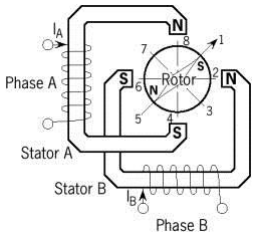
Rangkaian pengendali *motor stepper unipolar* lebih mudah dirancang karena hanya memerlukan satu *switch* atau transistor setiap lilitannya. Untuk menjalankan dan menghentikan *motor* ini cukup dengan menerapkan pulsa digital yang hanya terdiri atas tegangan positif dan nol (*ground*) pada salah satu terminal lilitan (*wound*) *motor* sementara terminal lainnya dicatu dengan tegangan positif konstan (VM) pada bagian tengah (*center tap*) dari lilitan.



Gambar 2.9Ilustrasi Motor Stepper Dengan Lilitan Unipolar

2.2.5. *Motor Stepper Bipolar*

Untuk *motor stepper* dengan lilitan *bipolar*, diperlukan sinyal pulsa yang berubah-ubah dari positif ke negatif dan sebaliknya. Jadi pada setiap terminal lilitan (A & B) harus dihubungkan dengan sinyal yang mengayun dari positif ke negatif dan sebaliknya. Karena itu dibutuhkan rangkaian pengendali yang agak lebih kompleks daripada rangkaian pengendali untuk *motor unipolar*. *Motor stepper bipola*r memiliki keunggulan dibandingkan dengan *motor stepper unipolar* dalam hal torsi yang lebih besar untuk ukuran yang sama.



Gambar 2.10Ilustrasi Motor StepperDengan Lilitan Bipolar

Motor stepper memiliki karakter yang lebih dibanding motor DC seperti berikut :

a) Sudut rotasi motor proporsional dengan pulsa input sehingga lebih mudah diatur.
b) Motor dapat langsung memberikan torsi penuh pada saat mulai bergerak.
c) Posisi dan pergerakan repetisinya dapat ditentukan secara presisi.
d) Memiliki respon yang sangat baik terhadap start, stop dan berbalik perputaran.
e) Sangat realibel karena tidak adanya brush yang bersentuhan dengan rotor seperti pada motor DC.
f) Dapat menghasilkan perputaran yang lambat sehingga beban dapat dikopel langsung ke porosnya.
g) Frekuensi perputaran dapat ditentukan secara bebas dan mudah pada range yang luas.

2.2.6. *Driver Motor Stepper*

Secara teoritis, sebuah motor stepper dapat digerakkan langsung oleh mikrokontroller. Dalam kenyataannya, arus dan tegangan yang dikeluarkan oleh

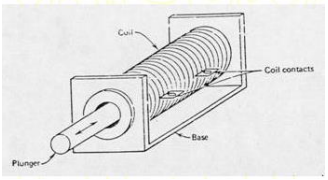
mikrokontroller terlalu kecil untuk menggerakkan sebuah motor stepper. Gerbang-gerbang *Transistor Transistor Logic* (TTL) mikrokontroller hanya mampu mengeluarkan arus dalam orde miliampere dan tegangan antara 2 sampai 2,5 Volt. Sementara itu untuk menggerakkan motor stepper diperlukan arus yang lebih besar (dalam orde ampere) dan tegangan berkisar 5 sampai 24 Volt. Untuk mengatasi masalah tersebut, diperlukan sebuah piranti tambahan yang memenuhi kebutuhan arus dan tegangan yang cukup besar. Rangkaian driver motor stepper merupakan rangkaian "open collector", dimana output rangkaian ini terhubung dengan ground untuk mencatu lilitan-lilitan motor stepper.

2.3 Solenoid

Solenoid adalah alat yang dapat mengubah energi listrik menjadi energi mekanik atau linier. *Solenoid* yang paling umum biasanya menggunakan medan magnet yang dibuat dari arus listrik yang ditrigger sebagai aksi kerja dorong atau tarik pada sebuah objek sebagai *strarter*, *valve*, *switch* dan *latches*.

Jenis paling sederhana dari *solenoid* mengandalkan dua aspek utama untuk fungsi *solenoid* tersebut, yaitu sebuah kawat (atau berenamel) terisolasi yang dibentuk menjadi gulungan ketat, dan batang yang terbuat dari besi atau baja. Batang besi atau baja merupakan feromagnetik, sebuah properti yang dapat berfungsi sebagai elektromagnetik saat diberi arus listrik.

Ketika diberi arus listrik, kawat yang dibentuk menjadi koil menerima arus. Medan magnet yang dihasilkan menarik besi atau batang baja dengan kuat. Batang yang dihubungkan pada sebuah pegas bergerak ke kumparan dan akan tetap pada posisinya sampai arus dihentikan, kondisi pegas saat ini menjadi tertekan. Ketika arus dimatikan, pegas kembali ke posisi semula dan menarik batang besi atau baja pada posisi awalnya.Gambar 2.11 dibawah menunjukkan bentuk *solenoid*.



Gambar 2.11Kumparan Fisik Solenoid

2.4 Algoritma Bresenham

Algoritma Bresenham adalah suatu algoritma yang menentukan titik-titik dalam dimensi yang membentuk pendekatan dekat dengan garis antara dua titik yang diberikan. Pendekatan ini biasa digunakan untuk menggambar garis pada layar komputer, karena hanya menggunakan integer penambahan, pengurangan dan pergeseran. Algoritma ini merupakan salah satu algoritma paling awal yang dikembangkan di bidang komputer grafis.

2.4.1 Algoritma Garis Bresenham

Garis lurus dinyatakan dinyatakan dalam persamaan :

$$y = mx + c \quad (1)$$; dimana :

- m : *gradient*
- c : konstanta.

Untuk menggambarkan piksel-piksel dalam garis lurus, parameter yang digunakan tergantung dari *gradient*, jika besarnya *gradient* diantara 0 dan 1, maka digunakan sumbu x sebagai parameter dan sumbu y sebagai hasil dari fungsi, sebaliknya, bila *gradient* melebihi 1, maka sumbu y digunakan sebagai parameter dan sumbu x sebagai hasil dari fungsi, hal ini bertujuan untuk menghindari terjadinya *gaps* karena adanya piksel yang terlewatkan. Hasil dari fungsi biasanya merupakan bilangan *real*, sedangkan koordinat piksel dinyatakan dalam bilangan *integer* (x,y), maka diperlukan operasi pembulatan kedalam bentuk *integer* terdekat. Penggambaran garis lurus dengan metode diatas dimulai dengan operasibilangan *real* untuk menghitung *gradient* m dan konstanta c.

$$m = (y2 - y1) / (x2 - x1) \quad (2)$$

$$c = y1 / m * x1 * \quad (3)$$

Operasi bilangan *real* berikutnya adalah menghitung nilai y dengan persamaan (1) Untuk mendapatkan koordinat piksel (x,y), untuk setiapnilai x, dari =x1 sampai x=x2, operasi inilah yang perlu dihindari,karena operasi ini memerlukan waktu operasi yang besar. Bresenham pada tahun 1965, melakukan perbaikan dari algoritma perhitungan koordinat piksel yang menggunakan persamaan (1), dengan cara menggantikan operasi bilangan real perkalian dengan operasi penjumlahan, yang kemudian dikenal dengan Algoritma Bresenham. Pada algoritma bresenham,

nilai y kedua dan seterusnya, dihitung dari nilai y sebelumnya, sehingga hanya titik y pertama yang perlu dilakukan operasi secara lengkap. Perbaikan algoritma ini ternyata tidak menghasilkan perbaikan yang cukup siginifikan. Perbaikan berikutnya dilakukan dengan cara menghilangkan operasi bilangan real dengan operasi bilangan integer. Operasi bilangan integer jauh lebih cepat dibandingkan dengan operasi bilangan real, terutama pada penambahan dan pengurangan.

1. Algoritma Bresenham untuk dx > dy
    a. Tentukan 2 titik yang akan dihubungkan dalam pembentukan garis.
    b. Tentukan salah satu titik disebelah kiri sebagai titik awal, yaitu (X0,Y0) dan titik lainnya sebagai titik akhir (X1,Y1).
    c. Hitung :
        - $Dx = x2 - x1$
        - $Dy = y2 - y1$
        - $d1 = 2 * Dx$
        - $d2 = 2 * Dy - 2 * Dx$
        - $e = d1 - dx$
        - $x = x1$
        - $y = y1$
    d. Gambar piksel di (x,y)
    e. Untuk setiap $e \geq 0$
        - Jika ya hitung $e = e + d2$ dan $x = x + 1$
        - Jika tidak hitung $e = e + d1$ dan $y = y + 1$
    f. Hitung $x = x + 1$
    g. Jika $x \geq x2$
        - Jika ya stop perhitungan
        - Jika tidak kembali ke langkah 4

2. Algoritma Bresenham untuk dx < dy
    a. Tentukan 2 titik yang akan dihubungkan dalam pembentukan garis.
    b. Tentukan salah satu titik disebelah kiri sebagai titik awal, yaitu (X0,Y0) dan titik lainnya sebagai titik akhir (X1,Y1).
    c. Hitung :

- $Dx = x2 - x1$
- $Dy = y2 - y1$
- $d1 = 2 * Dy$
- $d2 = 2 * Dy - 2 * Dx$
- $e = d1 - dy$
- $x = x1$
- $y = y1$

d. Gambar piksel di (x,y)

e. Untuk setiap $e \geq 0$

- Jika ya hitung $e = e + d2$ dan $x = x + 1$
- Jika tidak hitung $e = e + d1$ dan $x = x$

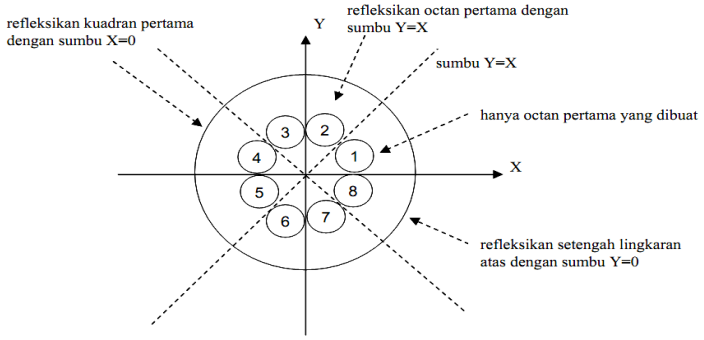
f. Hitung $y = y + 1$

g. Jika $y \geq y2$

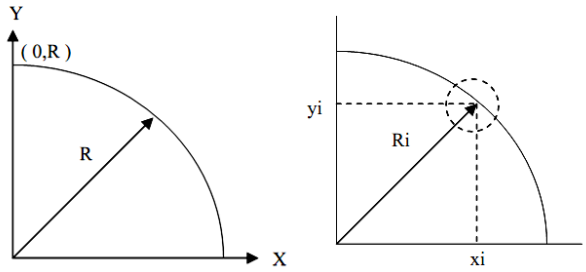
- Jika ya stop perhitungan
- Jika tidak kembali ke langkah 4

2.4.2 Algoritma Lingkaran Bresenham

Untuk menggambar sebuah lingkaran hanya diperlukan menggambar titik-titik pada oktan pertama saja, sedangkan titik-titik pada kuadran lain dapat diperoleh dengan mencerminkan titik-titik pada oktan pertama. Dari gambar dibawah ini titik pada oktan pertama dapat dicerminkan melalui sumbu Y = X untuk memperoleh titik-titik pada oktan kedua dari kuadran pertama. Titik-titik pada kuadran pertama dicerminkan melalui sumbu X = 0 untuk memperoleh titik-titik pada kuadran kedua.



Gambar 2.12 Gambar Lingkaran Dengan Merefleksikan Oktan Pertama



Gambar 2.13 Kuadran Pertama Dari Sebuah Lingkaran

Pada *gambar 2.13* Ri merupakan jarak dari titik (xi,yi) ke pusat lingkaran. Jarak titik (xi,yi) di lingkaran ke pusat lingkarat dapat dihitung dengan cara sebagai berikut:

$$Ri = \left|\sqrt{x_i^2 + y_i^2}\right| \; ; \; Ri^2 = \left|\sqrt{x_i^2 + y_i^2}\right| \; ; \; Ri^2 = |R^2|$$

Maka algoritma lingkaran bresenham:

$x_i = 0$ dan $y_i = R$

Ulangi sampai $y_i = 0$

Hitung Δ_i

Gambar titik di (x_i, y_i)

Jika $\Delta_i < 0$maka hitung δ

Jika $\delta \leq 0$ maka $x_i = x_i + 1$

Jika $\delta > 0$ maka $x_i = x_i + 1$ dan $y_i = y_i - 1$

Jika $\Delta_i > 0$ maka hitung δ'

Jika $\delta' \leq 0$ maka $y_i = y_i - 1$

Jika $\delta' > 0$ maka $x_i = x_i + 1$ dan $y_i = y_i - 1$

Jika $\Delta_i = 0$ maka $x_i = x_i + 1$ dan$y_i = y_i - 1$

Dengan rumus:

$\Delta_i = (x_i + 1)^2 + (y_i - 1)^2 - R^2$

$\delta = |(x_i + 1)^2 + (y_i)^2 - R^2| - |(x_i + 1)^2 + (y_i - 1)^2 - R^2|$

$\delta' = |(x_i)^2 + (y_i - 1)^2 - R^2| - |(x_i + 1)^2 + (y_i - 1)^2 - R^2|$

2.5 G-code

G-code, atau kode persiapan adalah fungsi kontrol numerik dalam bahasa pemrograman. *G-code* adalah kode posisi alat dan melakukan pekerjaan yang sebenarnya. Bahasa pemrograman *Numerical Control* (NC) adalah informal ckadang disebut *g-code*. Namun dalam kenyataannya, *g-code* ini hanya sebagian dari NC-bahasa pemrograman yang mengendalikan NC dan peralatan mesin CNC. *G-code* juga disebut kode persiapan, dan kata apapun dalam program CNC yang diawali dengan huruf 'G'. Umumnya ini adalah kode mengatakan alat mesin jenis tindakan untuk melakukan, seperti:

- Cepat bergerak
- Dikontrol *feed* bergerak dalam garis lurus atau busur
- Serangkaian langkah yang dikendalikan pakan akan mengakibatkan sebuah lubang tidak merasa bosan, sebuah benda memotong (diarahkan) ke dimensi tertentu, atau bentuk profil dekoratif ditambahkan ke pinggir sebuah benda.
- Set alat informasi seperti *offset*.

G-file kode dapat dihasilkan oleh perangkat lunak. Aplikasi tersebut biasanya menggunakan penerjemah disebut *post-prosesor* untuk kode output dioptimalkan untuk jenis mesin tertentu atau keluarga. *Post-prosesor* sering pengguna dapat diedit untuk memungkinkan penyesuaian lebih lanjut, jika perlu. Perangkat lunak tersebut harus disesuaikan untuk tiap jenis alat mesin yang akan digunakan untuk program ini. Berikut ini ulasan tentang *g-code* dan *m-code* pada mesin CNC. Artian kode “G” adalahmenyatakan gerakan, jenis-jenis *g-code* di kelompokan menjadi 2 kelompok.

- Jenis basic/dasar meliputi:
  G00: gerakan cepat tanpa pemakanan benda kerja (bergerak lurus)
  G01: gerakan memotong/pemakanan benda kerja (bergerak lurus)
  G02: gerakan memotong melingkar searah jarum jam
  G03: gerakan memotong melingkar berlawanan arah jarum jam
- Jenis lanjut/siklus meliputi:
  Gabungan langkah G00 dan G01 yang disedehanakan
  G90: sistem kordinat absolut
  G91: sistem kordinat incrimental

Tabel 2.4: Pengertian Dari Kode *G-Code*

NO	Kode	Penjelasan
1	G00	Pengeposisian bebas
2	G01	Interpolasi Lurus (gerak pemakanan)
3	G02	Interpolasi melingkar searah jarum jam (CW)
4	G03	Interpolasi melingkar berlawanan arah jarum jam (CCW)
5	G04	Program berhenti pada waktu tertentu
6	G10	Data program dapat di-input
7	G15	Pembatalan perintah koordinat polar
8	G16	Perintah koordinat polar
9	G17	Interpolasi helical
10	G20	Konversi satuan inchi (british)
11	G21	Konversi satuan mm (metric)
12	G28	Pengembalian posisi referensi
13	G31	perintah skip (melangkahi)
14	G33	Pembuatan ulir (Threading cutting)
15	G40	Cancel kompensasi cutter
16	G43, G44	Kompensasi panjang tool positif(G43), Negatif (G44)
17	G45	Menaikkan offset tool
18	G46	Menurunkan offset tool
19	G47	Menaikkan ganda offset tool
20	G48	Menurunkan ganda offset tool
21	G49	Pembatalan kompensasi panjang tool
22	G52	Penyatuan system koordinat local
23	G54, G55, G56, G57	Sistem koordinat workpiece
24	G60	Pengeposisian arah tunggal
25	G63	Pengerjalaan Tapping (ulir dalam)
26	G64	Pengerjaan pemotongan
27	G74	Menghitung putaran Tapping

28	G76	Pengerjaan proses canned cycle
29	G80	Pembatalan pengerjaan siklus
30	G81	Pengoperasian eksernal atau putaran drilling
31	G82	Keliling counter boring
NO	Kode	Penjelasan
32	G83	Peck drilling cycle
33	G84	Pengerjaan keliling Tapping
34	G85	Pengerjaan keliling boring
35	G90	Perintah system koordinat absolute
36	G91	Perintah system koordinat incremental
37	G94	Penentuan asutan pemakanan dalam (inchi/menit)
38	G95	Penentuan asutan pemakanan dalam (inchi/putaran)
39	G96	Kecepatan potong permukaan konstan
40	G98	Kembali ke titik initial di sebuah siklus
41	G99	Kembali ke titik R di sebuah siklus

Tabel 2.5: Kode Penunjang Kode Utama *G-Code*

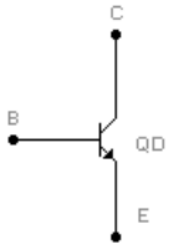
NO	Kode	Penjelasan
1	O	Nomor program
2	N	Nomor urut blok program
3	G	Khusus untuk kode perpindahan (lurus,(busur/melengkung dll)
4	X,Y,Z, U,V,W ,A,B,C	(kata ukuran) perintah perpindahan sumbu koordinat.Nilai ini berkaitan dengansystem inkremental
5	I,J,K	(kata ukuran) koordinat titik lengkung
6	R	(kata ukuran) jari-jari lengkungan
7	F	Kode pemakanan / rate of feed (per menit atau per putaran)
8	S	Kecepatan spindle (rpm)
9	T	Nomor tool
10	M	(kode tambahan) kontrol ON/OFF akan tool mesin

11	B	Table indexing, dll
12	D,H	Nomor offset
13	P,X	Lama waktu jeda (detik)
14	P	Nomor program pembantu (subprogram)
15	P,Q	Parameter keliling pengkalengan (canned cycle)

2.6 Transistor Darlington

Transistor adalah alat semikonduktor yang dipakai sebagai penguat, sebagai sirkuit pemutus dan penyambung (switching), stabilisasi tegangan, modulasi sinyal atau sebagai fungsi lainnya. Transistor dapat berfungsi semacam kran listrik, dimana berdasarkan arus inputnya (BJT) atau tegangan inputnya (FET), memungkinkan pengaliran listrik yang sangat akurat dari sirkuit sumber listriknya.

Pada umumnya, transistor memiliki 3 terminal, yaitu Basis (B), Emitor (E) dan Kolektor (C). Tegangan yang di satu terminalnya misalnya Emitor dapat dipakai untuk mengatur arus dan tegangan yang lebih besar daripada arus input Basis, yaitu pada keluaran tegangan dan arus output Kolektor.

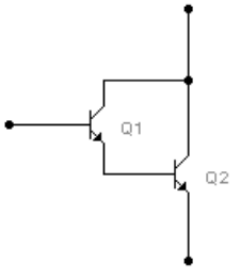


Gambar 2.14 Rangkaian Transistor

Transistor merupakan komponen yang sangat penting dalam dunia elektronik modern. Dalam rangkaian analog, transistor digunakan dalam amplifier (penguat). Rangkaian analog melingkupi pengeras suara, sumber listrik stabil (stabilisator) dan penguat sinyal radio. Dalam rangkaian-rangkaian digital, transistor digunakan sebagai saklar berkecepatan tinggi. Beberapa transistor juga dapat dirangkai sedemikian rupa sehingga berfungsi sebagai logic gate, memori dan fungsi rangkaian-rangkaian lainnya.

Pada pembahasan makalah ini akan lebih dijelaskan mengenai transistor darlington yang mana merupakan transistor pada terminal emitor yang bertegangan tinggi.

Transistor Darlington adalah rangkaian elektronika yang terdiri dari sepasang transistor bipolar (dwi kutub) yang tersambung secara tandem (seri). Sambungan seri seperti ini dipakai untuk mendapatkan penguatan (gain) yang tinggi, karena hasil penguatan pada transistor yang pertama akan dikuatkan lebih lanjut oleh transistor kedua. Penguatan arus listrik atau gain dari rangkaian transistor Darlington ini sering dituliskan dengan notasi β atau hFE.

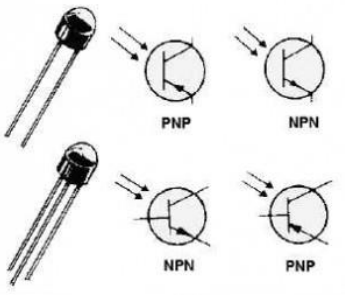


Gambar 2.15Rangkaian Transistor Darlington

Rangkaian transistor Darlington ditemukan pertama kali oleh Sidney Darlington yang bekerja di Laboratorium Bell di Amerika Serikat. Jenis rangkaian hasil penemuannya ini telah mendapatkan hak paten, dan banyak dipakai dalam pembuatan Sirkuit terpadu (IC atau Integrated Circuits) chip. Jenis rangkaian yang mirip dengan transistor Darlington adalah rangkaian pasangan Sziklai yang terdiri dari sepasang transistor NPN dan PNP. Rangkaian Sziklai sering dikenal sebagai rangkaian 'Complementary Darlington' atau 'rangkaian kebalikan dari Darlington'.

2.7 Phototransistor

Phototransistor merupakan jenis transistor yang bias basisnya berupa cahaya infra merah. Besarnya arus yang mengalir di antara kolektor dan emitor sebanding dengan intensitas cahaya yang diterima *phototransistor* tersebut. Simbol dari *phototransistor* ditunjukan pada gambar berikut.



Gambar 2.16Simbol Dan Bentuk Dari Phototransistor

2.7.1 Fungsi sensor *phototransistor*

Phototransistor sering digunakan sebagai saklar terkendali cahaya infra merah, yaitu memanfaatkan keadaan jenuh (saturasi) dan mati (*cut-off*) dari *phototransistor* tersebut. Prisip kerja *phototransistor* untuk menjadi saklar yaitu saat pada basis menerima cahaya infra merah maka *phototransistor* akan berada pada keadaan jenuh (saturasi) dan saat tidak menerima cahaya infra merah *phototransistor* berada dalam kondisi mati (*cut-off*) Stuktur *phototransistor* mirip dengan transistor *bipolar* (*bipolar junctoin transistor*). Pada daerah basis dapat dimasuki sinar dari luar melalui suatu celah transparan dari luar kamasan taransistor. Celah ini biasanya dilindungi oleh suatu lensa kecil yang memusatkan sinar di tepi sambungangan basis emitor.

2.7.2 Prinsip kerja sensor *phototransistor*

Sambungan antara basis dan kolektor, dioperasikan dalam catu balik dan berfungsi sebagai fotodioda yang merespon masuknya sinar dari luar. Bila tak ada sinar yang masuk, arus yang melalui sambungan catu balik sama dengan nol. Jika sinar dari energi *photon* cukup dan mengenai sambungan catu balik, penambahan pasangan *hole* dan elektron akan terjadi dalam *depletion region*, menyebabkan sambungan menghantar. Jumlah pasangan *hole* dan elektron yang dibangkitkan dalam sambungan akan sebanding dengan intensitas sinar yang mengenainya. Sambungan antara basis emitor dapat dicatu maju, menyebabkan piranti ini dapat difungsikan sebagai transistor *bipolar* konvensional.