

BAB II Tinjauan Pustaka

2.1 Kontroler

Kontroler adalah elemen yang berfungsi untuk membandingkan nilai masukan (*set point*) dengan nilai keluaran (*output*), menghitung berapa banyak koreksi yang diperlukan, dan mengeluarkan sinyal koreksi sesuai dengan hasil perhitungan tadi. (Gunterus, F:1977). Dalam sistem pengaturan dikenal banyak sekali macam kontroler, antara lain kontroler proporsional (P), kontroler PI (Proporsional Integral), PID (Proporsional, Integral, dan Differensial), dan sebagainya. Masing-masing kontroler mempunyai karakteristik masing-masing.

2.1.1 Kontroler Proporsional

Pada kontroler proporsional, keluarannya selalu sebanding dengan masukannya. Kontroler proporsional selalu memerlukan *error* untuk menghasilkan *output*. Hal ini menyebabkan terjadinya *offset (error steady state)* pada pemakaian kontroler proporsional.

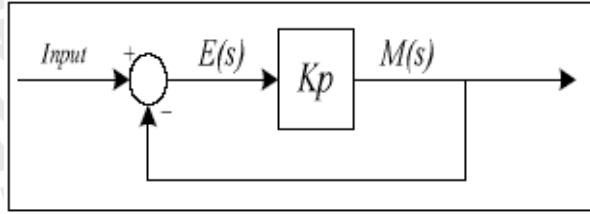
Hubungan antara keluaran kontroler $m(t)$ dan sinyal *error* $e(t)$ dari kontroler Proporsional adalah :

$$m(t) = K_p e(t) \dots\dots\dots (2.1)$$

atau dalam transformasi Laplace :

$$\frac{M(s)}{E(s)} = K_p \dots\dots\dots (2.2)$$

Dimana K_p adalah *Gain* (penguatan proporsional), $E(s)$ adalah *error* (selisih antara *input* dengan *output*) dan $M(s)$ adalah *output* kontroler. Sedangkan *Input/Set Point* merupakan nilai tertentu yang ingin dicapai dari pengendalian. Secara umum, prinsip kerja kontroler proporsional ditunjukkan dalam Gambar 2.1.



Gambar 2.1 Blok diagram kontroler Proporsional

Sumber : Ogata, 1997

2.1.2 Kontroler Integral

Untuk menghilangkan *offset* yang terjadi akibat pemakaian kontroler Proporsional, maka digunakanlah kontroler Integral. Kontroler Integral dapat menghasilkan *output* walaupun sudah tidak ada *input*. Namun, reaksi dari kontroler ini sangat lambat.

Pada kontroler Integral, harga keluaran kontroler $m(t)$ diubah dengan laju yang sebanding dengan sinyal *error* $e(t)$, sehingga

$$m(t) = K_i \int e(t) dt \dots\dots\dots (2.3)$$

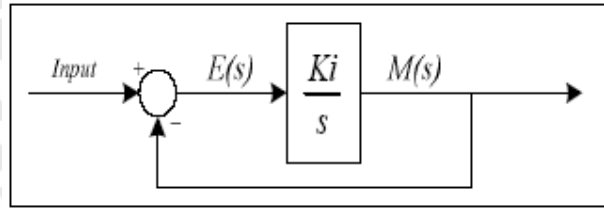
K_i adalah *Gain* (penguatan integral) dan merupakan konstanta yang dapat diatur. Dalam transformasi Laplace dapat dinyatakan :

$$\frac{M(s)}{E(s)} = \frac{K_i}{s} \dots\dots\dots (2.4)$$

Dimana :

$$K_i = \frac{K_p}{T_i} \dots\dots\dots (2.5)$$

Dengan T_i adalah waktu integral (waktu yang dibutuhkan kontroler integral untuk mencapai *output* sama dengan *input*, dengan satuan *minute/repeat*). Secara umum prinsip kerja kontroler integral ditunjukkan dalam Gambar 2.2.



Gambar 2.2 Blok diagram kontroler Integral.

Sumber : Ogata, 1997

2.1.3 Kontroler Differensial

Hubungan sinyal kontroler $m(t)$ dengan sinyal *error* $e(t)$ dari kontroler Differensial dapat dinyatakan dengan :

$$m(t) = K_d \frac{de(t)}{dt} \dots\dots\dots (2.6)$$

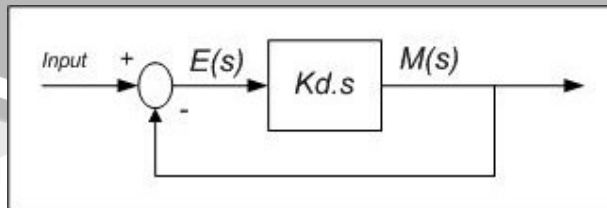
atau dalam transformasi Laplace dinyatakan dengan :

$$\frac{M(s)}{E(s)} = K_d s \dots\dots\dots (2.7)$$

di mana $K_d = K_p T_d$

Dengan K_d adalah *Gain* (penguatan differensial), dan T_d adalah waktu derivatif (waktu yang dibutuhkan kontroler differensial untuk mencapai *output* sebanding dengan perubahan *input*, dengan satuan detik atau menit).

Keluaran kontroler differensial merupakan turunan fungsi *input*. Kontroler differensial tidak dapat digunakan sendirian, karena unsur D tidak akan mengeluarkan *output* bila tidak terjadi perubahan *input*. Oleh karena itu, kontroler ini harus selalu digunakan bersama kontroler Proporsional, Integral, atau keduanya. Secara umum prinsip kerja kontroler differensial ditunjukkan dalam Gambar 2.3.



Gambar 2.3 Blok diagram kontroler Differensial.

Sumber : Ogata, 1997

2.1.4 Kontroler Proporsional Integral

Terjadinya *offset* pada penggunaan kontroler P menyebabkan perlunya mencari alternatif kontroler yang lain. Maka digunakanlah kontroler I. Namun, karena kelambatan reaksinya, maka kontroler I biasanya dipakai bersama dengan kontroler P menjadi kontroler PI.

Dengan demikian, maka kontroler PI dapat menghasilkan respon yang lebih cepat dari kontroler I, dan juga dapat menghilangkan *offset* yang ditimbulkan oleh kontroler P.

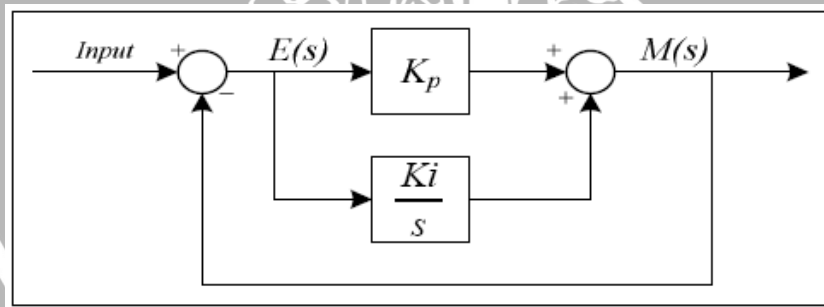
Aksi kontroler PI dinyatakan dalam persamaan :

$$m(t) = K_p e(t) + \frac{K_p}{T_i} e(t) dt \dots\dots\dots (2.8)$$

atau dalam transformasi Laplace dinyatakan dengan :

$$\frac{m(s)}{E(s)} = K_p \left(1 + \frac{1}{T_i s}\right) \dots\dots\dots (2.9)$$

Dalam hal ini K_p menyatakan penguatan dan T_i menyatakan waktu integral. Baik K_p maupun T_i dapat diatur. Kontroler Proporsional Integral (PI) dapat dibentuk dari gabungan kontroler Proporsional dan kontroler Integral. Secara umum prinsip kerja kontroler Proporsional Integral ditunjukkan dalam Gambar 2.4.



Gambar 2.4 Blok diagram kontroler Proporsional Integral

Sumber : Ogata, 1997

2.1.5 Kontroler Proporsional Integral Differensial

Gabungan aksi kontrol Proporsional, Integral dan Differensial mempunyai keunggulan dibandingkan dengan masing-masing dari tiga aksi kontrol tersebut, kontroler PID ini dapat dinyatakan sebagai berikut:

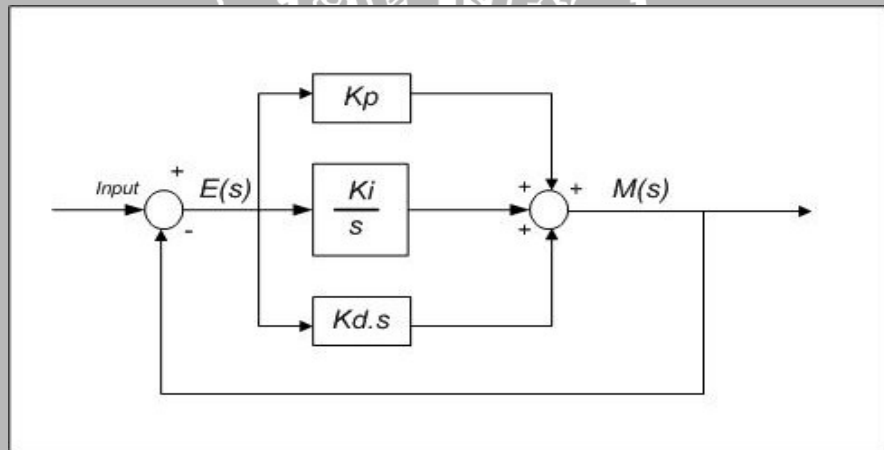
$$m(t) = K_p e(t) + \frac{K_p}{T_i} e(t) dt + K_p T_d \frac{de(t)}{dt} \dots\dots\dots (2.10)$$

dalam transformasi Laplace dinyatakan sebagai berikut :

$$\frac{M(s)}{E(s)} = K_p (1 + \frac{1}{T_i s} + T_d s) \dots\dots\dots (2.11)$$

Keuntungan penggunaan kontroler PID adalah mempercepat reaksi sistem, menghilangkan *error steady state*, dan mendapatkan energi ekstra di saat awal terjadinya perubahan *load* (beban). (Gunterus, 1977).

Secara umum prinsip kerja kontroler integral ditunjukkan dalam Gambar 2.5.

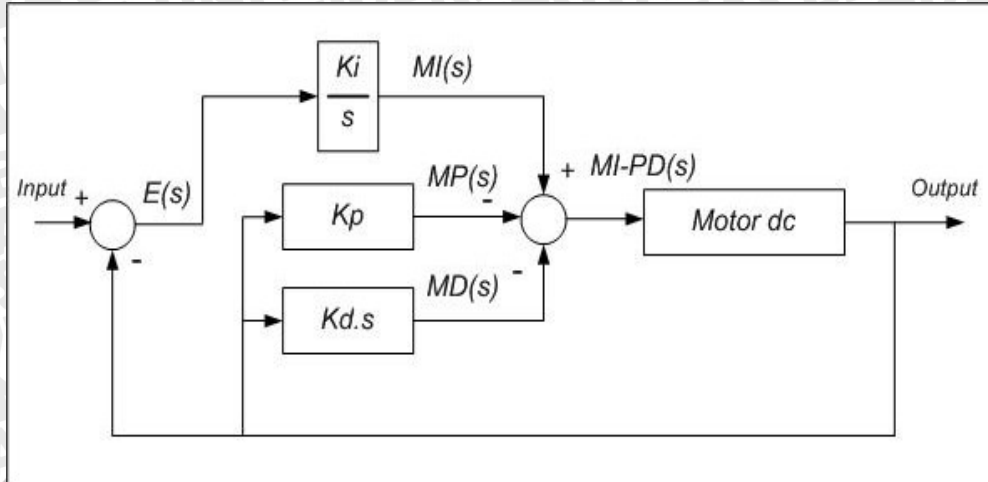


Gambar 2.5 Blok diagram kontroler PID

Sumber : Ogata, 1997

Kelemahan dari kontroler PID adalah apabila diberikan *input* unit step akan menimbulkan sinyal pulsa (*set point kick*) pada keluarannya karena pengaruh aksi differensialnya. Selain itu juga kerap dijumpai proses yang masih mengandung *overshoot* yang besar walaupun sudah diterapkan kontroler PID pada sistem tersebut. (Departemen Teknik Fisika Institut Teknologi Bandung, 2003)

2.1.6 Kontroler Integral – Proporsional Differensial (I-PD)



Gambar 2.6 Blok diagram kontroler I-PD

Sumber : Departemen Teknik Fisika Institut Teknologi Bandung, 2003

Gambar 2.6 merupakan blok diagram sistem kontrol I-PD. Pada kontroler I-PD, kontroler Integral bereaksi terhadap sinyal kesalahan dari motor dc, $E(s)$. Sedangkan kontroler Proporsional dan Differensial bereaksi pada sinyal keluaran motor dc itu sendiri.

$M_{I-PD}(s)$ merupakan sinyal referensi *input* ke motor dc. Pada skema Integral – Proporsional Differensial ini, $M_{I-PD}(s)$ adalah selisih antara keluaran kontroler Integral $M_I(s)$ dengan penjumlahan dari kontroler Proporsional $M_P(s)$ dan kontroler Differensial $M_D(s)$. Karena kontroler Integral dan kedua buah kontroler lainnya bereaksi pada sinyal yang berbeda (*output* sistem), maka sinyal $M_{I-PD}(s)$ tidak mengalami perubahan yang kasar, sehingga *output* pada motor dc tidak mengalami perubahan yang kasar pula, serta *overshoot* dapat dihindari. Untuk masukan unit step yang besar, lonjakan pada motor dc dapat dihindari, tidak seperti pada kontroler PID.

Begitupun dengan fenomena *set point kick* yang kerap muncul pada kontroler PID, sebagai kompensasi diperbesarnya komponen Differensial dengan tujuan meminimalisir *overshoot*. Pada kontroler I-PD, hanya kontroler Integral yang bereaksi terhadap sinyal kesalahan tanpa disertai aksi dari kontroler Differensial. Sehingga fenomena tersebut dapat dihindari. (Departemen Teknik Fisika Institut Teknologi Bandung, 2003).

2.2 Kontrol digital.

Pengendali digital dapat berupa komputer atau mikroprosesor atau mikrokontroler yang berisi algoritma kendali. Komputer adalah suatu peralatan digital yang beroperasi dalam waktu rinci, maka hanya dapat menerima informasi pada nilai diskrit dari waktu. Sistem waktu diskrit adalah sistem yang dapat dimodelkan oleh persamaan beda. (Charles L. Philips, Royce D. Harbor, 1996).

Operasi komputer harus linier dan tidak berubah terhadap waktu. Jadi dapat diasumsikan, masukan $t=0$ menghasilkan keluaran pada $t=0$ dan masukan pada $t=T$ menghasilkan keluaran $t=T$. Dimisalkan pada $t=0$ masukan adalah $e(0)$ dan keluaran adalah $m(0)$. Persamaan dapat diekspresikan sebagai :

$$m(0) = b_0 e(0) \dots\dots\dots(2.12)$$

dimana b_0 adalah suatu konstanta. Selanjutnya $e(0)$ dan $m(0)$ disimpan di memori komputer. Jadi, $m(T)$ dapat menjadi fungsi $e(0)$, $m(0)$, dan $e(T)$. Persamaan dapat mempunyai bentuk :

$$m(T) = b_0 e(T) + b_1 e(0) - a_1 m(0) \dots\dots\dots(2.13)$$

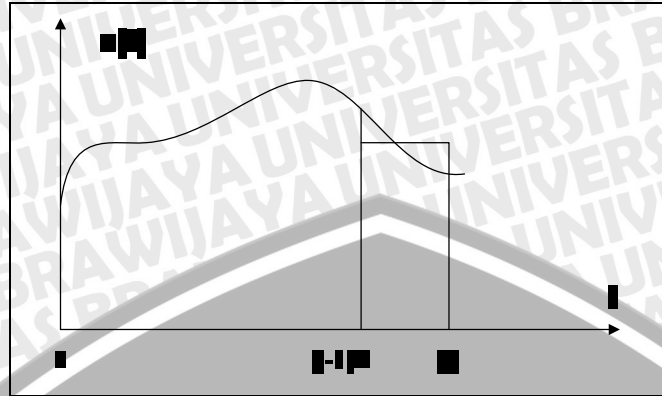
Persamaan (2.13) disebut persamaan beda.

2.2.1 Kontroler Proporsional Integral Differensial Digital.

Persamaan 2.10. menyatakan fungsi alih kontroler PID Analog. Apabila kontroler PID dibuat menjadi PID digital, maka perlu dilakukan transformasi bentuk kontinyu menjadi bentuk diskrit. Prosedur pengubahan ke dalam bentuk persamaan diskrit dapat dilakukan dengan berbagai metode. Salah satu metode yang dapat digunakan adalah aturan trapezoidal dan aturan simpson. Aturan trapezoidal diilustrasikan dalam Gambar 2.7. Berdasarkan metode Trapezoidal, maka persamaan integrasi numerik untuk persamaan 2.10. ditunjukkan dalam persamaan 2.14, sedangkan persamaan differensiasi numerik dapat dilihat dalam persamaan 2.15.

$$k_i \int_0^{kT} e(t) dt = k_i T / 2 \sum_{j=1}^k \{e[(j-i)T] + e(jT)\} \dots\dots\dots(2.14)$$

$$k_d \frac{de}{dt} = \frac{k_d}{T} \{e(kT) - e[(k-1)T]\} \dots\dots\dots(2.15)$$



Gambar 2. 7. Metode trapezoidal

Sumber : D'Souza,1988

Apabila persamaan 2.14. dan 2.15. digabungkan, akan diperoleh persamaan kontroler PID seperti ditunjukkan oleh persamaan 2.16.

$$m(kT) = k_p e(kT) + k_i T / 2 \sum_{j=1}^k \{e[(j-1)T] + e(jT)\} + k_d / T [e(kT) - e[(k-1)T]] \dots\dots (2.16)$$

dengan mengubah indeks k menjadi k-1 dan mengurangkan dengan persamaan 2.16, maka akan diperoleh:

$$m(k) - m(k-1) = k_p [e(k) - e(k-1)] + k_i T / 2 [e(k) + e(k-1)] + k_d / T [e(k) - 2e(k-1) + e(k-2)] \dots\dots (2.17)$$

- $m(k)$ = input motor saat ini
- $m(k-1)$ = input motor sebelum saat ini
- $e(k)$ = error saat ini
- $e(k-1)$ = error sebelum saat ini
- $e(k-2)$ = error sebelum sebelum saat ini

Dimisalkan :

$$a_0 = k_p + k_i T / 2 + k_d / T \dots\dots\dots(2.18)$$

$$a_1 = -k_p + k_i T / 2 - 2k_d / T \dots\dots\dots(2.19)$$

$$a_2 = k_d / T \dots\dots\dots(2.20)$$

maka didapatkan :

$$m(k) - m(k-1) = a_0 e(k) + a_1 e(k-1) + a_2 e(k-2) \dots\dots\dots(2.21)$$

Persamaan 2.21. dapat digunakan sebagai algoritma sistem pengendalian Proporsional Integral Differensial secara digital.

2.2.2 Kontroler Integral – Proporsional Differensial (I-PD) Digital.

Pengendali I-PD dapat disusun secara matematis menjadi I-PD diskrit.

Persamaan pengendali I-PD sebagai berikut :

$$m(t) = \frac{K_p}{T_i} e(t) dt - \left(K_p o(t) + K_p T_d \frac{do(t)}{dt} \right) \dots\dots\dots(2.22)$$

bila diubah ke dalam bentuk diskrit menjadi :

$$m(kT) = k_i T / 2 \sum_{j=1}^k \{e[(j-1)T] + e(jT)\} - k_p o(kT) - k_d / T [o(kT) - o[(k-1)T]] \dots\dots(2.23)$$

dengan mengurangkan indeks k dengan k-1 didapatkan :

$$m(k) - m(k-1) = k_i T / 2 [e(k) + e(k-1)] - \{k_p [o(k) - o(k-1)] + k_d / T [o(k) - 2o(k-1) + o(k-2)]\} \dots(2.24)$$

- $m(k)$ = input motor saat ini
- $m(k-1)$ = input motor sebelum saat ini
- $e(k)$ = error saat ini
- $e(k-1)$ = error sebelum saat ini
- $o(k)$ = output saat ini
- $o(k-1)$ = output sebelum saat ini
- $o(k-2)$ = output sebelum sebelum saat ini

dimisalkan :

$$b_0 = k_i T / 2 \dots\dots\dots(2.25)$$

$$b_1 = k_p + k_d / T \dots\dots\dots(2.26)$$

$$b_2 = k_p + 2 k_d / T \dots\dots\dots(2.27)$$

$$b_3 = k_d / T \dots\dots\dots(2.28)$$

maka didapatkan :

$$m(k) - m(k-1) = b_0 [e(k) + e(k-1)] - b_1 o(k) + b_2 o(k-1) - b_3 o(k-2) \dots(2.29)$$

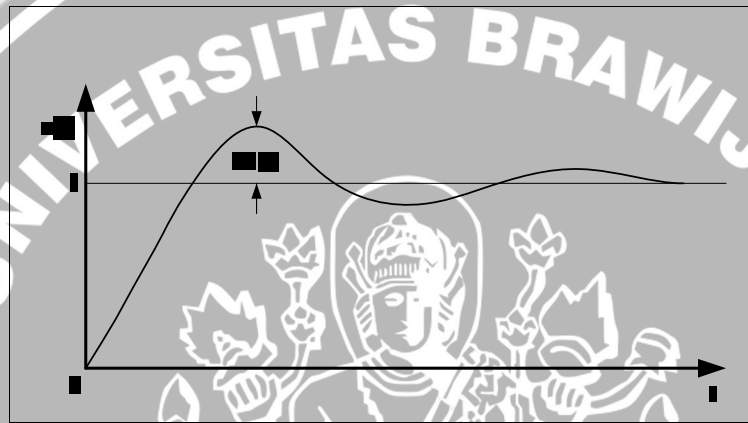
Persamaan 2.29. dapat digunakan sebagai algoritma sistem pengendalian I-PD secara digital.

2.3 Perancangan Kontroler Menggunakan Metode Ziegler-Nichols.

Ziegler dan Nichols mengemukakan aturan-aturan untuk menentukan nilai dari gain proporsional K_p , waktu integral T_i , dan waktu derivatif T_d berdasarkan

karakteristik respon transien dari plan yang diberikan. Penentuan parameter kontroler PID atau penalaan kontroler PID tersebut dapat dilakukan dengan bereksperimen dengan plan. (Ogata, K., 1997)

Terdapat dua metode yang disebut dengan aturan penalaan Ziegler-Nichols, pada kedua metode tersebut memiliki tujuan yang sama yaitu untuk mencapai 25% *maximum overshoot* pada respon unit step, seperti ditunjukkan pada Gambar 2.8.

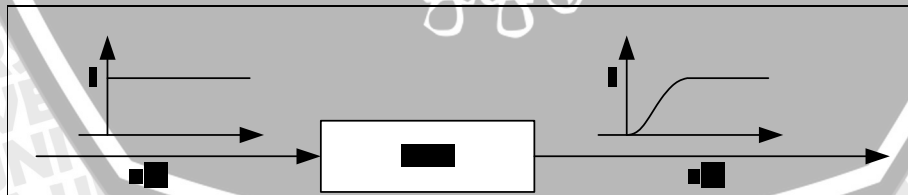


Gambar 2. 8. Kurva Respon Unit Step yang Menunjukkan 25% *Maximum Overshoot*

Sumber: Ogata, K., 1997

2.3.1 Metode Pertama

Metode pertama atau sering disebut metode kurva reaksi, respon dari plan dapat diperoleh secara eksperimental dengan masukan berupa unit step, seperti yang ditunjukkan pada Gambar 2.9.

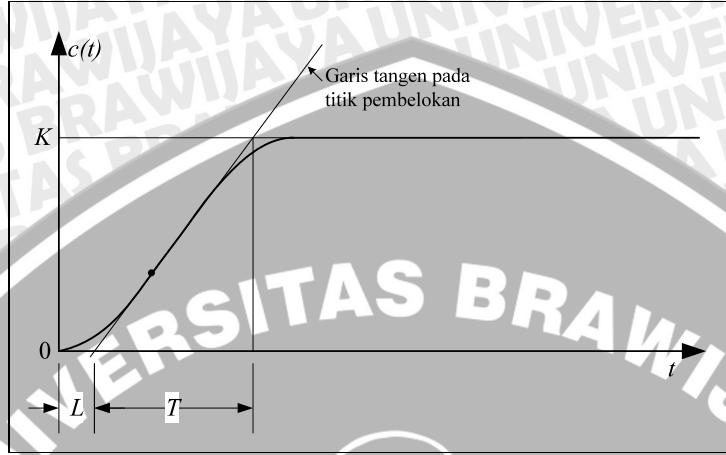


Gambar 2. 9. Respon Plant Terhadap Masukan Berupa Unit Step

Sumber: Ogata, K. 1997

Jika dalam plan tersebut terdapat integrator atau *dominan complex-conjugate poles*, maka kurva respon unit step berbentuk seperti huruf S, seperti

dalam Gambar 2.10. jika respon tidak memberikan bentuk kurva S, maka metode ini tidak berlaku. (Ogata, K., 1997).



Gambar 2. 10. Kurva Respon yang Berbentuk S
 Sumber: Ogata, K. 1997

Kurva berbentuk S tersebut dapat dikarakteristikan menjadi dua konstanta yaitu waktu tunda L dan konstanta waktu T . Waktu tunda dan konstanta waktu ditentukan dengan menggambar sebuah garis tangen pada titik pembelokan dari kurva S, dan menentukan perpotongan antara garis tangen dengan sumbu waktu t dan sumbu $c(t) = K$, seperti yang telah ditunjukkan dalam Gambar 2.10. Fungsi alih $C(s)/U(s)$ dapat dilakukan pendekatan dengan sistem orde satu dengan persamaan sebagai berikut:

$$\frac{C(s)}{U(s)} = \frac{K e^{-Ls}}{T s + 1} \dots\dots\dots (2.30)$$

Ziegler dan Nichols menyarankan untuk menentukan nilai-nilai dari K_p , T_1 dan T_d berdasarkan pada formula yang ditunjukkan dalam Tabel 2.1. (Ogata, K., 1997)

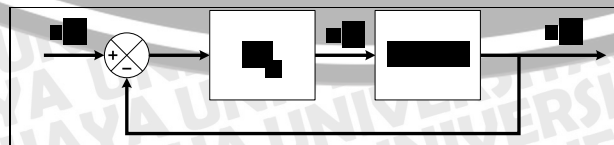
Tabel 2. 1. Aturan penalaan Ziegler-Nichols berdasarkan respon unit step dari plan

Tipe kontroler	K_p	T_1	T_d
P	$\frac{T}{L}$	∞	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	$2L$	$0.5L$

Sumber: Ogata, K. 1997

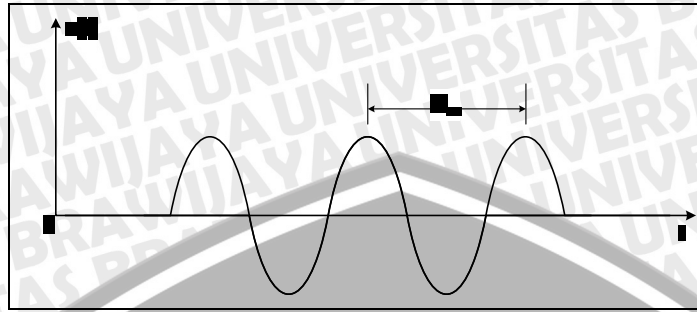
2.3.2 Metode Kedua

Dalam metode kedua ziegler-nichols, mula-mula yang dilakukan adalah membuat $T_1 = \infty$ dan $T_d = 0$. Kemudian hanya dengan menggunakan tindakan kontrol proporsional, dengan *set point* tertentu ($r(t)$), yang menghasilkan sinyal kontrol $u(t)$ sebagai *input* ke *plant* (bagian dari sistem kontrol yang mengandung variabel-variabel terkontrol), harga K_p ditingkatkan dari nol ke suatu nilai kritis K_{cr} , disini mula-mula keluaran memiliki osilasi yang berkesinambungan (Jika keluaran tidak memiliki osilasi berkesinambungan untuk nilai K_p manapun yang telah diambil, maka metode ini tidak berlaku). Dari keluaran yang beresilasi secara berkesinambungan, penguatan kritis K_{cr} (acuan nilai penguatan kontroler proporsional untuk menghasilkan keluaran yang beresilasi secara berkesinambungan) dan periode P_{cr} (jarak/waktu antara 2 buah puncak pada keluaran yang beresilasi secara berkesinambungan) dapat ditentukan. Secara lebih jelas ditunjukkan dalam Gambar 2.11 dan 2.12. Ziegler dan Nichols menyarankan penyetapan nilai parameter K_p , T_1 , dan T_d berdasarkan rumus yang diperlihatkan dalam Tabel 2.2.(Ogata, K., 1997)



Gambar 2. 11. Sistem Loop Tertutup dengan Kontroler Proporsional

Sumber: Ogata, K., 1997



Gambar 2. 12. Osilasi Berkesinambungan dengan Periode P_{cr}

Sumber: Ogata, K., 1997

Tabel 2. 2. Aturan Dasar Ziegler-Nichols Berdasarkan Critical Gain K_{cr} dan Critical Period P_{cr}

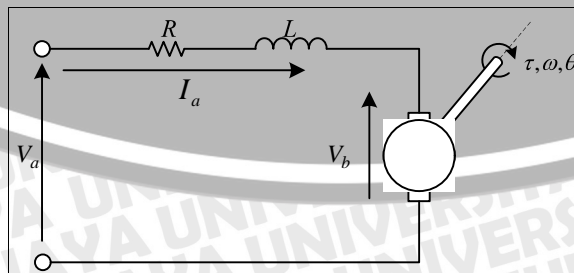
Type Kontroler	K_p	T_i	T_d
P	$0.50 K_{cr}$	∞	0
PI	$0.45 K_{cr}$	$\frac{1}{1.2} P_{cr}$	0
PID	$0.60 K_{cr}$	$0.5 P_{cr}$	$0.125 P_{cr}$

Sumber: Ogata, K., 1997

2.4 Motor DC Magnet Permanen

Motor DC (*direct current*) adalah peralatan elektromagnetik dasar yang berfungsi untuk mengubah tenaga listrik menjadi tenaga mekanik yang desain awalnya telah diperkenalkan oleh Michael Faraday lebih dari seabad lalu.

Rangkaian ekivalen dari sebuah motor DC magnet permanen dapat ditunjukkan seperti dalam Gambar 2.13.



Gambar 2. 13. Rangkaian Ekivalen Motor DC Magnet Permanen

Sumber: Pitowarno, E., 2006

Dengan notasi: V_a = tegangan armatur, I_a = arus motor, R = resistansi armatur, L = iduktansi lilitan armatur, V_b = tegangan induksi balik, τ = torsi motor, ω = kecepatan putar motor, θ = sudut putaran poros motor.

Persamaan tegangan V_a adalah,

$$V_a = L \frac{dI_a}{dt} + R I_a + K_b \omega \dots\dots\dots (2.31)$$

Dengan K_b adalah konstanta yang diukur dari tegangan yang dihasilkan oleh motor ketika berputar tiap satuan kecepatan (volt.det/rad). Magnitud dan polaritas K_b adalah fungsi dari kecepatan angular ω arah arus putar motor. (Pitowarno. 2006).

2.5 Driver Motor

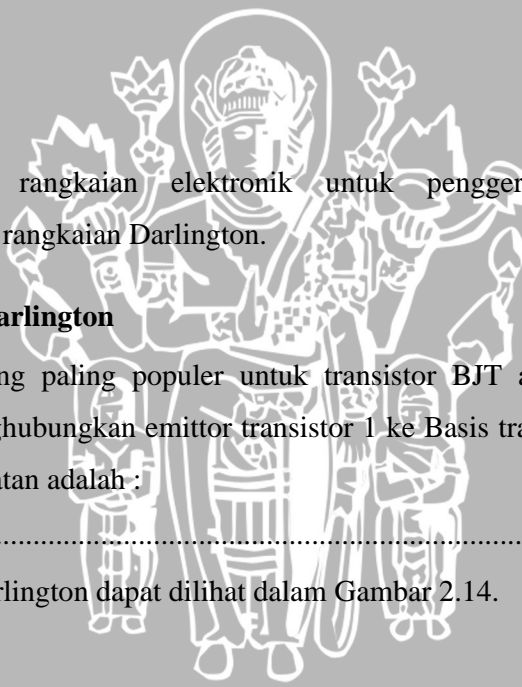
Driver atau rangkaian elektronik untuk penggerak motor DC menggunakan prinsip rangkaian Darlington.

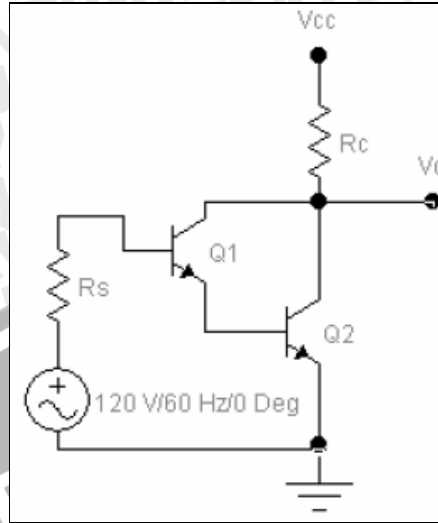
2.5.1 Rangkaian Darlington

Hubungan yang paling populer untuk transistor BJT adalah hubungan darlington yang menghubungkan emittor transistor 1 ke Basis transistor 2. Secara ideal besarnya penguatan adalah :

$$hfe_D = \beta_1 \beta_2 \dots\dots\dots (2.32)$$

Hubungan Darlington dapat dilihat dalam Gambar 2.14.





Gambar 2. 14. Hubungan Darlington

Sumber: Laboratorium Elektronika dan Instrumentasi Fisika ITB

$$K_v = \frac{v_o}{v_i} = \frac{R_o i_o}{R_i i_i} = K_i \frac{R_o}{R_i} \dots \dots \dots (2.33)$$

Dengan penguatan arus

$$K_i = \beta_i \beta_2 \dots \dots \dots (2.34)$$

$$R_o = R_c // \frac{1}{h_{oe2}} \cong R_c \dots \dots \dots (2.35)$$

$$R_i = h_{ie1} + (1 + \beta_i) h_{ie2} \dots \dots \dots (2.36)$$

Oleh karena h_{ie} berada pada emitor Q_1 , maka jika dilihat dari basis Q_1 tampak mempunyai nilai $(1 + \beta_i)$ kalinya. Sehingga:

$$K_{vi} = \frac{v_o}{v_i} = \frac{\beta_i \beta_2 R_c}{h_{ie1} + (1 + \beta_i) h_{ie2}} \dots \dots \dots (2.37)$$

2.6 Sensor

Sensor/switch merupakan suatu peralatan yang dapat mendeteksi suatu obyek tanpa kontak secara fisik. Sensor tersebut adalah suatu peralatan elektronik yang sensitif terhadap gerakan, panas, tekanan, cahaya, suhu dan beberapa macam tipe energi lainnya. Pada teknologi akhir-akhir ini, sensor banyak digunakan untuk memantau secara terus-menerus terhadap perubahan yang terjadi pada sesuatu yang dipantau, yang lebih menuju pada penggunaan sosial seperti pada

automatisasi rumah, teknik *biomedical*, dan untuk penggunaan kesejahteraan sosial lainnya.

Dengan kata lain sensor merupakan suatu pengubah dimana *input*-nya adalah suatu kejadian fisis dan *output*-nya adalah suatu ukuran kuantitatif dari kejadian tersebut. Sensor digunakan dalam desain transduser seperti potensiometer, *Light Depending Resistor* (LDR), *photodiode*, *phototransistor*, *termistor*, saklar *magnetic* dan lainnya.

Dalam rangkaian elektronika sensor selalu memerlukan rangkaian pengkondisi sinyal. Rangkaian pengkondisi sinyal ini untuk melakukan beberapa fungsi yaitu penguatan, penyesuaian impedansi dan pemfilteran.

2.6.1 *Optocoupler*

Optocoupler adalah alat yang dipakai untuk mengkopel cahaya dari suatu sumber ke *detector* tanpa adanya perantara. Oleh karena itu, piranti ini sering disebut dengan *optoisolator/optocoupler*.

Sinyal listrik (arus) pada *input* diubah menjadi sinyal optik dengan menggunakan sumber cahaya (biasanya LED). Sinyal optik tersebut akan diterima oleh detektor untuk diubah kembali menjadi sinyal listrik.

Umumnya *optocoupler* dipakai untuk mengisolasi sinyal listrik yang ada pada *input* dan *output* sehingga dapat digunakan transmisi sinyal antar rangkaian.

2.6.1.1 Prinsip Kerja *Optocoupler*

Optocoupler memiliki sebuah LED yang dihubungkan dengan masukan dan sebuah *photodiode* yang dihubungkan dengan keluaran. Prinsip kerja dari *optocoupler* adalah sebagai berikut:

- Jika antara *photodiode* dan LED dihalangi, maka *photodiode* akan *off* sehingga *output* dari *photodiode* akan berlogika *low*.
- Sebaliknya jika antara *photodiode* dan LED tidak dihalangi, maka *photodiode* akan *on* sehingga *output* dari *photodiode* akan berlogika *high*.

2.7 Sistem Komunikasi Data

Transmisi data merupakan proses pengiriman data dari sumber data ke penerima data melalui media pengiriman tertentu. Keberhasilan sebuah transmisi akan sangat dipengaruhi oleh mutu data yang ditransmisikan dan jenis media transmisi yang digunakan. Dalam sistem komputer, setiap karakter pada data disajikan dalam bentuk sederetan angka biner (binary digit) yang hanya berisi bit 1 atau 0, sedangkan untuk pengiriman data dapat menggunakan beberapa metode pengiriman, yaitu secara serial dan paralel.

2.7.1 Port Paralel

Sistem komunikasi data secara paralel memanfaatkan LPT *Port/Printer Port*, secara fisik berupa DB25-*female* (betina). Jenis *Port* ini mempunyai kecepatan yang lebih tinggi bila dibandingkan dengan jenis *Serial Port* karena data ditransmisikan serempak 8 bit dalam satu siklus mesin. Namun panjang kabel lebih pendek jika dibandingkan dengan *Serial Port*. *Hardware* dari DB25 ditunjukkan dalam Gambar 2.15, detail pin-pinnya dalam Gambar 2.16.

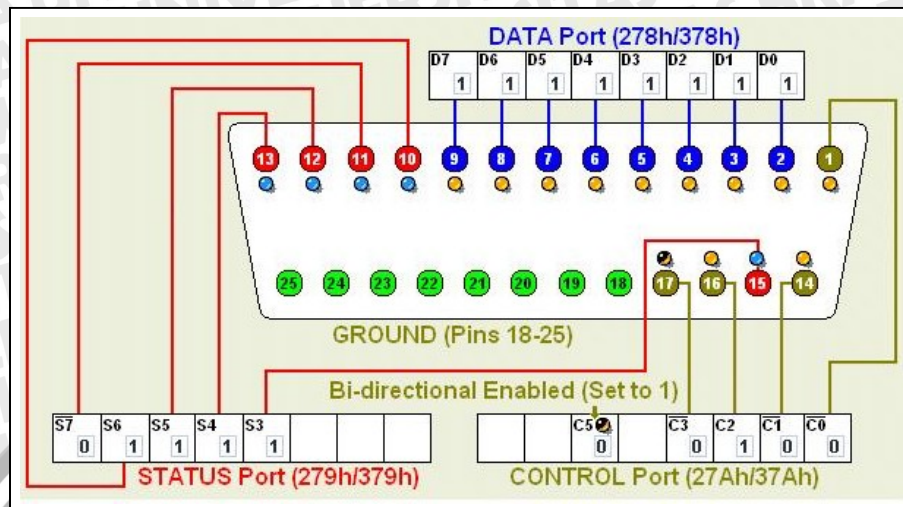
Secara umum LPT *Port* terbagi menjadi 3 saluran :

1. Saluran Data : digunakan untuk mentransfer data yang berasal dari dan ke LPT *Port*.
2. Saluran Status : digunakan sebagai penunjuk status divais yang terhubung dengan LPT *Port*.
3. Saluran Kontrol : digunakan untuk mengontrol data yang keluar masuk dari LPT *Port*.



Gambar 2.15 LPT *Port*

Sumber : Craig Peacock, 2000



Gambar 2.16 Detail pin LPT Port

Sumber : Craig Peacock, 2000

Berikut adalah tabel-tabel karakteristik untuk ketiga saluran tersebut (mode : *Byte*)

Tabel 2.3 Register Data

Offset	Nama	Sifat	No. Bit	Penggunaan	No. Pin
Base + 0, Defaultnya : \$378	Saluran Data	<i>Write Only</i> , untuk saluran 2 arah sifat menjadi <i>Read/Write</i>	Bit 7	Data 7	9
			Bit 6	Data 6	8
			Bit 5	Data 5	7
			Bit 4	Data 4	6
			Bit 3	Data 3	5
			Bit 2	Data 2	4
			Bit 1	Data 1	3
			Bit 0	Data 0	2

Sumber : Romy B Widodo, 2007

Tabel 2.4 Register Status

Offset	Nama	Sifat	No. Bit	Penggunaan	No. Pin
Base + 1, Defaultnya : \$379	Saluran Status	<i>Read Only</i>	Bit 7	<i>Busy</i>	11
			Bit 6	<i>Ack</i>	10
			Bit 5	<i>Paper Out</i>	12
			Bit 4	<i>Select</i>	13
			Bit 3	<i>Error</i>	15
			Bit 2	<i>IRQ</i>	
			Bit 1	<i>Reserved</i>	
Bit 0	<i>Reserved</i>				

Sumber : Romy B Widodo, 2007

Tabel 2.5 Register Kendali

Offset	Nama	Sifat	No. Bit	Penggunaan	No. Pin
Base + 2, Defaultnya : \$37A	Saluran Kendali	<i>Read/Write</i>	Bit 7	Tidak dipakai	
			Bit 6	Tidak dipakai	
			Bit 5	<i>Enable</i> saluran 2 arah	
			Bit 4	<i>Enable IRQ</i>	
			Bit 3	<i>Select Printer</i>	17
			Bit 2	<i>Initialize</i>	16
			Bit 1	<i>Auto- Linefeed</i>	14
			Bit 0	<i>Strobe</i>	1

Sumber : Romy B Widodo, 2007

Tabel 2.6 Mode *Byte*

Sinyal SPP	Pin	Aktif	I/O	Deskripsi
HostClk	1	H/L	O	Pulsa <i>low</i> pada akhir tiap transfer data mode <i>byte</i> menandakan <i>byte</i> yang diterima
HostBusy	14	H/L	O	<i>low</i> : PC siap untuk <i>byte</i> , <i>high</i> : <i>byte</i> diterima
1284 Active	17	H	O	<i>high</i> untuk PC dengan mode transfer 1284
Init	16	H	I	tidak digunakan pada proses <i>reversed</i>
Ptr Clk	10	H/L	I	<i>low</i> : data valid pada data bus
Ptr Busy	11	H	I	<i>Forward</i> untuk sinyal <i>busy</i>
Ack Data Register	12	H	I	ikut data valid menuju <i>high</i>
Xflag	13	H	I	tidak digunakan
Data Available	15	H	I	diset <i>low</i> peripheral untuk menunjukkan data <i>reverse</i> bisa dipakai
Data	2 s/d 9	H	I/O	data dari alat ke PC

Sumber : Romy B Widodo, 2007

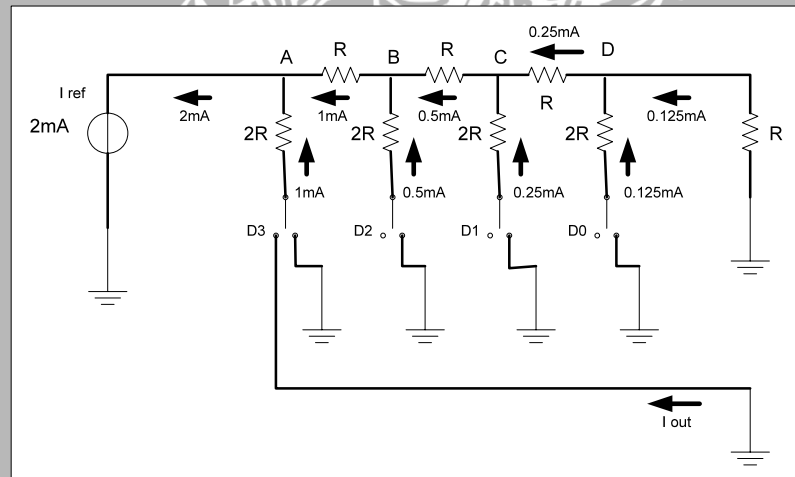
Untuk menyisipkan kode-kode *assembly* ke dalam perintah Pascal pada *Visual Basic/Delphi* diperlukan komponen bantuan. Tujuannya adalah untuk membuka *port* paralel pada sistem operasi Windows 2000, NT, dan XP. Kecuali bila sistem operasi yang digunakan berupa Windows 9x dan Me, maka hal tersebut tidak perlu dilakukan. Hal ini dikarenakan Microsoft telah menutup akses I/O *port* untuk alasan keamanan. Komponen tersebut berupa *Inpout32.dll* yang dapat di *download* di <http://www.processlibrary.com/directory/files/inpout32/>.

(Romy B Widodo, 2007)

2.8 DAC (*Digital to Analog Converter*)

Proses konversi Digital ke Analog (D/A) dan Analog ke Digital (A/D) mempunyai peranan yang vital dalam proses komunikasi PC dengan perangkat analog melalui port *Input/Output* (I/O). Pada dasarnya, konversi D/A adalah proses pengambilan harga yang dinyatakan dalam kode digital (misalnya : biner atau BCD) dan mengubahnya menjadi tegangan atau arus yang sebanding dengan nilai digitalnya. Tegangan atau arus ini merupakan besaran analog, karena untuk satu rentang tertentu dapat memiliki setiap harga yang berbeda.

Input-input digital A, B, C dan D biasanya diperoleh dari output register suatu sistem digital, $2^4 = 16$. Untuk setiap bilangan *input*, tegangan *output* konverter DAC memiliki nilai yang berbeda. Dalam kasus ini, tegangan *output* analognya sama dengan besarnya bilangan biner dalam volt. Nilai ini dua kali bilangan biner atau beberapa faktor perbandingan yang lain. Diagram Blok DAC dengan sistem R2R dapat dilihat pada Gambar 2.17



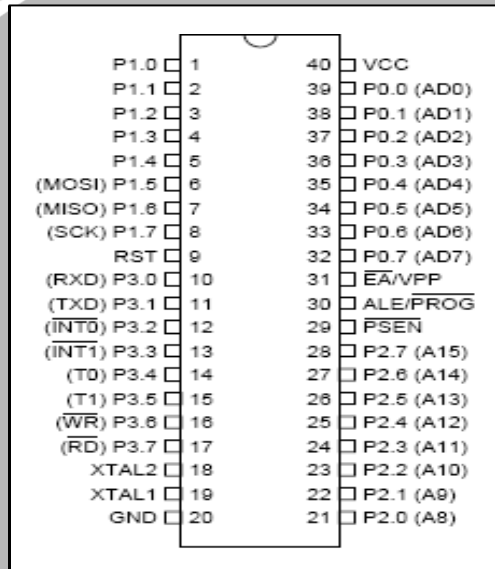
Gambar 2.17 Diagram Blok DAC dengan Sistem R2R.

Sumber : Malvino, 1996 : 340

2.9 Mikrokontroler AT89S52

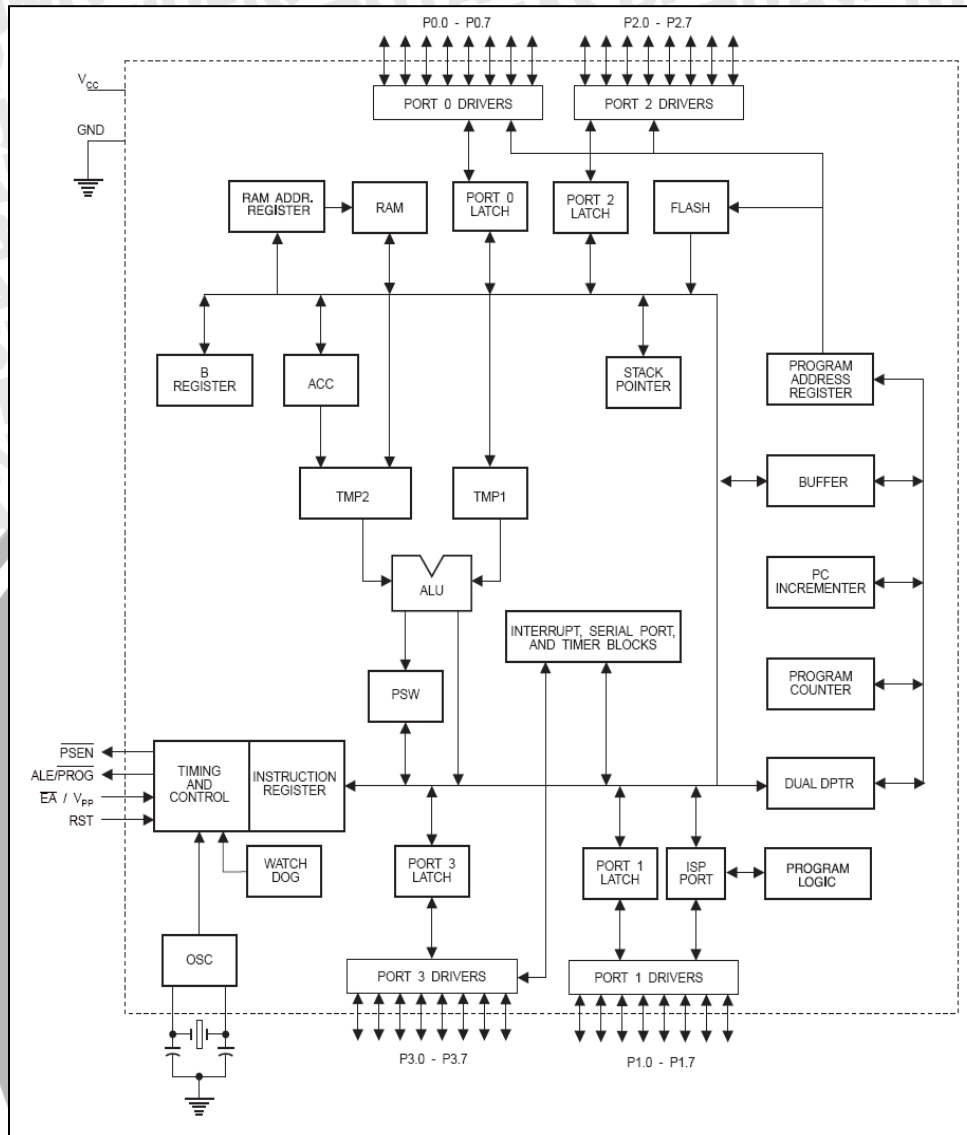
AT89S52 merupakan mikrokontroler keluaran Atmel dengan 8K *byte* Flash PEROM (*Programmable and Erasable Read Only Memory*), AT89S52 merupakan memori dengan teknologi *nonvolatile*, isi memori dari AT89S52 dapat diisi ulang ataupun dihapus berkali-kali.

Memori ini biasa digunakan untuk menyimpan *instruksi* (perintah) berstandart MCS-51 *code* sehingga memungkinkan memori ini untuk bekerja dalam model *single chip operation* (mode operasi keping tunggal) yang tidak memerlukan *external memory* (memori luar) untuk menyimpan *source code* tersebut. Gambar 2.18 menunjukkan pin-pin AT89S52.



Gambar 2.18. Pin AT89S52

Sumber : Atmel, 2005



Gambar 2.19. Blok diagram fungsional AT89S52

Sumber: Atmel, 2005

Fungsi dari tiap-tiap pin pada mikrokontroler AT89S52 yang ditunjukkan dalam Gambar 2.19 sebagai berikut:

A. Pin 1 sampai 8 (Port 1)

Merupakan port paralel 8 bit dua arah (*bidirectional*) yang dapat digunakan untuk I/O biasa atau menerima *low order address bytes* pada saat *Flash Programming*. Port ini mempunyai *internal pull up* dan berfungsi sebagai *input* dengan memberikan logika 1. Sebagai *output port 1* dapat memberikan *output sink* ke empat buah *input TTL*.

B. Pin 9 (RST)

Masukan *reset* yang aktif dengan memberikan *input high* selama 2 siklus mesin. Pin ini dihubungkan ke rangkaian *power on reset*.

C. Pin 10 sampai 17 (Port 3)

Merupakan *port* paralel 8 *bit* dua arah yang dapat digunakan untuk I/O biasa.

Memiliki fungsi khusus yang tampak dalam Tabel 2.7

Tabel 2.7. Deskripsi alternatif pin port 3

Kaki Port	Fungsi alternatif	Keterangan
P3.0	RXD	<i>serial input</i>
P3.1	TXD	<i>serial output</i>
P3.2	INT0	<i>external interrupt 0</i>
P3.3	INT1	<i>external interrupt 1</i>
P3.4	T0	<i>timer 0 external input</i>
P3.5	T1	<i>timer 1 external output</i>
P3.6	WR	<i>external data memory write strobe</i>
P3.7	RD	<i>external data memory read strobe</i>

Sumber: Atmel, 2005

D. Pin 18 (XTAL1)

Pin masukan ke rangkaian osilator internal. Sumber osilator luar juga dapat digunakan.

E. Pin 19 (XTAL2)

Pin keluaran ke rangkaian osilator internal, dipakai bila menggunakan osilator kristal.

F. Pin 20 (GND)

Dihubungkan ke V_{ss} atau *ground*.

G. Pin 21 sampai 28 (Port 2)

Merupakan *port* paralel dua arah yang dapat digunakan sebagai I/O biasa dan *high order address* pada saat mengakses memori secara 16 *bit*. Pada saat

mengakses memori secara 8 bit, port ini mengeluarkan isi dari port 2 berupa *Special Function Register*. Port 2 mempunyai *internal pull up* dan berfungsi sebagai *input* dengan memberikan logika 1. Sebagai *output*, port 2 dapat memberikan *output sink* ke empat buah *input* TTL.

H. Pin 29 (PSEN)

Program Store Enable (PSEN) merupakan sinyal pengontrol yang digunakan mengeksekusi program yang terletak pada memori eksternal masuk ke dalam bus selama proses pengambilan instruksi (*fetching*). PSEN aktif dua kali setiap siklus mesin, kecuali dua aktivasi PSEN dilompati saat mengakses memori data eksternal.

I. Pin 30 (ALE/PROG)

Berfungsi sebagai *Address Latch Enable* (ALE) yang menahan alamat memori eksternal selama pelaksanaan instruksi. Sedangkan pada saat *Flash Programming* (PROG) berfungsi sebagai *pulse input* untuk operasi normal ALE akan mengeluarkan sinyal *clock* pada pin ini dapat pula di-*disable* dengan men-*set bit* 0 dari *Special Function Register* di alamat 8EH. ALE hanya akan aktif pada saat mengakses memori eksternal (MOVX & MOVC).

J. Pin 31 (EA/VP)

Pada kondisi *low*, pin akan berfungsi untuk menjalankan program yang ada pada memori eksternal setelah sistem di-*reset*. Jika berkondisi *high* pin ini berfungsi untuk menjalankan program pada memori internal. Pada saat *Flash Programming* pin ini akan mendapat tegangan 12 volt (VP).

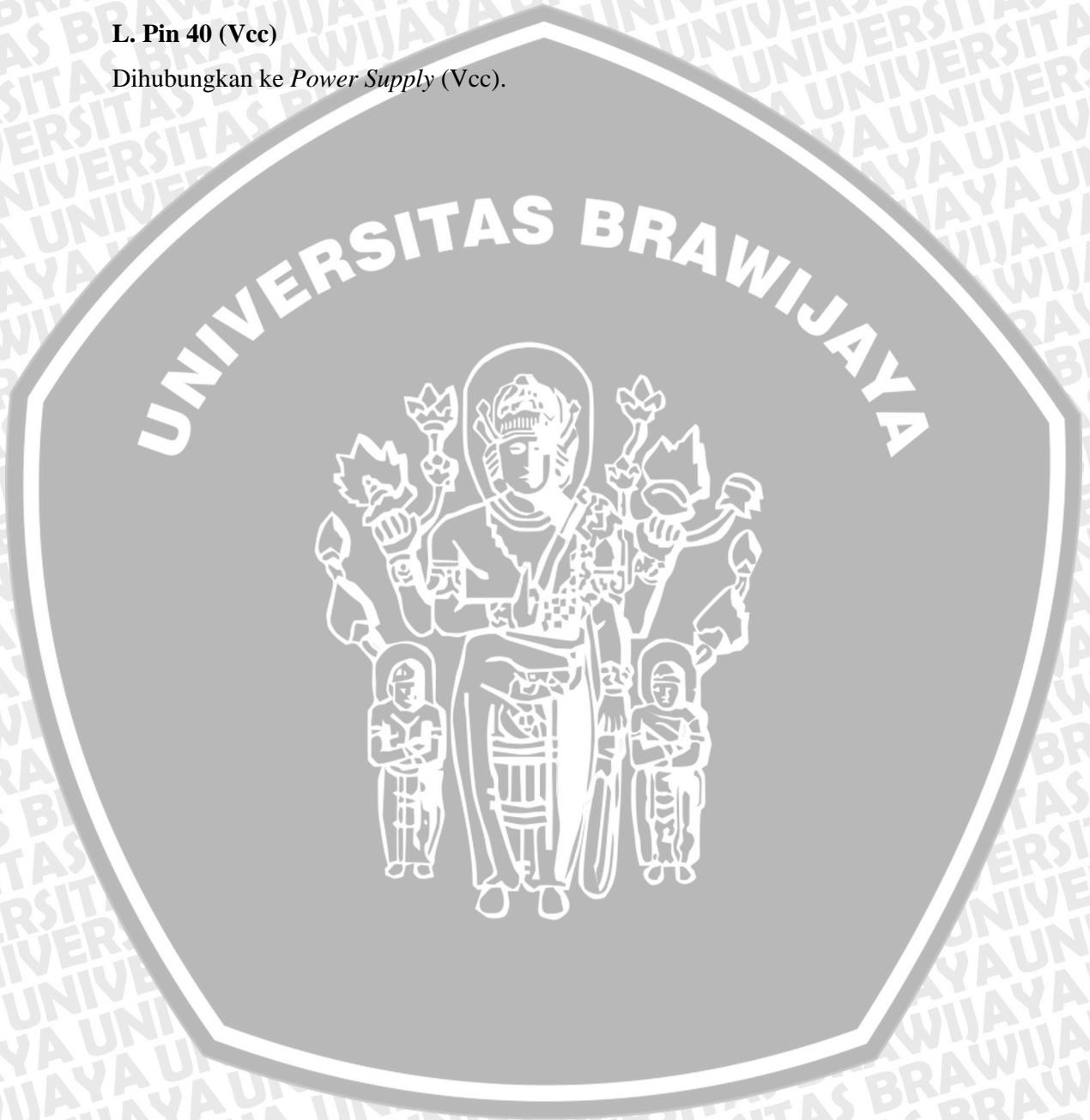
K. Pin 32 sampai 39 (Port 0)

Merupakan port paralel dua arah yang dapat digunakan sebagai I/O biasa, *low order multiplex address/data* ataupun menerima kode *byte* pada saat *Flash Programming*. Pada fungsi sebagai I/O biasa port ini dapat memberikan *output sink* ke delapan buah TTL *input* atau dapat diubah sebagai *input* dengan memberikan logika 1 pada port tersebut. Pada fungsi sebagai *low order*

multiplex address/data port ini akan mempunyai *internal pull up*. Pada saat *Flash Programming* diperlukan *external pull up* terutama pada saat verifikasi program.

L. Pin 40 (Vcc)

Dihubungkan ke *Power Supply (Vcc)*.



UNIVERSITAS BRAWIJAYA



This document was created with Win2PDF available at <http://www.daneprairie.com>.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.

