

## BAB IV

### Hasil dan Pembahasan

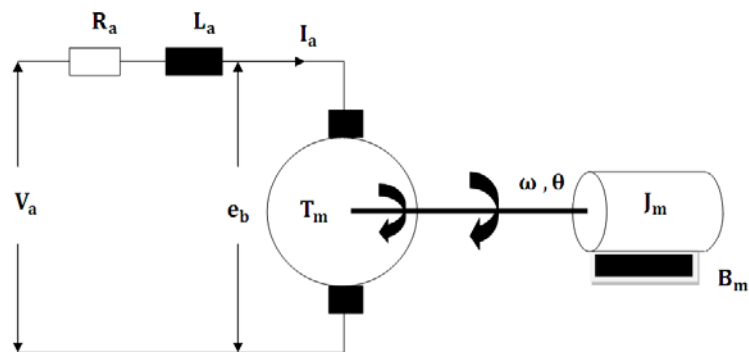
#### 4.1. Pendahuluan

Pada bab ini akan ditampilkan hasil data simulasi dan eksperimen. Ada dua jenis data yang akan ditampilkan. Yaitu:

1. Diagram blok untuk tanpa pengendalian PID dan diagram blok untuk pengendalian PID.
2. Kurva kecepatan motor DC.

#### 4.2. Prinsip kerja alat system kendali PID

Motor DC merupakan motor listrik dengan sumber listrik DC. Motor DC yang diterangkan berjenis motor DC dengan magnet permanen. Rangkaian motor DC dapat dilihat pada Gambar 4.1.



Gambar 4.1 Rangkaian Motor DC Magnet Permanen

Dari gambar 4.1 dapat diperoleh Persamaan ( 4.1 ) sampai ( 4.4 )

$$V_a(t) = R_a i_a(t) + L_a \frac{di_a(t)}{dt} + e_b(t) \quad 4.1$$

$$e_b(t) = K_b \cdot \omega(t) \quad 4.2$$

$$T_m(t) = K_m \cdot i_a(t) \quad 4.3$$

$$T_m(t) = J_m \cdot \frac{d\omega(t)}{dt} + B_m \cdot \omega(t) \quad 4.4$$

#### Keterangan

$V_a$  : tegangan armature (V)

$R_a$  : tahanan armature ( $\Omega$ )

$L_a$  : induktansi armature (H)

$i_a$  : arus armature (A)

$e_b$  : tegangan induksi balik emf (V)

$\omega$  : kecepatan putar (rad/s)

$T_m$  : torsi motor (N.m)

$\theta$  : posisi putaran poros motor (rad)

$J_m$  : momen inersia rotor ( $\text{kg} \cdot \text{m}^2$ )

$B_m$  : koefisien gesek viscous (N.m s/rad)

$K_m$  : konstanta torsi motor (N.m / A)

$K_b$  : konstanta tegangan induksi balik emf (V/(rad/s))

$$e_b(t) = K_b \cdot \omega(t) \quad 4.5$$

$$V_r + V_L + e_b - V_a = 0$$

$$i \cdot R + L \cdot \frac{di}{dt} + K_m \cdot \omega(t) - V_a = 0$$

Bentuk Laplace dari persamaan diatas :

$$i \cdot R + L \cdot \frac{di}{dt} + K_m \cdot \frac{d\theta}{dt} = V_a$$

$$I(s) \cdot R + L \cdot S \cdot I(s) + K_m \cdot S\theta(s) = V_a(s)$$

$$\Sigma T_m(t) = K_m \cdot i_a(t) \quad 4.6$$

$$\Sigma T = J \frac{d^2\theta}{dt^2} + B_m \cdot \frac{d\theta}{dt}$$

Bentuk laplace dari persamaan :

$$K_m \cdot i_a(t) = J \frac{d^2\theta}{dt^2} + B_m \cdot \frac{d\theta}{dt}$$

$$K_m \cdot I(s) = JmS^2\theta(s) + BmS\theta(s)$$

Subtitusikan persamaan(4.5 ) dengan persamaan (4.6) :

$$I(s) \cdot R + L \cdot S \cdot I(s) + K_m \cdot S\theta(s) = V_a(s) \quad 4.7$$

$$I(s)(R + L \cdot S) = V_a(s) - K_m \cdot S\theta(s)$$

$$I(s) = \frac{V_a(s) - K_m \cdot S\theta(s)}{(R + L \cdot S)}$$

$$K_m \cdot I(s) = JmS^2\theta(s) + BmS\theta(s)$$

$$K_m \left( \frac{V_a(s) - K_m \cdot S\theta(s)}{(R + L \cdot S)} \right) = JmS^2\theta(s) + BmS\theta(s)$$

$$K_m(V_a(s) - K_m \cdot S\theta(s)) = (R + L \cdot S)(JmS^2\theta(s) + BmS\theta(s))$$

Persamaan fungsi alih kecepatan poros rotor  $\omega(s)$  terhadap tegangan masukan  $V_a(s)$  dinyatakan seperti di persamaan (2.12).

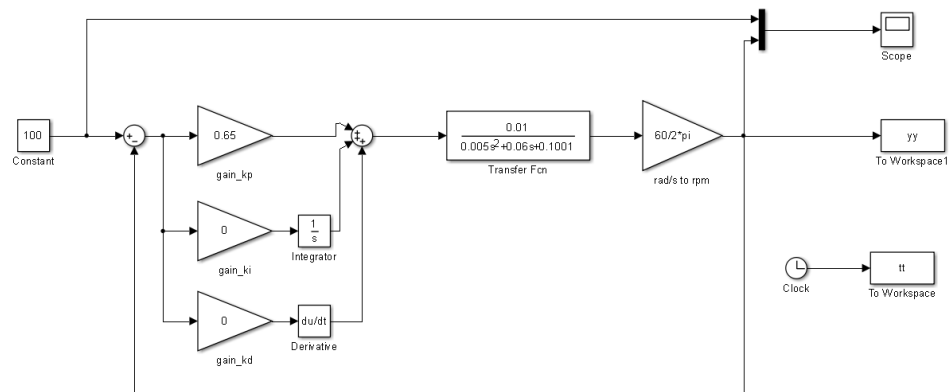
$$\frac{\omega(s)}{V_a(s)} = \frac{K_m}{L_a \cdot J_m \cdot s^2 + (R_a \cdot J_m + L_a \cdot B_m)s + (R_a \cdot B_m + K_b \cdot K_m)} \quad 4.8$$

Persamaan fungsi alih posisi poros  $\theta(s)$  terhadap tegangan masukan  $V_a(s)$  dinyatakan seperti dipersamaan (2.13)

$$\frac{\theta(s)}{V_a(s)} = \frac{K_m}{L_a \cdot J_m \cdot s^3 + (R_a \cdot J_m + L_a \cdot B_m)s^2 + (R_a \cdot B_m + K_b \cdot K_m)s} \quad 4.9$$

### 4.3. Pengujian alat

Untuk pengujian penelitian ini menggunakan simulasi pengendali PID motor DC yang terdapat pada software Matlab. Simulasi yang akan digunakan dalam penelitian ini seperti gambar 4.2.



Gambar 4.2 rangkaian simulasi motor DC pengendali PID

Pada gambar 4.2 tersebut terbagi menjadi tiga tahapan, yaitu :

#### 1. Input

Merupakan masukan tegangan yang digunakan sebagai sumber dari rangkaian motor DC tersebut. Nilai yang digunakan pada simulasi ini adalah konstan, yaitu 100.

#### 2. Proses

Terdapat beberapa komponen-komponen yang digunakan dalam proses pengendalian motor DC. Diantaranya :

##### 1. Pengendali proposional

Sebagai penguatan dengan gambar segitiga untuk memperkuat inputan yang masuk yang akan dilanjutkan oleh transfer function.

## 2. Pengendali integral

Untuk memperbaiki kesalahan keadaan tunak yang terjadi pada pengendali proposional yang sebelumnya.. Dengan menambahkan fungsi *integrator* yang terdapat dalam matlab. Namun apabila pemilihan  $1/T_i$  yang tidak tepat dapat menimbulkan masalah baru, yaitu dapat menimbulkan osilasi atau overshoot

## 3. Pengendali derivative

Dalam simulasi yang digunakan untuk penelitian ini digambarkan dengan fungsi *derivative* yang terdapat dalam *matlab*. Pengendali *derivative* sendiri digunakan untuk mengurangi kecepatan saat terjadi *overshoot* atau dapat dianggap sebagai "rem".

## 4. Transfer function

Merupakan hasil perhitungan yang telah dilakukan sebelumnya dari persamaan 4.9.

## 3. Output

Hasil dari tahapan proses yaitu gambaran sebuah kurva kecepatan dan kurva tegangan motor DC yang nantinya akan dianalisa.

Setelah melalui tiga tahapan diatas selanjutnya akan dianalisa hasil dari outputan-nya. Pengendali PID memiliki empat sifat yang dapat diketahui. Diantaranya :

1. Rise time

Merupakan berapa lama waktu yang dibutuhkan motor DC dengan pendali PID untuk mencapai posisi yang tunak.

2. Overshoot

Lonjakan yang terjadi ketika *rise time* yang menyebabkan motor DC berkerja lebih untuk kembali ke posisi tunak.

3. Settling time

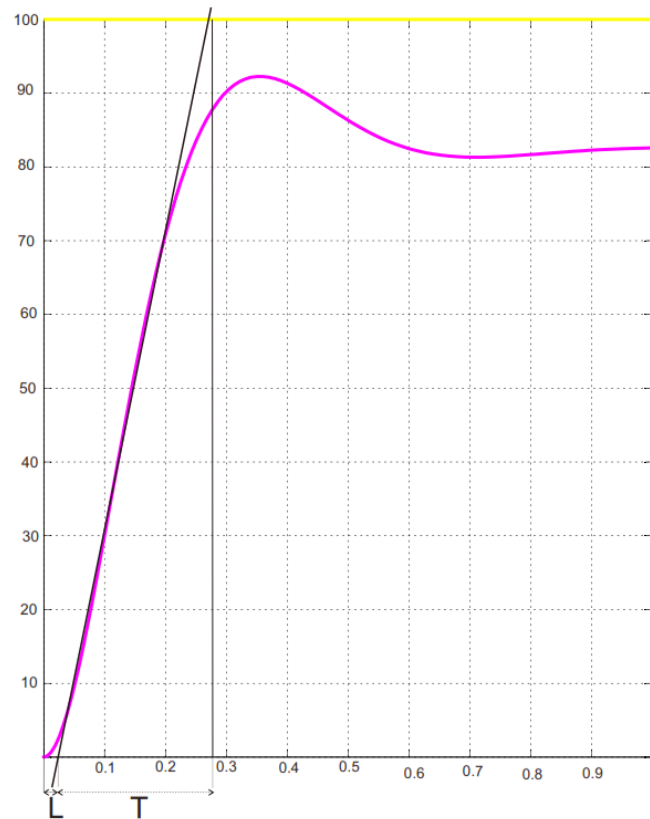
Waktu yang dibutuhkan motor DC menuju posisi stabil

4. Steady state eror

Perbandingan antara saat posisi stabil dengan posisi yang diinginkan.

Untuk menentukan nilai  $K_p$ ,  $K_i$  dan  $K_d$  dilakukan secara manual melalui hasil dari simulasi pengendalian PID motor DC matlab berdasarkan terori Ziegler Nichols dibawah ini :

Sebagai acuan untuk perbandingan simulasi motor DC nilai  $K_p$  ditentukan 0.65. untuk nilai  $K_i$  dan  $K_d$  tetap 0. Jalankan simulasi tersebut dan lihat hasil kurva yang ditampilkan oleh *scope*.



Gambar 4.3 kurva kecepatan motor DC

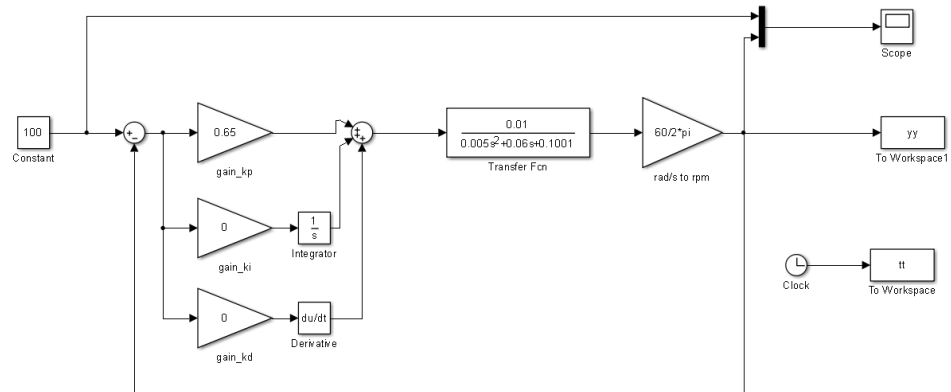
Dari gambar 4.2 dapat diambil data dari hasil garis kedua perpotongan atas dan bawah. Sehingga didapatkan :

$$L : 0.0343$$

$$T : 0.2743 - L = 0.2400$$

Lalu data tersebut dimasukkan kedalam coding yang telah ditentukan ke software Matlab melalui rumus coding yang berdasarkan rumus Ziegler Nichols dibawah ini :





Gambar 4.4 rangkaian percobaan simulasi pengendali PID motor DC

model controller	$K_p$	$T_i$	$T_d$
P	$\frac{T}{L}$	$\infty$	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	$2L$	$0.5L$

Table 4.1 acuan pengaturan dasar Ziegler Nichols

Lalu masukan nilai T dan L yang telah dihitung secara manual

model controller	$K_p$	$T_i$	$T_d$
P	$\frac{0.24}{0.0343}$	$\infty$	0
PI	$0.9 \times \frac{0.24}{0.0343}$	$\frac{0.343}{0.3}$	0
PID	$1.2 \times \frac{0.24}{0.0343}$	$2 \times 0.343$	$0.5 \times 0.343$

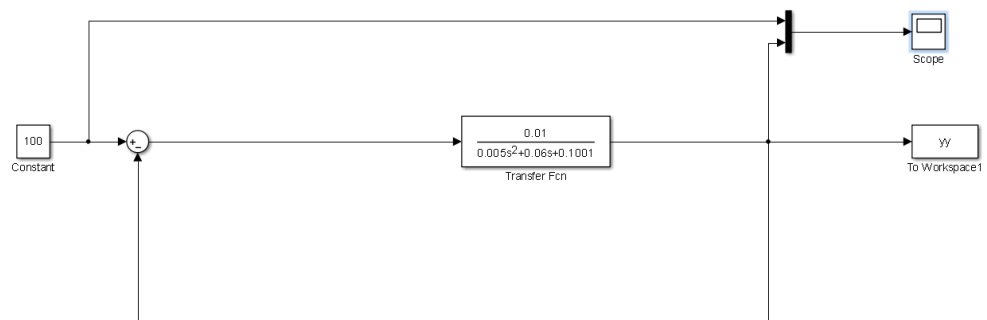
Sehingga didapatkan :

model controller	$K_p$	$T_i$	$T_d$
P	6.997085	$\infty$	0
PI	6.297376093	0.114333	0
PID	8.396502	0.0686	0.01715

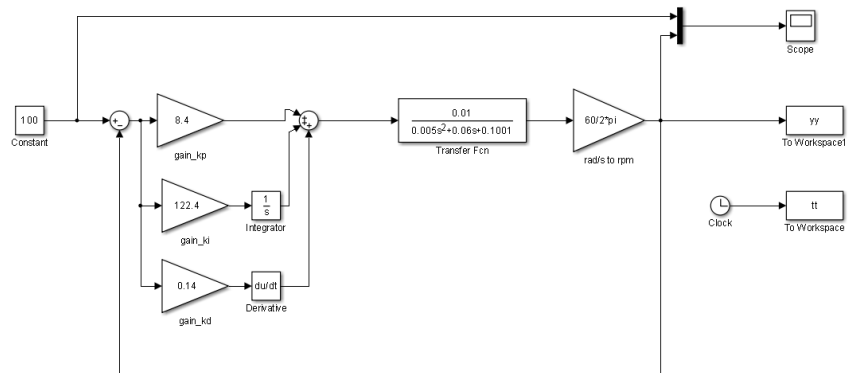
Table 4.2 hasil perhitungan berdasarkan acuan pengaturan dasar Ziegler Nichols

#### 4.4. Hasil dan pembahasan berkaitan dengan pembuatan rangkaian simulasi kendali kecepatan motor DC

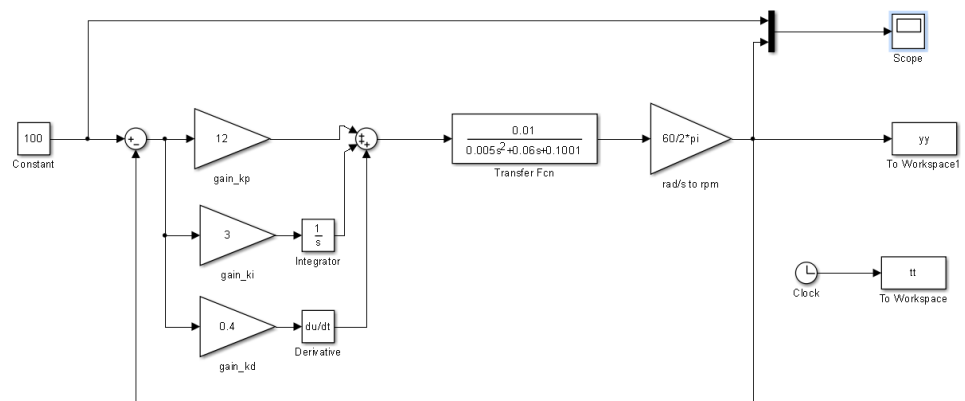
Pada sub bab ini akan ditampilkan 3 rangkaian yang digunakan untuk simulasi. Untuk gambar 4.3 merupakan gambar rangkaian motor DC tanpa pengendali. Sedangkan gambar 4.4 dan gambar 4.5 gambar rangkaian motor DC dengan pengendali PID yang memiliki nilai  $K_p$ ,  $K_i$  dan  $K_d$  yang berbeda-beda.



Gambar 4.5 rangkaian motor DC tanpa pengendali



Gambar 4.6 rangkaian kendali motor DC percobaan 1

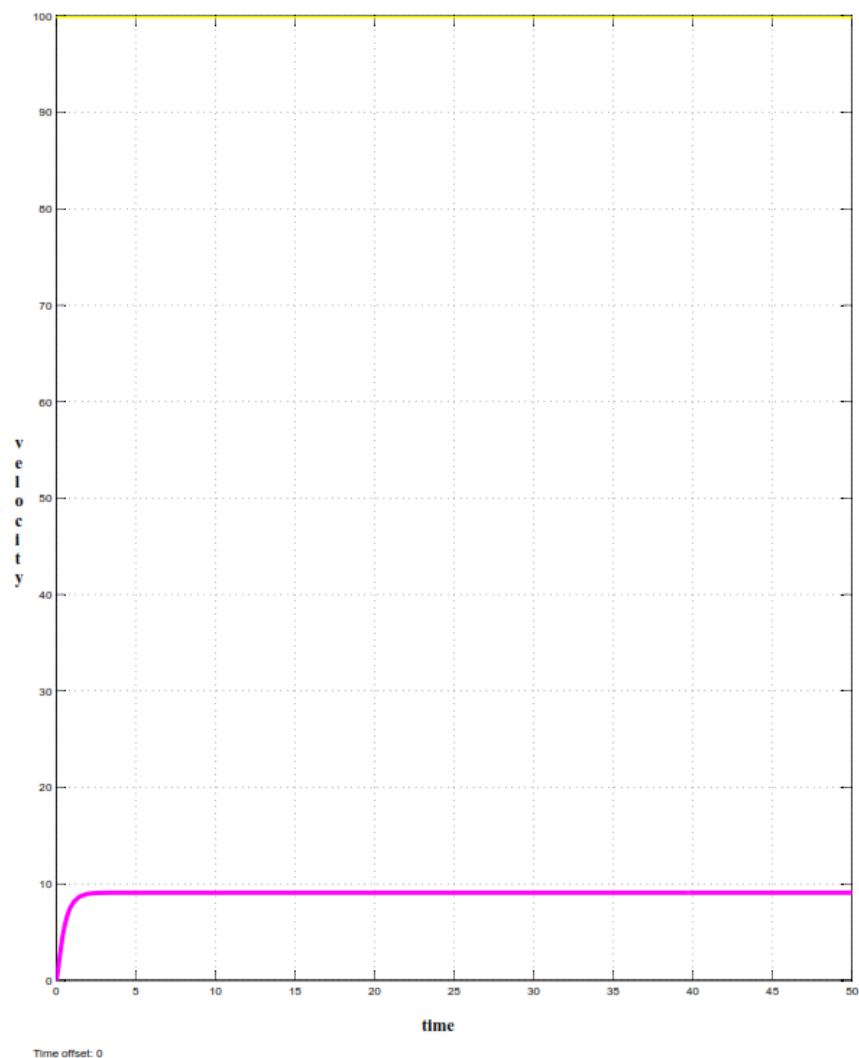


Gambar 4.7 rangkaian kendali motor DC percobaan 2

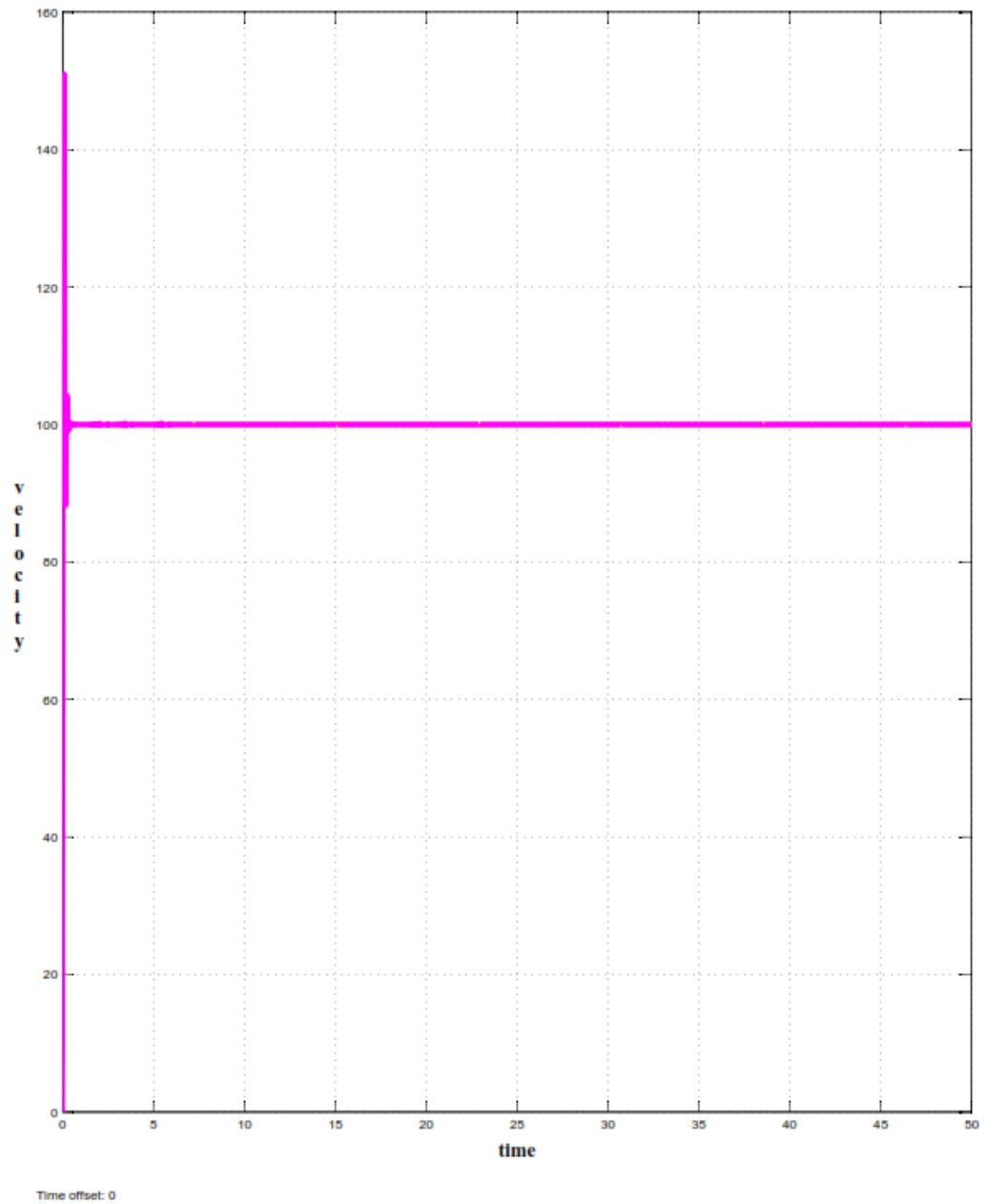
Berdasarkan gambar 4.5 rangkaian motor DC tanpa pengendali diatas,tidak menggunakan tambahan penguatan,integrator dan derivative sehingga terlihat lebih sederhana. Sedangkan untuk gambar 4.6 dan gambar 4.7 menggunakan penguatan, integrator dan derivative dengan nilai  $K_p = 8,4$ ,  $K_i = 122,4$  dan  $K_d = 0,14$  pada gambar 4.6 dan nilai  $K_p = 12$ ,  $K_i = 3$  dan  $K_d = 0,4$  pada gambar 4.7

#### 4.5. Hasil dan pembahasan berkaitan dengan simulasi kendali kecepatan motor DC

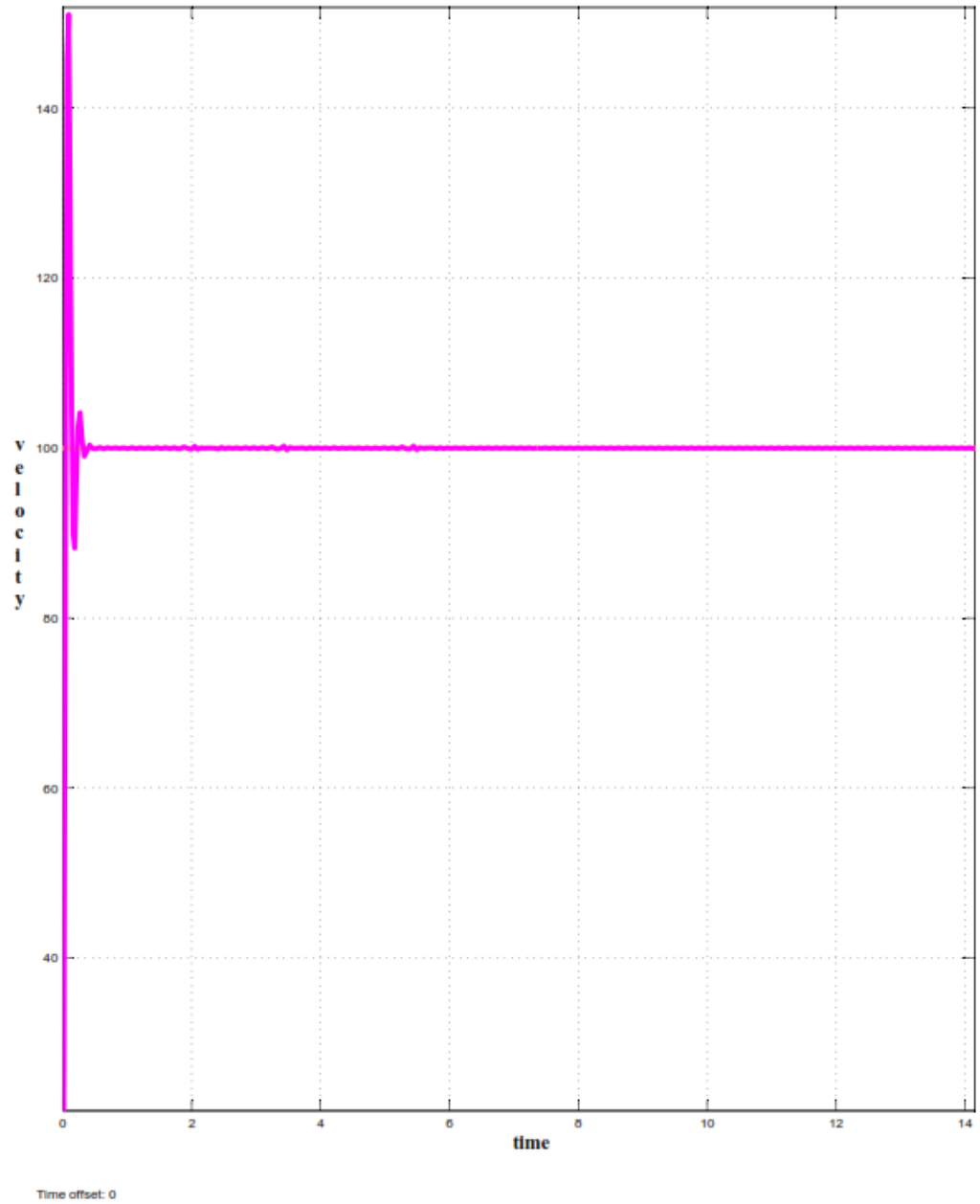
Pada sub bab ini akan ditampilkan data kecepatan dari hasil simulasi kendali kecepatan motor DC. Kurva kecepatan tanpa pengendali dapat dilihat pada gambar 4.8. sedangkan kurva kecepatan dengan pengendali PID dapat dilihat pada gambar 4.9 dan gambar 4.10



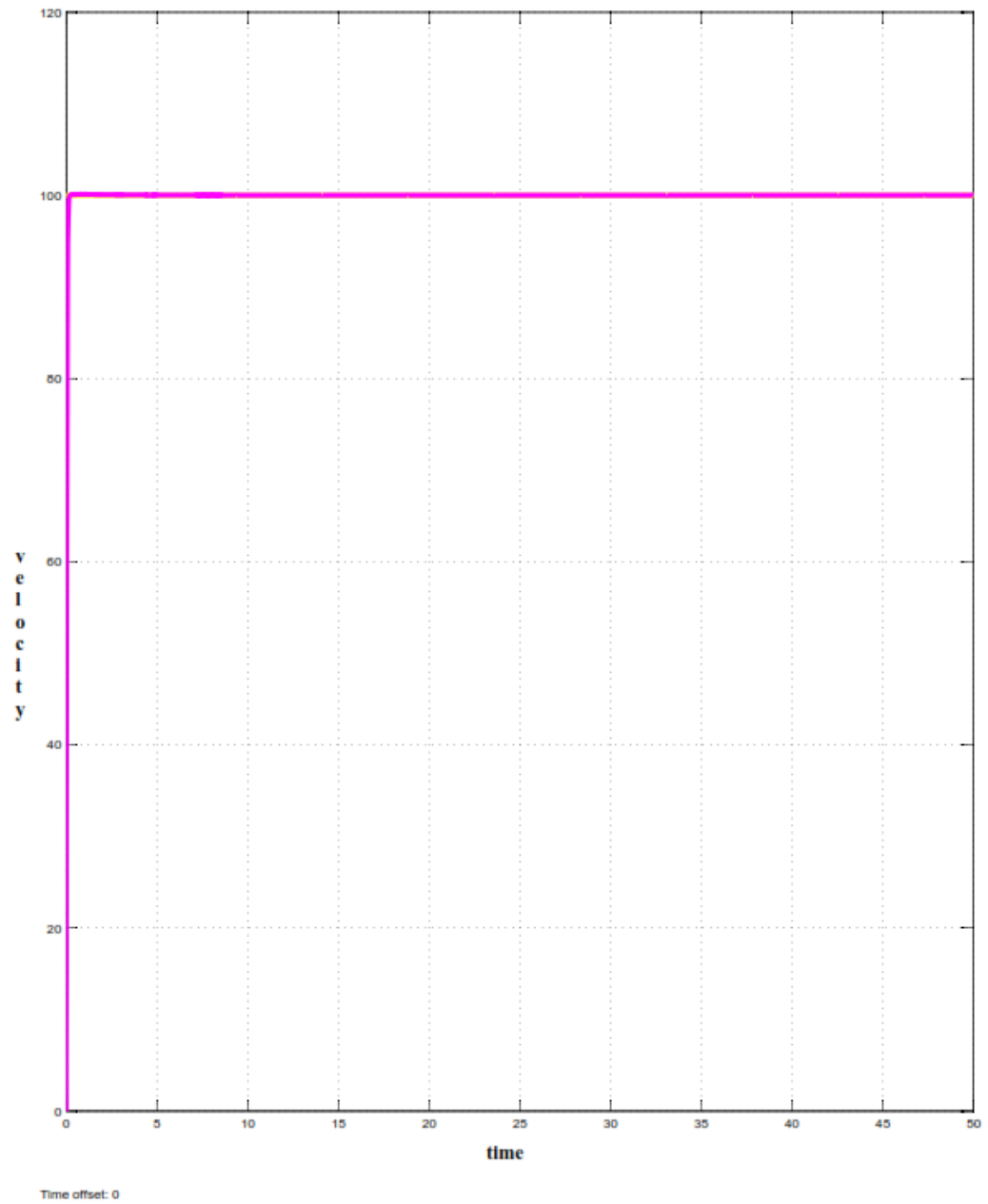
Gambar 4.8 hasil simulasi kurva kecepatan motor DC tanpa pengendali



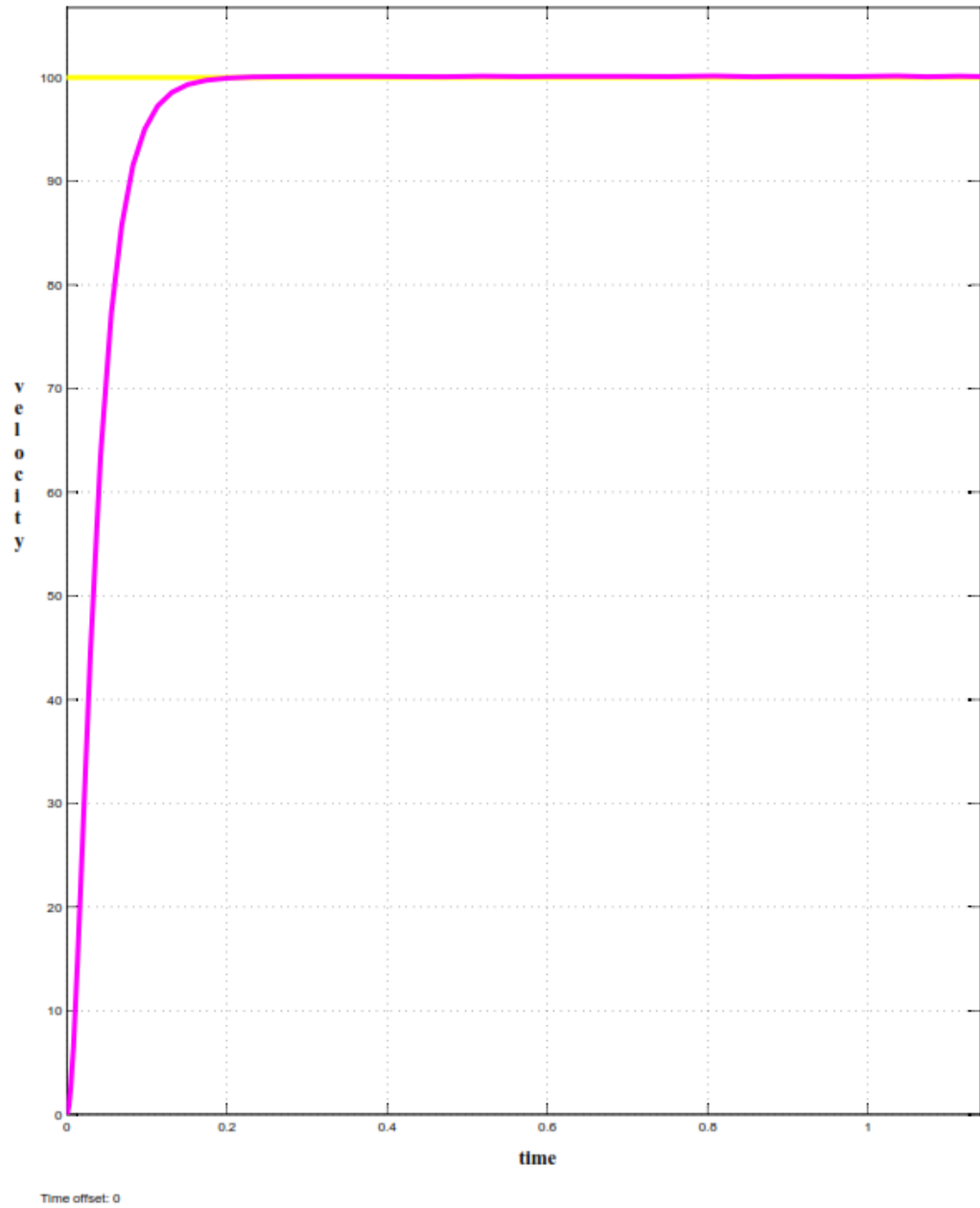
Gambar 4.9 hasil simulasi kurva kecepatan motor DC dengan pengendali  
PID percobaan 1



Gambar 4.10 hasil simulasi kurva kecepatan motor DC dengan pengendali PID percobaan 1 setelah diperbesar



Gambar 4.11 hasil simulasi kurva kecepatan motor DC dengan pengendali PID percobaan 2



Gambar 4.12 hasil simulasi kurva kecepatan motor DC dengan pengendali PID percobaan 2 setelah diperbesar



Dari gambar 4.8 , gambar 4.9 dan gambar 4.11 dapat disimpulkan bahwa pada gambar 4.8 tanpa pengendali PID memiliki *rise time* yang lama yaitu 1 detik sehingga membutuhkan waktu dan tegangan yang banyak untuk di menggerakkan motor DC dan belum mencapai kecepatan yang diinginkan yaitu 100 rpm,lalu pada gambar 4.9 mengalami kenaikan *rise time* yaitu 0,3 detik,namun terjadi *overshoot* pada kecepatan 149 rpm dan berosilasi sehingga tidak efektif untuk digunakan. Yang menyebabkan *overshoot* dan osilasi dikarena nilai  $K_p$ , $K_i$  dan  $K_d$  tidak sesuai sehingga terjadi osilasi. Untuk gmabar 4.11 sudah mengalami perubahan yang signifikan. Karena memiliki nilai  $K_p$ , $K_i$  dan  $K_d$  yang sesuai sebagai pengendali PID.