

# PENGENDALIAN KECEPATAN MOTOR DC MENGUNAKAN SENSOR ENCODER DENGAN KENDALI PI

Jumiyatun

Jurusan Teknik Elektro Fakultas Teknik Universitas Tadulako

E-mail: jum@untad.ac.id

## ABSTRACT

*Digital control system PI based on microcontroller AT 89S52 is made to control the speed of DC motor. Users can interact via keypad and LCD panel. This control system limits the positive control signal (saturation). The digital PI is a process of a program executed by using a computer, by entering a set point value (SP) then the data will be processed using a predefined PI setting, until the corresponding process variable (PV) is obtained so that the error is equal to 0, or Set point value = Process variable. Steady state errors that occur due to the restriction (saturation) by using a quarter-second sampling time.*

**Keywords:** DAC, Encoder, Pi Controller

## INTISARI

Sistem kendali digital PI berbasis mikrokontroler AT 89S52 dibuat untuk mengendalikan kecepatan motor DC. Pengguna dapat berinteraksi melalui keypad dan panel LCD. Sistem kendali ini membatasi sinyal kendali positif (saturasi). PI digital merupakan proses dari suatu program yang dijalankan dengan menggunakan komputer, dengan memasukkan nilai *set point* (SP) kemudian data akan diproses menggunakan *setting* PI yang sudah ditentukan, sampai mendapatkan *Process Variable* (PV) yang sesuai, sehingga *error* yang didapatkan sama dengan 0, atau nilai *Set point* = Variabel proses. Kesalahan *steady state* yang terjadi disebabkan oleh pembatasan (saturasi) dengan menggunakan waktu *sampling* seperempat detik.

Kata Kunci : Kendali PI, DAC, Encoder

## I. PENDAHULUAN

Di dalam suatu sistem kontrol kita mengenal adanya beberapa macam aksi kontrol, diantaranya yaitu aksi kontrol proporsional, aksi kontrol integral. Masing-masing aksi kontrol ini mempunyai keunggulan-keunggulan tertentu, dimana aksi kontrol proporsional mempunyai keunggulan respon yang cepat, aksi kontrol integral mempunyai keunggulan untuk memperkecil *error*, Agar diperoleh *output* dengan respon yang cepat dan *error* yang kecil kita dapat menggabungkan kedua aksi kontrol ini menjadi aksi kontrol PI, dan pada makalah ini sistem kendali yang digunakan adalah sistem kendali PI digital.

Dalam aksi kendali proporsional, *output* dari sistem kontrol selalu sebanding dengan *inputnya*. Sinyal *output* merupakan penguatan dari sinyal kesalahan dengan faktor tertentu, faktor penguatan ini merupakan konstanta proporsional dari sistem, yang dinyatakan

dengan  $K_p$ , dimana  $K_p$  ini mempunyai respon yang tinggi/cepat. Dalam aksi kendali integral, *output* dari kontroler ini selalu berubah selama terjadi penyimpangan, dan kecepatan perubahan *output* tersebut sebanding dengan penyimpangannya, konstantanya dinyatakan dengan  $K_i$ , dimana  $K_i$  ini mempunyai sensitivitas yang tinggi, yaitu dengan cara mereduksi *error* yang dihasilkan dari sinyal feedback. Makin besar nilai  $K_i$  maka sensitivitasnya akan semakin tinggi, tetapi waktu yang dibutuhkan untuk mencapai kestabilan lebih cepat, demikian pula sebaliknya.

Tujuan penelitian ini adalah memanfaatkan mikrokontroler AT 89S52 untuk mengendalikan kecepatan motor DC dengan menggunakan aksi kontrol PI yang *input set pointnya* dapat di *set* dan ditampilkan di dalam LCD.

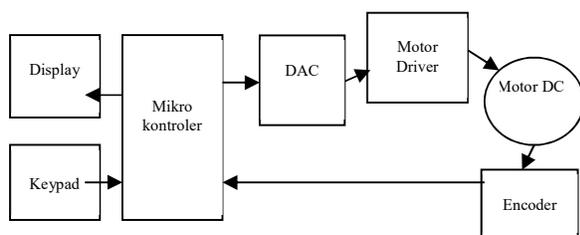
Sistem yang dibuat mempunyai batasan-batasan sebagai berikut :

1. Keypad hanya sebagai *input* yaitu *set point* (SP)

2. LCD hanya menampilkan kecepatan motor dan SP.
3. Motor menggunakan motor DC 12 Volt
4. Motor driver hanya bekerja pada satu arah
5. PI kontroler direalisasikan dalam program /software
6. Encoder digunakan sebagai sensor kecepatan

## II. PERANCANGAN SISTEM

Adapun desain sistem perencanaan dari penelitian ini adalah sebagai berikut:

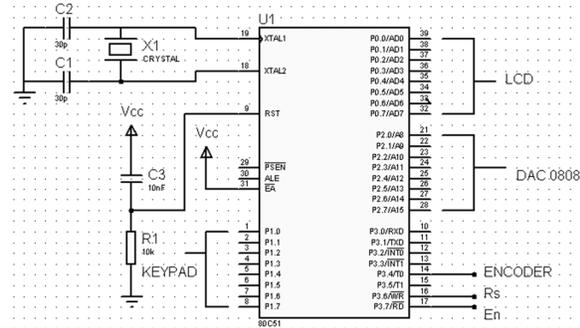


Gambar 1. Diagram blok sistem

Pertama-tama pengguna memberikan *input* kepada mikrokontroler berupa *set point* yang ditampilkan di layar LCD. Kemudian, perhitungan pengendali PI dilakukan oleh mikrokontroler berdasarkan data *set point* dan pembacaan sensor kecepatan (encoder). Hasil perhitungan dikirim ke DAC yang kemudian mengirim tegangan untuk menggerakkan motor DC. Setelah motor berjalan maka akan terdeteksi oleh encoder yang kemudian akan mengirimkan data sebagai umpan balik jika *output* yang dikeluarkan tersebut belum sesuai dan ditampilkan dalam layar LCD.

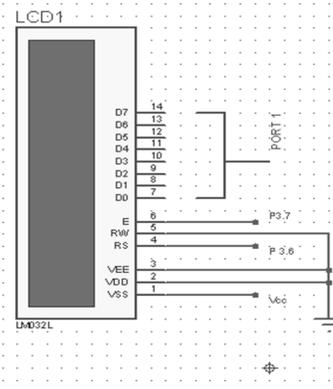
Berikut ini adalah bagian-bagian dari sistem kendali digital yang dibuat :

1. Mikrokontroler menggunakan atmel AT89S52 sebagai unit pemrosesan sentral, akan merespon data set point atau instruksi dari keypad dan menerima informasi dari encoder sesuai program yang ditanamkan pada kontroler. Pada Gambar 2 di bawah P0 digunakan untuk LCD, P1 digunakan untuk Keypad, P2 digunakan untuk DAC.



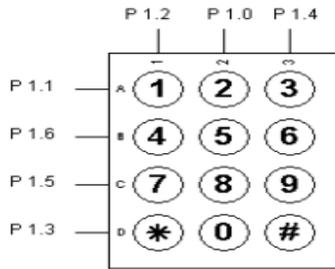
Gambar 2. Minimum Sistem AT 89S52

2. LCD hanya menampilkan kecepatan motor dan *set point* (SP). Untuk menggunakan LCD kita harus menginisialisasi terlebih dahulu karena tanpa inisialisasi maka LCD tersebut tidak akan berfungsi. Pin1 untuk Vcc. Pin 2 untuk ground. Pin 3 untuk pengaturan kontras LCD tapi dalam tugas akhir ini di groundkan untuk kontras yang maximum Pin 4 yaitu RS (*register select*), 0 untuk register perintah dan 1 untuk register data. Pin 5 R/W (*read/write*) dalam rangkaian dibawah digroundkan jadi hanya bisa untuk write saja. Pin 6 yaitu En (Enable clock LCD) jadi setiap pengiriman dan pembacaan data harus berlogika 1. Pin 7-14 digunakan untuk mengirimkan data/*output*.



Gambar 3. LCD

Keypad hanya sebagai *input* untuk Set point (SP). Prinsip kerjanya mencari bagian baris dan kolom yang terhubung contoh misalnya tombol 1 (satu) maka yang harus terhubung baris 1 yaitu P1.1 dengan Kolom 1 yaitu P1.2.



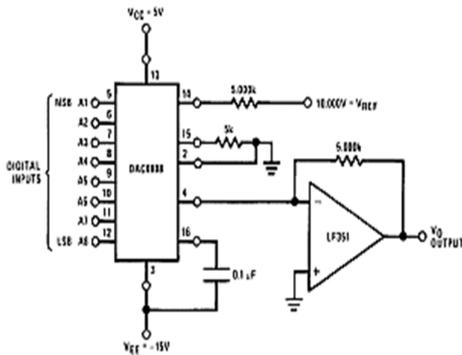
Gambar 4. Keypad

3. DAC berfungsi mengubah data digital menjadi analog. Misalkan Mikrokontroler mengeluarkan data 8 bit dan diinputkan ke DAC (Din) sebesar 11111111b atau 255 D dengan  $V_{ref}$  5 Volt maka output yang dikeluarkan DAC adalah 5 Volt. Sedangkan jika inputnya 1000000B atau 128D. Tegangan analog yang dikeluarkan DAC = (Data digital yang diterima DAC dibagi Data input maksimal) dikalikan tegangan referensi.

$$V_{out} = (D_{in}/D_{max}) \times V_{ref}$$

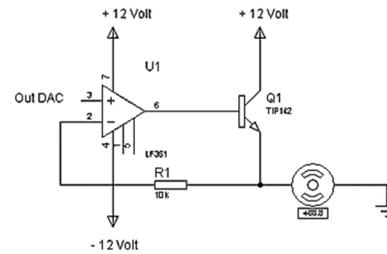
$$V_{out} = (128/255) \times 5 \text{ Volt} = 0,5 \times 5 \text{ Volt}$$

$$V_{out} = 2,5 \text{ Volt}$$



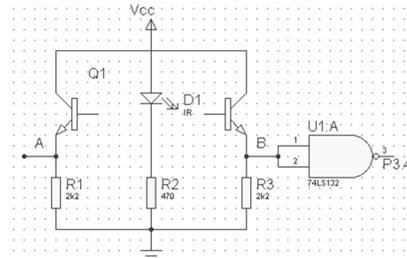
Gambar 5. Rangkaian DAC 0808

4. Driver motor menggunakan satu buah Op-Amp Lf 351 dan transistor TIP 142. Rangkaian ini bekerja karena mendapat input dari output DAC (Digital to Analog Converter) berupa tegangan kemudian masuk Op-Amp, disini Op-Amp hanya bekerja sebagai Tegangan follower dimana  $V_{in} = V_{out}$ . Transistor hanya sebagai pengendali arus yang berubah-ubah tergantung dari masukannya.



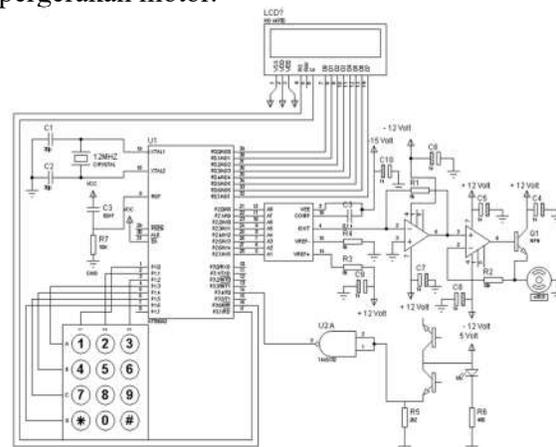
Gambar 6. Rangkaian motor driver

5. Motor menggunakan motor DC 12 Volt. Encoder sebagai umpan balik untuk mendeteksi kecepatan motor. menggunakan Encoder dari scroll mouse yang diambil phototransistor dan infra rednya saja, prinsip kerjanya begitu ada cahaya maka akan on. Rangkaian Encoder yang dilengkapi schmitt trigger yang berfungsi menghasilkan sinyal digital yang murni.



Gambar 7. Rangkaian Encoder

Masing-masing blok memiliki fungsi-fungsi yang berbeda-beda tetapi blok-blok tersebut berhubungan satu dengan yang lainnya sehingga rangkaian ini dapat berfungsi untuk mengontrol pergerakan motor.



Gambar 8. Rangkaian lengkap

#### A. Kontroler PI

PI digital sendiri merupakan suatu proses dari suatu program yang dijalankan dengan menggunakan komputer, dimana kita

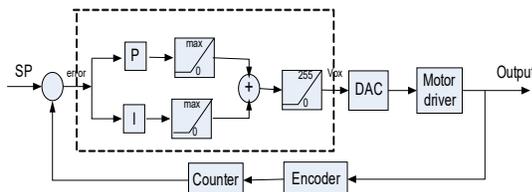
memasukkan nilai *set point* (SP) kemudian data akan diproses menggunakan nilai konstanta PI yang sudah ditentukan, sampai mendapatkan PV yang sesuai dengan SP dimana saat itu terjadi, error yang didapatkan sama dengan 0, atau nilai *Set Point* = Proses Variabel.

Algoritma kendali PI digital yang digunakan adalah sebagai berikut:

$$V_o = V_{o_{n-1}} + K_p(e_n - e_{n-1}) + K_i e_n T_s$$

Dimana :

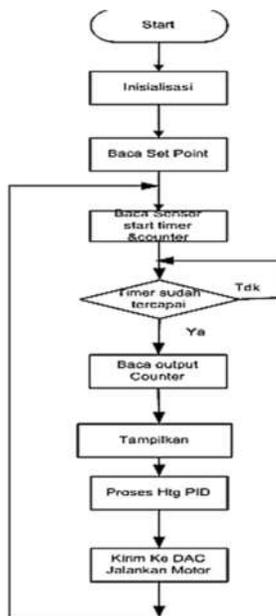
- $V_o$  = output kendali
- $V_{o_{n-1}}$  = output kendali sebelumnya
- $K_p$  = Konstanta Proporsional
- $K_i$  = Konstanta Integral
- $E_n$  = *error* sekarang
- $E_{n-1}$  = *error* 1 kali sebelumnya
- $T_s$  = *Time sampling*



Gambar 9. sistem kendali kontroler PI

**B. Algoritma Program**

Algoritma perangkat lunak secara keseluruhan dapat digambarkan dalam diagram alir seperti yang terlihat pada Gambar 10.



Gambar 10. Diagram alir algoritma perangkat lunak

**Keterangan:**

1. Start
2. Inisialisasi, terdiri dari inisialisasi LCD, isi PV dan Error dinolkan
3. Kontroler membaca *Set point* (SP) dari keypad
4. Proses awal baca sensor dengan menjalankan counter dan timer
5. Apakah *timer* sudah tercapai
6. Baca output counter yaitu nilai PV
7. Tampilkan nilai PV ke LCD
8. Proses hitung PI
9. Kirim hasilnya ke DAC sehingga motor berjalan
10. Kembali ke langkah no.4

**III. HASIL DAN PEMBAHASAN**

**A. Pengujian Perangkat Keras**

Dengan memasukan data *input* DAC yang berbeda-beda menghasilkan *output* yang ditampilkan pada Tabel 1.

Tabel 1. Hasil pengujian DAC dan motor driver

Input DAC (dec)	Arus DAC (mA)	Output DAC (Vdc)	Arus Input Transistor (µA)	Arus Motor (mA)
0	0	0	186	0
50	0,46	2,13	192	73,7
100	0,91	4,28	202	86,8
200	1,82	8,55	230	99,8
255	2,32	10,86	238	101,5

**B. Pengujian hasil akhir**

Setelah melakukan beberapa masukan (SP) ternyata terdapat perbedaan PV terhadap SP seperti yang terlihat dalam Tabel 2.

Tabel 2. Hasil pengujian

No.	SP	PV	No.	SP	PV
1	0	0	16	150	116
2	10	0	17	160	124
3	20	0	18	170	135
4	30	0	19	180	146
5	40	20	20	190	156
6	50	22	21	200	165
7	60	26	22	210	177
8	70	36	23	250	218
9	80	43	24	300	265
10	90	53	25	350	315
11	100	65	26	400	367
12	110	73	27	450	415
13	120	85	28	500	464
14	130	94	29	550	517
15	140	105	30	600	570

Berdasarkan Tabel 2, sistem yang dibuat mempunyai kisaran SP diatas 40. hal ini disebabkan oleh karakteristik motor (*dead zone*).

#### IV. KESIMPULAN

Alat sudah dapat bekerja sesuai yang diinginkan untuk mengendalikan kecepatan motor, meskipun hasilnya masih ada *error steady state*, yaitu perbedaan antara kecepatan yang diinginkan dengan yang terukur. Perbedaan ini disebabkan oleh saturasi yang dikenakan pada sinyal kendali.

#### DAFTAR PUSTAKA

- [1]. Richard, D. C. (1983). *Sistem Pengaturan*. Penerbit Erlangga, Jakarta.
- [2]. Jacob, J. M. (1989). *Industrial Control Electronics*. Prentice Hall, Inc.
- [3]. Christanto, D., Pusporini, K. (2004). *Panduan dasar mikrokontroler keluarga MCS-5*. Innovative Electronics, Surabaya.
- [4]. Ogata, K. (1990). *Teknik Kontrol Automatik*. Penerbit Erlangga, Jakarta.