

SISTEM KENDALI KECEPATAN MOTOR ARUS SEARAH DENGAN ALGORITMA PROPORTIONAL INTEGRAL DERIVATIVE (PID) DIGITAL BERBASIS FIELD PROGRAMMABLE GATE ARRAY

Helmy Widyantara¹⁾

1) S1/Jurusan Sistem Komputer, Sekolah Tinggi Manajemen Informatika & Teknik Komputer Surabaya email:
helmyid@stikom.edu

Abstract : A good control system should have a resistance to interference and has fast response and accurate, but the applications are still many who use the analog control system. In the analog control system, control method depends with Operational Amplifiers and magnitude of the analog electronic components, resistors and capacitors. As known analog component electronics has a weakness that is more sensitive to the surrounding conditions. It created the need for control system Proportional Integral Derivative (PID) digital. On the application of digital PID control system requires good hardware support, so that they can run the computing process quickly and can complete complex computations. With reliability PID Field Programmable Gate Array (FPGA) digital control system can be designed and implemented with logic gates.

Keywords : Control, DC Motor, PID Digital, FPGA

Sistem kendali PID (*Proportional Integral Derivative*) telah lama digunakan untuk berbagai keperluan kendali, pada awalnya penerapan sistem kendali masih menggunakan metode PID analog, dimana pada metodenya bergantung pada *Operational Amplifier* dan besaran komponen elektronika analog seperti Resistor dan Kapasitor. Pada perkembangan selanjutnya Sistem kendali PID dapat diimplementasikan secara digital dengan bantuan program (*software*) yang dijalankan oleh komputer. Komputer mampu melakukan komputasi dengan cepat sehingga perhitungan perhitungan yang rumit pada sistem kendali PID dapat dikerjakan, namun untuk penerapan otomatisasi pada industri (dilandang) sangat sulit untuk dilakukan karena harus menggunakan komputer untuk menjalankan program, sehingga menjadi tidak fleksibel dalam penggunaannya. Untuk itu perlu dicari alternatif pengganti komputer untuk penerapan PID digital.

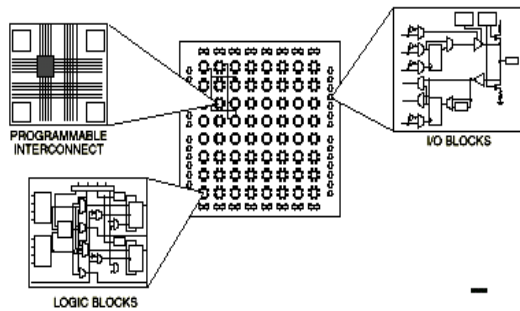
Penerapan sistem kendali PID digital memerlukan *hardware* pendukung yang bagus, *hardware* tersebut harus memiliki kemampuan untuk merespon sinyal masukan dan keluaran dengan cepat, mampu melakukan komputasi secara cepat, memiliki jumlah kaki masukan dan keluaran yang banyak, membutuhkan sumber tegangan yang rendah, memiliki daya tahan yang tinggi, dan proses implementasinya mudah. Tuntutan akan kecepatan proses dan unjuk kerja pengendali yang handal mendorong untuk mengimplementasikan sistem kendali PID digital dalam bentuk untai logika perangkat keras (*hardware logic*). Operasi dalam bentuk perangkat keras mempunyai kecepatan lebih tinggi dibandingkan dengan operasi yang dilakukan secara perangkat lunak oleh mikrokontroler, hal ini dikarenakan operasi dengan perangkat lunak membutuhkan waktu untuk menerjemahkan perintah-perintah program.

Implementasi operasi-operasi logika digital dalam bentuk perangkat keras dapat dilakukan dengan *Field Programmable Gate Array* (FPGA). FPGA di dalamnya memuat ribuan gerbang logika yang dapat diprogram untuk membentuk suatu operasi logika. FPGA dapat digunakan untuk mengimplementasikan sistem kombinasional dan sekuensial berkecepatan tinggi dengan lebar jalur data yang tidak terbatas. Hal ini membuat FPGA mampu melakukan operasi dengan tingkat keparalelan tinggi yang tidak mungkin dilakukan oleh mikrokontroler. Metode yang digunakan untuk mengimplementasikan operasi logika dengan FPGA yaitu dengan metode *Schematik* (berupa gambar untai elektronika digital dari perangkat lunak) maupun dengan bahasa pemrograman VHDL (*Very High Speed Hardware Diskription Language*).

FPGA

FPGA adalah sekumpulan array tak terhubung (*unconnected array*) yang tersusun atas gerbang logika. Adapun arsitektur internal FPGA bersifat seperti sekering (*fuse*) berbentuk array yang dapat diputus agar terbuka atau dibiarkan agar tertutup untuk menghubungkan berbagai *input* ke masing-masing gerbang.

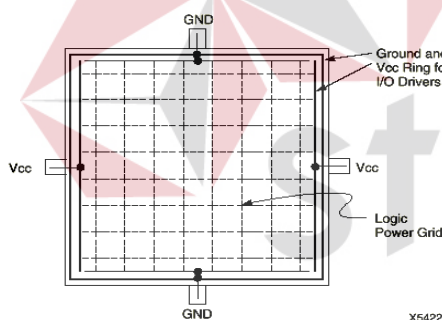
Pada dasarnya FPGA menonjolkan suatu arsitektur *gate-arrays* yaitu suatu matriks sel logika yang dikelilingi oleh banyak sel I/O, seperti yang diperlihatkan pada Gambar 1. *Segment metal* pada *interconnect* dapat difungsikan sebagai *programmable switches* yang berguna untuk membentuk suatu jaringan sinyal antar sel.



Gambar 1. Arsitektur FPGA

Dengan kehadiran FPGA kita tidak akan direpotkan lagi dengan masalah komponen hal ini dikarenakan FPGA merupakan kombinasi dari gerbang logika, register dan I/O yang didukung dengan kecepatan sistem yang cepat.

Daya dari FPGA dibagikan melalui suatu jaringan untuk menghasilkan kekebalan derau tinggi dan pemisahan antara logika dan I/O. Di dalam FPGA, V_{CC} dan $ground$ yang melingkar mengelilingi array logika menyediakan daya ke *driver* I/O, seperti pada Gambar 2 *Power Distribution*. Sebuah matrix V_{CC} independen dan $line\ ground$ menyediakan interior logika pada *device*, untuk lebih jelasnya dapat dilihat pada Gambar 2.



Gambar 2. Power Distribution

Jaringan Pembagi Daya ini memberikan catu stabil dan latar untuk semua logika internal, menyediakan paket pin *power eksternal* yang berhubungan dan *decoupled* yang sesuai. Biasanya, sebuah kapasitor 0,1 μF menghubungkan tiap V_{CC} dan *board* dari *Ground* akan menyediakan *decoupling* yang memadai.

Buffer output mampu men-drive atau menyusutkan arus sampai 12 mA dan pada kondisi terburuk *buffer output* mampu untuk men-drive atau menyusutkan arus hingga 10 kali lipat. Derau dapat dikurangi dengan meminimalkan beban kapasitan *eksternal* dan secara bersamaan mengurangi *output* transisi pada arah yang sama. Akan menguntungkan juga apabila beban *buffer* keluaran yang berat dekat pada latar. Blok I/O *buffer* keluaran mempunyai sebuah *slew-rate* pada *mode* terbatas (*default*) yang sebaiknya digunakan bilamana kecepatan naik dan turunnya *output* tidak pada titik kritis.

FPGA mempunyai dua konfigurasi elemen penting yaitu: *Configurable Logic Block* (CLB) dan *Input Output Block* (IOB). Adapun fungsi CLB adalah menyediakan elemen yang berfungsi untuk membentuk *logic user*, sedangkan IOB berfungsi untuk menyediakan antarmuka antara pin paket dan garis sinyal *internal*. Tiga tipe rangkaian lain yang tersedia antara lain : *Buffer* dengan tiga kondisi (TBUF) untuk menangani jalur panjang horizontal yang bekerja sama dengan tiap CLB. Lebar bus data *decoders* disediakan untuk mengelilingi setiap perangkat dan di dalamnya sudah terdapat osilator *internal*. *Programmable interconnect resource* menyediakan *path routing* untuk menghubungkan *input* dan *output* dari *configurable* elemen tersebut dengan jaringan yang sesuai.

PID DIGITAL

Pengendali PID (*Proporsional Integral Derivative*), merupakan gabungan dari tiga sistem kendali yang bertujuan untuk mendapatkan keluaran dengan *risetime* yang tinggi dan galat yang kecil. Seperti yang kita ketahui bahwa sistem kendali *proporsional* memiliki keunggulan yaitu *risetime* yang cepat tetapi sangat rentan dengan *overshot/undershot*, sedangkan sistem kendali *diverensial* memiliki keunggulan untuk memperkecil *delta error* atau meredam *overshot/undershot*. PID berdasarkan implementasinya dibedakan menjadi analog dan digital, PID analog diimplementasikan dengan komponen elektronika *resistor*, *capacitor*, dan *operational amplifier*, sedangkan PID digital diimplementasikan secara program.

PID digital pada dasarnya merupakan suatu proses dari suatu program yang dijalankan dengan menggunakan komputer. Dalam prosesnya nilai yang kita masukkan (*setting point*), dan nilai hasil pembacaan sensor saat ini (*present value*) diproses sehingga galat yang didapatkan sama dengan 0 (nol), atau nilai *setting point* sama dengan *present value*. Untuk dapat mengimplementasikan kendali PID pada sistem digital, maka PID harus diubah kedalam persamaan diskrit.

$$V_0 = K_p e + K_i \int e dt + K_d \frac{de}{dt} \quad (1)$$

Kemudian diturunkan

$$\frac{dV_0}{dt} = K_p \frac{de}{dt} + K_i \frac{d}{dt} \left(\int e dt \right) + K_d \frac{d^2 e}{dt^2} \quad (2)$$

$$\frac{dV_0}{dt} = K_p \frac{de}{dt} + K_i e + K_d \frac{d}{dt} \left(\frac{de}{dt} \right) \quad (3)$$

hasil turunan pada rumus 10, selanjutnya dikalikan dengan *time sampling* (T_s), sehingga didapatkan:

$$\frac{\Delta V_0}{T_s} = K_p \frac{\Delta e}{T_s} + K_i e + K_d \frac{\Delta}{T_s} \left(\frac{\Delta e}{T_s} \right) \quad (4)$$

$$\Delta V_0 = K_p \Delta e + K_i e T_s + K_d \Delta \left(\frac{\Delta e}{T_s} \right) \quad (5)$$

Harga ΔV_0 merupakan harga perubahan keluaran yang didapat dari keluaran sekarang dikurangi dengan keluaran sebelumnya.

$$\Delta V_0 = V_{0n} - V_{0n-1} \quad (6)$$

begitu juga pada perubahan galat

$$\Delta e = e_n - e_{n-1} \quad (7)$$

Sehingga persamaannya berubah menjadi

$$V_0 - V_{0n-1} = K_p(e_n - e_{n-1}) + \frac{K_d}{T_s}(\Delta e_n - \Delta e_{n-1}) \quad (8)$$

Pada kondisi akhir, perubahan Δ pada galat sebelumnya dapat didistribusikan menjadi :

$$\Delta e_{n-1} = e_{n-1} - e_{n-2} \quad (9)$$

Kemudian didistribusikan ke dalam persamaan menjadi

$$V_0 - V_{0n-1} = K_p(e_n - e_{n-1}) + K_i e T_s + \frac{K_d}{T_s}[(e_n - e_{n-1}) - (e_{n-1} - e_{n-2})] \quad (10)$$

Hasil akhir dari persamaan PID adalah sebagai berikut:

$$V_0 = V_{0n-1} + K_p(e_n - e_{n-1}) + K_i e T_s + \frac{K_d}{T_s}(e_n - 2e_{n-1} + e_{n-2}) \quad (11)$$

Dimana:

V_0 = Output

V_{0n-1} = Output Sebelumnya

K_p = Konstanta Proporsional

K_i = Konstanta Integral

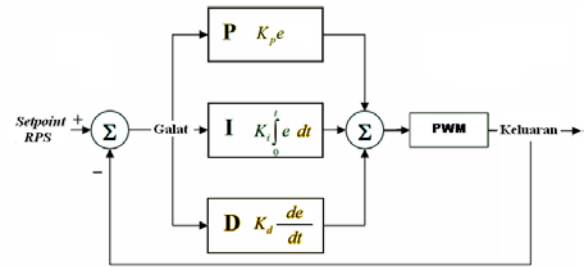
K_d = Konstanta derivative

e_n = Galat sekarang

e_{n-1} = Galat satu kali sebelumnya

e_{n-2} = Galat dua kali sebelumnya

T_s = *Time sampling*



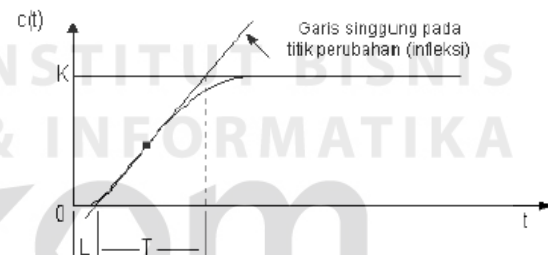
Gambar 3. Blok Diagram PD Digital

Aspek yang sangat penting dalam desain kendali adalah penentuan parameter kendali supaya sistem kalang tertutup memenuhi kriteria performansi yang diinginkan (Wicaksono, 2004). Adapun metode *tuning* kendali PID yang digunakan adalah Ziegler-Nichols. Menurut Ziegler-Nichols, harga harga K_p , K_i , dan K_d ditentukan oleh kurva proses reaksi dimana sistem dijalankan secara *open loop* (Thiang, 2000), seperti yang terlihat pada Gambar 4.

Dimana:

L = Time Lagging

T = Time Konstan



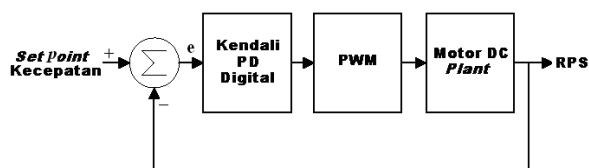
Gambar 4. Kurva Respon Motor

Perhitungan *Auto Tuning* PID menurut Ziegler-Nichols dapat dicari dengan persamaan di bawah ini:

$$K_d = T_d = \frac{L}{2} \quad (12)$$

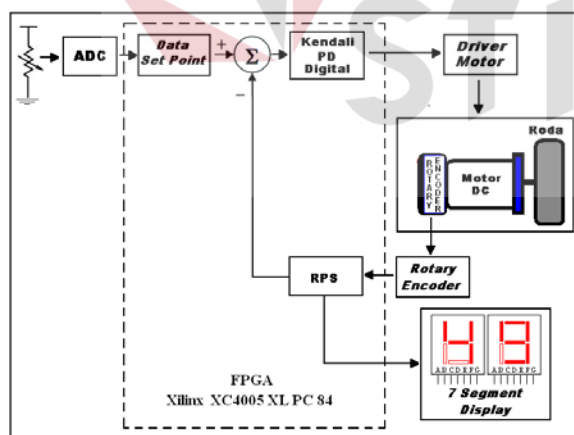
METODE

Sistem kendali kecepatan motor DC dengan algoritma PID digital tersusun atas beberapa bagian, dimana setiap bagiannya terdiri dari beberapa sub bagian yang memiliki fungsi yang saling berkait. Blok diagram sistem kendali kecepatan motor DC dengan algoritma PID digital terlihat pada Gambar 5.



Gambar 5. Blok Diagram Sistem

Sub bagian pada sistem yang dirancang adalah: *setpoint*, pembacaan kecepatan putar roda (RPS), dan unit sistem kendali PID digital. Diagram blok keseluruhan sistem perangkat keras dan perangkat lunak dapat dilihat pada Gambar 6. Pada Gambar 6, bagian yang berada di dalam kotak garis putus-putus merupakan perangkat lunak yang akan dirancang dan diimplementasikan dengan FPGA, sedangkan bagian di luarnya merupakan perangkat keras yang harus dibuat.



Gambar 6. Pemetaan Implementasi Sistem

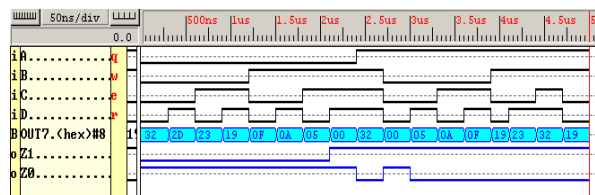
Pengujian

Pengujian dilakukan dengan cara melakukan pengujian per bagian secara simulasi pada bagian kendali PD yang didesain dan implementasi algoritma kendali PD pada FPGA.

HASIL DAN PEMBAHASAN

Simulasi Sub Unit Setpoint

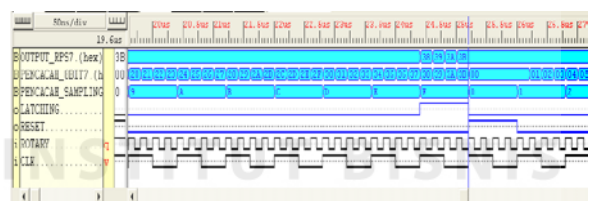
Sub unit *setpoint* berfungsi untuk memberikan masukan berupa besar nilai RPS yang dikehendaki oleh sistem. Sub unit *setpoint* menerima masukan dari ADC dan kemudian dikonversikan sesuai dengan nilai yang dikehendaki oleh sistem untuk selanjutnya dijadikan masukan FPGA. Hasil pengujian unit setpoint dapat dilihat pada Gambar 7.



Gambar 7. Hasil Simulasi Sub

Simulasi Sub Unit RPS

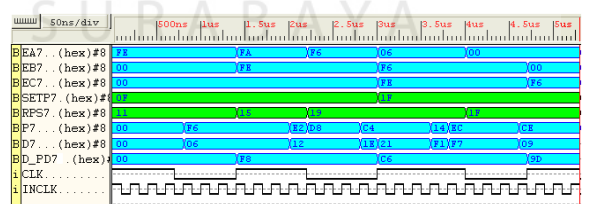
Sub unit ini terdiri dari beberapa bagian penyusun yaitu pencacah delapan *bit*, pencacah sampling dan *latching*. Hasil pengujian dari sub unit RPS dapat dilihat pada Gambar 8.



Gambar 8. Hasil Simulasi Sub Unit RPS

Simulasi Sub Unit PID

Hasil pengujian sub unit kendali PID terlihat pada Gambar 9. Pada Gambar 9, yang menyatakan hasil kendali PID adalah D_PD7. Sedangkan EA7, EB7 dan EC7 adalah error t, t-1 dan t-2.



Gambar 9. Hasil Simulasi Sub Unit Kendali PID

SIMPULAN

Dari hasil pengujian dan analisis yang telah dilakukan pada sistem kendali kecepatan pada motor DC dengan metode PID digital menggunakan FPGA, maka dapat ditarik kesimpulan bahwa sistem kendali kecepatan pada motor DC dengan metode PD digital dapat dirancang dan diimplementasikan secara perangkat keras menggunakan FPGA Xilinx XC4005XL, berdasarkan file laporan (file *report*) jumlah CLB yang terpakai adalah 196 atau 100% dari total jumlah CLB yang terdapat pada XC4005XL-PC84 dan jumlah pin I/O (IOBs) sebanyak 23 buah atau 35% dari kapasitas maksimalnya, hal ini mengindikasikan bahwa penggunaan gerbang logika telah dirancang secara optimal.

DAFTAR RUJUKAN

- Bejo, A, 2004, Kendali Kecepatan Motor Induksi Satu Fase Dengan Inverter PWM Berbasis Mikrokontroler dan FPGA. UGM, Yogyakarta.
- Chan, P.K., & Mourad, S. 1994, *Digital Design Using Field Programmable Gate Arrays*, Prentice-Hall, New Jersey. United States America.
- Gendroyono, P. 1999. Sistem Penggerak Motor Induksi Dengan Beban Berubah Menggunakan Inverter PWM Berbasis Mikrokontroler. UGM, Yogyakarta.
- Kusumawardani, S. 2001. Implementasi Sandi BCH (15,5) dengan FPGA XC4013. UGM, Yogyakarta.
- Lee, Samuel. C., 1976, *Digital Circuit and Logic Design*, Prentice-Hall, New Jersey.
- Partolo, R. 2004. Aplikasi Mikrokontroler 80C31 Sebagai Alat Pengendali Motor DCMagnet Permanen Dengan Metode PID dan Metode PWM. Jurnal Teknologi Elektro Vol.3 No.2, Juli 2004.
- Pasila, F., & Thiang, & Finaldi, O. 2000. Pengaturan Kecepatan Motor DCMenggunakan Kendali Hybrid PID-Fuzzy. *Industrial Electronic Seminar* 2000, Oktober 2000.
- Siahaan, I.H., & Sutantara, I.N. 2005. Simulasi Model Matematis Kontrol Sistem Traksi. Jurnal Teknik Mesin Vol.7 No.1, April 2005: (Online), (URL: <http://puslit.petra.ac.id/~puslit/journals/articles.php?PublishedID=MES05070106>, diakses 11Maret 2008).
- Sutikno, T. 2004. Pembangkit Sinyal PWM Sinusoida Dua Fasa Berbasis FPGA. UGM, Yogyakarta.
- Sudarsono, B., & Dewanto, J.2003. Pemodelan Sistem Gaya dan Traksi Roda. Jurnal Teknik Mesin Vol.5 No.2, Oktober 2003: (Online), (URL: <http://www.petra.ac.id/~puslit/journals/pdf.php?PublishedID=MES03050204>, diakses 27 November 2008).
- Tocci, R.J., 1985, *Digital Systems Principles and Applications*, third Edition, Prentice-Hall, New Jersey.
- Xilinx Corporation. 1999. *The Programmable Logic Data Book.2100 Logic Drive*. San Jose. California 95124. United States America