
SINKRONISASI KECEPATAN PUTARAN ELEKTRONIK *SOFT SWITCH* GEAR BERBASIS PID KONTROL PADA MOBIL HIBRIDA

SPEED SYNCHRONIZATION OF ELECTRONIC *SOFT SWITCH* GEAR BASED ON PID CONTROL FOR HYBRID CAR

Dedid Cahya Happyanto

Departemen Elektro Politeknik Elektronika Negeri Surabaya

Jl. Raya ITS Keputih Sukolilo Surabaya

Telp : (031) 5947280 Fax : (031) 5946114

E-mail: dedid@eepis-its.edu

ABSTRAK

Pada sistem penggerak mobil hibrida dalam penelitian digunakan penggerak motor elektrik dengan motor induksi tiga fasa dan penggerak mesin pembakaran dalam berbahan bakar minyak (BBM) sebagai penggerak cadangan. Untuk perpindahan kedua penggerak tersebut diperlukan sistem perpindahan transmisi penggerak yang lembut pada saat transisi perpindahan transmisi kedua penggerak tersebut. Untuk itu diperlukan teknik kendali pada elektronik switch gear pada transmisi penggerak mobil. Teknik kendali yang digunakan dengan kontrol PID untuk mensinkronkan kedua kecepatan putaran penggerak, sehingga pada saat perpindahan kedua transmisi dihasilkan switch gear yang soft. Langkah yang dilakukan adalah dengan pengaturan kecepatan putaran mesin BBM dan putaran kecepatan motor elektrik. Mesin BBM sebagai Master dan motor elektrik sebagai Slave karena pengaturan motor elektrik lebih mudah dari pada pengaturan mesin BBM, sehingga yang menyesuaikan kecepatan transmisi adalah motor elektrik. Pada saat akan dilakukan perpindahan transmisi kecepatan motor akan menyesuaikan kecepatan mesin BBM dan apabila sudah dicapai sinkronisasi dengan teknik kendali PID maka dilanjutkan dengan perpindahan transmisi. Penggerak utama mobil hibrida ini adalah motor elektrik, apabila terjadi kelemahan daya pada baterai, dilakukan pemindahan transmisi ke penggerak BBM dan secara bersamaan dilakukan pengisian baterai sampai penuh. Apabila pengisian baterai sudah penuh maka dilakukan perpindahan transmisi ke penggerak motor elektrik. Hasil simulasi menunjukkan bahwa kendali PID sesuai untuk diterapkan pada mobil hibrida ini karena mempunyai respon yang baik dan bekerja secara autotuning pada perubahan kecepatan mobil.

Kata Kunci: motor induksi tiga fasa, kontrol PID, elektronik switch gear, autotuning.

ABSTRACT

Hybrid car on the drive system used in the research with the electric motor driving a three-phase induction motors and propulsion engine fuel oil as a backup drive. For the second movement of the propulsion system required a gentle drive transmission shift during the transition transmission shift both the activator. It is necessary for the control of electronic engineering at Switch Gear drive transmission car. Control techniques are used to synchronize the two PID control rotation speed drive, so that when the second transfer switch gear transmission that produced soft. Step is to do with setting the fuel and the engine rotation speed of rotation speed of the electric motor. Gasoline engine and electric motor as the Master as a slave because the electric motor setup easier than setting fuel engine, so that adjusts the transmission rate is the electric motor. At the time of transfer will be done will customize a transmission speed of the motor fuel and engine speed is achieved when synchronization with PID control technique then continued with the transmission shift. The main driver of this car is a hybrid electric motor, in case of weakness on battery power, the new displacement to drive fuel transmission and battery charging at the same time made up with a full battery charge. When charging the battery is full then do shift the transmission to drive the electric motor. The test results showed that the corresponding PID controller is applied to the hybrid car because it has a good response and work on changing the speed of the car Autotuning.

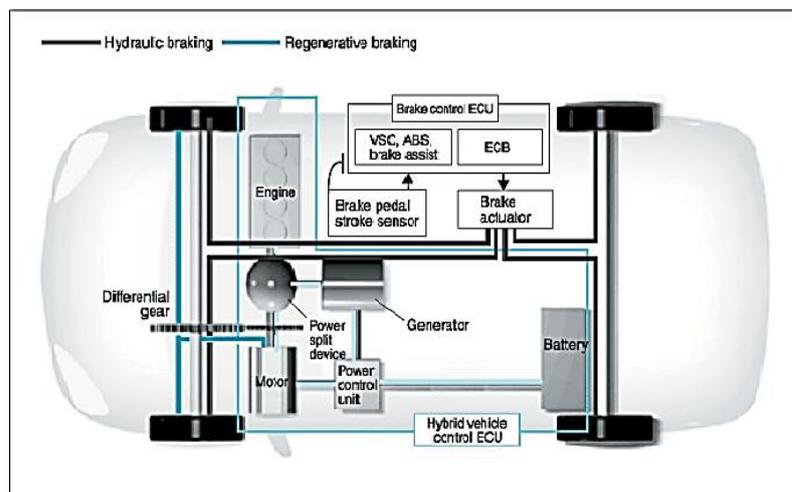
Keywords: three-phase induction motors, PID control, electronic switch gear, autotuning.

1 PENDAHULUAN

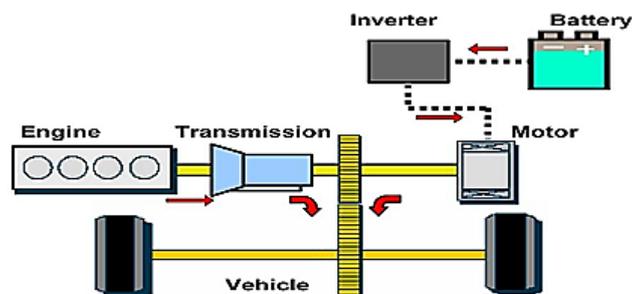
Pada saat ini bumi kita telah mengalami proses *global warming*. Hal ini dikarenakan karena banyaknya polusi udara yang ada. Polusi udara ini salah satunya berasal dari gas buang kendaraan bermotor. Oleh karena itu beberapa produsen mobil mulai melakukan perbaikan untuk mengantisipasi

kondisi tersebut. Beberapa produsen telah merancang sistem hibrida pada kendaraan bermotor, yaitu kendaraan yang menggabungkan mesin konvensional yang menggunakan bahan bakar minyak (mobil BBM) dan motor listrik yang disuplai arus listrik dari baterai atau aki. Dengan menggunakan sistem ini diharapkan dapat mengurangi emisi gas buang serta meningkatkan efisiensi bahan bakar. Penggunaan sistem hibrida pada kendaraan bermotor telah dimulai pada akhir tahun 1990-an ditandai dengan masuknya produsen ke pasar mobil hibrida. Pada kendaraan roda empat sistem hibrida telah diterapkan pada beberapa merk mobil, seperti Honda Insight (2005) dan Toyota Prius (2009). Sistem hibrida ini diharapkan mampu meningkatkan efisiensi bahan bakar, mengurangi emisi gas buang, memberikan performa dan akselerasi yang tinggi. Sementara itu, sistem hibrida yang digunakan pada kendaraan roda dua baru di pasaran oleh Fuboru (2009) yang berasal dari Cina yang memiliki spesifikasi penggerak mesin bensin 50 cc dan motor listrik 36 V 12 A. Kendaraan tersebut memiliki keunggulan yaitu mempunyai efisiensi bahan bakar yang tinggi serta emisi gas buang yang rendah. Namun kendaraan ini masih memiliki kekurangan yaitu belum *full hybrid* karena beberapa mekanisme masih menggunakan sistem manual baik sistem pengisian baterai maupun perpindahan mesin ke elektrik sehingga menyebabkan timbulnya hentakan pada saat perpindahan dari mesin ke elektrik atau sebaliknya.

Pada saat mobil hibrida berhenti yang bekerja adalah mesin bahan bakar, sedangkan generator, dan motor listrik tidak bekerja. Pada saat energi listrik di baterai mulai menipis dan kendaraan sedang berhenti, mesin berbahan bakar minyak (BBM) akan menyala sejenak untuk sedikit mengisi baterai. Mesin BBM memutar generator sehingga generator dapat menghasilkan energi listrik untuk mengisi ulang baterai. Bila kondisi EV (*Electric Vehicle*) Mode yang terdapat pada mobil hibrida maka mobil hanya digerakkan oleh motor listrik saja (maksimum sejauh 45 km jika baterai dalam kondisi penuh) dan kecepatan maksimum 45 km/hour.



Gambar 1. Sistem teknologi mobil hibrida [1]



Gambar 2. Sistem *switch gear* mobil hibrida [3]

Switch gear adalah sebuah mesin yang bekerja untuk memindahkan *gear* pada kendaraan bermotor agar kendaraan tersebut dapat berpindah gigi untuk melanjutkan ke gigi berikutnya untuk menambah kecepatannya. Disini *switch gear* dimaksudkan untuk memindahkan pergantian penggerak dari penggerak mesin ke penggerak motor elektrik atau sebaliknya pada saat kendaraan tersebut sedang berjalan.

2 BAHAN DAN METODE PENELITIAN

2.1 Simulator Elektronik *Switch Gear*

Bahan penelitian ini dirancang dan dibangun simulator untuk menggambarkan dua penggerak mobil, yaitu penggerak dengan mesin BBM (motor 1) dan penggerak motor elektrik (motor 2). Simulator ini terdiri dua bagian yang terdiri dari simulator penggerak dan sistem *programming*, *setting*, dan *monitoring*. Simulator penggerak pada Gambar 3 dan sistem *programming*, *setting*, dan *monitoring* pada Gambar 4.



Gambar 3. Simulator kontrol *switch gear* penggerak mesin dan motor elektrik



Gambar 4. Tampilan monitor simulator kontrol *switch gear*

2.2 Metodologi Penelitian

2.2.1 Teknik Kendali Sikronisasi Dua Motor

Gambar 5 menggambarkan dua sistem yang saling berinteraksi, dan untuk memudahkan dalam penyelesaian koordinasi dua sistem atau lebih. Sistem dinamik yang dinyatakan sebagai berikut:

$$S_1(U_1, Y_1, W_2) = 0 \text{ dan } S_2(U_2, Y_2, W_1) = 0 \quad (1)$$

dimana : $U_i \in \mathbb{R}^{n_i}$ (Riil), $W_i \in \mathbb{R}^{m_i}$ (Riil) dan $Y_i \in \mathbb{R}^{p_i}$ (Riil), untuk $i = 1, 2$. Dengan jumlah vektor output sistem dinamik $p = p_1 + p_2$, jumlah vektor sinyal kontrol sistem dinamik $n = n_1 + n_2$ dan vektor beban yang berinteraksi $m = m_1 + m_2$. dengan p_1, n_1, m_1 : vektor pada motor1; p_2, n_2, m_2 : vektor pada motor2. Konstrain diekspresikan oleh:

$$S_0 = \{(U, Y, W) : S_1 = 0, S_2 = 0\} \quad (2)$$

dimana $U = [U_1^T, U_2^T]^T \in \mathbb{R}^n$ vektor kontrol tambahan, $Y = [Y_1^T, Y_2^T]^T \in \mathbb{R}^p$ adalah vektor output sistem tambahan, $W = [W_1^T, W_2^T]^T \in \mathbb{R}^m$ adalah vektor yang menggambarkan beban interaksi antara dua sistem. Biasanya, fungsi *cost* dari sistem berganda adalah penjumlahan dari fungsi *cost* semua komponen sistem:

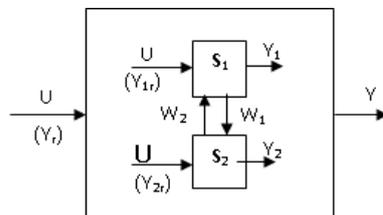
$$J(U, Y, W) = J_1(U_1, Y_1, W_2) + J_2(U_2, Y_2, W_1) \quad (3)$$

Masalah koordinasi sistem berganda dapat diterapkan dalam masalah optimasi : yaitu mendapatkan nilai fungsi *cost*, J , yang minimum dengan mengacu pada konstrain S_0 . Meskipun tidak ada pendekatan yang

umum untuk memecahkan masalah pada sistem berganda yang kompleks, namun beberapa konsep dan prinsip dasar koordinasi dapat ditunjukkan. Yang dinamakan model koordinasi. Dalam metode ini, masalah dibagi dalam dua level masalah optimasi. Pertama, perkiraan interaksi W tetap pada Z . Kemudian hitung nilai $H(z)$, $H(Z)$ kemudian diselesaikan secara iterasi hingga performansi yang diinginkan tercapai, yaitu:

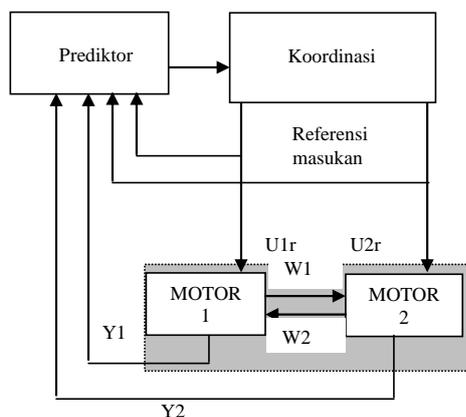
$$H(Z) = \min[J(U, Y, Z)] \text{ untuk } (U, Y, Z) \in S_0 \quad (4)$$

Interaksi kecepatan antara dua motor disampaikan pada Gambar 5.



Gambar 5. Interaksi dua sistem koordinasi [3]

Prinsip dasar koordinasi pada Gambar 6, dikatakan prinsip prediksi interaksi yang ditetapkan seperti dibawah ini. Diberikan $\hat{W} = [\hat{W}_1^T, \hat{W}_2^T]^T$ prediksi beban interaksi dan $W = [W_1^T, W_2^T]^T$ aktual beban interaksi dibawah sinyal kontrol U . Kemudian semua kondisi optimum akan dicapai jika prediksi memberikan nilai yang sama $\hat{W} = W$. Dari penjelasan diatas, pemecahan permasalahan optimasi tergantung pada struktur dan parameter dinamik dari subsistem level terendah (S_1 dan S_2 belum berinteraksi) dan bagaimana masing-masing subsistem berinteraksi yang dinyatakan secara matematik.



Gambar 6. Blok diagram sistem koordinasi

Dalam sistem hirarki yang sudah sesuai sebagaimana diharapkan, penambahan koordinator mestinya tidak mempengaruhi terhadap struktur internal atau parameter pada subsistem level terendah. Koordinator seharusnya hanya memberikan perintah-perintah koordinasi yang sesuai terhadap subsistem, sehingga setiap subsistem dapat didesain secara independen terhadap subsistem yang lain. Semakin tinggi tingkatannya semakin bagus hasil yang didapatkan.

Dalam struktur hirarki, masing-masing sub sistem ditunjukkan oleh pemetaan dari masukan referensi ke output. Kontroler masing-masing sub-sistem didisain secara terpisah dan tidak bergantung satu dengan yang lain. Agar tidak mengganggu struktur internal atau parameter sub sistem, hanya variabel kontrol yang menjadi masukan referensi dari sub-sistem.

Tujuan koordinasi adalah untuk memilih vektor referensi masukan yang sesuai agar terjadi minimisasi *error*. Memperkirakan prediksi dari hubungan keluaran utama pada masing-masing pilihan keluaran referensi yang tersedia, dan konstrain yang diekspresikan oleh penetapan aturan-aturan yang dihasilkan. Kemudian, dalam interval sampling tertentu, performansi yang diinginkan dapat diberikan oleh percobaan secara iterasi masukan beban yang berbeda dan mengaturnya menurut prediksi keluaran utama.

2.2.2 Teknik Kendali Elektronik Switch Gear

Teknik kendali yang digunakan dalam penelitian ini dengan menggunakan Proporsional Integrator Derivatif (PID). Kendali PID dapat dinyatakan dengan persamaan diferensial [5]:

$$m(t) = K_p \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right] \tag{5}$$

dimana $e(t) = r(t) - c(t)$, $r(t)$ adalah set point, $c(t)$ adalah variabel proses dan $m(t)$ adalah variabel manipulasi. Untuk mengimplementasikan pengontrol PID dengan perangkat digital, maka harus dilakukan konversi dari bentuk kontinu ke bentuk diskrit. Dengan melihat bahwa:

$$\frac{df}{dt} |_k = \frac{f_k - f_{k-1}}{\Delta t} \quad \text{dan} \quad \int e(t) dt = \sum_{k=1}^n e_k \Delta t \tag{6}$$

Maka persamaan (2) menjadi persamaan PID diskrit:

$$m(n) = K_p \left[T_d \left(\frac{e(n) - e(n-1)}{\Delta t} \right) + e(n) + \frac{1}{T_i} \sum_{k=1}^n e_k \Delta t \right] \tag{7}$$

dimana $m(n)$ menyatakan nilai m pada suatu interval $n\Delta t$, dengan n bilangan bulat. Dengan memperkenalkan parameter:

$$K_i = K_p \left(\frac{T_s}{T_i} \right) \tag{8}$$

$$K_d = K_p \left(\frac{T_d}{T_s} \right)$$

dimana $T_s = \Delta t =$ waktu sampling dari input ADC, maka persamaan (3) dapat dinyatakan dalam sebuah algoritma sebagai berikut:

$$s(n) = s(n-1) + e(n) \tag{9}$$

$$m(n) = K_p e(n) + K_i s(n) + K_d [e(n) - e(n-1)] \tag{10}$$

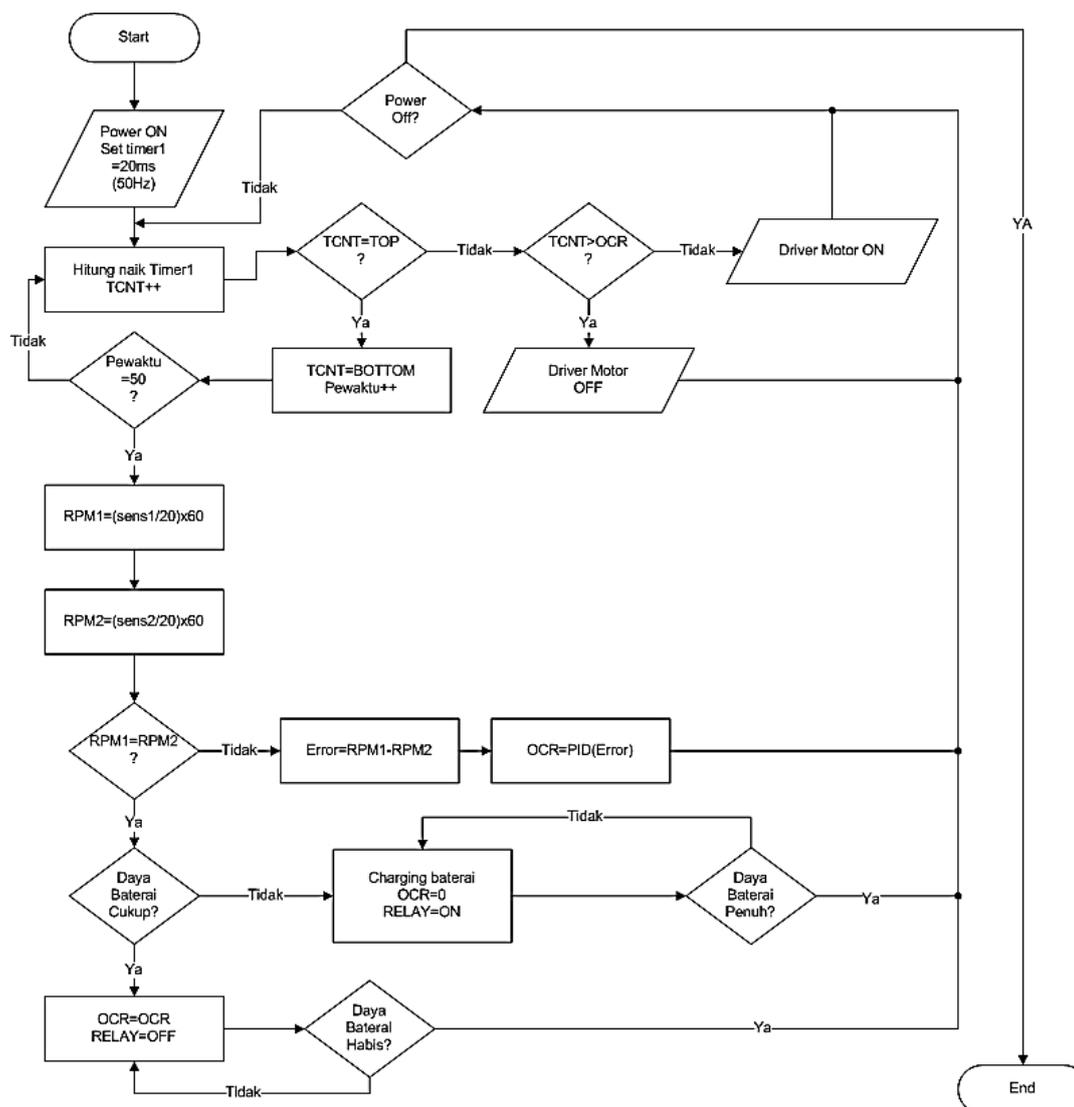
dimana $s(n) =$ jumlah seluruh nilai error pada interval 0 sampai nT_s .

2.2.3 Teknik Kendali dan Struktur Program

Realisasi program kendali PID dilakukan dengan menggunakan struktur program *foreground/background* [3]. Dengan demikian tugas program dapat dibagi dua yaitu *foreground task* dan *background task*. *Foreground task* adalah tugas program yang memerlukan akses waktu sesegera mungkin (*hard time constrains*) dan *background task* adalah tugas program yang tidak memerlukan akses yang sesegera mungkin (*soft time constrains*). Dalam hal perancangan kendali PID ini, *background task* adalah *routine program* utama yang terdiri dari instruksi inisialisasi dan instruksi *NOP* (*no operation*) untuk menunggu terjadinya *interrupt timer* 0. Apabila terjadi *interrupt* maka program utama akan dihentikan dan mikro kontroler langsung mengerjakan *foreground task* berupa *routine interrupt algoritma* PID diskrit. Bagian inisialisasi ditujukan untuk memberi nilai konfigurasi awal mikro kontroler, pendefinisian harga awal, dan pemberian nilai pada parameter kendali PID. Sebagai contoh, untuk mematikan/menghidupkan fasilitas-fasilitas mikro kontroler, seperti *interrupt* dan *timer* dilakukan dengan memberikan nilai yang sesuai pada *SFR*, dalam hal ini *SFR TCON* dan *TMOD*.

Routine interrupt akan mengerjakan *routine algoritma* PID diskrit yang diawali dengan prosedur I/O untuk mengambil nilai SP dan PV. Berdasarkan data input tersebut, akan dilakukan perhitungan algoritma kendali dengan memanggil prosedur PID. Hasil perhitungan akan dikeluarkan sebagai output dengan prosedur I/O. Setelah *routine interrupt* telah selesai dikerjakan, maka program utama akan kembali menjalankan operasi *NOP* secara berulang untuk menunggu terjadinya *interrupt timer* 0. Keuntungan penggunaan struktur *foreground/background* dengan mengaplikasikan *interrupt* dalam perancangan ini berhubungan dengan waktu sampling algoritma kendali PID.

Pada bagian ini telah dijelaskan mengenai algoritma PID dan terlihat bahwa waktu sampling T_s adalah faktor yang sangat penting dalam implementasi persamaan diskrit dari algoritma PID. Waktu sampling adalah waktu yang dibutuhkan oleh kendali untuk mengambil data input dari ADC.

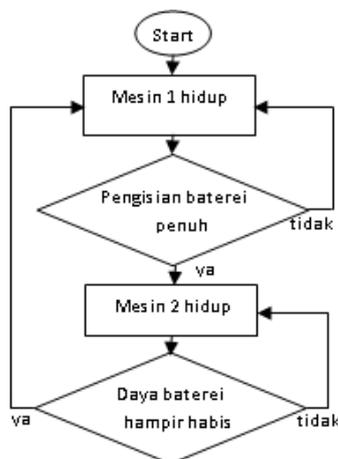


Gambar 7. Flowchart perpindahan sistem switch gear

Dari persamaan terlihat bahwa persamaan PID diskrit (persamaan 7) berlaku untuk waktu sampling yang konstan. Untuk itu perlu digunakan metode untuk mendapatkan waktu sampling konstan yang tidak tergantung pada lama waktu perhitungan PID. Waktu sampling kendali juga diatur agar memiliki nilai yang lebih kecil dari konstanta waktu plant yang akan dikontrol. Dalam hal pengendalian motor DC, dari identifikasi parameter didapat bahwa waktu sampling motor yang dipakai pada penelitian ini adalah kurang lebih 50 ms. Dari pertimbangan diatas, maka digunakan metoda *interrupt timer internal* mikro kontroler. Dengan demikian, pengendali akan dapat melakukan penyamplingan data input setiap terjadi *interrupt timer* pada selang waktu yang tetap dan tidak bergantung dengan lama operasi perhitungan algoritma kontrol.

Timer 0 mode 0 digunakan untuk menghitung waktu sampling. Apabila terjadi *overflow* pada register *TH0* dan *TLO*, maka akan terjadi *interrupt timer 0 (TF0)* sehingga program akan *jump* ke *interrupt service routine* untuk *timer 0*. *Interrupt service routine* akan mengerjakan algoritma kontrol PID yang dimulai dengan pengambilan input SP dan PV dari ADC dan diakhiri dengan mengeluarkan output kendali MV melalui rangkaian *driver*, lihat pada diagram *flowchart* pada Gambar 7.

Apabila tombol *Start* ditekan maka mikro kontroler akan melakukan pembacaan dan akan menghidupkan/menyalakan mesin 1 (mesin bakar). Selanjutnya sensor kecepatan 1 dan sensor kecepatan 2 menerima data percepatan dari putaran motor di mesin 1. Kemudian data kecepatan yang diterima pada sensor kecepatan 2 dimasukkan ke mesin 2 dan menghidupkan mesin 2 (motor elektrik). Selanjutnya akan diadakan komparasi kecepatan antara mesin 1 dan mesin 2, apabila dicapai kecepatan yang sama, maka solenoid akan melakukan *switching*, sehingga perpindahan *gear* dapat dilakukan.



Gambar 8. Flowchart pergantian sistem *switch gear* dengan daya baterai

Saat mikro kontroler melakukan pembacaan data dari sensor kecepatan 1 (sensor kecepatan menggunakan *rotary encoder*) dan sensor kecepatan 2 akan ikut merespon, lalu kemudian sistem pengisian baterai akan mulai mengisi saat mendapatkan pembacaan bahwa mesin 1 bekerja, *flowchart* pergantian sistem *switch gear* dengan daya baterai dilihat pada Gambar 8.

Setelah baterai terisi hampir penuh (mencapai pengisian baterai 90%), maka mesin 2 akan mulai bergerak, maka mesin 1 dan mesin 2 akan mulai sinkronisasi kecepatan. Setelah mesin 1 dan mesin 2 sinkron, maka proses *switching* akan bergerak dan mesin 2 akan bergerak menggantikan mesin 1 dengan menggunakan tenaga motor elektrik. Mesin 2 akan terus bergerak sampai indikator pada daya baterai hampir habis (daya baterai pada posisi 10%), kemudian mesin 1 akan mulai bergerak kembali dengan otomatis dan akan mulai mensinkronkan kecepatan dengan mesin 2. Setelah mesin 1 dan mesin 2 kecepatannya sinkron maka proses *switching* akan kembali terjadi dan tenaga penggerak akan mulai digantikan dengan mesin 1 kembali, sedangkan mesin 2 akan berhenti.

3 HASIL PENGUJIAN DAN PEMBAHASAN

3.1 Pengujian Sinkronisasi *Switch Gear*

Pada pengujian sinkronisasi *switch gear*, pertama-tama dilakukan penyalaan *power supply* dan dilakukan koneksi serial pada komputer. Setelah itu akan dilakukan *scanning* pada port yang akan digunakan pada komputer, sehingga mikro kontroler akan dapat membaca program yang ada pada visual basic (Gambar 9). Pada saat setelah port yang akan terpakai di komputer telah terscan dan telah dihubungkan, maka indikator baterai akan mulai menghitung turun (*counter down*) baterai, sehingga saat indikator baterai telah mencapai 10% dan mendekati 0, maka mesin 1 akan menyala (Gambar 10). Pada saat setelah motor 1 menyala maka indikator baterai akan naik dalam artian saat motor 1 menyala maka sensor generator akan mulai melakukan *charge* pada baterai dan baterai akan mulai mengisi, sehingga gambar indikator baterai pada *visual basic* akan mulai naik (Gambar 11). Lalu setelah indikator daya baterai mulai penuh, maka mesin kedua akan menyala dan baterai akan digunakan, karena mesin 2 adalah motor elektrik, sehingga apabila mesin 2 hidup, maka daya baterai akan digunakan lagi (Gambar 12). Pada saat energi baterai digunakan hingga indikator pada daya baterai mendekati 10%, maka pada saat itu juga, mesin 1 akan hidup secara otomatis dikarenakan sensor pada pengisi baterai telah mendeteksi bahwa baterai akan habis dan mendekati 10%, sehingga mesin 1 akan hidup secara otomatis (Gambar 13).



Gambar 9. Tampilan dari program simulator sinkronisasi *switch gear*



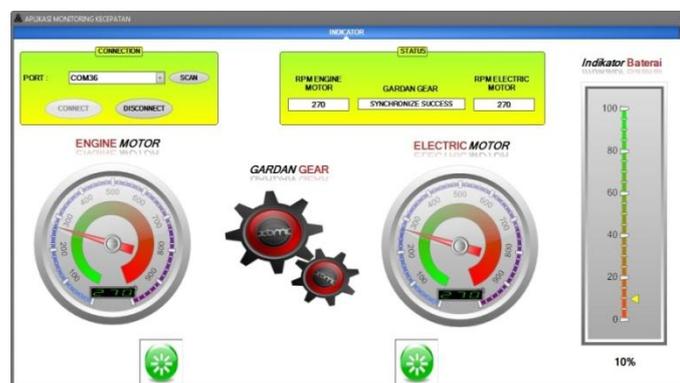
Gambar 10. Tampilan pada program saat mesin 1 hidup



Gambar 11. Tampilan saat indikator daya baterai mendekati penuh dan proses sinkronisasi



Gambar 12. Tampilan saat mesin 2 menyala dan energi baterai terpakai.



Gambar 13. Tampilan saat daya baterai 10% dan proses sinkronisasi untuk kembali ke mesin 1

Jadi setelah baterai yang telah akan terkuras habis dan mendekati 10% kembali menggunakan mesin 1, maka akan terjadi *cycling energy* yang akan menghemat penggunaan energi bensin, sehingga bahan bakar energi untuk daya motor 1 akan hemat terpakai.

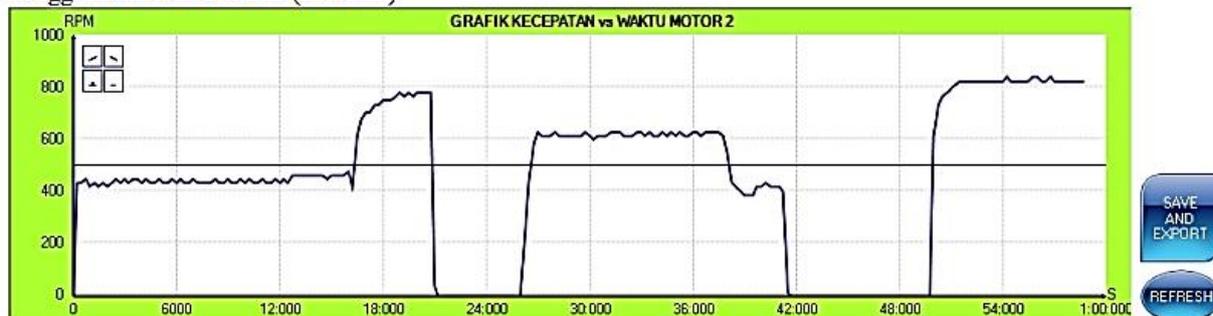
3.2 Pengujian Parameter PID (K_p , K_i , K_d)

Pengujian ini dilakukan untuk mengetahui hubungan antara variasi nilai K_p , K_i , K_d terhadap output sistem. Pemberian nilai K_p , K_i , K_d yang tepat akan berpengaruh terhadap penentuan nilai optimal. Percobaan dimulai dengan beberapa variasi nilai K_p , K_i dan K_d sehingga menghasilkan variasi output yang nantinya akan dibandingkan dengan nilai input referensi yang diberikan (Tabel 1). Dari beberapa kali percobaan, didapat data bahwa nilai optimal untuk ketiga komponen K_p , K_i , K_d adalah masing-masing $\pm 0,5$, $\pm 0,5$ dan $\pm 0,5$. Nilai yang paling berpengaruh pada alat simulasi yang dirancang adalah nilai K_p dimana semakin rendah nilai *error*, maka *switching system* akan semakin cepat untuk melakukan *switching*. Kemudian diimbangi dengan pemberian nilai K_i dan K_d yang tepat, maka didapatkan nilai sistem kendali PID yang optimal. Adapun nilai K_p sebesar 0,5 untuk memberikan kestabilan lebih tinggi di saat nilai K_i , dan K_d semakin kecil, sehingga *switch gear* dapat bekerja secara optimal.

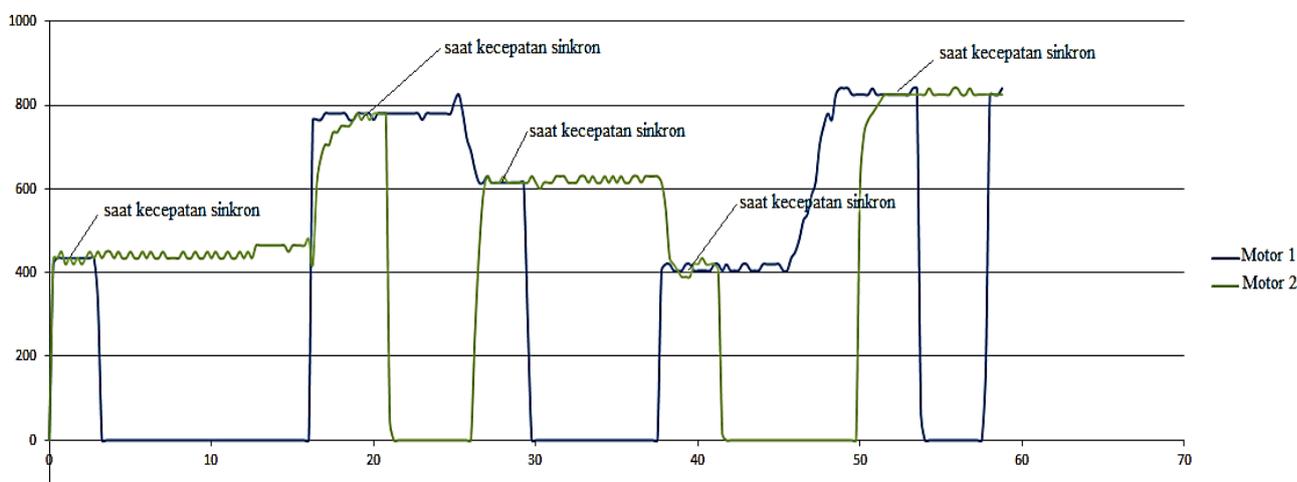
Penggerak Engine (motor 1)



Penggerak motor elektrik (motor 2)



(a)



(b)

Gambar 14. Hasil pengujian saat proses sinkronisasi mesin 1 dan mesin 2: (a) Penggerak mesin dan penggerak motor ditampilkan terpisah; (b) Penggerak mesin dan penggerak motor ditampilkan terpadu

Tabel 1. Pengujian variasi K_p , K_i , K_d

No	K_p	K_i	K_d	Input (%)	Output (%)
1	0,1	0,1	0,1	100	120
2	0,2	0,1	0,1	100	115
3	0,3	0,1	0,1	100	112
4	0,4	0,1	0,1	100	109
5	0,5	0,1	0,1	100	108
6	0,5	0,1	0,1	100	108
7	0,5	0,2	0,1	100	107
8	0,5	0,3	0,1	100	107
9	0,5	0,4	0,1	100	106
10	0,5	0,5	0,1	100	105
11	0,5	0,5	0,1	100	104
12	0,5	0,5	0,2	100	103
13	0,5	0,5	0,3	100	102
14	0,5	0,5	0,4	100	101
15	0,5	0,5	0,5	100	101

4 KESIMPULAN

Nilai optimal dicapai dari hasil percobaan menggunakan metode kontrol PID ini didapatkan nilai $K_p = \pm 0,5$, $K_i = \pm 0,5$ dan $K_d = \pm 0,5$ untuk *switch gear* yang lembut saat perpindahan mesin 1 (mesin bakar konvensional) ke mesin 2 (motor elektrik) atau sebaliknya berdasarkan daya baterai yang tersisi.

REFERENSI

- [1] B. Conlon, "Comparative analysis of single and combined hibrida electrically variable transmission operating modes," SAE, Warrendale, PA, Tech. Rep. 2005-01-1162, 2005.
- [2] H. Benford and M. Leising, "The lever analogy: A new tool in transmission analysis," SAE, Warrendale, PA, Tech. Rep. 810102, 1981.
- [3] Kawashima, Kiyotaka. Rolling Stability Control Based on Electronic Stability Program for In-Wheel-Motor Electric Vehicle. Tokyo : University of Tokyo. 2009
- [4] Lu, Jianbo. The Roll Stability Control System. United States : Ford Company. 2009.
- [5] W Eddy ; "PID for Line Follower", Chicago Area Robotics Group, 2007
- [6] B. Tabbache, A. Kheloui and M.E.H. Benbouzid, Design and Control of the Induction Motor Propulsion of an Electric Vehicle, IEEE VPPC, 2010
- [7] Dedid Cahya Happyanto, Soebagio, Mauridhi Hery Purnomo, Speed Synchronization of Induction Motor Model for the Transient Response of Electric Car, IJAR, Baku, Azer-Baijan, May 30, 2012.
- [8] Dedid Cahya Happyanto, Soebagio, Mauridhi Hery Purnomo, Hibrida Controller Based Intelligent Speed Synchronization Induction Motor For The Four Wheel Drive of Electric Car, JATIT, Pakistan, July, 2012, Vol. 41.
- [9] G.H. Gelb, N.A. Richardson, T.C.Wang, and B. Berman, "An electromechanical transmission for hibrida vehicle power trains—design and dynamometer testing," SAE, Warrendale, PA, Tech. Rep. 710235, Jan. 11–15, 1971.