

# Pengaruh *Duty Cycle* dan Frekuensi Terhadap Kecepatan Putar Motor DC

**Budihardja Murtianta**

Program Studi Teknik Elektro,  
Fakultas Teknik Elektronika dan Komputer,  
Universitas Kristen Satya Wacana, Salatiga  
budihardja.murtianta@staff.uksw.edu

## Ringkasan

Tulisan ini dibuat untuk mengetahui pengaruh perubahan frekuensi dan *duty cycle* terhadap kecepatan putar motor DC saat terbebani dan saat tidak dibebani yang diterapkan dengan menggunakan sistem PWM (*Pulse Width Modulation*). DC Chopper direalisasikan guna menerapkan sistem PWM. DC Chopper mengubah secara langsung dari *dc* ke *dc* dan biasanya hal ini disebut konverter *dc* ke *dc*. Alat ini disimulasikan dengan motor DC tipe 9V dan IGBT sebagai saklar chopper. Perancangan terdiri dari modul regulator 5V, modul microcontroller AT89S52, modul keypad, modul receiver IR sebagai pendeteksi kecepatan, modul transmitter IR, modul LCD, dan modul DC Chopper + driver. Setiap operasi dilakukan dari sebuah keypad dan besarnya kecepatan motor yang terdeteksi dapat ditampilkan pada LCD. Dari hasil pengujian didapatkan bahwa pemakaian DC Chopper untuk aplikasi motor, IGBT yang dipakai bekerja lebih baik saat mendapat masukan frekuensi lebih besar dari 1KHz. Dari hasil percobaan *duty cycle* 10% frekuensi pensaklaran chopper *dc* adalah >2Khz tanpa beban dan >1Khz dengan beban1 dan beban 2. Pada *duty cycle* 25% dan 50% adalah >3Khz tanpa beban, >2Khz pada beban 1 dan >1Khz pada beban 2. Pada *duty cycle* 75% adalah >2Khz tanpa beban, dan *duty cycle* 90% adalah >800Hz tanpa beban. Semakin tinggi nilai frekuensi maka semakin baik titik kerja dari DC Chopper.

**Kata kunci:** *duty cycle*, chopper, konverter

## 1. Pendahuluan

Motor listrik adalah salah satu komponen utama dalam bidang penggerak. Sistem pengaturan kecepatan motor listrik yang sering digunakan adalah sistem PWM, mengingat pengaturan kecepatan motor sistem PWM efisien. Sistem PWM merubah sinyal DC flat menjadi sinyal DC berfrekuensi. Frekuensi ini mempengaruhi kecepatan putar motor itu sendiri. Terutama saat motor tersebut diberi beban. Dengan latar belakang tersebut pada tulisan ini akan diteliti pengaruh *duty cycle* dengan frekuensi yang bervariasi terhadap kecepatan putar motor. Motor yang digunakan adalah motor DC seri. Rangkaian DC Chopper dipakai guna untuk menerapkan sistem PWM yang akan digunakan. Konverter DC ke DC (DC Chopper) digunakan untuk mengatur kecepatan motor arus searah (DC). DC Chopper adalah sebuah piranti yang mengubah tegangan sumber arus searah (DC) yang tetap menjadi tegangan DC yang arusnya dapat diatur [1]. Penggunaan DC Chopper memiliki keuntungan dalam pengaturan tegangan jangkarnya motor DC tanpa adanya disipasi panas yang berarti. Sebagai saklar dan pengantaran

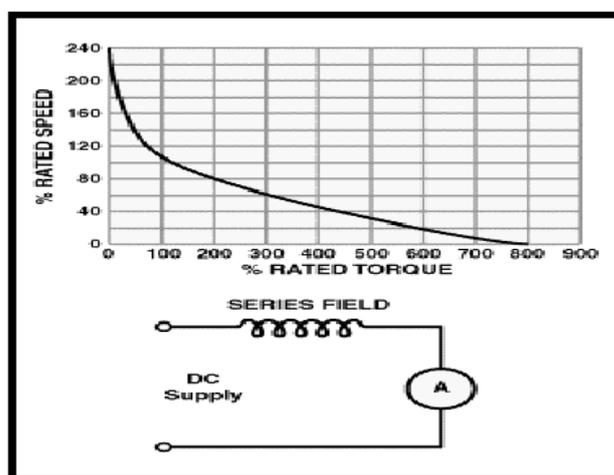
daya dipergunakan IGBT. Pengaruh frekuensi dan *duty cycle* dari DC Chopper terhadap kecepatan putar motor akan diteliti pada tulisan ini.

## 2. Dasar Teori

Pada bagian ini akan dijelaskan masing-masing fungsi dari tiap-tiap komponen penyusun tiap-tiap modul. Tidak seluruh komponen akan dijelaskan satu-persatu tetapi hanya beberapa komponen utama saja yang akan dijelaskan.

### 2.1. Motor DC

Motor DC biasanya digunakan dalam rangkaian yang memerlukan kepresisian yang tinggi untuk pengaturan kecepatan, pada torsi yang konstan. Pada motor seri, gulungan medan (medan *shunt*) dihubungkan secara seri dengan gulungan dinamo (A) seperti ditunjukkan dalam Gambar 1. Oleh karena itu, arus medan sama dengan arus dinamo [2]. Berikut tentang kecepatan motor seri (*Rodwell International Corporation, 1997; L.M. Photonics Ltd, 2002*). Kecepatan putar motor dibatasi pada 5000 RPM



Gambar 1. Karakteristik Motor DC Seri (*Rodwell International Corporation, 1999*)

### 2.2. Pengendali Mikro

Pada tulisan ini dipakai dua pengendali mikro. Pengendali mikro yang pertama berfungsi untuk pengolahan data-data masukan dari *keypad* yang kemudian diolah sehingga menghasilkan sinyal *PWM* dan dikirimkan ke *driver* motor. Pengendali mikro yang kedua digunakan untuk mengolah sinyal yang diterima dari modul pendeteksi kecepatan sehingga sinyal yang diterima dapat ditampilkan dalam satuan *RPM*. Pengendali mikro yang pertama dan kedua yang dipakai adalah pengendali mikro jenis *AT89S52*. *Microcontroller* tipe *Atmel AT89S52* termasuk kedalam keluarga *MCS51* merupakan suatu mikrokomputer *CMOS* 8-bit dengan daya rendah, kemampuan tinggi, memiliki 8K *byte Flash Programmable and Erasable Read Only Memory (PEROM)*. Perangkat ini dibuat menggunakan teknologi memori *nonvolatile*. Set *instruksi* dan kaki keluaran *AT89S52* sesuai dengan standar industri *80C51* dan *80C52*. *Atmel AT89S52* adalah mikrokomputer yang sangat bagus dan fleksibel dengan harga yang rendah untuk banyak aplikasi sistem kendali. Fasilitas yang terdapat dalam *AT89S52* antara lain :

1. Sesuai dengan produk-produk MCS-51.
2. Terdapat memori *flash* yang terintegrasi dalam sistem. Dapat ditulis ulang hingga 1000 kali.
3. Beroperasi pada frekuensi 0 sampai 33MHz.
4. Tiga tingkat kunci memori program.
5. Memiliki 256 x 8 bit RAM.
6. Terdapat 32 jalur masukan/keluaran terprogram.
7. 3 pewaktu/pencacah 6-bit (untuk 52) & 2 pewaktu/pencacah 16-bit (untuk 51).
8. 8 sumber interupsi (untuk 52) & 6 untuk 51.
9. Kanal serial terprogram.
10. Mode daya rendah dan mode daya mati.

*Microcontroller* keluarga MCS 51 memiliki *port-port* yang lebih banyak (40 *port I/O*) dengan fungsi yang bisa saling menggantikan sehingga *microcontroller* jenis ini menjadi sangat digemari karena hanya dalam sebuah *chip* sudah bisa mencakup untuk banyak kebutuhan.[3]

### 2.3. Penampil / Display

LCD adalah sebuah display *dot matrix* yang difungsikan untuk menampilkan tulisan berupa angka atau huruf sesuai dengan yang diinginkan. Digunakan LCD *dot matrix* dengan karakter 2 x 16, sehingga kaki-kakinya berjumlah 16 *pin*. Dalam modul ini menggunakan LCD *character* untuk menampilkan tulisan atau *character* saja. Tampilan LCD terdiri dari dua bagian, yakni bagian *panel LCD* yang terdiri dari banyak "titik" LCD dan sebuah *microcontroller* yang menempel di *panel* dan berfungsi mengatur 'titik-titik' LCD tadi menjadi huruf atau angka yang terbaca. Huruf atau angka yang akan ditampilkan dikirim ke LCD dalam bentuk kode ASCII, kode ASCII ini diterima dan diolah oleh *microcontroller* di dalam LCD menjadi 'titik-titik' LCD yang terbaca sebagai huruf atau angka. Dengan demikian tugas *microcontroller* pemakai tampilan LCD hanyalah mengirimkan kode-kode ASCII untuk ditampilkan.

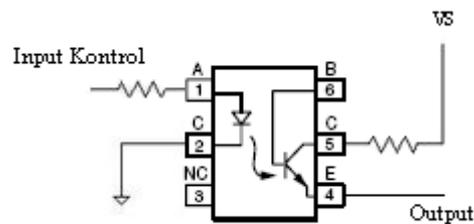
### 2.4. Optocoupler

*Optocoupler* merupakan kombinasi antara sebuah sumber cahaya (*LED*) dengan sebuah *photosensitive detector (phototransistor)*. Fungsi dari *optocoupler* tergantung dari jenisnya. Ada dua jenis *optocoupler*:

1. *Optocoupler* jenis 'U' sering dipakai untuk mengitung putaran dari suatu piringan.
2. *Optocoupler* jenis DIP ( *Dual In - line Package* ) berfungsi untuk melindungi atau mengisolasi rangkaian dari tegangan tinggi.

Pada tulisan ini digunakan *optocoupler* jenis 4N35 yang merupakan *IC driver* berfungsi memindahkan sinyal gelombang dari rangkaian kontrol ke rangkaian daya sekaligus sebagai pemisah tegangan kerja pada rangkaian kontrol dan rangkaian daya. Rangkaian *driver* yang diimplementasikan dengan IC 4N35 ditunjukkan pada Gambar 2.

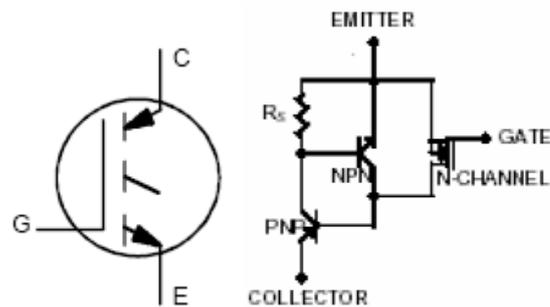
Pada *optocoupler* sisi masukan (kaki 1) dihubungkan dengan tahanan (pembatas arus yang melewati *LED*) dari keluaran rangkaian kontrol, sedangkan kaki 2 dihubungkan ke tegangan nol (*common ground*). Pada sisi keluaran, kaki 5 dihubungkan melalui tahanan ke tegangan sumber dan kaki 4 terhubung dengan beban dan merupakan terminal keluaran IC ini.



Gambar 2. Optocoupler 4N35

## 2.5. IGBT

IGBT (*Insulated-Gate Bipolar Transistor*) adalah piranti semikonduktor sebagai komponen sakelar yang sifatnya setara dengan gabungan sebuah transistor *bipolar (BJT)* dan sebuah *transistor* efek medan (*MOSFET*). Dengan gabungan sifat kedua elemen tersebut, IGBT mempunyai perilaku yang cukup ideal sebagai sebuah sakelar elektronik.[1] Di satu pihak IGBT tidak terlalu membebani sumber dan di pihak lain mampu menghasilkan arus yang besar bagi beban listrik yang dikendalikannya. Dengan demikian, terminal masukan IGBT mempunyai nilai impedansi yang sangat tinggi, sehingga tidak membebani rangkaian pengendalinya yang umumnya terdiri dari rangkaian logika. Ini akan menyederhanakan rancangan rangkaian pengendali (*controller*) dan penggerak (*driver*) dari IGBT. Di samping itu, kecepatan pensakelaran IGBT juga lebih tinggi dibandingkan komponen *BJT*, meskipun lebih rendah dari komponen *MOSFET* yang setara. Di lain pihak, terminal keluaran IGBT mempunyai sifat yang menyerupai terminal keluaran (*collector-emitter*) *BJT*. Dengan kata lain, pada saat keadaan menghantar, nilai tahanan menghantar ( $R_{on}$ ) dari IGBT sangat kecil, menyerupai  $R_{on}$  pada *BJT*. Pada tulisan ini digunakan IGBT jenis *GT15Q101* sebagai driver *dc chopper*nya. IGBT memiliki 3 terminal yaitu *gate*, *collector* dan *emitter*. Simbol dan Arsitektur IGBT dapat dilihat pada Gambar 3.



Gambar 3. Simbol dan Arsitektur IGBT

Karakteristik Switching IGBT:

### 1. IGBT Turn - On

Arus *gate* menentukan kecepatan *turn - on* pada IGBT. Proses *turn - on* IGBT lebih cepat dibandingkan proses *turn-on* pada *BJT*, dan lebih lambat dibandingkan proses *turn - on* pada *MOSFET* karena IGBT harus mengkonduksi *MOSFET* dan *BJT* yang ada didalam IGBT. Untuk membuat IGBT *turn - on*, kapasitan *input* diantara *gate* dan *emitter* diisi dengan tegangan  $V_{GE}$  yang lebih besar dibandingkan dengan tegangan *threshold*  $V_{GE(th)}$ . *GT15Q101* mempunyai  $V_{GE(th)} = 3 - 6$  V.

2. IGBT Turn – Off

Selama *IGBT turn - off*, kapasitansi *gate - emitter* pada masukan akan dikosongkan melalui hambatan seri  $R_{GE}$  antara *gate* dan *emitter*. Nilai minimal  $R_{GE}$  ditentukan berdasarkan suatu nilai dimana ada jaminan *IGBT* mengalami kondisi terkunci untuk berbagai kondisi arus dan tegangan selama pengoperasian termasuk beban resistif maupun induktif. Nilai maksimum  $R_{GE}$  ditentukan berdasarkan pengamatannya pada saat *turn - off*. Pengaturan kecepatan motor DC ini dilakukan oleh *DC Chopper* menggunakan *IGBT* yang berfungsi sebagai saklar. Pulsa *PWM* di biasakan ke kaki *gate* dari *IGBT*, pulsa *PWM* ini berbentuk suatu pulsa *ON / OFF* kotak dengan *duty cycle* yang bervariasi. Lebar pulsa satu gelombang dapat diubah , sehingga lebar pulsa *ON* dan lebar pulsa *OFF* nya tergantung pada lebar pulsa satu gelombang.[1] Dari perbandingan *duty cycle* maka di dapatkan rumus

$$V_{output} = \frac{\text{lebarpulsaON}}{\text{lebarpulsaON} + \text{lebarpulsaOFF}} \times V_s \tag{1}$$

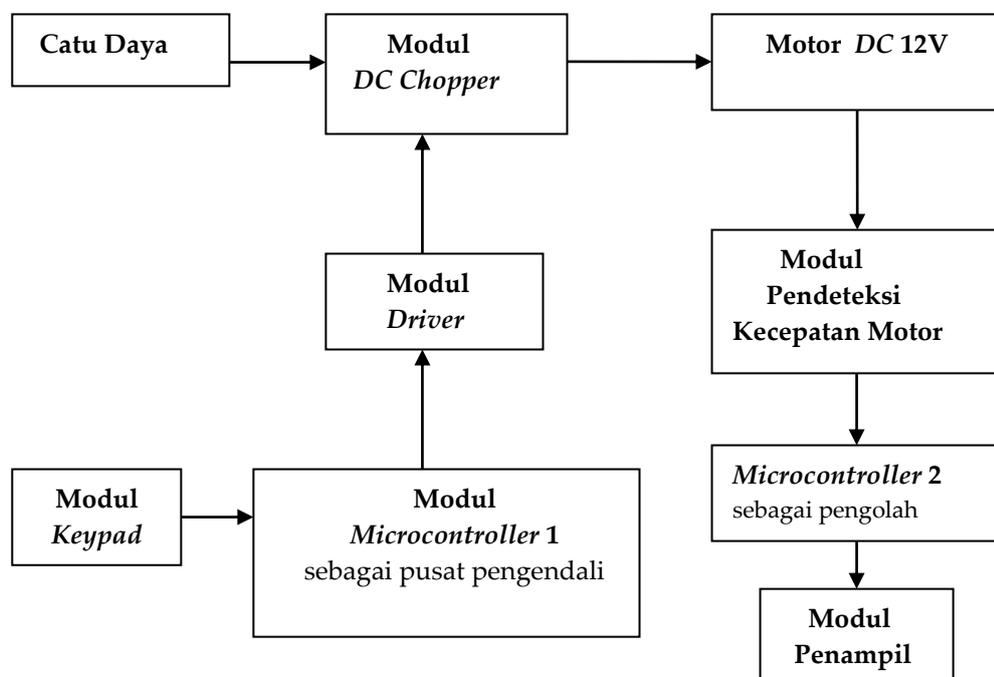
$$= \text{presentaseDutyCycle} \times V_s$$

Untuk hubungan arus kecepatan dan torsi:  $T = K\Phi I_a$

T = torsi , K = konstanta persamaan ,  $\Phi$  = fluks medan ,  $I_a$  = arus.

3. Perancangan Sistem

Secara lengkap diagram blok alat yang dibuat ditunjukkan oleh Gambar 4.



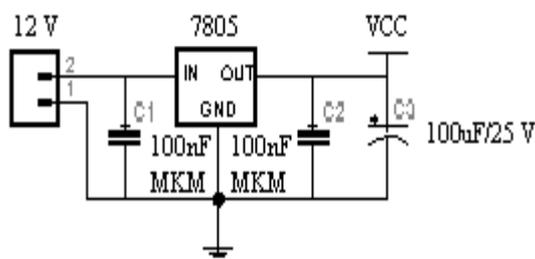
Gambar 4. Diagram Blok Alat

Perancangan dan realisasi perangkat keras dibagi dalam beberapa bagian yaitu :

1. Modul *Regulator* 5V.
2. Modul *Microcontroller*.
3. Modul *Keypad*.
4. Modul *LCD ( Liquid Crystal Display )*.
5. Modul *DC Choper + driver*.
6. Modul *Transmitter IR ( Infra Red )*.
7. Modul *Receiver IR* sebagai pendeteksi kecepatan.

### 3.1. Modul *Regulator*

Modul *regulator* 5 V berfungsi sebagai penstabil tegangan 5 V, tegangan inputnya sebesar 12 V yang berasal dari *power supply*. Tiap modul menggunakan satu buah *regulator* 5 V. Gambar *regulator* 5 Volt ditunjukkan oleh Gambar 5..



Gambar 5. Modul *Regulator* 5V

### 3.2. Modul *Microcontroller* [4]

Pada tulisan ini digunakan dua modul *microcontroller*. Kedua *microcontroller* yang digunakan di sini adalah *microcontroller* tipe AT89S52. AT89S52 memiliki memori *internal (Programmable and Erasable Read Only Memory* disingkat PEROM) sebesar 8 kB sehingga perangkat lunak yang dibuat dapat langsung disimpan di dalam ROM *internal* yang ada. *Microcontroller* ini mempunyai kemampuan *In Sysyem Programming (ISP)*. *Microcontroller* berfungsi mengatur dan mengendalikan seluruh proses operasi yaitu mengatur picuan IGBT, mengolah data dari *keypad*, menerima data dari modul pendeteksi kecepatan, menjalankan sistem PWM, dan menampilkan inputan dan hasil deteksi kecepatan pada LCD. *Microcontroller* AT89S52 mempunyai tiga *timer* yaitu *timer* 0, *timer* 1, dan *timer* 2. Modul ini menggunakan osilator kristal 12 MHz sebagai pembangkit *clock* sehingga dalam alat ini siklus kerja dari *microcontroller* adalah  $12X \frac{1}{12MHz} = 1\mu\text{detik}$ . Gambar

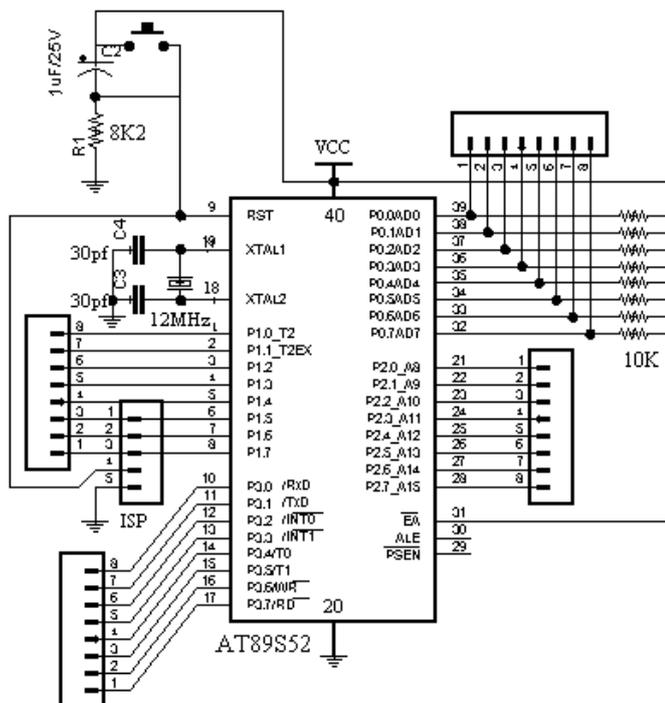
modul *microcontroller* dapat dilihat pada Gambar 6.

### 3.3. Modul *Keypad*

*Keypad* mempunyai banyak fungsi, yaitu untuk memasukkan besarnya frekuensi ( dalam Hz ) dan memasukkan besarnya *duty cycle* yang diinginkan ( dalam % ), juga untuk memulai dan menghentikan motor berputar, dan beberapa fungsi yang lain. *Keypad* yang digunakan di sini menggunakan *keypad* 4 × 4 dengan konfigurasi seperti Tabel 1.

Tombol 0 – 9 adalah tombol angka yang berguna sebagai masukan frekuensi dan *duty cycle* yang dikehendaki. *Enter* adalah tombol yang berguna setelah *user* memasukkan inputan frekuensi atau *duty cycle* yang dikehendaki. *Backspace* adalah tombol yang

berfungsi sebagai mana *backspace* pada *keyboard* berfungsi, yakni untuk menghapus data inputan jika *user* salah memasukkan masukan. *Reset* adalah tombol untuk mereset program dan kembali meminta inputan dari awal, disini nilai masukan yang sudah disimpan diawal akan hilang. *Start* sendiri adalah tombol untuk memulai berputarnya motor. Tombol *Stop* berfungsi untuk menghentikan putaran motor.



Gambar 6. Modul Microcontroller

### 3.4. Modul LCD (Liquid Crystal Display)

Modul penampil digunakan LCD 16 x 2 karakter, di mana LCD ini memiliki 8 jalur data dan 3 jalur kontrol. Data yang dikenali LCD adalah kode ASCII (*American Standard Code for Information Interchange*). Pin *E* (*enable*) berfungsi untuk menginformasikan bahwa ada data baru yang siap ditampilkan. Pin *RS* berfungsi untuk memberitahukan LCD bahwa data yang akan dikirim berupa data atau berupa kode pengaturan. Pin *RS* diset '1' jika mengirim data dan diset '0' jika mengirim kode pengaturan. Pin *R/W* berfungsi untuk memberi perintah baca (*read*) atau tulis (*write*), diset '1' untuk *read* dan diset '0' untuk *write*. Pin *R/W* diset '0' karena LCD hanya digunakan untuk menampilkan. Modul LCD ditunjukkan oleh Gambar 8.

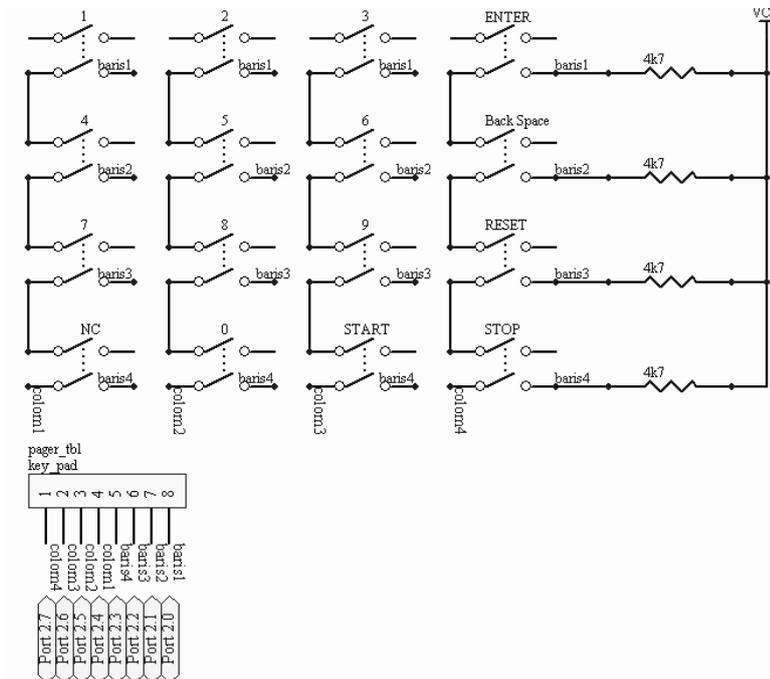
Kemampuan kontras LCD dapat diatur dengan cara mengubah-ubah nilai potensio R1. Berdasarkan lembar data (*data sheet*) LCD ini memiliki 192 karakter yang berbeda, lihat Tabel 3.4.

### 3.5. Modul DC Chopper + driver

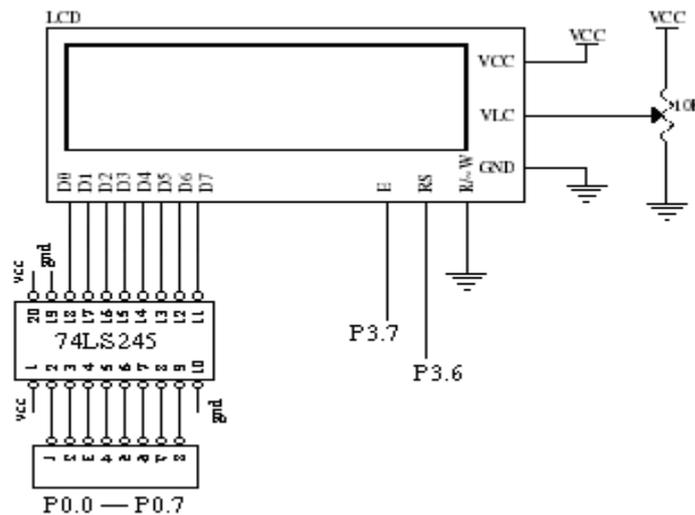
Pada modul ini digunakan sebuah IGBT tipe GT15Q101 yang berfungsi sebagai saklar. Sedangkan sebagai driver untuk memicu agar IGBT aktif digunakan sebuah *optocoupler*. Modul DC Chopper dan drivernya ditunjukkan oleh Gambar 9.

Tabel 1. Konfigurasi Tombol Keypad 4 X 4

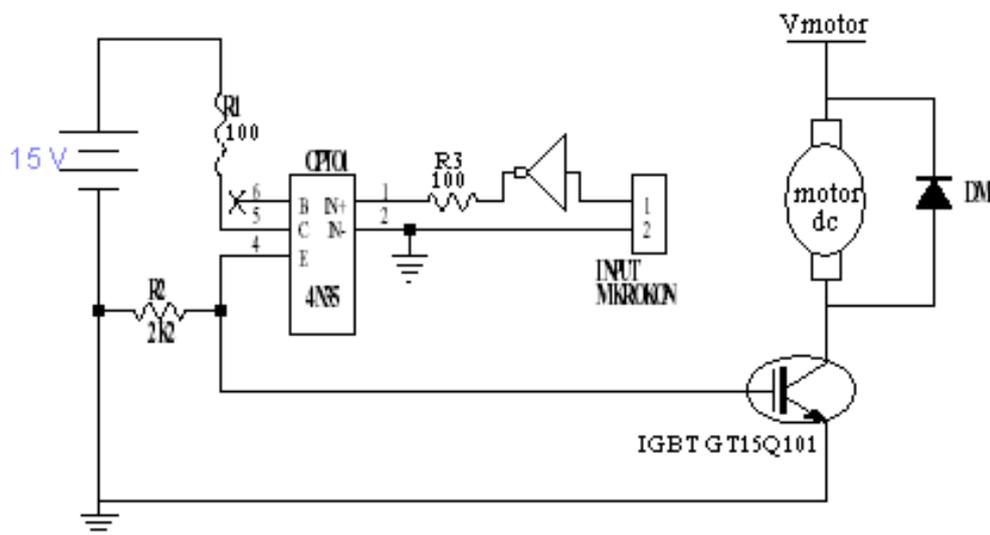
1	2	3	Enter
4	5	6	Backspace
7	8	9	Reset
	0	Start	Stop



Gambar 7. Rangkaian Keypad 4X4



Gambar 8. Modul LCD



Gambar 9. Modul DC Chopper + driver

Prinsip kerja dari rangkaian driver *DC Chopper* dapat dijelaskan sebagai berikut. Pada Gambar 10 terlihat bahwa untuk mengaktifkan transistor yang ada di dalam *optocoupler* 4N35 maka basis dari transistor tersebut harus dipicu dengan cara menyalakan *led* yang ada di dalam *optocoupler* 4N35. Berdasarkan *data sheet* dari *optocoupler* 4N35, untuk menyalakan *led* di dalam 4N35 dibutuhkan tegangan minimal 1,18V dan tegangan maksimalnya 1,5V dengan arus maksimum 100mA. Tegangan yang akan diberikan ke *led* adalah 1,2V dan arus yang melewati *led* adalah

$$I_{led} = \frac{(5 - 1,2)\text{Volt}}{100\Omega} = \frac{3,8\text{Volt}}{100\Omega} = 38 \text{ mA}$$

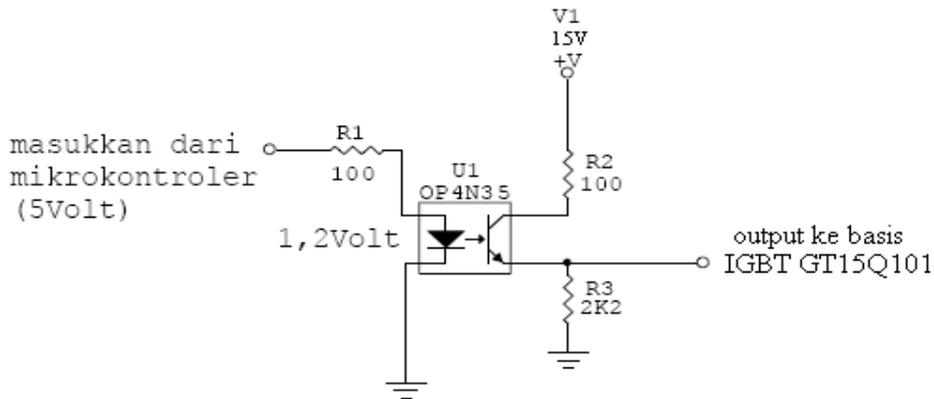
Sehingga saat IGBT GT15Q101 saturasi maka tegangan *output* transistornya adalah

$$V_{output} = \frac{R3}{R2 + R3} \times 15\text{Volt} = \frac{2K2}{100 + 2K2} \times 15\text{Volt} = 14,35 \text{ Volt}$$

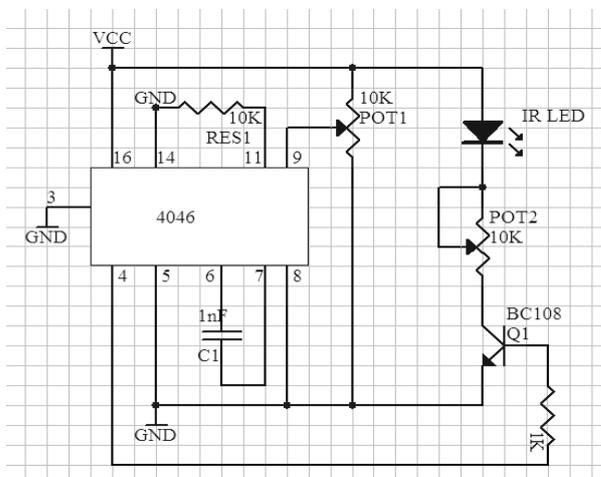
Tegangan picuan dari *microcontroller* akan membentuk pulsa yang lebar pulsa dan *duty cycle* yang dihasilkan akan sama dengan lebar pulsa serta *duty cycle* dari *output optocoupler*, hanya perbedaannya adalah amplitudo pulsa picuan *microcontroller* adalah 5 Volt sedangkan amplitudo pulsa *output optocoupler* adalah 14,35 Volt. Tegangan sebesar 14,35 Volt ini sudah cukup untuk memicu *basis* dari GT15Q101 agar IGBT tersebut aktif. Pada *output* dari *DC Chopper* terdapat dioda DM, dioda ini berfungsi untuk membuang arus dari motor DC yang bersifat *induktif* saat suplai tegangan dimatikan.

### 3.6. Modul Transmitter IR ( Infra Red )

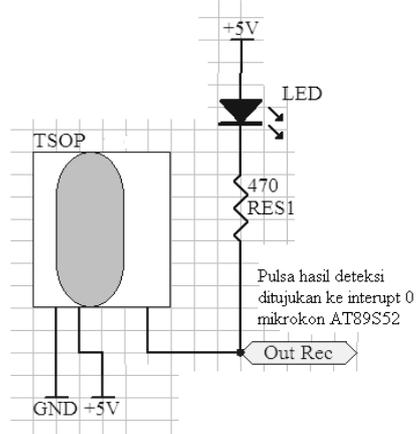
Modul ini berfungsi sebagai transmitter IR. VCO 4046 berfungsi sebagai osilator yang membangkitkan pulsa. Pulsa ini akan dikeluarkan lewat pin 4 untuk kemudian diumpankan ke basis dari transistor BC108 yang akhirnya menyebabkan IR LED mati hidup sesuai frekuensi yang dikeluarkan osilator. Modul transmitter IR ditunjukkan pada Gambar 11.



Gambar 10. Rangkaian Driver DC Chopper



Gambar 11. Modul Transmitter IR



Gambar 12. Modul Receiver IR

### 3.7. Modul Receiver IR sebagai pendeteksi kecepatan motor

Dengan konfigurasi sederhana ini, sinar IR dari *transmitter* dapat dideteksi. *Pin Out Rec* kemudian dihubungkan ke modul pendeteksi kecepatan motor. Gambar modul *receiver IR* dapat dilihat pada Gambar 12. Pada modul ini komponen *TSOP* akan selalu berhadapan dengan *led IR* pada modul *transmitter IR*. Jika antara *led IR* dan *TSOP* tidak terhalang maka *led* yang terdapat pada modul *receiver IR* akan menyala, itu berarti nilai data pada kaki *out rec* adalah '0'. Jika *led IR* dan *TSOP* terhalang maka *led* yang terdapat pada modul *receiver IR* akan padam, dan nilai kaki *out rec* adalah '1'. Pulsa tersebut kemudian diberikan ke kaki *interrupt eksternal 0* dari *microcontroller*. Kaki *interrupt eksternal 0* mendeteksi adanya peralihan logika ( transisi ) dari logika 1 ke logika 0. Jika terjadi transisi, maka perhitungan jumlah lubang akan ditambah 1. Sedangkan *Timer 1* pada *microcontroller* difungsikan sebagai *timer/pewaktu* selama 1 detik. Setelah timer 1 detik berakhir, maka nilai perhitungan jumlah lubang akan diambil. Kemudian berdasarkan pada nilai yang terhitung tersebut, *microcontroller* akan menghitung kecepatan berputar dari motor yang dikendalikan dan menampilkannya ke modul penampil. Kecepatan putar motor dapat dihitung dari banyaknya pulsa yang terdeteksi oleh pengindera optik selama satu detik dengan menggunakan cara perhitungan sebagai berikut :

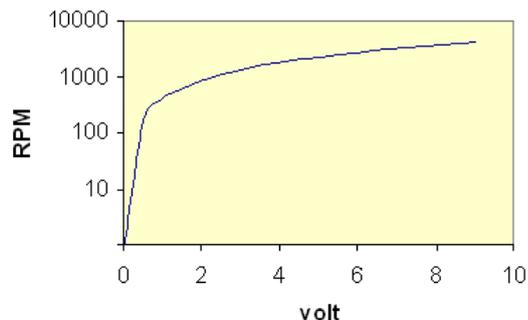
$$\text{Kecepatan motor (rpm)} = \text{Jumlah pulsa yang terhitung} \times \frac{60}{\text{JumlahLubang}}$$

Sebagai contoh, jumlah pulsa yang terhitung oleh pengindera optik selamasatu detik adalah sebanyak 100 pulsa. Maka kecepatan putar dari motor tersebut adalah :

$$\text{Kecepatan putar motor (rpm)} = 100 \times \frac{60}{10} = 600 \text{ rpm}$$

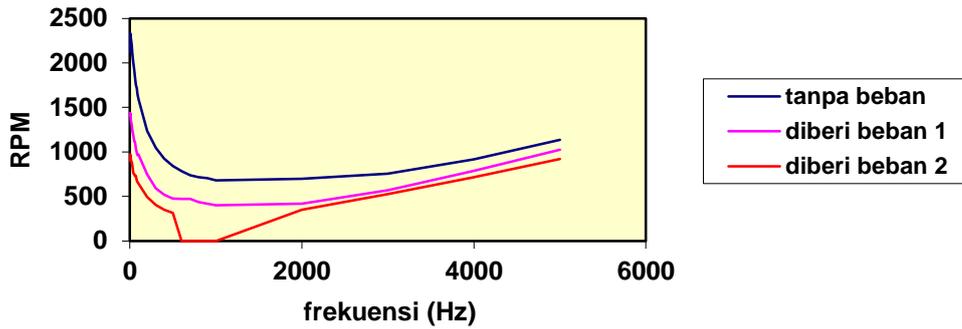
#### 4. Pengujian Sistem

Pada bahasan ini dilakukan pengujian terhadap sistem secara keseluruhan dengan tujuan untuk mengetahui apakah sistem pengatur kecepatan putar motor yang telah direalisasikan dapat bekerja sesuai dengan yang diinginkan. Pengujian pertama dilakukan pada sistem pengaturan kecepatan motor DC yaitu dengan merangkai seluruh modul yang telah diuji sebelumnya menjadi satu sistem yang terpadu. Pada saat pertama kali dihidupkan, alat meminta *user* memasukkan berapa besar frekuensi yang diinginkan. Jumlah frekuensi yang bisa diinputkan hanya sebatas 0 – 10000Hz saja. Jika *user* saat memasukkan frekuensi yang diinginkan salah, *user* bisa menghapus data tersebut dengan tombol *backspace*. Setelah *user* merasa sudah mengisi besar frekuensi yang diinginkan, program menunggu tombol *start* ditekan. Setelah ditekan maka program akan meminta *user* untuk memasukkan besar *duty cycle* yang diinginkan. Besaran *duty cycle* yang bisa diinputkan sudah disebutkan pada subbab batasan masalah. Jika *user* memasukkan data frekuensi dan *duty cycle* melebihi batas yang ditentukan maka program akan menghapus data tersebut dan meminta *user* untuk mengisi kembali. Motor akan berputar jika ada penekanan tombol *start*, dan motor akan berhenti berputar jika tombol *stop* ditekan ( bisa di *start* lagi ), atau tombol reset tidak bisa di *start* lagi). Kecepatan putar motor akan dideteksi oleh bagian pendeteksi kecepatan dan akan ditampilkan pada bagian penampil. Grafik pengujian motor DC yang langsung dihubungkan dengan sumber tegangan dapat dilihat pada Gambar 13.



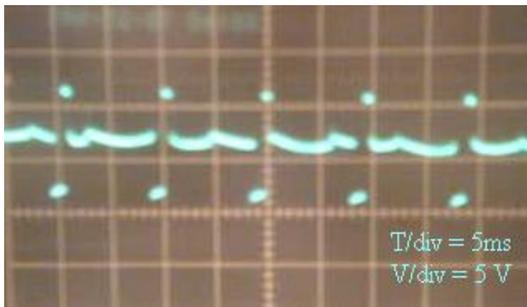
Gambar 13. Grafik kecepatan tanpa pengendali PWM

Dapat dilihat pada grafik diatas bahwa dengan adanya kenaikan tegangan maka kecepatan motor juga akan bertambah. Grafik hasil percobaan pengujian perubahan frekuensi dengan *duty cycle* dapat dilihat pada Gambar 13 (*duty* 10%), Gambar 17 (*duty* 25%), Gambar 18 (*duty* 50%), Gambar 19 (*duty* 75% dan 90%).

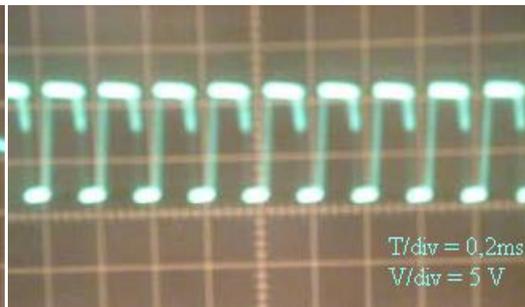


Gambar 14. Grafik pengujian *duty cycle* 10%

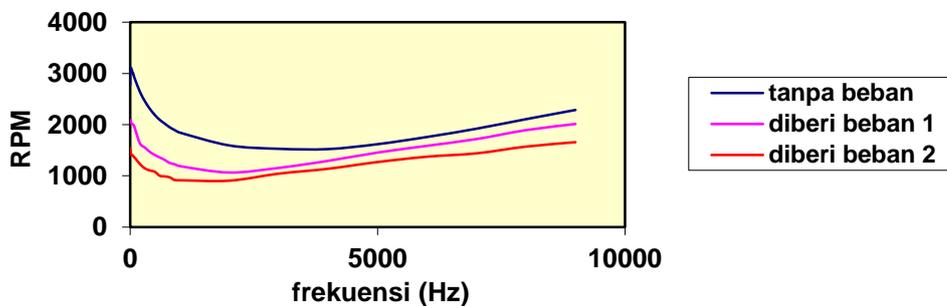
Pada percobaan dengan *duty cycle* 10% saya memberikan dua tipe beban, yakni beban pertama berupa balok segiempat tipis seragam beraturan dan beban kedua berupa balok segiempat seragama beraturan yang di kedua sisi panjangnya nya diberi beban tambahan. Titik kerja minimum dari driver motor yang menggunakan *DC Chopper* pada *duty cycle* 10 % adalah diatas atau lebih besar dari 2000Hz tanpa beban dan >1000Hz dengan beban 1 dan beban 2. Sedangkan nilai kecepatan yang didapat dari frekuensi dibawah 2000Hz merupakan nilai yang tidak valid. Gambar grafik sinyal dibawah 2000Hz dapat dilihat pada Gambar 15 (100Hz), dan Gambar sinyal frekuensi lebih dari 2000Hz dapat dilihat pada Gambar 16 (5000Hz). Bentuk gelombang pada Gambar 15 dan Gambar 16 merupakan bentuk gelombang sebelum motor dibebani.



Gambar 15. Gelombang VCE pada IGBT saat *duty cycle* 10% dan frekuensi 100Hz

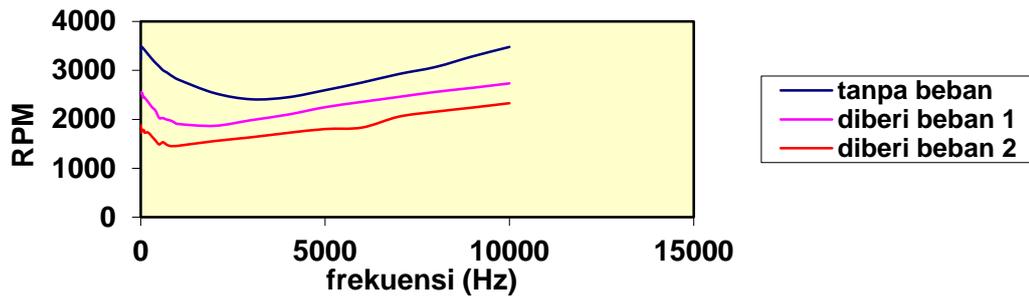


Gambar 16. Gelombang VCE pada IGBT saat *duty cycle* 10% dan frekuensi 5KHz

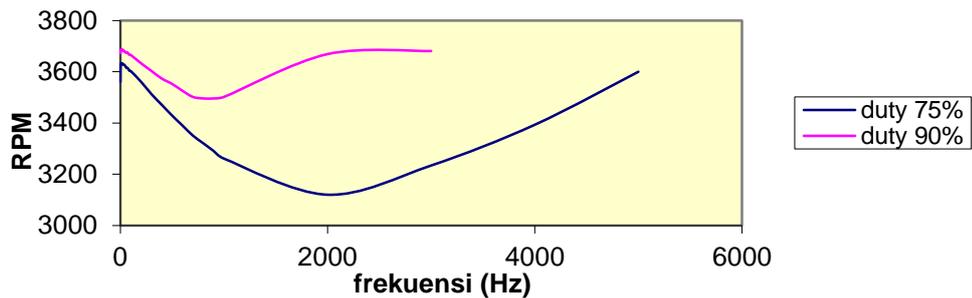


Gambar 17. Grafik pengujian *duty cycle* 25%

Dari hasil percobaan titik kerja minimum *DC Chopper* pada *duty cycle* 25% dan 50% adalah >3Khz tanpa beban, >2Khz pada beban 1 dan >1Khz pada beban 2.



Gambar 18. Grafik pengujian *duty cycle* 50%



Gambar 19. Grafik pengujian *duty cycle* 75% dan 90%

Untuk *duty cycle* 75% nilai frekuensi yang tepat untuk *DC Chopper* adalah lebih besar dari 2000Hz tanpa beban, sedangkan untuk *duty cycle* 90% adalah lebih besar dari 800Hz tanpa beban. Semakin tinggi nilai frekuensi maka semakin baik titik kerja dari *DC Chopper*. Nilai frekuensi seharusnya setinggi mungkin untuk mengurangi arus *ripple* beban dan meminimalkan jumlah tambahan deret induktor pada rangkaian beban.

## 5. Kesimpulan

Berdasarkan pengujian terhadap perangkat yang direalisasikan, maka kesimpulan yang diambil oleh penulis adalah sebagai berikut:

1. Nilai frekuensi seharusnya setinggi mungkin untuk mengurangi arus ripple beban dan meminimalkan jumlah tambahan deret induktor pada rangkaian beban.
2. Frekuensi pensakelaran *dc chopper* pada *duty cycle* 10% adalah >2Khz tanpa beban dan >1Khz dengan beban1 dan beban 2.
3. Frekuensi pensakelaran *dc chopper* pada *duty cycle* 25% dan 50% adalah >3Khz tanpa beban, >2Khz pada beban 1 dan >1Khz pada beban 2.
4. Frekuensi pensakelaran *dc chopper* pada *duty cycle* 75% adalah >2Khz tanpa beban, dan *duty cycle* 90% adalah >800Hz tanpa beban.
5. Semakin besar sinyal frekuensi sinyal *PWM* yang diberikan semakin baik pula kerja dari *DC Chopper*.
6. Keuntungan yang didapat dengan menggunakan *DC Chopper* yakni kecilnya disipasi daya.

## Daftar Pustaka

- [1] M. Rashid, *Power Electronics: Circuits, Devices, and Applications*, Prentice Hall International Editions, Second Edition.

- [2] S.J. Chapman, *Electric Machinery Fundamentals*, International Student Edition, McGraw-Hill, 1985.
- [3] A. Nalwan, Paulus, *Panduan Praktis Teknik Antarmuka dan Pemrograman Mikrokontroler AT89C51*, PT Elek Media Komputindo, Jakarta, 2003.
- [4] D.K. Sutantyo, *Arsitektur dan Pemrograman Mikrokontroler MCS-51*, Fakultas Teknik Elektronika dan Komputer, Universitas Kristen Satya Wacana, Salatiga, 2001.