

BAB 2 TINJAUAN PUSTAKA

2.1 *Tachometer*

Tachometer adalah sebuah instrument ukur atau alat ukur yang mampu untuk mengukur kecepatan putaran poros engkol atau piringan. Alat ini biasanya menampilkan revolutions per minute (RPM) pada sebuah pengukur skala analog, namun ada juga versi tampilan digital. Berdasarkan prinsip kerjanya, *tachometer* dibagi menjadi dua yaitu:

2.1.1 *Tachometer* kontak

Tachometer kontak adalah *tachometer* yang melakukan kontak fisik dengan poros atau kecepatan permukaan yang diukur. *Tachometer* ini terdiri dari poros logam dengan ujung kontak dari bahan karet, roda kontak permukaan karet, bagian konversi optik atau mekanis, dan tampilan pembacaan. Alat ini bekerja dengan cara membawa roda yang berputar bebas yang bersentuhan dengan sebuah cakram. Cakram tersebut akan mendorong roda untuk menciptakan pulsa. Pulsa-pulsa ini akan dihitung oleh *tachometer* dalam RPM.

2.1.2 *Tachometer* non-kontak

Tachometer jenis ini tidak melakukan kontak fisik dengan objek yang diuji. *Tachometer* non-kontak sudah termasuk *tachometer* optik, induktif, dan akustik. *Tachometer* optik terdiri dari pemancar, penerima dan tampilan pembacaan. *Tachometer* ini biasanya digunakan untuk pengukuran pada motor atau mesin listrik yang berputar. *Tachometer* induktif digunakan untuk mengukur kecepatan putaran mesin pembakaran dalam. *Tachometer* ini terdiri dari *pick-up* induktif (*coil*), pemilihan silinder dan tampilan pembacaan. Medan magnet yang dipancarkan koil mesin dideteksi oleh *pick-up* induktif *tachometer* dan diubah menjadi pembacaan rpm. *Tachometer* akustik digunakan pada mesin pembakaran internal. Alat ini terdiri dari transduser akustik, pemilihan silinder dan tampilan pembacaan. Suara yang dikeluarkan oleh busi mesin saat dinyalakan dideteksi oleh *tachometer*, dan ditampilkan pada tampilan pembacaan RPM.

Penelitian mengenai *tachometer* telah banyak dilakukan sebelumnya seperti pada penelitian yang membuat rancangan *tachometer* yang akan mengkalibrasi alat yang menghasilkan kecepatan putar agar sesuai dengan standar yang dipersyaratkan. Rancang bangun *tachometer* yang dibuat menggunakan sensor E18-D80NK, mikrokontroler, dan LCD tanpa menyentuh objek yang diukur [2]. Penelitian lain juga menggunakan *tachometer* untuk menganalisa pengaruh kemiringan sudut sudu pengarah angin terhadap tingkat efisiensi turbin angin sumbu vertikal dalam menghasilkan daya listrik [3]. Selain itu, terdapat penelitian menggunakan *tachometer* untuk menganalisa pengaruh sudu sudu *impeller* pada kinerja blower sentrifugal [4].

2.2 Kalibrasi *Tachometer*

Kalibrasi *tachometer* dapat dilakukan dengan banyak cara namun harus tetap mengacu pada dokumen standar yang telah diakui secara nasional maupun internasional. Salah satu dokumen acuan tersebut yaitu SANAS (*South African National Accreditation System*) TR 45-02. Dokumen ini berisi kriteria yang harus dipenuhi dalam melakukan kalibrasi *tachometer* [5].

2.2.1 Persyaratan lingkungan

- a) Suhu lingkungan dan gradien suhu di laboratorium tempat *tachometer* di kalibrasi harus memenuhi persyaratan yang relevan untuk bidang kalibrasi lingkup waktu dan frekuensi.
- b) Pencahayaan di dalam laboratorium harus memadai untuk memfasilitasi pengerjaan kalibrasi yang benar.
- c) Tingkat getaran di laboratorium harus diminimalkan sedemikian rupa sehingga tidak mempengaruhi hasil pengukuran.
- d) Jika kalibrasi dilakukan di tempat yang tidak melakukan pengendalian lingkungan, maka laboratorium harus memantau, mencatat, dan melaporkan suhu saat kalibrasi dilakukan.

2.2.2 Persyaratan teknis

- a) Kalibrasi *tachometer* optik non-kontak harus mencakup verifikasi bahwa pemancar dan penerima berfungsi pada rentang jarak yang ditentukan.

- b) Setiap alat uji harus memiliki desain yang memadai dan setiap pembebanan mekanis yang disebabkan oleh *unit under test* (UUT) tidak boleh mempengaruhi pengukuran.
- c) Kalibrasi *tachometer* kontak harus mencakup kopling langsung atau pengukuran menggunakan ujung kerucut karet yang disediakan dan/atau roda pengukur permukaan. Prosedur juga harus mengatasi sudut kontak ujung probe dengan benda yang berputar, tekanan yang diberikan, kondisi ujung karet atau roda, dan selip dalam kondisi kerja.
- d) Prosedur kalibrasi stroboskop harus membahas metode deteksi, dan jarak deteksi.
- e) Prosedur untuk kalibrasi *tachometer* induktif dan akustik harus membahas pemilihan langkah dan silinder.
- f) *Tachometer* harus dikalibrasi sesuai dengan prosedur kalibrasi pabrikan; jika prosedur tersebut tidak ada, laboratorium harus melakukan setidaknya 5 pengukuran melalui rentang *tachometer*. *Tachometer* ganda atau multirange harus dikalibrasi setidaknya 3 titik per jangkauan.
- g) *Tachometer* multi fungsi harus dikalibrasi untuk fungsi kontak dan non-kontak. Dalam kasus di mana hanya satu dari fungsi yang telah dikalibrasi, label dan sertifikat kalibrasi harus dengan jelas menunjukkan 'kalibrasi parsial'.
- h) Perhitungan ketidakpastian pengukuran untuk *tachometer* yang dikalibrasi oleh laboratorium harus mempertimbangkan setidaknya sumber berikut berikut:
- 1) Ketidakpastian pengukuran standar
 - 2) Resolusi standar
 - 3) Resolusi dari UUT
 - 4) Keselarasan probe *tachometer*
 - 5) Kesalahan akibat selip
 - 6) Stabilitas pengukuran
 - 7) Pengulangan pengukuran

8) Kondisi lingkungan

2.3 Motor DC

Motor DC adalah motor listrik yang membutuhkan input tegangan arus searah ke kumparan medan untuk diubah menjadi energi mekanik. Energi mekanik ini dapat digunakan untuk memutar pompa, kipas, *blower*, kompresor penggerak, alat pengangkat, dan sebagainya. Motor DC dikendalikan oleh angker dan medan. Stator dan rotor adalah bagian penting dari motor DC. Bagian motor DC yang tidak berputar adalah stator. Rotor adalah bagian yang berputar. Motor DC dengan kontrol jangkar menggunakan arus jangkar sebagai variabel kontrol. Medan stator dihasilkan oleh magnet permanen atau kumparan arus.

2.3.1 Komponen Utama Motor DC

a) Kutub Medan

Pada motor DC terdapat magnet. Magnet yang digunakan pada motor DC disebut sebagai magnet permanen yang berarti medan magnetnya selalu aktif. Ujung magnet yang berlawanan menarik sementara ujung yang serupa menolak. Medan magnet magnet membentang dari kutub selatan ke kutub utara. Bagian paling kuat dari medan magnet magnet ada di ujungnya. Hubungan dua kutub magnet akan menyebabkan perputaran pada motor DC. Dua magnet menciptakan medan yang sangat kuat; inilah mengapa dua magnet disertakan dalam motor DC di sekitar rotor sedemikian rupa sehingga medan magnet yang kuat melewati rotor.

b) Rotor

Rotor atau angker terbuat dari beberapa *disk* yang diisolasi satu sama lain dengan lembaran berlapis. Banyak *disk* mencegah terciptanya arus *eddy* yang besar. (Arus *Eddy* adalah alasan pelat diisolasi.) Arus *Eddy* masih ada tetapi jauh lebih kecil dan tidak mengganggu pengoperasian motor. Untuk efisiensi motor yang lebih besar, piringan rotor dibuat sekecil mungkin. Rotor adalah bagian dinamis dari motor yang digunakan untuk membuat putaran mekanis.

c) Komutator

Komutator terbuat dari pelat tembaga kecil yang dipasang pada poros dan berputar saat poros berputar. Rotasi rotor menyebabkan kutub catu daya ke kumparan berubah. Setiap kumparan terhubung ke dua pelat komutator yang diisolasi secara elektrik satu sama lain tetapi dihubungkan oleh kumparan. Dengan terminal positif dan negatif yang terhubung ke dua pelat komutator, arus mudah mengalir dan medan elektromagnetik dihasilkan.

2.3.2 Prinsip Kerja Motor DC

Medan magnet tercipta di dekat konduktor bila terdapat arus yang melewatinya. Arah medan magnet ditetapkan oleh arah aliran arus listrik pada konduktor. Bila konduktor berupa U (angker dinamo) diletakkan di antara kutub utara serta selatan medan magnet yang kuat konduktor akan berhubungan dengan medan magnet kutub sehingga menyebabkan konduktor berotasi.



Gambar 2. 1 Prinsip Kerja Motor DC [6]

Medan magnet muncul di celah udara ketika kumparan medan motor DC diberi energi. Medan magnet yang dibuat searah dengan jari-jari angker. Medan magnet memasuki angker dari sisi kutub Utara kumparan medan dan "keluar" angker dari sisi kutub Selatan kumparan medan. Konduktor yang terletak di kutub lain dikenai gaya dengan intensitas yang sama tetapi berlawanan arah. Kedua gaya berlawanan ini menciptakan torsi yang menyebabkan dinamo motor berputar. Besarnya torsi yang terjadi dapat dihitung menggunakan:

$$T = F \cdot r \quad (2.1)$$

Dimana:

T = torsi (Nm)

F = gaya tolak (N)

r = jarak sisi kumparan pada sumbu putar (meter)

Selama terdapat medan magnet di motor DC gaya ini akan berlangsung terus menerus. Oleh karena itu, untuk mendapatkan putaran yang terus menerus maka digunakan banyak konduktor [6]. Prinsip pengoperasian utama motor DC ini memungkinkannya untuk mengubah energi listrik dari arus searah menjadi energi mekanik melalui gerakan memutar, yang kemudian dapat digunakan untuk mendorong objek.

Hubungan antara kecepatan, flux medan dan tegangan ditunjukkan dalam persamaan berikut:

Gaya Elektromagnetik (E)

$$E = K\Phi N \quad (2.2)$$

Torque (T)

$$T = K\Phi I_a \quad (2.3)$$

Kecepatan Motor Tak Berbeban (N tidak berbeban (rpm)):

$$N_{tidak\ berbeban} (rpm) = T \times \left(\frac{N_{Max}}{km} \right) \quad (2.4)$$

Kecepatan Motor Berbeban (N berbeban (rpm))

$$N_{Max} - N_{tidak\ berbeban} = N_{berbeban} (rpm) \quad (2.5)$$

Arus:

$$I(A) = T \times Km/A \quad (2.6)$$

Daya masukan (Win):

$$T \times N_{berbeban} \times \text{Konfersi faktor antara } T \text{ dan } N = Win \quad (2.7)$$

Daya keluaran (Wout):

$$V \times A = Wout \quad (2.8)$$

Efisiensi:

$$Efisiensi (\%) = Wout/Win \quad (2.9)$$

Dimana:

E = gaya elektromagnetik (volt)

Φ = fluks medan

N = kecepatan putar RPM (putaran per menit)

T = torsi elektromagnetik (g/cm)

I_a = arus dinamo (A)

K = konstanta

V = tegangan (Volt)

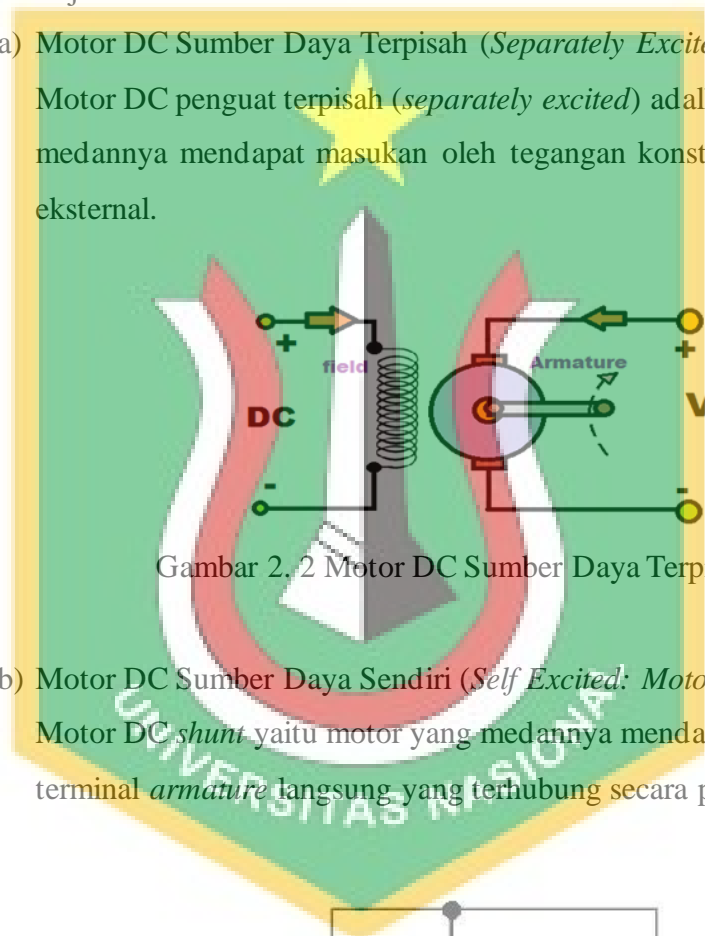
W_{out} = Daya keluaran (Watt)

W_{in} = Daya masukan (Watt)

2.3.3 Jenis-jenis Motor DC

a) Motor DC Sumber Daya Terpisah (*Separately Excite*)

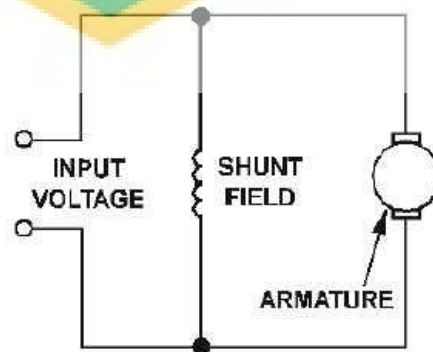
Motor DC penguat terpisah (*separately excited*) adalah motor DC yang medannya mendapat masukan oleh tegangan konstan *power supply* eksternal.



Gambar 2.2 Motor DC Sumber Daya Terpisah [7]

b) Motor DC Sumber Daya Sendiri (*Self Excited: Motor Shunt*)

Motor DC *shunt* yaitu motor yang medannya mendapat masukan dari terminal *armature* langsung yang terhubung secara paralel.



Gambar 2.3 Rangkaian Motor *Shunt* [8]

Karakteristik kedua jenis motor ini yaitu ketika beban pada ujung *shaft* motor naik, hingga kemudian torsi beban melebihi torsi induksi, maka akan terjadi perlambatan pada motor. Karena motor melambat tegangan induksi dalam mesin akan turun yang mengakibatkan arus *armature* meningkat. Dengan meningkatnya arus *armature*, maka torsi induksi motor pun akan meningkat dan akhirnya torsi motor akan sama dengan torsi beban pada kecepatan sudut yang lebih rendah [9].

Ada dua cara umum untuk mengontrol kecepatan motor DC, yaitu memvariasikan nilai *field resistance* dan memvariasikan tegangan terminal yang diinputkan ke *armature*.

c) Motor DC Seri

Disebut Motor DC seri karena gulungan medan (medan *shunt*) terhubung secara seri dengan gulungan kumparan motor DC (A). Sehingga arus medan sama dengan arus kumparan motor DC seperti ditunjukkan dalam Gambar 2.4.



Gambar 2. 4 Rangkaian Motor Seri [10]

Karakteristik terminal motor DC seri berbeda dengan motor DC *shunt*. Motor DC seri memiliki karakteristik penurunan yang signifikan pada kurva torsi-kecepatan. Hal ini diakibatkan bahwa fluks berbanding lurus dengan arus *armature*, saat beban pada motor naik, maka fluks

akan naik juga, kenaikan fluks ini lah yang akan mengurangi kecepatan motor [9]. Hanya ada satu cara efisien yang dapat dilakukan untuk melakukan pengontrolan kecepatan pada motor DC seri, yaitu dengan cara mengatur tegangan terminal motor. Dengan memperbesar tegangan terminal maka menghasilkan peningkatan kecepatan motor pada nilai torsi yang diberikan.

d) Motor DC Penguat Gabungan (*Compound*)

Motor Kompon DC merupakan gabungan motor seri dan *shunt*. Motor DC majemuk menggunakan fitur belitan medan seri dan *shunt*. Belitan untuk jangkar dihubungkan secara seri sedangkan belitan untuk medan adalah sambungan *shunt* atau paralel. Motor DC majemuk dibagi menjadi kumulatif dan diferensial. Dengan motor DC kumulatif, fluks medan *shunt* membantu fluks pada medan seri. Keduanya bergerak ke arah yang sama sedangkan fluks motor DC senyawa diferensial, untuk medan seri dan *shunt*, bergerak ke arah yang berlawanan. Motor DC senyawa kumulatif dan diferensial dapat memiliki *shunt* panjang atau pendek. Hal ini didasarkan pada *shunting* dari belitan medan *shunt*.



Gambar 2. 5 Rangkaian Motor Kompon [10]

2.4 Mikrokontroler

Mikrokontroler merupakan *chip* mikrokomputer yang secara fisik berupa sebuah IC (*Integrated Circuit*). Bagian utama dari sebuah mikrokontroler terdiri dari CPU (*Central Processing Unit*), RAM (*Random-Access Memory*), ROM (*Read-Only Memory*) dan port I/O (Input/Output). Mikrokontroler bekerja sesuai program (perangkat lunak) yang ditanamkan di dalamnya, dan program tersebut dibuat mengikuti aplikasi yang diinginkan. Mikrokontroler yang digunakan dalam penelitian ini yaitu Arduino UNO.

Arduino UNO merupakan salah satu *board* yang menggunakan mikrokontroler ATmega328, sehingga proses unggah data lebih cepat, tidak memerlukan *driver* USB dan dapat diprogram untuk keperluan akuisisi data. Adapun bagian-bagian dasar Arduino Uno, yaitu:

- a) Colokan Catu Daya Eksternal
Befungsi untuk memasok sumber daya listrik ke arduino uno saat tidak terhubung ke computer.
- b) Port USB
Befungsi untuk menghubungkan Arduino Uno ke komputer.
- c) Pin Analog
Pin ini mempunyai nilai yang bersifat analog. Nilai pin analog berkisar antara 0 – 1023.
- d) Pin Digital
Befungsi untuk menghubungkan perintah input dan output.
- e) Pin Suplai
Terdapat dua pin yang dapat digunakan yaitu 3.3 V dan 5 V

Spesifikasi Arduino Uno R3 dapat dilihat pada Tabel 2.1.

Tabel 2. 1 Spesifikasi Arduino R3

Spesifikasi	Keterangan
Mikrokontroler	ATMega328
Tegangan Pengoperasian	5 V
Tegangan masuk	7 – 12 V
Batas tegangan masuk	6 – 20 V
Arus DC tiap pin I/O	40 mA

Arus DC untuk pin 3.3 V	50 mA
Memori	32 kb
Bootloader	SRAM 2 kb
EEPROM	1 kb
Kecepatan <i>Clock</i>	16 MHz

Pada penelitian ini menggunakan Arduino Uno R3 karena berdasarkan spesifikasi yang dimiliki dinilai sudah memenuhi untuk menjalankan program yang akan dibuat.



Gambar 2. 6 Arduino Uno [11]

Arduino Uno terdiri dari perangkat lunak (Arduino Uno) dan perangkat keras (ATMega328).

a) Arduino IDE

Arduino Uno menggunakan bahasa pemrograman C yang dituliskan menggunakan Arduino IDE. Melalui Arduino IDE ini pembuat program bisa menanamkan programnya ke dalam mikrokontroler ATMega328 yang tertanam di dalam modul Arduino Uno, hal ini dinamakan dengan *sketch*. Kemampuan yang dimiliki IDE ini yaitu kemampuan melakukan penyusunan program yang dibuat tanpa harus menggunakan alat tambahan. Setiap program Arduino (biasa disebut *sketch*) mempunyai dua buah fungsi yaitu:

1) `void setup () { }`

Sebagai tempat untuk mendefinisikan lingkungan dan variabel yang akan digunakan secara global seperti input, output, dan lain-lain.

2) `void loop () { }`

Setelah *setup* (fungsi *void setup*) selesai, fungsi ini akan dijalankan. Semua program yang dibuat pada fungsi ini akan terus berjalan secara terus-menerus hingga *Arduino board* dalam keadaan mati.

b) ATmega328

ATmega328 adalah mikrokontroler *Advanced Virtual RISC* (AVR) yang mendukung pemrosesan data 8-bit. ATmega-328 memiliki memori *flash* internal 32KB. ATmega328 memiliki 1KB *Electrically Erasable Programmable Read-Only Memory* (EEPROM). Properti ini menunjukkan jika suplai listrik yang disuplai ke mikrokontroler dilepas, itupun dapat menyimpan data dan dapat memberikan hasil setelah menyediakannya dengan suplai listrik. Selain itu, ATmega-328 memiliki 2KB *Static Random Access Memory* (SRAM). ATmega 328 memiliki beberapa fitur berbeda yang menjadikannya perangkat paling populer di pasar saat ini. Fitur-fitur ini terdiri dari arsitektur *RISC* yang canggih, kinerja yang baik, konsumsi daya yang rendah, penghitung waktu nyata yang memiliki osilator terpisah, 6 pin PWM, Serial *USART* yang dapat diprogram, kunci pemrograman untuk keamanan perangkat lunak, throughput hingga 20 MIPS, dan lainnya.

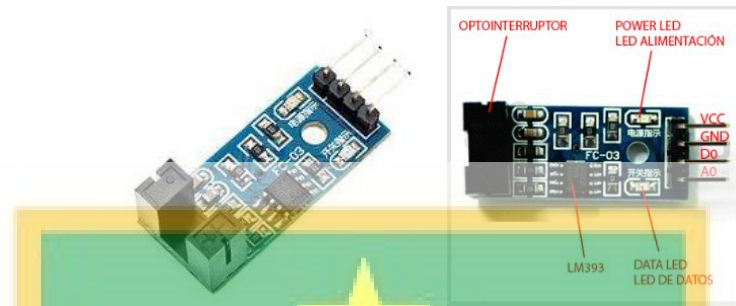
2.5 Sensor IR Kecepatan LM393

Sensor IR kecepatan LM393 adalah suatu komponen yang memanfaatkan cahaya. Modul LM393 adalah *counter* IR yang terdiri dari *transmitter* dan *receiver* IR. Jika ada penghalang yang diletakkan di antara sensor-sensor ini, sinyal dikirim ke mikrokontroler. Sensor ini biasa disebut juga sebagai *optocoupler* yang terbuat dari bahan semi-konduktor dan terdiri dari kombinasi LED (*Light Emitting Diode*) infra merah sebagai *transmitter* dan *phototransistor* sebagai *receiver*.

Berikut fungsi tiap pin modul sensor LM393:

a) Pin VCC sebagai penghubung dengan power supply sebesar 3.3 V atau 5 V.

- b) Pin GND sebagai penghubung ke ground.
- c) D0 sebagai output sinyal TTL.
- d) A0 tidak berfungsi karena sensor tidak mengeluarkan sinyal output analog.



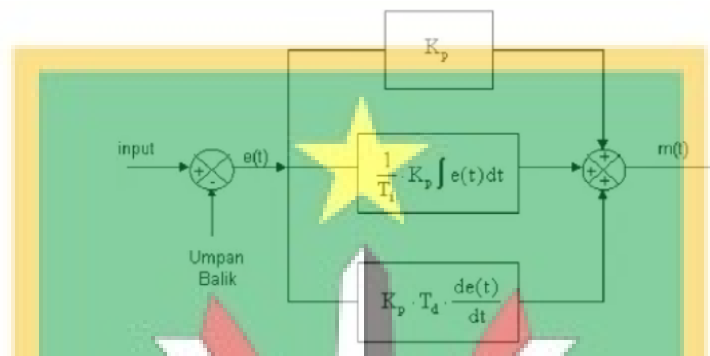
Gambar 2. 7 Sensor Kecepatan LM393 [12]

Sebelumnya telah banyak penelitian yang memanfaatkan sensor kecepatan LM393. Penelitian pertama menggunakan sensor kecepatan LM393 untuk mengukur kecepatan rotasi menggunakan mikrokontroler [13]. Hasil penelitian tersebut yaitu membuktikan bahwa sensor kecepatan LM393 dapat digunakan untuk mengukur kecepatan angular atau sudut dan hasil pengukuran yang akurat. Selain itu, ada yang menggunakan sensor kecepatan LM393 sebagai perekam kecepatan. Mereka merancang sistem perekam kecepatan sepeda motor saat terjadi kecelakaan lalu lintas. Hasil pengujian kemampuan sensor kecepatan LM393 dalam membaca RPM yaitu memiliki tingkat akurasi 98,05% [14]. Kemudian Firdaus, Khuria, Kusuma, Salvaningtyas, dan Azmita juga menggunakan sensor kecepatan LM393 pada penelitian mereka. Mereka merancang sistem pemantauan kecepatan turbin angin sumbu vertikal agar dapat memperoleh daya yang maksimal. Dari hasil penelitian mereka diperoleh persentase *error* sensor kecepatan LM393 terbesar adalah 4% yang artinya sangat akurat [15].

2.6 Kendali PID

Kendali PID merupakan kontroler dengan cara umpan balik. Sistem ini mengevaluasi variabel umpan balik menggunakan titik tetap untuk menghasilkan sinyal kesalahan. Berdasarkan hal tersebut, sistem akan

mengubah output sistem. Prosedur ini akan berlanjut hingga kesalahan mencapai nol. Jika tidak, nilai variabel umpan balik menjadi setara dengan titik tetap. Komponen kendali PID terdiri dari tiga jenis yaitu proporsional, integratif dan derivatif. Elemen-elemen ini masing-masing secara keseluruhan bertujuan untuk mempercepat sistem dalam menghilangkan *offset* dan menghasilkan perubahan awal yang signifikan.



Gambar 2. 8 Blok Diagram Kontrol PID [16]

2.6.1 Kontrol Proporsional

Kontrol proporsional memberikan output yang sebanding dengan kesalahan arus terhadap waktu ($e(t)$). Kontrol ini membandingkan titik yang diinginkan atau *set point* dengan nilai aktual atau nilai proses umpan balik. Kesalahan yang dihasilkan akan dikalikan dengan konstanta proporsional untuk mendapatkan output. Jika nilai *error*nya nol, maka keluaran kontroler ini adalah nol. Pengontrol ini memerlukan pembiasaan atau pengaturan ulang manual saat digunakan sendiri. Hal ini disebabkan karena tidak pernah tercapainya kondisi mapan (*steady state*). Sehingga kontrol ini memberikan operasi yang stabil tetapi selalu mempertahankan kesalahan kondisi *steady state* [17]. Kecepatan respons meningkat ketika konstanta proporsional meningkat.

2.6.2 Kontrol Integral

Karena keterbatasan kontrol proporsional di mana selalu ada *offset* antara variabel proses dan *setpoint*, diperlukan kontrol integral yang memberikan tindakan yang diperlukan untuk menghilangkan kesalahan keadaan tunak. Kontrol ini mengintegrasikan kesalahan selama periode

waktu hingga nilai kesalahan mencapai nol. Kontrol integral memegang nilai ke perangkat kontrol akhir di mana kesalahan menjadi nol. Kontrol integral juga menurunkan outputnya ketika kesalahan negatif terjadi. Kontrol ini membatasi kecepatan respons dan memengaruhi stabilitas sistem. Kecepatan respons ditingkatkan dengan menurunkan penguatan integral, K_i . Ketika penguatan kontrol integral menurun, kesalahan kondisi tunak juga terus menurun [17]. Untuk sebagian besar kasus, pengontrol PI digunakan khususnya di mana respons kecepatan tinggi tidak diperlukan. Saat menggunakan kontroler PI, output kontrol integral terbatas pada kisaran tertentu untuk mengatasi kondisi putaran integral di mana output integral terus meningkat bahkan pada kondisi kesalahan nol, karena nonlinier pada sistem.

2.6.3 Kontrol Derivatif

Kontrol integral tidak memiliki kemampuan untuk memprediksi perilaku kesalahan di masa mendatang. Jadi hanya bereaksi secara normal setelah *setpoint* diubah. Kontrol derivatif mengatasi masalah ini dengan mengantisipasi perilaku kesalahan di masa mendatang. Keluarannya bergantung pada laju perubahan kesalahan terhadap waktu, dikalikan dengan konstanta turunan. Kontrol ini memberikan *kick start* untuk output sehingga meningkatkan respons sistem. Kontrol derivatif meningkatkan stabilitas sistem dengan mengkompensasi kelambatan fase yang disebabkan oleh kontrol integral. Meningkatkan keuntungan derivatif meningkatkan kecepatan respon. Sehingga jika menggabungkan ketiga pengontrol ini, dapat diperoleh respons yang diinginkan untuk sistem.

Desain kontrol PID adalah parameter yang digunakan untuk memenuhi kriteria kinerja yang diinginkan untuk sistem *close loop*. *Tuning controller* adalah teknik yang dapat digunakan untuk meningkatkan nilai *response control* yang optimal. Banyak metode yang dapat dilakukan dalam *tuning controller* PID, seperti metode *Zieger-Nichols* dilakukan berdasarkan eksperimen dengan memberikan input step pada sistem dan mengamati hasilnya untuk memperoleh nilai parameter kontroler [18]. Metode *tuning*

lainnya yaitu metode *Root Locus*, metode *Fuzzy Logic*, metode *Heuristik*, dan metode *Genetic Algorithm (GA) based PID* [19]. Proses *tuning* juga dapat dipermudah dengan menggunakan aplikasi untuk melakukan simulasi terlebih dahulu contohnya aplikasi Matlab [20]. Pada penelitian Somwanshi dan Chao membuktikan bahwa metode *Fuzzy PID controller* memiliki hasil respon yang lebih baik dibanding dengan metode lain [21] [22].

Salah satu faktor terpenting dalam desain Kontroler PID adalah memastikan bahwa proses dikontrol secara akurat [23]. Adapun *response control* yang akan dijadikan sebagai penilaian kinerja kontrol PID baik atau tidak yaitu:

- a) *Peak Time* (T_p) : waktu yang diperlukan untuk mencapai nilai puncak pertama (*peak*).
- b) *Persen Overshoot* (%OS) : jumlah gelombang yang melakukan *overshoot* terhadap *steady-state* atau nilai akhir pada waktu puncak, diekspresikan dalam bentuk presentase terhadap nilai *steady-state*.
- c) *Settling Time* (T_s) : besarnya waktu yang diperlukan oleh osilasi teredam (*damped*) transien untuk bertahan pada $\pm 2\%$ nilai akhir.
- d) *Rise Time* (T_r) : waktu yang diperlukan untuk perubahan dari 10% menjadi 90% nilai akhir.
- e) *Offset* atau *Error* (e) : besar penyimpangan nilai antara *set point* dengan nilai *steady-state*.

Penelitian tentang sistem kendali motor DC menggunakan kontrol PID sudah banyak dilakukan sebelumnya dengan tujuan dan metode yang berbeda-beda. Penelitian pertama membuat suatu rancang bangun sistem kendali kecepatan putar motor DC di bidang pertanian yaitu pada mesin pengaduk bahan-bahan pertanian. Sistem kendali ini dibuat dengan tujuan menghasilkan sistem kendali yang baik pada mesin pengaduk agar dapat meningkatkan kinerja mesin pengaduk. Rancang bangun ini terdiri dari perangkat keras dan perangkat lunak. Komponen perangkat keras terdiri dari mikrokontroler, *Solid State Relay* (driver motor), motor DC (actuator), *rotary encoder* (sensor) dan *power supply* (sumber tegangan). Sedangkan perangkat lunak berisi algoritma bahasa program kendali PID. Dalam memperoleh nilai konstanta PID dilakukan tuning menggunakan *Metoda Internal Control* (MIC). Hasil

penelitian ini yaitu dapat memberikan kestabilan pada kecepatan putar 62, 130, 190, dan 252 RPM. Kemudian saat rancang bangun mesin pengaduk diuji menggunakan beban adonan roti dan kedelai menunjukkan *settling time* yang pendek, *overshoot* bernilai 0 dan nilai *error steady state* yang masih berada pada batas toleransi yaitu 2% dan 5% [24].

Kemudian pada penelitian yang lain dibuat suatu sistem kendali kecepatan motor DC dengan kontroler PID yang dilengkapi dengan antarmuka *Visual Basic* berbasis Arduino Uno. Tujuan penelitian ini yaitu ingin menghasilkan kecepatan motor DC yang stabil dan sesuai *set point* yang diinginkan. Sistem yang dibuat terdiri dari motor DC, Arduino Uno, sensor *encoder*, driver motor L298N, dan *visual basic*. *Visual basic* berfungsi sebagai antarmuka untuk memberikan input berupa set point dan parameter PID. Dalam menentukan parameter kontroler PID dilakukan metode *trial and error* hingga memperoleh nilai konstanta yang membuat sistem optimal. Hasil penelitian ini yaitu setting hanya dapat dilakukan pada rentang 700 RPM – 1200 RPM karena keterbatasan kemampuan sensor. Selain itu, kinerja kontroler PID yang dihasilkan nilai *overshoot* dengan rata-rata 11,79%, *time settling* dengan rata-rata 4,48% dan *error steady state* 2,5% [16].

Pada tahun 2021, dilakukan penelitian membuat kontrol PID pada kecepatan sudut motor DC dengan menggunakan model identifikasi sistem dengan tujuan model sistem dapat mendekati sistem sesungguhnya. Data identifikasi sistem adalah nilai masukan tegangan dan nilai keluaran kecepatan sudut. Untuk representasi model menggunakan model fungsi alih. Nilai kontrol PID diperoleh menggunakan fitur Tuning PID dengan Matlab. Komponen yang digunakan terdiri dari Arduino, sensor *encoder*, driver motor dan motor DC. Penelitian ini menggunakan beberapa parameter nilai variasi PWM dalam memperoleh nilai respon sistem motor DC sehingga dapat menentukan konstanta kontrol PID yang dapat menghasilkan respon terbaik. Hasilnya diperoleh *rise time* 9,4286 detik, *settling time* 18,5 detik dan *overshoot* 2 % [25].

Selain itu, dilakukan penelitian sistem kendali kecepatan motor DC pada *Conveyor* dengan metode kontrol PID [23]. Tujuan penelitian ini yaitu untuk

memfasilitasi sistem tenaga kerja manusia sekaligus dapat membantu menghemat waktu dan memaksimalkan pekerjaan di dunia industri. Penelitian dilakukan dengan cara membandingkan kinerja *conveyor* menggunakan kontrol PID dengan tanpa kontrol PID. Rancang bangun *conveyor* terdiri dari motor DC 12 V 100 RPM, driver motor *H-Bridge* L298D, motor *encoder*, dan LCD 16x2 sebagai display. Untuk parameter kontrol PID ditentukan dengan metode *trial and error* karakteristik respon PID. Hasil penelitian ini diperoleh *error* rata-rata pengukuran menggunakan kontrol PID yaitu 1,86% dengan waktu mencapai *steady state* 4,92 detik. Sedangkan *error* rata-rata untuk pengukuran tanpa kontrol PID yaitu 12,87% dengan waktu mencapai *steady state* tidak bisa ditentukan karena tidak ada kestabilan data.

Putera dan Hidayat juga melakukan penelitian mengenai kendali kecepatan motor DC. Penelitian ini yaitu kendali kecepatan motor DC menggunakan PID dengan pembacaan *encoder* sebagai *feedback*. Tujuan penelitian ini yaitu ingin mengimplementasikan pengendali PID sebagai pengendali kecepatan motor DC sehingga dapat memahami karakteristik masing-masing parameter pengendali PID terhadap kurva respon sistem. Komponen yang digunakan terdiri dari Arduino sebagai mikrokontroler, *driver* motor L298N, dan motor DC yang telah dilengkapi dengan sensor *encoder*. Hasil penelitian ini yaitu dapat diketahui pengaruh masing-masing parameter kontrol PID yaitu K_p , K_i , dan K_d terhadap respon sistem. Contohnya jika menambahkan parameter K_p dapat mempengaruhi berkurangnya *steady-state error*, mengurangi waktu *rise time*, namun dapat meningkatkan *overshoot*. Sedangkan menambahkan parameter K_i dapat mempengaruhi meningkatnya *overshoot*, mengurangi waktu *rise time*, dan mengurangi bahkan hampir menghilangkan nilai *steady-state error*. Lalu dengan menambahkan parameter K_d dapat mempengaruhi berkurangnya *overshoot*, mengurangi *rise time*, dan meningkatkan *undershoot* [17].

2.7 Pemodelan Identifikasi Sistem Matlab

Motor DC memiliki prinsip kerja apabila motor diberi daya maka arus akan mengalir ke dalam motor dan menghasilkan torsi putar yang sebanding dengan

ardusnya. Berdasarkan prinsip kerja tersebut suatu motor DC memiliki model matematis sesuai dengan karakteristiknya. Model matematis motor DC diperoleh melalui sinyal-sinyal input-output proses dan merupakan representasi fungsi alih (*transfer function*) [25]. Input sistem adalah tegangan yang diberikan pada motor DC sedangkan *output* sistem berupa kecepatan sudut motor DC.

Pada penelitian ini, digunakan perangkat lunak *Matlab* untuk menentukan model motor DC yang kemudian akan diuji hasil identifikasi sistem yang telah diperoleh. Identifikasi sistem dilakukan dengan cara menggunakan Sistem Identifikasi *toolbox* (SIT) pada *Matlab*. SIT adalah alat untuk membuat model matematika berdasarkan data input dan *output* sistem. Proses identifikasi menggunakan SIT lebih cepat dan lebih akurat jika dibandingkan dengan metode konvensional. [26].



Gambar 2. 9 Tampilan SIT pada Matlab