

BAB II TINJAUAN PUSTAKA

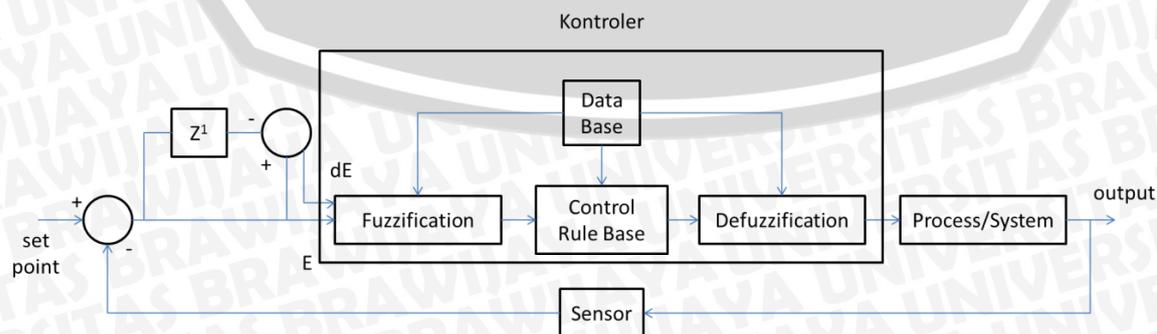
2.1 Kontroler Logika Fuzzy (KLF)

Ide dasar dari kontroler logika fuzzy muncul dari prinsip ketidakjelasan. Mungkin banyak orang kurang tertarik karena istilah ketidakjelasan tersebut. Hal yang jelas saja belum tentu dapat menyelesaikan masalah, apalagi hal yang tidak jelas. Itulah argumen dari sebagian orang. Teori fuzzy pertama kali dibangun dengan menganut prinsip berpikir teori himpunan. Dalam himpunan konvensional (*non fuzzy* atau *crisp*), elemen dari semesta adalah anggota atau bukan anggota dari himpunan. Dengan demikian, keanggotaan dari himpunan adalah tetap, yaitu “ya” (anggota himpunan) atau “tidak” (bukan anggota himpunan) (Robandi, 2006:50).

Himpunan fuzzy adalah bentuk umum dari himpunan biasa yang memiliki tingkat keanggotaan dari tiap-tiap elemen yang dibatasi dengan interval $[0,1]$, (baca: nol sampai satu). Oleh karena itu, fungsi keanggotaan himpunan fuzzy memetakan setiap elemen dari semesta dalam batas ruang yang dalam kebanyakan kasus diasumsikan sebagai unit interval. Satu perbedaan dasar dari himpunan *crisp* dan fuzzy adalah bahwa himpunan *crisp* selalu memiliki fungsi keanggotaan yang unik, sedangkan setiap himpunan fuzzy memiliki nilai keanggotaan yang terbatas dari fungsi keanggotaan yang mewakilinya. Hal ini memungkinkan fuzzy dapat diatur secara maksimum dalam situasi yang diberikan (Robandi, 2006:51).

2.1.1 Struktur Dasar Fuzzy Logic

Dalam sistem pengendalian dengan *fuzzy logic* dilibatkan suatu blok pengendali yang menerima satu atau lebih masukan dan mengeluarkan satu atau lebih keluaran ke *plant* atau blok lain sebagaimana ditunjukkan dalam Gambar 2.1. Komponen utama penyusun *fuzzy logic* adalah unit fuzzifikasi, fuzzy inference, dan defuzzifikasi.



Gambar 2. 1 Struktur Dasar Logika Fuzzy.
Sumber : Kuswandi (2007:45).



Suatu himpunan fuzzy (fuzzy set) A adalah dalam semesta pembicaraan (universe of discourse) U dinyatakan dengan fungsi keanggotaan (membership function) μ_A , yang harganya berada dalam interval $[0, 1]$ (Kuswandi, 2007:27). Secara matematika hal ini dinyatakan dengan

$$\mu_A: U \rightarrow [0, 1] \quad (2.1)$$

2.1.2 Fuzzifikasi

Proses fuzzifikasi merupakan proses untuk mengubah variabel non *fuzzy* (variabel numerik) menjadi variabel *fuzzy* (variabel linguistik). Nilai masukan-masukan yang masih dalam bentuk variabel numerik yang telah dikuantisasi sebelum diolah oleh *fuzzy logic* harus diubah terlebih dahulu ke dalam variabel *fuzzy*. Melalui fungsi keanggotaan yang telah disusun, maka dari nilai-nilai masukan tersebut menjadi informasi *fuzzy* yang berguna nantinya untuk proses pengolahan secara *fuzzy* pula. Proses ini disebut fuzzifikasi (Yan, 1994). Pedoman memilih fungsi keanggotaan untuk proses fuzzifikasi, menurut Jun Yan, menggunakan

1. Himpunan *fuzzy* dengan distribusi simetris.
2. Gunakan himpunan *fuzzy* dengan jumlah ganjil, berkaitan erat dengan jumlah kaidah (*rules*).
3. Mengatur himpunan *fuzzy* agar saling menumpuk.

Proses fuzzifikasi diekspresikan sebagai berikut (Kuswandi, 2007:27):

$$x = \text{fuzzyfier}(x_0) \quad (2.2)$$

dengan:

x_0 = nilai *crisp* variabel masukan.

x = himpunan *fuzzy* variabel yang terdefinisi.

fuzzyfier = operator fuzzifikasi yang memetakan himpunan *crisp* ke himpunan *fuzzy*.

2.1.3 Fuzzy Rule

Fuzzy Rule adalah bagian yang menggambarkan dinamika suatu sistem terhadap masukan yang dikarakteriskan oleh sekumpulan variabel-variabel linguistik dan berbasis pengetahuan seorang operator ahli. Pernyataan tersebut umumnya dinyatakan oleh suatu pernyataan bersyarat.

Dalam pengendali berbasis *fuzzy*, aturan pengendalian *fuzzy* berbentuk aturan "IF-THEN". Untuk sebuah sistem *Multi Input Single Output* (MISO) basis aturan pengendalian *fuzzy* berbentuk seperti berikut ini,

Rule 1 IF X is A_1 AND Y is B_1 THEN Z is C_1

Rule 2 IF X is A_2 AND Y is B_2 THEN Z is C_2

Rule n IF X is A_n AND Y is B_n THEN Z is C_n

Dengan X, Y, Z merupakan variabel linguistik, dimana X dan Y merupakan variabel masukan, dan Z merupakan variabel keluaran sistem. A_n , B_n , C_n merupakan nilai linguistik dari X, Y, dan Z (Yan, 1994) (Kuswandi, 2007:28).

2.1.4 Metode Inferensi MAX-MIN

Metode inferensi merupakan proses untuk mendapatkan keluaran dari suatu kondisi masukan dengan mengikuti aturan-aturan yang telah ditetapkan. Keputusan yang didapatkan pada proses ini masih dalam bentuk *fuzzy* yaitu derajat keanggotaan keluaran. Pada metode *Max-Min* aturan operasi minimum Mamdani digunakan untuk implikasi *fuzzy* (Yan, 1994). Sebagai contoh, terdapat dua basis kaidah atur *fuzzy*, yaitu:

R_1 : Jika x adalah A_1 dan y adalah B_1 maka z adalah C_1 .

R_2 : Jika x adalah A_2 dan y adalah B_2 maka z adalah C_2 .

$$\mu_{C_i} = \bigcup_{i=1}^n \alpha_i \wedge \mu_{C_i} \quad (2.3)$$

dengan $\alpha_i = \mu_{A_i}(X_0) \wedge \mu_{B_i}(Y_0)$

Pada metode penalaran MAX-MIN fungsi keanggotaan konsekuen dinyatakan dengan (Kuswandi, 2007:30)

$$\mu_{C_1'}(W) = \mu_{C_1} \vee \mu_{C_2} = [\alpha_1 \wedge \mu_{B_1}(W)] \vee [\alpha_2 \wedge \mu_{C_2}(W)] \quad (2.4)$$

$$\alpha_1 = \mu_{A_1}(X_0) \wedge \mu_{B_1}(Y_0) \quad (2.5)$$

$$\alpha_2 = \mu_{A_2}(X_0) \wedge \mu_{B_2}(Y_0) \quad (2.6)$$

2.1.5 Metode Defuzzifikasi *Weight Average*

Defuzzifikasi adalah proses untuk mendapatkan nilai numerik dari data *fuzzy* yang dihasilkan dari proses inferensi (Yan, 1994). Proses defuzzifikasi dinyatakan sebagai berikut (Kuswandi, 2007:31):

$$y_0 = \text{defuzzifier}(y) \quad (2.7)$$

dengan:

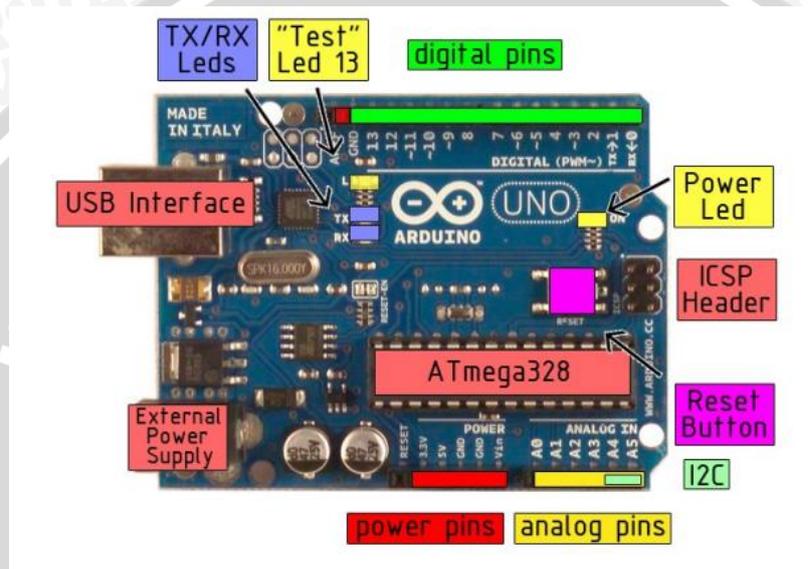
y = aksi kontrol *fuzzy*.

y_0 = aksi kontrol *crisp*.

defuzzifier = operator defuzzifikasi.

2.2 Arduino Uno

Arduino Uno adalah sebuah board mikrokontroler berbasis ATmega328. Arduino Uno mempunyai 14 pin digital input/output (6 di antaranya dapat digunakan sebagai output PWM), 6 input analog, sebuah osilator Kristal 16 MHz, sebuah koneksi USB, sebuah power jack, sebuah ICSP header, dan sebuah tombol reset. Arduino Uno memuat semua yang dibutuhkan untuk menunjang mikrokontroler, mudah menghubungkannya ke sebuah komputer dengan sebuah kabel USB atau mensuplainya dengan sebuah adaptor AC ke DC atau menggunakan baterai untuk memulainya (Krisnanda, 2014:18).



Gambar 2. 2 Skema Arduino Uno.

Sumber: Allied Electronics (2005:3).

Arduino Uno memiliki sejumlah fasilitas untuk berkomunikasi dengan komputer, Arduino lain, atau mikrokontroler lainnya. ATmega328 menyediakan UART TTL (5V) untuk komunikasi serial, yang tersedia di pin digital 0 (RX) dan 1 (TX). Sebuah ATmega8U2 sebagai saluran komunikasi serial melalui USB dan sebagai port virtual com untuk perangkat lunak pada komputer. *Firmware* '8 U2 menggunakan *driver* USB standar COM dan tidak ada *driver* eksternal yang diperlukan, namun pada OS Windows diperlukan sebuah *file* inf. Perangkat lunak Arduino terdapat serial monitor yang memungkinkan digunakan untuk memonitor data tekstual sederhana yang akan dikirim ke atau dari *board* Arduino. LED RX dan TX di *board* akan berkedip ketika data sedang dikirim melalui chip USB-*to*-serial dengan koneksi USB ke komputer (tetapi tidak untuk komunikasi serial pada pin 0 dan 1). Sebuah *Software Serial library* memungkinkan untuk berkomunikasi secara serial pada salah satu pin digital pada *board* Uno. ATmega328 juga mendukung I2C (TWI) dan komunikasi SPI. Perangkat lunak Arduino termasuk *Wire library* untuk menyederhanakan penggunaan bus I2C.

2.3 Motor BLDC

BLDC motor atau dapat disebut juga dengan BLAC motor merupakan motor listrik synchronous AC 3 fasa. Perbedaan pemberian nama ini terjadi karena BLDCM memiliki BEMF berbentuk trapezoid, sedangkan BLACM memiliki BEMF berbentuk sinusoidal. Walaupun demikian keduanya memiliki struktur yang sama dan dapat dikendalikan dengan metode six-step maupun PWM sinusoidal. Secara umum BLDCM terdiri dari dua bagian, yakni rotor, bagian yang bergerak, yang terbuat dari permanen magnet dan stator, bagian yang tidak bergerak, yang terbuat dari kumparan 3 fasa. Walaupun merupakan motor listrik synchronous AC 3 fasa, motor ini tetap disebut dengan BLDCM karena pada implementasinya BLDCM menggunakan sumber DC sebagai sumber energi utama yang kemudian diubah menjadi tegangan AC dengan menggunakan inverter 3 fasa. Tujuan dari pemberian tegangan AC 3 fasa pada stator BLDCM adalah menciptakan medan magnet putar stator untuk menarik magnet rotor. Motor BLDC ini dapat bekerja ketika stator yang terbuat dari kumparan diberikan arus 3 fasa. Akibat arus yang melewati kumparan pada stator timbul medan magnet (B): (Azzumar, 2012:20)

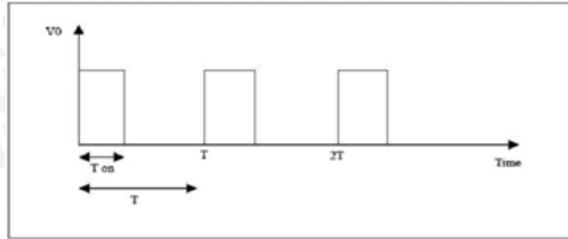
$$B = \frac{\mu Ni}{2l} \quad (2-8)$$

Di mana N merupakan jumlah lilitan, i merupakan arus, l merupakan panjang lilitan dan μ merupakan permeabilitas bahan. Karena arus yang diberikan berupa arus AC 3 fasa sinusoidal, nilai medan magnet dan polarisasi setiap kumparan akan berubah-ubah setiap saat. Akibat yang ditimbulkan dari adanya perubahan polarisasi dan besar medan magnet tiap kumparan adalah terciptanya medan putar magnet dengan kecepatan, (Azzumar, 2012:21)

$$n_s = \frac{120f}{p} \quad (2-9)$$

2.4 PWM (*Pulse With Modulation*)

PWM (*Pulse Width Modulation*) digunakan untuk mengatur kecepatan dari motor *servo*. Dimana kecepatan motor *servo* tergantung pada besarnya *duty cycle* yang diberikan pada motor *servo* tersebut. Pada sinyal PWM, frekuensi sinyal konstan sedangkan *duty cycle* bervariasi dari 0%-100%. Dengan mengatur *duty cycle* akan diperoleh keluaran yang diinginkan. Sinyal PWM (*Pulse Width Modulation*) secara umum dapat dilihat dalam Gambar 2.3 berikut (Krisnanda, 2014:15-16).



Gambar 2. 3 Gambar Sinyal PWM Secara Umum
Sumber: electronics-scheme.com

$$Duty\ cycle = \frac{T_{on}}{T} \times 100\% \quad (2-10)$$

Dengan:

T_{on} = Periode logika tinggi

T = Periode keseluruhan

$$V_{dc} = Duty\ cycle \times V_{cc}$$

Sedangkan frekuensi sinyal dapat ditentukan dengan rumus berikut:

$$f_{0n} = \frac{f_{clk} / 0}{N \cdot 256} \quad (2-11)$$

2.5 Baling-baling (Propeller)

Baling-baling adalah alat yang mengubah gerak putar menjadi daya dorong. Daya dorong inilah yang dimanfaatkan pesawat terbang dan kapal laut sebagai penghasil daya dorong utama. Pembahasan baling-baling pada tugas akhir ini dibatasi hanya pada paramater baling-baling yang digunakan dalam RC (*Radio Control*) *aeromodelling*. Ada beberapa parameter penting yang dimiliki baling-baling pada RC *aeromodelling*. Parameter-parameter ini bisa dijadikan pedoman untuk memilih baling-baling sesuai dengan kebutuhan:

1. Diameter dan *pitch*

Semua baling-baling RC yang tersedia memiliki 2 buah ukuran, yaitu diameter dan *pitch*. Diameter dihitung berdasarkan diameter lingkaran yang dibentuk saat baling-baling berputar. Jika baling dianalogikan sebagai sebuah sekrup, *pitch* merupakan jarak yang ditempuh oleh baling-baling jika diptar 1 putaran penuh. Semakin panjang diameter dan *pitch* baling-baling semakin banyak pula udara disapu dan semakin besar pula daya dorong yang dihasilkan. Tapi diameter dan *pitch* dari baling-baling ini harus disesuaikan dengan motor dan sumber daya yang digunakan. Biasanya produsen motor sudah memberikan spesifikasi baling-baling untuk motornya.

2. Jumlah bilah

Umumnya, jumlah bilah pada baling-baling RC *aeromodelling* adalah 2 bilah, Tetapi ada beberapa yang menggunakan 3 bilah dan 4 bilah. Semakin banyak bilah pada baling-

balung menyebabkan banyak udara yang disapu sehingga menghasilkan daya dorong yang lebih besar. Biasanya penambahan jumlah bilah bertujuan untuk memperkecil diameter balung-balung, tentunya untuk menghasilkan performa yang sama (dengan motor yang sama) *pitch*-nya harus dikurangi.

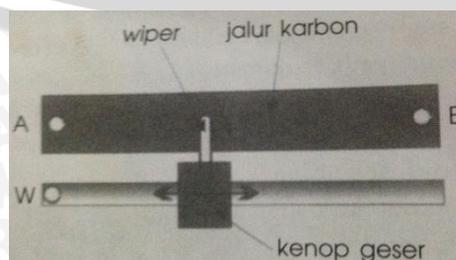
3. Arah putar

Dengan arah gaya dorong yang sama, balung-balung RC *aeormodelling* memiliki dua jenis arah putaran: searah jarum jam (CW, *clockwise*) dan berkebalikan arah jarum jam (CCW, *counter clockwise*). Arah putar ini menentukan *yawing moment* yang dihasilkan dari balung-balung. Pada *Becopter*, dibutuhkan sepasang balung-balung CW dan CCW agar *yawing moment* saling menghilangkan (Kristianto, 2012:13-15).

2.6 Potensiometer

Apabila kita ingin dapat mengubah-ubah nilai tahanan pada salah satu bagian rangkaian listrik, kita membutuhkan sebuah resistor variabel. Salah satu jenis resistor variabel adalah potensiometer. Jenis ini seringkali digunakan untuk aplikasi-aplikasi pengaturan volume suara pada perangkat audio. Potensiometer seringkali disebut sebagai 'pot' saja. Gambar 2.4 memperlihatkan sebuah potensiometer geser, yang memiliki sebuah pita film, disebut sebagai 'jalur' (track), yang terbuat dari karbon.

Terminal ketiga dari komponen ini disambungkan ke pa yang disebut *wiper*. Wiper adalah sebuah *strip* (lempengan kecil dan tipis) logam yang bersifat lentur, yang menempel dan menekan kuat pada jalur karbon untuk membentuk suatu hubungan listrik. Wiper terpasang pada sebuah kenop geser, yang digunakan untuk memindah-mindahkan posisi wiper di sepanjang jalur karbon. Dengan berpindahnya wiper pada jalur karbon, jarak antara salah satu ujung jalur (misalnya, ujung A) dengan posisi wiper (W) mengalami perubahan. Perubahan jarak ini berdampak pada berubahannya nilai tahanan listrik antara A dan W. Tahanan listrik potensiometer dapat memiliki nilai antara nol hingga nilai penuh yang dapat diberikan seluruh jalur karbon. Pot-pot geser seringkali digunakan pada perangkat audio untuk mengatur tanggapan frekuensi (Bishop, 2002:32).



Gambar 2. 4 Pemodelan potonsiometer.

Sumber : Bishop (2002:32)

