

# Implementasi Logika *Fuzzy* Pada Sendok Makan Penderita *Parkinson*

Muhammad Naufal Halim<sup>1</sup>, Ahmad Zarkasih<sup>1</sup>, Arti Dian Nastiti<sup>2</sup>

<sup>1</sup> Fakultas Ilmu Komputer, Universitas Sriwijaya, Palembang, Indonesia

<sup>2</sup> Balai Jaringan Informasi dan Komunikasi, Badan Pengkajian dan Penerapan Teknologi  
Email : zarkasi98@gmail.com

**Abstrak**— Penelitian ini dimaksudkan untuk membuat rancang bangun alat bantu makan berupa sendok yang dikhususkan untuk penderita Parkinson. Implementasi sendok penderita Parkinson dalam penelitian ini menggunakan sensor 3-Axis gyroscope accelerometer MPU-6050, mikrokontroler ESP32, dan motor stepper SG-90 untuk menstabilkan posisi sendok akibat getaran tangan penderita Parkinson. Data yang diperoleh dari penelitian ini adalah sensor MPU-6050 yang mendeteksi getaran tangan penderita lalu data tersebut distabilkan menggunakan logika Fuzzy yang berfungsi untuk meminimalisir error pada sensor MPU-6050.

**Kata Kunci**— ESP32, Logika Fuzzy, MPU-6050, Parkinson, Sendok Parkinson, SG-90.

## I. PENDAHULUAN

Perkembangan dunia teknologi informasi dari hari ke hari semakin bertambah pesat. Hampir seluruh aspek kehidupan manusia kini tidak terlepas dari pengaruh teknologi. Teknologi dapat mempermudah seseorang dalam melakukan segala hal, seperti halnya dalam dunia kedokteran dan kesehatan yang akhir-akhir ini kian berkembang. Karena atas dasar hal itu, kebutuhan akan teknologi robotik yang dapat membantu manusia di dalam bidang kesehatan menjadi semakin tinggi.

Penyakit Parkinson merupakan penyakit yang paling umum, terutama pada lanjut usia. Penyakit Parkinson adalah sebuah penyakit progresif neurodegenerative yang menyebabkan kekurangan kontrol motor progressive, kognitif, dan visual visual. Hal itu menunjukkan bahwa Penyakit Parkinson adalah penyakit kedua yang paling sering dijumpai setelah penyakit Alzheimer dan mempengaruhi bagian terbesar populasi dari lanjut usia. Pada tahun 2015 jumlah lansia sebanyak 20 juta berdasarkan data BPS, dan 200.000 orang diantaranya menderita Parkinson [1].

Banyak penelitian untuk merancang perangkat bantu makan bagi penderita parkinson. Kebanyakan rancangan perangkat merupakan perangkat yang menggantikan alat penyuaian makan kepada penderita.

Salah satu metode yang dapat digunakan untuk alat bantu penderita Parkinson ini menggunakan sensor MPU6050 untuk mendeteksi dan merasakan getaran tangan penderita. Tiga buah motor stepper SG-90 digunakan untuk menstabilkan posisi sendok makan penderita. Dan mikrokontroler ESP32 digunakan untuk memproses keseluruhan program. Lalu penggunaan logika fuzzy adalah untuk mengurangi error yang besar dari hasil keluaran sensor MPU-6050.

Berdasarkan uraian diatas, penelitian ini bertujuan merancang *prototype* sendok parkinson yang bisa bermanfaat

untuk penderita parkinson yang telah diimplementasikan logika fuzzy agar pembacaan sensor lebih akurat.

## II. STUDI PUSTAKA

### A. Penyakit Parkinson

Penyakit Parkinson adalah sebuah kelainan pada sistem saraf pusat, yang meliputi sebuah degenerasi awal dari beberapa sel saraf pada bagian terdalam dari otak yang dinamakan basal ganglia dan secara khusus sebuah kehilangan sel-sel saraf (neurons) pada sebuah bagian dari batang otak yang dinamakan substantia nigra. Yang mana sel-sel ini membuat reaksi kimia saraf (neurochemical) untuk membuat pesan agar dapat mengkoordinasikan gerakan yang normal.

Penyakit Parkinson ini terjadi kurang lebih sama antara wanita dan pria. Gejala awal mungkin bisa muncul pada berbagai umur, meskipun dibawah 40 tahun tidak umum dan dibawah 20 sangat jarang. Pada umumnya terjadi pada umur 60 sampai dengan 70 tahun. Umur rata-rata penderita Penyakit i adalah 59 tahun.

Beberapa gejala umum pada pendeita Parkinson antara lain tremor / bergetar pada jari, jempol, tangan, dagu atau bibir, tulisan tangan mengecil, kehilangan indra penciuman, gangguan tidur, kesulitan/gangguan dalam bergerak atau berjalan, sembelit, suara menjadi lebih kecil [1].

Insiden penyakit parkinson di Amerika Serikat sekitar 1 juta orang pada tahun 2010 sedangkan diseluruh dunia penderita mencapai 5 juta orang. Kebanyakan individu yang mengalami penyakit parkinson berusia lebih dari 60 tahun. Penyakit Parkinson terjadi pada sekitar 1% individu berusia 60 tahun dan sekitar 4% pada orang yang berusia 80tahun. Karena harapan hidup secara keseluruhan meningkat, jumlah orang dengan penyakit parkinson akan meningkat di masa depan [2].

### B. Mikrokontroler

Mikrokontroler merupakan sebuah chip yang fungsinya sebagai pengontrol rangkaian elektronik dan umumnya dapat menyimpan program didalamnya. Meskipun mempunyai bentuk yang jauh lebih kecil dari suatu komputer pribadi dan komputer mainframe, mikrokontroler dibangun dari elemen-elemen dasar yang sama. Secara sederhana, komputer akan menghasilkan output spesifik berdasarkan inputan yang diterima dan program yang dikerjakan.

Seperti umumnya komputer, mikrokontroler adalah alat yang mengerjakan instruksi-instruksi yang diberikan kepadanya. Artinya bagian terpenting dan utama dari suatu sistem terkomputerisasi adalah program itu sendiri. Program ini mengintruksikan komputer untuk melakukan jalinan yang panjang dari aksi-aksi sederhana untuk melakukan tugas yang lebih kompleks yang diinginkan oleh programmer. Dengan kata lain, mikrokontroler adalah suatu alat elektronika digital yang mempunyai masukan dan keluaran serta kendali dengan program yang bisa ditulis dan dihapus dengan cara khusus [3].

### C. ESP32

ESP32 adalah mikrokontroler yang dikenalkan oleh Espressif Sistem dan berfungsi untuk menampung dan memproses semua port dan ic sehingga bisa mengontrol driver sehingga port atau device yang terhubung ke Mikrokontroler tersebut dapat berjalan dengan baik. Mikrokontroler ini juga memiliki kemampuan untuk terhubung dengan internet melalui jaringan wireless tanpa tambahan board lagi karena sudah tersedia modul WiFi dalam chip sehingga sangat mendukung untuk membuat sistem aplikasi Internet of Things [4].

ESP32 dikenalkan oleh Espressif Sistem yang merupakan penerus dari mikrokontroler ESP8266. Mikrokontroler ESP32 memiliki keunggulan yaitu sistem berbiaya rendah, dan juga berdaya rendah dengan modul WiFi yang terintegrasi dengan chip mikrokontroler serta memiliki bluetooth dengan mode ganda dan fitur hemat daya menjadikannya lebih fleksibel. ESP32 kompatibel dengan perangkat seluler dan aplikasi IoT (Internet of Things). Mikrokontroler ini dapat digunakan sebagai sistem mandiri yang lengkap atau dapat dioperasikan sebagai perangkat pendukung mikrokontroler host [5].

### D. MPU-6050

MPU-6050 adalah chip IC inverse yang didalamnya terdapat sensor Accelerometer dan Gyroscope yang sudah terintegrasi. Accelerometer digunakan untuk mengukur percepatan, percepatan gerakan dan juga percepatan gravitasi. Accelerometer sering digunakan untuk menghitung sudut kemiringan, dan hanya dapat melakukan dengan nyata ketika statis dan tidak bergerak. Untuk mendapatkan sudut akurat kemiringan, sering dikombinasikan dengan satu atau lebih gyro dan kombinasi data yang digunakan untuk menghitung sudut. Gyroscope adalah perangkat untuk mengukur atau mempertahankan orientasi, yang berlandaskan pada prinsip-prinsip momentum sudut. Mikrokontroler ATmega128 merupakan salah satu varian dari mikrokontroler AVR 8-bit. Beberapa fitur yang dimiliki adalah memiliki beberapa memori yang bersifat non-volatile, yaitu 128Kbytes of In-System Self-Programmable Flash program memory (128Kbytes memory flash untuk pemrograman), write/erase cycles: 10.000 Flash/ 100.000 EEPROM (program dalam mikrokontroler dapat diisi dan dihapus berulang kali sampai 10.000 kali untuk flash memori atau 100.000 kali untuk penyimpanan program/data di EEPROM) [6].

MPU-6050 adalah chip IC inverse yang didalamnya terdapat sensor Accelerometer dan Gyroscope, dimana:

#### 1) Gyroscope

Gyroscope berfungsi untuk mengukur/menentukan orientasi suatu benda berdasarkan pada ketetapan momentum sudut. Pengertian lain dari gyroscope berfungsi untuk menentukan gerakan sesuai dengan gravitasi yang dilakukan oleh pengguna. Gyroscope memiliki peranan yang sangat penting dalam hal mempertahankan keseimbangan suatu benda yang dapat menentukan kemiringan pada sumbu X, Y, dan Z. Prinsip kerja dari gyroscope berotasi maka akan memiliki nilai keluaran. Apabila gyroscope berotasi searah dengan jarum jam pada sumbu Z, maka tegangan output yang dihasilkan akan mengecil sedangkan gyroscope berotasi berlawanan arah dengan jarum jam pada sumbu Z maka tegangan output yang dihasilkan akan membesar. Pada saat gyroscope tidak sedang berotasi atau berada pada keadaan diam maka tegangan output akan sesuai dengan nilai offset gyroscope tersebut [7].

Gyroscope terdiri dari (rotor) yang berputar pada sumbu putar (spin axis). Sumbu putar ini terpasang pada suatu kerangka yang disebut imbal (inner-most gimbal). Inner-most gimbal terpasang pada inner gimbal. Inner gimbal terpasang pada outer gimbal yang merupakan kerangka terluar. Dengan memiliki tiga gimbal maka gyroscope mempunyai kemampuan untuk berputar pada tiga sumbu putar (3 degree of rotational freedom). Meskipun gyroscope mempunyai 3 degree of rotational freedom, namun rotor akan tetap berada pada posisinya selama rotor berputar. Saat ketiga kerangka gimbal berputar, maka rotor tidak mengikuti putarannya.

Sensor gyroscope dapat mendeteksi gerakan sesuai gravitasi atau dengan kata lain mendeteksi gerakan pengguna. Gyroscope memiliki keluaran berupa kecepatan sudut dari 3 arah sumbu yaitu sumbu X yang nantinya akan menjadi sudut phi (kanan dan kiri), sumbu Y nantinya menjadi sudut theta (atas dan bawah) dan sumbu Z nantinya menjadi sudut psi (depan dan belakang)[7].

#### 2) Accelerometer

*Accelerometer* adalah alat yang digunakan untuk mengukur percepatan, mendeteksi dan mengukur getaran (vibrasi), dan mengukur percepatan akibat gravitasi (inklinasi). Sensor accelerometer mengukur percepatan akibat gerakan benda yang melekat padanya.

*Accelerometer* dapat digunakan untuk mengukur getaran pada mobil, mesin, bangunan, dan instalasi pengamanan. Sensor *accelerometer* juga dapat diaplikasikan pada pengukuran aktivitas gempa bumi dan peralatan-peralatan elektronik, seperti permainan 3 dimensi, mouse komputer, dan telepon. Untuk aplikasi yang lebih lanjut, sensor ini banyak digunakan untuk keperluan navigasi [8].

### E. Logika Fuzzy

Logika fuzzy (*Fuzzy Logic*) adalah metode mendefinisikan variable yang samar-samar seperti kata “agak cepat”, “agak lambat”, “sedikit panas” dan sebagainya. Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak, atau kombinasi keduanya [9].

Konsep logika fuzzy pertama kali diperkenalkan pada bulan Juni 1965 oleh Professor Lotfi A. Zadeh dari Universitas California untuk menyatakan kelompok/himpunan yang dapat dibedakan dengan himpunan lain berdasarkan derajat keanggotaan dengan batasan yang tidak begitu jelas (samar) [10].

Logika Fuzzy Mamdani merupakan salah satu bagian dari Fuzzy Inference Sistem yang berguna untuk penarikan kesimpulan atau suatu keputusan terbaik dalam permasalahan yang tidak pasti [11]. Metode ini lebih cocok apabila input diterima dari manusia, sehingga lebih diterima oleh banyak pihak. Adapun kelemahan dari Logika Fuzzy Mamdani adalah metode ini hanya dapat digunakan untuk data dalam bentuk kuantitatif saja, tidak dapat dipergunakan untuk data yang berbentuk kualitatif [11].

Menurut Kusumadewi [12], metode Mamdani sering juga dikenal dengan nama Metode Max-Min. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan output, diperlukan 4 tahapan:

1. Pembentukan himpunan *fuzzy*  
Pada Metode Mamdani, baik variabel *input* maupun variabel *output* dibagi menjadi satu atau lebih himpunan *fuzzy*.
2. Aplikasi fungsi implikasi  
Pada Metode Mamdani, fungsi implikasi yang digunakan adalah Min.
3. Komposisi Aturan  
Tidak seperti penalaran monoton, apabila sistem terdiri dari beberapa aturan, maka inferensi diperoleh dari kumpulan dan korelasi antar aturan. Ada 3 metode yang digunakan dalam melakukan inferensi sistem fuzzy, yaitu: max, additive dan probabilistik OR (probor).

#### a. Metode Max (*Maximum*)

Pada metode ini, solusi himpunan *fuzzy* diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah *fuzzy*, dan mengaplikasikannya ke output dengan menggunakan operator OR (union). Jika semua proposisi telah dievaluasi, maka output akan berisi suatu himpunan *fuzzy* yang merefleksikan kontribusi dari tiap-tiap proposisi. Secara umum dapat dituliskan pada persamaan (1)

$$\mu_{sf} \leftarrow \max(\mu_{sf}[x_i], \mu_{kf}[x_i]) \quad (1)$$

dengan:

$\mu_{sf}[x_i]$  = nilai keanggotaan solusi *fuzzy* sampai aturan ke-i;

$\mu_{kf}[x_i]$  = nilai keanggotaan konsekuen *fuzzy* aturan ke-i;

Apabila digunakan fungsi implikasi MIN, maka metode komposisi ini sering disebut dengan nama MAX-MIN atau MIN-MAX atau MAMDANI.

- b. Metode Additive (*Sum*) Pada metode ini, solusi himpunan *fuzzy* diperoleh dengan cara melakukan bounded-sum terhadap semua output daerah *fuzzy*. Secara umum dituliskan pada persamaan (2):

$$\mu_{sf} \leftarrow \min(1, \mu_{sf}[x_i] + \mu_{kf}[x_i]) \quad (2)$$

dengan:

$\mu_{sf}[x_i]$  = nilai keanggotaan solusi *fuzzy* sampai aturan ke-i;

$\mu_{kf}[x_i]$  = nilai keanggotaan konsekuen *fuzzy* aturan ke-i;

- c. Metode Probabilistik OR (*probor*)

Pada metode ini, solusi himpunan *fuzzy* diperoleh dengan cara melakukan product terhadap semua output daerah *fuzzy*. Secara umum dituliskan pada persamaan (3):

$$\mu_{sf} \leftarrow (\mu_{sf}[x_i] + \mu_{kf}[x_i]) - (\mu_{sf}[x_i] - \mu_{kf}[x_i]) \quad (3)$$

dengan:

$\mu_{sf}[x_i]$  = nilai keanggotaan solusi *fuzzy* sampai aturan ke-i;

$\mu_{kf}[x_i]$  = nilai keanggotaan konsekuen *fuzzy* aturan ke-i;

#### 4. Penegasan (*defuzzy*)

Input dari proses defuzzifikasi adalah suatu himpunan *fuzzy* yang diperoleh dari komposisi aturan-aturan *fuzzy*, sedangkan output yang dihasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam range tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai output. Ada beberapa metode defuzzifikasi pada komposisi aturan MAMDANI, antara lain:

- a. Metode Centroid (*Composite Moment*)

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat ( $z^*$ ) daerah *fuzzy*. Secara umum dirumuskan pada persamaan (4) dan (5)

$$z^* = \frac{\int_z z\mu(z)dz}{\int_z \mu(z)dz} \text{ untuk variabel kontinu} \quad (4)$$

$$z^* = \frac{\sum_{j=i}^n z_j^\mu(z_j)}{\sum_{j=i}^n \mu(z_j)} \text{ untuk variabel diskrit} \quad (5)$$

- b. Metode Bisektor

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai pada domain *fuzzy* yang memiliki nilai keanggotaan setengah dari jumlah total nilai keanggotaan pada daerah *fuzzy*. Secara umum dituliskan pada persamaan (6):

$z_p$  sedemikian sehingga

$$\int_{R_i}^p \mu(z) dz = \int_p^{R_n} \mu(z) dz \quad (6)$$

- c. Metode *Mean of Maximum* (MOM)  
Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai rata-rata domain yang memiliki nilai keanggotaan maksimum.
- d. Metode *Largest of Maximum* (LOM)  
Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.
- e. Metode *Smallest of Maximum* (SOM)  
Pada metode ini, solusi crisp diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

#### F. Motor Servo

Motor servo adalah sebuah perangkat sebagai aktuator putar (motor) yang dirancang dengan sistem kontrol umpan balik loop tertutup (servo), sehingga dapat di set-up atau di atur untuk menentukan dan memastikan posisi sudut dari poros output motor. Motor servo merupakan perangkat yang terdiri dari motor DC, serangkaian gear, rangkaian kontrol dan potensiometer. Serangkaian gear yang melekat pada poros motor DC akan memperlambat putaran poros dan meningkatkan torsi motor servo, sedangkan potensiometer dengan perubahan resistansinya saat motor berputar berfungsi sebagai penentu batas posisi putaran poros motor servo.

Motor servo biasa digunakan dalam aplikasi-aplikasi di industri, selain itu juga digunakan dalam berbagai aplikasi lain seperti pada mobil mainan radio kontrol, robot, pesawat, dan lain sebagainya [13].

Motor Servo yang digunakan pada robot ini adalah motor servo jenis Tower Pro Micro Servo SG90. Motor servo jenis ini akan bekerja secara baik jika pada bagian pin kontrolnya diberikan sinyal PWM dengan frekuensi 50Hz dengan periode sebesar 20 ms. Pemberian besar pulsa dari mikrokontroler menentukan besar sudut yang harus dilakukan oleh motor servo.

Pengaturan sudut motor servo diperlukan untuk mengetahui gerakan dari motor servo dan pulsa yang harus diberikan ke motor servo dalam pergerakan ke kanan atau ke kiri. Dari pulsa yang diberikan, kita dapat melihat gerakan motor servo. Di mana pada saat sinyal dengan frekuensi 50Hz tersebut dicapai pada kondisi Ton duty cycle 1.5 ms, maka rotor dari motor akan berhenti tepat di tengah-tengah (sudut 90° / netral). Pada saat Ton duty cycle dari sinyal yang diberikan kurang dari 1.5ms, maka rotor akan berputar ke arah kiri dengan membentuk sudut yang besarnya linier terhadap besarnya Ton duty cycle sampai batas minimum 0°, dan akan bertahan diposisi tersebut. Dan sebaliknya, jika Ton duty cycle dari sinyal yang diberikan lebih dari 1.5ms, maka rotor akan berputar ke arah kanan dengan membentuk sudut yang linier pula terhadap besarnya Ton duty cycle sampai batas sudut maksimum 180°, dan bertahan diposisi

tersebut[13]. Untuk lebih jelasnya karakteristik motor servo dapat dijelaskan oleh tabel 2.4 dibawah ini

### III. METODOLOGI PENELITIAN

Tahapan-tahapan metodologi pada proyek ini sebagai berikut :

1. Metode Literatur  
Pada tahapan metode ini penulis melakukan studi pustaka dengan mencari serta mengumpulkan berbagai sumber referensi berupa literatur yang terdapat pada buku, internet maupun sumber lainnya tentang Sendok Parkinson, Sensor MPU-6050, dan Logika Fuzzy.
2. Metode Konsultasi  
Pada tahapan metode ini penulis melakukan konsultasi dengan orang – orang yang memiliki pengetahuan dan pengalaman terhadap permasalahan didalam proyek yang dibuat oleh penulis.
3. Metode Perancangan Sistem  
Pada tahapan metode ini penulis melakukan rancangan terhadap sistem baik berupa *software* maupun *hardware*.
4. Metode Pengujian  
Pada tahapan metode ini penulis melakukan pengujian terhadap rancangan sistem yang dibuat apakah sistem dapat bekerja sehingga diperoleh data yang akurat dari hasil pengujian proyek ini.
5. Metode Analisa dan Kesimpulan  
Pada tahapan metode ini penulis melakukan analisis dari pengujian sistem dengan tujuan untuk mengetahui kekeurangan dari hasil penelitian proyek, sehingga dapat digunakan untuk pengembangan penelitian selanjutnya dan setelah menganalisa dibuatlah kesimpulan dari hasil pengujian.

### IV. HASIL DAN PEMBAHASAN

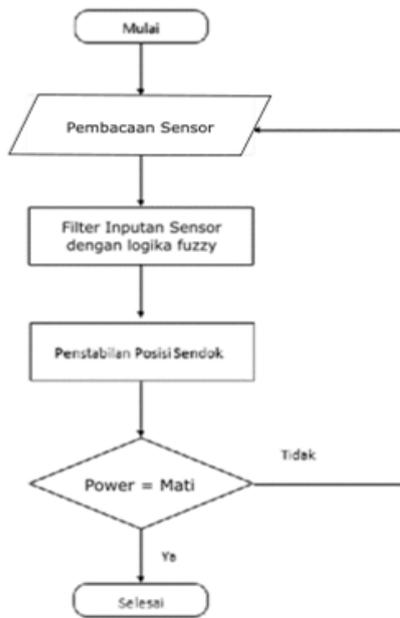
#### A. Rancangan Alat

##### 1) Perancangan Alat

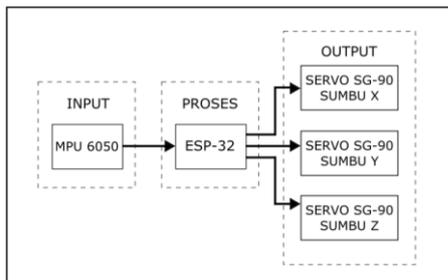
Pada perancangan alat sendok *Parkinson* ini, metodologi yang digunakan adalah penerapan logika *fuzzy* pada rancang bangun sendok makan untuk penderita *Parkinson*. Kerangka kerja sistem dapat dilihat dalam Gambar 1.

Pada Gambar 2 menjelaskan proses awal dari sistem dimulai dari sensor gyroscope (MPU-6050) yang mendeteksi posisi dari pemegang sendok, dari hasil pembacaan tersebut selanjutnya akan dibaca oleh ESP32 dan data akan disimpan di ESP32 selaku mikrokontroler dari sendok.

Selanjutnya, data yang masuk ke ESP32 akan dikelompokkan menggunakan metode logika fuzzy mamdani yang kemudian akan ke program penggerak *servo* dari sendok tersebut, dalam proses mamdani data sumbu x, y dan z akan diklompokkan guna mengatur arah putaran dari *servo* sendok itu sendiri, pengelompokkan berguna untuk memperkecil ruang lingkup dari data yang akan diproses oleh program sehingga putaran *servo* yang dibutuhkan akan semakin akurat.



Gambar 1 Diagram Kerangka Kerja Sistem

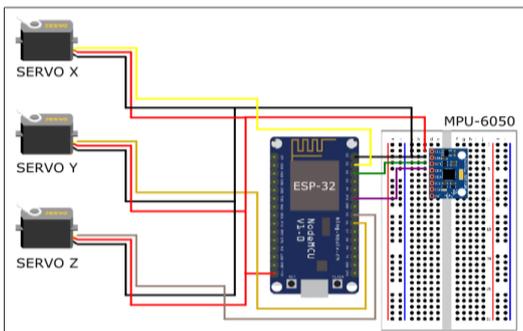


Gambar 2 Diagram Blok Alur Sistem

Jika nilai inputan sudah *difilter*, maka proses selanjutnya nilai akan dikonversikan ke output berupa *servo*, dimana *servo* akan bergerak sesuai poros yang sudah ditentukan agar kepala sendok tetap stabil saat menerima getaran yang berasal dari tangan penderita *Parkinson* saat memegang sendok.

### 2) Perancangan Rangkaian Sistem

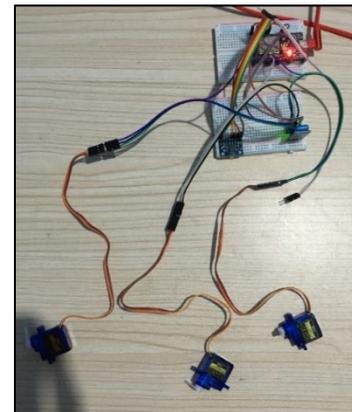
Dikarenakan MPU-6050 memiliki 3 buah sumbu, maka setiap sumbu memiliki outputnya masing-masing yaitu servo SG-90 sehingga terdapat 3 buah servo untuk output dari rangkaian sendok Parkinson. Berikut skematik rangkaian keseluruhan dari sistem sendok Parkinson (ditunjukkan pada Gambar 3) dan tabel pin konektor pada rangkaian sendok parkinson (ditunjukkan pada Tabel 1). Hasil rangkaian di Gambar 4.



Gambar 3 Skematik Rangkaian Keseluruhan Sendok Parkinson

Tabel 1 Pin Konektor Servo SG-90, ESP32, dan MPU-6050 Table Type Styles

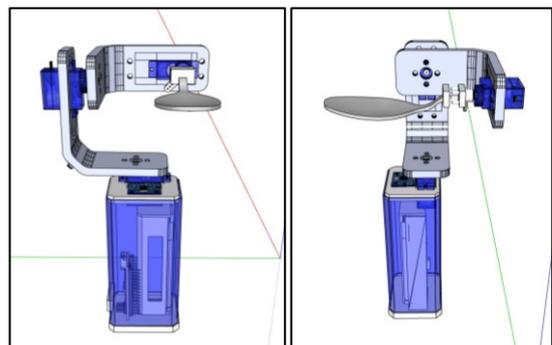
Pin Input dan Output		ESP32
MPU-6050	VCC	5V
	GND	GND
	SCL	P22
	SDA	P21
Servo SG-90	VCC	5V
	GND	GND
	PWM (Servo X)	P23
	PWM (Servo Y)	P18
	PWM (Servo Z)	P19



Gambar 4 Rangkaian Utuh Sendok Parkinson

### 3) Perancangan Cetak Biru Sistem

Setelah cetak biru dari lengan dan gagang telah dibuat, maka proses terakhir adalah menggabungkan kedua cetak biru tersebut menjadi satu cetak biru utuh dari sendok Parkinson. Berikut cetak biru keseluruhan dari sendok Parkinson (ditunjukkan pada Gambar 5).



Gambar 5 Cetak Biru Keseluruhan Sendok Parkinson

### 4) Pembuatan Lengan Sendok

Setelah rangkaian selesai dibuat, dan cetak biru dari sendok Parkinson telah selesai, maka proses selanjutnya adalah pembuatan sendok Parkinson secara keseluruhan yaitu membuat bentuk sendok Parkinson dari hasil cetak biru yang telah dibuat beserta pemasangan rangkaiannya. Pada tahap ini bahan dan alat yang digunakan berupa: kertas, karton padi, cutter, lem, pena, gunting, dan penggaris.

Proses pertama adalah membuat lengan dari sendok *Parkinson*, pada proses ini penulis melakukan beberapa revisi bentuk rancangan, sehingga menghasilkan bentuk akhir sebagai berikut (ditunjukkan pada Gambar 6).



Gambar 6 Lengan Sendok Parkinson (sumbu y kiri dan sumbu z di kanan)

Dari Gambar 6, Terdapat sedikit perubahan bentuk sendok yang asli dan yang cetak biru, karena bertujuan untuk menyesuaikan posisi kabel dari *servo* SG-90.

Proses selanjutnya adalah membuat gagang dari sendok *parknson*, pembuatan gagang disesuaikan dengan ukuran ESP32, kabel dari rangkaian sistem dan cetak biru yang sudah dibuat. Kemudian bentuk direvisi sehingga didapatlah bentuk sebagai berikut (ditunjukkan pada Gambar 7).



Gambar 7 Gagang Sendok Parkinson (Kiri kondisi tertutup, Kanan kondisi terbuka)

Setelah lengan dan gagang telah dibuat dan sesuai yang diharapkan, maka proses terakhir adalah menggabungkan kedua komponen tersebut. Berikut hasil dari penggabungan kedua komponen tersebut (ditunjukkan pada gambar 3.8).



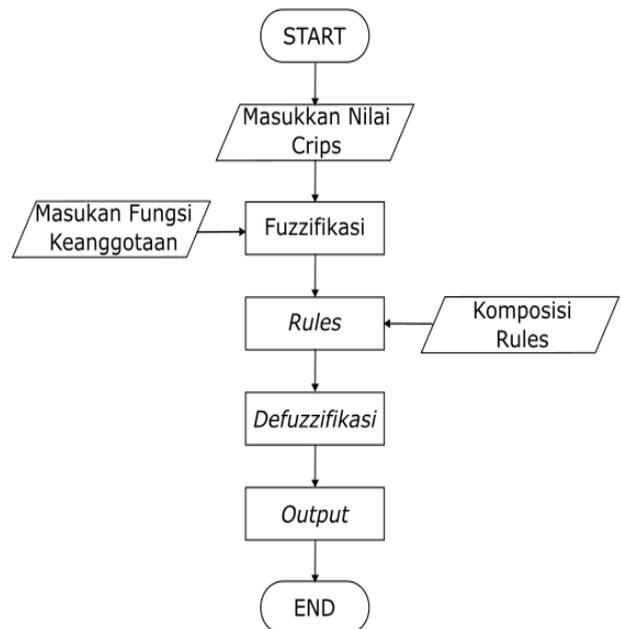
Gambar 8 Prototype Sendok Parkinson.

### 5) Pembuatan Program

Perancangan program sendok *Parkinson* dilakukan beberapa tahap, yaitu perancangan program sensor MPU-6050, perancangan program *servo* SG-90, dan terakhir menggabungkan kedua program tersebut menjadi satu program sendok *Parkinson*.

Pada tahap ini, ESP32 akan deprogram agar mampu membaca inputan dari sensor MPU-6050 dan kemudian inputan mampu difilter menggunakan logika fuzzy sehingga data yang dihasilkan lebih akurat. Berikut program pembaca inputan dari sensor MPU-6050.

Inputan yang diberikan oleh sensor memiliki satuan radian, kemudian satuan tersebut diubah kedalam derajat sesuai pada gambar 3.18, setelah Inputan berhasil dibaca, maka proses selanjutnya adalah memfilter inputan dari sensor MPU-6050 dengan logika fuzzy. Berikut alur program filter dari MPU-6050 (ditunjukkan pada Gambar 9).



Gambar 9 Alur Filter dari Program Sendok Parkinson

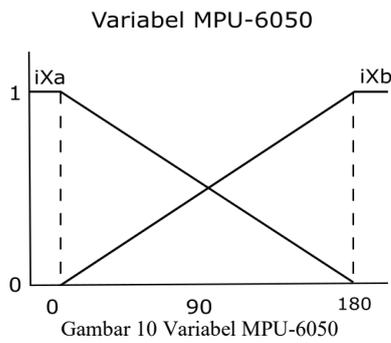
#### a) Fuzzyfikasi

Pada tahap ini, variabel yang digunakan ada 2 yaitu variabel MPU-6050 (ditunjukkan pada Gambar 10) dan variabel Servo (ditunjukkan pada gambar 11), nilai yang diberikan berupa derajat yang kemudian akan dikeluarkan berupa derajat pula. Berikut parameter nilai *input* dan *output*:

- MPU-6050( $iX_a, iX_b$ ) = derajat
- Servo( $oX_a, oX_b$ ) = derajat
- Output(LB, CP) = derajat

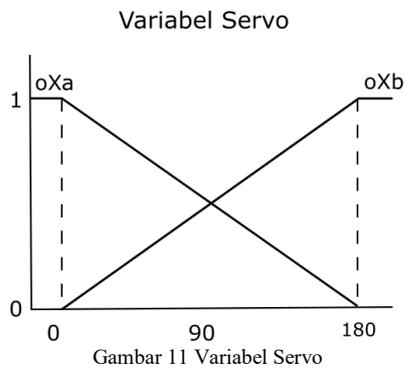
Dan berikut linguistik dari *input* dan *output* :

- $iX_a$  = inputan sumbu X a
- $iX_b$  = inputan sumbu X b
- $oX_a$  = inputan nilai servo sumbu X a
- $oX_b$  = inputan nilai servo sumbu X b
- LB = variabel Lambat
- CP = variabel Cepat



$$\mu_{MPU-6050iXa}[x] = \begin{cases} 1 & , x < 0 \\ \frac{180-x}{180} & , 0 \leq x \leq 180 \\ 0 & x > 180 \end{cases} \quad (7)$$

$$\mu_{MPU-6050iXb}[x] = \begin{cases} 0 & , x < 0 \\ \frac{x}{180} & , 0 \leq x \leq 180 \\ 1 & x > 180 \end{cases} \quad (8)$$

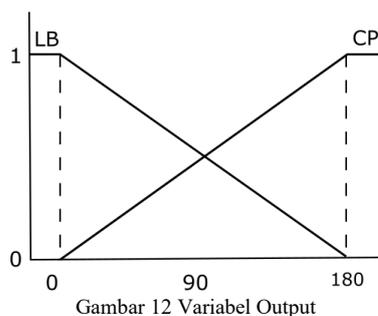


$$\mu_{ServooXa}[y] = \begin{cases} 1 & , y < 0 \\ \frac{180-y}{180} & , 0 \leq y \leq 180 \\ 0 & y > 180 \end{cases} \quad (9)$$

$$\mu_{ServooXb}[y] = \begin{cases} 0 & , y < 0 \\ \frac{y}{180} & , 0 \leq y \leq 180 \\ 1 & y > 180 \end{cases} \quad (10)$$

Variabel data pada Gambar 10, 11, dan 12 memiliki satuan inputan derajat, *range data* dan fungsi keanggotaan direpresentasikan kedalam table data sebagai berikut (ditunjukkan pada Tabel 2, 3, dan 4).

Variabel Output



$$\mu_{OutputLB}[z] = \begin{cases} 1 & , z < 0 \\ \frac{180-z}{180} & , 0 \leq z \leq 180 \\ 0 & z > 180 \end{cases} \quad (11)$$

$$\mu_{OutputCP}[z] = \begin{cases} 0 & , z < 0 \\ \frac{z}{180} & , 0 \leq z \leq 180 \\ 1 & z > 180 \end{cases} \quad (12)$$

Tabel 2 Data Inputan Variabel MPU-6050

MPU-6050	Range Data	Fungsi Keanggotaan
iXa	(0, 180)	Bahu Kiri
iXb	(180, 0)	Bahu Kanan

Tabel 3. Data Outputan Variabel Servo

Servo	Range Data	Fungsi Keanggotaan
oXa	(0, 180)	Bahu Kiri
oXb	(180, 0)	Bahu Kanan

Tabel 4. Data Variabel Output

Servo	Range Data	Fungsi Keanggotaan
LB	(0, 180)	Bahu Kiri
CP	(180, 0)	Bahu Kanan

Kemudian semua data tersebut dimasukkan kedalam program dengan kode program sebagai berikut (ditunjukkan pada Gambar 13 dan 14).

```

if (servo0ValueMap <= 0) // Variabel MPU-6050 iXa
{float output1 = 1;}
else if (servo0ValueMap > 0 && servo0ValueMap < 180){
  ixa = (servo0ValueMap-0) / 180; float output1 = ixa;}
else if (servo0ValueMap >= 180)
{float output1 = 0;}
if (servo0ValueMap <= 0) // Variabel MPU-6050 iXb
{float output2 = 0;}
else if (servo0ValueMap > 0 && servo0ValueMap < 180){
  ixb = (180 - servo0ValueMap) / 180; float output2 = ixb;}
else if (servo0ValueMap >= 180)
{float output2 = 1;}

```

Gambar 13 Kode Program Fuzzifikasi Variabel MPU-6050

```

if (Servo0Value <= 0) // Variabel Servo oXa
{float outputakhirA = 1;}
else if (Servo0Value > 0 && Servo0Value < 180){
  oxa = (Servo0Value-0) / 180; float outputakhirA = oxa;}
else if (Servo0Value >= 180)
{float outputakhirA = 0;}
if (Servo0Value <= 0) // Variabel Servo oXb
{float outputakhirB = 0;}
else if (Servo0Value > 0 && Servo0Value < 180){
  oxb = (180 - Servo0Value) / 180; float outputakhirB = oxb;}
else if (Servo0Value >= 180)
{float outputakhirB = 1;}

```

Gambar 14 Kode Program Fuzzifikasi Variabel Servo

### b) Rules

Setelah melalui tahap fuzzifikasi, maka proses selanjutnya adalah memasukkan nilai *fuzzy* tersebut ke tahap

rules. Pada tahap ini *rules* yang diberikan hanya ada 4 yang terdapat pada tabel sebagai berikut (ditunjukkan pada Tabel 5).

Tabel 5 Data Rules Dari Variabel MPU-6050 dan Variabel Servo

MPU-6050	Servo	Output
iXa	oXa	LB
iXa	oXb	LB
iXb	oXa	LB
iXb	oXb	CP

Berikut sample program *rules* dari program (ditunjukkan pada Gambar 15, 16, dan 17).

Penentuan fungsi keanggotaan hasil komposisi diambil berdasarkan metode MAX yang kemudian metode tersebut yang digunakan saat defuzzifikasi.

```
//RULES
=====
if(Servo0ValueMap <= 0 && Servo0Value <= 0)
{
  if(output1 < outputakhirA){
    lb = output1;
  }
  else if(output1 > outputakhirA){
    lb = outputakhirA;
  }
  else if(output1 == outputakhirA){
    lb = output1;
  }

  hasillb = lb;
  hasilcp = 0;
}
}
```

Gambar 15 Kode Program Rules 1

```
if(Servo0ValueMap <= 0 && Servo0Value > 0 && Servo0Value < 180)
{
  if(output1 < outputakhirA){
    lb1 = output1;
  }
  else if(output1 > outputakhirA){
    lb1 = outputakhirA;
  }
  else if(output1 == outputakhirA){
    lb1 = output1;
  }

  if(output1 < outputakhirB){
    lb2 = output1;
  }
  else if(output1 > outputakhirB){
    lb2 = outputakhirB;
  }
  else if(output1 == outputakhirB){
    lb2 = output1;
  }

  if(lb1 < lb2){
    hasillb = lb1;
  }
  else if(lb1 > lb2){
    hasillb = lb2;
  }
  else if(lb1 == lb2){
    hasillb = lb1;
  }

  hasilcp = 0;
}
=====
```

Gambar 16 Kode Program Rules 2

```
if(Servo0ValueMap <= 0 && Servo0Value >= 180)
{
  if(output1 < outputakhirB){
    cp = output1;
  }
  else if(output1 > outputakhirB){
    cp = outputakhirB;
  }
  else if(output1 == outputakhirB){
    cp = output1;
  }

  hasillb = 0;
  hasilcp = cp;
}
}
```

Gambar 17 Kode Program Rules 3

### c) Defuzzifikasi

Untuk tahap defuzzifikasi, metode yang digunakan adalah Centroid dengan tipe kontinyu, maka kode program defuzzifikasinya adalah sebagai berikut (ditunjukkan pada Gambar 18).

```
//0-10-20-30-40-50-60-70-80-90-100-110-120-130-140-150-160-170-180
int Centroid = 1710;
int CentroidX = 19;

float hasilXA = (Centroid*hasillbX)+(Centroid*hasilcpX);
float hasilXB = (CentroidX*hasillbX)+(CentroidX*hasilcpX);

float hasilakhir0 = hasilXA/hasilXB;
```

Gambar 18 Kode Program Defuzzifikasi

Dapat dilihat pada Gambar 17, pengambilan nilai “Centroid” diambil berdasarkan pengelompokkan nilai dari setiap himpunan pada variabel MPU-6050, Servo, dan Output yang kemudian akan di tambahkan, penggunaan 1 buah Centroid tersebut karena ketiga variabel tersebut memiliki variabel yang sama. Nilai “Centroid” tersebut akan dikalikan dengan hasil fuzzifikasi yang sudah melalui tahap rules dengan nama “hasillb” atau hasil dari himpunan LB, dan “hasilcp” untuk hasil himpunan CP.

Selanjutnya untuk pengambilan nilai “CentroidX” diambil berdasarkan jumlah banyak angka yang ditambahkan pada rumus “Centroid” yang kemudian akan dikali dengan “hasillb” dan “hasilcp”.

Berikut persamaan yang digunakan untuk defuzzifikasi program (13):

$$z^* = \frac{\int_z z\mu(z)dz}{\int_z \mu(z)dz} \quad (13)$$

Nilai hasil defuzzifikasi kemudian akan dikirim ke *servo* dengan satuan derajat yang selanjutnya akan dieksekusi oleh *servo* untuk menstabilkan sendok.

### B. Pengujian

Di tahapan ini dilakukan pengujian sendok *parkinson* secara langsung dengan menggunakan objek tangan tenulis dan beberapa orang lain, pengujian dilakukan dengan cara menggetarkan Sendok *Parkinson* dengan mempelajari

pergerakan *Tremor* yang diderita para penyakit *Parkinson* dan didapatkan bahwa contoh sampel yang telah didapat tidak terjatuh dan dapat bekerja dengan baik. Berikut gambar saat sendok digunakan oleh objek (ditunjukkan pada Gambar 19).



Gambar 19 Percobaan Alat Saat Digunakan

Penulis mengambil beberapa sample foto untuk mengetahui jika sendok digerakkan ke arah tertentu, apakah kepala sendok masih sesuai dengan posisi awal atau tidak, dan berikut hasil yang didapatkan (ditunjukkan pada Gambar 20).



Gambar 20 Gambar 4. 1 Percobaan Alat Saat Digunakan (Kiri miring, Kanan tegak)

Dari hasil pengujian alat secara keseluruhan, pengujian dilakukan sebanyak 12 kali dengan total keberhasilan alat sebanyak 9 kali dan kegagalan alat sebanyak 3 kali. Dimana dari hasil pengujian ini didapatkan presentase *error* dengan persamaan (17).

$$Presentase\ Error = \frac{JP - JK}{JP} \times 100\% \quad (17)$$

dengan:

JP = Jumlah Pengujian  
JK = Jumlah Keberhasilan

Dari persamaan diatas, dapat dihitung presentase *error* dari sendok *Parkinson* ini adalah sebagai berikut.

$$Presentase\ Error = \frac{12 - 9}{12} \times 100\% = 33.3\%$$

Tabel 6 Data Hasil Pengujian Alat Sendok Parkinson

Pengujian Ke –	Hasil Pengujian	Keterangan
1	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
2	Gagal	Terjadi kesalahan arah putaran pada servo y dan z
3	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
4	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
5	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
6	Gagal	Sendok bergerak tak tentu arah saat digunakan
7	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
8	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
9	Gagal	Sendok bergerak tak tentu arah saat digunakan
10	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
11	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna
12	Berhasil	Sendok berhasil menstabilkan gerakan dari pengguna

Berdasarkan pengujian, alat yang dibuat sudah berjalan sesuai yang diharapkan yaitu menstabilkan sendok saat pengguna memberikan gerakan atau getaran ke sendok, dimana saat pengguna memberikan getaran atau gerakan, *servo* akan secara otomatis bergerak untuk menstabilkan sendok, sehingga kepala sendok tidak bergeming saat terjadi getaran.

## V. KESIMPULAN

Dari hasil percobaan yang telah dilakukan, Penggunaan pembuatan sendok *Parkinson* menggunakan sensor MPU-6050 sangat memungkinkan, karena setelah menggunakan metode *fuzzy logic* bisa meminimalisir *error* dari getaran tangan penderita *Parkinson*, dan hasil memungkinkan sesuai dengan harapan yaitu menstabilkan tangan penderita *parkinson*. Hasil pengujian didapatkan bahwa dengan tanpa menggunakan logika *fuzzy margin error* menjadi jauh berbeda sedangkan ketika menggunakan algoritma logika *fuzzy*. Dimana Pada percobaan diatas tanpa menggunakan *filter* pada sudut X menghasilkan rata-rata sudut *error* sebesar 22.68, pada sudut Y sebesar 18.155 dan pada sudut Z sebesar 22.7. Lalu penggunaan logika *fuzzy* pada percobaan diatas mampu mengurangi sudut *error* pada pembacaan sensor mpu-6050, pada sudut X menghasilkan rata-rata sudut *error* sebesar 1.64, Pada sudut Y sebesar 0.178 dan pada sudut Z sebesar 4.28.

## DAFTAR PUSTAKA

- [1] W. Aditya, "Klasifikasi Penyakit Parkinson Menggunakan Artificial Neural Network (Ann) Berdasarkan Ekstraksi Fitur Multifractal Detrended Fluctuation Analysis (Mfdfa) Pada Sinyal Gait," Institut Teknologi Sepuluh Nopember, 2017.
- [2] G. Gunawan, M. Dalhar, and S. N. Kurniawan, "Parkinson and stem cell therapy," *Malang Neurol. J.*, vol. 3, no. 1, pp. 39–46, 2017.
- [3] F. Friendly, "Rancang Bangun Tongkat Tunanetra Menggunakan Sensor Ultrasonik Dengan Gps Tracking Berbasis Mikrokontroler," Universitas Komputer Indonesia, 2019.
- [4] A. Zarkasi, D. D. Mahendra, M. A. Fadilla, and M. N. Halim, "Rancang Bangun Sendok Penderita Parkinson Menggunakan Mikrokontroler ESP-32," in *Annual Research Seminar (ARS)*, 2020, vol. 5, no. 1, pp. 242–246.
- [5] H. Kusumah and R. A. Pradana, "Penerapan Trainer Interfacing Mikrokontroler dan Internet Of Things Berbasis Esp32 pada Mata Kuliah Interfacing," *J. Cerita*, vol. 5, no. 2, pp. 120–134.
- [6] B. Firman, "Implementasi Sensor IMU MPU6050 Berbasis Serial I2C pada Self-Balancing Robot," *J. Teknol. Technoscintia*, pp. 18–24, 2016.
- [7] S. N. Ubay, W. A. Kusuma, and Z. Sari, "Pengembangan Sistem Monitoring Langkah Kaki Dengan Sensor Mpu6050 Untuk Menghitung Jumlah Penurunan Berat Badan Berbasis Android," *Repositor*, vol. 2, no. 2, pp. 122–127, 2020.
- [8] V. R. Alma'i, W. Wahyudi, and I. Setiawan, "Aplikasi Sensor Accelerometer Pada Deteksi Posisi," Jurusan Teknik Elektro Fakultas Teknik, 2011.
- [9] M. Nadhif and others, "Aplikasi Fuzzy Logic Untuk Pengendali Motor DC Berbasis Mikrokontroler ATmega8535 Dengan Sensor Photodiode," Universitas Negeri Semarang, 2015.
- [10] H. Prasetyo and U. Sutisna, "Implementasi Algoritma Logika Fuzzy untuk Sistem Pengaturan Lampu Lalu Lintas Menggunakan Mikrokontroler," *Techno (Jurnal Fak. Tek. Univ. Muhammadiyah Purwokerto)*, vol. 15, no. 2, pp. 1–8, 2014.
- [11] N. Febriany, F. Agustina, and R. Marwati, "Aplikasi metode fuzzy mamdani dalam penentuan status gizi dan kebutuhan kalori harian balita menggunakan software MATLAB," *J. EurekaMatika*, vol. 5, no. 1, pp. 84–96, 2017.
- [12] S. Kusumadewi and H. Purnomo, "Aplikasi Logika Fuzzy untuk pendukung keputusan," *Yogyakarta Graha Ilmu*, pp. 33–34, 2004.
- [13] A. K. MUHAMAD, "Aplikasi Accelerometer pada Penstabil Monopod Menggunakan Motor Servo," Politeknik Negeri Sriwijaya, 2016.