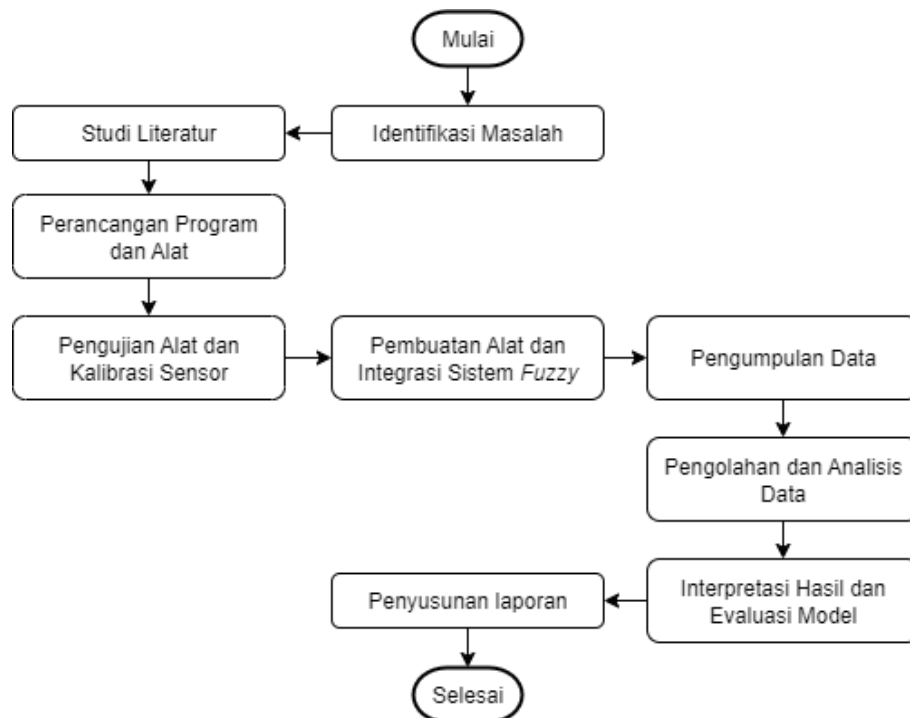


BAB 3

METODE PENELITIAN

3.1 Tahapan Penelitian

Dalam melaksanakan penelitian tersebut, penulis menyusun tahapan-tahapan yang dilakukan untuk mencapai tujuan penelitian. Tahapan penelitian yang disusun dalam penelitian tersebut dapat dilihat dalam bentuk diagram alir pada Gambar 3.1.

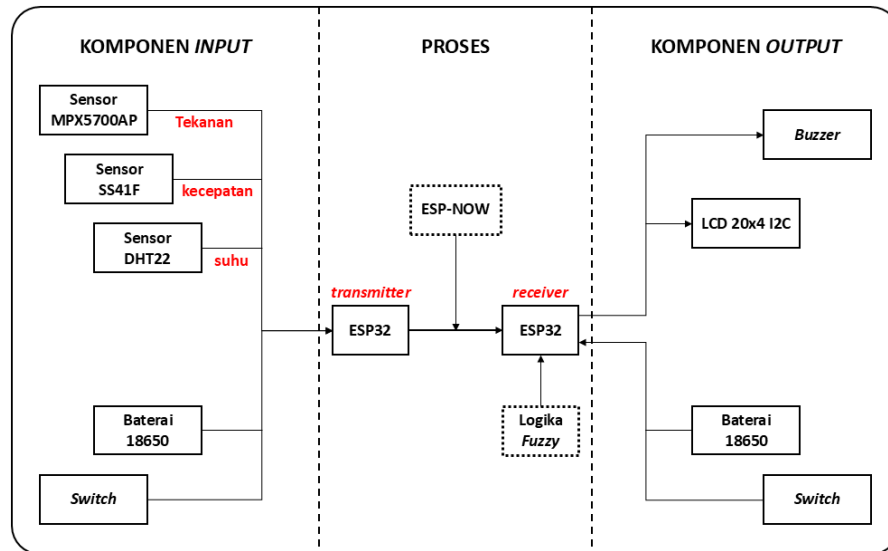


Gambar 3.1 Diagram alir tahapan penelitian

Metode yang digunakan dalam penelitian tersebut adalah metode *research and development* (R&D). Metode *research and development* merupakan pendekatan yang digunakan untuk mengembangkan suatu produk atau teknologi baru, serta menguji efektivitas dan efisiensinya [37]. Metode *research and development* digunakan dalam penelitian tersebut untuk merancang, mengembangkan, menguji, serta mengevaluasi sistem pemantauan kondisi ban berdasarkan parameter tekanan ban, suhu lingkungan, dan kecepatan kendaraan, dengan penerapan logika *fuzzy*, untuk meningkatkan efisiensi pemantauan kondisi ban.

3.2 Model Sistem Penelitian

Model sistem penelitian tersebut berupa perancangan sistem pemantauan kondisi ban kendaraan bermotor berbasis ESP32 dengan penerapan logika *fuzzy*, seperti yang ditunjukkan pada Gambar 3.2.



Gambar 3.2 Desain dan model sistem

Data tekanan ban dari sensor MPX5700AP, suhu dari sensor DHT22, dan kecepatan kendaraan dari sensor SS41F dikumpulkan oleh mikrokontroler ESP32 pada rangkaian *transmitter* yang terpasang di ban kendaraan, lalu mengirimkannya ke rangkaian *receiver* pada dasbor motor. *Receiver* memproses data menggunakan logika *fuzzy* untuk menentukan kondisi ban, menampilkan informasi secara *real-time* pada LCD, dan mengendalikan *buzzer* sebagai peringatan kepada pengemudi apabila ban dalam kondisi tidak baik.

3.3 Desain Alat

3.3.1 Alat dan Bahan Penelitian

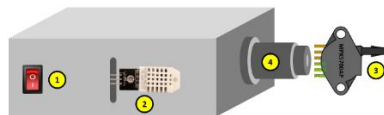
Komponen-komponen yang digunakan untuk membangun sistem pemantauan kondisi ban dalam penelitian tersebut beserta fungsinya dijabarkan sebagai berikut:

- 1) mikrokontroler ESP32 sebagai pengendali utama yang mengatur komunikasi antar-perangkat (sensor dan aktuator), mengumpulkan data dari sensor, memproses data, dan menampilkan data kondisi ban;
- 2) sensor MPX5700AP sebagai pembaca nilai tekanan ban, memberikan informasi mengenai tekanan udara pada ban dalam satuan psi;

- 3) sensor DHT22 sebagai pembaca suhu lingkungan, mengukur suhu sekitar dalam satuan derajat celsius;
- 4) sensor SS41F sebagai pembaca kecepatan kendaraan, mengukur kecepatan rotasi ban melalui perubahan medan magnet yang terdeteksi oleh sensor dan data yang terbaca dapat dihitung per-satuan waktu dan dikonversi ke dalam satuan rpm dan km/jam;
- 5) LCD 20x4 I2C sebagai penampil visual untuk menampilkan informasi mengenai tekanan, suhu, kecepatan, dan kondisi ban secara *real-time* kepada pengemudi;
- 6) *buzzer* sebagai pemberi peringatan suara kepada pengemudi apabila kondisi ban memerlukan perhatian; dan
- 7) baterai Li-Ion 18650 7.4 V sebagai sumber daya utama.

3.3.2 Desain *Transmitter*

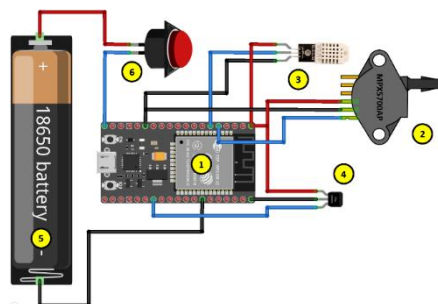
Pada *transmitter*, terdapat tiga komponen utama, yaitu sensor tekanan MPX5700AP, sensor suhu DHT22, dan sensor *hall effect* SS41F. Seluruh data hasil pembacaan sensor tersebut akan diproses mikrokontroler ESP32 untuk dikirimkan ke *receiver*. Selain itu, terdapat komponen pendukung seperti baterai dan *switch*. Baterai yang digunakan adalah Li-Ion 18650 7.4 V sebagai sumber daya utama yang *supply*-nya dikendalikan oleh *switch*.



Gambar 3.3 Ilustrasi perangkat *transmitter*

Keterangan:

- 1) *Switch*
- 2) Sensor DHT22
- 3) Sensor MPX5700AP
- 4) Selang *adaptor*



Gambar 3.4 Skema rangkaian *transmitter*

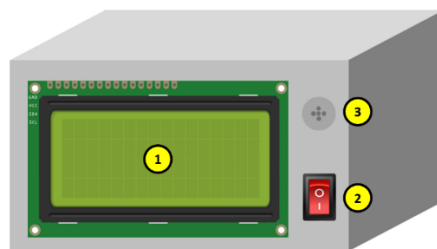
Keterangan:

- 1) ESP32
- 2) Sensor MPX5700AP
- 3) Sensor DHT22
- 4) Sensor SS41F
- 5) Baterai 18650
- 6) *Switch*

Komponen-komponen yang digunakan pada perangkat *transmitter* akan dikemas dalam 3D-*print box* dengan ukuran panjang, lebar, dan tinggi berturut-turut adalah 10 cm, 7.5 cm, dan 3.5 cm. Kemudian, beberapa bagian dilubangi untuk menempatkan sensor DHT22 dan saluran udara (selang *adaptor*) sensor MPX5700AP. Desain fisik perangkat *transmitter* tersebut mempertimbangkan aspek efisiensi ruang, sehingga bentuknya cukup kompak, serta dapat dipasang langsung pada bagian pelek ban sepeda motor tanpa mengganggu rotasinya.

3.3.3 Desain *Receiver*

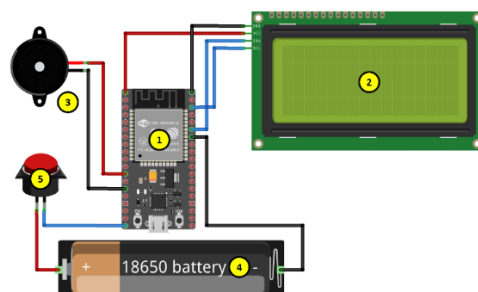
Receiver memiliki dua komponen utama, yaitu LCD 20x4 dengan modul I2C dan *buzzer*. LCD digunakan untuk menampilkan data hasil pembacaan dari seluruh sensor pada perangkat *transmitter*, yang dikirimkan secara nirkabel. Data yang ditampilkan pada LCD adalah tekanan udara ban (dalam satuan psi), suhu (dalam satuan °C), kecepatan rotasi ban (dalam satuan km/jam), dan status kondisi ban (BAIK atau WASPADA atau BURUK). Sementara itu, *buzzer* berperan sebagai indikator suara atau alarm yang akan aktif apabila ban dalam kondisi tidak baik.



Keterangan:

- 1) LCD 20x4 I2C
- 2) *Switch*
- 3) *Buzzer*

Gambar 3.5 Ilustrasi perangkat *receiver*



Keterangan:

- 1) ESP32
- 2) LCD 20x4 I2C
- 3) *Buzzer*
- 4) Baterai 18650
- 5) *Switch*

Gambar 3.6 Skema rangkaian *receiver*

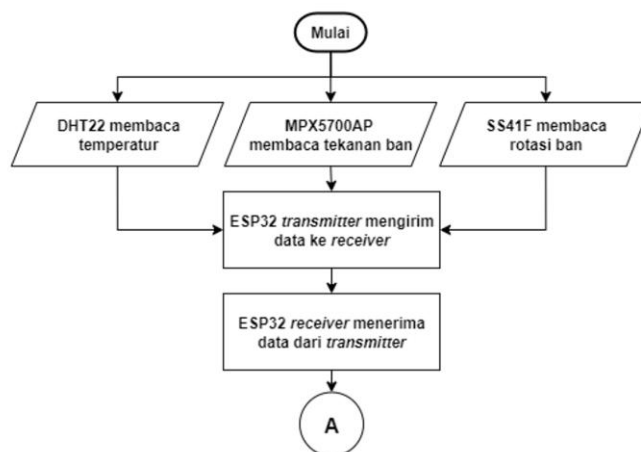
Semua komponen yang digunakan pada perangkat *receiver* akan dirakit dalam *3D-printed box* dengan ukuran panjang, lebar, dan tinggi berturut-turut adalah 14.5 cm, 9.5 cm, dan 5 cm. Kemudian, salah satu sisi dari *box* perangkat akan dilubangi untuk menempatkan komponen *buzzer*. Desain perangkat *receiver* memiliki ukuran yang lebih besar daripada *transmitter* dan mampu untuk mengakomodasi LCD 20x4 yang memiliki ukuran yang cukup besar.

3.4 Desain Program

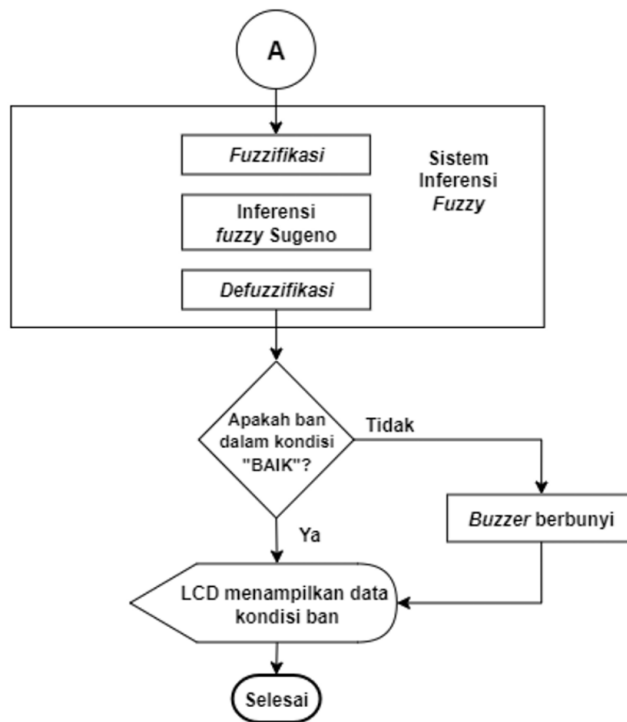
3.4.1 Algoritma Sistem

Desain program digunakan untuk menjelaskan alur dari program Arduino IDE. Program dimulai dengan inialisasi *library* dan *pin* yang digunakan untuk menghubungkan ESP32 dengan sensor MPX5700AP, sensor DHT22, dan sensor SS41F pada rangkaian *transmitter*, serta LCD 20x4 I2C dan *buzzer* pada rangkaian *receiver*. Setelah inialisasi, sistem membaca nilai tekanan, suhu, dan kecepatan kendaraan. Nilai *input* yang terbaca kemudian dikirim oleh *transmitter*.

Receiver menerima data dari *transmitter*, lalu melakukan proses *fuzzifikasi* terhadap nilai-nilai tersebut. Aturan-aturan *fuzzy* digunakan untuk mendapatkan nilai derajat aktivasi (*firing strength*, yaitu *min* atau *max*). Selanjutnya, nilai *output* didapatkan dengan melakukan proses penghitungan menggunakan metode *weighted average*, atau rata-rata terbobot/tertimbang. Diagram alir pemrograman sistem dalam penelitian tersebut dapat dilihat pada Gambar 3.7 dan Gambar 3.8.



Gambar 3.7 Diagram alir pemrograman sistem



Gambar 3.8 Lanjutan diagram alir pemrograman sistem

3.4.2 Parameter *Input*

Penentuan parameter dalam perancangan sistem pemantauan kondisi ban kendaraan bermotor dalam penelitian tersebut diperlukan sebagai pedoman *input* logika *fuzzy*. Detail mengenai parameter *input* logika *fuzzy* disajikan pada Tabel 3.1.

Tabel 3.1 Parameter *input* sistem

No.	Parameter	Nilai
1	Tekanan ban	Tekanan angin normal yang disarankan untuk kendaraan bermotor roda dua adalah pada rentang 22 hingga 36 psi [13].
2	Suhu lingkungan	Rata-rata suhu normal di Indonesia berkisar antara 20 – 30 °C, sedangkan suhu tertinggi yang pernah tercatat di Indonesia adalah 40 °C [38], [39].
3	Kecepatan kendaraan	Batas kecepatan berkendara di Indonesia adalah 30 km/jam untuk kawasan permukiman [40].

Penelitian tersebut menetapkan rentang untuk setiap himpunan *fuzzy* pada masing-masing parameter *input* dengan mengacu pada data yang tersedia pada Tabel 3.1 dengan beberapa penyesuaian untuk nilai rentang setiap himpunan *fuzzy*. Variabel tekanan ban memiliki nilai yang termasuk ke dalam kategori normal pada rentang 22–36 psi. Dengan demikian, ditetapkan bahwa: tekanan 0–16 psi adalah sangat kurang; 18–22 psi adalah kurang; 24–34 psi adalah normal; dan 36–40 psi adalah tinggi. Kemudian, rata-rata suhu normal di Indonesia berkisar antara 20–30 °C, dengan suhu tertinggi adalah 40 °C. Dengan demikian, ditetapkan bahwa: suhu 0–18 °C adalah dingin; 22–30 °C adalah normal; dan 34–40 °C adalah panas. Selanjutnya, untuk parameter kecepatan berkendara, ditetapkan bahwa: 0–16 km/jam adalah lambat; 18–28 km/jam adalah normal; dan 30–120 km/jam adalah cepat, di mana nilai 120 km/jam ditetapkan berdasarkan nilai kecepatan maksimal yang ada di spidometer kebanyakan motor *matic*.

3.4.3 Himpunan Logika *Fuzzy*

Sistem logika *fuzzy* dalam penelitian tersebut terdiri dari tiga variabel *input*, yaitu tekanan, suhu, dan kecepatan. Himpunan *fuzzy* yang ditetapkan dalam penelitian tersebut dapat dilihat pada Tabel 3.2.

Tabel 3.2 Himpunan *fuzzy*

Variabel	Himpunan	Semesta	Domain
Tekanan (psi)	Sangat Kurang	[0, 40]	[0, 18]
	Kurang		[16, 24]
	Normal		[22, 36]
	Tinggi		[34, 40]
Suhu (°C)	Dingin	[0, 40]	[0, 22]
	Normal		[18, 34]
	Panas		[30, 40]
Kecepatan (km/jam)	Lambat	[0, 120]	[0, 18]
	Normal		[16, 30]
	Cepat		[28, 120]

3.4.4 Fuzzifikasi Parameter Tekanan

Domain untuk variabel tekanan dapat dilihat pada Tabel 3.3.

Tabel 3.3 Fuzzifikasi parameter tekanan

Kategori	Domain Himpunan $[x]$
Sangat Kurang	$[0, 0, 16, 18]$
Kurang	$[16, 18, 22, 24]$
Normal	$[22, 24, 34, 36]$
Tinggi	$[34, 36, 40, 40]$

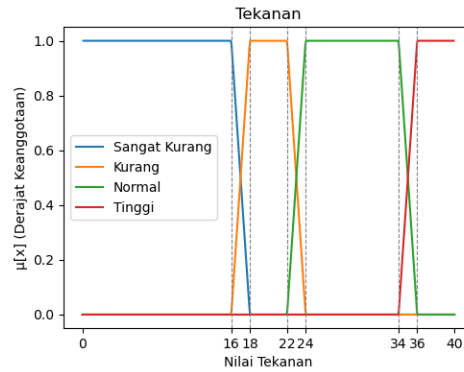
Selanjutnya, fungsi keanggotaan variabel tekanan dijabarkan sebagai berikut:

$$\mu_{Sangat\ Kurang}[x] = \begin{cases} 0; & x < 0 \\ 1; & 0 \leq x \leq 16 \\ \frac{18-x}{18-16}; & 16 < x \leq 18 \\ 0; & x > 18 \end{cases} \quad (3.1)$$

$$\mu_{Kurang}[x] = \begin{cases} 0; & x < 16 \\ \frac{x-16}{18-16}; & 16 \leq x < 18 \\ 1; & 18 \leq x \leq 22 \\ \frac{24-x}{24-22}; & 22 < x \leq 24 \\ 0; & x > 24 \end{cases} \quad (3.2)$$

$$\mu_{Normal}[x] = \begin{cases} 0; & x < 22 \\ \frac{x-22}{24-22}; & 22 \leq x < 24 \\ 1; & 24 \leq x \leq 34 \\ \frac{36-x}{36-34}; & 34 < x \leq 36 \\ 0; & x > 36 \end{cases} \quad (3.3)$$

$$\mu_{Tinggi}[x] = \begin{cases} 0; & x < 34 \\ \frac{x-34}{36-34}; & 34 \leq x < 36 \\ 1; & x \geq 36 \end{cases} \quad (3.4)$$



Gambar 3.9 Grafik himpunan *fuzzy* untuk parameter tekanan

3.4.5 Fuzzifikasi Parameter Suhu

Domain untuk variabel suhu dapat dilihat pada Tabel 3.4.

Tabel 3.4 Fuzzifikasi parameter suhu

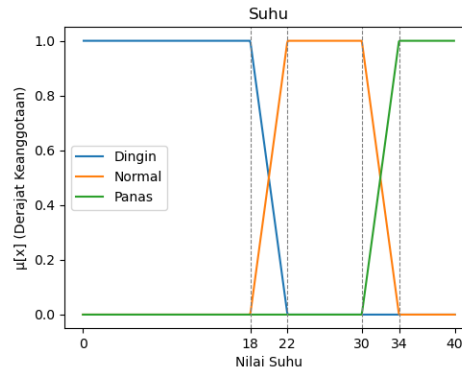
Kategori	Domain Himpunan [x]
Dingin	[0, 0, 18, 22]
Normal	[18, 22, 30, 34]
Panas	[30, 34, 40, 40]

Selanjutnya, fungsi keanggotaan variabel suhu dijabarkan sebagai berikut:

$$\mu_{Dingin}[x] = \begin{cases} 0; & x < 0 \\ 1; & 0 \leq x \leq 18 \\ \frac{22 - x}{22 - 18}; & 18 < x \leq 22 \\ 0; & x > 22 \end{cases} \quad (3.5)$$

$$\mu_{Normal}[x] = \begin{cases} 0; & x < 18 \\ \frac{x - 18}{22 - 18}; & 18 \leq x < 22 \\ 1; & 22 \leq x \leq 30 \\ \frac{34 - x}{34 - 30}; & 30 < x \leq 34 \\ 0; & x > 34 \end{cases} \quad (3.6)$$

$$\mu_{Panas}[x] = \begin{cases} 0; & x < 30 \\ \frac{x - 30}{34 - 30}; & 30 \leq x < 34 \\ 1; & x \geq 34 \end{cases} \quad (3.7)$$



Gambar 3.10 Grafik himpunan *fuzzy* untuk parameter suhu

3.4.6 Fuzzifikasi Parameter Kecepatan

Domain untuk variabel kecepatan dapat dilihat pada Tabel 3.5.

Tabel 3.5 Fuzzifikasi parameter kecepatan

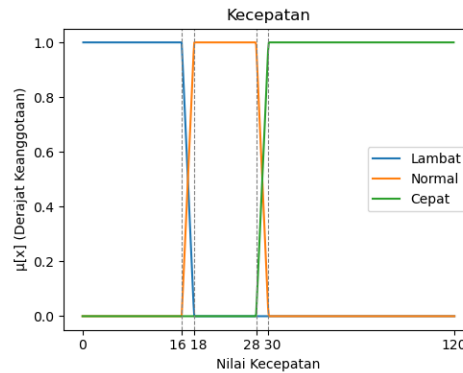
Kategori	Domain Himpunan [x]
Lambat	[0, 0, 16, 18]
Normal	[16, 18, 28, 30]
Cepat	[28, 30, 120, 120]

Selanjutnya, fungsi keanggotaan variabel kecepatan dijabarkan sebagai berikut:

$$\mu_{Lambat}[x] = \begin{cases} 0; & x < 0 \\ 1; & 0 \leq x \leq 16 \\ \frac{18-x}{18-16}; & 16 < x \leq 18 \\ 0; & x > 18 \end{cases} \quad (3.8)$$

$$\mu_{Normal}[x] = \begin{cases} 0; & x < 16 \\ \frac{x-16}{18-16}; & 16 \leq x < 18 \\ 1; & 18 \leq x \leq 28 \\ \frac{30-x}{30-28}; & 28 < x \leq 30 \\ 0; & x > 30 \end{cases} \quad (3.9)$$

$$\mu_{Cepat}[x] = \begin{cases} 0; & x < 28 \\ \frac{x-28}{30-28}; & 28 \leq x < 30 \\ 1; & x \geq 30 \end{cases} \quad (3.10)$$



Gambar 3.11 Grafik himpunan *fuzzy* untuk parameter kecepatan

3.4.7 Inferensi *Fuzzy*

Dalam penelitian ini, sistem pemantauan kondisi ban dirancang untuk memprediksi keadaan ban ke dalam tiga kategori *output*, yaitu BAIK, WASPADA, dan BURUK. Kategori-kategori tersebut ditetapkan berdasarkan prinsip logika *fuzzy* yang memungkinkan pembuat model (*model designer*) memiliki kebebasan dalam mendefinisikan label keluaran (*output labels*), selama penentuan tersebut bersifat logis, konsisten, serta dilakukan secara rasional dan dapat dipertanggungjawabkan [31]. Prinsip tersebut kemudian menjadi dasar untuk basis aturan dalam pemodelan sistem logika *fuzzy* (*fuzzy rule based systems*). Basis aturan *fuzzy* yang digunakan pada sistem pemantauan kondisi ban kendaraan dalam penelitian tersebut dapat dilihat pada Tabel 3.6.

Tabel 3.6 Matriks basis aturan *fuzzy*

Tekanan / Suhu		S U H U			Suhu / Kecepatan	
		Dingin	Normal	Panas		
T E K A N A N	Sangat Kurang	B	B	B	Lambat	K E C E P A T A N
		B	B	B	Normal	
		B	B	B	Cepat	
	Kurang	W	W	W	Lambat	
		W	W	W	Normal	
		W	B	B	Cepat	
	Normal	G	G	G	Lambat	
		G	G	G	Normal	
		G	G	G	Cepat	
	Tinggi	G	W	B	Lambat	
		W	W	B	Normal	
		W	W	B	Cepat	

Pada Tabel 3.6, setiap kolom mewakili kategori linguistik untuk parameter suhu, yaitu: Dingin; Normal; dan Panas, sedangkan baris vertikal merepresentasikan kategori linguistik untuk parameter tekanan, yaitu: Sangat Kurang; Kurang; Normal; dan Tinggi. Setiap baris kategori linguistik tekanan memiliki sub-divisi yang menunjukkan kategori linguistik kecepatan, yaitu: Lambat; Normal; dan Cepat. Isi atau komponen matriks menggambarkan *output* sistem berdasarkan kombinasi kategori linguistik dari parameter *input*, yaitu kondisi ban, dengan kategori linguistik: BAIK (G/*Good*); WASPADA (W); dan BURUK (B/*Bad*). Pembacaan matriks dilakukan dengan: (1) memilih baris berdasarkan kategori tekanan; (2) memilih kolom berdasarkan kategori suhu; dan (3) memilih sub-baris berdasarkan kategori kecepatan.

Dalam konteks penelitian ini, ketiga kategori *output* model dijabarkan sebagai berikut:

1. BAIK

Kondisi ban dikategorikan BAIK apabila tekanan, suhu, dan kecepatan kendaraan berada dalam rentang normal dan aman menurut batas toleransi sistem atau domain yang ditetapkan pada Tabel 3.2. Pada kondisi tersebut, ban tidak memiliki indikasi potensi risiko atau penurunan performa, sehingga pengendara tidak perlu melakukan tindakan korektif segera terhadap ban.

2. WASPADA

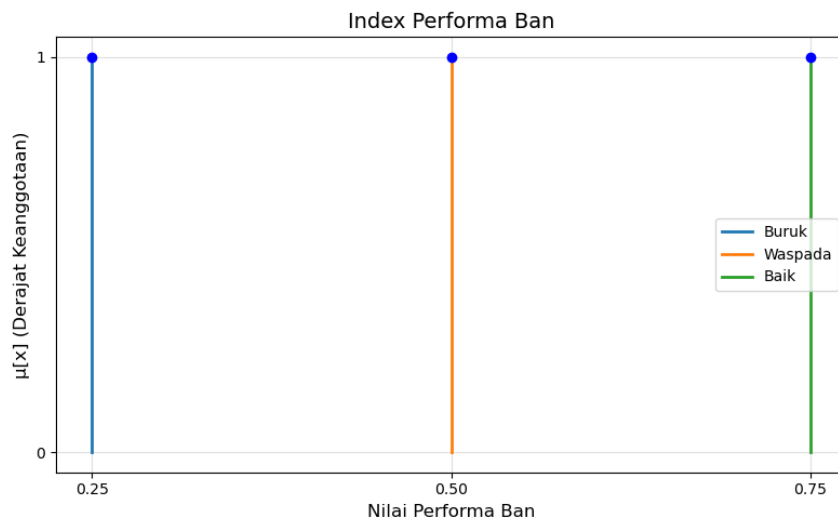
Kondisi ban dikategorikan WASPADA apabila ada salah satu parameter yang mendekati ambang batas toleransi atau mulai menunjukkan penyimpangan dari kondisi ideal, tetapi belum berada pada tingkat yang sangat berbahaya. Misalnya, ketika tekanan ban sedikit kurang atau sedikit lebih tinggi dari rentang normal, suhu ban meningkat dan mulai melewati rentang yang dianggap berbahaya, ataupun kecepatan yang relatif tinggi, yang mana kondisi-kondisi tersebut meningkatkan risiko terhadap ban. Kategori WASPADA bersifat peringatan dini dan menjadi sinyal bagi pengendara untuk melakukan pemeriksaan atau penyesuaian lebih lanjut terhadap ban.

3. BURUK

Kondisi ban dikategorikan BURUK apabila terdapat parameter yang secara signifikan keluar dari batas aman, sehingga dapat menimbulkan risiko keselamatan. Kondisi tersebut, misalnya, adalah ketika tekanan ban sangat rendah, suhu lingkungan yang sangat tinggi, atau kombinasi kondisi yang abnormal lainnya dengan kecepatan tinggi. Pada kondisi tersebut, ban perlu segera dicek, sehingga sistem akan memberikan sinyal peringatan bagi pengendara.

3.4.8 Defuzzifikasi

Dalam sistem inferensi logika *fuzzy* Sugeno yang digunakan pada penelitian tersebut, parameter *output* berupa kondisi ban yang dikategorikan ke dalam tiga variabel linguistik, yaitu Baik, Waspada, dan Buruk, dengan menggunakan fungsi keanggotaan *singleton*, di mana setiap kategori linguistik direpresentasikan oleh nilai *crisp* tunggal (*singleton*), seperti pada Gambar 3.12.



Gambar 3.12 Grafik fungsi keanggotaan untuk parameter *output*

Penelitian tersebut menetapkan nilai *singleton* sebagai berikut: 0.25 untuk kategori BURUK; 0.50 untuk kategori WASPADA; dan 0.75 untuk kategori BAIK. Nilai-nilai tersebut ditetapkan untuk merepresentasikan tingkat performa ban dengan rentang skala 0 hingga 1, di mana nilai yang lebih rendah menunjukkan kondisi ban yang lebih buruk dan nilai yang lebih tinggi menunjukkan kondisi ban yang lebih baik.

Tahap *defuzzifikasi* dalam sistem inferensi *fuzzy* Sugeno pada penelitian tersebut dilakukan untuk menghitung *output crisp* berdasarkan aturan *fuzzy* yang telah ditetapkan. Proses tersebut menggunakan metode rata-rata terbobot (*weighted average*), di mana kontribusi dari setiap aturan *fuzzy* dihitung berdasarkan derajat aktivasi (w_i) dan nilai *output crisp* (z_i) dari aturan tersebut.

3.5 Pembuatan Program dan Alat

Pada tahap tersebut, dilakukan pembuatan program dan perancangan alat yang akan digunakan untuk menjawab rumusan masalah yang telah ditentukan, yaitu sistem pemantauan kondisi ban berbasis ESP32 berdasarkan parameter tekanan, suhu, dan kecepatan, dengan penerapan logika *fuzzy*. Perancangan alat dilakukan dengan membuat skema rangkaian komponen-komponen pada sistem menggunakan *software* Fritzing. Selain itu pemrograman untuk alat tersebut dilakukan dengan menggunakan *software* Arduino IDE dan bahasa pemrograman Python.

3.6 Pengujian Alat

Setelah perancangan perangkat keras dan program dilakukan, tahap berikutnya adalah pengujian alat untuk memastikan bahwa seluruh komponen yang digunakan, terutama sensor-sensor, dapat berfungsi dengan baik dan memberikan hasil pembacaan yang akurat. Pengujian ini bertujuan untuk mengevaluasi performa alat atau perangkat keras, serta sebagai dasar untuk proses kalibrasi sensor agar nilai-nilai yang diperoleh mendekati atau sesuai dengan alat ukur standar yang digunakan sebagai pembanding. Adapun sensor-sensor yang diuji dalam penelitian tersebut mencakup sensor tekanan MPX5700AP, sensor DHT22, sensor SS41F, serta komponen pendukung lainnya, seperti LCD, *buzzer*, dan *switch*. Detail pengujian sensor dijabarkan sebagai berikut:

1. Sensor MPX5700AP

Pengujian sensor MPX5700AP dilakukan dengan membandingkan data hasil pembacaan sensor dengan data TPMS. Pengujian dilakukan di berbagai kondisi, yaitu di ruangan terbuka dan di ban dengan variasi tekanan udara untuk mengetahui respons sensor terhadap perubahan tekanan.

2. Sensor DHT22

Pengujian sensor DHT22 dilakukan dengan membandingkan data hasil pembacaan sensor dengan data termometer. Pengujian dilakukan di beberapa kondisi, seperti di dalam ruangan, di luar ruangan dengan kondisi normal, di luar ruangan dengan terik matahari, dan lain sebagainya.

3. Sensor SS41F

Pengujian sensor SS41F dilakukan dengan membandingkan data hasil pembacaan sensor dalam mendeteksi magnet dengan data takometer. Pengujian dilakukan menggunakan ban motor roda dua yang diputar dengan kecepatan yang berbeda-beda.

Selain itu, pengujian komponen-komponen pendukung lainnya pada perangkat keras dijabarkan sebagai berikut:

1. LCD 20x4 I2C

Pengujian LCD dilakukan dengan pemrograman untuk menampilkan teks menggunakan *library* I2C. *Output* yang diharapkan adalah LCD dapat menampilkan teks yang diprogram pada Arduino IDE.

2. Buzzer

Pengujian *buzzer* dilakukan dengan diberi sinyal logika digital *HIGH* dari mikrokontroler ESP32. *Output* yang diharapkan adalah *buzzer* dapat aktif dan mengeluarkan suara sesuai dengan logika yang diprogram.

3. Switch

Pengujian *switch*/sakelar dilakukan dengan memberikan *input* mekanis berupa penekanan tombol. *Output* yang diharapkan adalah LED menyala ketika *switch* ditekan (menghasilkan tegangan/*HIGH*).

4. ESP32

Terdapat dua ESP32: ESP32 yang berperan sebagai *transmitter* dan ESP32 yang berperan sebagai *receiver*. Pengujian kedua ESP32 tersebut dilakukan dengan memprogram ESP32 *transmitter* untuk mengirimkan data ke ESP32 *receiver*. *Output* yang diharapkan adalah ESP32 *transmitter* dapat mengirimkan data ke ESP32 *receiver* dan ESP32 *receiver* dapat ditampilkan di *serial monitor* atau di LCD.

3.7 Pengumpulan Data

Pengumpulan data dilakukan dalam delapan tahapan dengan skenario yang berbeda untuk memenuhi seluruh parameter *input* dari model logika *fuzzy* yang dirancang. Data yang diambil dari setiap skenario pengujian berjumlah 35, sehingga total seluruh data yang dikumpulkan adalah 280 data. Skenario pengujian 1–4 dapat dilihat pada Tabel 3.7 dan skenario pengujian 5–8 dapat dilihat pada Tabel 3.8. Pengumpulan data skenario 1–4 dilakukan pada kondisi dengan suhu yang lebih rendah, yaitu di bawah atau sama dengan 26 °C. Sementara itu, pengujian untuk skenario 5–8 dilakukan pada kondisi dengan suhu yang lebih tinggi, yaitu di atas 26 °C.

Tabel 3.7 Skenario pengumpulan data (1–4)

Kondisi Pengujian	Skenario Pengujian			
	1	2	3	4
Suhu	$\leq 26\text{ }^{\circ}\text{C}$	$\leq 26\text{ }^{\circ}\text{C}$	$\leq 26\text{ }^{\circ}\text{C}$	$\leq 26\text{ }^{\circ}\text{C}$
Tekanan Ban	37 psi	29 psi	20 psi	12 psi
Kecepatan Kendaraan	10–40 km/jam	10–40 km/jam	10–40 km/jam	10–40 km/jam

Tabel 3.8 Skenario pengumpulan data (5–8)

Kondisi Pengujian	Skenario Pengujian			
	5	6	7	8
Suhu	$> 26^{\circ}\text{C}$	$> 26^{\circ}\text{C}$	$> 26^{\circ}\text{C}$	$> 26^{\circ}\text{C}$
Tekanan Ban	37 psi	29 psi	20 psi	12 psi
Kecepatan Kendaraan	10–40 km/jam	10–40 km/jam	10–40 km/jam	10–40 km/jam

Dalam tahap *fuzzifikasi*, terdapat 36 basis aturan *fuzzy* yang ditetapkan pada model yang dibangun dalam penelitian tersebut, di mana 36 aturan tersebut merupakan kombinasi dari 4 variabel linguistik parameter tekanan, 3 variabel linguistik parameter suhu, dan 3 variabel linguistik parameter kecepatan. Dengan demikian, pengumpulan data yang dilakukan dalam penelitian tersebut akan memenuhi seluruh basis aturan *fuzzy* yang telah ditetapkan.

3.8 Pengolahan dan Analisis Data

Setelah data terkumpul, tahap berikutnya adalah pengolahan dan analisis data. Pengolahan data bertujuan untuk mengorganisasi data yang terkumpul menjadi format yang dapat dipahami untuk diinterpretasikan, sedangkan analisis data bertujuan untuk menentukan tingkat akurasi sistem ataupun algoritma yang digunakan pada alat pemantauan kondisi ban berbasis ESP32 berdasarkan parameter tekanan ban, suhu lingkungan, dan kecepatan kendaraan, dengan penerapan logika *fuzzy*.

3.8.1 Analisis Perangkat Keras

Analisis perangkat keras dilakukan dengan membandingkan data yang diperoleh dari sensor-sensor yang digunakan dalam sistem pemantauan kondisi ban pada penelitian tersebut dengan data yang diperoleh dari alat pembanding yang sudah ditetapkan, yaitu TPMS M9 sebagai pembanding sensor MPX5700AP, termometer untuk sensor DHT22, dan takometer untuk sensor SS41F. Analisis tersebut dilakukan dengan menggunakan perhitungan galat, akurasi, standar deviasi, serta pengujian hipotesis sampel ganda untuk memperkuat validitas dan reliabilitas hasil pengujian alat.

Analisis perangkat keras diawali dengan perhitungan nilai galat absolut dan galat relatif. Galat absolut adalah selisih antara nilai sebenarnya (eksak) dengan nilai perkiraan, yang mana dalam penelitian tersebut nilai eksak adalah nilai dari data pembanding, sedangkan nilai perkiraan adalah nilai dari data sensor. Sementara itu, galat relatif adalah perbandingan antara galat absolut dengan nilai eksak, yang dinyatakan dalam bentuk persentase. Nilai galat tersebut diperlukan untuk mengetahui keakuratan dan presisi sensor dengan menghitung akurasi dan standar deviasi.

Akurasi merupakan ukuran sejauh mana hasil pengukuran dari sensor mendekati nilai sebenarnya yang ditunjukkan oleh alat pembanding. Semakin kecil nilai galat yang dihasilkan, maka semakin tinggi pula akurasi sebuah sensor. Sementara itu, standar deviasi merupakan ukuran penyebaran atau variasi data terhadap nilai rata-ratanya. Standar deviasi yang rendah dapat menjadi indikasi bahwa sensor menghasilkan data yang

stabil, sehingga sensor dapat dikatakan presisi. Persamaan yang digunakan untuk menghitung nilai galat dan akurasi dapat dilihat pada Persamaan 3.11 dan 3.12 [41], [42], sedangkan perhitungan standar deviasi dilakukan dengan menggunakan Persamaan 3.13 [43].

$$Galat = \frac{\text{Selisih nilai pembacaan sensor}}{\text{Nilai alat ukur standar}} \times 100\% \quad (3.11)$$

$$Akurasi = \frac{\text{Nilai pembacaan sensor}}{\text{Nilai alat ukur standar}} \times 100\% \quad (3.12)$$

$$\text{Standar deviasi} = \sqrt{\frac{\sum (x - \bar{x})^2}{(n - 1)}} \quad (3.13)$$

Keterangan:

x : Variabel dari sensor

\bar{x} : Nilai rata-rata keseluruhan variabel sensor

n : Jumlah data

Analisis perangkat keras dilanjutkan dengan uji hipotesis sampel ganda, yaitu pengujian hipotesis yang menggunakan data dari dua sampel yang diperoleh dari dua populasi (data sensor dan data pembanding) untuk mengetahui apakah terdapat perbedaan yang secara statistik cukup berarti (*significant*) di antara parameter-parameter populasi tersebut [44]. Terdapat dua jenis pengujian hipotesis yang dilakukan dalam penelitian ini, yaitu uji hipotesis varians dan uji hipotesis *mean*.

Dalam uji hipotesis varians, varians sampel (s^2) digunakan untuk mengambil kesimpulan mengenai varians populasi (σ^2). Jadi, di dalam uji tersebut, sampel acak dari dua populasi diambil untuk dihitung nilai varians data dari masing-masing sampel dan hasilnya digunakan sebagai dasar untuk membandingkan varians populasi. Pengujian hipotesis varians dilakukan dengan menggunakan uji F. Persamaan-persamaan yang digunakan dalam uji hipotesis varians dapat dilihat pada Persamaan 3.14, 3.15, dan 3.16 [44].

$$\bar{x}_a = \frac{\sum x}{n} \quad (3.14)$$

Keterangan:

\bar{x}_a : Rata-rata dari sampel populasi a
 x : Sampel populasi a
 n : Jumlah sampel

$$s_a^2 = \frac{\sum (x - \bar{x})^2}{n - 1} \quad (3.15)$$

Keterangan:

s_a^2 : Varians sampel populasi a

$$RU_F = \frac{s_a^2}{s_b^2} \quad (3.16)$$

Keterangan:

RU_F : Rasio uji F
 s_a^2 : Varians sampel populasi a
 s_b^2 : Varians sampel populasi b

Tujuan utama uji hipotesis *mean* adalah untuk menentukan apakah terdapat perbedaan yang signifikan di antara data dua populasi. Uji hipotesis *mean* dengan ukuran data lebih dari 30 atau varians-variens populasinya diketahui dilakukan dengan menggunakan uji Z. Persamaan yang digunakan dalam uji hipotesis *mean* dapat dilihat pada Persamaan 3.17 dan 3.18 [44].

$$\sigma_{\bar{x}_a - \bar{x}_b} = \sqrt{\frac{\sigma_a^2}{n_a} + \frac{\sigma_b^2}{n_b}} \quad (3.17)$$

Keterangan:

σ_a^2 : Varians populasi a σ_b^2 : Varians populasi b
 n_a : Jumlah sampel a n_b : Jumlah sampel b

$$RU_Z = z_{test} = \frac{\bar{x}_a - \bar{x}_b}{\sigma_{\bar{x}_a - \bar{x}_b}} \quad (3.18)$$

Keterangan:

RU_Z : Rasio uji Z
 \bar{x}_a : Rata-rata sampel a
 \bar{x}_b : Rata-rata sampel b

3.8.2 Analisis Metrik Evaluasi Model

Analisis algoritma dilakukan untuk mengevaluasi model logika *fuzzy* Sugeno yang dibangun dalam mengidentifikasi kondisi ban untuk menghasilkan sebuah *output* kategorikal (BAIK, WASPADA, atau BURUK) berdasarkan tiga parameter *input*, yaitu tekanan, suhu, dan kecepatan. *Confusion matrix* memberikan gambaran menyeluruh terhadap kinerja model dengan menunjukkan perbandingan antara hasil prediksi sistem dan kondisi aktualnya.

Kerangka *confusion matrix* dalam penelitian ini dapat dilihat pada Tabel 3.9, 3.10, dan 3.11. Tabel 3.9 merupakan kerangka *confusion matrix* untuk kelas atau label BAIK, sementara Tabel 3.10 merupakan kerangka *confusion matrix* untuk kelas WASPADA, dan Tabel 3.11 merupakan kerangka *confusion matrix* untuk kelas BURUK. Dalam penyusunan *confusion matrix*, terdapat empat komponen utama yang menjadi dasar pembacaan dan interpretasi hasil evaluasi model, yaitu:

- 1) *True Positive* (TP), merupakan kondisi di mana ban secara aktual termasuk ke dalam sebuah kategori tertentu (BAIK, WASPADA, atau BURUK) dan model juga berhasil memprediksi kondisi ban dengan tepat sebagai kategori tersebut. Dengan kata lain, *true positive* (TP) merupakan jumlah data yang benar-benar tergolong dalam suatu kelas dan berhasil dikenali oleh sistem sebagai bagian dari kelas tersebut;
- 2) *True Negative* (TN), merupakan kondisi di mana ban tidak termasuk ke dalam kategori tertentu dan model juga tidak mengklasifikasikan data tersebut ke dalam kategori tersebut;
- 3) *False Positive* (FP), merupakan kondisi yang terjadi ketika kondisi aktual ban tidak termasuk ke dalam kategori tertentu, tetapi model justru salah mengklasifikasikan data tersebut ke dalam kategori tersebut;
- 4) *False Negative* (FN), merupakan kondisi di mana ban sebenarnya termasuk ke dalam suatu kategori tertentu, namun model salah mengenali kondisi aktual ban dan malah mengklasifikasikannya ke dalam kategori lain.

Tabel 3.9 *Confusion matrix* untuk kelas BAIK

Confusion Matrix untuk kelas BAIK		Kondisi Ban Aktual		
		BAIK	WASPADA	BURUK
Prediksi Kondisi Ban	BAIK	TP	FP	FP
	WASPADA	FN	TN	TN
	BURUK	FN	TN	TN

Tabel 3.10 *Confusion matrix* untuk kelas WASPADA

Confusion Matrix untuk kelas WASPADA		Kondisi Ban Aktual		
		BAIK	WASPADA	BURUK
Prediksi Kondisi Ban	BAIK	TN	FN	TN
	WASPADA	FP	TP	FP
	BURUK	TN	FN	TN

Tabel 3.11 *Confusion matrix* untuk kelas BURUK

Confusion Matrix untuk kelas BURUK		Kondisi Ban Aktual		
		BAIK	WASPADA	BURUK
Prediksi Kondisi Ban	BAIK	TN	TN	FN
	WASPADA	TN	TN	FN
	BURUK	FP	FP	TP

Setelah menyusun dan menganalisis *confusion matrix*, tahapan selanjutnya dalam mengevaluasi kinerja model logika *fuzzy* Sugeno yang dibangun dalam penelitian tersebut adalah menghitung metrik *accuracy* (akurasi), *precision* (presisi), *recall* (sensitivitas), dan *F1 score* (skor F1). Keempat metrik tersebut bertujuan untuk memberikan gambaran secara kuantitatif mengenai seberapa baik sistem dapat mengenali dan membedakan kondisi-kondisi ban yang termasuk ke dalam kelas atau label BAIK, WASPADA, atau BURUK. Persamaan untuk menghitung nilai metrik *accuracy*, *precision*, *recall*, dan *F1 score* untuk mengevaluasi model dapat dilihat pada Persamaan 3.19, 3.20, 3.21, dan 3.22 [6].

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (3.19)$$

$$Precision = \frac{TP}{TP + FP} \quad (3.20)$$

$$Recall = \frac{TP}{TP + FN} \quad (3.21)$$

$$F1\ Score = \frac{2 (Recall \times Precision)}{Recall + Precision} \quad (3.22)$$

Accuracy (akurasi) mengukur proporsi keseluruhan prediksi yang benar terhadap seluruh jumlah data yang diuji. Metrik tersebut memberikan gambaran umum mengenai tingkat keberhasilan model secara keseluruhan tanpa mempertimbangkan distribusi setiap kelas. *Precision* (presisi) menunjukkan proporsi prediksi yang benar untuk suatu kelas tertentu dibandingkan dengan seluruh prediksi yang mengarah pada kelas tersebut, sedangkan *recall* (sensitivitas) mengukur kemampuan model dalam mengenali semua data yang sebenarnya termasuk ke dalam kelas tertentu, yaitu rasio antara jumlah data yang berhasil dikenali sebagai kelas tersebut terhadap jumlah aktual seluruh data yang termasuk ke dalam kelas tersebut. Sementara itu, *F1 score* merupakan rata-rata harmonik dari *precision* dan *recall* yang digunakan untuk memberikan keseimbangan di antara kedua metrik tersebut, terutama pada kondisi di mana terdapat ketidakseimbangan data antar-kelas.

Dalam evaluasi performa model klasifikasi yang memiliki lebih dari dua kelas atau sistem klasifikasi multi-kelas (*multiclass classification*), seperti pada model logika *fuzzy* Sugeno dalam penelitian ini yang mengategorikan kondisi ban ke dalam kelas atau label BAIK, WASPADA, atau BURUK, perhitungan *precision*, *recall*, dan *F1 score* untuk setiap kelas tidak memberikan gambaran secara menyeluruh. Oleh karena itu, digunakan pendekatan mode rerata untuk mendapatkan ukuran agregat kinerja model secara lebih komprehensif. Terdapat tiga jenis mode rerata, yaitu *macro*

average (mode rerata makro), *micro average* (mode rerata mikro), dan *weighted average* (mode rerata terbobot/tertimbang) [45].

Macro average menghitung rata-rata sederhana dari metrik (*precision*, *recall*, *F1 score*) untuk setiap kelas tanpa mempertimbangkan jumlah sampel per kelas. Artinya, setiap kelas diberikan bobot yang sama besar, tanpa memandang apakah kelas tersebut memiliki sedikit atau banyak data. Sementara itu, *micro average* menghitung total TP, FP, dan FN dari semua kelas terlebih dahulu, kemudian menggunakan total tersebut untuk menghitung *precision*, *recall*, dan *F1 score*. *Micro average* lebih mencerminkan performa model terhadap keseluruhan sampel, sedangkan *macro average* menekankan perlakuan yang sama terhadap setiap kelas, sehingga kelas minoritas (*minor class*) memiliki kontribusi yang setara dengan kelas mayoritas, terlepas dari frekuensinya [45], [46].

Sementara itu, *weighted average* menghitung rata-rata metrik per kelas dengan memberikan bobot proporsional (*support*) berdasarkan jumlah sampel aktual di masing-masing kelas. Dalam metode tersebut, *precision*, *recall*, dan *F1 score* untuk setiap kelas dihitung terlebih dahulu, lalu dikalikan dengan *support* dan dirata-ratakan. Dengan kata lain, kelas dengan jumlah data yang lebih banyak akan memberikan kontribusi lebih besar terhadap nilai akhir. Persamaan yang digunakan untuk menghitung *macro*, *micro*, dan *weighted average* dapat dilihat pada Persamaan 3.23 hingga Persamaan 3.31 [6].

1. Persamaan *Macro Average*

$$Macro_{Precision} = \frac{\sum Precision}{n} \quad (3.23)$$

$$Macro_{Recall} = \frac{\sum Recall}{n} \quad (3.24)$$

$$Macro_{F1\ Score} = \frac{2(Ma_p \times Ma_R)}{Ma_p + Ma_R} \quad (3.25)$$

Keterangan:

Ma_p : *Macro precision* n : Jumlah kelas
 Ma_R : *Macro recall*

2. Persamaan *Micro Average*

$$Micro_{Precision} = \frac{\sum TP}{\sum TP + \sum FP} \quad (3.26)$$

$$Micro_{Recall} = \frac{\sum TP}{\sum TP + \sum FN} \quad (3.27)$$

$$Micro_{F1\ Score} = \frac{2(Mi_P \times Mi_R)}{Mi_P + Mi_R} \quad (3.28)$$

Keterangan:

TP : *True positive*

Mi_P : *Micro precision*

FP : *False positive*

Mi_R : *Micro recall*

FN : *False negative*

3. Persamaan *Weighted Average*

$$Weighted_{Precision} = \frac{\sum(S \times P)}{\sum S} \quad (3.29)$$

$$Weighted_{Recall} = \frac{\sum(S \times R)}{\sum S} \quad (3.30)$$

$$Weighted_{F1\ Score} = \frac{2(W_P \times W_R)}{W_P + W_R} \quad (3.31)$$

Keterangan:

S : *Support*

W_P : *Weighted precision*

P : *Precision*

W_R : *Weighted recall*

R : *Recall*

3.9 Ilustrasi Implementasi Alat

Sistem pemantauan kondisi ban yang dibangun dalam penelitian tersebut terdiri dari dua perangkat, yaitu *transmitter* dan *receiver*. *Transmitter* merupakan perangkat yang dipasang pada pelek ban dan memiliki sensor-sensor yang berfungsi untuk membaca data kondisi ban dan mengirimkan data hasil pembacaannya ke perangkat *receiver*. Perangkat *transmitter* memiliki ukuran panjang, lebar, dan tinggi berturut-turut 10 cm, 7.5 cm, dan 3.5 cm, sedangkan *receiver* memiliki ukuran panjang, lebar, dan tinggi berturut-turut 14.5 cm, 9.5 cm, dan 5 cm. Ilustrasi implementasi *transmitter* pada pelek ban motor dan *receiver* pada dasbor pengemudi dapat dilihat pada Gambar 3.13.



Gambar 3.13 Ilustrasi implementasi transmitter dan receiver pada kendaraan

Adapun cara kerja sistem pemantauan kondisi ban berbasis ESP32 dengan penerapan logika *fuzzy* Sugeno yang dibangun pada penelitian tersebut dijabarkan sebagai berikut:

A. *Input*

1. Sensor MPX5700AP mendapat data tekanan ban dalam satuan psi;
2. Sensor DHT22 mendapat data suhu lingkungan dalam satuan °C;
3. Sensor SS41F mendapat data rotasi ban dalam satuan rpm, yang kemudian dikonversi ke dalam satuan km/jam; dan
4. ESP32 yang berfungsi sebagai *transmitter* menerima data dari sensor MPX5700AP, sensor DHT22, dan sensor SS41F.

B. *Proses*

1. ESP32 yang berfungsi sebagai *transmitter* mengirimkan data-data mengenai kondisi ban kepada *receiver*;
2. ESP32 yang berfungsi sebagai *receiver* menerima data-data mengenai kondisi ban yang dikirim oleh ESP32 *transmitter*; dan
3. seluruh proses pertukaran data antara ESP32 *transmitter* dan *receiver* dilakukan dengan menggunakan protokol komunikasi ESP-NOW.

C. *Output*

1. ESP32 yang berfungsi sebagai *receiver* kemudian mengolah data kondisi ban yang diterima dengan menerapkan model logika *fuzzy* Sugeno untuk menentukan kondisi ban;
2. data kondisi ban ditampilkan pada LCD 20x4 I2C; dan
3. *buzzer* akan menyala untuk mengirimkan peringatan apabila kondisi ban tidak baik.

3.10 Jadwal Penelitian

Penyusunan jadwal penelitian bertujuan untuk meningkatkan efisiensi proses penelitian, sehingga pelaksanaannya dapat dilakukan dengan optimal. Jadwal penelitian yang direncanakan penulis dapat dilihat pada Tabel 3.12.

Tabel 3.12 Rencana jadwal penelitian

No.	Kegiatan	Bulan ke-					
		1	2	3	4	5	6
1.	Identifikasi masalah						
2.	Studi literatur						
3.	Pembuatan program dan perancangan alat						
4.	Pembuatan alat						
5.	Pengujian alat dan pengumpulan data						
6.	Pengolahan dan analisis data						
7.	Interpretasi hasil dan evaluasi						
8.	Penyusunan laporan						