

## BAB IV

### PENGUJIAN DAN ANALISIS

Pengujian berguna untuk mengukur tingkat keberhasilan dari system atau alat yang dibuat. Sehingga hasil yang diharapkan dapat tercapai dengan baik. Pengujian dilakukan dengan beberapa tahap, mulai dari pengukuran *modular* antar modul komponen hingga uji keseluruhan sistem pada alat.

#### 4.1 Pengujian

##### 4.1.1 Pengujian Sensor MPU6050 GY512

Pengujian MPU6050 GY512 dilakukan dengan cara mengukur tingkat keberhasilan sensor dalam mengirimkan hasil pengukuran percepatan/akselerasi sensor dan kecepatan sudut dari sensor MPU6050 GY512 ke mikrokontroller ESP32 menggunakan komunikasi serial I2C.

Pada pengujian ini dilakukan dengan situasi dan kondisi seperti berikut :

- Tempat : *Indoor*
- Waktu : Siang
- Alat : Arduino Terminal
- Metode uji : Debug Serial
- Hasil : Terlampir pada tabel IV.1

Pada tabel IV.1 merupakan sampel data dari pengujian pembacaan dari sensor MPU6050 GY512. Data pembacaan terdiri dari Accelerometer 3 Axis (X, Y, Z) dan Gyroscopemeter 3 Axis (X, Y, Z).

Tabel IV.1 Sampel data pengujian sensor MPU6050

Accelerometer			Gyroscopemeter			Hasil Pembacaan
Ax	Ay	Az	Gx	Gy	Gz	
2288	-4368	13536	-221	-33	116	Terbaca normal
2344	-4460	13564	-221	-11	110	Terbaca normal

2408	-4488	13520	-239	0	138	Terbaca normal
2444	-4324	13404	-231	-10	101	Terbaca normal
2324	-4488	13572	-217	-12	106	Terbaca normal
2300	-4484	13564	-221	-29	107	Terbaca normal
2396	-4400	13404	-218	-33	107	Terbaca normal
2412	-4492	13556	-212	-32	120	Terbaca normal
2364	-4416	13524	-202	14	121	Terbaca normal
2368	-4368	13620	-233	-20	88	Terbaca normal

Data dari tabel IV.1 menunjukkan tingkat keberhasilan pembacaan data asli dari sensor yang selanjutnya akan diolah untuk mendapatkan percepatan (g) dan kemiringan (°).

Keterangan data dari tabel VI.1 meliputi:

- Ax nilai untuk Akselerometer axis x
- Ay nilai untuk Akselerometer axis y
- Az nilai untuk Akselerometer axis z
- Gx nilai untuk Gyroscopemeter axis x
- Gy nilai untuk Gyroscopemeter axis y
- Gz nilai untuk Gyroscopemeter axis z

Berdasarkan tabel perbandingan IV.1 menunjukkan bahwa tingkat keberhasilan sistem dalam membaca data dari sensor MPU6050 GY512 adalah:

$$\begin{aligned}
 \text{Keberhasilan} &= \frac{\text{Terbaca normal}}{\text{banyaknya sample percobaan}} \times 100\% = \frac{10}{10} \times 100\% \\
 &= 100\%
 \end{aligned}$$

Pada gambar IV.1 merupakan pembacaan data asli dari sensor MPU6050 dengan menggunakan serial monitor Arduino IDE.

```

COM5
23:13:14.797 -> AcX = 2288 | AcY = -4368 | AcZ = 13536 | Tmp = 29.05 | GyX = -221 | GyY = -33 | GyZ = 116
23:13:15.098 -> AcX = 2344 | AcY = -4460 | AcZ = 13564 | Tmp = 29.05 | GyX = -221 | GyY = -11 | GyZ = 110
23:13:15.445 -> AcX = 2408 | AcY = -4488 | AcZ = 13520 | Tmp = 29.14 | GyX = -239 | GyY = 0 | GyZ = 138
23:13:15.800 -> AcX = 2444 | AcY = -4324 | AcZ = 13404 | Tmp = 29.05 | GyX = -231 | GyY = -10 | GyZ = 101
23:13:16.101 -> AcX = 2324 | AcY = -4488 | AcZ = 13572 | Tmp = 29.14 | GyX = -217 | GyY = -12 | GyZ = 106
23:13:16.449 -> AcX = 2300 | AcY = -4484 | AcZ = 13564 | Tmp = 29.14 | GyX = -221 | GyY = -29 | GyZ = 107
23:13:16.804 -> AcX = 2396 | AcY = -4400 | AcZ = 13404 | Tmp = 29.05 | GyX = -218 | GyY = -33 | GyZ = 107
23:13:17.104 -> AcX = 2412 | AcY = -4492 | AcZ = 13556 | Tmp = 29.09 | GyX = -212 | GyY = -32 | GyZ = 120
23:13:17.453 -> AcX = 2364 | AcY = -4416 | AcZ = 13524 | Tmp = 29.09 | GyX = -202 | GyY = 14 | GyZ = 121
23:13:17.807 -> AcX = 2368 | AcY = -4468 | AcZ = 13620 | Tmp = 29.09 | GyX = -233 | GyY = -20 | GyZ = 88
23:13:18.108 -> AcX = 2228 | AcY = -4560 | AcZ = 13604 | Tmp = 29.00 | GyX = -216 | GyY = -20 | GyZ = 135
23:13:18.456 -> AcX = 2252 | AcY = -4424 | AcZ = 13448 | Tmp = 29.05 | GyX = -206 | GyY = -17 | GyZ = 97
23:13:18.810 -> AcX = 2312 | AcY = -4500 | AcZ = 13540 | Tmp = 29.05 | GyX = -213 | GyY = -10 | GyZ = 103
23:13:19.111 -> AcX = 2268 | AcY = -4424 | AcZ = 13608 | Tmp = 29.14 | GyX = -216 | GyY = -15 | GyZ = 117
23:13:19.458 -> AcX = 2296 | AcY = -4428 | AcZ = 13492 | Tmp = 29.14 | GyX = -223 | GyY = -27 | GyZ = 106
23:13:19.812 -> AcX = 2320 | AcY = -4476 | AcZ = 13552 | Tmp = 29.05 | GyX = -204 | GyY = -19 | GyZ = 122
23:13:20.159 -> AcX = 2332 | AcY = -4440 | AcZ = 13604 | Tmp = 29.09 | GyX = -213 | GyY = -11 | GyZ = 96
23:13:20.460 -> AcX = 2332 | AcY = -4536 | AcZ = 13584 | Tmp = 29.00 | GyX = -232 | GyY = -3 | GyZ = 113
23:13:20.815 -> AcX = 2300 | AcY = -4424 | AcZ = 13564 | Tmp = 29.09 | GyX = -223 | GyY = -10 | GyZ = 112
23:13:21.163 -> AcX = 2224 | AcY = -4480 | AcZ = 13540 | Tmp = 29.09 | GyX = -226 | GyY = -3 | GyZ = 139
23:13:21.464 -> AcX = 2324 | AcY = -4464 | AcZ = 13588 | Tmp = 29.14 | GyX = -215 | GyY = -36 | GyZ = 116
23:13:21.819 -> AcX = 2300 | AcY = -4520 | AcZ = 13568 | Tmp = 29.19 | GyX = -225 | GyY = -15 | GyZ = 135
23:13:22.167 -> AcX = 2276 | AcY = -4380 | AcZ = 13504 | Tmp = 29.00 | GyX = -216 | GyY = -14 | GyZ = 117
23:13:22.468 -> AcX = 2260 | AcY = -4448 | AcZ = 13632 | Tmp = 29.05 | GyX = -237 | GyY = -6 | GyZ = 118
23:13:22.822 -> AcX = 2284 | AcY = -4476 | AcZ = 13448 | Tmp = 28.95 | GyX = -225 | GyY = -25 | GyZ = 115
23:13:23.169 -> AcX = 2352 | AcY = -4520 | AcZ = 13584 | Tmp = 29.05 | GyX = -233 | GyY = -6 | GyZ = 106
23:13:23.470 -> AcX = 2352 | AcY = -4480 | AcZ = 13452 | Tmp = 29.00 | GyX = -212 | GyY = -21 | GyZ = 121
23:13:23.827 -> AcX = 2268 | AcY = -4472 | AcZ = 13480 | Tmp = 29.09 | GyX = -224 | GyY = -11 | GyZ = 104
23:13:24.174 -> AcX = 2292 | AcY = -4492 | AcZ = 13452 | Tmp = 29.09 | GyX = -218 | GyY = -6 | GyZ = 121
23:13:24.476 -> AcX = 2340 | AcY = -4508 | AcZ = 13516 | Tmp = 29.09 | GyX = -219 | GyY = -11 | GyZ = 122
23:13:24.830 -> AcX = 2308 | AcY = -4416 | AcZ = 13468 | Tmp = 29.14 | GyX = -236 | GyY = -12 | GyZ = 100

```

Gambar IV.1 Pengujian pembacaan dari sensor MPU6050 GY512 dengan *Serial Monitor* Arduino IDE.

#### 4.1.2 Pengujian *G-Force*

Pengujian *g-force* dilakukan dengan cara mengukur tingkat keberhasilan pengolahan data akselerasi/percepatan *accelerometer* menjadi *g-force* untuk mendeteksi tumbukan pada saat kecelakaan. Data hasil pengujian *g-force* dapat dilihat pada tabel IV.2.

Pada pengujian ini dilakukan dengan situasi dan kondisi seperti berikut :

- Tempat : *Indoor* dan *Outdoor*
- Waktu : Siang
- Alat : Arduino Terminal
- Metode uji : Debug Serial
- Hasil : Terlampir pada tabel IV.2

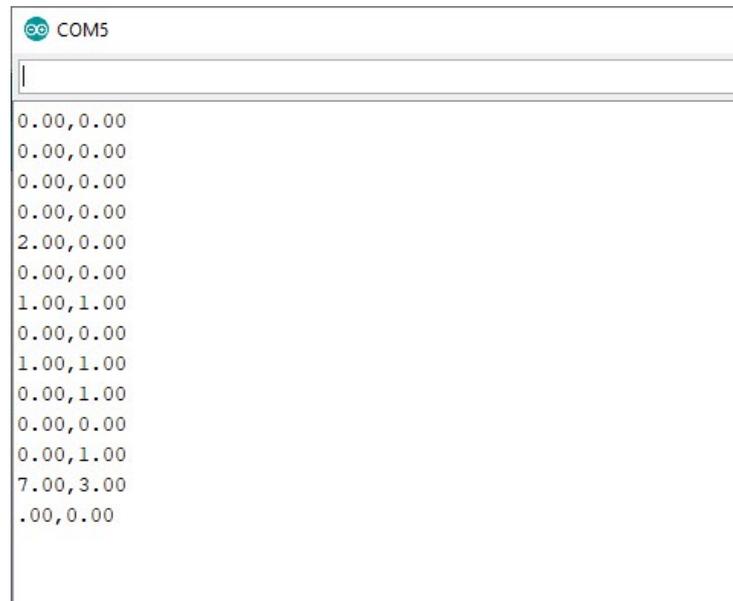
Tabel IV.2 Data hasil uji *g-force*.

No.	GforxeX	GforceY	Kesesuaian
1	0	0	Sesuai
2	0	0	Sesuai
3	0	0	Sesuai
4	0	0	Sesuai
5	2	0	Sesuai
6	0	0	Sesuai
7	1	1	Sesuai
8	0	0	Sesuai
9	1	1	Sesuai
10	0	1	Sesuai
11	0	0	Sesuai
12	0	1	Sesuai
13	7	3	Tidak Sesuai

Berdasarkan tabel perbandingan IV.2 menunjukkan bahwa tingkat keberhasilan sistem dalam pengolahan data menjadi *g-force* adalah:

$$\text{Keberhasilan} = \frac{\text{kesesuaian}}{\text{banyaknya percobaan}} \times 100\% = \frac{12}{13} \times 100\% = 92,3\%$$

Pada gambar IV.2 merupakan tampilan berupa hasil pembacaan *g-force* dengan menggunakan *Serial Monitor* Arduino.



Gambar IV.2 Pembacaan data g-force menggunakan *Serial Monitor* Arduino.

#### 4.1.3 Pengujian Kemiringan

Pengujian kemiringan dilakukan dengan cara mengukur tingkat keberhasilan pengolahan data kecepatan sudut *gyroscope* menjadi kemiringan untuk mendeteksi posisi pengguna alat pada saat kecelakaan seperti ditunjukkan pada tabel IV.3.

Pada pengujian ini dilakukan dengan situasi dan kondisi seperti berikut :

- Tempat : *Indoor* dan *Outdoor*.
- Waktu : Siang
- Alat : Arduino Terminal
- Metode uji : Debug Serial
- Hasil : Terlampir pada tabel IV.3

Tabel IV.3 Data hasil pengujian kemiringan.

No	Nilai (°)	Posisi Diharapkan	Kesesuaian

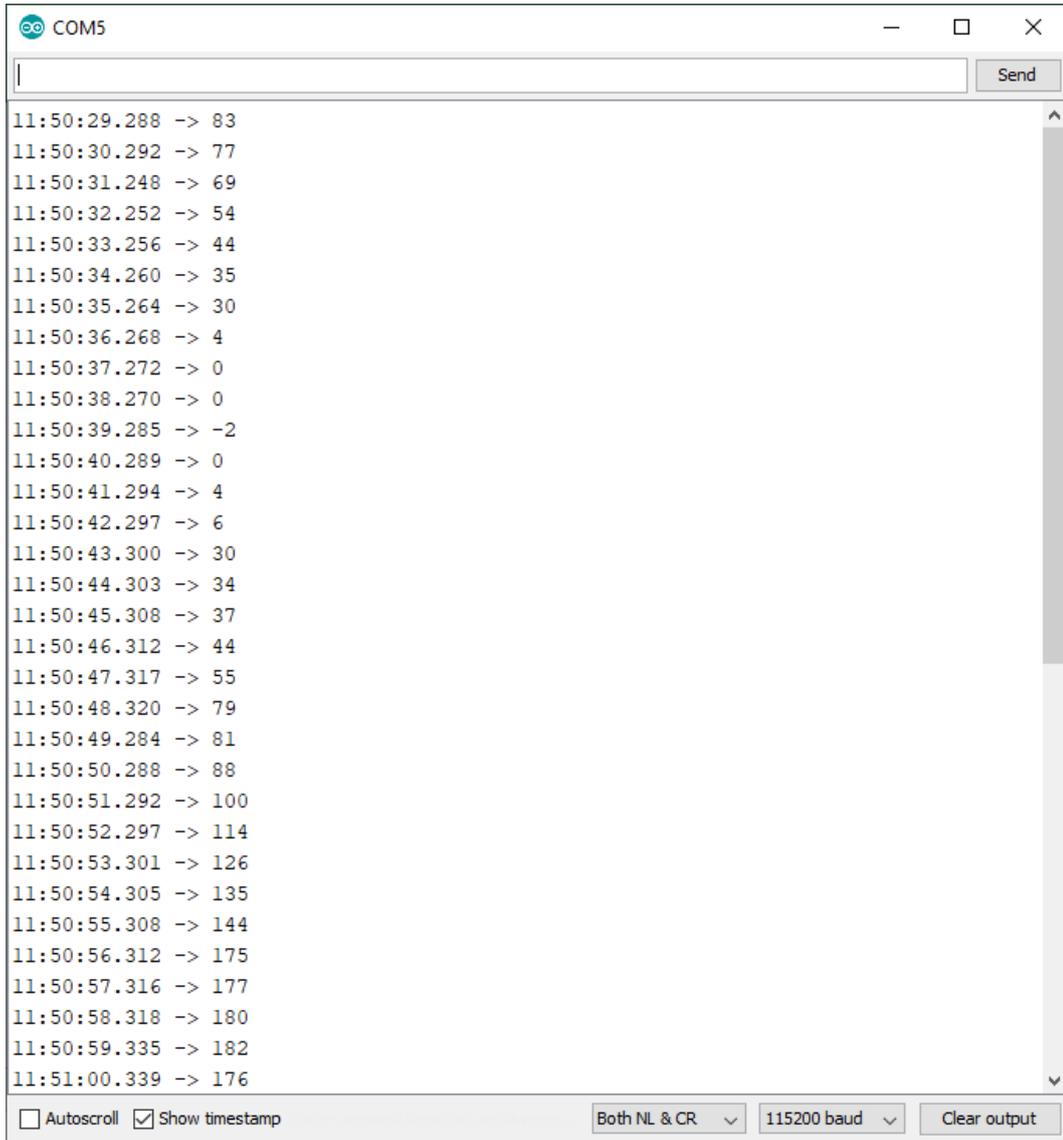
1	83	Tegak	sesuai
2	77	Tegak	sesuai
3	69	Tegak	sesuai
4	54	Sedikit Miring Kiri	sesuai
5	44	Sedikit Miring Kiri	sesuai
6	35	Sedikit Miring Kiri	sesuai
7	30	Sedikit Miring Kiri	sesuai
8	4	Miring Kiri	sesuai
9	0	Miring Kiri	sesuai
10	0	Miring Kiri	sesuai
11	-2	Miring Kiri	tidak sesuai
12	0	Miring Kiri	sesuai
13	4	Miring Kiri	sesuai
14	6	Miring Kiri	sesuai
15	30	Sedikit Miring Kiri	sesuai
16	34	Sedikit Miring Kiri	sesuai
17	37	Sedikit Miring Kiri	sesuai
18	44	Sedikit Miring Kiri	sesuai

19	55	Sedikit Miring Kiri	sesuai
20	79	Tegak	sesuai
21	81	Tegak	sesuai
22	88	Tegak	sesuai
23	100	Tegak	sesuai
24	114	Tegak	sesuai
25	126	Sedikit Miring Kanan	sesuai
26	135	Sedikit Miring Kanan	sesuai
27	144	Sedikit Miring Kanan	sesuai
28	175	Miring Kanan	sesuai
29	177	Miring Kanan	sesuai
30	180	Miring Kanan	sesuai
31	182	Miring Kanan	tidak sesuai
32	176	Miring Kanan	sesuai

Berdasarkan pada tabel IV.3 menunjukkan tingkat keberhasilan sistem dalam memproses kemiringan adalah :

$$Keberhasilan = \frac{\text{kesesuaian}}{\text{banyaknya percobaan}} \times 100\% = \frac{30}{32} \times 100\% = 93,75\%$$

Pada gambar IV.3 menunjukkan hasil data yang diperoleh saat pengujian dengan menggunakan *serial monitor* Arduino IDE (*Arduino Terminal*)



The screenshot shows the Arduino IDE Serial Monitor window titled 'COM5'. The window contains a list of data points, each consisting of a timestamp followed by a value. The data points are as follows:

Timestamp	Value
11:50:29.288	83
11:50:30.292	77
11:50:31.248	69
11:50:32.252	54
11:50:33.256	44
11:50:34.260	35
11:50:35.264	30
11:50:36.268	4
11:50:37.272	0
11:50:38.270	0
11:50:39.285	-2
11:50:40.289	0
11:50:41.294	4
11:50:42.297	6
11:50:43.300	30
11:50:44.303	34
11:50:45.308	37
11:50:46.312	44
11:50:47.317	55
11:50:48.320	79
11:50:49.284	81
11:50:50.288	88
11:50:51.292	100
11:50:52.297	114
11:50:53.301	126
11:50:54.305	135
11:50:55.308	144
11:50:56.312	175
11:50:57.316	177
11:50:58.318	180
11:50:59.335	182
11:51:00.339	176

The window also features a 'Send' button at the top right, a scroll bar on the right side, and a control bar at the bottom with the following options:  Autoscroll,  Show timestamp, Both NL & CR (dropdown), 115200 baud (dropdown), and Clear output.

Gambar IV.3 Data hasil pengujian kemiringan dengan *Arduino Terminal*.

#### 4.1.4 Pengujian Implementasi *Fuzzy Logic*

Pengujian implementasi *fuzzy logic* dilakukan dengan cara mengukur tingkat keberhasilan pengolahan data *g-force* dan kemiringan untuk mendeteksi

tingkat risiko pada saat kecelakaan dengan persentase risikonya. Hasil data dapat dilihat pada tabel IV.4.

Pada pengujian ini dilakukan dengan situasi dan kondisi seperti berikut :

- Tempat : *Indoor*
- Waktu : Siang
- Alat : Arduino Terminal dan Bluetooth Terminal
- Metode uji : Debug Serial dan Debug Bluetooth.
- Hasil : Terlampir pada tabel IV.4.

Tabel IV.4 Data hasil pengolahan menggunakan *Logika Fuzzy*.

No	Gforce (g)	Kemiringan (°)	Persentase Tingkat Kecelakaan (%)	Label Kecelakaan	Kesesuaian
1	0	85	12.5	Tidak Kecelakaan	Sesuai
2	0	85	12.5	Tidak Kecelakaan	Sesuai
3	0	85	12.5	Tidak Kecelakaan	Sesuai
4	0	85	12.5	Tidak Kecelakaan	Sesuai
5	0	85	12.5	Tidak Kecelakaan	Sesuai
6	2	73	12.5	Tidak Kecelakaan	Sesuai
7	3	68	12.5	Tidak Kecelakaan	Sesuai
8	5	55	12.5	Tidak Kecelakaan	Sesuai
9	6	45	12.5	Tidak Kecelakaan	Sesuai

10	7	37	27.5	-	Tidak Sesuai
11	7	32	52.5	-	Tidak Sesuai
12	8	28	62.5	Kecelakaan Cukup Parah	Sesuai
13	8	27	62.5	Kecelakaan Cukup Parah	Sesuai
14	8	8	62.5	Kecelakaan Cukup Parah	Sesuai
15	8	8	62.5	Kecelakaan Cukup Parah	Sesuai
16	8	7	62.5	Kecelakaan Cukup Parah	Sesuai
17	8	6	62.5	Kecelakaan Cukup Parah	Sesuai
18	8	7	62.5	Kecelakaan Cukup Parah	Sesuai
19	8	6	62.5	Kecelakaan Cukup Parah	Sesuai
20	7	36	32.5	-	Tidak Sesuai
21	5	54	12.5	Tidak Kecelakaan	Sesuai
22	1	79	12.5	Tidak Kecelakaan	Sesuai

23	0	88	12.5	Tidak Kecelakaan	Sesuai
24	0	90	12.5	Tidak Kecelakaan	Sesuai
25	1	101	12.5	Tidak Kecelakaan	Sesuai
26	3	116	12.5	Tidak Kecelakaan	Sesuai
27	5	128	12.5	Tidak Kecelakaan	Sesuai
28	5	135	12.5	Tidak Kecelakaan	Sesuai
29	6	145	25	-	Tidak Sesuai
30	6	147	28.12	-	Tidak Sesuai
31	7	147	47.5	-	Tidak Sesuai
32	7	177	62.5	Kecelakaan Cukup Parah	Sesuai
33	8	180	62.5	Kecelakaan Cukup Parah	Sesuai
34	7	180	62.5	Kecelakaan Cukup Parah	Sesuai
35	6	179	50	-	Tidak Sesuai
36	0	180	12.5	Tidak Kecelakaan	Sesuai
37	7	175	62.5	Kecelakaan Cukup Parah	Sesuai
38	3	181	12.5	Tidak Kecelakaan	Sesuai

39	6	180	50	-	Tidak Sesuai
40	7	180	62.5	Kecelakaan Cukup Parah	Sesuai
41	7	182	62.5	Kecelakaan Cukup Parah	Sesuai
42	7	181	62.5	Kecelakaan Cukup Parah	Sesuai
43	6	184	50	-	Tidak Sesuai
44	7	186	62.5	Kecelakaan Cukup Parah	Sesuai
45	6	184	50	-	Tidak Sesuai
46	6	184	50	-	Tidak Sesuai
47	7	182	62.5	Kecelakaan Cukup Parah	Sesuai
48	6	183	50	-	Tidak Sesuai
49	5	173	37.5	Kecelakaan Ringan	Sesuai
50	0	99	12.5	Tidak Kecelakaan	Sesuai
51	0	95	12.5	Tidak Kecelakaan	Sesuai
52	0	93	12.5	Tidak Kecelakaan	Sesuai
53	0	91	12.5	Tidak Kecelakaan	Sesuai

54	0	90	12.5	Tidak Kecelakaan	Sesuai
----	---	----	------	------------------	--------

Berdasarkan pada tabel IV.4 menunjukkan tingkat keberhasilan sistem dalam memproses *Fuzzy Logic* adalah :

$$\text{Keberhasilan} = \frac{\text{kesesuaian}}{\text{banyaknya percobaan}} \times 100\% = \frac{42}{54} \times 100\% = 77,7\%$$

## 4.2 Analisis

Setelah dilakukan beberapa pengujian alat dengan nilai keberhasilan sebagai berikut.

- Uji sensor MPU6050 GY512 = 100%
- Uji g-force = 92,3%
- Uji kemiringan = 93,75%
- Uji implementasi logika fuzzy = 77,7%

Dengan hasil tersebut dapat dianalisis bahwa sistem algoritma logika *fuzzy* yang diterapkan pada alat bekerja dengan baik. Namun nilai keberhasilannya relatif rendah. Penulis menyimpulkan rendah karena implementasi logika *fuzzy* ini akan diterapkan pada kasus kecelakaan, dimana pada kasus kecelakaan seharusnya sistem tidak boleh gagal dalam memberikan keputusan akhir.

Penulis menganalisis penyebab utama dari beberapa kegagalan sistem dikarenakan pengaruh nilai sensor yang tidak stabil pada keadaan yang begitu cepat.