

Prototipe Sistem Deteksi Kemacetan Jalan Raya Berbasis *Internet Of Things* (IoT)

Phisca Aditya Rosyady^{*}), Muslih Rayullan Feter, Zakky Ahmad Ikhsan M
Program Studi Teknik Elektro, Fakultas Teknologi Industri, Universitas Ahmad Dahlan,
Yogyakarta

* email: phisca.aditya@te.uad.ac.id

Abstract

Along with the progress of the times and technology is currently increasing the number of motorized vehicles on the highway. However, this increase in the number of motorized vehicles is not matched by an increase in road volume capacity, causing traffic congestion. The purpose of this study was to find out information about traffic conditions at highway intersections. This research is a prototype that describes a four-way intersection that has infrared sensors in each path. This infrared sensor is used as a vehicle detector. Tests on the prototype made various traffic conditions that are relevant to the actual situation. This research utilizes the concept of the Internet of things (IoT) prototype which is made to be connected to the internet network so that users can find out traffic conditions remotely and in real time. One of the information media used in this research is Twitter. The results of this study indicate that the prototype made can work well. The infrared sensor used can work optimally and can detect vehicles precisely at a sensitivity range of 4.5 cm. The average delay in sending notification tweets is 18 seconds.

Keywords: Traffic Congestion, Infrared, IoT, Twitter

1. Pendahuluan

Kemacetan lalu lintas mencerminkan ketidakseimbangan kondisi kemampuan ruas jalan dalam menampung kendaraan yang melintas pada ruas jalan tersebut. Kemacetan sering terjadi pada persimpangan yang terdapat lampu lalu lintas. Persimpangan ini dapat berupa simpang empat [1]. Antrian kendaraan bermotor saat lampu lalu lintas berwarna merah menjadi salah satu faktor terjadinya penumpukan kendaraan bermotor. Masalah kemacetan ini selalu membuat masyarakat menjadi resah [2]. Bagi kelompok maupun individu kemacetan dapat mengakibatkan kerugian yang besar. Salah satu contohnya adalah waktu yang terbuang sia-sia sehingga menimbulkan keresahan pada pengguna jalan. Keresahan ini dapat mengakibatkan pengguna jalan tergesa-gesa dalam berkedara sehingga menjadikan salah satu faktor terjadinya kecelakaan [3].

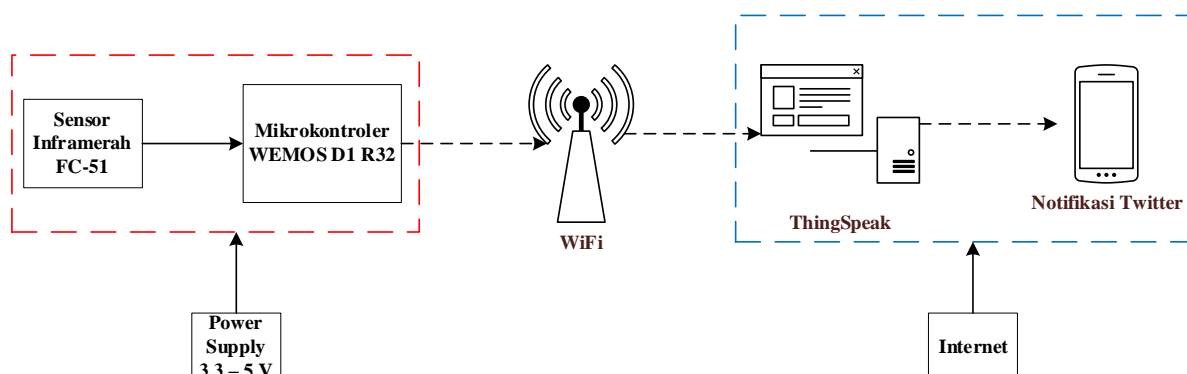
Berbagai cara dilakukan untuk mengurangi masalah kemacetan salah satunya adalah penelitian terdahulu yang membahas pengaturan lampu lalu lintas secara otomatis tetapi tidak disertai dengan monitoring kemacetan yang terjadi pada saat itu juga (*real time*) [4]. Minimnya informasi mengenai daerah yang mengalami kemacetan maka diperlukan suatu media informasi yang dapat memberikan informasi kondisi kemacetan lalu lintas di jalan raya pada saat itu juga (*real time*) [5]. Adapun salah satu cara untuk mengetahui informasi kemacetan dengan menggunakan aplikasi Google Maps, akan tetapi penggunaan aplikasi ini kurang akurat. Keakuratan aplikasi Google Maps ini memanfaatkan GPS pada *smartphone* pengguna yang berada di jalan raya, apabila terdapat kepadatan pengguna *smartphone* dan GPS menyala pada saat itu juga maka akan tertera kemacetan di aplikasi Google Maps, akan tetapi tidak semua orang menyalakan GPS pada *smartphone*-nya [6]. Prototipe penelitian ini dibuat menggunakan delapan sensor inframerah yang terbagi dalam empat ruas jalan dengan setiap ruas berisikan dua sensor inframerah diharapkan dapat lebih akurat dalam mendeteksi

kemacetan. Jarak antar sensor pada satu ruas jalan adalah 10 cm, setiap sensor inframerah ini bekerja untuk mendeteksi kendaraan yang kemudian digunakan sebagai pengkategorian dari tingkat kemacetan pada masing-masing ruas jalan [7]. Konsep yang dibuat pada prototipe ini adalah *Internet of Things (IoT)*, di mana prototipe akan terhubung dengan jaringan internet sehingga pengguna jalan dapat mengetahui kondisi lalu lintas secara *real time*. Pengguna *smartphone* dapat dengan mudah untuk mengakses informasi mengenai kondisi lalu lintas dengan menggunakan aplikasi Twitter [8]. Pemanfaatan Twitter sebagai salah satu media informasi yang sudah berkembang saat ini, diharapkan dapat mengurangi kemacetan pada jalan raya. Dengan mengetahui kondisi ruas jalan yang akan dilewati, tentu dapat membuat pengguna jalan terhindar dari kemacetan dan dapat memilih jalan alternatif lainnya.

2. Metode Penelitian

2.1 Diagram Blok

Dalam diagram permodelan pada Gambar 1 menunjukkan sistem secara keseluruhan. Pada sistem yang dibangun, WEMOS D1 R32 digunakan sebagai pemroses utama, sedangkan sensor inframerah FC-51 digunakan untuk mendeteksi halangan atau kendaraan pada prototipe. Pendeteksi halangan dengan sensor inframerah FC-51 memanfaatkan prinsip pemantulan sinar inframerah. Di mana transmiter akan memancarkan sinar inframerah dan kemudian diterima oleh *receiver* untuk mendapatkan letak objek dengan frekuensi yang sudah ditentukan pada IC LM393 dalam rangkaian modul sensor inframerah FC-51 [9].

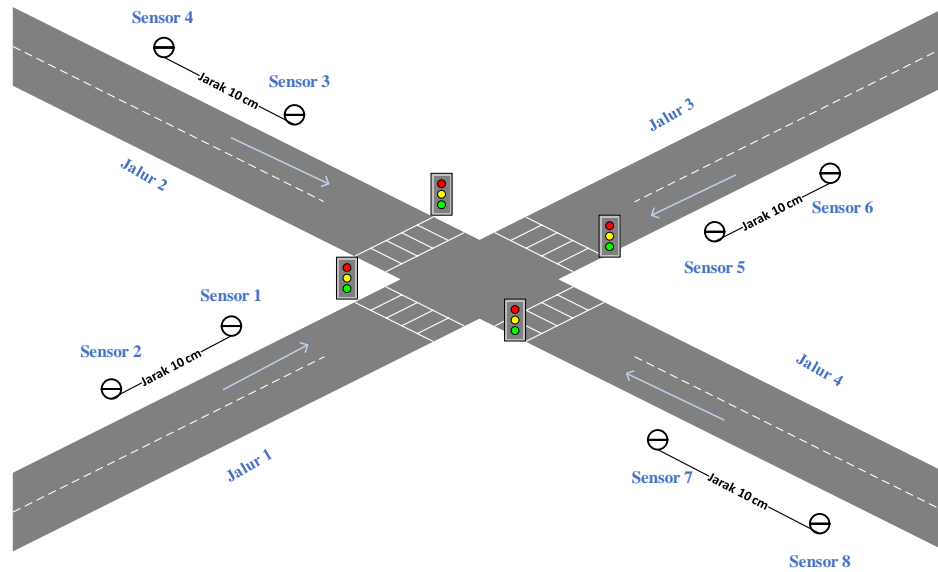


Gambar 1. Diagram Blok

Data dari sensor akan dikirim ke internet dengan menggunakan platform ThingSpeak, karena WEMOS D1 R32 sudah didukung dengan modul ESP32 sehingga sistem sudah mendukung untuk terkoneksi dengan WiFi dan Bluetooth sehingga tidak memerlukan modul tambahan. Perancangan sistem deteksi kemacetan ini memanfaatkan platform ThingSpeak untuk menyimpan data dan menampilkan dalam bentuk grafik, untuk notifikasi ketika ada halangan yang terdeteksi sistem akan mengirimkan pesan dalam bentuk *tweet*. Isi *tweet* berupa kondisi jalan pada setiap jalur [10].

2.2 Desain Prototipe

Gambar 2 merupakan desain dan skema perancangan *hardware* sistem deteksi kemacetan lalu lintas. Model prototipe yang akan dibuat adalah simpang empat yang berpotongan, kemudian akan diberikan rancangan penempatan komponen-komponen pada prototipe lalu lintas ini [11].



Gambar 2. Desain Prototipe

Spesifikasi dari desain prototipe yang dibangun yakni, panjang jalan jalur 1 dan jalur 3 adalah 57 cm untuk jalur 2 dan 4 adalah 65 cm. Lebar jalan pada setiap jalurnya 9 cm. sensor yang digunakan pada prototipe ini adalah inframerah FC-51 yang berjumlah 8 sensor. Masing – masing jalur memiliki 2 sensor dengan jarak antar sensor 10 cm [12]. Prototipe ini terbatas pada persimpangan jalan raya di mana dapat berupa simpang empat yang terdapat lalu lintasnya dan ruas yang dapat memuat dua buah kendaraan. Kendaraan ini meliputi mobil, truk, bus, dan lain sebagainya, kendaraan roda dua tidak termasuk.

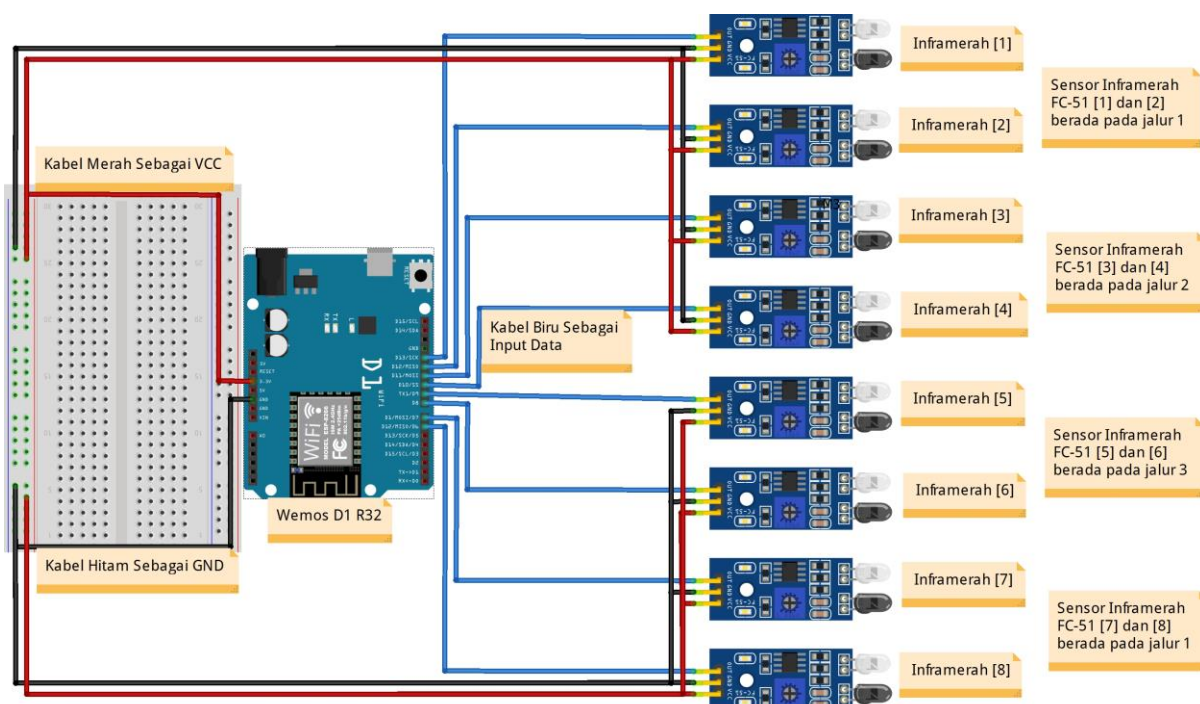
2.3 Skematik Rangkaian

Pada perancangan ini, sensor inframerah akan digunakan sebagai pendeteksi kendaraan pada setiap jalur. Sensor ini akan memberikan sinyal atau penanda masukan ke mikrokontroler untuk dikategorikan kondisi tingkat kemacetan pada setiap jalur. Berikut ini diagram pengkabelan dari prototipe yang dibuat pada Gambar 3.

Perancangan ini akan menempatkan sensor inframerah sebagai indikator untuk mengkategorikan kemacetan pada prototipe yang dibuat pada Gambar 3. Input daya yang diberikan adalah 3,3 V dan disebar pada setiap sensor inframerah. Pin yang digunakan pada mikrokontroler untuk menerima data adalah pin digital. Di mana pin digital ini akan membaca kondisi 0 (LOW) dan kondisi 1 (HIGH) [13]. Konfigurasi WEMOS D1 R32 dengan Inframerah FC-51 ditunjukkan pada Tabel 1.

Tabel 1. Konfigurasi WEMOS D1 R32 dengan Inframerah FC-51

Wemos D1 R32	Sensor Inframerah FC-51
VIN	3,3 V
GND	GND
IO18	INPUT – PIN SENSOR 1
IO19	INPUT – PIN SENSOR 2
IO23	INPUT – PIN SENSOR 3
IO13	INPUT – PIN SENSOR 4
IO14	INPUT – PIN SENSOR 5
IO27	INPUT – PIN SENSOR 6
IO16	INPUT – PIN SENSOR 7
IO17	INPUT – PIN SENSOR 8



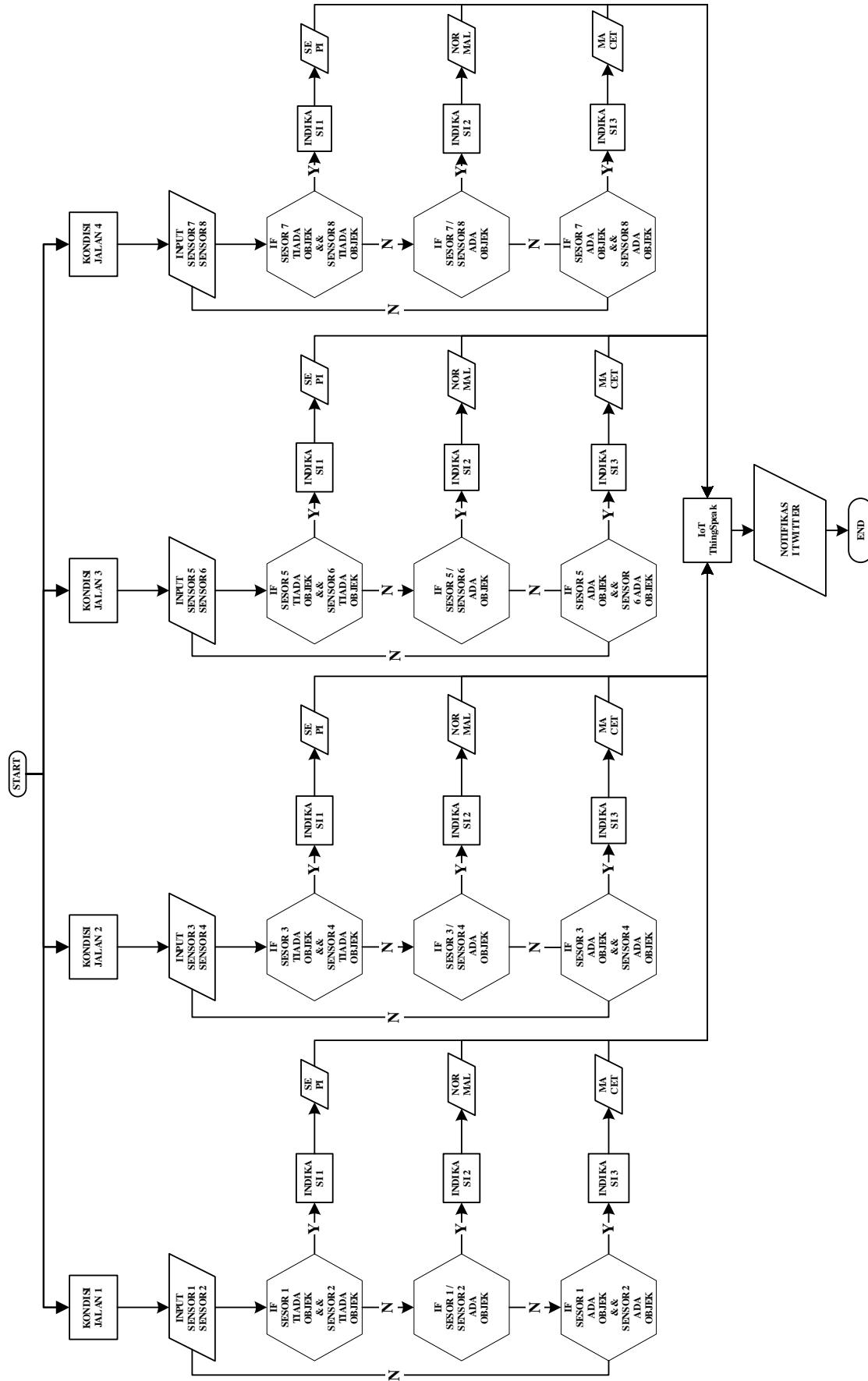
Gambar 3. Sekema Pengkabelan

2.4 Perancangan *Flowchart*

Flowchart disebut juga dengan diagram alir, gambar diagram yang menunjukkan simbol-simbol terhubung oleh panah untuk membentuk suatu alur proses atau *workflow* berfungsi untuk menjelaskan alur kerja suatu sistem atau untuk menggambarkan algoritma proses suatu mesin [14]. Diagram alir pada prototipe ini dapat dilihat pada Gambar 4.

Berdasarkan diagram alir pada Gambar 4, terdapat empat proses awal yakni proses kondisi jalan satu, kondisi jalan dua, kondisi jalan tiga dan kondisi jalan empat. Masing-masing proses dari setiap kondisi jalan diberikan masukan sensor inframerah FC-51 berfungsi untuk mendeteksi kendaraan. Penggunaan dua sensor inframerah yang diletakan pada masing-masing jalan ini berfungsi sebagai pengkategori dan menambah keakuratan dalam mendeteksi kemacetan. Sensor inframerah FC-51 bertanggung jawab untuk mengambil data. Data yang diambil hasilnya adalah berupa bilangan 0 (*Low*) dan 1 (*High*) karena pada dasarnya sensor inframerah merupakan sensor dalam kategori digital, dimana dalam kondisi 1 tidak mendeteksi kendaraan dan apabila kondisi 0 maka akan mendeteksi kendaraan [15].

Setelah kendaraan terdeteksi maka dibuat kategori kondisi lalu lintas yakni Sepi, Normal dan Macet [16]. Untuk kondisi Sepi, kedua sensor inframerah tidak mendeteksi kendaraan. Kondisi Normal adalah kondisi di mana salah satu sensor inframerah mendeteksi adanya kendaraan. Terakhir untuk kondisi Macet adalah kondisi kedua sensor inframerah mendeteksi adanya kendaraan. Apabila kondisi dari setiap jalan telah terpenuhi maka akan diproses dan diberikan label indikasi yakni 1 untuk kondisi Sepi, 2 untuk kondisi Normal dan 3 untuk kondisi Macet [17]. Setelah masing-masing jalan telah mendapatkan label indikasi maka akan dikirimkan pada server ThingSpeak menggunakan jaringan WiFi yang tersedia dan di tampung pada web ThingSpeak. Data yang telah di tampung pada web ThingSpeak akan di teruskan pada *react* ThingTwet yang kemudian akan membuat *tweet* pada halaman aplikasi Twitter. Pengguna akan mendapatkan notifikasi apabila mengikuti *channel* yang telah dibuat atau dapat juga mengakses *channel*nya [18].

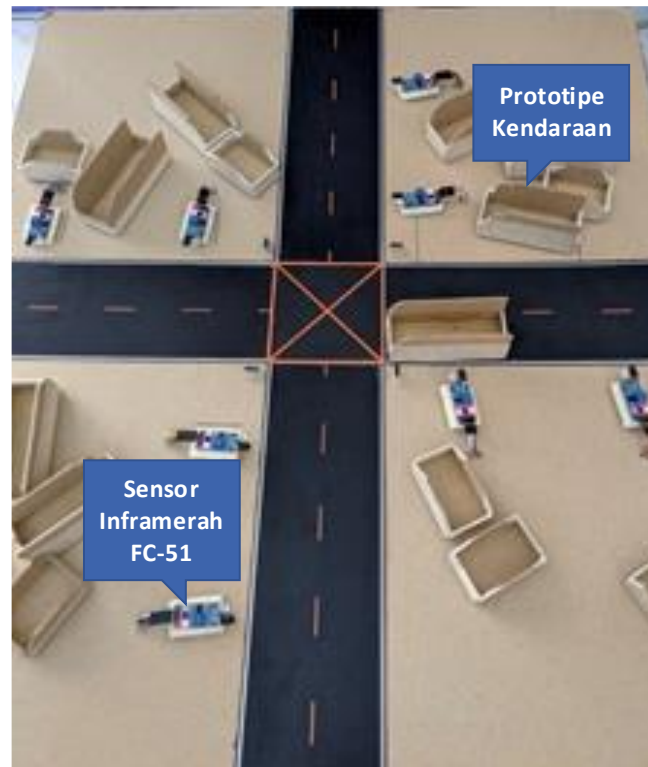


Gambar 4. Diagram Alir

3. Hasil dan Pembahasan

3.1 Pengujian Sensor Inframerah FC-51

Pada penelitian ini, dilakukan pengujian sensor inframerah FC-51 bertujuan untuk mengetahui bagaimana respon sensor terhadap halangan atau kendaraan pada prototipe yang dibuat. Sensor ini akan mengirimkan hasil dalam bentuk digital yakni *high* dan *low* [19].



Gambar 5. Inframerah mendeteksi Kendaraan

Gambar 5 menunjukkan keadaan sensor inframerah dalam kondisi hidup ditandai dengan kondisi satu LED yang hidup pada modul sensor inframerah dan ketika sensor inframerah mendeteksi objek akan ditandai dengan dua lampu LED yang menyala pada modul sensor inframerah. Sensor inframerah yang telah diuji ini memberikan hasil seperti pada Tabel 2.

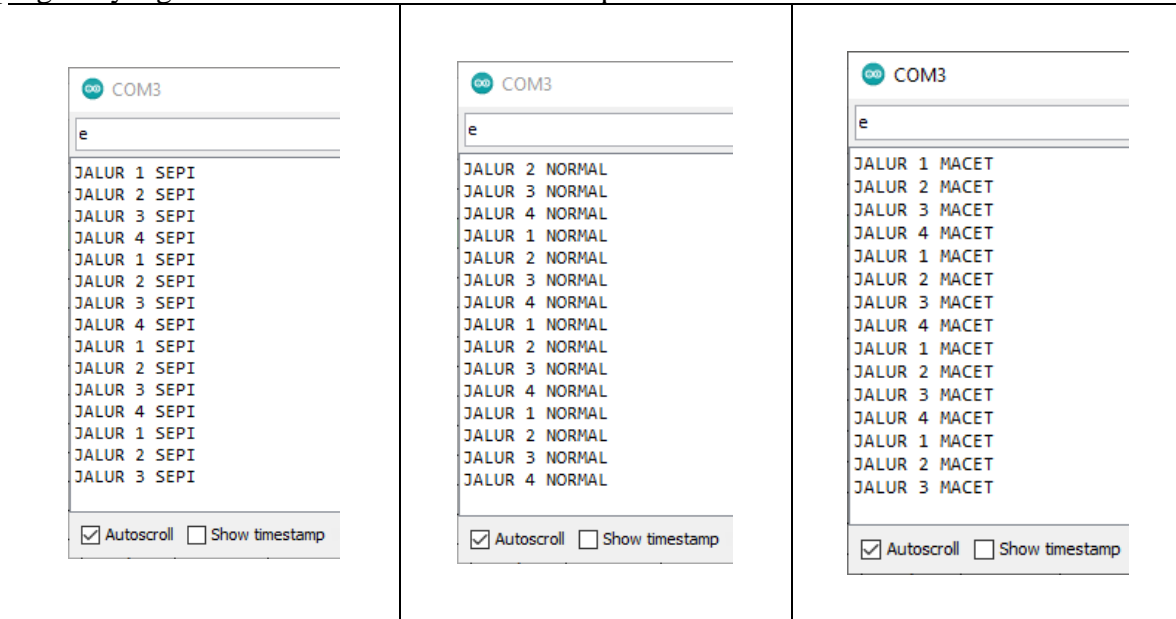
Tabel 2. Pengujian Sensor Inframerah

	Jalur 1	Jalur 2	Jalur 3	Jalur 4
Sensor 1	✓	✓	✓	✓
Sensor 2	✓	✓	✓	✓

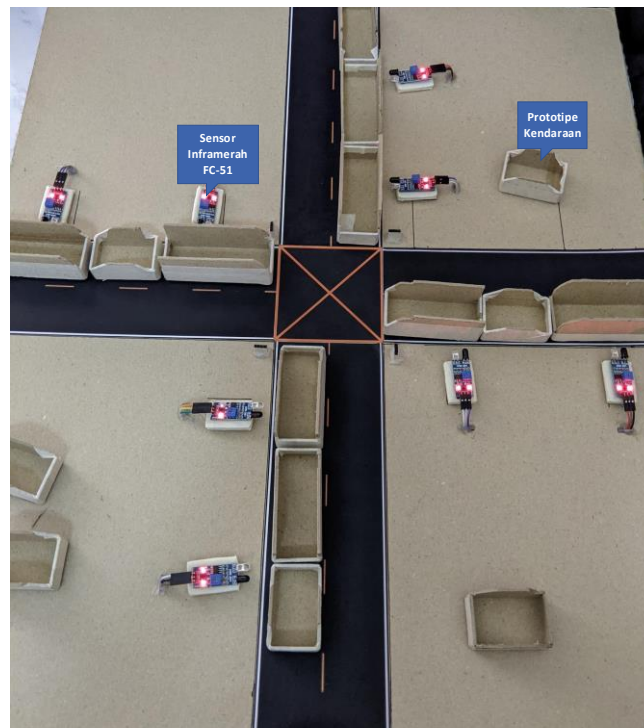
Tabel 2 menunjukkan bahwa sensor inframerah dapat bekerja mendeteksi halangan atau kendaraan dengan baik. Pengujian ini dilakukan dengan cara diberikan halangan di depan masing-masing sensor agar dapat mendeteksi halangan tersebut yang bertujuan untuk mengetahui modul sensor inframerah terdapat *error* atau tidak. Jika tidak terdapat *error* maka akan diberi tanda *ceklist* seperti pada Tabel 2. Rentang jarak yang digunakan untuk mendeteksi kendaraan adalah 4,5 cm. Digunakan rentang jarak 4,5 cm dikarenakan lebar total jalan adalah 9 cm yang kemudian dibagi menjadi dua bagian yakni ruas kiri dan ruas kanan.

3.2 Pengujian Program Sistem Deteksi Kemacetan

Setelah dilakukan pengujian pada sensor inframerah dilanjutkan dengan pengujian program sistem deteksi kemacetan. Program yang telah dibuat pada *platform* Arduino IDE dilakukan pengujian dalam beberapa kondisi lalu lintas yang relevan pada keadaan real yakni kondisi Sepi, Normal dan Macet [20]. Hasil dari program Arduino IDE dapat dilihat pada Gambar 6 yang menunjukkan hasil pada serial monitor. Pengujian prototipe dapat dilihat pada Gambar 7 yang menunjukkan kondisi semua jalur dalam keadaan Macet, sehingga sesuai dengan program yang telah dibuat dalam Arduino IDE pada Gambar 6.



Gambar 6. Pengujian Sistem Pada Serial Monitor



Gambar 7. Kondisi Semua Jalur Macet

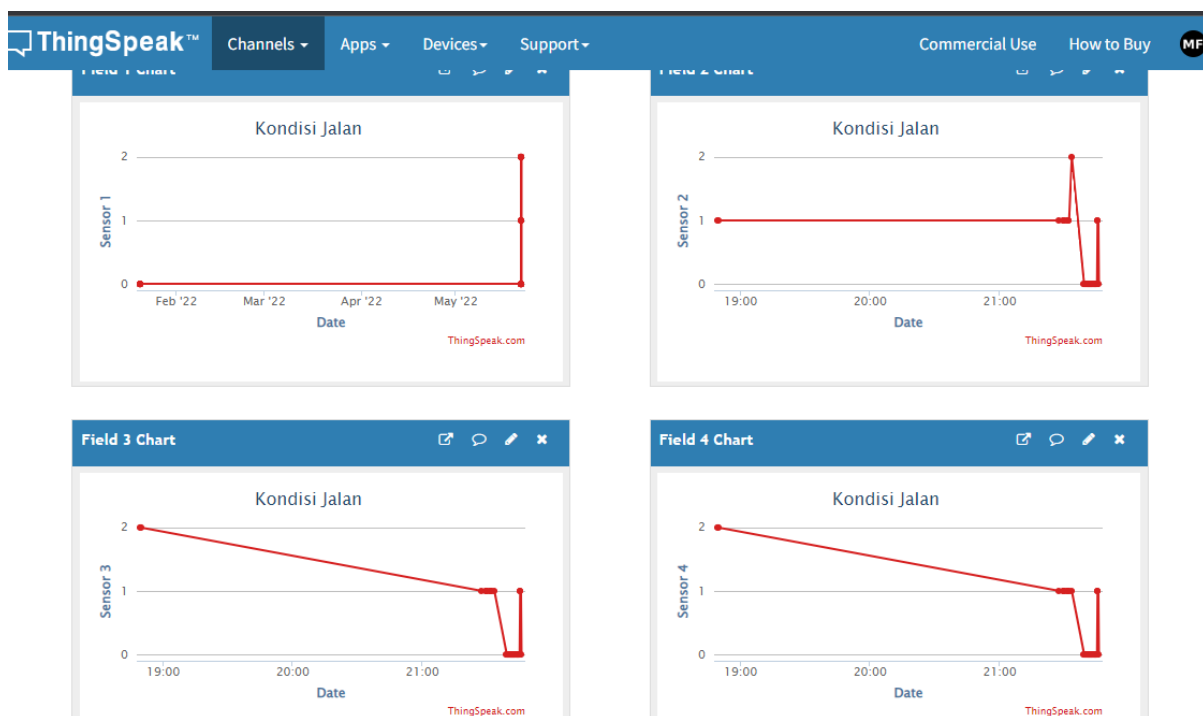
Tabel 3. Hasil Pengujian Program

	Jalur 1	Jalur 2	Jalur 3	Jalur 4
Sepi	✓	✓	✓	✓
Normal	✓	✓	✓	✓
Macet	✓	✓	✓	✓

Pada Tabel 3 menunjukkan bahwa program yang dibuat pada Arduino IDE dapat berjalan dengan baik. Dalam pengujian ini, tidak terjadi adanya *error*, baik dalam program yang dibuat maupun sensor inframerah yang digunakan. Pengujian ini membandingkan program pada Arduino IDE dengan pengujian pada prototipe dimana ketika prototipe diberikan kondisi lalu lintas semua jalur dalam kondisi Macet seperti Gambar 7 apakah program yang telah dibuat akan sesuai dengan kondisi lalu lintas pada prototipe. Apabila kondisi sesuai maka pada Tabel 3 akan diberikan tanda *ceklist* yang menandakan tidak terjadi kesalahan.

3.3 Pengujian Internet of Things (IoT)

Pengujian ini bertujuan untuk mengetahui hasil pembacaan data dari sensor inframerah yang dikirimkan melalui mikrokontroler WEMOS D1 R32 dengan menggunakan jaringan WiFi [21]. *Platform* yang digunakan untuk IoT pada prototipe ini adalah ThingSpeak. Gambar 8 menunjukkan hasil data pengujian sensor inframerah dalam bentuk grafik yang telah diberikan label indikasi kategori kondisi lalu lintas dimana 0 adalah kondisi Sepi, 1 adalah kondisi Normal dan 2 adalah kondisi Macet. Data dari pengujian yang dilakukan sebelumnya ditampung dalam *database* ThingSpeak yang dapat diakses melalui web <https://thingspeak.com/>.



Gambar 8. Grafik Kondisi Jalur 1 Macet Dan Jalur 2 Sampai Dengan Jalur 4 Normal

Dalam setiap pengujian ini dilakukan dengan kondisi permulaan prototipe tidak terhubung dengan sumber tegangan. Kemudian saat sistem sudah terhubung dengan sumber tegangan didapatkan delay waktu pengiriman data dari WEMOS D1 R32 ke *platform* IoT ThingSpeak adalah 15 detik akan tetapi waktu ini akan berubah tergantung dari jaringan internet itu sendiri [22].

3.4 Pengujian Prototipe Secara Keseluruhan

Pengujian prototipe secara keseluruhan bertujuan untuk mengetahui kecepatan pengiriman *tweet* dari WEMOS D1 R32 ke halaman aplikasi Twitter. Untuk menentukan kemungkinan kondisi relevan dengan keadaan real peneliti menggunakan Persamaan kombinasi (1) lalu kombinasi yang dirumuskan adalah satu jalan dengan tiga kondisi yakni sepi, normal dan macet (2).

$$nCr = \frac{n!}{(n-r)!r!} \tag{1}$$

Dari Persamaan (1), didapatkan hasil kombinasi Persamaan (2);

$$1C3 \times 1C3 \times 1C3 \times 1C3 \tag{2}$$

$$= \frac{3!}{(3-1)!1!} \times \frac{3!}{(3-1)!1!} \times \frac{3!}{(3-1)!1!} \times \frac{3!}{(3-1)!1!} \tag{3}$$

$$= \frac{3!}{2!1!} \times \frac{3!}{2!1!} \times \frac{3!}{2!1!} \times \frac{3!}{2!1!} \tag{4}$$

$$= \frac{3 \times 2!}{2! \times 1} \times \frac{3 \times 2!}{2! \times 1} \times \frac{3 \times 2!}{2! \times 1} \times \frac{3 \times 2!}{2! \times 1} \tag{5}$$

$$= 3 \times 3 \times 3 \times 3 = 81 \tag{6}$$

Dari perhitungan persamaan (3), (4) dan (5) didapatkan hasil pada persamaan (6), 81 macam kemungkinan kondisi yang relevan untuk dijadikan bahan pengujian dalam pada penelitian. ini dapat dilihat pada Tabel 4.

Tabel 4. Kondisi Lalu Lintas yang Relevan

No	Kondisi Jalan				No	Kondisi Jalan			
	Jalan 1	Jalan 2	Jalan 3	Jalan 4		Jalan 1	Jalan 2	Jalan 3	Jalan 4
1	S	S	S	S	43	N	N	M	S
2	S	S	S	N	44	N	N	M	N
3	S	S	S	M	45	N	N	M	M
4	S	S	N	S	46	N	M	S	S
5	S	S	N	N	47	N	M	S	N
6	S	S	N	M	48	N	M	S	M
7	S	S	M	S	49	N	M	N	S

... Lanjutan Tabel 4

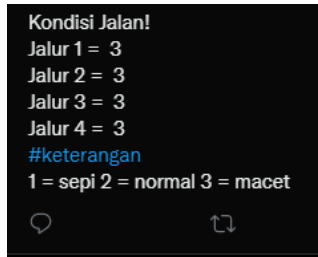
8	S	S	M	N	50	N	M	N	N
9	S	S	M	M	51	N	M	N	M
10	S	N	S	S	52	N	M	M	S
11	S	N	S	N	53	N	M	M	N
12	S	N	S	M	54	N	M	M	M
13	S	N	N	S	55	M	S	S	S
14	S	N	N	N	56	M	S	S	N
15	S	N	N	M	57	M	S	S	M
16	S	N	M	S	58	M	S	N	S
17	S	N	M	N	59	M	S	N	N
18	S	N	M	M	60	M	S	N	M
19	S	M	S	S	61	M	S	M	S
20	S	M	S	N	62	M	S	M	N
21	S	M	S	M	63	M	S	M	M
22	S	M	N	S	64	M	N	S	S
23	S	M	N	N	65	M	N	S	N
24	S	M	N	M	66	M	N	S	M
25	S	M	M	S	67	M	N	N	S
26	S	M	M	N	68	M	N	N	N
27	S	M	M	M	69	M	N	N	M
28	N	S	S	S	70	M	N	M	S
29	N	S	S	N	71	M	N	M	N
30	N	S	S	M	72	M	N	M	M
31	N	S	N	S	73	M	M	S	S
32	N	S	N	N	74	M	M	S	N
33	N	S	N	M	75	M	M	S	M
34	N	S	M	S	76	M	M	N	S
35	N	S	M	N	77	M	M	N	N
36	N	S	M	M	78	M	M	N	M
37	N	N	S	S	79	M	M	M	S
38	N	N	S	N	80	M	M	M	N
39	N	N	S	M	81	M	M	M	M
40	N	N	N	S					
41	N	N	N	N					
42	N	N	N	M					

Keterangan :
S = Sepi
N = Normal
M = Macet

Pada Tabel 4, diuraikan 81 kondisi lalu lintas yang mungkin terjadi pada simpang empat dengan tiga kondisi disetiap jalannya. Pengujian prototipe secara real dan pengujian menggunakan serial monitor dapat berjalan dengan baik. Kemudian didapatkan juga pengiriman data dari WEMOS D1 R32 ke server *platform* ThingSpek dapat berjalan dengan baik, ditandai dengan munculnya *tweet* yang masuk dalam halaman aplikasi Twitter seperti pada Gambar 13 sampai dengan Gambar 15.

Secara keseluruhan prototipe yang dibuat dapat berjalan dengan baik dan berjalan sesuai dengan yang di kehendaki peneliti. Kemudian, dilakukan analisis kecepatan pengiriman

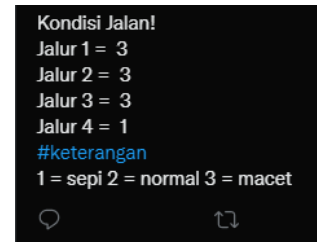
data dari WEMOS D1 R32 menuju server ThingSpeak yang ditampilkan pada halaman Twitter [23]. Jaringan WiFi yang terhubung pada WEMOS D1 R32 ini menggunakan penyedia layanan internet “Indosat”. Layanan internet ini menggunakan *smartphone* sebagai *HotSpot*, dengan memanfaatkan kemampuan modul ESP32 yang sudah tertanam pada WEMOS D1 R32 dan dapat terkoneksi dengan WiFi dan Bluetooth dengan mudah. Hasil analisis dimuat pada Tabel 5.



Gambar 13. Tweet Menunjukkan Kondisi Semua Jalur Macet



Gambar 14. Tweet Menunjukkan Jalur 1 Sampai Jalur 3 Macet Dan Jalur 4 Normal



Gambar 15. Tweet Menunjukkan Jalur 1 Sampai Jalur 3 Macet Dan Jalur 4 Sepi

Tabel 5. Analisis Waktu Delay

No	Waktu (detik)			
	Twitter	Stopwatch	Real	Error
1	0	1	1	0%
2	2	23	21	10%
3	3	15	12	25%
4	1	19	18	6%
5	5	26	21	24%
6	4	19	15	27%
7	2	20	18	11%
8	0	18	18	0%
9	0	12	12	0%
10	2	12	10	20%
11	1	13	12	8%
12	2	20	18	11%
13	0	5	5	0%
14	3	24	21	14%
15	2	9	7	29%
16	5	35	30	17%
17	3	25	22	14%
18	2	13	11	18%
19	4	22	18	22%
20	3	23	20	15%
21	1	14	13	8%
22	1	8	7	14%
23	3	18	15	20%
43	1	24	23	4%
44	2	11	9	22%
45	1	6	5	20%
46	0	14	14	0%
47	0	5	5	0%
48	2	17	15	13%
49	2	10	8	25%
50	1	18	17	6%
51	3	12	9	33%
52	2	23	21	10%
53	1	17	16	6%
54	3	17	14	21%
55	0	11	11	0%
56	0	18	18	0%
57	25	87	62	40%
58	1	10	9	11%
59	0	8	8	0%
60	2	22	20	10%
61	0	15	15	0%
62	4	11	7	57%
63	0	58	58	0%
64	3	16	13	23%
65	0	22	22	0%

... Lanjutan Tabel 5

24	0	5	5	0%
25	1	13	12	8%
26	3	21	18	17%
27	0	8	8	0%
28	3	41	38	8%
29	0	3	3	0%
30	2	28	26	8%
31	4	17	13	31%
32	2	20	18	11%
33	4	37	33	12%
34	0	41	41	0%
35	4	47	43	9%
36	2	33	31	6%
37	2	15	13	15%
38	1	27	26	4%
39	0	18	18	0%
40	3	25	22	14%
41	0	9	9	0%
42	60	121	61	98%
66	1	8	7	14%
67	2	18	16	13%
68	0	5	5	0%
69	1	17	16	6%
70	2	16	14	14%
71	0	10	10	0%
72	0	6	6	0%
73	4	25	21	19%
74	3	20	17	18%
75	2	14	12	17%
76	47	110	63	75%
77	3	16	13	23%
78	3	27	24	13%
79	1	17	16	6%
80	2	24	22	9%
81	0	14	14	0%
Rerata:			18	13%

Keterangan:
0 = waktu "sekarang"

Pengujian ini memberikan target di mana waktu kecepatan pengirim minimal kurang dari 60 detik. Hasil nilai waktu pada Twitter adalah *delay* waktu sesaat setelah melakukan *refresh* halaman pada aplikasi Twitter. Nilai real didapat dari perhitungan antara nilai waktu pada *stopwatch* dan nilai waktu pada Twitter ditentukan pada persamaan (7).

$$\text{nilai real} = \text{nilai waktu pada stopwatch} - \text{nilai waktu Twitter} \quad (7)$$

Kemudian, untuk perhitungan nilai *error* menggunakan Persamaan (8).

$$\text{persen error} = \frac{(\text{nilai waktu stopwatch} - \text{nilai waktu real})}{\text{nilai waktu real}} \times 100\% \quad (8)$$

Pada analisis ini didapatkan tiga kondisi waktu pengiriman yang melebihi batas waktu yang diharapkan. Berikut kondisi yang dimaksud;

1. Kondisi jalan 1 sampai dengan jalan 3 dalam kondisi normal dan jalan 4 dalam kondisi macet.
2. Kondisi jalan 2 sampai jalan 3 sepi dan kondisi jalan 1 serta 4 dalam kondisi macet.
3. Kondisi jalan 1 sampai 2 dalam kondisi macet dan kondisi jalan 3 normal serta jalan 4 dalam kondisi sepi.

Setelah dilakukan perhitungan, didapat persen nilai *error* dari masing-masing waktu yang ditunjukkan pada Tabel 5 dan didapatkan rerata dari nilai *error* sebesar 13%. Dari tingkat persen yang rendah ini, dapat dipastikan untuk *delay* pengiriman notifikasi *tweet* masih tergolong *real time* sehingga pengguna mendapatkan notifikasi kondisi lalu lintas sesegera mungkin. *Delay* pengiriman ini dapat berubah apabila menggunakan penyedia layanan internet yang lainnya.

Pada pengujian ini, menggunakan pengaturan pada web ThingSpeak agar setiap terdeteksi kendaraan atau halangan akan mengirimkan notifikasi pada Twitter sehingga apabila tidak terjadi pergantian kondisi lalu lintas maka sistem tidak akan mengunggah kondisi lalu lintas pada halaman Twitter. Apabila sistem terus menerus mengunggah kondisi lalu lintas maka akan terjadi spam pada halaman Twitter. Diharapkan ketika pengguna setelah mendapatkan notifikasi Twitter apabila terjadi kemacetan pengguna dapat menentukan jalan alternatif yang akan dilewati. Sehingga kemacetan dapat berkurang dikarenakan tidak terjadinya penumpukan kendaraan yang berlebih pada jalan yang sudah terdeteksi kemacetan.

4. Kesimpulan

Dalam penelitian ini prototipe dibuat berbentuk persegi dengan menggunakan total 8 (delapan) sensor inframerah FC-51. Dengan 2 (dua) sensor disetiap jalan dapat diperoleh kondisi lalu lintas yang lebih optimal. Metode pengambilan data pada penelitian ini dengan cara meletakkan prototipe kendaraan di atas ruas jalan yang telah dibuat. Jarak masing-masing sensor inframerah FC-51 pada setiap ruas jalan adalah 10 cm. Rentang sensitifitas sensor adalah 4.5 cm dengan lebar jalan 9 cm. Sensor dapat mendeteksi objek dengan akurat. Hasil pengujian menunjukkan *delay* pada kecepatan jaringan dalam pengiriman *tweet* rata – rata adalah 18 detik. Nilai error pada *delay* pengiriman *tweet* yang didapatkan adalah 13%.

Referensi

- [1] Fatimah, S., Syakdiah, S., & Kusumawiranti, R. (2022). Kebijakan Pemerintah dalam Mengatasi Kemacetan di Kota Yogyakarta (Studi Penelitian di Jalan Malioboro di JalanTentara Pelajar). *POPULIKA*, 10(1), 24-41.
- [2] Girsang, W. E. J. (2020). Analisis Kerugian Pengguna Jalan Akibat Kemacetan Lalu Lintas di Kota Medan. [Online]. Available: <https://repositori.usu.ac.id/handle/123456789/31014>
- [3] Purba, A., & Sadnowo, A. (2017). Pengembangan Sistem Monitoring Lampu Lalu-Lintas Berbasis Microcontroller dengan Sms Jaringan GSM. <http://cantilever.unsri.ac.id>.
- [4] Ramadhan, H. K. (2020). Prototipe Lampu Lalu Lintas Dengan Pewaktuan Adaptif. *JTET (Jurnal Teknik Elektro Terapan)*, 9(1), 1-5.
- [5] Rosyady, P. A., & Sumiharto, R. (2018). Highway Visual Tracking System using Thresholding and Hough Transform. *Jurnal Ilmiah Teknik Elektro Komputer dan Informatika*, 4(2), 93-99.
- [6] Siswaya, S., Sunardi, S., & Yudhana, A. (2018). Sistem Pakar Sebagai Pengendali Lampu Lalu-Lintas Pada Persimpangan Jalan Menggunakan Fuzzy Logic Berbasis Android. *Prosiding SNST Fakultas Teknik*, 1(1).
- [7] Espressif Systems. (2022). ESP32WROOM32. [Online]. Available: <https://www.espressif.com/en/support/download/documents>.
- [8] Rosyady, P. A., Fajeri, F., & Agustian, M. A. (2022). Pengukuran Kedalaman dan Koordinat Jalan Berlubang Menggunakan Sensor Ultrasonik dan GPS Berbasis Internet of Things (IoT). *AVITEC*, 4(1), 1-12.
- [9] Siswaya, S., Sunardi, S., & Yudhana, A. (2021). Analisis Sistem Traffic Light Untuk Optimalisasi dan Antisipasi Kemacetan Lalu Lintas Berbasis Android. *Respati*, 16(3), 86-91.
- [10] BR, N. R. (2019). Prototype Smart Traffic Light Otomatis Berbasis Atmega-328 dengan Sensor Jarak. *JE-Unisla*, 4(2), 265-268.
- [11] Rivai, D. P., & Santoso, B. (2019). Model Rekayasa Traffic Light Menggunakan Arduino. *Jurnal Cosphi*, 3(1).

- [12] Rusmana, D. (2019). Design Of Traffic Light Using Infrared Sensor Based on Fuzzy Logic. *Incomtech*, 8(2), 37-42.
- [13] Juniana, P., & Hakim, L. (2019). Kendali Lampu Lalu Lintas Dengan Menggunakan Metode Fuzzy Logic Mamdani. *J. Terap. Teknol. Inf*, 3(1), 1-10.
- [14] Saputra, I., & Slameta, S. (2020, September). Prototipe Penggunaan Sensor Ultrasonik Terintegrasi Dengan Jaringan Internet Google Firebase Untuk Pengaturan Durasi Lampu Lalu Lintas. In *Prosiding Industrial Research Workshop and National Seminar* (Vol. 11, No. 1, pp. 287-292).
- [15] Alaidi, A., Aljazaery, I., Alrikabi, H., Mahmood, I., & Abed, F. (2020). Design and implementation of a smart traffic light management system controlled wirelessly by arduino. *International Journal of Interactive Mobile Technologies*, vol. 14, no. 7, pp. 32–40, 2020, doi: 10.3991/ijim.v14i07.12823.
- [16] Mudjanarko, S. W., Winardi, S., & Limantara, A. D. (2017). Pemanfaatan internet of things (iot) sebagai solusi manajemen transportasi kendaraan sepeda motor. *Pros. Semin. Nas. Apl. Teknol. Prasarana Wil. X*, August, pp. 1–16, 2017, doi: 10.31219/osf.io/6ue4b.
- [17] Pravalika, V., & Prasad, C. R. (2019). Internet of things based home monitoring and device control using Esp32. *International Journal of Recent Technology and Engineering*, 8(1S4), 58-62.
- [18] Mariyam, M., Asparizal, A., & Azkiya, A. (2018). Pengembangan Simulasi Pengendalian Lampu Lalu Lintas Dan Pendeteksi Kepadatan Berbasis Arduino Mega 2560 Menggunakan Ldr Dan Laser. *Lentera Dumai*, 9(2).
- [19] Fezari, M., & Al Dahoud, A. (2018). Integrated development environment “IDE” for Arduino. *WSN applications*, 1-12.
- [20] Limantara, A. D., Winardi, S., & Mudjanarko, S. W. (2017). Pemanfaatan Internet of Things (IoT) Sebagai Solusi Manajemen Transportasi Kendaraan Sepeda Motor. [Online]. Available: <https://www.researchgate.net/publication/320699658>
- [21] Ramadhan, Z. A., Mohammed, B. K., & Alwaily, A. H. (2021). Design and implement a smart traffic light controlled by internet of things. *Periodicals of Engineering and Natural Sciences (PEN)*, 9(4), 542-548.
- [22] Novikov, A., Novikov, I., & Shevtsova, A. (2019). Modeling of traffic-light signalization depending on the quality of traffic flow in the city. *Journal of applied engineering science*, 17(2), 175-181.
- [23] Shenbagavalli, S., Priyadharshini, T., Sowntharya, S., Manikandan, P., & Saravanan, D. S. (2020). Design and implementation of smart traffic controlling system. *International Journal of Engineering Technology Research & Management*, 4(4), 28-36.