

Analisis Perbandingan Kinerja Sensor Jarak HC-SR04 dan GP2Y0A21YK Menggunakan Thingspeak dan Wireshark

Iman Hedi Santoso dan Arif Indra Irawan

Fakultas Teknik Elektro, Universitas Telkom

Jl. Telekomunikasi No. 1, Terusan Buah Batu, Sukapura, Dayeuhkolot, Bandung 40257

e-mail: imanhedis@telkomuniversity.ac.id

Abstrak—*Internet of Things (IoT)* sampai saat ini merupakan topik yang sangat menarik untuk diteliti. Hal tersebut disebabkan oleh luasnya peranan yang dapat dilakukan IoT pada kehidupan umat manusia. Penelitian ini bertujuan untuk membandingkan kinerja dua buah sensor pada sebuah sistem pendeteksi jarak berbasis IoT dengan fokus parameter yang diuji adalah: pembacaan sensor, QoS pengiriman data, dan kebutuhan daya. Adapun dua sensor yang menjadi subyek perbandingan adalah HC-SR04 dan GP2YA21YK. Sebagai alat bantu analisis, penelitian ini menggunakan dua *tool*, yaitu *Thingspeak* dan *Wireshark*. Hasil uji kinerja menunjukkan bahwa dalam hal ketepatan penentuan jarak, HC-SR04 memiliki kinerja yang jauh lebih baik dibandingkan GP2YA21YK. Pada HC-SR04, rata-rata galat pembacaan sebesar 0,82 cm, sedangkan pada GP2YA21YK sebesar 14,40 cm. Sementara itu, ditinjau dari parameter QoS, kedua sistem sensor menunjukkan kinerja yang hampir sepadan, besar *packet loss* sama-sama 0%, nilai *throughput* sebesar 37,01 kbps pada HC-SR04 dan 38,12 kbps pada GP2YA21YK. Sedangkan untuk *delay*, sensor HC-SR04 memberikan nilai sebesar 33,55 ms, dan pada GP2YA21YK sebesar 26,1 ms. Selanjutnya, berdasarkan kebutuhan daya, sistem sensor yang menggunakan HC-SR04 mengkonsumsi daya 14,36% lebih kecil dibandingkan sistem yang menggunakan GP2YA21YK. Dengan merujuk pada hasil-hasil pengukuran dan visualisasi menggunakan *Wireshark* dan *Thingspeak*, dapat disimpulkan bahwa dalam sistem deteksi jarak menggunakan sensor HC-SR04 lebih baik dibandingkan sistem dengan sensor GP2YA21YK.

Kata kunci: *Internet of things, thingspeak, wireshark, sensor, jarak*

Abstract—Until now, Internet of Things (IoT) is a very interesting topic to research. This is due to the wide role that IoT can play in human life. This study aims to compare the performance of two sensors in an IoT-based distance detection system with the focus of the parameters being tested: sensor readings, QoS of data transmission, and power requirements. The two sensors that are the subject of comparison are HC-SR04 and GP2YA21YK. As an analytical tool, this research uses two tools, namely Thingspeak and Wireshark. The performance test results show that in terms of accuracy in determining distance, the HC-SR04 has a much better performance than the GP2YA21YK. On HC SR04, the average reading error is 0.82 cm, while on GP2YA21YK it is 14.40 cm. Meanwhile, in terms of QoS parameters, the two sensor systems show almost commensurate performance, the packet loss is both 0%, the throughput value is 37.01 kbps on HC-SR04 and 38.12 kbps on GP2YA21YK. As for the delay, the HC-SR04 sensor gives a value of 33.55 ms, and on GP2YA21YK it is 26.1 ms. Furthermore, based on power requirements, sensor systems using the HC-SR04 consume 14.36% less power than the system that use GP2YA21YK. By referring to the results of measurements and visualizations using Wireshark and Thingspeak, it can be concluded that the distance detection system using the HC-SR04 sensor is better than the system with the GP2YA21YK.

Keywords: *Internet of things, thingspeak, wireshark, sensor, distance*

I. PENDAHULUAN

Hari ini para peneliti dunia begitu tertarik dengan *Internet of Things (IoT)*, karena IoT begitu dipercaya dapat memberikan solusi atas tantangan-tantangan sosial yang dihadapi masyarakat dunia. Diantara tantangan-tantangan sosial tersebut adalah hal-hal seperti: kesejahteraan, perubahan demografi dan kesehatan; *food security* dan pertanian berkelanjutan; energi yang aman, bersih, dan efisien; transportasi yang cerdas, hijau, dan terintegrasi; perubahan iklim, lingkungan, efisiensi sumberdaya alam,

dan bahan mentah; masyarakat yang inklusif, inovatif, dan reflektif; keamanan masyarakat. Dengan peranannya tersebut, maka IoT akan menjadi bagian dari internet masa depan, yang mana didalamnya milyaran objek yang bersifat heterogen akan saling terhubung [1], [2]. Gambar 1 memperlihatkan bahwa dimasa depan segala sesuatu hampir semuanya akan terhubung, mulai dari alat komunikasi tradisional, seperti: PC, laptop, *smarthphone*, atau *tablet*, sampai dengan alat-alat rumah tangga, seperti: lemari pendingin, pintu garasi, dan dari sistem-sistem industri sampai kepada sistem-sistem yang melekat pada

manusia [3].

IoT lahir dari gabungan antara teknologi *Wireless Sensor Network* (WSN), *Machine-to-Machine* (M2M), dan *Radio-Frequency Identification* (RFID), dimana dalam IoT, sensor-sensor dan aktuator bercampur secara *seamless* [4]. Sebagai salah satu *enabling technology* bagi terwujudnya IoT, WSN telah memungkinkan kemampuan *ubiquitous sensing* dapat dilakukan pada berbagai sektor kehidupan saat ini. WSN menawarkan kemampuan untuk mengukur, menginterpretasi, dan mengerti indikator lingkungan, dari mulai masalah ekologi dan sumber daya alam sampai ke masalah-masalah di perkotaan. Dengan kemampuan yang dimiliki WSN, jumlah sensor-sensor yang tergabung dalam WSN selalu mengalami peningkatan dari tahun ke tahun.

Berdasarkan penjelasan di atas, dapat diketahui bahwa peranan sensor dalam era IoT sangatlah vital. Oleh karena itu, penelitian ini melakukan pendalaman pada kinerja sensor yang biasa digunakan dalam proyek-proyek IoT. Dari sekian banyak jenis sensor yang tersedia, penelitian ini memilih sensor jarak sebagai topik pembahasan. Dua jenis sensor jarak yang digunakan dalam penelitian ini adalah HC-SR04 dan GP2Y0A21YK. Kedua jenis sensor ini menggunakan cara kerja yang berbeda untuk

menentukan jarak. HC-SR04 menggunakan sinyal ultrasonik, sedangkan sensor GP2Y menggunakan sinar inframerah.

Sudah ada beberapa penelitian yang topik utamanya adalah sensor HC-SR04 dan GP2YA21YK dengan alat bantu analisis Wireshark dan Thingspeak. Diantaranya, penelitian yang dilaksanakan oleh Sorongan [5], penelitian ini membahas mengenai pemanfaatan sistem sensor HC-SR04 untuk memonitor tangki SPBU. Sebagai alat bantu analisis kinerja dari sistem sensor yang dikembangkan, penelitian ini menggunakan platform Thingspeak dan Virtuino. Hanya saja, penelitian [5] belum melakukan perbandingan uji kinerja antara sensor ultrasonik dengan sensor inframerah, dan juga tidak menggunakan Wireshark sebagai *tool* untuk mengukur QoS. Penelitian [6] telah melakukan analisa perbandingan kinerja sensor inframerah dan ultrasonik untuk pengukuran jarak, hanya saja penelitian ini belum menggunakan *tool* Thingspeak dan Wireshark sebagai alat bantu uji kinerja. Selanjutnya, Acharya [7] dan Pasha [8] telah menggunakan Thingspeak sebagai alat bantu penelitian di bidang IoT. Penelitian [7] menggunakan Thingspeak untuk memantau sistem irigasi dan sedangkan penelitian [8] menggunakannya untuk memantau kondisi lingkungan. Sementara itu, Lee, *et al.*

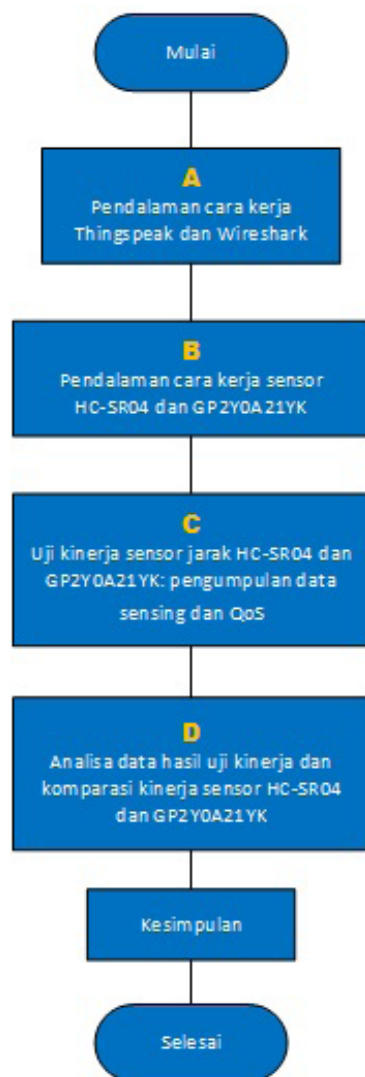


Gambar 1. *Internet of Things* dalam berbagai sektor (diolah dari kembali dari [3], [4])

[9] telah melaksanakan uji QoS untuk sistem berbasis protokol *Message Queuing Telemetry Transport* (MQTT) dengan menggunakan Wireshark. Namun penelitian [7–9] tersebut tidak menggunakan sensor jarak sebagai subjek penelitiannya.

Dengan memperhatikan penelitian-penelitian yang sudah dilakukan sebelumnya, maka penelitian ini akan mengkaji perbandingan kinerja sensor ultrasonik dan inframerah, dengan menggunakan bantuan 2 *tool*, yaitu Thingspeak dan Wireshark. ThingSpeak adalah layanan *platform* analitik IoT yang memungkinkan kita dapat mengumpulkan, memvisualisasikan, dan menganalisis aliran data langsung di-*cloud* [10], sedangkan Wireshark adalah merupakan tool penganalisa paket-paket yang dipertukarkan oleh berbagai macam protokol dalam jaringan [11].

Untuk selanjutnya penelitian ini disusun dengan urutan sebagai berikut: setelah pendahuluan pada Bagian I, Bagian II menjelaskan mengenai Metodologi Penelitian, selanjutnya Bagian III menjelaskan mengenai Hasil dan Pembahasan, dan ditutup pada Bagian IV dengan Kesimpulan.



Gambar 2. Metodologi penelitian

II. METODE

Penelitian ini menganalisis perbandingan kinerja dari 2 sensor jarak, yaitu: HC-SR04 dan GP2Y0A21YK, menggunakan *tool* Wireshark dan Thingspeak. Adapun langkah-langkah yang dilaksanakan dalam penelitian ini dapat dilihat pada Gambar 2. Berikut ini adalah deskripsi kegiatan yang akan dilaksanakan pada bagian A, B, C, D, dan E yang terdapat pada gambar tersebut.

Bagian A: Mendalami cara kerja masing-masing sensor jarak. Dengan mengetahui cara kerja masing-masing sensor tersebut, maka proses uji kinerja akan lebih tepat dilakukan dan analisa data dapat dilakukan dengan lebih objektif.

- Bagian B: Menjelaskan implementasi sistem sensor beserta arsitektur jaringannya, baik menggunakan HC-SR04 atau GP2Y0A21YK. Sistem sensor ini akan digunakan untuk menguji kinerja dua sensor jarak yang digunakan.
- Bagian C: Menjelaskan uji kinerja tiap-tiap sistem sensor dalam mendeteksi jarak yang terbentuk antara sensor dengan benda/halangan yang ada didepannya. Untuk menjelaskan kinerja, akan dilakukan analisis data terhadap hasil pengukuran tiap-tiap sistem sensor yang dikirimkan dan ditampilkan di Thingspeak.
- Bagian D: Menjelaskan uji kinerja tiap-tiap sistem sensor dalam proses pengiriman data ke Thingspeak. Uji kinerja menggunakan bantuan *tool* Wireshark. Adapun parameter QoS yang dikaji adalah *throughput*, *delay*, dan *packet loss*.
- Bagian E: Menjelaskan uji kebutuhan daya tiap-tiap sistem sensor. Untuk melakukan uji ini, penelitian menggunakan sensor arus INA219 agar data-data mudah dikumpulkan untuk selanjutnya dilakukan analisis.
- Kesimpulan: Setelah bagian A s/d E dikerjakan, maka kita dapat menarik kesimpulan, manakah yang terbaik dari kedua sistem sensor tersebut. Setelah kesimpulan didapat, maka penelitian ini dapat dinyatakan selesai.

III. HASIL DAN PEMBAHASAN

A. Pendalaman Cara Kerja Sensor HC-SR04

Sensor ultrasonik atau dinamakan sensor sonar adalah sensor yang menggunakan suara ultrasonik untuk mendeteksi objek yang berada dihadapannya dan dapat digunakan untuk menghitung jarak terhadap objek tersebut. Salah satu sensor ultrasonik yang populer adalah HC-SR04 seperti yang terlihat pada Gambar 3.

Pada sensor HC-SR04 terdapat 4 pin, seperti terlihat pada Gambar 4. Kegunaan masing-masing pin tersebut adalah sebagai berikut: VCC dihubungkan ke tegangan 5V DC. Trig dihubungkan ke pin digital Arduino. Pin inilah yang mengirimkan isyarat dalam bentuk gelombang suara berfrekuensi 40 kHz. Cukup dengan membuat

logika “HIGH - LOW” maka sensor akan memancarkan gelombang ultrasonik. Echo dihubungkan ke pin digital Arduino. Pin inilah yang bertanggungjawab untuk menerima gelombang yang dipantulkan oleh objek yang berada di depan sensor. Selama gelombang ultrasonik belum diterima, maka logika pin ECHO akan HIGH. Setelah gelombang ultrasonik diterima, maka pin ECHO akan berlogika LOW. GND dihubungkan ke ground.

Spesifikasi dari sensor HC-SR04 terlihat pada Tabel 1 dimana jangkauan jarak yang dapat dideteksi dengan baik oleh sensor HC-SR04 adalah 2–400 cm, dengan frekuensi kerja 40 kHz, dan arus yang dibutuhkan oleh sensor untuk bekerja sebesar 15 mA [12].

Untuk mengetahui jarak sebuah benda dengan menggunakan sensor HC-SR04, sebuah sinyal pulsa dengan durasi 10 μ s diinputkan pada pin Trigger sensor tersebut. Sebagai respon atas input trigger tersebut, *transducer* sensor mengirimkan 8 pulsa sinyal ultrasonik pada frekuensi 40 kHz. Saat *transducer* mengirimkan sinyalnya, pin ECHO berada dalam kondisi HIGH. Pin ECHO akan berubah menjadi LOW jika sensor mendeteksi pantulan dari 8 pulsa yang dikirimkan tersebut. Dengan demikian durasi pulsa yang diterima oleh pin ECHO dapat digunakan untuk menentukan jarak sebuah benda dengan sensor [13].

Gambaran dari proses deteksi jarak tersebut dapat dilihat pada Gambar 5 dan 6. Gambar 5 menunjukkan sebuah keadaan dimana pada jangkauan pengukuran sensor tidak terdapat benda dihadapannya. Sedangkan

pada Gambar 6, terdapat benda dihadapan sensor pada jarak yang masih dapat dideteksi oleh sensor. Pulsa selebar 38 ms menandakan tidak adanya benda dihadapan sensor [12], [13].

Jarak (J) dapat ditentukan dari durasi pulsa (D) menggunakan persamaan berikut dimana satuan J dalam cm:

$$J = \frac{0,034 \text{ cm/}}{2} \times D (\text{ s}). \quad (1)$$

Persamaan (1) tersebut diturunkan dari beberapa rumusan yang diwakili oleh (2), (3) dan (4). Adapun rumusan-rumusan tersebut adalah sebagai berikut:

- Rumus yang menghubungkan parameter jarak (s), kecepatan (v), dan waktu (t):

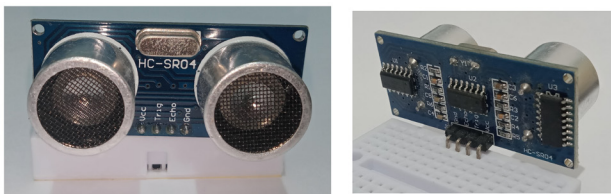
$$s = v \cdot t, \quad (2)$$

dimana s adalah jarak bolak-balik sensor $\rightarrow$ benda $\rightarrow$ sensor dalam cm, v kecepatan rambat bunyi di udara bebas sebesar 340 m/s atau 0,034 cm/ μ s, dan t merupakan durasi pulsa (D) dalam μ s.

- Maka jarak bolak-balik sensor $\rightarrow$ benda $\rightarrow$ sensor sebesar: $s = 0,034 \text{ cm/}\mu\text{s} \times D (\mu\text{s})$. Dan jarak dari

Tabel 1. Spesifikasi Sensor HC-SR04 [12]

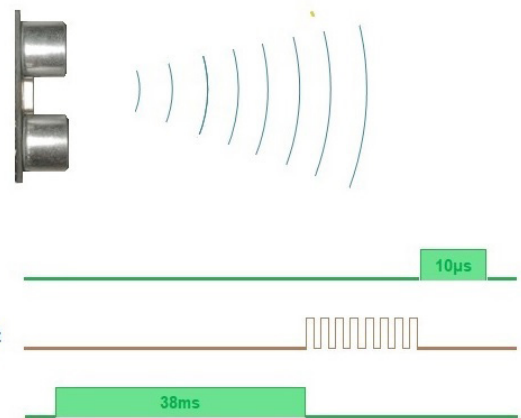
Parameter	Nilai
Working Voltage	DC 5 V
Working Current	15 mA
Working Frequency	40 kHz
Max Range	400 cm
Min Range	2 cm
Measuring Angle	15o



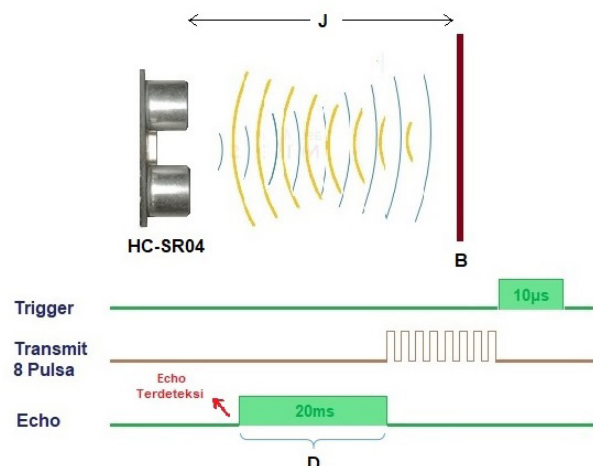
Gambar 3. Sensor jarak HC-SR04



Gambar 4. Pin-pin pada sensor HC-SR04



Gambar 5. Sensor jarak tanpa benda pada jangkauan pengukuran (diolah dari kembali dari [13])



Gambar 6. Sensor dengan benda pada jangkauan pengukuran (Diolah dari kembali dari [13])

sensor ke benda adalah $J = s/2$, karena durasi pulsa merupakan waktu perjalanan sinyal ultrasonik secara bolak-balik antara sensor dengan benda.

- Dengan demikian (1) dapat ditulis kembali menjadi:

$$J = \frac{S}{2} D. \quad (3)$$

Jadi jika durasi pulsa sebesar 23.529,4 μs , maka jarak antara sensor dengan benda adalah:

$$J = \frac{0,034 \frac{\text{cm}}{\mu s} \times D(\mu s)}{2} = \frac{0,034 \frac{\text{cm}}{\mu s} \times 23.529,4 \mu s}{2} \cong \frac{800}{2} = 400 \text{ cm}. \quad (4)$$

B. Pendalaman Cara Kerja Sensor GP2Y0A21YK

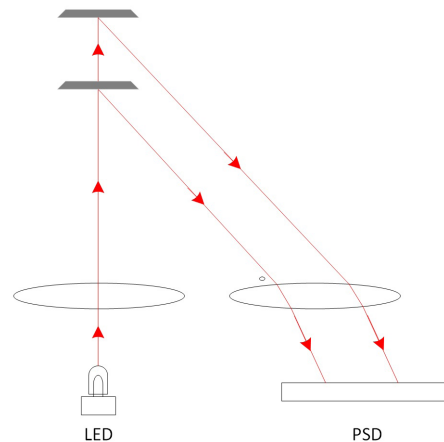
Sensor GP2Y0A21YK, untuk selanjutnya diringkas GP2Y, merupakan sensor jarak yang menggunakan sinar inframerah sebagai sarana untuk pengukuran jarak. Sensor ini menggunakan tegangan antara 4,5 V hingga 5,5 V. Jarak yang dapat diukur antara 10 cm hingga 80 cm. Pengukuran dilakukan berdasarkan pantulan inframerah oleh objek yang berada di depan sensor [14]. Tampilan fisik dari sensor inframerah GP2Y, diperlihatkan pada Gambar 7.

Untuk mengukur jarak sebuah objek, sensor GP2Y menggunakan LED Inframerah (IR) sebagai pemancar sinar inframerah dan sebuah position-sensible photo detector (PSD) sebagai pendeteksi jarak. Gambar 8 memperlihatkan bahwa pada masing-masing perangkat terdapat lensa yang berfungsi untuk memfokuskan sinar. Terlihat pada Gambar 9, seberkas cahaya yang sempit ($\lambda = 870 \pm 70 \text{ nm}$) [15] akan dipancarkan oleh LED kemudian setelah terpantul dari objek P1 atau P2, berkas tersebut akan

diarahkan oleh lensa kedua untuk mendarat pada PSD di titik tertentu. PSD akan memberikan tegangan output yang berbeda tergantung pada posisi jatuhnya sinar. Gambar 9 memperlihatkan, objek P1 dan P2 akan memberikan titik jatuh yang berbeda pada PSD untuk sumber cahaya yang sama yang telah dipancarkan oleh IR LED.

Sensor GP2Y akan memberikan tegangan output yang berbanding terbalik terhadap jarak yang diukur, sehingga ketika jarak ukur meningkat, maka tegangan yang dihasilkan sensor akan menurun. Gambar 10 memperlihatkan relasi jarak terhadap tegangan tersebut. Terlihat pada gambar tersebut, jarak 6,25 cm ditampilkan secara khusus karena memberikan nilai tegangan output tertinggi yang dihasilkan sensor. Ketika jarak lebih kecil atau lebih besar dari 6,25 cm, nilai tegangan output akan menurun.

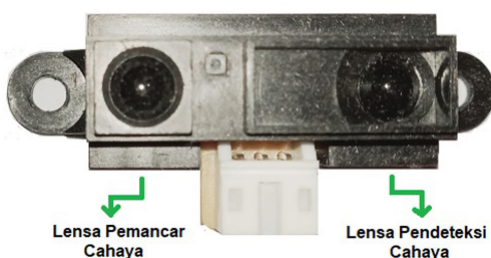
Semua sensor memiliki rentang pengukuran tertentu dimana hasil pengukuran tersebut dapat digunakan. Pada sensor GP2Y, terdapat nilai maksimum dan minimum untuk jarak yang dapat diukur. *Datasheet* sensor GP2Y [16] telah menentukan bahwa jarak yang dapat diukur oleh sensor berada pada rentang 10 – 80 cm. Dengan demikian, jarak objek dengan sensor, tidak boleh terlalu dekat atau terlalu jauh. Untuk rentang sensor mulai dari 10 – 80 cm, tegangan *output* (V) pada Gambar 10 dapat didekati dengan persamaan berikut:



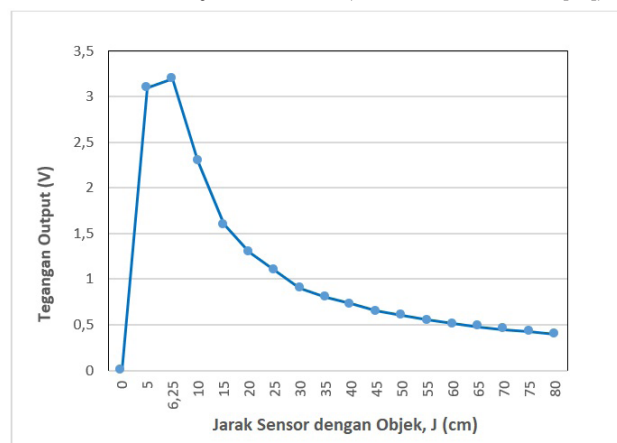
Gambar 9. Cara Kerja Sensor GP2Y (diolah dari kembali dari [15])



Gambar 7. Sensor Jarak GP2Y



Gambar 8. Fungsi Lensa-Lensa Pada Sensor GP2Y



Gambar 10. Gambar Karakteristik Sensor GP2Y (diolah dari kembali dari [16], [14])

$$V = 15,755J^{-0,836}, \quad (5)$$

dimana V adalah tegangan *output* sensor dalam volt. Maka, untuk mengkonversi tegangan keluaran sensor GP2Y menjadi jarak, kita tinggal memodifikasi rumus di atas menjadi:

$$J = 27,047V^{-1,196}. \quad (6)$$

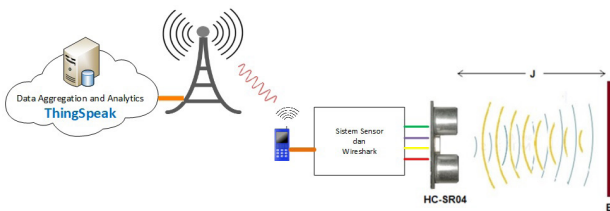
Rumus inilah yang dapat digunakan oleh mikrokontroler untuk menentukan jarak berdasarkan tegangan yang masuk dari sensor GP2Y.

C. Implementasi Sistem untuk Uji Kinerja Sensor

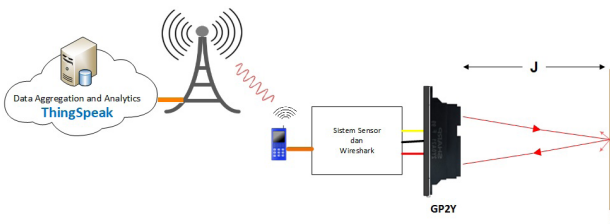
Gambar 11 dan Gambar 12 berturut-turut memperlihatkan sistem pengukuran jarak menggunakan sensor HC-SR04 (ultrasonik) dan GP2Y (inframerah). Terlihat pada kedua gambar tersebut, sebuah penghalang terletak didepan sensor pada jarak J (cm). Kedua sistem sensor tersebut terhubung pada *cloud* Thingspeak dengan menggunakan komunikasi data menggunakan jaringan internet via sistem seluler.

Gambar 13 memperlihatkan pengkabelan antara sensor HC-SR04 dengan Arduino dilengkapi modul ESP8622-01 untuk transmisi data ke Thingspeak. Sedangkan pengkabelan menggunakan sensor GP2Y diperlihatkan pada Gambar 14.

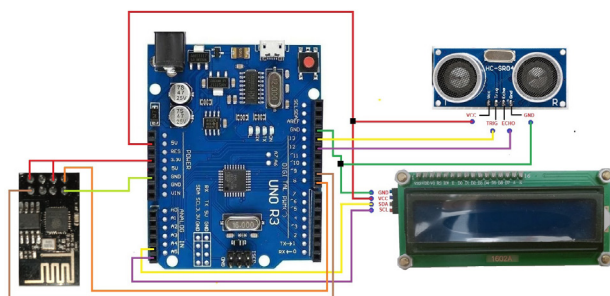
Untuk menguji kinerja dari sensor jarak ini, penelitian



Gambar 11. Sistem sensor menggunakan HC-SR04



Gambar 12. Sistem sensor menggunakan GP2Y

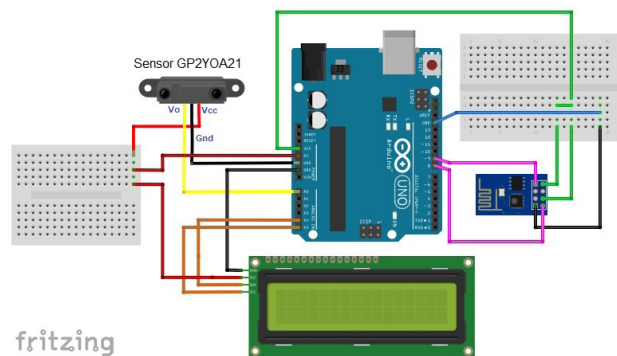


Gambar 13. Pengkabelan sistem sensor menggunakan HC-SR04 dan Arduino

ini menggunakan skenario-skenario sebagai berikut: 1) Menguji hasil pembacaan sensor untuk jarak baca yang berubah-ubah, 2) Menguji QoS sistem sensor dalam hal pengiriman data, dan 3) Menguji Kebutuhan Daya Sistem Sensor

D. Uji Pembacaan Jarak (J) oleh Sistem Sensor

Pada pengujian ini, besar J dimulai dari 10 cm, lalu meningkat secara bertahap dengan inkremen sebesar 10 cm. Hasil pengujian dapat dilihat pada Gambar 15 dan 16, dimana kedua gambar tersebut berasal dari data hasil pembacaan sensor yang dikirimkan ke *cloud* Thingspeak. Gambar 15 merupakan bentuk visualisasi pembacaan data dengan perioda pengukuran 20 s, sedangkan Gambar 16 merupakan data rata-rata dari pembacaan sensor yang melakukan *sensing* setiap 20 s. Terlihat pada gambar tersebut, sensor HC-SR04 memberikan hasil pembacaan



Gambar 14. Pengkabelan sistem sensor menggunakan GP2Y dan Arduino



Gambar 15. Tampilan grafis pada Thingspeak

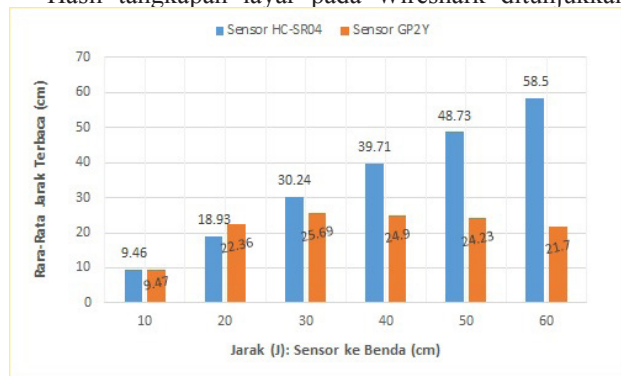
jarak yang lebih baik, grafik pada Gambar 15 berbentuk seperti tangga yang meningkat dimana hal tersebut sesuai dengan jarak ukur (J) yang semakin meningkat. Sementara itu, hasil pembacaan sensor GP2Y menunjukkan hasil yang tidak konsisten, grafik cenderung tidak konsisten dan tidak berbentuk tangga.

Data pada Gambar 16 memperjelas hasil kinerja dua sensor tersebut, rata-rata jarak yang terbaca oleh sensor ultrasonik menunjukkan hasil yang mendekati dari jarak J , sedangkan hasil pembacaan sensor GP2Y menunjukkan ketidakakuratan yang meningkat pada jarak yang meningkat. Rata-rata galat pembacaan pada sistem sensor HC-SR04 sebesar 0,82 cm, sedangkan pada sistem berbasis GP2Y sebesar 14,40 cm. Hasil sensing jarak pada sistem sensor GP2Y masih cukup akurat untuk jarak maksimum 20 cm, sedangkan mulai 30 – 60 cm hasil pembacaan sensor cenderung konstan di angka ± 24 cm, bahkan pada jarak 60 cm hasil pembacaan sensor sangat tidak akurat, yaitu sebesar 21,7 cm.

Hasil pembacaan sensor ultrasonik HC-SR04 lebih akurat dibandingkan sensor inframerah GP2Y disebabkan metode *sensing* pada HC-SR04 akan menghasilkan lebih sedikit *bias* dibandingkan metode pada GP2Y. Jika merujuk pada sub.bab III.A., pengukuran jarak pada HC-SR04 hanya bergantung pada parameter D (durasi pulsa), ketika echo diterima maka parameter D langsung dapat ditentukan. Setelah nilai D diketahui maka jarak dari sensor ke benda (J) dapat ditentukan dengan persamaan sederhana. Sementara itu, pada sensor GP2Y, penentuan jarak lebih rumit, hasil *sensing* bergantung pada beberapa parameter yang berpotensi menimbulkan bias.

A. Uji QoS Sistem Sensor dalam Pengiriman Data

Hasil tangkapan layar pada Wireshark ditunjukkan



Gambar 16. Rata-rata hasil pembacaan sensor

pada Gambar 17, dimana data tersebut terdapat sejumlah data yang dapat digunakan untuk meneliti QoS jaringan. Data dengan lingkaran berwarna merah adalah sepasang data yang diperlukan untuk menentukan tiap-tiap data *throughput* dan *delay*.

Data dalam lingkaran merah pada kolom *length* menunjukkan panjang paket dalam *byte*, sedangkan data yang berada dibawah kolom *time since previous frame in this TCP stream*, merupakan data *round-trip time*. Data *round-trip time* bersesuaian dengan dua data dalam lingkaran biru, karena selisih dari dua data dalam lingkaran biru adalah nilai *round-trip time* untuk tiap-tiap pengiriman data.

Dari Wireshark dapat diperoleh beberapa parameter QoS, seperti: *packet loss*, *throughput*, *delay*, dan *jitter*. Penelitian ini mengamati 3 parameter QoS, yaitu: *packet loss*, *delay*, dan *throughput*. Gambar 18 dan Gambar 19 masing-masing menunjukkan data *packet loss* seperti yang ditunjukkan pada *tool* Wireshark.

Terlihat pada Gambar 18 dan Gambar 19 di atas bahwa *packet loss* untuk kedua sistem sensor adalah 0%. Hal tersebut menunjukkan bahwa paket-paket data yang dikirimkan oleh sistem sensor dapat diterima dengan baik oleh server Thingspeak.

Perbandingan data *delay* untuk pengiriman data dari dua sistem sensor ke *platform* Thingspeak ditunjukkan Gambar 20. Pada penelitian ini, data *delay* merupakan data aproksimasi *end-to-end delay*. Nilai *end-to-end delay* dapat didekati oleh persamaan berikut:

$$\text{Aproksimasi end-to-end delay} = \frac{\text{round-trip time}}{2}. \quad (7)$$

Terlihat pada Gambar 20 nilai *delay* rata-rata pada kedua sistem sensor berada pada level puluhan *milisecon*, namun besar *delay* pada sistem sensor HC-SR04 sedikit lebih besar dibandingkan *delay* pada sistem yang menggunakan GP2Y. Rerata *delay* pada sistem HC-SR04 sebesar 33,55 ms, sedangkan pada sensor GP2Y sebesar 26,1 ms. Dengan demikian, sistem sensor dengan sensor GP2Y memiliki *delay* lebih kecil dengan selisih sebesar 22,2 % atau sekitar 7,45 ms. Sementara itu, untuk parameter *throughput*, Sedangkan perbandingan data kinerja *throughput* untuk kedua sistem sensor ditunjukkan pada Gambar 21. Terlihat rata-rata *throughput* untuk kedua sistem sensor besarnya hampir sama. Rerata *throughput* sistem sensor GP2Y besarnya 38,12 kbps, sedikit lebih

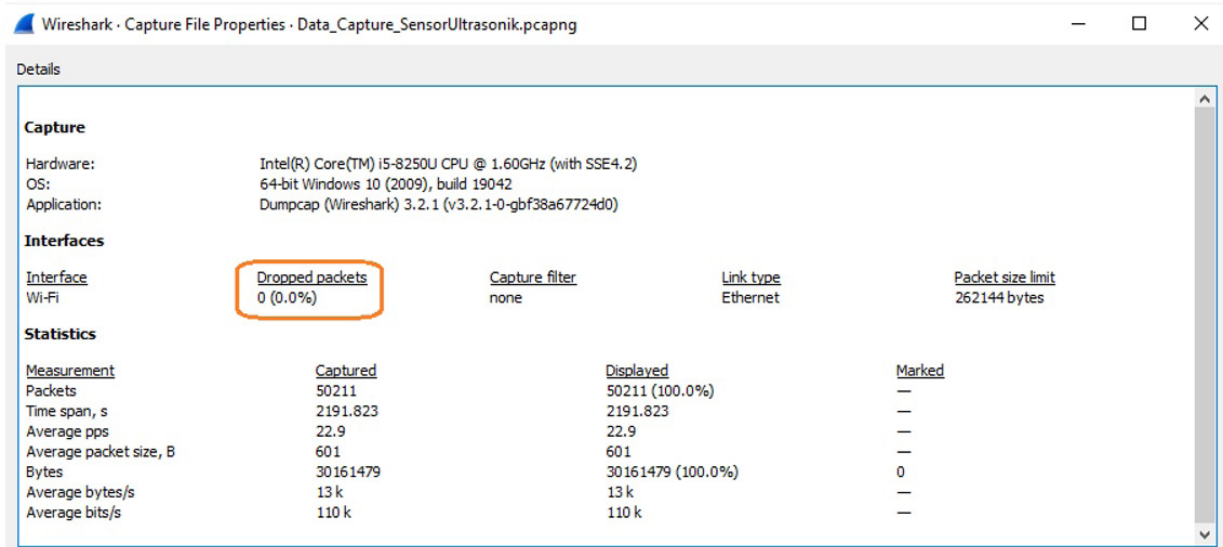
No.	Time	Source	Destination	Protocol	Length	Time since previous frame in this TCP stream	Info
77	10:20:36,013678	192.168.43.230	184.106.153.149	TCP	58	0.000000000	62099 → 80 [SYN] Seq=0 Win=2920 Len=0 MSS=1460
78	10:20:36,074958	184.106.153.149	192.168.43.230	TCP	58	0.061280000	80 → 62099 [SYN, ACK] Seq=0 Ack=1 Win=4200 Len=0 MSS=1400
79	10:20:36,076996	192.168.43.230	184.106.153.149	TCP	54	0.002038000	62099 → 80 [ACK] Seq=1 Ack=1 Win=2920 Len=0
82	10:20:37,207966	192.168.43.230	184.106.153.149	HTTP	120	1.130970000	GET /update?api_key=WKQ021DF4EA7M7BP&field1=9.42&field2=145.90
83	10:20:37,279576	184.106.153.149	192.168.43.230	TCP	54	0.071610000	80 → 62099 [ACK] Seq=1 Ack=67 Win=4266 Len=0
84	10:20:37,726731	184.106.153.149	192.168.43.230	TCP	55	0.447155000	80 → 62099 [PSH, ACK] Seq=1 Ack=67 Win=4266 Len=1
85	10:20:37,727076	184.106.153.149	192.168.43.230	TCP	54	0.000345000	80 → 62099 [FIN, ACK] Seq=2 Ack=67 Win=4266 Len=0
86	10:20:37,781200	192.168.43.230	184.106.153.149	TCP	54	0.054124000	62099 → 80 [FIN, ACK] Seq=67 Ack=3 Win=2918 Len=0
87	10:20:37,834783	184.106.153.149	192.168.43.230	TCP	54	0.053583000	80 → 62099 [ACK] Seq=3 Ack=68 Win=4266 Len=0
1.	10:20:56,301146	192.168.43.230	184.106.153.149	TCP	58	0.000000000	62100 → 80 [SYN] Seq=0 Win=2920 Len=0 MSS=1460

Gambar 17. Data pada *Wireshark* untuk perhitungan *delay* dan *throughput* sistem sensor

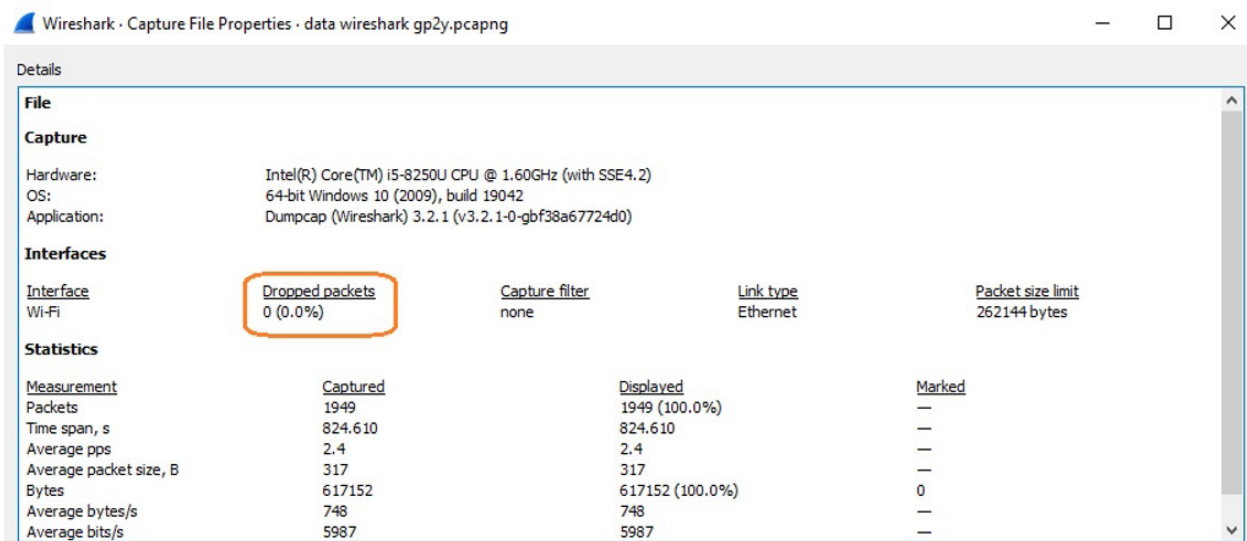
besar dibandingkan pada sistem sensor HC-SR04 yang besarnya 37,01 kbps. Sistem dengan sensor GP2Y memiliki rerata *throughput* lebih besar 2,9% dibandingkan sistem dengan HC-SR04. Selisih rata-rata *throughput* untuk kedua sistem sensor sebesar 1,11 kbps.

Data QoS memang menunjukkan sistem sensor

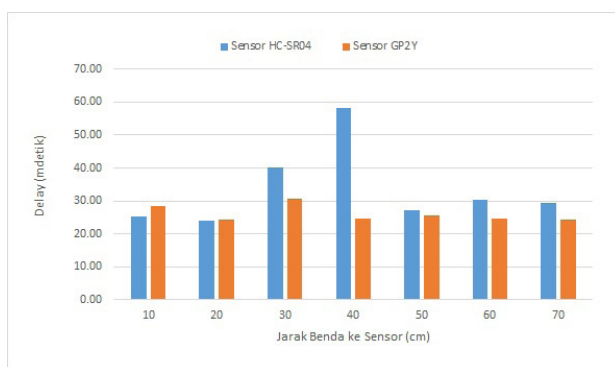
GP2Y sedikit lebih baik dibandingkan sistem HC-SR04, tetapi perbedaan tersebut tidak terlalu signifikan. Pada komunikasi data yang menggunakan internet, sedikit variasi pada nilai *throughput* dan *delay* sangatlah wajar, hal tersebut terjadi karena kondisi trafik di internet yang sangatlah dinamis. Bahkan menurut referensi [17], pada



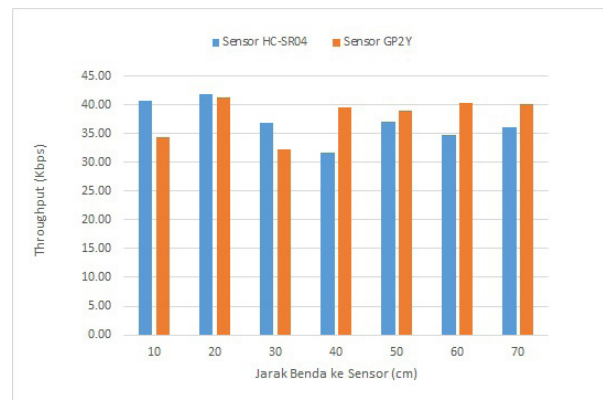
Gambar 18. Packet loss pada sistem sensor menggunakan HC-SR04



Gambar 19. Packet loss pada sistem sensor menggunakan GP2Y



Gambar 20. Perbandingan delay untuk kedua sensor



Gambar 21. Perbandingan throughput untuk kedua sensor

komunikasi data selain *voice* dan *video*, batasan *throughput* dan *delay* tidaklah ketat.

E. Uji Kebutuhan Daya pada Sistem Sensor

Pada penelitian ini, kebutuhan daya sistem sensor diwakili oleh besar arus yang ditarik oleh sistem. Untuk mengukur besar arus yang masuk ke dalam sistem sensor jarak, sensor arus INA219 diletakkan pada titik masuk catudaya. Nilai yang ditunjukkan oleh INA219 arus berupa besaran arus dalam satuan mA. Gambar 22 menunjukkan instalasi power supply untuk sistem sensor yang digunakan pada penelitian ini, baik menggunakan HC-SR04 maupun GP2Y, dimana pada sistem catu daya ini sudah terpasang sensor arus INA219.

Kebutuhan arus untuk kedua jenis sensor pada Thingspeak dapat diilustrasikan pada Gambar 23. Terlihat bahwa arus yang ditarik oleh rangkaian sensor HC-SR04 lebih jelas dan teratur dibandingkan arus pada sensor GP2Y. Sementara itu, data pada Tabel 2 menunjukkan bahwa rata-rata arus yang ditarik oleh sistem sensor HC-SR04 pada tiap-tiap jarak mayoritas lebih kecil dibandingkan arus yang dibutuhkan oleh sistem sensor GP2Y. Sistem sensor jarak yang menggunakan GP2Y membutuhkan

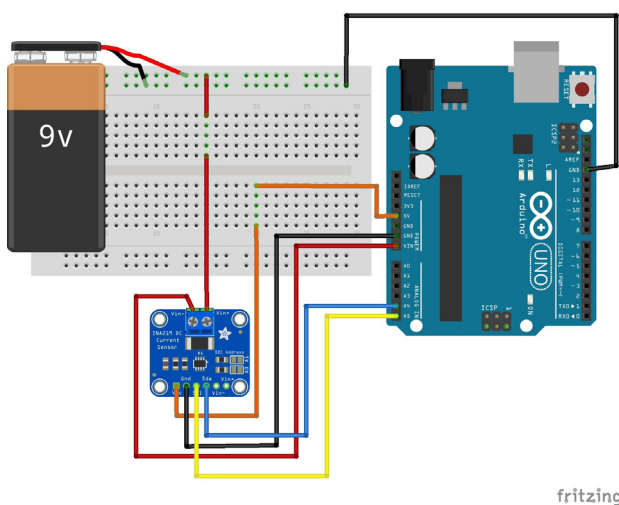
daya 14,36% lebih besar dibandingkan sistem dengan HC-SR04.

IV. KESIMPULAN

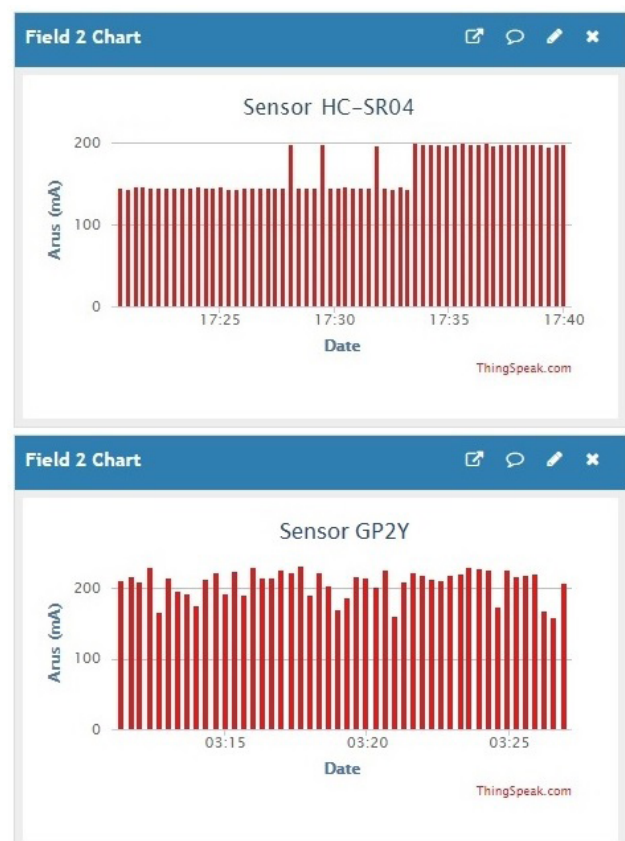
Penelitian ini telah memperlihatkan bahwa Thingspeak dan Wireshark merupakan dua *tool* yang sangat membantu dalam mengukur dan menganalisis kinerja dua sistem sensor berbasis IoT. Hasil uji kinerja menunjukkan bahwa dalam hal ketepatan menentukan jarak, sensor HC-SR04 memiliki kinerja yang jauh lebih baik dibandingkan sensor GP2Y. Hal tersebut terlihat pada rata-rata galat deteksi jarak oleh sensor HC-SR04 yang jauh lebih kecil dibandingkan sensor GP2Y. Pada HC-SR04, rata-rata galat pembacaan sebesar 0,82 cm, sedangkan sensor GP2Y menunjukkan angka sebesar 14,40 cm. Sementara itu, ditinjau dari parameter QoS, kedua sistem sensor menunjukkan kinerja yang hampir sepadan, yaitu *packet loss* sama-sama 0%, *throughput* pada HC-SR04 sebesar 37,01 kbps dan pada GP2Y sebesar 38,12 kbps. Sedangkan untuk *delay*, sistem dengan HC-SR04 memberikan *delay* sebesar 33,55 ms, dan pada GP2Y sebesar 26,1 ms. Selanjutnya, berdasarkan kebutuhan daya, sistem sensor yang menggunakan HC-SR04 mengkonsumsi daya 14,36% lebih kecil. Dengan merujuk pada hasil-hasil pengukuran dan visualisasi menggunakan Wireshark dan Thingspeak, dapat disimpulkan bahwa sistem pendeteksi jarak menggunakan sensor HC-SR04 lebih baik dibandingkan sistem dengan sensor GP2YA21YK.

Tabel 2. Kebutuhan Energi pada Rangkaian Sensor Jarak

Jarak Sensor ke Benda (J) (cm)	Rerata Konsumsi Arus pada Sistem Sensor HC-SR04 (mA)	Rerata Konsumsi Arus pada Sistem Sensor GPY2 (mA)
10	145.60	207.95
20	152.27	198.50
30	170.43	204.09
40	198.84	212.06
50	198.67	198.13
60	198.57	196.61
Rata-rata	177.40	202.89



Gambar 22. Sistem power supply dan penempatan sensor arus pada sistem sensor



Gambar 23. Hasil pengukuran arus pada Thingspeak

REFERENSI

- [1] L. Atzori, A. Iera, and G. Morabito, "Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm," *Ad Hoc Networks*, vol. 56, pp. 122–140, 2017.
- [2] S. Li, L. Da Xu, and S. Zhao, "The internet of things: a survey," *Inf. Syst. Front.*, vol. 17, no. 2, pp. 243–259, 2015.
- [3] A. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, "Internet of things: A survey on enabling technologies, protocols, and applications," *IEEE Commun. Surv. tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.
- [4] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions," *Futur. Gener. Comput. Syst.*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [5] E. Sorongan, Q. Hidayati, and K. Priyono, "ThingSpeak sebagai Sistem Monitoring Tangki SPBU Berbasis Internet of Things," *JTERA (Jurnal Teknol. Rekayasa)*, vol. 3, no. 2, p. 219, 2018.
- [6] R. T. Yunardi, "Analisa Kinerja Sensor Inframerah dan Ultrasonik untuk Sistem Pengukuran Jarak pada Mobile Robot Inspection," *Setrum Sist. Kendali-Tenaga-elektronika-telekomunikasi-komputer*, vol. 6, no. 1, p. 33, 2017.
- [7] C. Acharya and S. Kuzhalvaimozhi, "Irrigation and Internet of Things Platform," *Int. Research J. of Engineering and Technol.*, pp. 1643–1646, 2016.
- [8] S. Pasha, "Thingspeak Based Sensing and Monitoring System for IoT with Matlab Analysis," *Int. J. New Technol. Res.*, vol. 2, no. 6, pp. 19–23, 2016.
- [9] S. Lee, H. Kim, D. K. Hong, and H. Ju, "Correlation analysis of MQTT loss and delay according to QoS level," in *Proc. Int. Conf. Inf. Netw.*, Jan. 2013, pp. 714–717.
- [10] Thingspeak. (view July 2021). Learn more about ThingSpeak. [Online]. Available: https://thingspeak.com/pages/learn_more.
- [11] L. Bock, *Learn Wireshark: Confidently navigate the Wireshark interface and solve real-world networking problems*. Packt Publishing Ltd, 2019.
- [12] Indoware. (view July 2021). Ultrasonic ranging module HC - SR04. [Online]. Available: <http://www.micropik.com/PDF/HCSR04.pdf>.
- [13] L. M. Engineers (view July 2021). How HC-SR04 ultrasonic sensor works & interface it with arduino. [Online]. Available: <https://lastminuteengineers.com/arduino-sr04-ultrasonic-sensor-tutorial/>.
- [14] Taghipour, J. "The Effect of Filtering on Proximity Sensors." *PhD diss.*, California State Polytechnic University, Pomona, 2018.
- [15] de Bakker, B. (view July 2021). How to use a SHARP GP2Y0A21YK0F IR distance sensor with Arduino. [Online]. Available: <https://www.makerguides.com/sharp-gp2y0a21yk0f-ir-distance-sensor-arduino-tutorial/>.
- [16] Alldatasheet (view July 2021). GP2Y datasheet (pdf) sharp electronic components. [Online]. Available: <https://www.alldatasheet.com/datasheet-pdf/pdf/412635/SHARP/GP2Y0A21YK0F.html>.
- [17] ITU-T, "G.1010: End-user multimedia QoS categories," *Int. Telecommun. Union*, vol. 1010, 2001, [Online]. Available: http://scholar.google.com.au/scholar?hl=en&q=ITU-T+Recommendation+G.1010&btnG=&as_sdt=1,5&as_sdtp=#7.