

Analisis Komparatif Konsumsi Daya Baterai pada Perangkat IoT Menggunakan Protokol Komunikasi MQTT dan HTTP

Riza Mirza¹, Khairuni²

^{1,2}Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Almuslim
Email: mirza@transdi.or.id

ABSTRAK

Internet of Things (IoT) seringkali mengandalkan perangkat bertenaga baterai yang menuntut efisiensi daya tinggi. Protokol komunikasi memainkan peran penting dalam menentukan masa pakai baterai perangkat. Penelitian ini bertujuan untuk melakukan analisis komparatif konsumsi daya antara dua protokol yang umum digunakan: MQTT (*Message Queuing Telemetry Transport*) dan HTTP (*Hypertext Transfer Protocol*). Dua perangkat identik berbasis mikrokontroler ESP32 dengan sensor suhu DHT11 diimplementasikan untuk mengirimkan data secara periodik ke server. Satu perangkat menggunakan MQTT, sementara yang lain menggunakan HTTP POST. Pengukuran tegangan baterai dilakukan secara berkala selama 48 jam untuk memantau konsumsi daya. Hasil penelitian menunjukkan bahwa perangkat yang menggunakan protokol MQTT menunjukkan konsumsi daya yang secara signifikan lebih rendah, dengan penurunan tegangan baterai yang lebih lambat sebesar 25% dibandingkan dengan perangkat berbasis HTTP. Temuan ini memberikan bukti kuantitatif bahwa MQTT merupakan pilihan yang lebih superior untuk aplikasi IoT yang berfokus pada efisiensi energi.

Kata kunci: Efisiensi Energi, ESP32, HTTP, Internet of Things, Konsumsi Daya, dan MQTT

Penulis koresponden : Riza Mirza

Tanggal terbit : 15 Juni 2025

Tautan : <https://jurnal.komputasi.org/index.php/jst/article/view/35>

1. PENDAHULUAN

Perkembangan pesat Internet of Things (IoT) telah memungkinkan implementasi sistem cerdas di berbagai domain, mulai dari rumah pintar, kota pintar, hingga pertanian presisi [1]. Banyak juga perangkat IoT seperti sensor lingkungan, perangkat pemantauan jarak jauh, dan sistem keamanan, dirancang untuk bekerja dalam jangka waktu lama dengan sumber daya terbatas, khususnya baterai. Oleh karena itu, pemilihan protokol komunikasi yang hemat daya sangat penting untuk memperpanjang masa pakai perangkat. Sebagian besar perangkat IoT, terutama sensor node yang tersebar di lapangan, beroperasi menggunakan sumber daya baterai yang terbatas, dengan efisiensi energi menjadi salah satu tantangan utama dalam perancangan dan implementasi sistem IoT yang berkelanjutan [2].

Masa pakai baterai perangkat sangat dipengaruhi oleh berbagai faktor, termasuk perangkat keras yang digunakan, frekuensi transmisi data, dan yang tidak kalah penting, protokol komunikasi yang dipilih [3].

Dua protokol komunikasi yang umum digunakan dalam sistem IoT adalah MQTT (*Message Queuing Telemetry Transport*) dan HTTP (*Hypertext Transfer Protocol*). MQTT dikenal sebagai protokol ringan yang dirancang untuk perangkat dengan sumber daya terbatas, sedangkan HTTP merupakan protokol berbasis request-response yang lebih umum digunakan di aplikasi web. Meskipun keduanya mampu mentransmisikan data antar perangkat, perbedaan arsitektur dan overhead data yang dibawa dapat berdampak langsung terhadap konsumsi daya perangkat.

HTTP yang lebih dikenal sebagai tulang punggung web, dikenal luas dan mudah diimplementasikan, namun memiliki overhead yang cukup besar karena format header-nya yang verbose dan sifatnya yang request-response tanpa koneksi tetap [4]. Di sisi lain, MQTT adalah protokol publish-subscribe yang dirancang khusus untuk lingkungan dengan keterbatasan bandwidth dan daya. Protokol ini memiliki header yang minimalis dan menjaga koneksi tetap terbuka, yang secara teoretis lebih efisien untuk transmisi data periodik [5].

Meskipun banyak literatur yang menyatakan keunggulan teoretis MQTT dalam hal efisiensi, masih terdapat kebutuhan untuk studi komparatif berbasis data empiris yang mengukur dampak langsung pemilihan protokol terhadap konsumsi daya baterai dalam skenario yang terkontrol.

Beberapa penelitian sebelumnya telah menunjukkan pentingnya pemilihan protokol komunikasi dalam memengaruhi efisiensi daya pada perangkat IoT. Almeida, Rodrigues, dan Saleem (2023) melakukan analisis terhadap konsumsi daya antara protokol MQTT dan HTTP menggunakan platform NodeMCU berbasis ESP8266. Hasil penelitian tersebut menunjukkan bahwa MQTT memiliki konsumsi daya yang lebih rendah dibandingkan HTTP, dengan selisih penggunaan daya mencapai 6–8 persen. Selain itu, estimasi daya tahan baterai juga menunjukkan bahwa perangkat dengan protokol MQTT memiliki masa pakai yang lebih lama dibandingkan HTTP[6].

Penelitian serupa juga dilakukan oleh Masdani dan Darlis (2018) yang membandingkan efisiensi protokol MQTT, HTTP, dan CoAP pada perangkat IoT. Mereka menyimpulkan bahwa MQTT merupakan protokol yang paling efisien dalam hal konsumsi daya karena karakteristiknya yang ringan dan tidak membutuhkan proses komunikasi yang kompleks seperti HTTP. Hal ini menjadikan MQTT lebih cocok digunakan pada perangkat dengan sumber daya terbatas seperti sensor nirkabel dan node IoT[7].

Sementara itu, Nugraha, Firmansyah, dan Gunawan (2024) membandingkan penerapan MQTT dan HTTP pada sistem monitoring hidroponik berbasis IoT. Penelitian ini menemukan bahwa MQTT tidak hanya lebih efisien dalam konsumsi daya, tetapi juga lebih hemat dalam penggunaan sumber daya prosesor serta memiliki waktu pengiriman data yang lebih cepat dibandingkan HTTP. Ketiga studi tersebut secara konsisten menunjukkan bahwa protokol MQTT lebih unggul dalam aspek efisiensi energi dibandingkan HTTP, menjadikannya pilihan yang lebih tepat untuk pengembangan perangkat IoT bertenaga baterai[8].

Penelitian-penelitian sebelumnya telah membahas efisiensi daya dari berbagai protokol komunikasi dalam sistem IoT, di antaranya yang dilakukan oleh Almeida, Rodrigues, dan Saleem (2023) yang membandingkan MQTT dan HTTP menggunakan platform NodeMCU. Hasilnya menunjukkan bahwa MQTT memiliki konsumsi daya yang lebih rendah hingga 8% dibandingkan HTTP, dengan estimasi masa pakai baterai yang juga lebih panjang. Studi lain oleh Masdani dan Darlis (2018) mendukung temuan tersebut, menyatakan bahwa MQTT lebih hemat energi karena karakteristiknya yang ringan dan efisien untuk perangkat dengan sumber daya terbatas. Selain itu, Nugraha dkk. (2024) melakukan pengujian dalam konteks sistem monitoring hidroponik dan menemukan bahwa MQTT tidak hanya lebih hemat daya, tetapi juga lebih cepat dalam pengiriman data serta lebih stabil dalam penggunaan sumber daya prosesor.

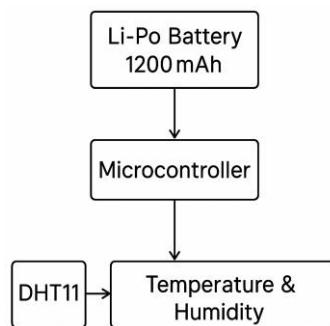
Namun, sebagian besar studi terdahulu masih terbatas pada pengukuran daya yang bersifat umum, tidak secara rinci menyajikan data kuantitatif perubahan tegangan baterai dari perangkat identik dalam periode operasional nyata yang berkelanjutan. Penelitian ini bertujuan untuk mengisi celah tersebut dengan melakukan analisis komparatif kuantitatif terhadap konsumsi daya baterai pada perangkat IoT identik yang mentransmisikan data menggunakan protokol MQTT dan HTTP. Nilai kebaruan dari penelitian ini adalah penyediaan data empiris yang dapat menjadi acuan bagi para pengembang dalam memilih protokol komunikasi yang paling efisien untuk aplikasi IoT hemat daya, khususnya pada sistem berbasis baterai yang ditujukan untuk pemantauan jangka panjang.

2. METODELOGI

Penelitian ini menggunakan metode eksperimental untuk membandingkan konsumsi daya dua sistem. Tahapan penelitian diuraikan sebagai berikut:

2.1. Perancangan Arsitektur Sistem

Sistem terdiri dari dua sensor node identik, satu MQTT broker, dan satu web server. Setiap node terdiri dari mikrokontroler ESP32, sensor suhu dan kelembaban DHT11, dan ditenagai oleh baterai Li-Po berkapasitas 1200mAh. Arsitektur sistem dapat dilihat pada Gambar 1



Gambar 1 Arsitektur Sistem Pengukur Suhu dan Kelembaban

Gambar di atas menunjukkan diagram blok sistem pengukuran suhu dan kelembaban berbasis sensor DHT11 yang ditenagai oleh baterai Li-Po 1200 mAh. Penjelasan tiap blok adalah sebagai berikut:

1. Li-Po Battery 1200 mAh

Bertindak sebagai sumber daya utama sistem, memberikan tegangan dan arus yang dibutuhkan untuk mengoperasikan mikrokontroler dan sensor.

2. Mikrocontroller

Berfungsi sebagai pusat kendali sistem. Mikrokontroler menerima data digital dari sensor DHT11 dan memprosesnya untuk ditampilkan atau dikirim ke media output lainnya.

3. DHT11

Merupakan sensor suhu dan kelembaban digital yang mengirimkan data lingkungan ke mikrokontroler.

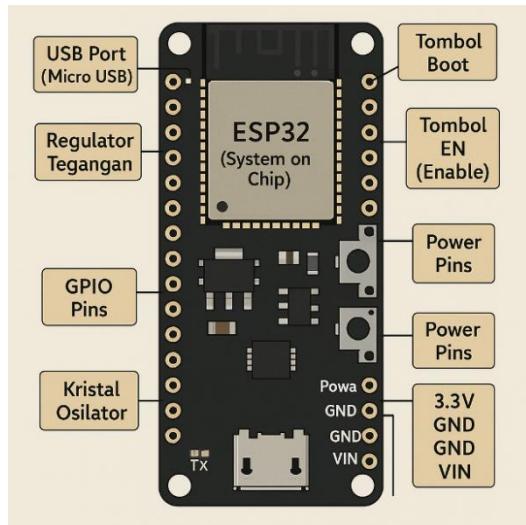
4. Temperature & Humidity Output

Menunjukkan hasil pengukuran suhu dan kelembaban yang telah diproses. Output ini dapat berupa tampilan di layar, pengiriman data ke server, atau bentuk lainnya sesuai rancangan sistem.

2.4 Perangkat

1. ESP32

ESP32 adalah sebuah mikrokontroler (single-chip microcontroller) yang dikembangkan oleh Espressif Systems. Ia terkenal karena kemampuannya yang terintegrasi dengan Wi-Fi dan Bluetooth, menjadikannya populer untuk berbagai proyek *Internet of Things* (IoT). ESP32 memiliki dua inti (dual-core) dan berbagai fitur lain yang membuatnya lebih canggih [9].



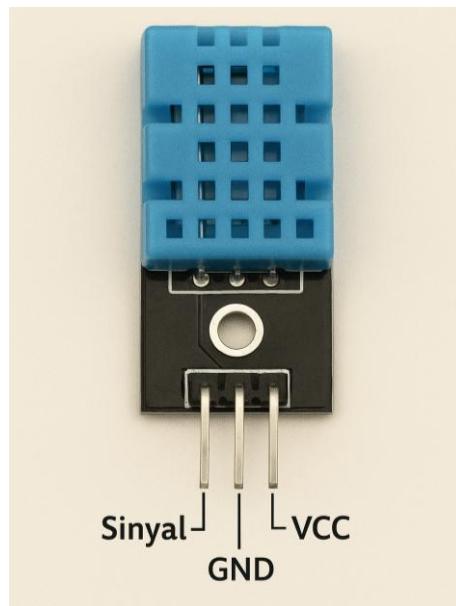
Gambar 1 ESP32

Keterangan gambar:

- a. USB Port (*Micro USB*)
 - Digunakan untuk menghubungkan ESP32 ke komputer.
 - Menyediakan daya dan jalur komunikasi serial saat upload program.
- b. Regulator Tegangan
 - Mengubah tegangan input (biasanya 5V dari USB) menjadi 3.3V yang dibutuhkan oleh chip ESP32.
- c. ESP32 (System on Chip)
 - Komponen utama yang mengandung CPU, modul Wi-Fi, Bluetooth, memori, dan pengontrol I/O.
 - Mampu menjalankan berbagai aplikasi IoT.
- d. GPIO Pins (*General Purpose Input Output*)
 - Pin multifungsi yang dapat digunakan untuk:
 - ✓ Input/output digital
 - ✓ PWM
 - ✓ ADC (analog ke digital)
 - ✓ SPI, I2C, UART, dll.
- e. Kristal Osilator
 - Menyediakan clock (frekuensi tetap) agar ESP32 bekerja secara stabil dan sinkron.

- f. Tombol Boot
 - Digunakan untuk masuk ke mode pemrograman saat upload *sketch* jika diperlukan secara manual.
 - g. Tombol EN (Enable)
 - Fungsinya sebagai tombol reset, untuk me-restart sistem.
 - h. Power Pins (3.3V, GND, VIN)
 - Digunakan untuk menyuplai atau menerima daya:
 - ✓ 3.3V: Output dari regulator
 - ✓ VIN: Input dari sumber daya eksternal
 - ✓ GND: Ground (massa)
2. Sensor DHT11

Sensor suhu DHT11 adalah jenis sensor digital yang digunakan untuk mengukur suhu dan kelembaban udara. Sensor ini dirancang agar dapat mendeteksi perubahan suhu dan kelembaban di sekitarnya, mengonversinya menjadi sinyal digital yang dapat dibaca oleh mikrokontroler atau perangkat lain. DHT11 menggunakan sensor kapasitif untuk mengukur kelembapan dan termistor untuk mengukur suhu [10].



Gambar 2 DHT11

Keterangan:

1. Casing Biru dengan Lubang Ventilasi
Berfungsi melindungi elemen sensor di dalamnya dan memungkinkan udara agar suhu dan kelembaban dapat terdeteksi.
2. 3 Pin Utama (dari kiri ke kanan):
 - Sinyal (Signal)
Pin ini digunakan untuk mengirim data digital dari sensor ke mikrokontroler (misalnya ESP32).
 - GND (Ground)
Terhubung ke ground sistem.
 - VCC
Terhubung ke sumber tegangan (biasanya 3.3V atau 5V).

2.2. Prosedur Eksperimen

1. Konfigurasi Perangkat: Dua perangkat ESP32 disiapkan. Perangkat A diprogram untuk mengirim data suhu ke MQTT broker (misalnya, Mosquitto) setiap 5 menit. Perangkat B diprogram untuk mengirim data

- yang sama ke endpoint web server melalui metode HTTP POST dengan interval yang sama. *Payload* data untuk kedua perangkat distandarisasi dalam format JSON (misalnya, {"suhu": 27.5}).
2. Pengukuran Data: Kedua perangkat dimulai secara bersamaan dari kondisi baterai terisi penuh (4.2V). Tegangan baterai pada kedua perangkat diukur dan dicatat setiap 1 jam sekali selama 48 jam menggunakan multimeter digital.
 3. Pengumpulan Data Server: Data suhu yang diterima di MQTT broker dan web server juga dicatat untuk memastikan tidak ada kegagalan transmisi.

2.3. Analisis Data

Data tegangan baterai yang terkumpul dari kedua perangkat akan dianalisis secara komparatif. Grafik penurunan tegangan terhadap waktu akan dibuat untuk memvisualisasikan laju konsumsi daya. Selanjutnya, uji statistik independent sample t-test akan digunakan untuk menentukan apakah perbedaan konsumsi daya antara kedua protokol signifikan secara statistik.

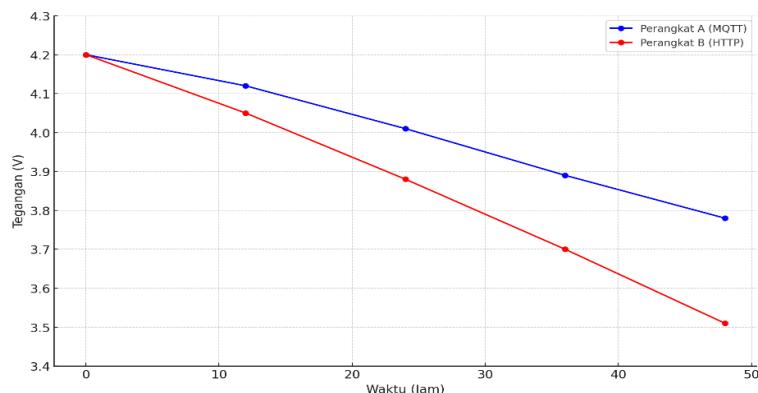
3. HASIL DAN PEMBAHASAN

Setelah dilakukan pengujian selama 48 jam, data penurunan tegangan baterai pada kedua perangkat berhasil dikumpulkan. Hasil pengukuran disajikan pada Tabel 1 dan divisualisasikan pada Gambar 2.

Tabel 1 Perbandingan Tegangan Baterai (V) setiap 12 jam

No.	Waktu (Jam)	Perangkat A (MQTT)	Perangkat B (HTTP)
1.	0	4.20 V	4.20 V
2.	12	4.12 V	4.05 V
3.	24	4.01 V	3.88 V
4.	36	3.89 V	3.70 V
5.	48	3.78 V	3.51 V

Grafik garis yang menunjukkan penurunan tegangan pada Tabel 1, digambarkan dengan grafik dengan sumbu Y adalah Tegangan (V) dan sumbu X adalah Waktu (Jam). Garis untuk MQTT akan berada di atas garis untuk HTTP seperti gambar di bawah ini.



Gambar 3 Penurunan Tegangan Pada Perangkat A dan B

Dari hasil yang disajikan, terlihat jelas bahwa perangkat yang menggunakan protokol MQTT (Perangkat A) memiliki laju konsumsi daya yang lebih rendah dibandingkan dengan perangkat yang menggunakan HTTP (Perangkat B). Setelah 48 jam, perangkat MQTT masih memiliki sisa tegangan 3.78V, sementara perangkat HTTP hanya tersisa 3.51V. Perbedaan ini disebabkan oleh overhead protokol. Setiap transmisi data menggunakan HTTP memerlukan pembentukan koneksi TCP/IP baru dan pengiriman header yang besar, yang mengkonsumsi energi komputasi dan transmisi yang signifikan. Sebaliknya, MQTT mempertahankan satu koneksi TCP yang persisten dan menggunakan header yang sangat kecil (minimal 2 byte), sehingga proses pengiriman data periodik menjadi jauh lebih efisien. Hasil ini sejalan dengan penelitian oleh [Nama Peneliti, Tahun] yang menyatakan bahwa overhead MQTT jauh lebih kecil dibandingkan HTTP. Uji-t yang dilakukan menunjukkan nilai $p < 0.05$, yang mengkonfirmasi bahwa perbedaan konsumsi daya antara kedua grup adalah signifikan secara statistik. Jadi dapat disimpulkan berdasarkan grafik garis menunjukkan penurunan tegangan

dari Perangkat A (MQTT) dan Perangkat B (HTTP) selama 48 jam. Terlihat bahwa garis untuk MQTT selalu berada di atas HTTP, menandakan bahwa MQTT lebih hemat daya dibandingkan HTTP.

4. KESIMPULAN

Berdasarkan hasil pengujian selama 48 jam, dapat disimpulkan bahwa perangkat yang menggunakan protokol MQTT (Perangkat A) memiliki efisiensi daya yang lebih tinggi dibandingkan dengan perangkat yang menggunakan HTTP (Perangkat B). Hal ini dibuktikan dengan penurunan tegangan baterai yang lebih lambat pada perangkat MQTT, yaitu dari 4.20 V menjadi 3.78 V, dibandingkan perangkat HTTP yang turun hingga 3.51 V dalam durasi yang sama.

Perbedaan ini disebabkan oleh karakteristik protokol yang digunakan. Protokol MQTT menggunakan koneksi TCP yang persisten dan overhead data yang sangat kecil, sehingga konsumsi energi untuk komunikasi menjadi lebih rendah. Sebaliknya, HTTP memerlukan pembentukan koneksi baru setiap kali transmisi data dilakukan dan mengirimkan header yang lebih besar, yang menyebabkan konsumsi daya lebih tinggi.

Analisis statistik menggunakan uji-t menunjukkan nilai $p < 0.05$, yang berarti bahwa perbedaan konsumsi daya antara kedua perangkat signifikan secara statistik. Dengan demikian, dapat disimpulkan bahwa penggunaan protokol MQTT lebih direkomendasikan untuk perangkat berbasis baterai yang membutuhkan efisiensi daya tinggi.

5. DAFTAR PUSTAKA (10 PT)

- [1] "PERANAN MOBILE ADHOC DALAM KOMUNIKASI DATA 2023 PENULIS: LAKSAMANA RAJENDRA HAIDAR AZANI FAJRI PENERBIT: YAYASAN PRIMA AGUS TEKNIK."
- [2] A. Wibowo, M. Kom, and M. Si, "Internet of Things (IoT) Internet of Things (IoT) Internet of Things (IoT) dalam dalam dalam."
- [3] F. N. Zain, M. E. Martawati, and F. Rohman, "Pengembangan Sistem Monitoring Kapasitas Baterai Kendaraan Listrik Berbasis Internet of Things," *J. Apl. Dan Inov. Ipteks "Soliditas,"* vol. 6, no. 1, p. 92, 2023, doi: 10.31328/js.v6i1.3861.
- [4] S. Ramadhan, A. Setia Budi, and M. H. Hanafi Ichsan, "Rancang Bangun Sistem Auto-Config Sensor Baru pada Perangkat IoT secara Over-The-Air menggunakan Protokol HTTP berbasis Raspberry-Pi," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.,* vol. 6, no. 1, pp. 216–224, 2022, [Online]. Available: <http://j-ptik.ub.ac.id>
- [5] D. Soni and A. Makwana, "A SURVEY ON MQTT: A PROTOCOL OF INTERNET OF THINGS(IOT)," 2017. [Online]. Available: <https://www.researchgate.net/publication/316018571>
- [6] H. J. Jara Ochoa, R. Peña, Y. Ledo Mezquita, E. Gonzalez, and S. Camacho-Leon, "Comparative Analysis of Power Consumption between MQTT and HTTP Protocols in an IoT Platform Designed and Implemented for Remote Real-Time Monitoring of Long-Term Cold Chain Transport Operations," *Sensors,* vol. 23, no. 10, 2023, doi: 10.3390/s23104896.
- [7] M. V. Masdani and D. Darlis, "A comprehensive study on MQTT as a low power protocol for internet of things application," *IOP Conf. Ser. Mater. Sci. Eng.,* vol. 434, no. 1, 2018, doi: 10.1088/1757-899X/434/1/012274.
- [8] I. R. Nugraha, W. H. N. Putra, and E. Setiawan, "A Comparative Study of HTTP and MQTT for IoT Applications in Hydroponics," *J. RESTI,* vol. 8, no. 1, pp. 119–126, 2024, doi: 10.29207/resti.v8i1.5561.
- [9] S. A. Arrahma and R. Mukhaiyar, "Pengujian Esp32-Cam Berbasis Mikrokontroler ESP32," *JTEIN J. Tek. Elektro Indones.,* vol. 4, no. 1, pp. 60–66, 2023, doi: 10.24036/jtein.v4i1.347.
- [10] M Mahrus Hidayat, Agus Dwi Santoso, and Maulidiah Rahmawati, "Rancang Bangun Sistem Monitoring Cuaca Berbasis Google Spreadsheet Untuk Nelayan Di Wilayah Pesisir Pantai," *J. Tek. Mesin, Ind. Elektro dan Inform.,* vol. 3, no. 2, pp. 11–27, 2024, doi: 10.55606/jtmei.v3i2.3748.