



Analisa Power Mode ESP32 Untuk Catu Daya Pada Sistem Berbasis IoT

Yani Prabowo¹, Suwasti Broto², Tatang Wirawan Wisnuadji³, Siswanto⁴
^{1,2,3,4} Universitas Budi Luhur
wisjhnuadji@budiluhur.ac.id

Abstract

One of the best choices for IoT applications based on WiFi is the ESP32 microcontroller. In addition, ESP32 is a device that has built-in BLE (Bluetooth Low Energy) support, so it is very suitable for portable applications. However, one of the problems that arise for portable applications is the existence of battery power support, so a way must be found to be able to apply significant battery usage savings techniques, which in the ESP32 microcontroller is known as Sleep-Mode, which aims to extend the duration of battery power usage on microcontroller. The method that will be applied to the system is an analysis of the use of Li-Ion 18650 batteries as the main support for providing electricity for the ESP32. The first thing to do is to analyze the power supply for several Power Modes from the ESP32, where time duration tests will be carried out for Li-Ion 18650 batteries for each type of ESP32 Power Mode, namely Active Mode, Modem Sleep Mode, Light Sleep Mode, Deep Sleep Mode, and Hibernation Mode. The results obtained are the Li-Ion battery capacity values that must be provided to support the operation of the ESP32 system device as the heart of the portable IoT system.

Keywords: IoT, ESP32, BLE, power mode, portable

Abstrak

Salah satu pilihan terbaik untuk kebutuhan aplikasi IoT yang berbasis pada WiFi adalah mikrokontroler ESP32. Selain itu ESP32 merupakan device yang memiliki dukungan BLE (Bluetooth Low Energy) yang bersifat built in, sehingga sangat cocok digunakan untuk aplikasi yang bersifat portable. Akan tetapi salah satu masalah yang timbul untuk aplikasi portable adalah keberadaan dukungan tenaga baterai, sehingga harus dicari suatu cara untuk dapat menerapkan teknik penghematan penggunaan baterai secara signifikan, dimana dalam mikrokontroler ESP32 dikenal sebagai Sleep-Mode, yang bertujuan untuk memperpanjang durasi penggunaan tenaga baterai pada mikrokontroler. Metoda yang akan diterapkan pada sistem adalah analisa penggunaan baterai Li-Ion 18650 sebagai pendukung utama penyediaan tenaga listrik bagi ESP32. Yang harus dilakukan adalah melakukan analisa catu daya untuk beberapa Power Mode dari ESP32, dimana akan dilakukan pengujian durasi waktu untuk baterai Li-Ion 18650 untuk tiap jenis Power Mode ESP32, yaitu Active Mode, Modem Sleep Mode, Light Sleep Mode, Deep Sleep Mode, Dan Hibernation Mode. Hasil yang didapatkan adalah Nilai kapasitas baterai Li-Ion yang harus disediakan untuk mendukung pengoperasian perangkat sistem ESP32 sebagai jantung dari sistem IoT yang bersifat portabel tersebut..

Kata kunci : IoT, ESP32, BLE, power mode, portabel

1. Pendahuluan

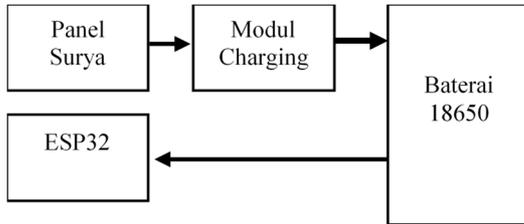
Penggunaan peralatan untuk keperluan pengendalian dan pemantauan berbasis IoT, pada saat ini sudah merupakan kebutuhan dalam kehidupan sehari-hari, terutama pada sentra sentra pelayanan publik, perkantoran, dan industri, namun demikian tak jarang pula peralatan tersebut juga harus dipasang pada daerah-daerah yang lokasinya jauh dari jangkauan sumber energi listrik, seperti di pedesaan, daerah terpencil, daerah hutan dan pegunungan, sehingga penggunaan peralatan portabel menjadi pilihan utama[1][2]. Namun demikian ada kendala dalam pengadaan sumber listriknya. Oleh karenanya penggunaan tenaga surya melalui panel panel Solar Cell menjadi pilihan yang tak terelakkan[3]. Salah satu jenis

mikrokontroler yang berbasis IoT yang sangat mudah didapatkan di pasaran adalah ESP32[4], dengan rancangan dan bentuk yang ringkas dan kompak, serta konsumsi energi listrik yang relatif kecil dan hemat, disertai fitur-fitur pengendalian yang cukup lengkap, menjadikan ESP32 ini mudah untuk diintegrasikan dengan sistem pembangkit listrik Solar Cell, dimana efisiensi penggunaan daya, serta efektivitas alat menjadi persyaratan yang penting[5]. Demikian pula baterai Li-Ion 18650 yang murah dan mudah didapatkan dipasaran, merupakan salah satu alasan mengapa jenis baterai ini digunakan dalam penelitian ini[6][7]. Kapasitas baterai ini dapat dinaikkan dan diturunkan sesuai dengan skala kebutuhan energi listrik pada sistem. Dengan cara melakukan konfigurasi packing baterai secara serial

maupun paralel, untuk menghasilkan kapasitas energi Modul Charging ini memiliki spesifikasi pada Tabel 1. yang dibutuhkan.

2. Metode Penelitian

Sistem memiliki diagram blok seperti digambarkan pada gambar 1. di bawah ini



Gambar 1. Diagram blok sistem

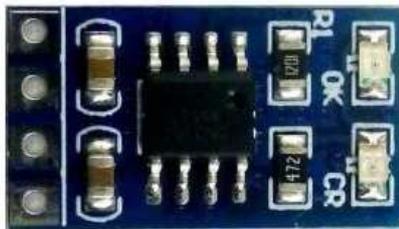
Dalam penelitian ini digunakan 1 keping panel surya 50WP, dilengkapi dengan SCC MPPT 10A, dan 5 baterai Li-Ion 18650 3000mAh paralel.

Panel Surya dalam sistem ini berfungsi untuk menangkap pancaran sinar matahari kemudian merubahnya menjadi energi listrik, yang kemudian arus listriknya akan digunakan mengisi baterai setelah melalui Modul Charger. Panel surya yang digunakan adalah jenis Panel Surya Mini[8], seperti yang terlihat pada gambar 2.



Gambar 2. Panel Surya Mini

Modul Charging ini berfungsi untuk mengatur aliran arus listrik yang berasal dari panel surya dan kemudian melakukan pengisian pada baterai yang tersedia, secara kontinu selama panel surya masih mendapatkan pancaran energi dari matahari[9]. Modul Charging SD05CRMA terlihat pada gambar 3. Di bawah ini. Besar kecilnya energi yang dihasilkan untuk melakukan charging pada baterai, bergantung pula pada ukuran dari panel surya. Sedangkan di Indonesia rata-rata proses photovoltaic hanya berlangsung selama 5 jam, karena banyak daerah tropis, sering awan menutupi matahari sehingga dalam satu hari rata-rata hanya dihasilkan waktu efektif 5 jam.



Gambar 3. Modul Charging SD05CRMA

Tabel 1. Karakteristik Arus SD05CRMA

R_1	I_{CH}
1.18 K	1.0 A
1.47 K	0.8 A
2.37 K	0.5 A
11.8 K	0.1 A

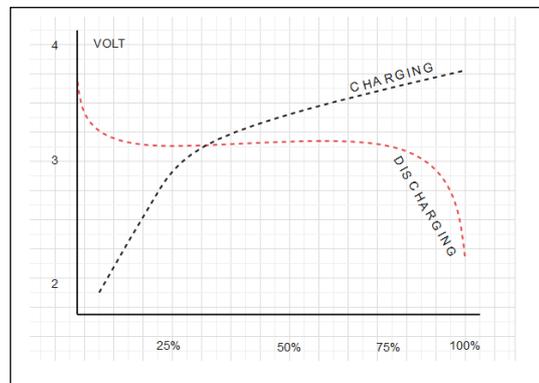
Baterai yang dipergunakan seperti yang terlihat pada Gambar 4. Di bawah ini.



3.7V Li-ion

Gambar 4. Baterai Li-Ion 18650 3000mAh

Untuk Tipe Baterai yang digunakan adalah Jenis Li-Ion 18650 3000mAh, baterai ini memiliki karakteristik seperti pada Gambar 4., dimana grafik untuk pengisian baterai (charging), serta grafik untuk pengosongan baterai (discharging) terlihat dalam satu grafik sehingga keduanya dapat dibandingkan satu sama lain[10]. Grafik tersebut dapat dilihat pada Gambar 5.

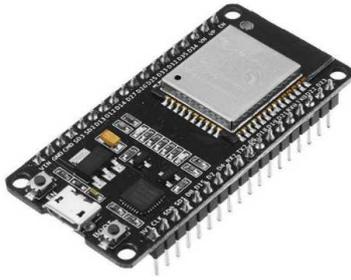


Gambar 5. Charging dan Discharging Baterai Li-Ion 18650

Charging: Proses Pengisian pada baterai atau charging tidak dimulai dari titik Nol, disebabkan karena baterai memiliki residu, yaitu muatan tetap sehingga memiliki tegangan tertentu yang rendah sekitar 2 Volt, sehingga ketika proses pengisian dimulai, maka tegangan pada baterai juga akan ikut naik, sesuai dengan grafik Charging di atas. **Discharging:** Proses Discharging atau pengosongan dimulai ketika baterai diberi beban, rate kapasitas beban biasanya dinyatakan dengan 0.1C, 0.2C, 1C, 2C dan lain sebagainya. Semakin besar nilai C, maka grafik akan cenderung turun dengan slope atau kemiringan yang

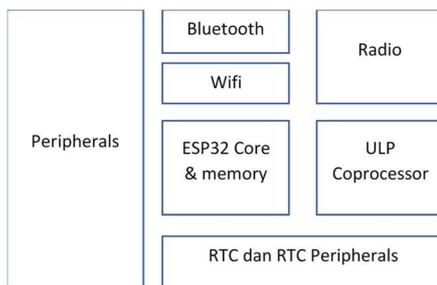
lebih tajam, sehingga kapasitas baterai akan lebih cepat terkuras habis.

ESP32 DevKit dapat dilihat pada gambar 6 di bawah ini.



Gambar 6. ESP32 DevKit

Bagan dari ESP32 pada umumnya adalah seperti pada gambar 7 di bawah ini[11]



Gambar 7. Diagram Blok ESP32

ESP32 memiliki 5 jenis Power Mode, antara lain: Active Mode, Modem Sleep, Light Sleep, Deep Sleep dan Hibernation. Sedangkan status sinyal masing masing Power Mode ESP 32 dapat dilihat secara rinci pada Tabel 2.

Tabel 2. Status Sinyal pada Modul ESP32

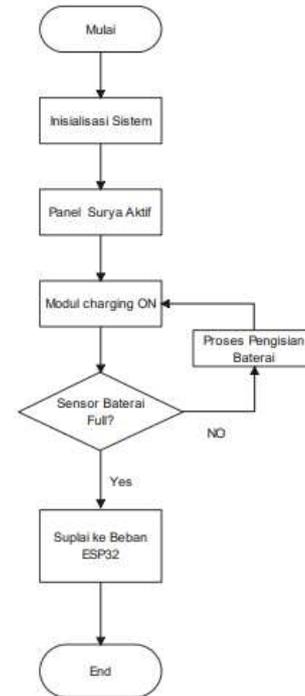
Hardware	Active Mode	Modem Sleep Mode	Light Sleep Mode	Deep Sleep Mode	Hibernation
CPU	ON	ON	PAUSE	OFF	OFF
WiFi/BT	ON	OFF	OFF	OFF	OFF
RTC	ON	ON	ON	ON	OFF
ULP CoProcessor	ON	ON	ON	ON / OFF	OFF

Kita dapat mereduksi konsumsi energi listrik dari ESP32 ini berkurang sampai 95%, dengan menerapkan fungsi optimisasi penggunaan daya listrik selama ESP32 digunakan. Adapun tujuan penggunaan Power Mode adalah agar daya tahan baterai yang digunakan oleh sistem menjadi sehemat mungkin. Caranya adalah dengan melakukan penyetelan pada fungsi fungsi ESP32, seperti diketahui bahwa peralatan berbasis IoT ini banyak dipasang pada lokasi lokasi yang tidak ada pasokan listrik dari jaringan listrik konvensional, seperti Jaringan PLN pada umumnya. Sehingga penggunaan baterai memiliki peranan yang sangat penting. Sebagai sumber catu daya perangkat IoT. Sedangkan pasokan listrik dari Panel Surya juga terbatas, mengingat faktor cuaca, dimana tidak memungkinkan Panel Surya menerima secara terus menerus pancaran sinar matahari.

Sehingga faktor efisiensi panel surya juga relatif kecil, terutama untuk tropis dimana Hujan dan awan sering kali menutupi panel, dimana akan berakibat turunnya pasokan listrik untuk baterai.

2.1. Flowchart Sistem

Untuk menggambarkan logika sistem ketika beroperasi maka dapat digambarkan dalam sebuah flowchart seperti pada gambar 8. Dibawah ini.



Gambar 8. Flowchart Sistem

Keterangan: Mulai, Inisialisasi Sistem, sistem diaktifkan dan sistem siap beroperasi. Panel Surya Aktif: Dalam tahap ini Panel Surya mulai menghasilkan arus listrik, dengan asumsi ada cahaya matahari yang diterima. Modul Charging ON: Modul ini aktif, dan sensor level baterai mulai bekerja. Sensor Baterai Full? : Jika NO, artinya baterai dalam kondisi LOW, maka proses pengisian baterai dimulai. Jika YES: maka Baterai akan memberikan daya kepada beban, berupa chip ESP32. End.

3. Hasil dan Pembahasan

Mode Aktif ESP32, pada saat aktif dengan mode ini menggunakan arus sebesar 260 mA, pada saat modul wifi, bluetooth, radio, esp32 Core, ULP coprosesor, Peripherals dan RTC arus yang dibutuhkan adalah 790mA.

ESP32 Modem Sleep, dalam mode sleep modem semuanya aktif kecuali wifi, bluetooth dan radio. Itu juga membuat CPU tetap berjalan dan clock RTC dapat dikonfigurasi. Menggunakan mode ini chip mengkonsumsi sekitar 3mA pada kecepatan lambat dan 20mA pada kecepatan tinggi. Untuk menjaga koneksi WiFi/Bluetooth tetap hidup, Wi-Fi, Bluetooth, dan radio

dibangunkan pada interval yang telah ditentukan. Ini dikenal sebagai Association Sleep Pattern.

Selama pola sleep ini, ESP32 beralih antara mode aktif dan mode sleep mode. Untuk mencapai ini, ESP32 terhubung ke router dalam mode stasiun menggunakan mekanisme DTIM. Waktu tidur ditentukan oleh waktu interval suar DTIM router yang biasanya 100ms hingga 1000ms.

ESP32 Light Sleep mirip dengan sleep modem. Selama mode light sleep, CPU dijeda dengan mematikan pulsa clock-nya, sementara RTC dan koprosesor ULP tetap aktif. Hal ini menghasilkan konsumsi daya yang lebih rendah dibandingkan mode sleep modem yaitu sekitar 0.8mA.

Dalam mode deep sleep, CPU, sebagian besar RAM, dan semua periferal digital dimatikan. Satu-satunya bagian dari chip yang tetap beroperasi adalah: Koprosesor ULP, Pengontrol RTC, Periferal RTC, RTC memori cepat dan lambat Chip mengkonsumsi sekitar 0,15 mA (jika koprosesor ULP aktif) hingga 10µA. Selama mode sleep, CPU utama dimatikan, sedangkan koprosesor UltraLowPower (ULP) dapat mengambil pembacaan sensor dan membangunkan CPU kapan pun diperlukan. Pola tidur ini dikenal sebagai pola yang dipantau sensor ULP. Ini berguna untuk merancang aplikasi di mana CPU perlu dibangunkan oleh peristiwa eksternal, atau timer, atau kombinasi keduanya, sambil mempertahankan konsumsi daya minimal. Seiring dengan CPU, memori utama chip juga dinonaktifkan. Akibatnya semua yang tersimpan di memori itu terhapus dan tidak bisa diakses., tetapi memori RTC tetap menyala, isinya dipertahankan bahkan selama sleep mode dapat diambil setelah chip dibangunkan.

ESP32 Hibernation mode, dalam mode hibernasi chip menonaktifkan osilator internal 8 MHz serta koprosesor ULP. Memori pemulihan RTC juga dimatikan, yang berarti tidak dapat menyimpan data apa pun saat dalam mode hibernasi. Segala sesuatu dimatikan kecuali hanya satu timer RTC (pada jam lambat) dan beberapa GPIO RTC yang digunakan untuk membangunkan chip dari mode hibernasi, power yang dikonsumsi adalah sekitar 2.5µA dalam mode hibernasi.

Hasil diatas dapat dinyatakan dalam sebuah tabel seperti yang terlihat pada Tabel 2.

Power Mode	Arus (mA)	Durasi (jam)
Active Mode		
Wi-Fi Tx packet 13dBm~21dBm	260	11.5
Wi-Fi/BT Tx packet 0dBm	120	25
Wi-Fi/BT Rx and listening	90	33.3

Power Mode	Arus (mA)	Durasi (jam)
Modem Sleep	20	150
Light Sleep	0.8	3750
Deep Sleep	0.01	300000
Hibernation	0.0025	1200000

Tabel 2 menunjukkan penggunaan arus listrik dalam beberapa Power Mode ESP32 yang ada, baterai yang digunakan dalam penelitian ini adalah jenis Li-Ion 18650, 3000 mAh.

Pada Active Mode ini, terlihat bahwa proses transmisi atau TX data menyebabkan transmitter harus mengeluarkan daya listrik yang lebih besar, jika dibandingkan pada saat proses RX, hal ini akan menyebabkan penggunaan arus pada baterai menjadi lebih besar pada saat proses RX.

Demikian pula jika proses TX data menggunakan level dBm yang lebih besar, artinya bahwa semakin kuat sinyal radio yang dipancarkan akan menyebabkan penggunaan daya listrik yang semakin besar, dari tabel 2 dapat dinyatakan bahwa proses TX dengan level sinyal sebesar 13dBm-21dBm akan mengkonsumsi arus sebesar 260mA, sedangkan proses TX dengan daya 0 dBm mengkonsumsi arus listrik sebesar 120 mA.

Untuk Proses RX atau Listening, membutuhkan arus listrik sebesar 90 mA. Sedangkan semakin besar konsumsi arus pada Active Mode ini, akan semakin memperkecil durasi waktu ketahanan baterai. Untuk Active Mode dimana sistem hanya melakukan listening atau RX saja, maka durasi waktu menjadi yang paling besar, yaitu 33.3 jam, disusul dengan Active mode TX, 0dBm yang memiliki durasi waktu untuk baterai sebesar 25 jam, sedangkan untuk Active Mode dengan 13dBm-21dBm memiliki durasi waktu untuk baterai sebesar 11.5 jam.

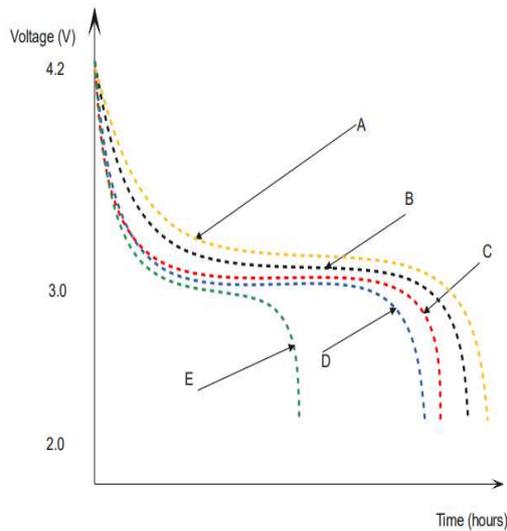
Pada Mode Sleep ini penggunaan arus listrik adalah sebesar 20 mA, dari hasil pengukuran didapatkan durasi waktu untuk baterai berkapasitas 3000 mAh sebesar 150 jam.

Untuk mode Light Sleep ini arus yang dikonsumsi adalah sebesar 0.8 mA, sehingga untuk baterai berkapasitas 3000 mAh, durasi waktunya adalah sebesar 3750 jam.

Pada mode Deep Sleep ini arus listrik yang dibutuhkan semakin kecil, yaitu sebesar 0.01 mA, sehingga untuk baterai jenis Li-Ion 18650 berkapasitas 3000 mAh akan memiliki durasi 300 ribu jam.

Pada mode Hibernation ini, penggunaan arus listrik sangat kecil, karena banyak fungsi modul yang berada dalam kondisi OFF, dan ini membutuhkan arus sebesar 0.0025 mA, sehingga durasi waktu untuk jenis Li-Ion 18650 berkapasitas 3000 mAh memiliki durasi 1.2 juta jam.

Untuk semua Power Mode ESP32 dapat dirangkum dalam sebuah grafik yang dapat dilihat pada gambar 9.



Gambar 9. Grafik Power Mode ESP32

A: Hibernation, B: Deep Sleep, C: Light Sleep, D: Modem Sleep, E: Active Mode

4. Kesimpulan

Hasil uji coba menunjukkan bahwa penggunaan Power Mode dalam pengoperasian modul ESP32, konsumsi daya listrik terbesar dan paling dominan adalah pada Active Mode. Ditunjukkan pada Grafik E, dimana dimana waktu penggunaan Baterai Li-Ion 18650 memiliki durasi waktu yang paling singkat yaitu 11.5 jam. Kondisi seperti ini dapat diatasi dengan penggunaan Power mode yang lain seperti Modem Sleep, Light Sleep dan Deep Sleep, sehingga ketika modul ESP32 dalam kondisi tidak aktif atau idle, penggunaan baterai dapat dihemat, sehingga durasi penggunaan baterai lebih panjang.

Daftar Rujukan

- [1] A. Prafanto, E. Budiman, P. P. Widagdo, G. M. Putra, and R. Wardhana, "Pendeteksi Kehadiran menggunakan ESP32 untuk Sistem Pengunci Pintu Otomatis," *JIT (Jurnal Teknol. Ter.*, vol. 7, no. 1, p. 37, 2021, doi: 10.31884/jtt.v7i1.318.
- [2] A. Ramschie, J. Makal, R. Katuuk, and ..., "Pemanfaatan ESP32 Pada Sistem Keamanan Rumah Tinggal Berbasis IoT," ... *Work. Natl. ...*, pp. 4-5, 2021, [Online]. Available: <https://jurnal.polban.ac.id/ojs-3.1.2/proceeding/article/view/2688/2076>
- [3] D. A. R. Fajrika hadnis Putra, Kemas Muslim Lhaksana, "Aplikasi IoT untuk Rumah Pintar dengan," *J. e-Proceeding Eng.*, vol. 5, no. 1, pp. 1746-1760, 2018.
- [4] P. A. A. P. Basabilik, "Rancang Bangun Sistem Pendeteksi Jatuh Berbasis Internet of Things," *Prism. Fis.*, vol. 9, no. 2, pp. 110-116, 2021.
- [5] T. S. Rao, P. Pranay, S. Narayana, Y. Reddy, Sunil, and P. Kaur, "ESP32 Based Implementation of Water Quality and Quantity Regulating System," *Proc. 3rd Int. Conf. Integr. Intell. Comput. Commun. Secur. (ICIIC 2021)*, vol. 4, no. Iciiic, pp. 122-129, 2021, doi: 10.2991/ahis.k.210913.016.
- [6] M. T. Afif and I. A. P. Pratiwi, "Analisis Perbandingan Baterai Lithium-Ion, Lithium-Polymer, Lead Acid Dan Nickel-Metal Hydride Pada Penggunaan Mobil Listrik-Review," *J. Rekasaya*

- Mesin*, vol. 6, no. 2, pp. 95-99, 2015.
- [7] D. R. Kim, J. W. Kang, T. H. Eom, J. M. Kim, J. Lee, and C. Y. Won, "Rapid-charging solution for 18650 cylindrical lithium-ion battery packs for forklifts," *J. Electrochem. Sci. Technol.*, vol. 9, no. 3, pp. 184-194, 2018, doi: 10.5229/JECST.2018.9.3.184.
- [8] T. Haryanto, "Perancangan Energi Terbarukan Solar Panel Untuk Essential Load Dengan Sistem Switch," *J. Tek. Mesin*, vol. 10, no. 1, p. 43, 2021, doi: 10.22441/jtm.v10i1.4779.
- [9] T. Majaw, R. Deka, S. Roy, and B. Goswami, "Solar Charge Controllers using MPPT and PWM: A Review," *ADBU J. Electr. Electron. Eng.*, vol. 2, no. 1, pp. 1-4, 2018, [Online]. Available: <https://media.neliti.com/media/publications/287658-solar-charge-controllers-using-mppt-and-66d6c4aa.pdf>
- [10] X. Hua and A. Thomas, "Effect of dynamic loads and vibrations on lithium-ion batteries," *J. Low Freq. Noise Vib. Act. Control*, vol. 40, no. 4, pp. 1927-1934, 2021, doi: 10.1177/14613484211008112.
- [11] Ardutech, "Mengenal ESP32 Development Kit untuk IoT," *Web Page*, 2020. <https://www.ardutech.com/mengenal-esp32-development-kit-untuk-iot-internet-of-things/> (accessed Sep. 02, 2022).