

Prototipe Penekan Tombol Otomatis Simulator Meter kWh Untuk Pengisian Token Berbasis Mikroprosesor Dan API Telegram

Prototype of kWh Meter Simulator Automatic Button Presser for Token Recharge Based on Microprocessor and Telegram API

¹Vera Firmansyah ^{*}), ¹Karlina Oktaviyana, ¹H. Syauqi Fathan, ¹Azis Muslim, ¹Dudi Adi Firmansyah, ¹Nandang Gunawan T.W, ¹Willi Sutanto dan ¹Gianto

¹Akademi Metrologi dan Instrumentasi, Kabupaten Sumedang, Indonesia, 45362

^{*}) *corresponding email: verafirmansyah@akmet.ac.id*

Abstrak

Perusahaan Listrik Negara (PLN) telah beralih dari sistem pembayaran pascabayar ke prabayar untuk energi listrik, mengingat adanya kelemahan dalam sistem pascabayar. Dalam metode prabayar, pengguna diharuskan memasukkan angka token 20-digit ke dalam meter kWh dengan menggunakan papan tombol. Namun, terdapat beberapa kendala yang muncul terkait dengan posisi meter kWh dan tingkat mobilitas pengguna yang tinggi. Oleh karena itu, penerapan sistem *Internet of Things* (IoT) pada tombol otomatis untuk meter kWh diharapkan dapat memberikan kemudahan bagi pengguna. Pembuatan prototipe menggunakan komponen utama, yaitu *Raspberry* sebagai komputer sirkuit tunggal, Telegram sebagai input dan output pengguna beserta 12 buah solenoid sebagai media penekan papan tombol. Pengujian kelayakan sistem dilakukan dengan menguji akurasi penekanan, waktu pengisian, dan keberterimaan pesan pada bot Telegram. Akurasi untuk setiap tombol dengan pengujian 20 kali penekanan pada *delay* 2,00 detik hingga 0,04 detik mendapatkan hasil 100% akurat. Pengujian waktu pengisian yang dilakukan dengan *delay* antara 0,10 detik hingga 0,04 detik menghasilkan akurasi 100%, yang lebih cepat dibandingkan dengan metode pengisian manual. Pada pengujian keberterimaan pesan Telegram, sebanyak 13 pesan diterima dengan akurasi 100%. Pengembangan lebih lanjut dapat dilakukan dengan mempertimbangkan tampilan bot, input dan output pengguna, dan wadah disesuaikan pada meter kWh.

Kata Kunci: Meter kWh, API Telegram, Raspberry, IoT.

Abstract

The Indonesian State Electricity Company, PLN, has shifted from a postpaid to a prepaid electricity billing system to address the shortcomings of the former. The prepaid system necessitates users to enter a 20-digit token into the kWh meter via a keypad. However, constraints related to kWh meter placement and high user mobility have been identified. Consequently, integrating *Internet of Things* (IoT) technology into automated buttons for kWh meters is anticipated to enhance user convenience. The core components employed in this prototype are a *Raspberry Pi* single-board computer, Telegram for user interaction, and 12 solenoids to actuate the keypad buttons. System feasibility testing is carried out by testing the accuracy of suppression, filling time, and receiving messages on bot Telegrams. Accuracy for each button with a test of 20 presented at a delay of 2.00 seconds to 0.04 seconds got 100% accurate results. The testing of charging time conducted with a delay between 0.10 seconds and 0.04 seconds resulted in 100% accuracy, which is faster than the manual charging method. In the testing of Telegram message acceptance, 13 messages were received with 100% accuracy. Further development can be done by considering the bot display, user inputs and outputs, and customized containers on the kWh meter.

Keywords: kWh Meter, Telegram API, Raspberry, IoT

Makalah diterima 10 Juli 2024 – makalah direvisi 11 September 2024 – disetujui 18 September 2024

Karya ini adalah naskah akses terbuka dengan lisensi [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/).



1. Pendahuluan

Meter kWh, juga dikenal sebagai meter energi listrik, adalah alat ukur yang digunakan untuk mengukur konsumsi energi listrik dalam kilowatt-hour (kWh). Meter kWh bekerja dengan prinsip mengukur arus listrik yang melewati konduktor dan tegangan listrik antara dua titik. Alat ini menggunakan pengukuran arus dan tegangan untuk menghitung daya listrik dalam satuan watt, dan kemudian mengintegrasikan daya ini selama periode waktu tertentu untuk mendapatkan energi total yang dikonsumsi, yaitu kilowatt-hour. Meter kWh biasanya dilengkapi dengan tampilan digital atau analog yang menunjukkan konsumsi energi dalam bentuk angka. Beberapa meter kWh juga dilengkapi dengan fitur tambahan seperti pemantauan *real-time*, komunikasi jaringan, atau kemampuan mengirimkan data pembacaan secara otomatis [1]-[4]. Meter kWh

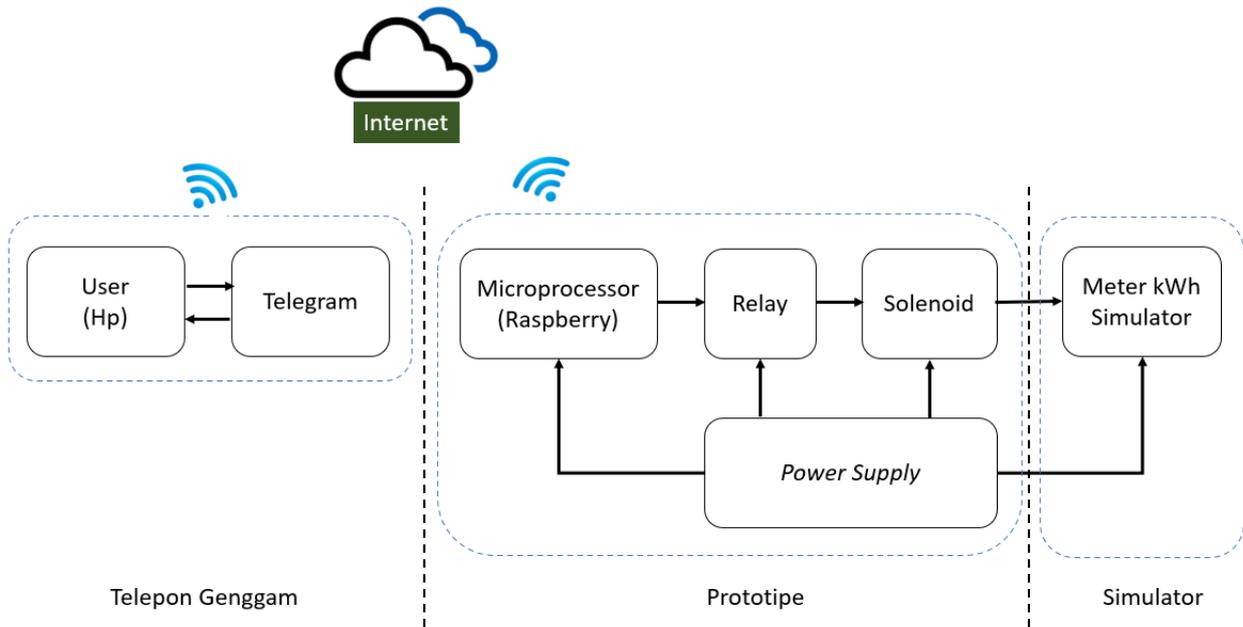
memiliki peranan yang krusial dalam pengukuran serta penagihan biaya penggunaan energi listrik. Alat ini memungkinkan penyedia layanan listrik untuk menentukan tingkat konsumsi energi oleh pelanggan dan menyusun tagihan yang tepat berdasarkan jumlah energi yang telah digunakan. Selain itu, meter kWh juga membantu pengguna dalam memantau dan mengelola konsumsi energi mereka untuk tujuan efisiensi dan penghematan energi [5]. Sebelumnya telah diteliti juga alat untuk pemantauan penggunaan energi listrik dengan membandingkan sisa token di *website* dan pada alat meter kWh, namun tidak dapat dipastikan bagaimana proses pengisian token ke meter kWh tersebut [6]. Telah dibuat juga meter kWh dengan komunikasi berbasis GSM, namun terbatas pada proses monitoring [7].

Meter kWh Prabayar adalah jenis meteran listrik yang memungkinkan pengguna untuk membayar sebelumnya untuk penggunaan energi listrik. Sistem ini memungkinkan pengguna untuk dapat mengontrol atas pengeluaran energi listrik mereka dan memantau penggunaan energi secara *real-time*. Meter kWh Prabayar biasanya dilengkapi dengan sebuah kartu atau token yang berisi kredit listrik. Pengguna harus memasukkan kartu atau token ke dalam meteran untuk mengisi ulang kredit listrik mereka. Jumlah kredit akan dikurangi secara otomatis seiring dengan penggunaan energi listrik. Keuntungan dari meter kWh Prabayar adalah sebagai berikut: kontrol pengeluaran, mencegah tagihan tertunda, pengelolaan budget yang lebih baik, dan memotivasi efisiensi energi. Namun, ada juga beberapa keterbatasan dari meter kWh Prabayar, seperti kebutuhan untuk secara teratur mengisi ulang kredit listrik dan kemungkinan terputusnya pasokan listrik jika kredit habis.

Untuk melengkapi keterbatasan meter kWh Prabayar tersebut pada penelitian ini dikembangkan sebuah prototipe penekan tombol otomatis simulator meter kWh untuk pengisian token berbasis mikroprosesor dan API Telegram. Untuk menguji keberhasilan prototipe tersebut, dirancang sebuah alat simulator meter kWh, yang selanjutnya dikenal sebagai simulator. Alat ini merupakan perangkat elektronik sederhana yang dibangun menggunakan Arduino, dilengkapi dengan tombol mekanik yang dirancang sedemikian rupa agar menyerupai tombol pada meter kWh. Sedangkan prototipe menggunakan komputer sirkuit tunggal jenis *Raspberry*, yang sudah memiliki fitur terintegrasi yang cukup lengkap, terutama dalam hal komunikasi melalui jaringan nirkabel (wifi) agar lebih mudah diintegrasikan dengan API Telegram [8]. Salah satu contoh penggunaan *Raspberry* dalam memonitor ketinggian air [9]. API Telegram merupakan jembatan komunikasi antara prototipe penekan tombol dengan pengguna melalui telepon genggam [10]. Selain itu, penggunaan IoT dengan *Raspberry* telah dilakukan dalam bidang lain yaitu untuk pemantauan cuaca [11], monitoring kualitas air [12], monitoring ketinggian air menggunakan Telegram [13] dan penggunaan Telegram pada otomasi rumah tinggal [14]. Selain fitur komunikasi yang lengkap, *Raspberry* juga memiliki pin yang dapat dimanfaatkan untuk mengontrol solenoid ketika diperintahkan menekan tombol pada proses pengisian token. Pada penelitian ini masih terbatas pada pengisian token dengan hasil pengujian akurasi penekanan, waktu pengisian token, dan pengujian API Telegram. Dengan mengimplementasikan manajemen penggunaan energi listrik di rumah berbasis IoT seperti yang telah dipublikasikan [15], prototipe ini sangat membantu dalam pencatatan semua token yang telah digunakan untuk kWh tersebut.

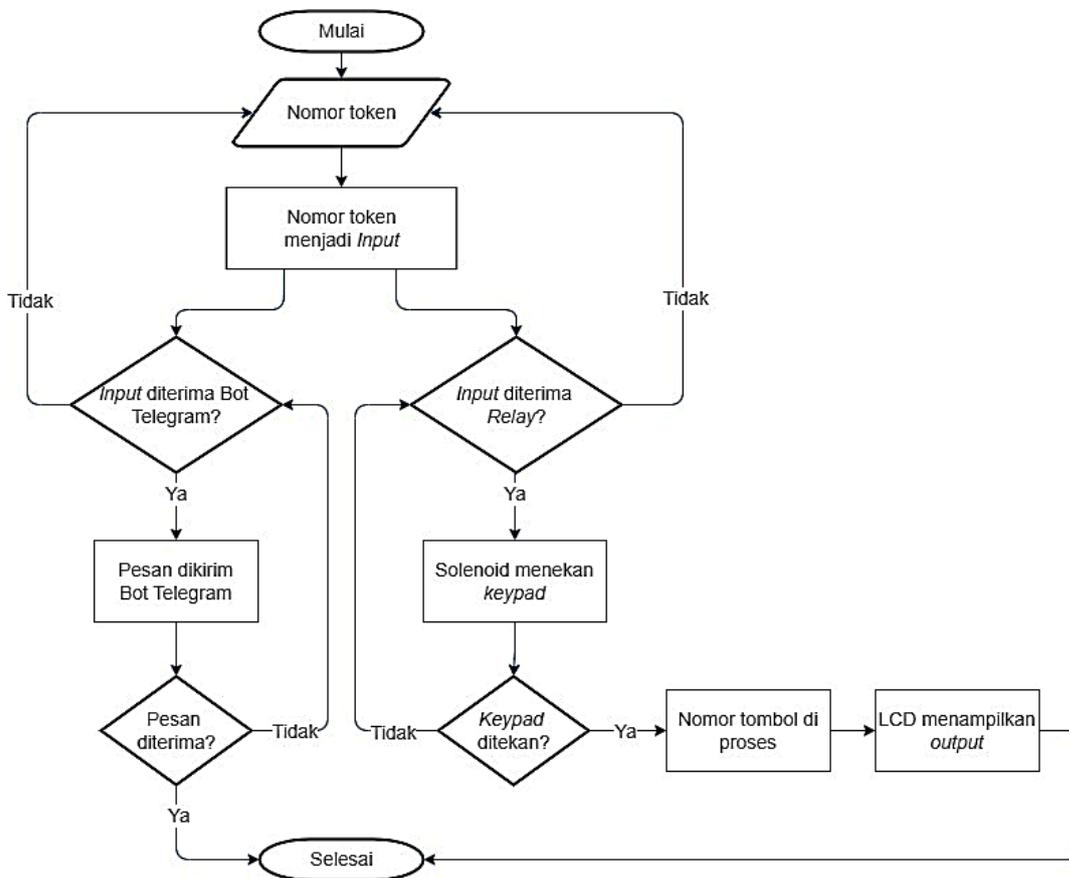
2. Metode

Rancang bangun prototipe ini mengikuti proses blok diagram seperti pada Gambar 1. Pengguna berkomunikasi dengan *Raspberry* melalui telepon genggam yang telah terinstall aplikasi Telegram. Telegram merupakan salah satu platform media sosial yang dapat dijalankan pada seluruh sistem operasi, baik bersifat *mobile* maupun komputer personal. Telegram menyediakan modul gratis berupa API yang dapat berkomunikasi ke semua platform tersebut. Telegram dapat menjembatani komunikasi antara pengguna dengan telepon genggamnya dan *Raspberry* yang terhubung ke internet. Token yang dikirim oleh pengguna diterima oleh *Raspberry* dan diproses dengan bahasa pemrograman *Python* untuk mengaktifkan *relay* dan mengeksekusi solenoid.



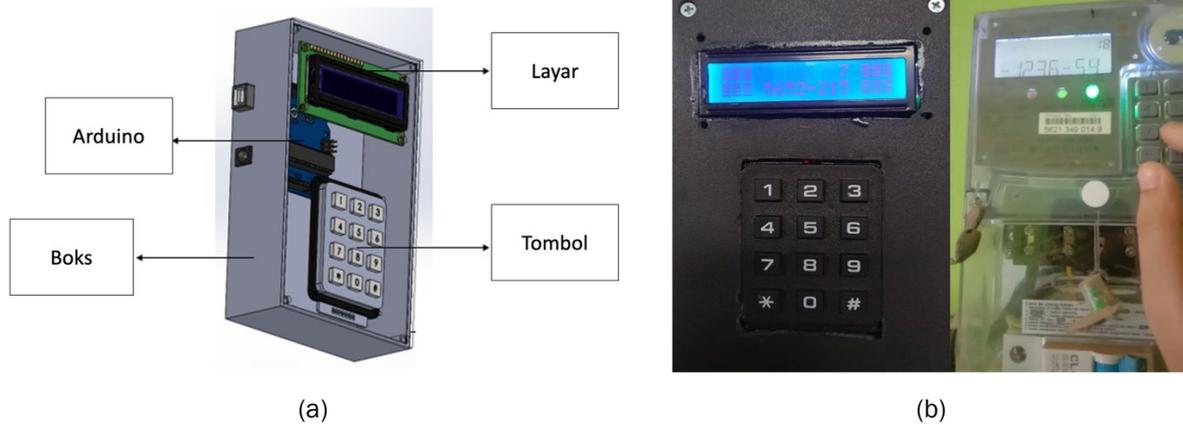
Gambar 1. Blok diagram prototipe.

Adapun prinsip kerja prototipe ini diawali dengan input token melalui aplikasi Telegram sebanyak 20 digit. Selanjutnya dari 20 digit itu diproses oleh bot Telegram satu per satu. Ketika 1 digit token dikirim, maka pada saat yang sama service yang terdapat di *Raspberry* memproses untuk setiap digitnya melalui relay yang terhubung langsung dengan solenoid sampai selesai dan dilanjutkan dengan digit berikutnya. Setiap digit tertentu sudah diarahkan ke solenoid tertentu juga. Untuk memahami lebih lengkap proses kerjanya dapat dilihat pada Gambar 2 dalam bentuk *flowchart*.



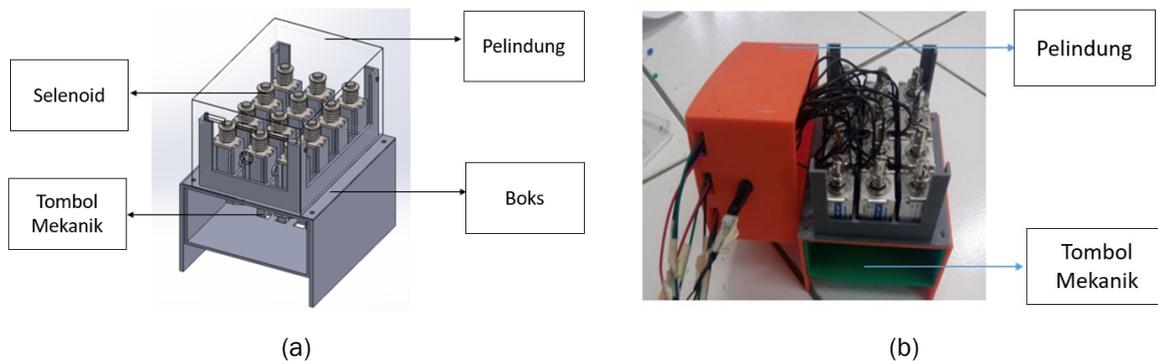
Gambar 2. Flowchart prinsip kerja.

Secara garis besar terdapat 2 buah alat yang dibangun, yaitu: simulator dan prototipe penekan tombol. Simulator menggunakan Arduino yang ditambahkan *display* dan *keypad*, dibuat sedemikian hingga proses pengisian token sesuai dengan meter kWh prabayar yang sudah terpasang. Simulator ini dapat memperlihatkan fungsi meter kWh untuk pengisian 20 digit sebagai token yang diperlihatkan di *display* dan 1 buah tombol validasi (*enter*). Sedangkan prototipe penekan tombol dibuat menggunakan *Raspberry* yang ditambahkan solenoid. *Raspberry* merupakan sebuah unit komputer sirkuit tunggal yang dengan beberapa pin, yang dapat diprogram sesuai dengan kebutuhan. Solenoid berfungsi sebagai alat bantu penekan tombol *keypad* pada simulator. Pembuatan simulator bertujuan untuk memudahkan dalam pengujian prototipe. Prototipe diletakan tepat di atas simulator dengan satu tombol *keypad* ditekan oleh satu solenoid. Pada Gambar 3.b terdapat sebuah simulator yang disandingkan dengan meter kWh yang sudah terpasang dengan jaringan listrik. Simulator ini tidak terhubung dengan sumber listrik dan hanya dilengkapi dengan *keypad* serta *display*.



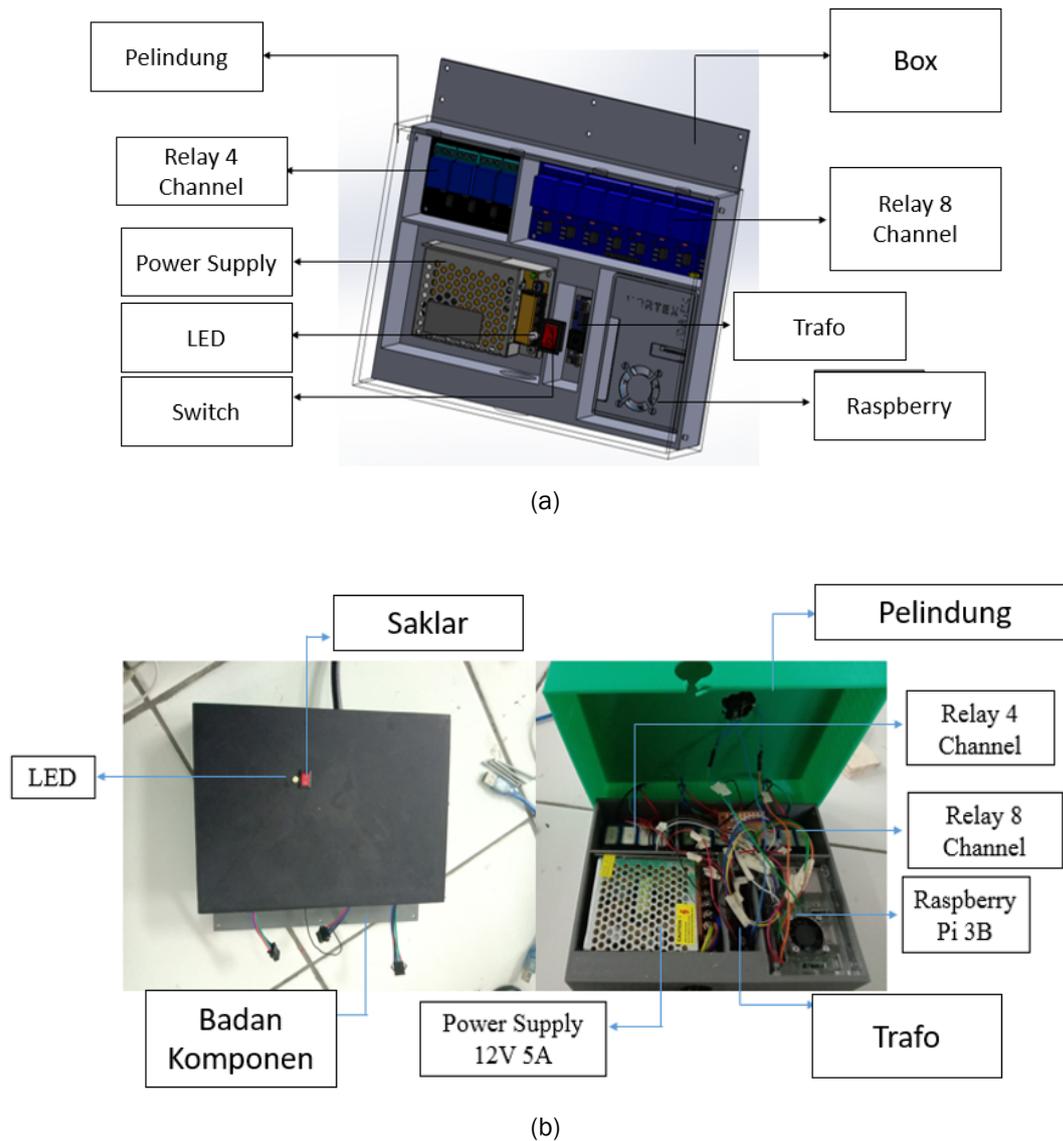
Gambar 3. kWh meter simulator (a) Ilustrasi desain 3D (b) Perbandingan proses pengisian token kWh simulator dengan kWh prabayar yang telah terpasang.

Gambar 3.a merupakan simulator yang diilustrasikan dengan gambar 3 dimensi, sedangkan Gambar 3.b merupakan proses penyalarsan fungsi pengisian token pada simulator dengan meter kWh prabayar. Prototipe penekan otomatis dibagi menjadi 2 bagian, yaitu bagian penekan yang didalamnya 12 solenoid dan bagian yang terdiri dari *Raspberry*, *relay*, dan sumber tegangan DC.



Gambar 4. (a) Desain 3D boks solenoid (b) Tampilan Boks Solenoid.

Gambar 4 memperlihatkan salah satu bagian penting dari prototipe, yaitu bagian pertama yang berupa penekan tombol *keypad* di meter kWh simulator. Gambar 4.a merupakan ilustrasi gambar tempat solenoid dan pada Gambar 4.b merupakan implementasi ilustrasi gambar 3 (tiga) dimensi menjadi sebuah prototipe. Sedangkan bagian kedua adalah komponen relay yang jumlahnya disesuaikan dengan jumlah solenoid yaitu sebanyak 12 buah. Pada bagian ini juga terdapat *Raspberry* sebagai komponen utamanya. Untuk lebih jelasnya dapat bagian kedua dapat dilihat pada Gambar 5.



Gambar 5. (a) Desain 3D boks solenoid (b) Aktualisasi boks kontrol.

3. Hasil dan Diskusi

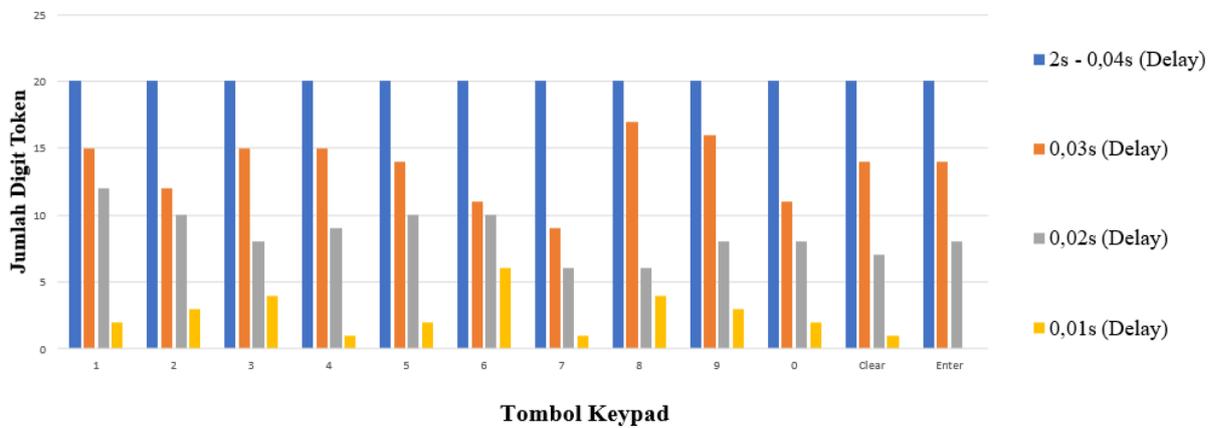
Pengujian simulator dilakukan dengan membandingkan fungsi kerja antara simulator dengan meter kWh yang sudah terpasang pada saat dilakukan pengisian token. Gambar 1.b menampilkan perbandingan penunjukan saat tombol *enter* ditekan pada simulator dengan meter kWh. Indikasi tampilan “berhasil” pada meter kWh adalah kesesuaian kombinasi dan jumlah angka yang dimasukkan, sementara itu untuk simulator indikasi tampilan “berhasil” tergantung pada jumlah angka yang dimasukkan, apabila kurang dari 20 angka maka akan tampil “gagal”, sedangkan untuk kombinasi tidak diikuti sertakan dalam indikasi “gagal” dan “berhasil”. Hasil Pengujian untuk penunjukan saat *enter* ditekan antara simulator dengan meter kWh terdapat kesamaan dalam segi tampilan.

Untuk prototipe dilakukan beberapa pengujian. Pengujian pertama yaitu pengujian akurasi penekanan yang dilakukan dengan cara mengirimkan pesan melalui Telegram yang berisikan 20 digit angka yang sama secara berurutan pada angka 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, C dan E. Contohnya ketika mengirimkan angka 1 yang diulang sebanyak 20 kali, seperti 11111111111111111111 (20 digit). Pengujian ini juga dilakukan dengan cara menggunakan *delay* yang berbeda untuk setiap digit, yaitu: 0,01 s, 0,02 s, 0,03 s, 0,04 s dan 2,00 s. Antara *delay* 0,04 s sampai 2,00 s dilakukan pengujian untuk setiap 0,01 s penambahannya. Namun, hasil akurasi dari 0,04 s sampai 2,00 s mendapatkan hasil akurat yang sama yaitu sebesar 100%. Pengujian ini dilakukan untuk menguji kemampuan respon terendah dari *Raspberry* dalam mengeksekusi solenoid untuk menekan tombol. Hasil pengujian dapat dilihat pada Tabel 1.

Tabel 1. Rerata akurasi untuk setiap *delay*.

<i>Delay</i> (s)	Rerata Akurasi (%)
2,00	100
0,04	100
0,03	68
0,02	43
0,01	12

Pada Tabel 1 di atas dapat dilihat bahwa respon terendah dari *Raspberry* untuk mengeksekusi solenoid yaitu pada 0,04 s dengan tingkat keberhasilan 100%. Jika *Raspberry* menggunakan *delay* < 0,04 s untuk setiap pengiriman digitnya, maka didapatkan hasil akurasi yang bervariasi di bawah 100%. Hal ini disebabkan oleh keterbatasan sistem mekanik yang dihubungkan dengan solenoid dan *relay*. Untuk itu *Raspberry* yang digunakan dapat menghasilkan akurasi penekanan 100% jika dalam kode *Python* dengan *delay* terendah 0,04 s waktu terendahnya.



Gambar 6. Grafik perbandingan akurasi untuk setiap *delay*.

Pada Gambar 6 dapat dilihat bahwa, setiap tombol *keypad* yang ditekan sebanyak 20 kali, dari tombol 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, *Clear* dan *Enter* dikirim melalui *Telegram* dengan *delay* yang berbeda, keberhasilan penekanan untuk 20-digit terjadi pada *delay* > 0,04 s. Jika *delay* < 0,04 s untuk 20 kali penekanan ternyata yang berhasil terkirim kurang dari 20-digit. Pengujian kedua yaitu membandingkan waktu pengisian token prototipe yang dilakukan secara otomatis melalui telepon genggam dengan waktu pengisian token meter kWh secara manual. Manual disini yaitu menekan 20-digit token langsung ke meter kWh yang telah terpasang. Pada pengujian ini dilakukan dengan cara menekan 20 angka secara acak yang diulang sebanyak 10 kali dengan angka yang berbeda yang dilakukan pada simulator dan dibandingkan waktunya yang dimulai dari penekanan digit pertama hingga enter. Hasilnya dapat dilihat pada Tabel 2.

Tabel 2. Perbandingan Waktu Input Token Manual dengan Prototipe

No	Manual	Prototipe	
	Rerata Waktu (s)	Delay (s)	Rerata Waktu (s)
1		2,00	93,92
2		1,90	89,92
3		1,80	87,23
4		1,70	83,54
5		1,60	78,37
6		1,50	73,05
7		1,40	70,12
8		1,30	66,73
9		1,20	60,44
10		1,10	56,36
11		1,00	51,86
12		0,90	47,69
13	18,54	0,80	44,02
14		0,70	39,72
15		0,60	36,20
16		0,50	30,94
17		0,40	27,32
18		0,30	23,18
19		0,20	19,37
20		0,10	14,67
21		0,09	13,34
22		0,08	12,58
23		0,07	12,31
24		0,06	11,86
25		0,05	11,39

Pada Tabel 2 terlihat bahwa pengisian token manual rata-rata membutuhkan waktu kurang lebih 18,54 s dengan 10 kombinasi 20-digit token yang dilakukan oleh satu orang. Sebaliknya, untuk prototipe dengan pengisian token otomatis, total waktu pengisian berkisar antara 11,9 s untuk *delay* 0,05 s hingga 93,92 s untuk *delay* 2 s. Jika dibandingkan dengan pengisian manual, waktu pengisian prototipe lebih cepat dibandingkan manual pada *delay* 0,04 s sampai *delay* 0,1 s dengan akurasi tetap 100%.

4. Kesimpulan

Prototipe dan simulator berhasil dibuat dan berfungsi dengan baik. Simulator berfungsi sebagai alat bantu uji prototipe. Pengisian token secara manual rata-rata membutuhkan waktu selama 18,54 s yang dilakukan oleh satu orang dengan 10 kombinasi 20-digit token. Sedangkan untuk prototipe yang pengisian tokennya secara otomatis, total waktu pengisian dari 11,9 s untuk *delay* 0,05 s sampai dengan 93,92 s untuk *delay* 2 s. jika Dibandingkan dengan manual agar waktu pengisian prototipe lebih cepat dengan manual, maka *Raspberry* pada program *Python* menggunakan waktu *delay* tercepat antara 0,04 s sampai 0,1 s. Sehingga *delay* tercepat yang dapat digunakan yaitu 0,04 s agar hasilnya tetap di akurasi 100%.

Referensi (Sub Bab)

- [1] O. S. Omole, D. Akpobasah, and A. A. Atayero, "Development of a smart, low-cost and iot-enabled system for energy management," in *3rd International Conference on African Development Issues (CU-ICADI 2016)*, 2016.
- [2] S. Priya, A. Srivastava, S. Kumar Jindal, and S. Kumar Sahoo, "Design and implementation of a smart energy meter based on internet of things and neural network approach," in *3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT)*, 2018.
- [3] K. Ragul, M. Mukul, P. Vijay, and P. Rajkumar, "Design and implementation of smart energy meter using IoT," *International Research Journal of Engineering and Technology*, 2020, [Online]. Available: www.irjet.net
- [4] A. Yadhav and V. Sarma, "GSM based smart energy meter," *International Research Journal of Modernization in Engineering Technology and Science @International Research Journal of Modernization in Engineering*, vol. 202, pp. 2582–5208, 2020, [Online]. Available: www.irjmets.com
- [5] N. Al Ofi, A. Azeez, F. N. Hasoon, and S. Manic, "Design and implementation of an energy meter system for optimized cost using Internet of Things (IoT) Technology Administration," *Applied Computing Journal*, vol. 1, no. 1, pp. 55–65, 2021, [Online]. Available: <http://www.aaaa-p.com/index.php/acj>,
- [6] Y. Novriandry, D. Triyanto, and Suhardi, "Prototype sistem monitoring dan pengisian token listrik Prabayar menggunakan Arduino Uno berbasis website," *Jurnal Komputer dan Aplikasi*, pp. 61–72, 2020.
- [7] M. R. Nikhil et al., "IoT and GSM based smart energy meter," *INTERNATIONAL JOURNAL OF INNOVATIONS IN ENGINEERING RESEARCH AND TECHNOLOGY [IJERT]*, vol. 6, no. 2, 2019.
- [8] H. Dipak Ghael, L. Solanki, G. Sahu, and A. Professor, "A review paper on Raspberry Pi and its applications," *International Journal of Advances in Engineering and Management (IAEM)*, vol. 2, p. 225, 2008, doi: 10.35629/5252-0212225227.
- [9] M. A. Karegar, J. Kusche, F. Geremia-Nieviniski, and K. M. Larson, "Raspberry Pi Reflector (RPR) : a low-cost water-level monitoring system based on GNSS interferometric reflectometry," *Water Resour Res*, vol. 58, no. 12, Dec. 2022, doi: 10.1029/2021WR031713.
- [10] D. Andrian Nugraha and Amirullah, "Telegram application for monitoring, controlling and protecting the consumption of household electrical appliances," *Jurnal Teknik Elektro*, vol. 15, no. 1, pp. 1–1110, 2023.
- [11] D. N. Bestari and A. Wibowo, "An IoT-Based real-time weather monitoring system using Telegram bot and thingsboard platform," *International Journal of Interactive Mobile Technologies*, vol. 17, no. 6, pp. 4–19, 2023, doi: 10.3991/ijim.v17i06.34129.
- [12] R. Bogdan, C. Paliuc, M. Crisan-Vida, S. Nimara, and D. Barmayoun, "Low-cost Internet-of-Things water-quality monitoring system for rural areas," *Sensors*, vol. 23, no. 8, Apr. 2023, doi: 10.3390/s23083919.
- [13] M. Safii et al., "Monitoring ketinggian permukaan air menggunakan Telegram bot berbasis NODEMCU ESP8266," *METIK JURNAL*, vol. 6, no. 2, pp. 123–132, Dec. 2022, doi: 10.47002/metik.v6i2.384.
- [14] S. Vivekanandan, S. Kumar, R. Srinath, and R. Vinayak, "Smart energy efficient home automation using Telegram," in *ITSPWC 2022*, European Alliance for Innovation n.o., Aug. 2022. doi: 10.4108/eai.14-5-2022.2318872.
- [15] F. Condon, J. M. Martínez, A. M. Eltamaly, Y. C. Kim, and M. A. Ahmed, "Design and implementation of a cloud-IoT-based home energy management system," *Sensors*, vol. 23, no. 1, Jan. 2023, doi: 10.3390/s23010176.