Design and Development of an ESP32-Based Battery Monitoring System for Container Cranes with Automatic Transfer Switching (ATS) Mechanism and Telegram Notifications

Lasykar Alim Perdana, Denny Irawan
Department of Electrical Engineering, Faculty of Engineering, Muhammadiyah University of Gresik
Department of Electrical Engineering, Faculty of Engineering, Muhammadiyah University of Gresik
E-mail: lasykarperdana17@gmail.com, den2mas@umg.ac.id

***Abstract* — The development of Internet of Things (IoT) technology has provided convenience in monitoring and controlling industrial systems, including container cranes that utilize batteries as the primary power source. This study aims to design and implement an ESP32-based battery monitoring system equipped with an Automatic Transfer Switch (ATS) mechanism and Telegram notifications. The system utilizes a voltage sensor and an ACS712 current sensor to monitor battery conditions in real time, while relay modules and contactors are employed to perform automatic power source switching when the main battery voltage decreases.The research methodology includes hardware design, software development, IoT system integration, and system performance testing. Voltage sensor testing was conducted using 30 sampling trials and resulted in an average error value of 0.72%, while current sensor testing also used 30 sampling trials and produced an average error value of 1.86%. The ATS system testing demonstrated that the power source switching process was successfully carried out automatically when the main battery voltage reached 11 V. In addition, the ESP32 successfully transmitted real-time Telegram notifications with a response time of approximately 2 seconds during the switching process. Based on the test results, the proposed system is capable of operating stably and effectively, supporting battery monitoring while maintaining power supply continuity.**

Keywords* — *Automatic Transfer Switch (ATS), ESP32, Internet of Things (IoT), Battery Monitoring, Voltage Sensor, Current Sensor, Telegram.

I. INTRODUCTION

Container cranes are lifting machines used to transfer containers from ships to land and from land to ships [1]. The implementation of ESP32-based monitoring technology is capable of improving the efficiency and reliability of automated systems in industrial environments because it enables real-time monitoring of system conditions [2]. Batteries or accumulators are crucial components in container crane systems that function as electrical energy storage devices in the form of chemical energy.

In conventional cranes, the batteries commonly used are lead-acid batteries consisting of lead plates and sulfuric acid electrolyte solution, with an operating voltage of approximately 12 volts. A 12V battery generally consists of six cells, where each cell produces approximately 2V [3]. A decline in battery performance can cause operational disturbances, such as reduced light intensity, slow starter response, and even system failure. If this condition is left unresolved, it may result in excessive load on the alternator, potentially leading to system damage [4].

An imbalance between battery usage and the charging supply from the alternator can also accelerate battery deterioration. Therefore, periodic monitoring and maintenance are required to ensure optimal battery performance and prevent potential damage [5].

Along with the advancement of technology, the Internet of Things (IoT) enables devices to be interconnected through internet networks, allowing monitoring and control processes to be carried out in real time and remotely [6]. The Internet of Things (IoT) has also continued to develop rapidly in the era of globalization and the Industrial Revolution 4.0, along with the increasing human need and capability to utilize technology [7].

II. MATERIALS AND METHODS

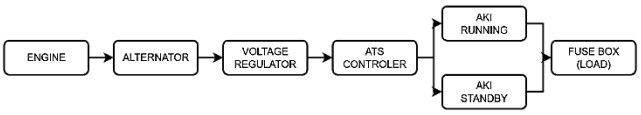


Fig 1. ATS Modeling

The process begins with the engine generating mechanical energy to drive the alternator, thereby producing AC voltage. This voltage is then converted into DC voltage and stabilized by a voltage regulator. Subsequently, the DC voltage is supplied to the ATS controller, which manages the selection of the power source between the running battery and the standby battery. The ATS itself is a development of the COS (Change Over System). The fundamental difference between the two systems lies in their operating methods. ATS operates automatically without manual intervention, whereas COS requires manual operation by an operator [8].

Under normal conditions, the system uses the running battery as the primary power source. If the voltage decreases, the ATS will transfer the power source to the standby battery to maintain the continuity of the electrical supply. The selected power is then distributed to the fuse box, which functions as both the power distribution center and load protection system.

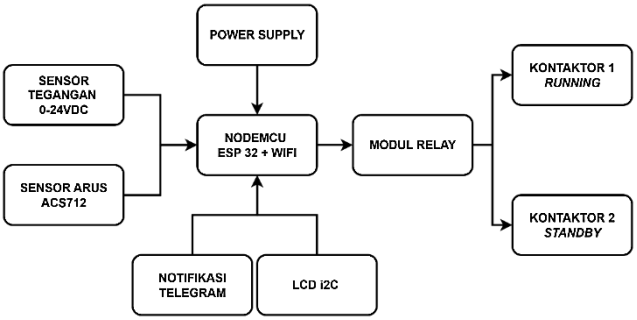


Fig 2. Block System

The system block diagram illustrates the relationship between the main components in the monitoring and Automatic Transfer Switch (ATS) system, where the ESP32 functions as the central controller connecting all devices. The ESP32 is equipped with a dual-core processor, large memory capacity, and support for various communication interfaces such as UART, SPI, I²C, ADC, and PWM. With its high computational capability, WiFi connectivity, and efficient power consumption, the ESP32 is well suited for monitoring applications, automatic control systems, and IoT-based systems [9]. This system utilizes a voltage sensor and an ACS712 current sensor to detect battery conditions in real time. The voltage sensor is used to measure electrical voltage values through a voltage divider circuit so that the signals can be processed by the microcontroller [10]. Meanwhile, the ACS712 current sensor is used to detect the magnitude of electric current flowing through the load and generate analog signals that can be read by the microcontroller for monitoring purposes [11]. The ACS712 sensor is capable of measuring both AC and DC currents by utilizing the magnetic field generated by the current flowing through a conductor [12].

The sensor reading data are then transmitted to the ESP32 to be processed and analyzed as the basis for decision-making. When the voltage reaches a predetermined threshold, the ESP32 activates the relay module to control the running contactor and standby contactor, enabling automatic power source transfer. A relay is an electrically controlled switch and is classified as an electromechanical component consisting of an electromagnet (coil) and mechanical switch contacts. Its operating principle utilizes electromagnetic force, where a low-power electric current is used to actuate the switch contacts, allowing higher-voltage currents to be connected or disconnected [13]. A DC magnetic contactor operates using an iron-core coil. When electric current flows through the coil, the iron core becomes magnetized, and the resulting magnetic force is used to pull the armature so that the contacts can automatically connect or disconnect [14].

In addition, the measurement results are displayed on a 16x2 I2C LCD for direct monitoring purposes. The I2C communication system consists of two channels, namely SCL (Serial Clock) and SDA (Serial Data), which function to transmit and receive data between I2C devices and the microcontroller [15]. The monitoring data are also transmitted to the Telegram application through a WiFi connection, allowing users to monitor system conditions remotely. Telegram can be utilized as a real-time monitoring and notification platform in IoT systems because it supports data communication through internet networks [16]. The entire circuit is powered by a 5V 10A power supply, which functions to rectify and filter the AC input voltage in order to produce a stable DC voltage for the system requirements.

A. Hardware Design

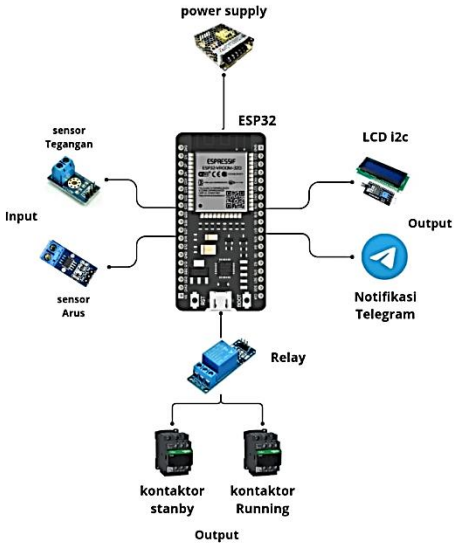


Fig 3. Hardware Design

The hardware design aims to ensure that each component within an electronic system can operate harmoniously, effectively, and stably according to its intended function.

This system utilizes the ESP32 as the main controller connected to voltage and current sensors to monitor battery conditions. The sensor data are processed by the ESP32 to determine the system status. The microcontroller also controls the relay module, which functions to activate the contactor for automatic power source switching. The power supply provides stable DC voltage to all components, while the monitoring results are displayed on the LCD and transmitted in real time to Telegram, enabling remote monitoring of battery conditions.

B. System Design



Fig 4. System Design

The device is designed within a panel box measuring approximately 30 × 40 cm, which functions as both a protective enclosure and an organized arrangement for all system components. Inside the panel, the components are

arranged systematically to facilitate installation, maintenance, and troubleshooting processes.

At the top section of the panel, a 5V 10A power supply is installed to provide DC voltage for the control system. Next to it is the ESP32, which serves as the main microcontroller and data processing center. The ESP32 receives input from the voltage sensor and ACS712 current sensor to monitor battery conditions in real time. The middle section of the panel consists of two Sneijer contactors that function as power switches in the Automatic Transfer Switch (ATS) system. These contactors are controlled by a relay module connected to the ESP32, allowing the power transfer process to be carried out automatically based on the detected voltage condition. At the front section of the system, a 16x2 I2C LCD is used to directly display voltage and current values, enabling users to monitor the system without additional devices. The bottom section of the panel is connected to two GS batteries as energy sources, namely the main battery (running) and the backup battery (standby). The load used in the testing process is a 12V 0.3A DC fan, which functions as a simulation of power consumption.

The system operates by reading the battery voltage and current through sensors installed on the main line. The collected data are processed by the ESP32 to determine the battery condition. When the main battery voltage reaches a predetermined threshold, the ATS system activates the relay to control the Sneijer contactor so that the power source is transferred to the backup battery. In addition, the system is capable of displaying real-time information on the LCD and sending notifications through Telegram [17].

III. RESULTS AND DISCUSSION

Data collection was carried out to evaluate the performance of the designed system, including the accuracy of sensor readings, the response of the Automatic Transfer Switch (ATS) system, and the successful delivery of notifications through Telegram. The testing process was conducted under several operational conditions using a 12V 0.3A DC fan as a simulation of a real load. The obtained data were then analyzed quantitatively by comparing the sensor readings with standard measuring instruments (multimeters), and qualitatively to assess the overall system performance.

A. Voltage Sensor Testing

Voltage sensor testing was carried out by comparing the sensor readings with measurements obtained from a multimeter connected in parallel to the battery. The purpose of this testing was to determine the accuracy level of the voltage sensor. To improve data stability, measurements were conducted using 30 sampling trials, and the average value was calculated.

The voltage reading process on the ESP32 was performed by reading the ADC value from the voltage sensor. The ADC value was then converted into voltage using Equation (1) below:

$$\mathrm{V_{modul}} = \frac{\mathrm{Value} \times 5}{1024} \quad (1)$$

After the module voltage value was obtained, the value was recalculated using the voltage divider equation in Equation (2) to determine the actual battery voltage:

$$Hasil = \frac{\mathrm{V_{modul}}}{\left(\frac{R1}{R1 + R2}\right)} \quad (2)$$

Description:
Value = ADC value obtained from the sensor reading
Vmodule = sensor output voltage
R1 & R2 = voltage divider resistors
Result = actual battery voltage

TABLE 1. VOLTAGE SENSOR TESTING

No	Multimeter Voltage (V)	Voltage Sensor (V)	Error (%)
1	12.60	12.58	0.16
2	12.60	12.46	1.11
3	12.60	12.50	0.79
4	12.60	12.44	1.27
5	12.60	12.59	0.08
6	12.60	12.47	1.03
7	12.60	12.55	0.40
8	12.60	12.45	1.19
9	12.60	12.52	0.63
10	12.60	12.60	0.00
11	12.10	12.08	0.17
12	12.10	11.97	1.07
13	12.10	12.02	0.66
14	12.10	11.95	1.24
15	12.10	12.09	0.08
16	12.10	11.96	1.16
17	12.10	12.04	0.50
18	12.10	11.98	0.99
19	12.10	12.10	0.00
20	12.10	11.99	0.91
21	11.50	11.48	0.17
22	11.50	11.37	1.13
23	11.50	11.42	0.70
24	11.50	11.35	1.30
25	11.50	11.50	0.00
26	11.00	10.98	0.18
27	11.00	10.87	1.18
28	11.00	10.93	0.63
29	11.00	10.86	1.27
30	11.00	11.00	0.00
Average			0.72%

Based on the results of 30 sampling trials, the voltage sensor demonstrated relatively stable readings with error values ranging from 0% to approximately 1.30%. The differences in readings were influenced by sensor tolerance, the stability of the ESP32 ADC, and the battery voltage condition during measurement. Although several reading deviations were observed, the average error value of 0.72% remained within an acceptable tolerance range, indicating that the voltage sensor is suitable for use in an ESP32-based battery monitoring system.

B. Current Sensor Testing

Current sensor testing was conducted by comparing the readings of the ACS712 current sensor with those obtained from a multimeter connected in series with the load. The purpose of this testing was to determine the accuracy level of the sensor in measuring current within the system. To obtain more stable data, measurements were carried out using 5 sampling trials, and the average value was calculated.

The current reading process was performed by reading the ADC value from the ACS712 sensor using the ESP32.

The ADC value was then converted into ADC voltage using Equation (3):

$$\mathrm{ADC}_{voltage} = ADC\ X\ \left(\frac{\mathrm{V}_{ref}}{4095}\right) \quad (3)$$

The ADC voltage value was then recalculated using the voltage divider circuit to obtain the sensor output voltage:

$$Sensor_{voltage} = \mathrm{ADC}_{voltage}\ X\ \left(\frac{R1 + R2}{R1}\right) \quad (4)$$

Afterward, the current value was calculated using the ACS712 sensor sensitivity equation in Equation (5):

$$Arus = \frac{Sensor_{voltage} - Offset}{Sensitivitas} \quad (5)$$

Description:

ADC = analog reading value from the ESP32
Vref = ESP32 reference voltage of 3.3 V
4095 = 12-bit ADC resolution of the ESP32
R1 & R2 = voltage divider resistors
Offset = sensor output voltage under no-load condition
Sensitivity = ACS712 sensitivity value

TABLE 2. CURRENT SENSOR TESTING

No	Test Load	Multimeter Current (A)	Current Sensor (A)	Error (%)
1	Lampu DC 12V	0.12	0.11	8.33
2	Lampu DC 12V	0.15	0.15	0.00
3	Fan DC 12V	0.28	0.27	3.57
4	Fan DC 12V	0.32	0.31	3.13
5	Charger HP	0.42	0.41	2.38
6	Charger HP	0.48	0.49	2.08
7	Lampu DC + Charger HP	0.57	0.56	1.75
8	Fan DC + Lampu DC	0.63	0.62	1.59
9	Fan DC + Charger HP	0.74	0.73	1.35
10	Fan DC + Charger HP	0.80	0.81	1.25
11	2 Lampu DC 12V	0.26	0.25	3.85
12	2 Fan DC 12V	0.61	0.60	1.64
13	2 Charger HP	0.91	0.89	2.20
14	Fan + Lampu DC + Charger	0.95	0.94	1.05
15	Fan + 2 Lampu DC	0.71	0.70	1.41
16	Fan + 2 Charger HP	1.18	1.17	0.85
17	2 Fan + Lampu DC	0.86	0.85	1.16
18	2 Fan + Charger HP	1.05	1.04	0.95
19	2 Fan + 2 Lampu	0.98	0.97	1.02
20	Lampu + 2 Charger HP	1.02	1.00	1.96
21	Fan + Lampu + 2 Charger	1.36	1.35	0.74
22	2 Fan + 2 Charger HP	1.52	1.50	1.32
23	3 Fan DC 12V	0.93	0.92	1.08
24	3 Lampu DC 12V	0.39	0.38	2.56
25	3 Charger HP	1.35	1.33	1.48
26	2 Fan + Lampu + Charger	1.24	1.23	0.81
27	2 Fan + 2 Lampu + Charger	1.48	1.46	1.35
28	Fan + 3 Charger HP	1.62	1.60	1.23
29	2 Fan + 3 Charger HP	1.87	1.85	1.07
30	2 Fan + 2 Charger + Lampu	1.74	1.72	1.15
Average				1,86%



Fig 5. Load Current Testing

Based on the results of 30 sampling trials using various loads such as a 12V DC lamp, a 12V DC fan, and a mobile phone charger, the ACS712 current sensor was able to measure current relatively stably and achieved an average error value of 1.86%. The differences in readings were influenced by the sensitivity of the ACS712 sensor, the stability of the ESP32 ADC, and current fluctuations in each load variation. Although several differences were observed between the sensor readings and the multimeter measurements, the obtained error values were still within an acceptable tolerance range. Therefore, the ACS712 sensor is considered suitable for current monitoring in the ESP32-based ATS system.

C. ATS System Testing

The ATS system testing was conducted to evaluate the system's capability to automatically transfer the power source based on the measured voltage value.

TABLE 3. ATS SYSTEM TESTING

No	Voltage (V)	ATS Status	Description
1	12.5	Baterai 1	Normal
2	11.8	Baterai 1	Normal
3	11.0	Baterai 1	Switch
4	12.2	Baterai 2	Stabil

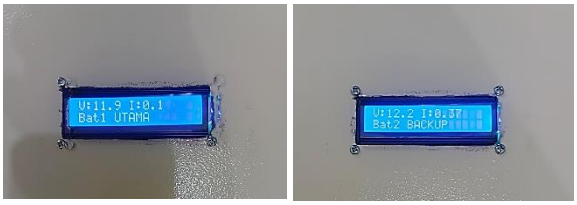


Fig 6. ATS Testing

The ATS system successfully performed power source transfer from the main battery to the backup battery when the voltage reached approximately 11V. This indicates that the switching system operated in accordance with the designed specifications.

D. Telegram Notification Testing

The testing was conducted to ensure that the system was capable of sending real-time notifications whenever an ATS switching event occurred.

TABLE 4. TELEGRAM NOTIFICATION TESTING

No	System Condition	Notification	Response Time
1	Sistem normal	Tidak ada	-
2	Tegangan ≤ 11V ATS aktif	Terkirim	2 detik

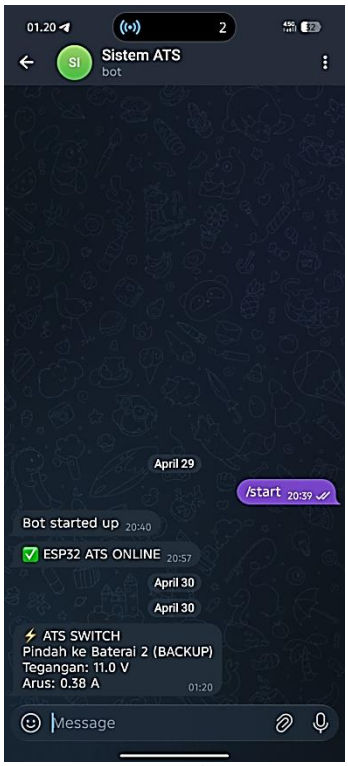


Fig 7. Telegram Notification Testing

The ESP32 successfully transmitted notifications to Telegram with an average response time of 2 seconds. This indicates that the IoT communication system operated properly and can be effectively utilized for remote monitoring.

IV. CONCLUSION

Based on the design, implementation, and testing results of the ESP32-based battery monitoring system with an Automatic Transfer Switch (ATS) mechanism and Telegram notifications, several conclusions can be drawn as follows:

1. The ESP32-based battery monitoring system successfully operated to monitor voltage and current in real time.
2. The voltage sensor testing conducted through 30 sampling trials resulted in an average error value of 0.72%, indicating good sensor accuracy.
3. The ACS712 current sensor testing using variations of DC fan and DC lamp loads resulted in an average error value of 1.86%, indicating that the sensor is still suitable for current monitoring applications.
4. The ATS system successfully performed automatic transfer from the main battery to the backup battery when the voltage reached 11 V.
5. The ESP32 successfully transmitted real-time Telegram notifications with an average response time of approximately 2 seconds during the ATS switching process.

REFERENCES

[1] P. H. TOGAR TIMOTEUS GULTOM, SUHELMI, "ANALISA DAYA LISTRIK MOTOR HOIST DAN MOTOR TROLLEY PADA CONTAINER CRANE DALAM PROSES," *R E L E (Rekayasa Elektr. dan Energi)*, vol. 6, no. 1, pp. 55–60, 2023.

[2] M. I. W. S. Danang Erwanto, Dian Efytra Yuliana, Yoga Pebri Aprilia, Ahfan Naofal, "IMPLEMENTASI IOT BERBASIS ESP32 UNTUK PENGATURAN VENTILASI RUANG KELAS SECARA OTOMATIS DAN EFISIEN," *J. ELKON*, vol. 5, no. 1, pp. 70–77, 2025.

[3] D. I. Jihan Arganatta Nugraha, "RANCANG BANGUN INKUBATOR TELUR UNGGAS BERTENAGA HYBRID MENGGUNAKAN PERHITUNGAN OTOMATIS," *J. Media Teknol.*, vol. 12, no. 1, pp. 137–143, 2025.

[4] I. Prasetyo and I. Saputro, "Perbaikan dan perawatan aki basah," *SURYA Tek.*, vol. 2, no. 2, pp. 16–21.

[5] I. M. A. Ahmad Fahrizal Fahmi, "Alat pendeteksi kerusakan cell accu," *JRM*, vol. 06, no. 03, pp. 1–7, 2021.

[6] R. F. A. Tundo, Sodik, Kiki Setiawan, "Penerapan IoT dengan Algoritma Fuzzy dan Mikrokontroler ESP32 dalam Monitoring Penyiraman," *J. Indones. Manaj. Inform. dan Komun.*, vol. 5, no. 3, pp. 2915–2924, 2024.

[7] D. Hidayat and I. Sari, "MONITORING SUHU DAN KELEMBABAN BERBASIS INTERNET of THINGS (IoT)," *J. Penelit. Tek. Inform. Univ. Prima Indones. Medan*, vol. 4, no. April, pp. 525–530, 2021.

[8] A. Wisnu, A. Kristanto, S. T. Elektro, F. Teknik, and U. N. Surabaya, "Pemodelan Automatic Transfer Switch (ATS) Pada System Smartgrid Pembangkit Photovoltaic," *J. Tek. Elektro*, vol. 11, pp. 351–360, 2022.

[9] C. E. Savitri and N. P. Is, "Sistem Monitoring Parkir Mobil berbasis Mikrokontroller Esp32," *J. Ampere*, vol. 7, no. 2, pp. 135–144, 2022.

[10] S. I. A. Erick Radwitya, "PERANCANGAN ALAT MONITORING TEGANGAN DAN ARUS LISTRIK BERBASIS MIKROKONTROLER ARDUINO UNTUK RUMAH," *J. SIMETRIK (Sipil, Mesin, List.*, vol. 16, no. 1, pp. 43–52, 2026.

[11] M. Hasanuddin, A. Wiguna, and M. F. Aditia, "Pengembangan Sistem Pemantauan Beban Listrik Menggunakan Sensor Berbasis Arduino untuk Industri Skala Kecil," *J. Electr. Eng. Res.*, vol. 1, no. 1, pp. 1–9, 2025.

[12] D. Mohammad Yasin, Esa Apriaskar, "Simulasi Monitoring Arus, Tegangan, dan Daya Panel Surya," *Emit. J. Tek. Elektro Simulasi*, vol. 23, no. 2, pp. 87–92, 2023.

[13] R. M. W. N. Slamet Purwo Santosa and Abstrak, "RANCANG BANGUN ALAT PINTU GESER OTOMATIS MENGGUNAKAN MOTOR DC 24 V," *J. Ilm. Elektrokrisna*, vol. 9, no. 1, pp. 38–45, 2021.

[14] D. L. K. Kalika Khaldan Nurshofa, Aldi Fadillah Ramadan, Nur Fajri Faiz, Paul Manurung, "Penggunaan Kontaktor pada Sistem Pengaman Motor Induksi 3 Phasa," *J. Tek. Mesin, Ind. Elektro dan Inform.*, vol. 3, no. 4, pp. 280–289, 2024.

[15] N. A. Istiqomah and R. N. Rohmah, "Sistem Pengurasan Akuarium Otomatis Berdasarkan Suhu dan Kekeruhan Dilengkapi Monitoring Jarak Jauh,"

vol. 17, no. 2, pp. 2580–2582, 2024.

[16] R. Gustia, M. Hasanah, W. Febriani, and A. Fradana, “Perancangan Smart Home Berbasis Iot Menggunakan Esp32 , Telegram dan Spreadsheet,” *J. Pus. AKSES Kaji. Teknol. Artif. Intell.*, vol. 5, no. 3, pp. 695–700, 2025.

[17] A. B. Amin *et al.*, “MONITORING DAN PENGENDALI KUALITAS AIR PADA TAMBAK UDANG SECARA OTOMATIS BERBASIS TELEGRAM,” *J. Tek. Elektro dan Inform.*, vol. 18, no. 2, pp. 45–51, 2023.