EMITOR: Jurnal Teknik Elektro

Design of an Arduino-Based Inverse Type Overcurrent Relay

Verra Aulia, Masing, Bella C. N, I Wayan E. S., Marselino Pakorong*, Priyo H. C. Ali
Department of Electrical Engineering – Samarinda State Polytechnic
Samarinda, Indonesia
*marselino161@gmail.com

Abstract – *An overcurrent relay is an important device in protection systems to protect electrical equipment from damage caused by excessive overcurrent. The objective of this study is to design and implement an overcurrent relay system that can detect and mitigate overcurrent using the inverse time protection principle, developed using the popular and accessible Arduino platform. An overcurrent relay (OCR) is an electrical protection device that operates based on overcurrent detection. There are two main characteristics, namely inverse time and constant time. Inverse Time Relay cuts off overcurrent with an operating time that decreases as the current increases, while Constant Time Relay has a fixed operating time. This research discusses the working principles, differences in characteristics, and applications of both types of relays to improve the effectiveness of protection in electrical power systems.*

Keywords – *Arduino; overcurrent relay; inverse time protection; constant time relay; electrical protection system.*

I. Introduction

IN electric power systems, continuity of supply and operations are two very important aspects to maintain. Disturbances such as overcurrent can cause damage to equipment, power outages, and even potential safety hazards such as fire. Therefore, a reliable protection system is needed to quickly and accurately detect and isolate compromised parts of the system.

One of the most widely used protection devices for detecting overcurrent is the Over Current Relay (OCR). OCR works on the principle of detecting current exceeding normal limits and activating circuit disconnection to protect electrical installations from further damage. Over Current Relays are indeed a critical and widely used protection device in electrical systems, designed to detect and interrupt excessive currents to prevent equipment damage. Multiple studies [1, 2] consistently confirm OCRs as the most economical and essential component in power system protection. These relays operate by monitoring current levels and automatically tripping circuit breakers when current exceeds predetermined thresholds. For instance, [2] demonstrated that OCRs can work in two primary modes: instantaneously disconnecting circuits when current immediately exceeds maximum safe levels, and providing time-delayed protection for sustained overcurrent conditions. The reliability of OCRs has been validated across various technological implementations, including FPGA, CMOS, and microcontroller-based designs, underscoring their versatility and critical role in electrical system safety. In its application, OCR can be classified based on its time characteristics into two main types, namely Inverse Time and Constant Time Over Current Relay. The evidence for this classification comes primarily from the research question itself, with supporting context from the literature. The study [3] confirms that time overcurrent devices have "inverse time-current characteristics," meaning greater fault currents result in shorter operating times. While the sources extensively discuss inverse time characteristics—such as standard, very, and extremely inverse types— [4, 5] there is limited explicit discussion of the Constant Time type in the provided sources. More research would be needed to fully characterize the Constant Time OCR's specific properties.

The Inverse Time Over Current Relay has the characteristic that the greater the current disturbance, the faster the relay will work. The Inverse Time Over Current Relay definitively works faster with greater current disturbances, a characteristic confirmed by multiple research sources. The evidence is strong and consistent across studies. The classic IEEE paper [3] explicitly

The manuscript was received on November 5, 2025, revised on November 8, 2025, and published online on November 28, 2025. Emitor is a Journal of Electrical Engineering at Universitas Muhammadiyah Surakarta with ISSN (Print) 1411 – 8890 and ISSN (Online) 2541 – 4518, holding Sinta 3 accreditation. It is accessible at https://journals2.ums.ac.id/index.php/emitor/index.

states that "greater the fault currents shorter are their operating times." Moreover, [6] further elaborates that very inverse type overcurrent relays have "a longer trip delay time for smaller fault currents and faster at large currents." This inverse time-current characteristic means the relay's response is dynamically proportional to the current magnitude: as the fault current increases, the relay's operating time decreases proportionally. This design allows protective systems to respond more quickly to severe electrical disturbances, minimizing potential damage to electrical infrastructure, as demonstrated in [7]. The foundational principles of inverse time operation have been expanded through various hardware and algorithmic implementations, including Arduino-based systems [8], adaptive control approaches using PID [9], and ANN-based automation [10]. Additionally, experimental validation of three-phase inverse relay performance in practical laboratory environments [11] supports the theoretical understanding. Overall, the accumulated literature strongly reinforces that inverse-time characteristics are central to efficient and coordinated overcurrent protection in modern power systems.

This allows the protection system to adapt to the severity of the interference and prioritize the most affected parts of the system. This characteristic also supports the coordination of protection between relays in a single electric power system, so that interference can be addressed locally without affecting a larger area. Adaptive protection systems enable dynamic, localized response to electrical interference by intelligently adjusting relay settings in real-time. Multiple studies demonstrate this capability. The concept of adaptive relaying was first defined in [12], where it was described as utilizing the system's changing status for online relay setting adjustments. Later, [13] specifically showed how such systems can maintain relay coordination locally by controlling backup relay parameters, thereby preventing widespread disruption. Furthermore, [14] reinforced this by characterizing adaptive protection as a real-time mechanism for modifying protective actions based on evolving system conditions. The evidence spans multiple research efforts, providing robust support for the concept of localized and flexible protection strategies that can prioritize and mitigate system interference with minimal broader impact. Other studies also emphasize the role of automation, multi-agent coordination, and machine learning in enabling adaptive protection across modern smart grids [15–20].

In contrast, the Constant Time Over Current Relay operates with a fixed working time, regardless of the magnitude of the current interference. This type of relay is typically used in systems that require consistent uptime of operation, such as in protection on generators, motors, or other critical loads. Selecting the right type of OCR is essential to achieve the selectivity, sensitivity, and reliability of the protection.

In 2023, Galang Nazhrullah and Aria Kharisma explained that overcurrent protection systems play an important role in protecting electrical equipment from damage caused by overcurrent disturbances. In their research, they designed an Arduino Nano microcontroller-based protection system that uses a shunt sensor to detect current. The research method used was experimentation with a simulation and field test approach. The test results showed that the system was able to detect overcurrent in real time and provide a disconnection response with a low error rate of around 0.97%, as well as being able to control the relay to automatically disconnect the current [21].

In 2024, Becky Arya Wijaya and colleagues presented the design of an Arduino-based inverse time overcurrent relay. In this study, a Time Multiplier Setting (TMS) value of 0.05 seconds was used to regulate the relay's operating time based on the magnitude of the fault current. An experimental approach was used in the testing, by assembling a complete circuit and measuring the relay operating time against overcurrent variations. The results showed that the system was able to work according to inverse time characteristics with an average cut-off time error of 2.07%. This system is considered feasible for application in medium-scale electrical load protection and as a learning medium [22].

In 2019, Aceng Daud designed an Arduino Uno-based overcurrent and short circuit protection system. This system utilizes an ACS712 current sensor to detect current and control the circuit breaker relay. The research method used was experimentation through testing the actual current and system response time to disturbances. The test results showed that the protection system was able to operate within a time range of 100–200 ms when detecting currents exceeding the threshold or short circuits. This system is considered effective as a practical tool for learning and field applications [23].

II. Research Methods

This study was carried out over a period of 16 weeks. Research activities began in February 2025 and ended in May 2025. This time span is used for all stages, from planning to evaluation of research results. The place where the research was carried out was at the Protection and Microprocessor Laboratory, Department of Electrical Engineering, Samarinda State Polytechnic.

i. Data Types and Sources

This study uses two types of data, namely primary data and secondary data. Primary data was obtained directly from the observation and testing of the overcurrent relay system, including current reading data by the PZEM-004T sensor, the relay working time, and the system's response to current changes and working modes. This data was collected through a series of experiments at the Protection and Microprocessor Laboratory of the Department of Electrical Engineering, Samarinda State Polytechnic.

Meanwhile, secondary data is taken from various reference sources such as books, journals, and standard documents related to Overcurrent Relay, inverse and constant characteristic data, and the application of Arduino UNO in the protection system. These references are used to strengthen the theoretical basis as well as testing in research.

ii. Block Diagram Image

This block diagram shows how the protection system works to protect the electrical load from overcurrent using two types of characteristics, namely standard inverse and constant time, with Arduino as the main controller. The system is equipped with a keypad for setting input, an LCD for data display, and a buzzer as an alarm. The power source consists of DC voltage for Arduino and AC voltage 220V for load. The data from CT is primary data, while the system configuration comes from user input as secondary data. The system is designed to provide automatic protection against overcurrent in an efficient and controlled manner. A block diagram can be seen in Figure 1.

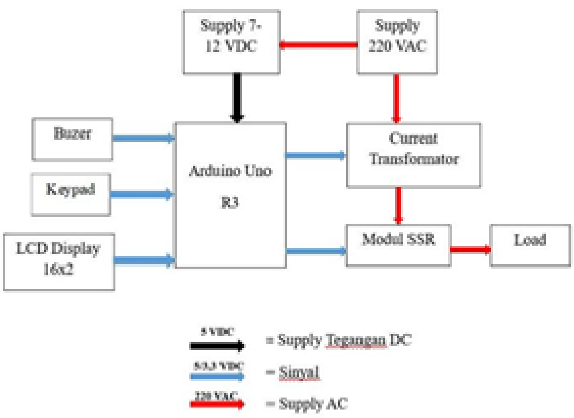

Figure 1: Block Diagram of the Arduino-Based Overcurrent Protection System

iii. Arduino Pin Configuration

Table 1 indicates the use of pins as inputs and outputs in the overcurrent protection system. Pins 2 through 9 are used to read the input from the 4x4 keypad, pin 10 for the buzzer as an alarm, and pins 11–12 for communication with the PZEM-004T module as the current meter. Pin 13 is used to control the Solid State Relay (SSR), while the SDA and SCL pins are used as outputs to display data on the 16x2 I2C LCD. This configuration supports automatic and real-time monitoring and control of the system.

Table 1: Input and Output Pins on Arduino Uno

No.	Pin	Pin Mode	Component	Information
1	2–9	Input	Keypad 4x4	Used to connect keypad keys 4x4 as user input
2	10	Input	Buzzer	Used as a relay off marker
3	11&12	Input & Output	PZEM-004T	Communication with PZEM-004T v3.0 module (current meter)
4	13	Output	Solid State Relay (SSR)	Controls SSR as an electronic switch
5	SDA & SCL	Output	LCD I2C 16x2	Displays system information

iv. Flowchart

The flowchart in Figure 2 shows the workflow of a microcontroller-based overcurrent protection system with two working modes: inverse time and constant time. The system starts with the setting of the nominal current and the type of relay, which is displayed on the LCD. After that, the system reads the load current. If the current exceeds the specified limit, the system calculates the relay's operating time according to the selected type—using the inverse time formula. Once the time is met, the SSR will close to break the load. The system will return to normal once the reset button (#) is pressed, or the SSR can be manually reopened with the A button.

The flowchart starts with "Start" and continues to "Nominal Current Setting," which defines the current threshold in the system. Then, the process moves to "Display Nominal on LCD," showing the current value on the LCD. Following this, the relay operates between "Open SSR" and "Close SSR" depending on current conditions. The process also includes reading the reset button status — "Button (A) is Pressed" or "Button (#) is Pressed" — which resets or reactivates the system accordingly. This structure ensures clear logic control and reliable coordination between sensing, decision-making, and relay operation in real-time protection scenarios.

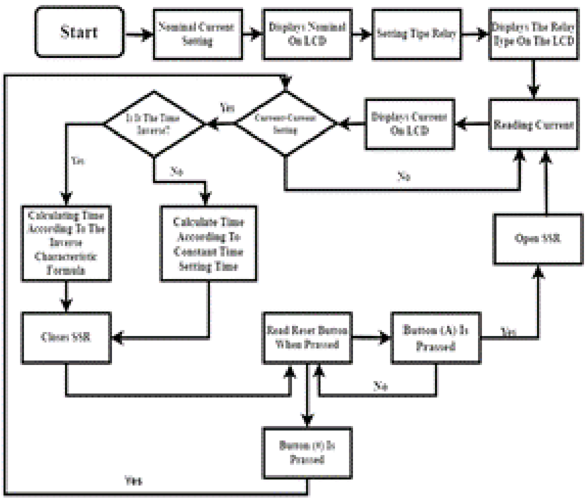

Figure 2: Flowchart of the Overcurrent Protection System Workflow

v. *Surveillance Diagram*

A wiring diagram, also known as a circuit diagram, is a visual or schematic representation of a wiring system or electrical circuit. This diagram shows how electrical components are connected and interact in a system. Wiring diagrams are used to make it easier to understand, design, install, maintain, and repair electrical systems. A diagram of the wiring system can be seen in Figure 3.

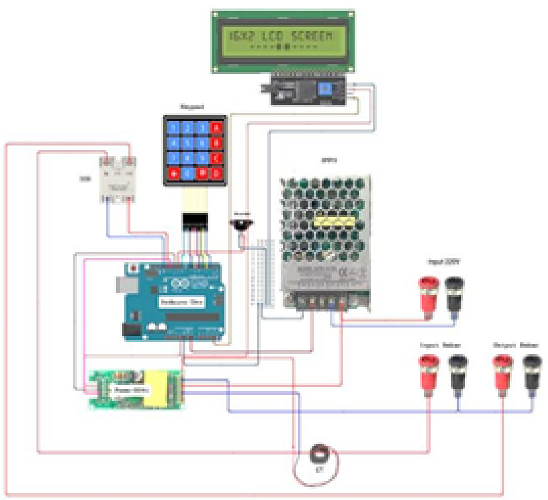

Figure 3: Wiring Diagram of the Arduino-Based Overcurrent Protection System

This diagram shows the Arduino Uno-based current monitoring and protection system consisting of a 16x2 LCD, a 4x4 keypad, PZEM-004T module, a current transformer (CT), a solid state relay (SSR), and a switched-mode power supply (SMPS). The Arduino reads the electrical parameters of the load using PZEM and CT, and then displays them on the LCD. Users can set the current limit and protection mode via the keypad. If the current exceeds the limit, the Arduino activates the SSR to break the load. The SMPS provides the power supply for the entire system.

III. Results and Discussion

i. *Solid State Relay (SSR) Testing*

The purpose of this test is to determine whether the Solid State Relay (SSR) is still in normal condition. The way to determine whether the SSR is in normal condition is by sending a signal to the relay. If the SSR is given a HIGH signal, it will operate, indicated by the SSR indicator light turning on and current flowing to the load. Meanwhile, if the SSR is given a LOW signal, it will not operate, indicated by the indicator light turning off and almost no current flowing.

Testing the SSR with a HIGH signal can be seen in Figure 4.

Figure 4 shows the SSR test given a HIGH signal. This test shows that when the SSR receives a HIGH signal, it works properly, as indicated by the indicator light turning on and the current flowing to the load.

Furthermore, the SSR test when given a LOW signal can be seen in Figure 5.

Figure 5 shows the testing of the SSR when given a LOW signal. This test shows that when the SSR is given a LOW signal, it does not operate, as indicated by the indicator light turning off and the current being very small or practically nonexistent, namely 0.003 A. These results confirm that the SSR operates normally according to input signal logic.

ii. *Inverse Time Tables and Graphs*

In electrical power protection systems, one of the commonly used methods to protect equipment from overcurrent is the inverse time method. This method works on the principle that the greater the fault current, the faster the protection system must disconnect the circuit. This is done to minimize equipment damage and maintain system stability.

The inverse time characteristic is influenced by two main parameters, namely the current-to-nominal current ratio (I/I_{set}) and the Time Multiplier Setting (TMS). The current ratio indicates how much the current exceeds the normal limit, while TMS functions as a time multiplier that regulates the sensitivity level of the protection system. The combination of these two parameters determines how long the delay time is before the system disconnects the current.

The results of the inverse time characteristic testing for TMS 0.1 can be seen in Table 2.

Figure 4: Solid State Relay (SSR) testing with a HIGH signal

Figure 5: Solid State Relay (SSR) testing with a LOW signal

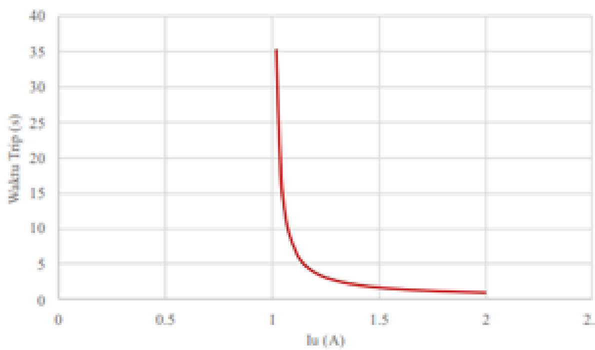

Figure 6: Graph of inverse time characteristic test results on TMS 0.1

Table 2 shows the results of testing the inverse time characteristics at TMS 0.1. Based on the test results, the average error obtained was 1.04%. These results indicate that the designed device behaves according to the inverse time characteristics for TMS 0.1.

The graph of the inverse time characteristic test results at TMS 0.1 can be seen in Figure 6.

Figure 6 shows a graph of the inverse time characteristic test results on TMS 0.1. The figure displays the relationship between the current ratio and trip time in a system using the TMS 0.1 setting. The test shows that as the current ratio increases, the relay trip time decreases, confirming that the protection system follows the inverse time characteristic accurately.

Table 3 shows the results of testing the inverse

Table 2: Results of Inverse Time Characteristic Testing on TMS 0.1

No	I_s (A)	I_u (A)	Calculated Trip Time (s)	Test Trip Time (s)	Error (%)
1	1	1.02	35.34	33.54	5.09
2	1	1.04	17.84	17.34	2.80
3	1	1.06	12.01	11.45	4.66
4	1	1.08	9.09	9.09	0.00
5	1	1.10	7.34	7.41	0.95
6	1	1.12	6.17	6.03	2.27
7	1	1.14	5.34	5.40	1.12
8	1	1.16	4.71	4.67	0.85
9	1	1.18	4.22	4.25	0.71
10	1	1.20	3.83	3.75	2.09
11	1	1.22	3.51	3.47	1.14
12	1	1.24	3.25	3.19	1.85
13	1	1.26	3.02	2.89	4.30
14	1	1.28	2.83	2.80	1.06
15	1	1.30	2.66	2.56	3.76
16	1	1.32	2.51	2.48	1.20
17	1	1.34	2.38	2.37	0.42
18	1	1.36	2.27	2.23	1.76
19	1	1.38	2.17	2.14	1.38
20	1	1.40	2.07	2.05	0.97
21	1	1.42	1.99	1.97	1.01
22	1	1.44	1.91	1.89	1.05
23	1	1.46	1.84	1.83	0.54
24	1	1.48	1.78	1.76	1.12
25	1	1.50	1.72	1.71	0.58
26	1	1.52	1.66	1.64	1.20
27	1	1.54	1.61	1.61	0.00
28	1	1.56	1.57	1.55	1.27
29	1	1.58	1.52	1.51	0.66
30	1	1.60	1.48	1.47	0.68
31	1	1.62	1.44	1.43	0.69
32	1	1.64	1.41	1.40	0.71
33	1	1.66	1.37	1.37	0.00
34	1	1.68	1.34	1.34	0.00
35	1	1.70	1.31	1.32	0.76
36	1	1.72	1.28	1.27	0.78
37	1	1.74	1.26	1.26	0.00
38	1	1.76	1.23	1.23	0.00
39	1	1.78	1.21	1.20	0.83
40	1	1.80	1.18	1.17	0.85
41	1	1.82	1.16	1.16	0.00
42	1	1.84	1.14	1.13	0.88
43	1	1.86	1.12	1.12	0.00
44	1	1.88	1.10	1.10	0.00
45	1	1.90	1.08	1.08	0.00
46	1	1.92	1.07	1.07	0.00
47	1	1.94	1.05	1.05	0.00
48	1	1.96	1.03	1.03	0.00
49	1	1.98	1.02	1.02	0.00
50	1	2.00	1.00	1.00	0.00
				Average Error (%)	1.04

Table 3: Results of Inverse Characteristic Testing on TMS 0.2

No	I_s (A)	I_u (A)	Calculated Trip Time (s)	Test Trip Time (s)	Error (%)
1	1	1.02	70.68	70.00	0.96
2	1	1.04	35.68	33.96	4.82
3	1	1.06	24.01	20.37	15.16
4	1	1.08	18.18	16.71	8.09
5	1	1.10	14.67	14.01	4.50
6	1	1.12	12.34	12.15	1.54
7	1	1.14	10.67	10.34	3.09
8	1	1.16	9.42	9.56	1.49
9	1	1.18	8.44	8.36	0.95
10	1	1.20	7.66	7.56	1.31
11	1	1.22	7.03	6.86	2.42
12	1	1.24	6.49	6.45	0.62
13	1	1.26	6.04	5.98	0.99
14	1	1.28	5.66	5.60	1.06
15	1	1.30	5.32	5.23	1.69
16	1	1.32	5.03	4.95	1.59
17	1	1.34	4.77	4.65	2.52
18	1	1.36	4.54	4.49	1.10
19	1	1.38	4.33	4.24	2.08
20	1	1.40	4.15	4.07	1.93
21	1	1.42	3.98	3.89	2.26
22	1	1.44	3.83	3.78	1.31
23	1	1.46	3.69	3.65	1.08
24	1	1.48	3.56	3.50	1.69
25	1	1.50	3.44	3.42	0.58
26	1	1.52	3.33	3.31	0.60
27	1	1.54	3.23	3.19	1.24
28	1	1.56	3.13	3.10	0.96
29	1	1.58	3.05	3.00	1.64
30	1	1.60	2.96	2.94	0.68
31	1	1.62	2.89	2.84	1.73
32	1	1.64	2.82	2.77	1.77
33	1	1.66	2.75	2.73	0.73
34	1	1.68	2.68	2.65	1.12
35	1	1.70	2.62	2.59	1.15
36	1	1.72	2.57	2.55	0.78
37	1	1.74	2.51	2.49	0.80
38	1	1.76	2.46	2.45	0.41
39	1	1.78	2.41	2.41	0.00
40	1	1.80	2.37	2.36	0.42
41	1	1.82	2.32	2.31	0.43
42	1	1.84	2.28	2.28	0.00
43	1	1.86	2.24	2.24	0.00
44	1	1.88	2.20	2.20	0.00
45	1	1.90	2.17	2.17	0.00
46	1	1.92	2.13	2.13	0.00
47	1	1.94	2.10	2.10	0.00
48	1	1.96	2.07	2.07	0.00
49	1	1.98	2.04	2.04	0.00
50	1	2.00	2.01	2.01	0.00
				Average Error (%)	1.59

time characteristics on TMS 0.2. Based on the test results, the average error obtained was 1.59%. These results indicate that the designed device follows the inverse time characteristics accurately for TMS 0.2.

The graph of the inverse time characteristic test results for TMS 0.2 can be seen in Figure 7.

Figure 7 shows the graph of the inverse time characteristic test results on TMS 0.2. The figure presents the correlation between the current ratio and trip time, where the protection system demonstrates the inverse relationship—trip time decreases as the current increases. The consistent downward trend confirms the reliability of the inverse time principle implemented in the Arduino-based protection system.

Table 4 shows the results of testing the inverse

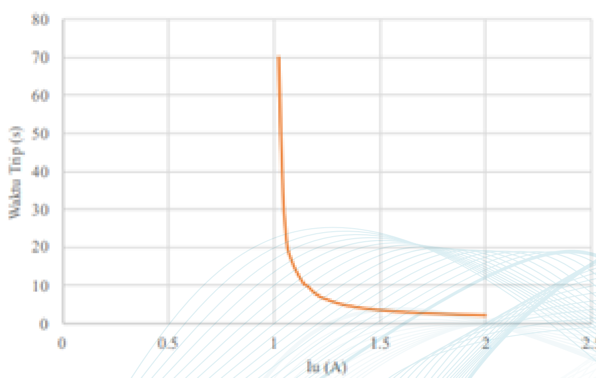

Figure 7: Graph of Inverse Time Characteristic Test Results on TMS 0.2

time characteristics on TMS 0.3. Based on the test results, the average error obtained was 1.41%. These results indicate that the designed device is in good accordance with the inverse time characteristics for TMS 0.3.

The graph of the inverse time characteristic test results on TMS 0.3 can be seen in Figure 8.

Figure 8 illustrates the inverse relationship between the current ratio and relay trip time at TMS 0.3. As the multiple of pickup current increases, the trip time decreases, confirming that the Arduino-based relay closely follows the standard inverse time characteristic and maintains consistent coordination performance across the tested current range.

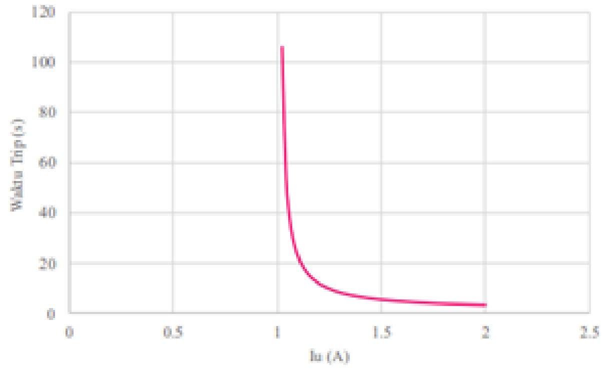

Figure 8: Graph of Inverse Time Characteristic Test Results on TMS 0.3

Table 4: Results of Inverse Characteristic Testing on TMS 0.3

No	I_s (A)	I_u (A)	Calculated Trip Time (s)	Test Trip Time (s)	Error (%)
1	1	1.02	106.03	92.00	13.23
2	1	1.04	53.52	53.01	0.95
3	1	1.06	36.02	35.83	0.53
4	1	1.08	27.27	26.43	3.08
5	1	1.10	22.01	21.01	4.54
6	1	1.12	18.51	18.21	1.62
7	1	1.14	16.01	15.59	2.62
8	1	1.16	14.13	13.59	3.82
9	1	1.18	12.67	12.23	3.47
10	1	1.20	11.50	11.14	3.13
11	1	1.22	10.54	10.42	1.14
12	1	1.24	9.74	9.63	1.13
13	1	1.26	9.07	8.91	1.76
14	1	1.28	8.49	8.41	0.94
15	1	1.30	7.98	7.86	1.50
16	1	1.32	7.54	7.46	1.06
17	1	1.34	7.15	7.00	2.10
18	1	1.36	6.81	6.73	1.17
19	1	1.38	6.50	6.41	1.38
20	1	1.40	6.22	6.15	1.13
21	1	1.42	5.97	5.93	0.67
22	1	1.44	5.74	5.68	1.05
23	1	1.46	5.53	5.46	1.27
24	1	1.48	5.34	5.28	1.12
25	1	1.50	5.16	5.09	1.36
26	1	1.52	4.99	4.94	1.00
27	1	1.54	4.84	4.80	0.83
28	1	1.56	4.70	4.73	0.64
29	1	1.58	4.57	4.54	0.66
30	1	1.60	4.45	4.41	0.90
31	1	1.62	4.33	4.29	0.92
32	1	1.64	4.22	4.19	0.71
33	1	1.66	4.12	4.08	0.97
34	1	1.68	4.03	4.05	0.50
35	1	1.70	3.94	3.95	0.25
36	1	1.72	3.85	3.87	0.52
37	1	1.74	3.77	3.77	0.00
38	1	1.76	3.69	3.71	0.54
39	1	1.78	3.62	3.62	0.00
40	1	1.80	3.55	3.57	0.56
41	1	1.82	3.49	3.49	0.00
42	1	1.84	3.42	3.42	0.00
43	1	1.86	3.36	3.36	0.00
44	1	1.88	3.31	3.33	0.60
45	1	1.90	3.25	3.27	0.62
46	1	1.92	3.20	3.25	1.56
47	1	1.94	3.15	3.17	0.63
48	1	1.96	3.10	3.14	1.29
49	1	1.98	3.05	3.03	0.66
50	1	2.00	3.01	3.00	0.33
				Average Error (%)	1.41

IV. Conclusion

Based on the results of the research and discussions, it can be concluded that the implementation of overcurrent protection in an Arduino-based current monitoring system operates effectively in accordance with the three applied protection modes: Normal mode, Inverse Time mode, and Constant Time mode. The system is capable of disconnecting the load when the measured current exceeds the specified maximum limit, with trip times that match the designed protection logic.

The use of the PZEM-004T sensor is proven to be effective for current measurement and overcurrent detection, while the LCD successfully displays real-time system information. The Solid State Relay (SSR) responds correctly to control signals, ensuring reliable load disconnection during fault conditions. In addition, the protection system can be reset via keypad input after a trip event, providing practical and user-friendly operation. Overall, the results show that the proposed Arduino-based overcurrent relay system is feasible to be applied as a low-cost, educational, and functional protection solution in electrical power systems.

References

[1] P. Kumar, V. Kumar, and R. Pratap, "Digital design and implementation of an overcurrent relay on fpga," *2017 14th IEEE India Council International Conference (INDICON)*, pp. 1–5, 2017. [Online]. Available: https://api.semanticscholar.org/CorpusID:52960505

[2] A. A. Hameed, A. J. Sultan, and M. F. Booneya, "Design and implementation a new real time overcurrent relay based on arduino mega," *IOP Conference Series: Materials Science and Engineering*, vol. 881, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:243192740

[3] "Computer representation of overcurrent relay characteristics," *IEEE Transactions on Power Delivery*, vol. 4, no. 3, pp. 1659–1667, 7 1989. [Online]. Available: http://dx.doi.org/10.1109/61.32656

[4] P. Kumar, V. Kumar, and R. Pratap, "Digital design and implementation of an overcurrent relay on FPGA," in *2017 14th IEEE India Council International Conference (INDICON)*. IEEE, Dec. 2017, pp. 1–5. [Online]. Available: http://dx.doi.org/10.1109/INDICON.2017.8487732

[5] H. Prasetijo and D. T. Nugroho, "Overcurrent relays coordination: Comparison characteristics standard inverse, very inverse and extremely inverse," *Journal of Physics: Conference Series*, vol. 1367, no. 1, p. 012051, Nov. 2019. [Online]. Available: http://dx.doi.org/10.1088/1742-6596/1367/1/012051

[6] S. P. of Samarinda, S. P. of Samarinda, S. P. of Samarinda, S. P. of Samarinda, and S. P. of Samarinda, "Arduino-based overcurrent relay design with very inverse type," 2024. [Online]. Available: https://ijcsrr.org/wp-content/uploads/2024/11/41-2211-2024.pdf

[7] M. Ojaghi and R. Ghahremani, "Piece–wise linear characteristic for coordinating numerical overcurrent relays," *IEEE Transactions on Power Delivery*, vol. 32, no. 1, pp. 145–151, 2 2017. [Online]. Available: http://dx.doi.org/10.1109/TPWRD.2016.2578324

[8] S. Syukriyadin, I. D. Sara, and F. Mashuri, "Design of a digital standard inverse overcurrent relay using arduino uno as a microcontroller," in *2018 International Conference on Electrical Engineering and Informatics (ICELTICs)(44501)*. IEEE, 9 2018, pp. 45–50. [Online]. Available: http://dx.doi.org/10.1109/ICELTICS.2018.8548835

[9] A. A. Zainul, A. Ramasamy, I. Z. Abidin, and F. H. Nagi, "Overcurrent time delay determination using gain scheduled pid controllers," in *2009 3rd International Conference on Energy and Environment (ICEE)*. IEEE, 12 2009, pp. 89–93. [Online]. Available: http://dx.doi.org/10.1109/ICEENVIRON.2009.5398663

[10] K. Gulez, R. Yumurtaci, and M. Uzunoglu, "The control of inverse time-overcurrent relay by ann (artificial neural networks) for automation of a system," in *Proceedings of the 6th International Conference on Optimization of Electrical and Electronic Equipments*, vol. 1. IEEE, 1998, pp. 241–244. [Online]. Available: http://dx.doi.org/10.1109/OPTIM.1998.710480

[11] W. Wahyono, W. P. Widyaningsih, N. Pribadi, H. Nur, M. F. Kelfin, and F. Shafira, "Pengujian kinerja relay inverse time overcurrent 3 phasa untuk menunjang praktikum sistem proteksi," *Prosiding Seminar Nasional NCIET*, vol. 1, no. 1, dec 20 2020. [Online]. Available: http://dx.doi.org/10.32497/nciet.v1i1.119

[12] J. Codling, S. House, J. Joice, K. Labhart, J. Richards, J. Tenbusch, M. Tullis, T. Wilkerson, and N. Rostamkolai, "Adaptive relaying: A new direction in power system protection," *IEEE Potentials*, vol. 15, no. 1, pp. 28–33, 1996. [Online]. Available: http://dx.doi.org/10.1109/45.481373

[13] H. Bisheh, B. Fani, and G. Shahgholian, "A novel adaptive protection coordination scheme for radial distribution networks in the presence of distributed generation," *International Transactions on Electrical Energy Systems*, vol. 31, no. 3, 2021. [Online]. Available: http://dx.doi.org/10.1002/2050-7038.12779

[14] B. P. Poudel, A. Bidram, M. J. Reno, and A. Summers, "Zonal machine learning-based protection for distribution systems," *IEEE Access*, vol. 10, pp. 66 634–66 645, 2022. [Online]. Available: http://dx.doi.org/10.1109/ACCESS.2022.3184865

[15] H. Wan, K. K. Li, and K. P. Wong, "An adaptive multiagent approach to protection relay coordination with distributed generators in industrial power distribution systems," *IEEE Transactions on Industry Applications*, vol. 46, no. 5, pp. 2118–2124, 2010. [Online]. Available: http://dx.doi.org/10.1109/TIA.2010.2059492

[16] A. A. Shobole and M. Abafogi, "Adaptive protection in smart distribution networks: Coordination demonstration of multi-agent systems," in *2023 5th Global Power, Energy and Communication Conference (GPECOM)*, 2023, pp. 483–488. [Online]. Available: http://dx.doi.org/10.1109/GPECOM58364.2023.10175704

[17] V. Ferrari and Y. Lopes, "Dynamic adaptive protection based on iec 61850," *IEEE Latin America Transactions*, vol. 18, no. 7, pp. 1302–1310, 2020. [Online]. Available: http://dx.doi.org/10.1109/TLA.2020.9099773

[18] P. McLaren, H. Li, and E. Dirks, "Self adaptive impedance relays," in *Engineering Solutions for the Next Millennium. 1999 IEEE Canadian Conference on Electrical and Computer Engineering (Cat. No.99TH8411)*, vol. 3, 1999, pp. 1314–1318. [Online]. Available: http://dx.doi.org/10.1109/CCECE.1999.804883

[19] H. C. Kılıçkıran, H. Akdemir, İ. Şengör, B. Kekezoğlu, and N. G. Paterakis, "A non-standard characteristic-based protection scheme for distribution networks," *Energies*, vol. 11, no. 5, p. 1241, 2018. [Online]. Available: http://dx.doi.org/10.3390/EN11051241

[20] B. Chattopadhyay, M. Sachdev, and T. Sidhu, "An on-line relay coordination algorithm for adaptive protection using linear programming technique," *IEEE Transactions on Power Delivery*, vol. 11, no. 1, pp. 165–173, 1996. [Online]. Available: http://dx.doi.org/10.1109/61.484013

[21] G. Nazhrullah and A. Kharisma, "Relay proteksi arus lebih berbasis mikrokontroller arduino," *PoliGrid*, vol. 4, no. 1, pp. 32–40, Jun. 2023. [Online]. Available: https://doi.org/10.46964/poligrid.v4i1.9

[22] B. A. Wijaya, A. Ihsandi, O. Nainggolan, and P. Murdiyat, "Rancang bangun rele arus lebih tipe inverse time dengan time multiplier setting 0,05 detik berbasis arduino," *PoliGrid*, vol. 5, no. 1, pp. 27–33, Jun. 2024. [Online]. Available: https://doi.org/10.46964/poligrid.v5i1.38

[23] A. Daud, J. Teknik, and K. Energi, "Rancang bangun modul proteksi arus beban lebih dan hubung singkat," *Jurnal Teknik Energi*, vol. 9, no. 1, pp. 37–44, Nov. 2019.