

IoT-based Running Time Monitoring System for Machine Preventive Maintenance Scheduling

Erwin Sitompul^{1*)} and Agus Rohmat²⁾

^{1,2)} Study Program of Electrical Engineering, Faculty of Engineering,
President University, Cikarang, Indonesia

Corresponding Email: ^{*)} sitompul@president.ac.id

Abstract – Machines are valuable assets that need to be protected from damage and failure through proper maintenance measures. This paper proposes a system that automatically monitors the running time of machines and sends notifications regarding their preventive maintenance (PM) schedules. The system core consists of a programmable logic controller (PLC) and a human-machine interface (HMI). The HMI is connected to an online platform via an internet connection provided by a router so that the monitoring result can be accessed via an Android smartphone or laptop/PC. This IoT-based running time monitoring system (IRTMS) will be particularly helpful in implementation at a production site that consists of multiple various machines. The PM items of a machine may vary from cleaning, changing a single component, to an overhaul, each with a different time interval. The key results of using the IRTMS are that the user will have an overview of the PM schedules anytime and anywhere, and the preparation of material, components, or tools can be known ahead of time. For simulation purposes, a prototype was constructed by using components as used in the industrial real-life condition. Four output connections were provided to simulate the simultaneous monitoring of four machines. The IRTMS prototype is tested and completely successful in doing the running time monitoring, the running time reset, the PM notifications, and the remote access for monitoring and control.

Keywords: PLC, HMI, IoT, preventive maintenance (PM), online platform.

I. INTRODUCTION

Maintenance can be generally defined as the attention and care given to an asset to extend its useful lifetime. Naturally, there are no man-made objects that will not be damaged on use, but with correct and regular repair and treatment, their duration of functionality can be prolonged [1]. According to [2], maintenance includes all activities related to maintaining a certain level of availability and reliability of a system and its components and its ability to perform at a standard level of quality.

Based on a survey reported in [3], the percent of planned production time that is becoming downtime is 13.3 % on average. Machine failures and maintenance measures, whether before or after the failures, contribute to downtime. In manufacturing industries, the maintenance costs are estimated between 15 % and 70 %

of the cost of goods produced. It is estimated that approximately one-third of these costs are unnecessary and avoidable [4]-[6]. Thus, it is crucial to optimize the maintenance measures, so that they are not conducted too early or too late. By doing so, the downtime and the maintenance cost can be reduced, while at the same time the production efficiency can be increased.

Preventive maintenance (PM) is one type of maintenance performed at predetermined intervals, which can be based on production time or production cycle. However, up to 50 % of PM in manufacturing is applied unnecessarily [7]. This can be surely traced back to the inaccurate record of the above-mentioned production time or production cycle.

A qualitative study about the aspects of effective PM has been conducted in [8], where the root cause analysis of the ineffective PM in a semiconductor company is also presented. The study proposed that the maintenance schedule must be incorporated with the production schedule to achieve an efficient and effective manufacturing system.

The accuracy of PM requires the help of a controller so that it can be planned automatically without relying only on erroneous human calculation and memory. Robert *et al* in [9] highlight the implementation of the programmable logic controller (PLC) in a predictive maintenance scheme. A PLC-based control system with the supporting software and hardware is presented. Another use of a PLC in combination with a human-machine interface (HMI) for downtime recording is found in [10]. The motivation is to be able to perform get an accurate cost calculation due to downtime. Some problems regarding data recapitulation and historical storage are mentioned.

The aforementioned problems are nonexistent in the system proposed by [11], where a record system for machine usage time directly uses a PC as the HMI, with Arduino Mega as the microcontroller. However, although the total working time of an operator in various machines while doing a production task can be summarized and integrated for further PM maintenance purposes, the dedicated use of one PC is found impractical.

In this paper, a system to improve the accuracy of PM based on machine running time is proposed. The PM due is not be based on planned production time or projected production quantity, as it is normally practiced. On the

contrary, the due is based on the actual running time of the machine.

The proposed system is IoT-based and utilizes a PLC and an HMI as the core. The inputs of the PLC are connected to the power supply line of the machines. The state of this supply line is monitored and recorded to provide the running time of the machines. The running time monitoring can be done directly through the HMI with its touch screen, or remotely through an Android phone or a laptop/PC. The ethernet connection of the HMI enables it to be connected to a router. The internet connection of the router links the HMI over an online platform to the Android phone or the laptop/PC.

The proposed system is to be referred to as the IoT-based running time monitoring system (IRMTS). A prototype of the system is to be constructed for simulation and testing purposes. The prototype is expected to be able to monitor four machines simultaneously. The PM schedules of each machine, as a whole (by overhaul) or partial (by components or treatments), are to be kept by the IRMTS. It is expected to generate ahead-of-time reminders regarding every impending PM via email notification and HMI screen notification.

After the problem identification mentioned above, the ensuing steps of the research methodology are the study of maintenance measures and the formulation of the design specification. Next, the design will be implemented and tested for some operation scenarios. In the end, a conclusion will be drawn.

II. METHODOLOGY

A. Maintenance Measures

Maintenance measures can be in the form of various actions taken to protect or repair an asset so that it can operate in an acceptable condition.

After passing the period where early (infant mortality) failures may occur, the performance of equipment in general or a machine, in particular, deteriorates slowly through its operating time [2]. The graph can be seen in Figure 1 [12,13]. At a certain point P , the change in performance becomes detected. If no maintenance is given, the performance decreases further and reaches point F , where the functional failure occurs and the machine cannot work anymore.

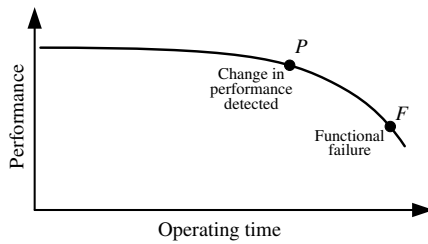


Figure 1. The performance of a machine as the function of operating hours

Long before point P , the predictive (condition-directed) and preventive (time-directed) maintenance can be taken

to improve the machine's performance. Between point P and F , the weight will be on preventive maintenance. By the time point F is reached, reactive maintenance, as a response to functional or catastrophic failure, is incurred.

The complexity of PM scheduling increases with the number of machines that are integrated into a production site. Every machine comes with a specific need of material, components, and tools, at a specific time of implementation. As an example, Table 1 shows the optimal PM schedule for the three most critical components of the turbines at Ombilin steam power plant owned by PLN (State Electricity Company) Indonesia [14]. The complete table is surely longer.

Due to this fact, an effective and efficient PM can only be done with the help of a computer. The PM can be automatically set to follow certain time intervals as required by the components. This is to be done based on the actual running time of the corresponding machine and not merely based on the number of days or planned production time.

Table 1. PM Schedule for Turbine's Critical Components at Ombilin Steam Power Generation

Item	PM Interval
Overhaul	8,000 hours
Membrane check	960 hours
Turning gear check	1,150 hours
Bearing check	908 hours
Membrane turbine replacement	3,410 hours
Turning gear replacement	4,500 hours
Bearing replacement	8,000 hours

B. Design Specification

1) Programmable Logic Controller (PLC)

A PLC is a ruggedized computer used for industrial automation of a part or an entire production process. As a digital controller, a PLC makes a logical decision based on the monitored inputs and outputs. The industrial applications require PLCs to be reliable, while easy to program and simple to diagnose.

In the development, the PLC's functionality is no longer limited to logical functions only. PLC can now carry out relatively complex arithmetic calculations and additional functions such as communication and documentation via the internet [15,16].

In this project, the HollySys Hollias LM3107E, as seen in Figure 2 is chosen. It has 14 digital inputs and 10 digital outputs, which are sufficient for the project at hand [17]. This PLC can be purchased at a reasonably low price and is shipped with free programming software.



Figure 2. HollySys Hollias LM3107E PLC

2) Human Machine Interface (HMI)

HMI is very useful to visualize the process in a form that can be easily understood by humans. Some functions of the HMI are monitoring plant conditions, setting parameter values, and giving actions following occurred circumstances. HMI can also become the media to show warning signs in the form of notifications. Data trajectories, whether real-time or historical, can also be displayed on an HMI.

HMI is also able to be used in programming and troubleshooting of the PLC programs. This is because the HMI can load applications that later on can be used to change or manipulate PLC programs or any other software programming from other sources.

The HMI Weintek MT8071iE1 is chosen due to its ability to connect with a lot of PLC types, including HollySys. This HMI, as shown in Figure 3, is equipped with 2 communication ports: RS 232 and RS 485 [18].



Figure 3. Weintek MT8071iE1 HMI

3) EasyAccess 2.0

A remote HMI access is conventionally a difficult intention. Traditionally, HMIs are designed to connect only to local industrial control equipment such as PLCs or other automation controllers. This is due to the required hardware interfacing, the network parameter settings, and the security considerations. The prior possible implementation allows only the connection with HMI at a time.

Weintek's EasyAccess 2.0 offers a new platform to access compatible HMIs from anywhere at any time, through an internet connection. The connection over this online platform makes the remote monitoring, troubleshooting, and reconfiguration of HMI and PLC easy, simple, and efficient. HMI projects can be updated via the internet connection, and later on, PLC programs can be passed through, updated, and monitored via ethernet connection [19]. The immediate benefits of using EasyAccess 2.0 include the significant reduction of downtime, technician's travel time, and labor costs. From this point on, the terms EasyAccess 2.0 and EasyAccess will be used interchangeably.

Figure 4 illustrates the connection using the EasyAccess system. The HMI and the laptop/PC are connected to a dedicated virtual private network (VPN). Thus, the data exchange is carried out in a peer-to-peer connection, without any need for firewalls.

4) Router and Relay

A router is hardware required in a computer network to route data packets from one network to another network.

Routers have become sophisticated nowadays, that their connection to the computers in the LAN can be established not only by using a cable connection but also with wireless technology.

Figure 5(a) shows the router chosen to be used in the proposed system, the MikroTik hAP-Lite wireless router [20]. This router is required to provide internet access to the HMI device so that the device can be accessed by EasyAccess. The router is connected to the HMI via an ethernet connection, while it is wirelessly connected to a mobile network to get internet access.



Figure 4. Secure connection with minimal setup and configuration using EasyAccess 2.0 [19]

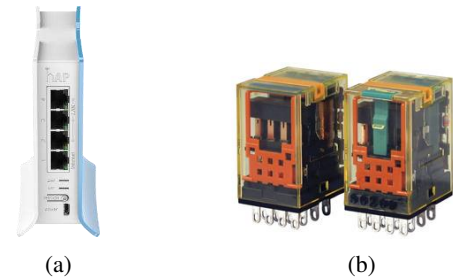


Figure 5. (a) MikroTik RB941-2nD (hAP-Lite) wireless router; (b) IDEC RU4S-NF-A220 relays

5) Relay

A relay is an electromagnetic switch that is operated by using a small electric current to control a much larger electric current or voltage. The switching mechanism in a relay includes an electromagnet that closes or opens the respective high-current or high-voltage circuit. As it leverages a weak input to a stronger output that powers a load, a relay can also be considered as an amplifier.

The proposed system is designated to control high-power machines such as a booster pump, compressor, turbine, and boiler. To simulate this, IDEC RU4S-NF-A220 relays are integrated into the proposed system.

The relays are shown in Figure 5(b). By using the relay, the PLC that sends the signal to the relay with 24 VDC and 260 mA will be able to power an electric load with 220 VAC and up to 6A [21].

6) Block Diagram

Figure 6 displays the block diagram of the proposed IRTMS. On the office side, the user uses an Android phone or a laptop/PC to connect to the EasyAccess by using an internet connection. On the factory side, the entire system is controlled by the HollySys PLC. The PLC is connected to 4 simulated machines, which are a booster pump, a compressor, a turbine, and a boiler.

The relays are deployed at the connections to the

booster pump and the compressor. This is to create a connection where these machines can be turned on and off manually (on-site) or remotely (via the internet). Whereas, the direct connection to the turbine and the boiler means that the proposed system only can monitor their running time. Both machines must be turned on or turned off manually (on-site).

The programming of the PLC is conducted by using the user's laptop/PC. The internet connection between the HMI and the EasyAccess platform is established by the MikroTik router.

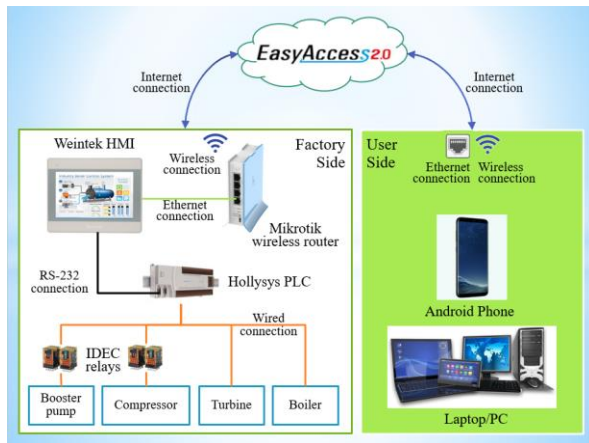


Figure 6. The block diagram of the proposed system

7) Schematic Design

The schematic design of the IRTMS is shown in Figure 7. The metal box is designed to be 20 cm deep, 40 cm wide, and 50 cm high. As can be seen in Figure 7(a), the HMI (1) is installed at the door, facing forward. Below it, a panel of two push-buttons (2) and two switches (3) is aligned. The push-buttons are used to simulate the on-site pushbuttons of the booster pump and the compressor. The switches are used to simulate the on-site switches of the turbine and the boiler.

If the door is opened, the arrangement is as shown in Figure 7(b). The PLC (4) is placed top-left. The router (5) is mounted below the PLC. Two control relays (6), R_1 and R_2 , are to be used in conjunction with two power relays (7), R_3 and R_4 . The terminal blocks (8) are used to secure and insulate the wire connections. The power socket (9) is required to power the router. Finally, a 6-A miniature circuit breaker or MCB (10) is installed to protect the whole circuit from overcurrent.

Further explanation regarding the control relays and the power relays will be given now. As mentioned before, the manual (on-site) turn on and turn off of all 4 machines are simulated by the 2 push-buttons and the 2 switches. The push-buttons and the switches are connected as the control signals to the PLC's inputs. If any of the 2 push-buttons is physically pressed, the status of the corresponding machine will be toggled between active and inactive. The PLC will toggle the power relays R_3 or R_4 , respectively, and the machine power supply is toggled, too.

For remote turn on and turn off (via HMI, Android phone, or laptop/PC), if the button of the booster pump or the compressor is pressed, the control relays R_1 or R_2 will be toggled. At the same time, R_1 or R_2 is used as the input

to the PLC that again toggles R_3 or R_4 . Thus, each pair of one push-button and one control relay forms one two-way switch.

In both cases, whether manual control or remote control, the IRTMS immediately starts to count the running time every time the status of a machine is active and stops when the machine is inactive. The counted time is to be saved in the PLC's memory.

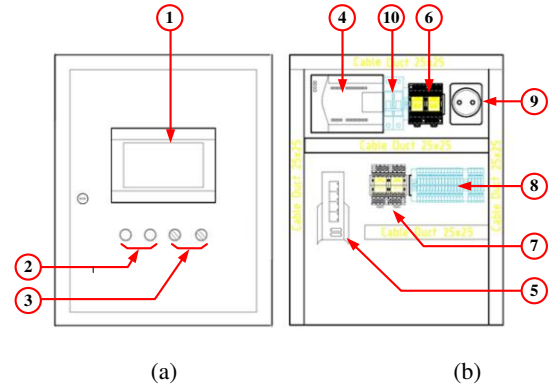


Figure 7. The schematic design of the proposed system
(a) front view; (b) inside view

C. Design Implementation

1) Flowchart

The flowchart of the IRTMS system is presented in Figure 8. After the start, the PLC checks the currently active machine(s), as indicated by the proper input signals. The running time of every active machine is added by 1 second after this time is elapsed. If the second count of an active machine reaches 3,600 seconds, then the respective hour-count is added by 1 hour, and the respective second count is reset to 0.

Then, the data of the remaining time until the next PM schedule will be updated. The HMI checks whether any of the remaining time is equal to 240 hours (10 days), 120 hours (5 days), or 72 hours (3 days). If it is the case, the HMI will one-time turn on an alarm sound for 60 seconds, display a screen notification, and send an email notification to a registered email address. Thus, by having such multiple reminders, the user can plan a future maintenance action with adequate time. The scope of the maintenance can also be informed in advance, so that the required material, components, or tools can be prepared.

The hour-count of the machines can also be reset to 0 by privileged users. This is normally done after completing a certain maintenance action. If the reset button is pressed and the authentication is successful, the desired hour-count can be selected and set to 0. Here, a cycle is completed and the process comes back to the PLC detecting the currently active machine(s).

2) The Programming of PLC

The programming of the PLC is conducted by using the Codesys Power Pro V2.1.2B software. Within one project file for the IRTMS, every machine obtains a separate program page. This modular approach makes the program troubleshooting and expansion easy. The addition of new machines can be accommodated by creating additional

program pages.

Figure 9 shows the main program page, PLC_PRG, which is used for the booster pump. The program page Time_Count (FB) is the program that counts the running time of the machines. This program consists of 10 networks, which is later compiled and becomes 1 function block (FB). Each FB is adjusted to the number of PM items that are being monitored from a certain machine. In this work, this number ranges from 3 to 5.

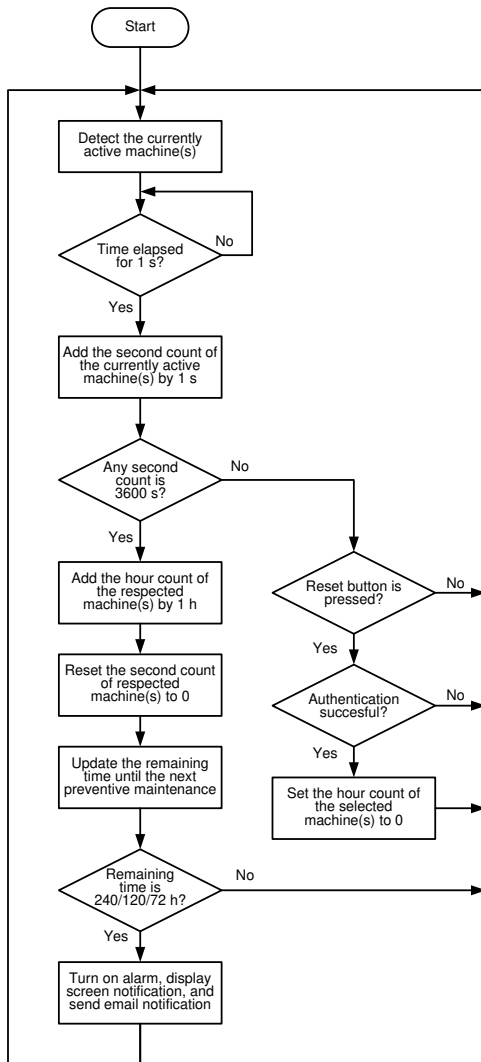


Figure 8. Flowchart of the running time monitoring system

3) The Programming of HMI

The programming of the HMI is done by using Weintek's EasyBuilder Pro V60001267 software. For the first step, the communication between the HMI and the PLC is constructed by matching the parameters of both devices. Figure 10 shows, that to do this, the *Home* menu bar (1) must be selected. Then, after clicking the *System Parameters* (2), a dialog box appears. At *Local PLC 1*, the PLC type is to be set to HollySys LE/LM (3). Afterward, the communication type is to be set to RS-232. The serial port is set to COM 1, with the rate of 38,400 baud.

Extensive programming work follows. This is to set up the HMI so it can fulfill the objective of the IRTMS. The

HMI program is expected to read the running time data from the PLC's memory and to display it in such a way that the HMI becomes an informative and user-friendly interface.

The programming includes the settings of the various user classes (Figure 11). The user classes are useful to assign certain privileges such as resetting the counted running time, setting/resetting of the set-point for a certain PM item of a machine, or merely able to turn on and turn off the machines. Three user classes are created: *Management*, *Technician*, and *Operator*.

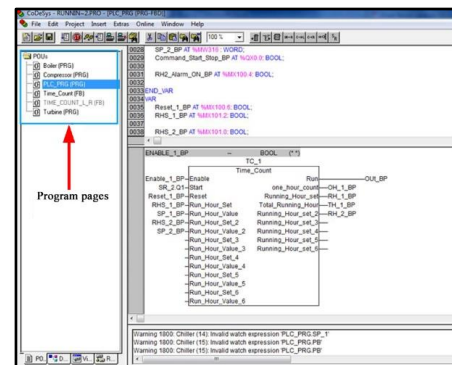


Figure 9. PLC program pages in Codesys

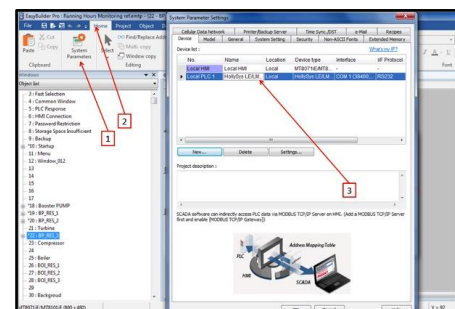


Figure 10. Initial parameter adjustments between the HMI and the PLC in EasyBuilder Pro

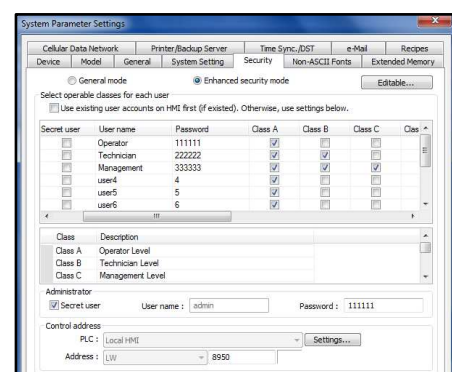


Figure 11. The setting of the various user classes

4) Constructed System

The realization of the schematic design is shown in Figure 12. In Figure 12(a), the green lamp indicates that the booster pump is turned on, while the red lamp does the same for the compressor.

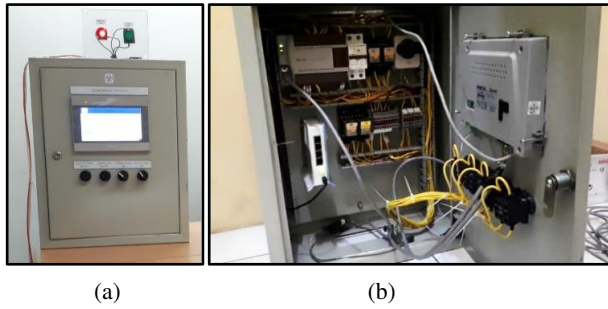


Figure 12. The schematic design realization of the proposed system (a) front view; (b) inside view

The main screen of the HMI is shown in Figure 13. All machines are depicted, with red indicator lamps next to each of them. If a machine is currently turned on, then the corresponding indicator lamps will light up.

If the button below a certain machine is pressed, then the corresponding machine screen will be opened. Figure 14 presents the machine screens for the booster pump and the turbine. In each machine screen, the list of PM items is presented, which is different from one machine to another. Next to the right, the counted running time is shown. These numbers increase while the machine operates.

Further to the right, the reset buttons for all the PM items are placed. They are followed by the set-points, which indicate then the number of the hour when a certain PM item should be conducted.



Figure 13. The HMI's main screen via laptop/PC

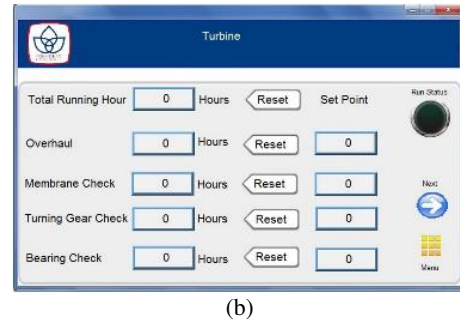
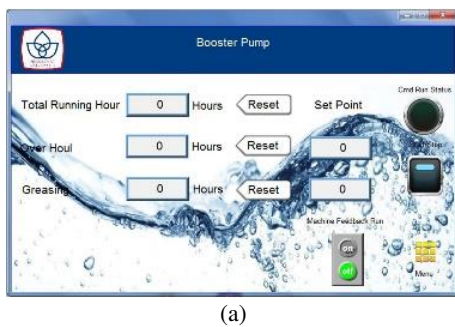


Figure 14. The HMI's machine screens; (a) Booster pump screen; (b) Turbine screen

5) Data Collection and Result Analysis

A data collection is conducted during a number of tests conducted to the IRTMS. All the features of the system in terms of the running time monitoring, the running time reset, the PM notifications, and the remote access for monitoring and control, are checked.

Since the IRTMS is fully a digital system that does not involve any measurement of continuous variables such as temperature, pressure, or velocity, the response of the system and the obtained data only have 2 states: OK or not OK, accomplished or not accomplished. Thus, although the tests were conducted multiple times, as long as the results are the same, there will be only one data and results reported.

III. RESULTS AND DISCUSSION

A. Test of Running Time Monitoring

In this section, the constructed IRTMS is tested to measure, record, and display the machine running time. To shorten the test time, a 1 hour-count is simulated to be equal to 10 seconds. The machines are turned on and off manually by using the push-buttons and the switches, and remotely through the HMI screen.

Each method is tested 10 times. Two of the results for each switching method are presented in Table 2. Before one test is started, the hour-counts are reset to 0. All the tests deliver the results as expected. The running time of all machines can be monitored successfully.

Table 2. Test Results of Running Time Monitoring

Machine	Switching method	Turn on duration	Running time count	As expected
Booster pump	Push-button	5 min	30 hours	✓
	Push-button	10 min	60 hours	✓
	HMI	5 min	30 hours	✓
	HMI	10 min	60 hours	✓
Compressor	Push-button	5 min	30 hours	✓
	Push-button	10 min	60 hours	✓
	HMI	5 min	30 hours	✓
	HMI	10 min	60 hours	✓
Turbine	Switch	15 min	90 hours	✓
	Switch	20 min	120 hours	✓
Boiler	Switch	15 min	90 hours	✓
	Switch	20 min	120 hours	✓

B. Test of Running Time Reset

In this test, an attempt to reset nonzero recorded running time and to change the schedule of the machines' PM items (or the set-point) is conducted. To do so, a user needs to have the privileged class of *Management*. Otherwise, for the classes of *Technician* and *Operator*, the reset button is locked and the running time reset cannot be done.

The result of the test is shown in Table 3. If provided with adequate privilege, the pressed reset button will respond to open the further page on the HMI screen. Later, the user can reset the desired running time or change the PM schedule.

Table 3. Test Results of Running Time Reset

Machine	Reset button pressed		As expected
	without privilege	with privilege	
Booster pump	Access denied	Running time reset done	✓
Compressor	Access denied	Running time reset done	✓
Turbine	Access denied	Running time reset done	✓
Boiler	Access denied	Running time reset done	✓

C. Test of Email Notification and HMI Screen Notification

In this test, the sending of one email notification and the display of HMI screen notification is simulated. The booster pump is taken as a sample. The remaining time for the next PM schedule of the booster pump is set to 250 hours (simulated to be equal to 2,400 seconds). Thus, the email and the screen notification will due within the next 10 hours (simulated 100 seconds), when the 240-hour reminder mark is reached.

Figure 15 shows the appearance of the email notification after the 240-hour mark is reached. The information regarding the required material, components, and tools is included in the message field of the email. The content of this email notification can be extended further according to the need of the user.

Figure 16 presents the appearance of the HMI screen by the time a PM reminder occurs. The name of the respective machine is shown on the screen, to ease the user to further investigate the details on the machine screen. To come back to the main page, the alarm acknowledge (ACK ALARM) button must be pressed.

D. Test of Remote Access for Monitoring and Control

This test is conducted by arranging a connection between the HMI and an Android phone. The HMI is connected to the EasyAccess platform via a wireless internet connection given to the router.

Figure 17 shows the HMI screen as it is exactly seen on the phone. The monitoring and control action can be conducted via the phone as if the user is physically present in front of the HMI screen. The running time of all four machines can also be seen on the phone screen. Based on the privilege level, the reset button and the set-point can

also be set and changed. The booster pump and the compressor can be turned on and of alternatingly from the Android phone, the HMI screen, or from the push-buttons. The result of the test is summarized in Table 4.



Figure 15. The email notification for the booster pump's PM

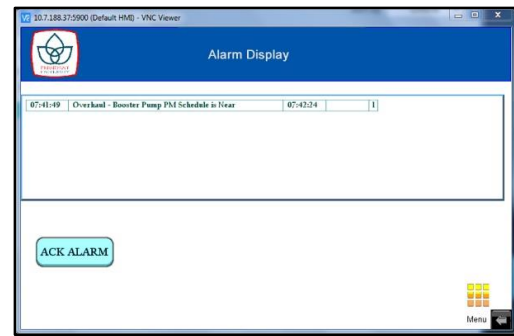


Figure 16. The HMI screen notification for booster pump's PM (via laptop/PC)

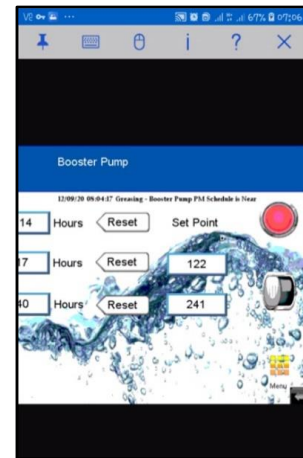


Figure 17. The screen of the Android phone as it is connected to the HMI

Table 4. Test Results of Remote Access for Monitoring and Control

Machine	Remote monitoring	Remote control	As expected
Booster pump	Functioning	Functioning	✓
Compressor	Functioning	Functioning	✓
Turbine	Functioning	Not available	✓
Boiler	Functioning	Not available	✓

E. Discussion

The proposed IRTMS is successfully constructed. On the hardware aspect, the IRTMS prototype is constructed by using components and materials that fulfill the standards for industrial application such as the PLC, the HMI, the relays, the MCB, the terminal blocks, the push-

buttons, the switches, and the cables. The whole prototype is encased in a solid and durable metal box that follows industrial norms, too. This becomes the strength of the IRTMS to be further demonstrated in a real operating environment.

On the software aspect, the IRTMS prototype requires the synergy between the PLC programming, the HMI programming, and the integration of a router and an online platform. This yields a desired remote accessibility of the HMI for the machines' monitoring and control.

Furthermore, the IRTMS prototype perfectly fulfills the objective to support the PM scheduling. The tests on running time monitoring, running time reset, PM notifications, and remote access are completed successfully.

In its current state, the IRTMS prototype does not yet support a pass-through function, where the HMI can work as a converter so that the user can use the Android phone or the laptop/PC to change or edit the PLC programs. A PLC with an ethernet connection makes this possible. The interconnectivity in implementation with multiple IRTMSs at a production site can be improved by using a router with wireless internet service provider (WISP) mode.

IV. CONCLUSION

An IoT-based running time monitoring system (IRTMS) is proposed in this paper. Through this automatic monitoring, the system is successful in achieving two key objectives. Firstly, the system keeps the preventive maintenance (PM) schedules of multiple machines. Thus, the user can get the schedule overview anytime and anywhere. Secondly, the system generates reminder notifications at a certain remaining time. By this, the user can prepare the material, components, or tools required for the PM in advance.

By scheduling the PM based on the actual running time, the efficiency of the PM will be increased. The PM is not scheduled based on the planned production time or projected production quantity anymore while omitting the downtime to the calculation.

To the best of the authors' knowledge, the proposed system for PM scheduling represents a novel approach and mechanism that has never been done before by any researchers. A series of tests were conducted to simulate and prove the operability and the functionality of the IRTMS. The successful results encourage the real-life implementation of the system in industrial applications.

REFERENCES

- [1] S. Assauri, *Production and Operation Management*, Revised Edition, Jakarta: Publication Institute of Faculty of Economics and Business, University of Indonesia, 2008.
- [2] M. Ben-Daya, *Handbook of Maintenance Management and Engineering*. London: Springer, 2009.
- [3] M. Tabikh, *Downtime Cost and Reduction Analysis: Survey Results*, Master Thesis, KPP321, Mälardalen University, (2014).
- [4] D.S. Thomas, *The costs and benefits of advanced maintenance in manufacturing*, National Institute of Standards and Technology, U.S. Department of Commerce, 2018.
- [5] R.K. Mobley, *An introduction to predictive maintenance*. Amsterdam: Butterworth-Heinemann, 2002.
- [6] M. Bevilacqua and M. Braglia, "The analytic hierarchy process applied to maintenance strategy selection," *Reliability Engineering & System Safety*, vol. 70, no. 1, pp. 71–83, 2000.
- [7] G.W. Vogl, B.A. Weiss, and M. Helu, "A review of diagnostic and prognostic capabilities and best practices for manufacturing," *Journal of Intelligent Manufacturing*, vol. 30, no. 1, pp. 79–95, 2016.
- [8] H. Ab-Samat, L. N. Jeikumar, E. I. Basri, N. A. Harun, and S. Kamaruddin, "Effective Preventive Maintenance Scheduling: A Case Study," *International Conference on Industrial Engineering and Operations Management*, pp. 1249–1257, 2012.
- [9] S.G. Robert, N. Bizon, and M. Oproescu, "The importance of PLC in the predictive maintenance of electronic equipment," *10th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, 2018.
- [10] Afianto, D.R.A. Abdullah, and S. Ardi, "Design automation control of downtime recording machine in the manufacturing industry," *Exploring Resources, Process, and Design for Sustainable Urban Development: Proceedings of the 5th International Conference on Engineering, Technology, and Industrial Application (ICETIA)* 2018.
- [11] A.K. Yanti and E. Sitompul, "Designing and Prototyping Record System for Machine Usage Time," *Journal of Electrical And Electronics Engineering*, vol. 2, no. 1, p. 1, 2019.
- [12] A.C. Márquez, V. G.-P. Díaz, and J.F.G. Fernández, *Advanced Maintenance Modelling for Asset Management: Techniques and Methods for Complex Industrial Systems*. Cham: Springer International Publishing, 2018.
- [13] E. Hupje, *9 Types of Maintenance, Road to Reliability*. [Online]. Available: <https://roadtoreliability.com> [December 20, 2019].
- [14] T. Taufik and S. Septyani, "Penentuan Interval Waktu Perawatan Komponen Kritis pada Mesin Turbin di P.T. PLN (Persero) Sektor Pembangkit Ombilin," *Jurnal Optimasi Sistem Industri*, vol. 14, no. 2, p. 238, 2016.
- [15] H. Jack, "Automating manufacturing systems with PLCs," May 4, 2007. [Online]. Available: <https://electrical-engineering-portal.com/>. [Accessed February 1, 2019].
- [16] R. Buinac, D. Marsic, and G. Malcic, "Web-based control of programmable logic controllers in process control computers education," *2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 2014.
- [17] HollySys, *LM Micro Series PLCMT8070iE*, May 1, 2011. [Online]. Available: <http://hollysys.com.sg>. [Accessed February 1, 2019].
- [18] Weintek Labs, *HMI MT8071iE1*, December 6, 2019. [Online]. Available: <https://www.weintek.com>. [Accessed April 1, 2019].
- [19] Weintek Labs, *EasyAccess 2.0*, 2014. [Online]. Available: <https://www.weintek.com/globalw/Software/EasyAccess.aspx>. [Accessed April 1, 2019].
- [20] SIA Mikrotikls, *hAP lite RB941-2nD Router*. [Online]. Available: <https://mikrotik.com/product/RB941-2nD>. [Accessed April 1, 2019].
- [21] IDEC Corporation, *RU Series Universal Relays*. [Online]. Available: <https://idec.com>. [Accessed April 1, 2019].