A Prototype of an IoT-based Production Performance and Quality Monitoring System Using NodeMCU ESP8266

Erwin Sitompul[1], Trias Wulandari[2], and Mia Galina[3]

Department of Electrical Engineering,
Faculty of Engineering,
President University, Cikarang
[1]sitompul@president.ac.id, [2]triaswulan@gmail.com, [3]miagalina@president.ac.id

Abstrak

Suatu proses produksi membutuhkan pemantauan kinerja dan kualitas agar jumlah dan mutu produk jadi yang dihasilkan dalam suatu rentang waktu produksi dapat dijaga sesuai standar. Dengan kualitas produk terbagi dalam tiga kategori (OK, *Repair*, dan *Scrap*), jumlah produk yang dihasilkan oleh proses *finishing* produksi segel karet pintu mobil di PT. XYZ masih dilakukan secara manual. Penghitungan yang selama ini dilakukan oleh operator sambil bekerja rentan terhadap ketidakakuratan. Selain itu, hasil penghitungan harus diproses secara manual oleh seorang administrator di akhir shift kerja. Proses rekapitulasi yang lambat ini menyebabkan pemantauan kinerja dan kualitas produksi (PPQM) tidak dapat dilakukan secara optimal. Dalam tulisan ini diajukan suatu purwarupa sistem PPQM berbasis IoT sebagai pemecahan masalah. Purwarupa ini memilik tiga tombol yang dapat langsung ditekan oleh operator untuk mencatat kualitas setiap produk jadi, dengan kuantitas yang selalu diakumulasikan. Mikrokontroler NodeMCU mengirimkan data produksi ke platform IoT Blynk melalui koneksi internet. Data yang mendekati waktu-nyata ini kemudian dapat dipantau melalui aplikasi seluler Blynk. Aplikasi ini juga menampilkan suatu tingkat kinerja dan kualitas (PQ rate) sebagai indikator pemantauan. Sensor inframerah digunakan untuk mendeteksi objek kerja. Konsistensi operator untuk memasukkan data dijaga dengan penggunaan suatu mekanisme penjepit. Sensor inframerah diuji dan bekerja dengan baik di dalam jarak baca 8 mm. Tombol harus ditekan selama minimal 0,98 detik agar data yang dimasukkan dapat dikirim oleh NodeMCU ke server dengan benar. Makro MS Excel ditulis agar data produksi harian dapat diproses secara otomatis dan cepat. Hasil simulasi menunjukkan bahwa sistem berhasil menyederhanakan dan meningkatkan akurasi pencatatan data produksi. Selain itu, PPQM yang mendekati waktu nyata menjadi dimungkinkan.

Kata kunci: kinerja dan kualitas produksi, NodeMCU, Blynk, IoT

Abstract

A production process requires performance and quality monitoring to maintain the product quantity and quality to a prescribed standard. With the product quality classified into three categories (OK, Repair, and Scrap), the quantity count in a finishing process of a car door rubber seal production at PT. XYZ is still done manually. This count is conducted by an operator during the work and is prone to inaccuracies. Furthermore, the count result must be manually processed by an administrator at the end of a work shift. This slow recapitulation process makes optimal production performance and quality monitoring (PPQM) not possible.

Received: 9 Februari 2022 Revised: 23 Maret 2022 Accepted: 26 Maret 2022

In this paper, a prototype of an IoT-based system for PPQM is proposed as the solution. The prototype provides three pushbuttons for the operator to directly record the quality of each product, while the quantity is automatically added up. NodeMCU microcontroller sends the production data to Blynk IoT-platform via internet connection. The quasi-real-time data can later be monitored through Blynk mobile application. The application also displays a performance-and-quality rate (PQ rate) as a monitoring indicator. One infrared sensor is utilized to detect the work objects. The operator consistency to enter the data is maintained by the use of a clamp mechanism. Test results show that the infrared sensor works very well within the detection range of 8 mm. The pushbuttons must be pressed for at least 0.98 second, so the input can be correctly relayed by NodeMCU to the server. An MS Excel macro is developed so that the production data can be processed automatically and quickly. The simulation results show that the proposed system can successfully simplify and increases the accuracy of the production data record. Besides, it makes a quasi-real-time PPQM possible.

Keywords: production performance and quality, NodeMCU, Blynk, IoT

1. Introduction

Productivity is an essential issue in manufacturing industries. Increasing productivity can only be achieved by increasing capacity or efficiency. In general, the effort to increase productivity is to find the balance between cost, quantity, and quality. One method that can be used to systematically improve production efficiency is Overall Equipment Effectiveness (OEE). OEE is divided into 3 measures: Availability, Performance, and Quality [1]. This method eases the investigation to find specific waste or losses that affect productivity. Monitoring of input and output of manufacturing processes is also one of the deciding factors in productivity improvement, which covers a wide range of technology and management approaches [2]. A good monitoring system helps to provide an early indication of substandard productivity [3].

PT. XYZ is a company that manufactures car spare parts. One of the production lines produces car door rubber seals. The production performance and quality monitoring (PPQM) of the line's finishing process is still done manually. The operator is expected to keep the record of the product while working, based on the quantity and the corresponding quality category. In the early stage of computerization and digitalization, one solution to production performance monitoring was the implementation of radio frequency identification (RFID), as shown in [4]. Here, real time location system was implemented to obtain multi-item work-in-process visibility within a manufacturer. Nowadays, the methods, tools, and standards used to measure the performance develops to suit the Industry 4.0, which is characterized by the use of cyber physical systems [5]. Hence, the intention of the authors is to implement an Internet-of-Things (IoT)-based monitoring system to cope with the development of technology.

Based on the observations conducted by the authors, this manual record keeping is vulnerable to human error. The operator tends to make inaccurate counts based on subjective estimation only. Besides, obtaining historical production data requires many efforts. At the end of a work shift, the operator must first submit the record to an administrator. Then the data is entered into a database before it can be presented on the following day. Thus, it is not possible for a shift supervisor to swiftly intervene in the middle of a work shift, in case any decrease in product quantity or quality occurs. To solve the problem mentioned above, the authors propose a PPQM system.

The PPQM system is expected to be easy to use and able to keep the operator entering the production data with a high level of consistency and accuracy. Besides, the system is to be equipped with IoT-capability. By this, the production data can be monitored in quasi-real-time and without being limited by distance and location. Additionally, it is desirable to have a tool that enables automatic and quick data recapitulation at the end of a production day. The realization of the PPQM system is to be done with the utilization of NodeMCU as the microcontroller. NodeMCU has found various IoT-related applications such as data acquisition of clean water usage in [6] and electrical energy consumption in [7]. In other works, [8] and [9] use the microcontroller in monitoring parameters of greenhouses such as air temperature, air humidity, light intensity, and soil moisture. Recent implementations of IoT in production processes can be found in [10], where automatic product counting and identification is done by using a QR code scanner, and in [11], where weights of products are measured and directly recorded in a web database. To the best of the authors' knowledge, the proposed PPQM system is a novel application of IoT in the field of performance monitoring in manufacturing.

2. Methods

2.1. Overall Equipment Effectiveness (OEE)

One of the best practices in manufacturing process is the measuring of Overall Equipment Effectiveness (OEE) [1,12]. OEE indicates how a manufacturing process is performed compared to its full potential, under given time, material, and facilities. A perfect OEE is achieved by obtaining perfect interrelated factors of availability (the manufacturing process is conducted without interruption), performance (the process is conducted at maximum speed or cycle time), and quality (the process only produces good parts). Figure 1 shows the overview of the three factors of OEE.

The factors are calculated according to the following formulas:

$$\text{Availability rate} = \frac{\text{Running time}}{\text{Available operating time}} \times 100\% \tag{1}$$

$$\text{Performance rate} = \frac{\text{Actual output}}{\text{Target output}} \times 100\% \tag{2}$$

$$\text{Quality rate} = \frac{\text{Good output}}{\text{Actual output}} \times 100\% \tag{3}$$

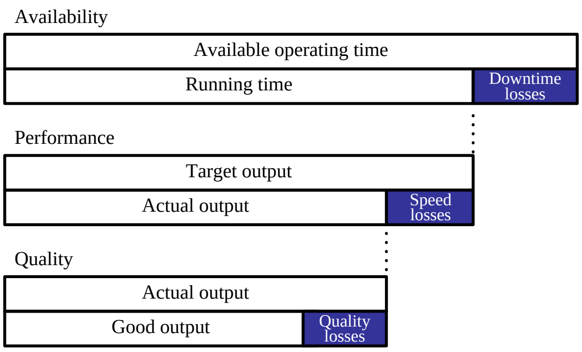


Figure 1. The three factors of OEE

If the OEE of a process is accurately measured, then a systematic improvement to the underlying process can be performed. The root causes of the six major losses such as breakdowns, setup/adjustments, minor stops/idling, low speed, start-up rejects, and production rejects, can be identified and effective solutions can be formulated and implemented [13].

2.2. Finishing Process of Car Door Rubber Seal Production at PT. XYZ

PT. XYZ produces car door rubber seals, as shown in Figure 2. Rubber seal serves to attenuate noises, prevent water leaks, and eliminate metal-on-metal contact.

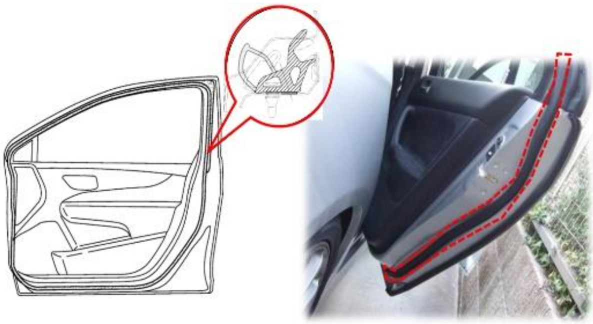


Figure 2. Car door rubber seal

The finishing process of rubber seal production is done by an operator by putting a pair of rubber seals on an assembly jig, as shown in Figure 3.

Figure 3. An operator conducting the finishing process of rubber seals

Figure 4 depicts the stages of the finishing process. In each cycle, a pair of pre-processed rubber seals is placed parallelly (1). Each pair are then glued (2) and clipped (3), resulting in two finished products.

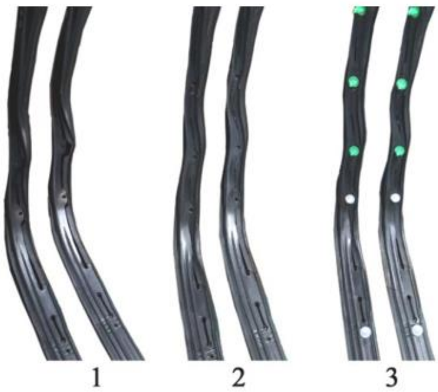


Figure 4. Car door rubber seal

There are 3 quality categories for the finished rubber seals. OK is given to products that fully meet the criteria and conditions requested by the customers. Repair is given to products that have minor defects in appearance but can later be repaired by the operator. Scrap is given to products that are so defective that their functionality is impaired and they cannot be installed to the body of a car.

In the proposed PPQM system, a combination of performance rate and quality rate as given by equations (2) and (3) is utilized as the main indicator. The performance-and-quality rate (PQ rate) is given by:

$$\text{PQ Rate} = \frac{\text{Good output}}{\text{Target output}} \times 100\% \tag{4}$$

The good output is the sum of products with OK or Repair quality category. Meanwhile, the target output is calculated based on the expected cycle time of 75 seconds per product.

2.3. Design Specification

2.3.1. NodeMCU ESP8266

As a step to realize the IoT-capability of the proposed PPQM system, the authors choose NodeMCU ESP 8266. The microcontroller, as can be seen in Figure 5, is low-cost open-source hardware that can be connected to various IoT platforms.

The ESP8266 Wi-Fi module integrated with the microcontroller NodeMCU is preferred due to its simplicity. Previously, the authors have experimented with the use of Arduino Uno and the separated ESP8266 Wi-Fi module. The circuitry and the programming were more complicated than the ones using the integrated NodeMCU ESP8266.

Figure 5. NodeMCU ESP8266 [14]

This microcontroller can be programmed to configure and manipulate devices connected to it, whether directly via its physical inputs or remotely via the internet [14]. Thus, in the PPQM system, NodeMCU serves as the connecting bridge between the physical world of the finishing process and the virtual world of the cloud server.

2.3.2. Blynk Platform

Blynk provides a platform to develop iOS and Android mobile applications that allow the user to interact with a wide range of microcontrollers, including NodeMCU. Any hardware connected to NodeMCU can be monitored and controlled via the internet through the application.

Blynk also provides an application builder that allows the user to create a project interface. On this interface, a variety of input and output components can be included. Received data can be saved, sent, and displayed in various forms. The communication between the mobile application and the microcontroller is governed by a cloud-based server [15].

Blynk is chosen after exploring other IoT platforms such as ThingSpeak, Firebase, and Thinger.io. The main reasons are the authors' familiarity and subjectivity to its straightforward features that directly supports the need of simple interconnection between the system parts.

2.3.3. TCRT5000 Infrared Reflective Sensor

An infrared sensor is an electronic device that is used to detect the presence of objects based on the reflection and the absorption principle of infrared radiation. In the proposed PPQM system, the authors utilize the TCRT5000 infrared reflective sensor, which is shown in Figure 6.

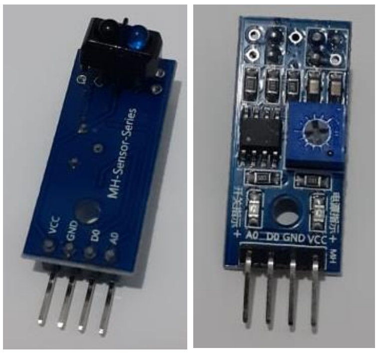


Figure 6. TCRT5000 infrared reflective sensor

TCRT5000 consists of an IR LED (infrared light-emitting diode) as the transmitter and a phototransistor as the receiver. The IR LED emits an invisible wave that will be reflected by the object in front of it. The reflection is then detected by the phototransistor [16]. Thus, this sensor serves to detect the presence and the absence of the rubber seals on their assembly jig, as a sign that the current finishing process is already finished or the next can be started.

2.3.4. MG995 Servo Motor

MG995 servo motor, as shown in Figure 7, is an actuator with closed-loop control. This means that the rotation of the shaft can be accurately set between 0° and 180°. The servo motor is driven by governing the input to its controller in the form of pulse width modulation [17].

In the proposed PPQM system, the servo motor serves to open or close a clamp mechanism, which is intended to remind the operator to consistently enter the production data based on the corresponding quality category.

Figure 7. MG995 servo motor [17]

2.3.5. MS Excel Macro

The Blynk application builder features a certain widget that enables the export of comma-separated value (CSV) files. This file can be opened by using spreadsheet software for further process and analysis. MS Excel is a commonly used spreadsheet software that can be programmed to open and process such CSV files. A sequence of actions or commands can be programmed by using a macro. Later, the macro can be run repeatedly so that the repetitive sequence can be performed automatically and quickly [18].

The programming of a macro is initiated by opening the developer tab, as shown in Figure 8. To process a file with macro, the file is to be stored in a macro-enabled workbook (XLXM).

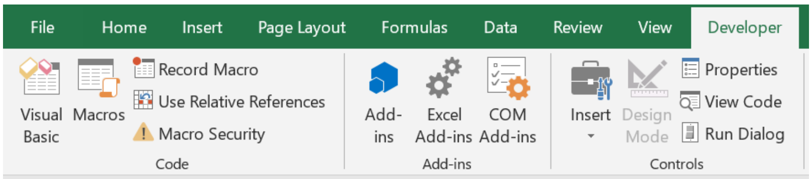


Figure 8. Menu on MS Excel developer tab

A macro will be developed to support the proposed PPQM system. After a CSV file is sent from the Blynk mobile application and downloaded to a designated folder, the file can be processed conveniently to show the historical production data. This eliminates the manual data recapitulation currently conducted by an administrator at the end of a work shift.

2.3.6. Block Diagram

The proposed PPQM system is a coordinated implementation of hardware and software, with the block diagram presented in Figure 9. NodeMCU receives inputs from 3 pushbuttons and 1 infrared sensor. The pushbuttons correspond to the quality category of OK (green), Repair (Yellow), and Scrap (red). One output of NodeMCU is directed to drive the servo motor. NodeMCU is also connected to the Blynk server via an internet connection. A developed Blynk mobile application is executed from a compatible mobile device to display the data of the finishing process in the form of numbers and charts.

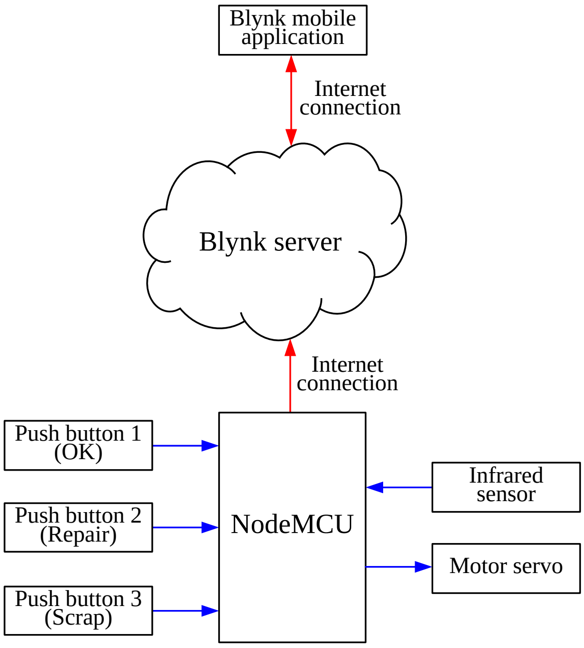


Figure 9. The block diagram of the proposed system

2.3.7. Prototype Design

The prototype of the PPQM system has the functional design as shown in Figure 10. A pair of rubber seals is processed in each production cycle. The assembly jig is used to fix the position of the rubber seals during the finishing process. By the time the rubber seals are put on the jig, their presence will be detected by the infrared sensor. Then, the servo motor will close the clamp. The rubber seals can still be further adjusted to a desired position on the jig, but they are locked by the clamp and cannot be taken out anymore.

Once the finishing process is completed, the operator should press the pushbuttons in total of two times, in any possible combinations, according to the quality category of the product. After the pushbuttons are pressed for the second time, the clamp will be opened and the rubber seals can be taken out from the jig. The next production cycle is ready to commence.

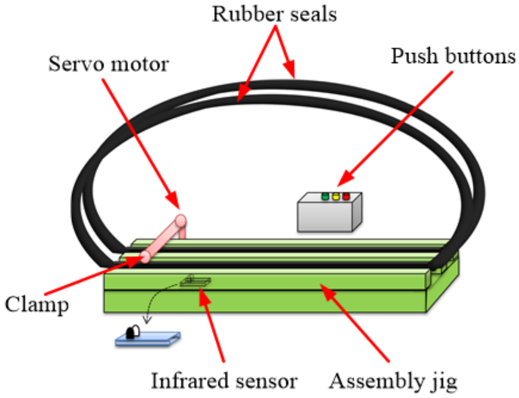


Figure 10. The functional design of the prototype

Figure 11 shows the schematic design of the pushbutton box (1), the clamp mechanism (2) with a servo motor seat and the clamp, and the infrared sensor box (3). In the prototype of the proposed PPQM system, the work object rubber seals will be replaced by black rectangular bars. All items mentioned above are to be created by using a 3-D printer with polylactic acid (PLA) material.

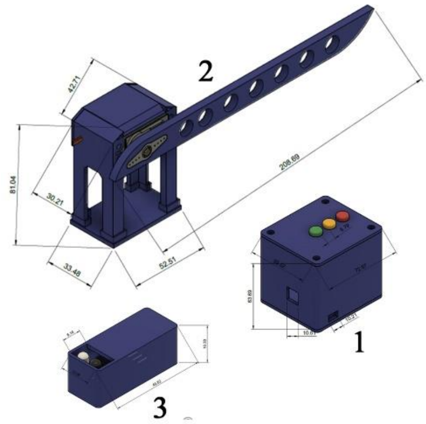


Figure 11. The schematic design of the prototype parts

2.4. Design Implementation

2.4.1. NodeMCU Circuit and Programming

The circuit around NodeMCU is realized with the configuration as shown in Figure 12. The pushbuttons, the infrared sensor, and the servo motor are appropriately connected to the corresponding digital pins and ground.

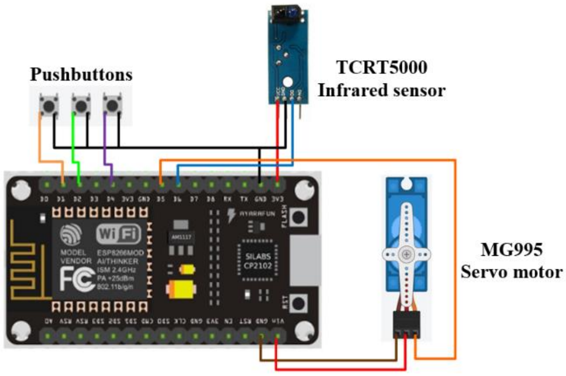


Figure 12. NodeMCU circuitry

The program to be uploaded to the NodeMCU is made by using the compatible Arduino IDE. The flowchart of this program is presented in Figure 13.

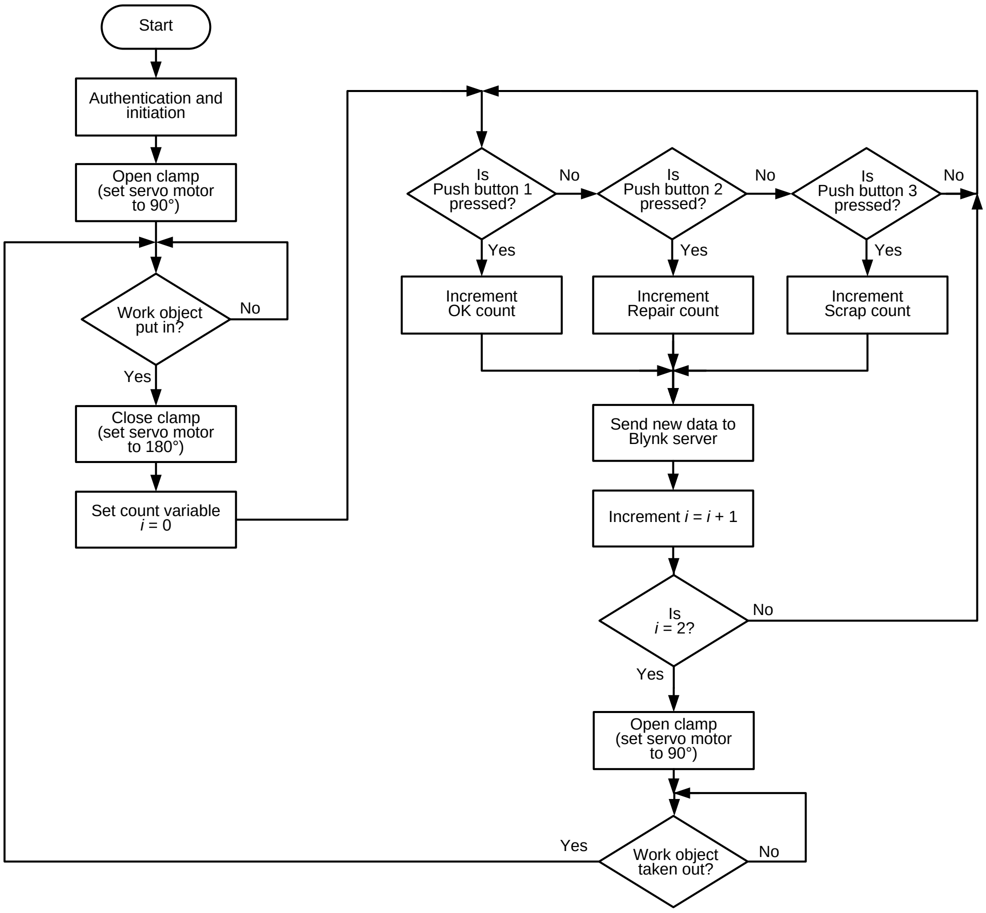


Figure 13. The flowchart of the NodeMCU program

At the start, the connection between NodeMCU and Blynk server is established and the initiation is conducted. The clamp is set to an open position. If the work objects are detected, the clamp is closed. Then, a temporary variable is reset to 0. This variable increments by 1 every time a pushbutton is pressed.

After the finishing process is completed, the operator is required to press the pushbuttons twice, with the combination according to the achieved quality category. This data is directly sent to the server. If the value of the temporary variable is equal to 2, then the clamp is opened and the work objects can be collected as finished products. The clamp stays opened until new work objects are placed on the assembly jig. At this time, the clamp is closed again and a new production cycle begins.

2.4.2. Blynk Mobile Application Programming

The Blynk mobile application is to be programmed so that the required data can be presented on the mobile device screen as desired by the user.

Figure 14(a) shows the Edit Mode of the application builder, where 6 data streams are selected to be displayed on the main screen. The OK, Repair, and Scrap counts are obtained from NodeMCU. The Target value starts from 0 and is increased by 1 every 75 seconds, which is the cycle time of one product. The Actual value is equal to the sum of OK and Repair products. Finally, the PQ rate is determined by using Equation (4), which is the ratio of the Actual value to the Target value multiplied by 100%.

The appearance of the main screen of the mobile application in Play Mode is shown in Figure 14(b). All six data streams are presented in numerical format. Besides, three data streams (OK, Repair, and Scrap counts) are also shown in a graph at the lower part of the screen.

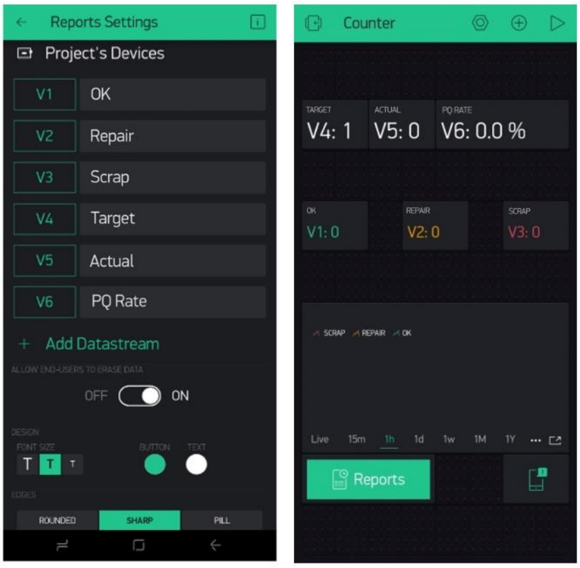


Figure 14. Programming of Blynk mobile application; (a) Edit mode, (b) Play mode

3. Results and Discussions

3.1. Infrared Sensor Test

In this stage, the infrared sensor is tested to detect a replica of rubber seals in the form of black rectangular bars. The objective of the test is to find the optimal distance for the infrared sensor to detect the work objects.

The work objects are placed at 7 different distances to the infrared sensor, varying from 20 mm to 2 mm. For every distance, 3 tests are conducted. The result is shown in Table 1, with two possible outcomes: D for detected and ND for not detected.

Table 1. The test result of infrared sensor

Distance (mm)	Experiment Number*		
	1	2	3
20	ND	ND	ND
15	ND	ND	ND
10	D	ND	D
8	D	D	D
6	D	D	D
4	D	D	D
2	D	D	D

*D : Detected, ND : Not detected

The test concludes that reliable detection of the work objects will be obtained if the distance between the sensor and the work objects is below 8 mm. In this distance range, the sensor can perform reliable detections. Such a small distance is required since the black work objects do not reflect infrared wave very well. Thus, for proper detection by the phototransistor receiver, the distance must be set small.

3.2. Push Button Response Test

This test is intended to find any required consideration for the operator when the operator enters the data into the PPQM system by using pushbuttons. Two conditions are investigated: the first is when the system is not connected to the Blynk server, while the second when it is connected. In both situations, the pushbuttons are pressed for various durations. The response of the system is shown on a laptop's serial monitor. If the data entered through a pushbutton can be read successfully, then the corresponding count increments by 1. Otherwise, the count will not change.

The test result under the first situation is shown in Table 2. The system can respond quickly to the data entered using pushbuttons. Given the press duration between 0.40–1.44 seconds, every pushbutton press is followed by the successful increment of the corresponding count. Indeed, when the system is not connected to the server, there is no requirement for the system to perform any server-related commands. The system does not need to deal with delay and latency, making quick responses possible.

Table 2. The test result of push button response without connection to Blynk Server

Press time (second)	Respond in serial monitor (count)	Result
Pushbutton 1 (OK/Green)		
0.41	OK = 1	C
0.42	OK = 2	C
0.52	OK = 3	C
0.62	OK = 4	C
0.74	OK = 5	C
0.89	OK = 6	C
0.92	OK = 7	C
0.96	OK = 8	C
1.13	OK = 9	C
Pushbutton 2 (Repair/Yellow)		
0.40	Repair = 1	C
0.49	Repair = 2	C
0.46	Repair = 3	C
0.62	Repair = 4	C
0.73	Repair = 5	C
0.88	Repair = 6	C
1.02	Repair = 7	C
0.96	Repair = 8	C
1.34	Repair = 9	C
Pushbutton 3 (Scrap/Red)		
0.39	Scrap = 1	C
0.52	Scrap = 2	C
0.44	Scrap = 3	C
0.68	Scrap = 4	C
0.92	Scrap = 5	C
0.93	Scrap = 6	C
1.12	Scrap = 7	C
1.20	Scrap = 8	C
1.44	Scrap = 9	C

*C : Counted, NC : Not counted

The test result under the second situation is presented in Table 3. When connected to the server via the internet, the system requires the pushbuttons to be pressed for at least 0.98 seconds. If so, the data entered can be regarded as valid and the corresponding count can increase.

As shown in Figure 14, 6 data need to be sent in every loop of the program. By the authors' experience, every line of Blynk virtual write command contributes to a delay of approximately 100 ms. This magnitude is in accordance to the delay experienced by a number Blynk users (100-200 ms per Blink.virtual Write command), as can be found at Blynk official community site https://community.blynk.cc. The delay due to these commands, summed up with eventual internet connectivity latency, can easily result in a total delay of around 800 ms.

Table 3. The test result of push buttons response with connection to Blynk server

Press time (second)	Respond in serial monitor (count)	Result
Pushbutton 1 (Green)		
0.53	OK = 0	NC
0.42	OK = 0	NC
0.69	OK = 0	NC
0.52	OK = 0	NC
0.76	OK = 0	NC
0.86	OK = 0	NC
0.92	OK = 1	C
1.13	OK = 2	C
1.29	OK = 3	C
Pushbutton 2 (Yellow)		
0.32	Repair = 0	NC
0.43	Repair = 0	NC
0.66	Repair = 0	NC
0.78	Repair = 0	NC
0.64	Repair = 0	NC
0.86	Repair = 0	NC
0.92	Repair = 1	C
1.13	Repair = 2	C
1.29	Repair = 3	C
Pushbutton 3 (Red)		
0.44	Scrap = 0	NC
0.67	Scrap = 0	NC
0.58	Scrap = 0	NC
0.73	Scrap = 0	NC
0.86	Scrap = 0	NC
0.67	Scrap = 0	NC
1.10	Scrap = 1	C
0.98	Scrap = 2	C
0.99	Scrap = 3	C

*C : Counted, NC : Not counted

3.3. Overall Prototype Testing

Figure 15 depicts the operation steps of the prototype of the PPQM system. At first, the clamp is in an open position with no work objects (1). Then, the work objects are placed in the emulated assembly jig (2). The infrared detects the work objects and the clamp is closed (3).

After the finishing process is completed, the green pushbutton is pressed once, indicating that the first product is OK (4). The clamp is still in a closed position. The same pushbutton is pressed once more, indicating that the second product is OK (5). This is directly followed by the opening of the clamp. The work objects can now be taken out and the production cycle can be repeated from the start (1).

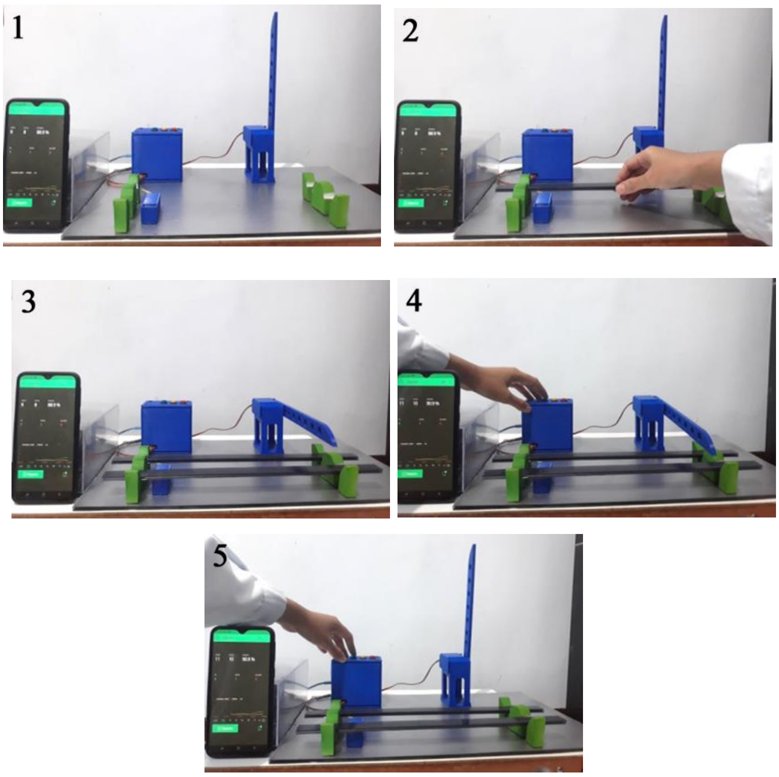


Figure 15. Test of the prototype of PPQM system

During a production process, the update made by every data is displayed on the screen of the Blynk mobile application. Table 4 shows the progress of a simulated production process as a sequence of production data is given to the PPQM system. The target value is increased according to the required cycle time, in this case, 75 seconds. Every OK and Repair product contributes to the increase of actual value, while Scrap does not.

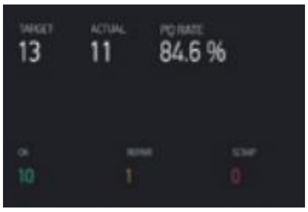
At the initial condition, the target value is 11 and the actual value is 9 (8 OK and 1 Repair), which corresponds to the PQ rate of 81.8%. In step 1, the resulted product is OK and pushbutton 1 is pressed. The actual value is now changed to 10 (9 OK and 1 Repair), increasing the PQ rate to 90.9% since the target value is still 11.

In step 2, again an OK product is finished. This increases the actual value to 11 (10 OK and 1 Repair), while the target value is already increased to 13. The PQ rate is now reduced to 84.6%. Steps 3 and 4 can readily be examined with the same procedure.

In step 5, as Scrap product is made, the actual value does not increase and stays at 13. With a target value of 15, the PQ rate is given by 86.7%.

These sequences of tests are conducted successfully, proving that the PPQM system works perfectly as expected. The mobile application shows the automatic increment of the target value according to the cycle time, while the actual value is updated based on the input entered into the system. The PQ rate can be monitored at any time. This value is a perfect indicator for the work shift supervisor to intervene in case it falls below a certain limit.

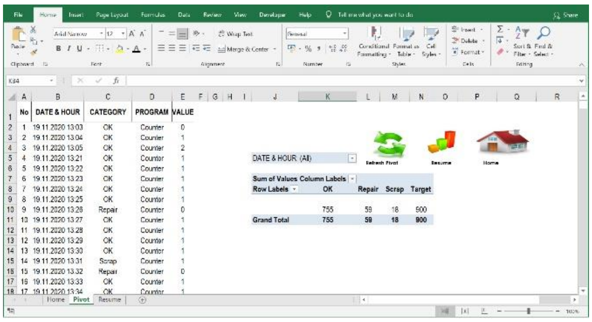
Table 4. The test result of production data sequence

<table>
<tr><th>Step</th><th>Input</th><th>Screen of Blynk Application</th></tr>
<tr><td>Initial condition</td><td>-</td><td>
</td></tr>
<tr><td>1</td><td>OK</td><td>
</td></tr>
<tr><td>2</td><td>OK</td><td>
</td></tr>
<tr><td>3</td><td>Repair</td><td>
</td></tr>
<tr><td>4</td><td>OK</td><td>
</td></tr>
<tr><td>5</td><td>Scrap</td><td>
</td></tr>
</table>

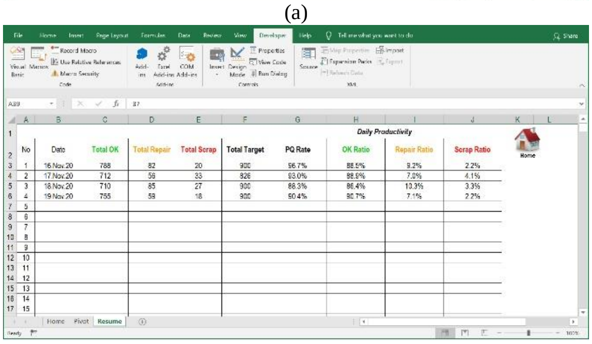
3.4. Excel Macro Test

After a successful MS Excel macro development, the data can now be processed automatically and quickly to present the important information from the production process. The CSV file received from the Blynk mobile application is now opened by the macro. Figure 16(a) shows an example of a pivot table obtained from one production day. The data recapitulation can be done by clicking the “Refresh Pivot” button. The result is directly shown in the table on the right-hand side.

Figure 16(b) shows the data summary of 4 production days, where the PQ rate is calculated on one of the columns. Further data processing can be done to extend the analysis, for example, the ratio of each quality category to the actual value.



(a)



(b)

Figure 16. The data recapitulation using MS Excel macro; (a) Pivot table of one production day; (b) Recapitulation of multiple production days

4. Conclusions

A prototype of the production performance and quality monitoring (PPQM) system is proposed in this paper. The prototype is meant to solve the monitoring problem due to inaccurate records and slow recapitulation of the production data at a finishing process in a manufacturing company PT. XYZ. PQ rate is proposed as the indicator for the performance and quality, based on the Overall Equipment Effectiveness (OEE) approach. PQ rate is defined as the ratio of the actual quantity of good products to the target quantity based on the product's cycle time. The PPQM system is IoT-enabled by the utilization of the NodeMCU microcontroller and Blynk platform. The operator uses pushbuttons to enter the production data, which is classified based on product quality. The data is virtually written to a cloud server. A mobile application is created to access and export the data via the internet connection. Besides, the application serves as a monitoring interface, displaying the PQ rate at anytime from anywhere. The prototype is built to emulate the operator's real workplace. For the system to function properly, the infrared sensor needs to be set 8 mm or closer to the work object at the assembly jig and the pushbuttons must be pressed for at least 0.98 seconds. Besides, a spreadsheet macro is developed as a tool to make production data recapitulation automatic and quick. The overall functionality of the PPQM system is tested successfully and it offers a solution to the current monitoring problems.

Possible further development includes the feature for the operator to make data corrections. Preventive maintenance based on the number of finished products can also be better scheduled by the use of the proposed system. A realization in real environment is expected to be conducted as future implementation

References

[1] Z. Aman, L. Ezzine, J. Fattah, and A. Lachhab, "Improving Efficiency of a Production Line by Using Overall Equipment Effectiveness: A Case Study," *Proc. in International Conference on Industrial Engineering and Operations Management*, Rabat, Morocco, 2017.

[2] K. Hon, "Performance and Evaluation of Manufacturing Systems," *CIRP Annals*, vol. 54, no. 2, pp. 139–154, 2005.

[3] P. Jonsson and M. Lesshammar, "Evaluation and Improvement of Manufacturing Performance Measurement Systems - The Role of OEE," *International Journal of Operations & Production Management*, vol. 19, no. 1, pp. 55–78, 1999.

[4] I. Arkan and H. Van Landeghem, "Evaluating the performance of a discrete manufacturing process using RFID: A case study", *Robotics and Computer-Integrated Manufacturing*, vol. 29, no. 6, pp. 502–512, 2013.

[5] P. Tambare, C. Meshram, C.-C. Lee, R. J. Ramteke, and A. L. Imoize, "Performance measurement system and quality management in data-driven Industry 4.0: A review," *Sensors*, vol. 22, no. 1, pp. 224, 2021.

[6] E. Sitompul and S. M. Samsudin, " IoT-based Data Acquisition of Clean Water Usage Using NodeMCU," *Proc. in International Conference on Sustainable Engineering and Creative Computing (ICSECC)*, Indonesia, 2020.

[7] A. Suryana, M. R. Fauzi, R. A. Ramadhan, and F. Muiz, "Monitoring Electrical Energy Consumption Through Impulse Detection Using LDR Sensors and Relay With NodeMCU ESP8266," *Proc. in 5th International Conference on Computing Engineering and Design (ICCED)*, 2019.

[8] Y. Astutik, Murad, G. M. D. Putra, and D. A. Setiawati, "Remote Monitoring Systems in Greenhouse Based on NodeMCU ESP8266 Microcontroller and Android," *Proc. in 2nd International Conference on Bioscience, Biotechnology, And Biometrics*, 2019.

[9] Z. Wan, Y. Song, and Z. Cao, "Environment Dynamic Monitoring and Remote Control of Greenhouse with ESP8266 NodeMCU," *Proc. in IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, 2019.

[10] S. Parningotan and T. Mulyanto, "Rancang Bangun Prototipe Alat Penghitung Produk Secara Otomatis Dengan Konsep Internet of Things (IoT) Berbasis Mikrokontroller (Arduino Uno)," *Electro Luceat*, vol. 6, no. 1, pp. 74–81, 2020.

[11] A. Sunata and Rino, "Rancang Bangun Alat Penghitung Jumlah Produksi Dengan Menggunakan Microcontroller Load Cell Berbasis Web Service," *Jurnal Algor*, vol. 1, no. 2, pp. 59–66, 2020.

[12] F. U. Nisbantoro, R. Jinan, and H. H. Purba, "Measurement Overall Equipment Effectiveness on Injection Moulding Machine: A Case Study in Injection Moulding Manufacturing Industry," *International Journal of Engineering Research and Advanced Technology*, vol. 4, no. 8, pp. 62–69, 2018.

[13] D. Hogfeldt, "Plant Efficiency: A Value Stream Mapping and Overall Equipment Effectiveness Study," *Master Thesis*, Department of Applied Physics and Mechanical Engineering, Lulea University of Technology, 2005.

[14] Zerynth, NodeMCU v3, Zerynth.

[15] Blynk Inc., We make Internet of Things simple for you, 2016. [Online]. Available: https://blynk.io. [Accessed July 10, 2021].

[16] TCRT5000 TCRT5000L. Vishay Semiconductors. Lithuania. 2019.

[17] MG995 High Speed Servo Actuator. Marlin P. Jones & Assoc., Inc. Lake Park. 2010.

[18] Microsoft Official Academic Course. Microsoft Excel 2016. Quick Start. New Jersey: John Wiley & Sons Inc. 2016.