International Journal of
Science and Engineering Investigations vol. 8, issue 85, February 2019
Received on February 21, 2019 ISSN: 2251-8843

IoT Remote Data Logger Platform for Multiple Sensors and Robust Data Connection

Harnan Malik Abdullah[1], Fridy Mandita[2]
[1]University of Brawijaya
[2]University of 17 Agustus 1945
([1]harnan_malik@ub.ac.id, [2]fridymandita@untag-sby.ac.id)

Abstract- A new data logger platform has been created based on raspberry pi hardware. This platform consists of remote data logger devices and online servers. The Remote devices accommodate various input sensors having voltage output (0-5V) or current output (4-20mA). A software was developed to accommodate sensor readings, store data, and send data to online servers via the internet. Data transmission is done periodically by implementing a compression algorithm to reduce data sending capacity. On the online server side, software is also developed to decompress and store data. In addition, web page based applications are made for the management and identification of remote data loggers and the appearance of sensor data both in real time and historically. In this paper, we have demonstrated testing platforms using two kinds of sensors namely temperature sensor with current sensor output and distance sensor with voltage sensor output. The results show that the system can function according to its function.

Keywords- *IoT, Remote, Data Logger, Multi Sensor, Robust*

I. INTRODUCTION

Remote data logger is a device used to store measurement data and send data from the measurement point to the server. Siew et al [1] made a remote data logger study based on wireless. In this study, communication between measurement devices with PC servers using XBee or Bluetooth. Mentioned also the maximum range of XBee is 30m (indoor) and 100m (outdoor) while Bluetooth only reaches a range of distances between 20m - 30m. Wahyuni and Wijaya [2] also use XBee as a communication tool for data loggers monitoring the performance of solar panels. The use of XBee or Bluetooth has distance limitations according to the maximum capability of its range.

The limited communication distance between the server and this measurement device can be overcome by utilizing the existing telecommunications network infrastructure such as the cellular telephone network. Kovacs et al [3] made a remote data logger study for weather monitoring using GPRS GSM network data communication. ATMega64 microcontroller is used as the main component of the data processing processor before being sent to the server. In addition, a PC on site is needed to configure the device. Munandar and Syamsi [4] make Universal Data Logger based on SMS and GPRS Data dial services using GSM networks. The term universal here emphasizes the use of selected communication network choices, namely SMS and GPRS data dial. In both studies, it was not stated that data that has been sent and stored on the server can be accessed online using the internet network.

At present, the world trend of industry has entered the era or fourth generation called Industry 4.0. In further discussion, Marr [5] explains that Industry 4.0 introduces the so-called "Smart Factory", in which Cyber-Physical Systems monitors the factory's physical processes or equipment and makes decentralized decisions. The system is supported by the Internet of Things (IoT) technology, where devices that are used can communicate and cooperate with one another and also with humans in real-time through internet networks [6]. IoT technology is being adapted in agriculture to support increased food security by implementing remote data loggers for agricultural system monitoring and automation [7] as well as monitoring agricultural land [8]. However, it has not been discussed the application of compression algorithms in the process of sending data sent to the server via the internet network.

In this study, a Remote Data Logger module will be created that is able to read sensor data in real time and then process the data and send it to the server using the internet network. Data sent will be compressed first using the Huffman Coding [9] method. On the server side, the data received will be returned to the original data and then stored in the database. It is expected that this compression technique will reduce the data usage quota and increase the security of the data sent.

II. SYSTEM ARCHITECTURE

Online data logger system that has been created is depicted in Figure 1. Sensors read physical variables such as temperature, humidity, etc. The sensor readings will be read by the remote data logger device, stored and sent to the online server. Data transmission to the server is done through the internet network. The data on the online server is processed to be accessed by users through the internet network. Therefore, it will be discussed in more detail about remote data logger hardware, software on remote data loggers, and software on the server.

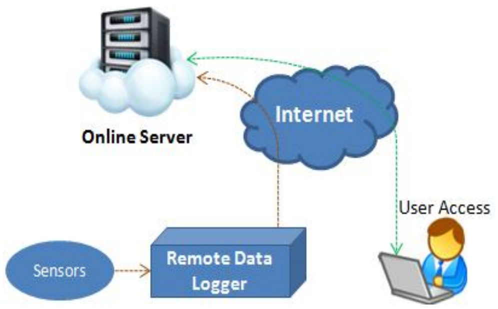

Figure 1. General System Diagram

A. *Hardware System Architecture*

Remote data logger hardware is made as a platform for various types of sensors. Therefore, two types of inputs are made with two different standards. As depicted in Figure 2, the first input is a voltage input the second is the input current. The standard for input voltage is 0-5V. Therefore, the sensor output to be connected to this input must be in accordance with the voltage standard. Or given a matching circuit so that the sensor output matches to the voltage standard. The 0 volt voltage represents the minimum value of the sensor reading, while the 5 volt voltage indicates its maximum reading.

The current input interface is provided for sensors that use the 4-20mA standard. This interface is provided because the sensor standard with a 4-20mA transmission system is widely used in the industrial world. As similar with the voltage input interface, 0 mA current is the lower limit of sensor readings while 20mA is the upper limit. Due to further process, this current input is converted to voltage standard using IC ISO-A4-P3-O4.

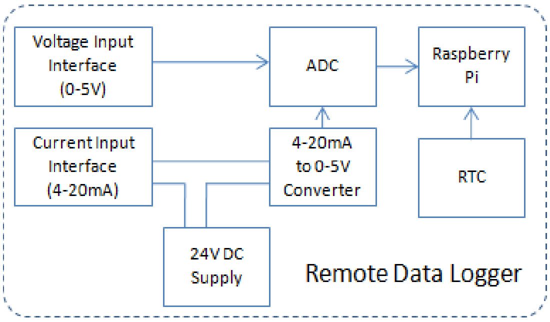

Figure 2. Remote Data Logger Hardware Diagram

Raspberry pi is used as a data processor on a remote data logger. This module reads the sensor data, stores it in local server and then sends it to the online server. Because raspberry pi only accepts digital input data, the ADC is used to convert analog voltages from the input interface. There are two ADC that are used where one is used for input voltage interface, while the other one is for input current interface that has been converted to analog voltage. The ADC IC used is MPC3202 where communication between raspberry pi and the ADC component uses the SPI (Standard Peripheral Interface) protocol.

Because the data logger must record sensor data along with the timed information, then the RTC (real time clock) device is used to provide accurate time on raspberry PI. The IC used for RTC is DS1307. Communication between the Raspberry Pi and the RTC uses the I2C (Inter Integrated Circuit) interface. Moreover, the ADC and RTC components use the integrated EMS Shield Pi Logger module where there are ADC and RTC components in it.

In terms of data connection to the internet, there are three types of connections that can be used, namely using an ethernet cable, wifi, and also a cellular usb modem. The first two connections are certainly more suitable when used in a place that has a local network infrastructure. As for being placed in remote areas, the choice of cellular network connection is a more appropriate choice. Therefore, the choice of connection is based on availability of existing internet infrastructure at the place where the remote data logger is placed.

B. *Software System Architecture*

Software is created on the remote side of the data logger and on the server side. Logically, the program algorithm refers to the diagram in Figure 3. The following will be discussed in more detail for each sub-section of the software referring to the figure. In addition, user interface web is developed for user data access.

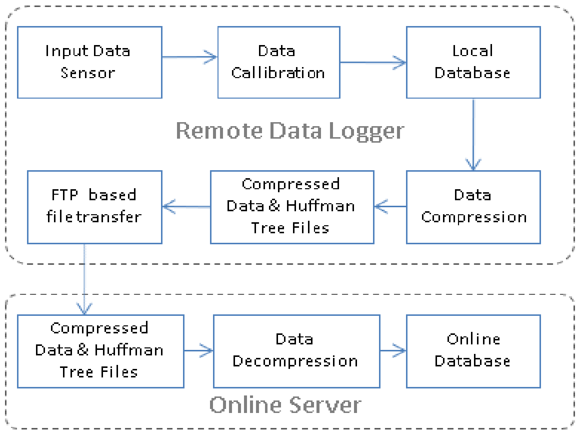

Figure 3. Software Architecture

1) *Reading Sensor Data and Callibration*

Sensor data input is already in the form of voltage value data originating from the ADC device. This data is calibrated so that it gets the value according to the actual sensor readings value. Equation 1 is a formula for conducting a calibration where V_d is the ADC output value, V_{max} is maximum sensor outpot voltage (can not exceed 5V) D_s is the sensor reading value, D_{max} and D_{min} are respectively the upper and lower limits of sensor readings. Also added is the ΔD variable to provide compensation for errors that occur in sensor readings.

$$D_s = \frac{V_d}{V_{\max}} \times (D_{\max} - D_{\min}) + D_{\min} + \Delta D \qquad (1)$$

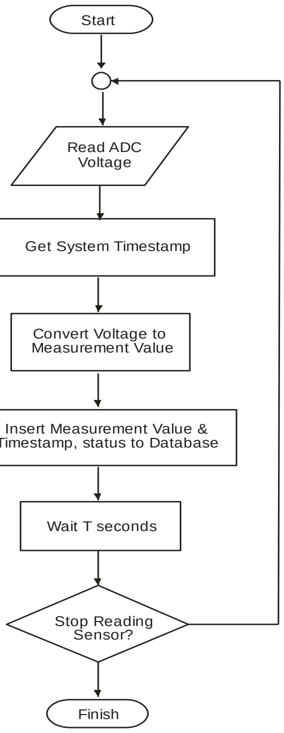

Figure 4. Flowchart of Sensor Reading

After the data is diverted, the next step is to store the data into a local database that is in the remote data logger. At the time of storing this data, also noted the time of storage and sensors. Therefore, each sensor will be given a unique identity to distinguish between sensors one with another sensor. The process of reading and storing sensor data is done in real time with the shortest period of time every minute. This period can be adjusted to the desired requirements.

2) Local Database

Local databases are databases that exist in data logger devices, in this case they are inside Raspberry Pi storage media. This database is used as a backup storage media. On the other hand, this local database can be used as a data storage buffer when sending data to the server is done in a fairly long period such as once a day.

In addition to storing sensor data, this database is also used to store configuration data from remote data loggers, namely sensor configuration, sensor calibration, remote data logger identity, device login, as well as online server address configuration. The local database table is depicted in Figure 5.

3) Data Compression

Before the data is sent to the server, data compression is done to make the data size smaller. The compression method used is huffman code. The process of compression with the huffman code begins by reading the data and then building the huffman tree. Based on this huffman tree, data is encoded and represented in the form of an encoding binary file. This file will be sent to an online server and will be accompanied by a text file containing the huffman tree. Because the sending of files to online servers is done every hour, the sensor reading data that has been stored in the database is compressed and converted into files according to that period.

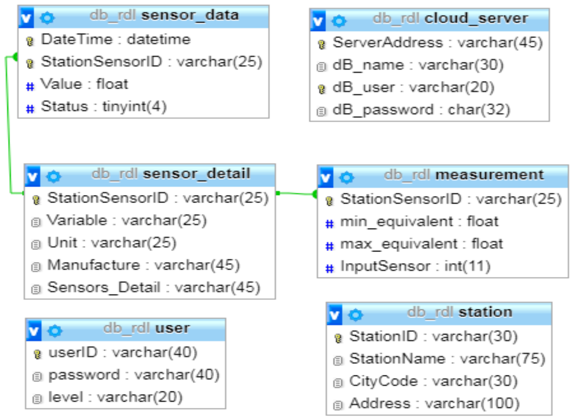

Figure 5. Local Database Table

4) File Transfer

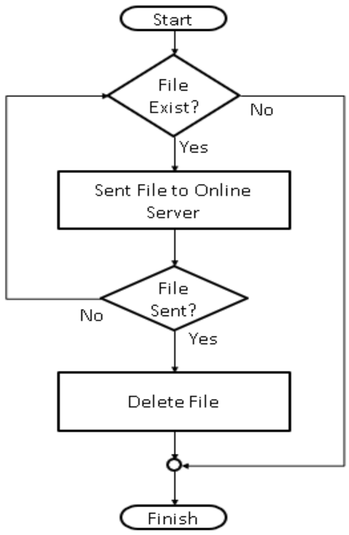

Figure 6. FTP Transfer Flow Chart

Sending data files to an online server is done using the FTP protocol. The choice of this method is because FTP is at the top layer rather than the network layer. So if there is a change in the communication path at the bottom layer of the network, it does not affect the process of sending data. This is one of the things that allows datalogger platforms to use various types of internet network connections.

5) Data Decompression and Online Database

On the server side, decompress the incoming data file and then enter it into the database. Decompressing this file is based on the huffman tree for the file so that original data is obtained from sensor readings in the remote area. The database on the server side is structured according to the database structure in the remote data logger. When adding a new remote data logger, it is necessary to enter the identity of the remote data loger, as well as the ID for each sensor installed on the remote data logger device. This device ID and sensor ID must be the same as those configured in the remote data logger.

6) User Web Interface

This web interface page is used to view sensor data for every configured remote data logger. Its developed for real time data as well as historical data. User can access this web page by browser online. Since the system is aimed to have multi remote data loggers, the web user interface provide option to view particular sensor from certain remote data logger.

III. SYSTEM IMPLEMENTATION

The discussion on the implementation of the system will begin with a discussion per section then proceed with the discussion of the overall implementation. As discuss earlier, remote data logger can have two type of input e.g. voltage input and current input. In this work, two type of sensor were used for testing purpose namely distance sensor and temperature sensor. The distance sensor has 0-5V output where the temperature sensor has 4-20mA output. The discussion sequence is done by calculating implementations ranging from hardware setup, device configuration both on the remote data logger side and online server, to overall system data logger.

A. Raspberry Pi Configuration

Raspberry Pi is a mini computer that functions as the main data processor [11] as shown in Figure 7. In this study, the operating system used is Raspbian Stretch as shown in Fig. 8.

Figure 7. Raspberry Pi Hardware

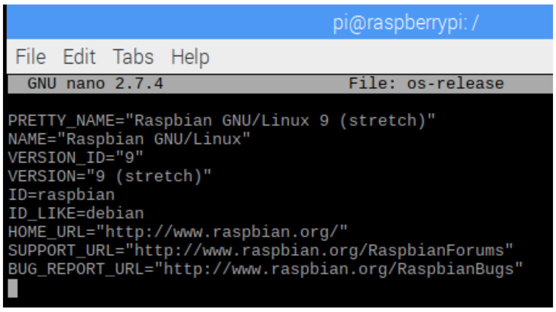

Figure 8. Raspbian OS

B. Real Time Clock

Raspberry Pi by default does not have a real time clock (RTC) module like most computers. This RTC functions as a digital clock that will continue to run with its own battery power supply. Therefore, this research adds an RTC module so that the configuration of the time remains running even though the raspberry pi power supply is turned off. RTC hardware that is used uses the EMS Logger Pi Shield model which is hardware into one module with the Analog to Digital Converter (ADC) component. RTC is used based on the type DS1307 IC. This module is installed directly into the Raspberry Pi GPIO pin as shown in Figure 9.

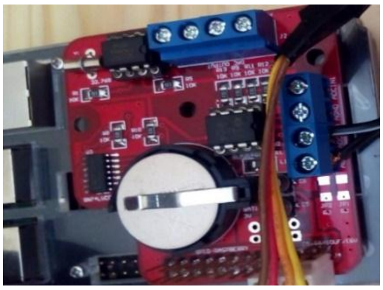

Figure 9. EMS Logger Pi Shield With RTC

This RTC module must be configured first on rapberry pi so that it can be read as well as setting the clock according to local time. In addition, raspberry pi will also use the RTC clock as the main reference. The results of the RTC configuration are shown in Figure 10 below.

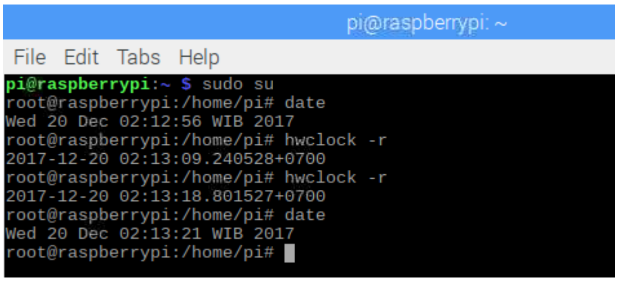

Figure 10. EMS Logger Pi Shield With RTC

C. Analog to Digital Conversion

The ADC used is integrated with the EMS Logger Pi Shield module which has dual 12-bit ADC channels based on MPC3202 with reference to 5V. The ADC on the Pi Shield EMS Logger is shown in Figure 11. Testing this ADC input by providing a variable voltage of 0-5V on the ADC input channel. The ADC test diagram block is shown in Figure 12. In the Raspberry Pi Python program is created which can read and display the results of the ADC reading in the form of volts. The laptop functions to access raspberry pi by using putty software to run an ADC python reader program. Result show that the program reading is similar with input voltage.

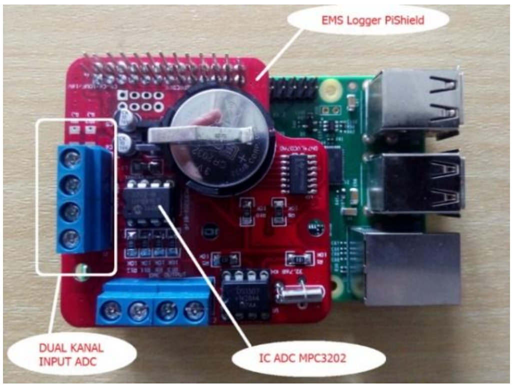

Figure 11. Modul ADC yang Terintegrasi di EMS Logger PiShield

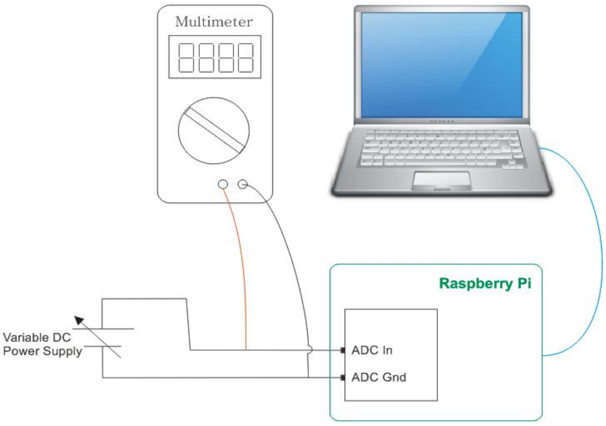

Figure 12. ADC Testing Diagra

D. Current (4-20mA) to Voltage (0-5V) Converter

As discussed in earlier, the data logger device made in this study uses a 4-20mA input standard. Therefore, a current converter is needed to voltage from 4-20mA to 0-5V which then becomes an ADC signal. The converter used in the research is an ISO-A4-P3-O4 type IC. The converter is tested by connecting current sources with the addition of series resistors to reduce the risk of short circuit. The test block block as shown in Figure 13 below.

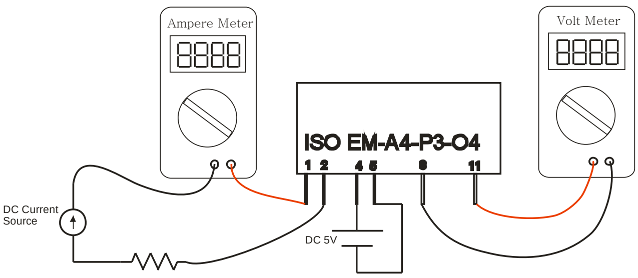

Figure 13. Testing Diagram of 4-20mA to 0-5V

Figure 14. DC Current Source using Long Wei PS-305D

TABLE I. CURRENT TO VOLTAGE TESTING RESULT

Input Current (mA)	Voltage (volt)
4	0,007
5	0,311
6	0,623
7	0,933
8	1,257
9	1,578
10	1,891
11	2,193
12	2,501
13	2,805
14	3,118
15	3,421
16	3,731
17	4,031
18	4,330
19	4,641
20	5,000

The current source used is the PS-305D DC Power Supply which can be adjusted to the output current. The power supply voltage is set to 24 volts. Then the output current is adjusted so as to produce amperemeter readings according to table 1. Then note the output voltage generated by the converter. From the test results in Table 1, it is seen that when the input current is 4mA the output voltage is zero volts. The output voltage increases linearly as the input current increases to the input current 20mA the output voltage is 5 volts.

E. Internet Connection

1) LAN and WLAN

Connection from raspberry pi to the internet can use three alternatives, namely using an ethernet cable (LAN), wifi (WLAN), and or USB modem. On Raspberry Pi there are already LAN interface devices as well as WLAN interfaces. The following is the configuration of the two interfaces.

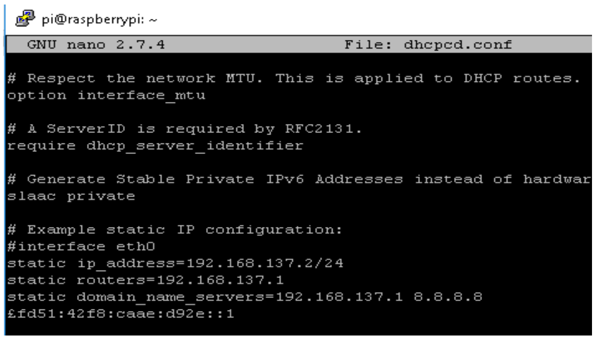

Figure 15. LAN Configuration

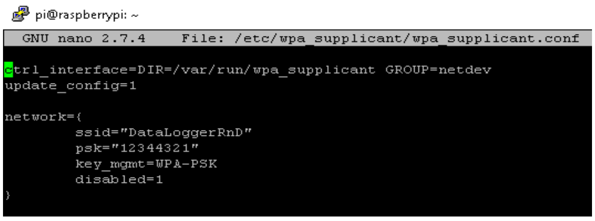

Figure 16. WLAN Configuration

2) *Modem Seluler 3G/4G*

This cellular modem is used to connect to the internet. Using a modem on Raspberry Pi is not like a Windows computer that just plugs and plays. In order for the modem to be used on raspberry pi, the modem configuration must be done, namely installation of usb_modswitch, tunnel ppp, and installation of dial-up software which in this study used sakis3g. For computability testing, the connection is run manually with the command *./sakis3g --interactive*.

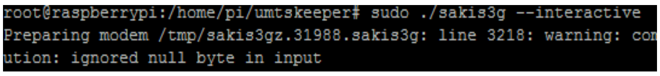

Figure 17. Sakis3g Modem Internet Diul Up

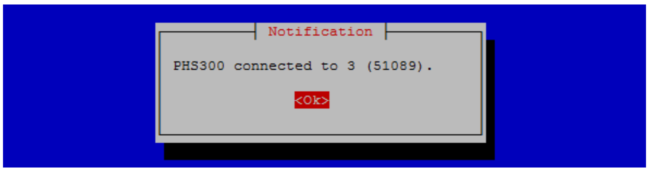

Figure 18. Modem Internet Connected

If the modem connection is done automatically, then information about USB Device ID consisting of Vendor ID (VID) and Product ID (PID) needs to be known by executing the lsusb command on the terminal interface as shown in Figure 4.9. The display shows that the modem used is the ZTE brand with ID 19d2: 1253. The first four digits (19d2) are decimal hex codes for Vendor ID, while the second four digits (1253) are Product ID. This VID and PID information is used for configurations in sakis3g and must be ensured that the ID for modem mode is not storage.

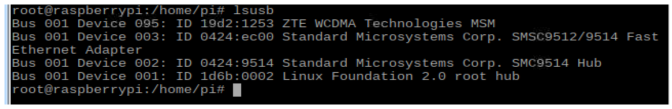

Figure 19. USB VID dan PID

TABLE II. USB MODEM COMPATIBILITY TESTING

No.	Merek dan Model	Chipset & Device ID	Compatibily
1.	Prolink PHS300	ZTE 19d2:1253	Yes
2.	Prolink PHS101	ZTE 19d2:0151	Yes
3.	Vodafone K3765	Huawei 12d1:1520	Yes
4.	ADVAN DT-10	Qualcomm 05c6:6000	No
5.	HNZ 4G Telkomsel Modem	Not Detected	No

From testing several modems, not all brands are compatible with the Raspbian operating system. In this study, 5 modem models have been tested as shown in the test results in Table 2. From the table, modems that use ZTE and Huawei chipsets are compatible with Raspbian OS.

F. *Remote Data Logger Complete Module*

The following figure is a complete form of the remote data logger device. Has two types of inputs and 3 types of internet connections.

Figure 20. Remote Data Logger Hardware Platform

G. *Sensor Attachment*

To test the functionality of a datalogger device, in this study two types of sensors were used, namely temperature sensors and proximity sensors. The temperature sensor will be connected to the input current in the remote data logger, while the proximity sensor is connected to the input voltage.

1) *PT100 Temperature Sensor Attachment*

This temperature sensor is used for testing the input of remote data logger devices for 4-20mA input standards. In this study, the sensor used was PT100 RTD temperature sensor (Figure 21). This sensor is a temperature sensor that has a resistance value that increases with increasing temperature around the sensor. Information about the ideal value of sensor resistance based on temperature can be obtained from the datasheet [11]. This shows that the sensor output variable is a resistance variable.

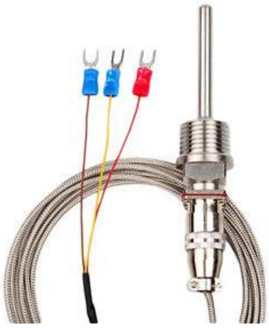

Figure 21. RTD PT100 Sensor

To get an output variable in the form of a standard 4-20mA current, a PT100 transmitter is required such as Figure 22. This transmitter has a standard sensor temperature reading range. For example, if the transmitter input reading range is 0-250 ^{o}C, then at 0 degrees the transmitter output current is 4mA, while at 250 degrees the output current is 20mA.

Figure 22. Transmitter PT100 4-20mA with Range 0-250^{o}C

This sensor has the characteristic that the resistance will increase with increasing temperature around the sensor. The ideal resistance to temperature can be seen in the Pt100 [11] sensor datasheet. For the purposes of testing 4-20mA Pt100 transmitters, a variable resistor is used to replace the temperature sensor. The resistor of this variable is set so that it is equivalent to PT100 resistance at the measurement temperature. Each time the treatment is applied, the resistor is set to a new value to the transmitter to see the output current.

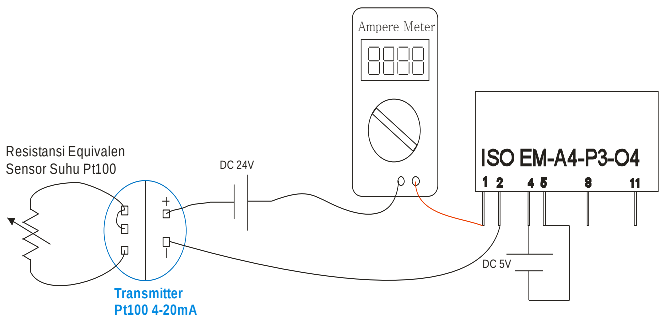

Figure 23. Transmitter PT100 4-20mA Diagram Test

The PT100 4-20mA transmitter has a range / reading range of input temperature and output current. In this study, the transmitter used has a measurement temperature range 0 - 250 ^{o}C with 4-20mA output. This means that ideally when the Pt100 sensor resistance is at zero degrees, the transmitter output is 4mA, whereas when the Pt100 ideal resistance is at 250 degrees, the transmitter output current is 20mA.

Table 3 is the result of testing the 4100mA PT100 transmitter. The results show that a zero output current is obtained when the equivalent resistance value is 100 ohms which represents a temperature of 100 oC. Whereas when the resistor value is 194.07 ohms (representing a temperature of 250 oC), the output current shows a value close to 20mA. Current output increases linearly with increasing PT100 equivalent resistance.

TABLE III. PT100 TRANSMITTER TEST

Resistansi (ohm)	Suhu Ekuivalen (oC)	Rata-rata
100,0	0	4,010
103,9	10	4,639
107,79	20	5,283
111,67	30	5,849
115,54	40	6,562
119,4	50	7,182
123,24	60	7,949
127,07	70	8,471
130,89	80	9,122
134,7	90	9,813
138,5	100	10,433
157,31	150	13,778
175,84	200	17,071
194,07	250	19,994

Testing the temperature sensor is done by placing the temperature sensor in the environment with a certain temperature then measured the output resistance value of the sensor. This value will be compared with the ideal resistance from the sensor datasheet.

There are three scenario conditions that are carried out, namely the placement of sensors at room temperature, temperature of 0 ^{o}C, and at a temperature of 100 ^{o}C. To get the room temperature, the sensor is placed in a closed room assuming that the temperature in the room is not easily affected by changes in temperature from outside the room. A room temperature thermometer is used to see the temperature at the time of testing, then the sensor resistance value is measured.

To run the second scenario, which is 0 ^{o}C, the sensor is placed in a basin containing ice. A thermometer is also placed along with the sensor to monitor the temperature until it reaches zero degrees. Under these conditions, sensor resistance is measured using an ohm meter. The heated water will boil at 100 ^{o}C. This physical property is used to test the sensor at 100 ^{o}C by placing the sensor on boiling water and measuring its resistance. Testing with these three scenarios is shown in Figure 24 and the test results can be seen in Table 4.

Figure 24. PT100 Sensor Test

TABLE IV. PT100 RESISTANCE TEST FOR DIFFERENT TEMPERATURE

Temp. (oC)	R Ideal (Ω)	R Test (Ω)	ΔR (Ω)	ΔT (oC)
0	100	102,5	2,5	6,49
28	110,9	113,4	2,45	6,49
100	138,51	141	2,49	6,47

In table 4 above, Ideal R is the PT100 sensor resistance at the test temperature based on the sensor datasheet, R Test is the sensor resistance measured when the sensor is placed in place according to the test temperature, ΔR is the difference in resistance between R Test and Ideal R at test temperature , while ΔT is the temperature difference between the test temperature and the equivalent temperature of R Test. Based on the datasheet, the change in temperature per 1^{o}C causes a change in resistance of 0.3851Ω. The magnitude of the ΔR value will cause a temperature reading error of ΔT. Therefore, the value of ΔT will be used as a calibration value according to equation 1.

1) Distance Sensor Attachment

The distance sensor used is Sharp 2Y0A21 which has a measurement range between 10-80cm. Outside of that range, sensor readings become inaccurate. This sensor works by using infrared light, has 3 pins where two pins for the power supply while the other pin is the output of the sensor. Because the output of this sensor is not linear with distance, Arduino nano V3 is used to adjust the distance so that the output voltage is generated with a range of 0-5V. So that the distance of 10cm will produce a voltage of 0 and 80cm will produce a voltage of 5V, which is used as the input voltage from the remote data logger.

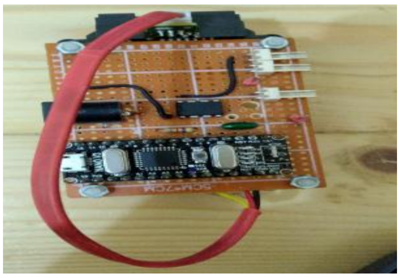

Figure 25. Distance Sensor Module

H. *Software Implementation Remote Data Logger*

A Web based dashboard was developed for remote data loggers. This is used to provide identity to the remote data logger, configuration of sensors, configuration of online server addresses, calibration of sensors, to see the reading of sensor data. Figure 26 is the main page of the remote data logger device dashboard.

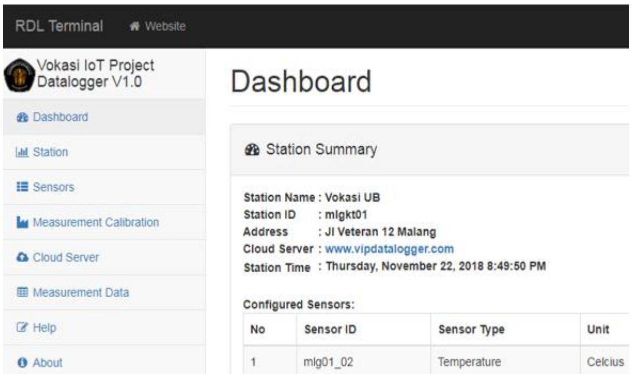

Figure 26. Remote Data Logger Dashbord

Each remote data logger device installed is considered a station. Providing identity for each station is important because the system is made to accommodate the use of multi stations. On the other hand, each sensor installed is also given a unique identity. It aims to distinguish reading data from sensors one with another sensor. Giving identity sensors is done using the IdStation_nomorSensor rule. Figure 27 is a display of station identity configuration while Figure 28 is a sensor identity configuration.

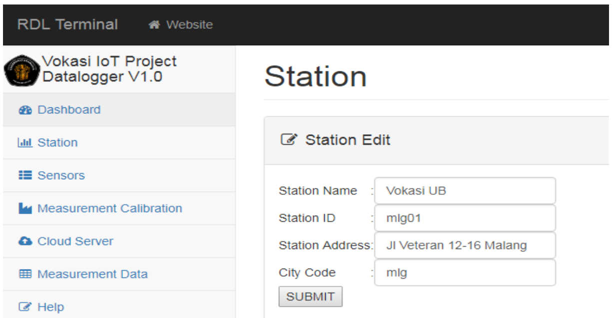

Figure 27. Station Configuration

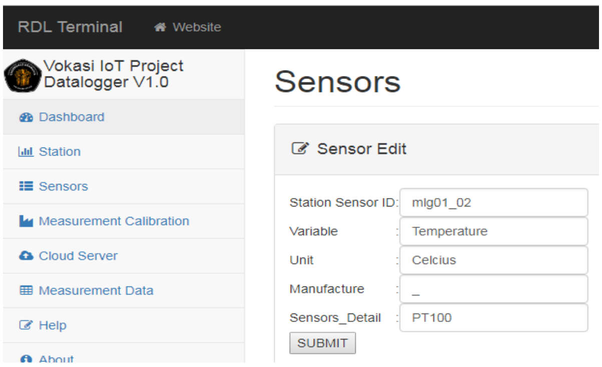

Figure 28. Sensor Configuration

As explained earlier, the system made accommodates the sensor calibration process. This calibration is needed because the sensor reading is not necessarily accurate as it should be. Figure 29 is an example of a calibration for a temperature sensor reading a temperature sensor.

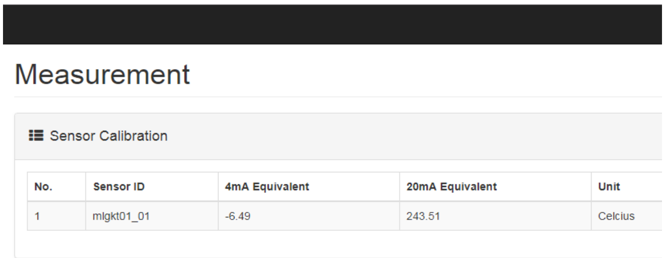

Figure 29. Sensor Calibration

I. *Server Datalogger*

The data logger server is the center of data collection sent by all sensors on all remote datalogger devices. The server for the trial can be accessed online at the web address. The data on the server can be displayed in table view or in chart view as depicted in Figure 30.

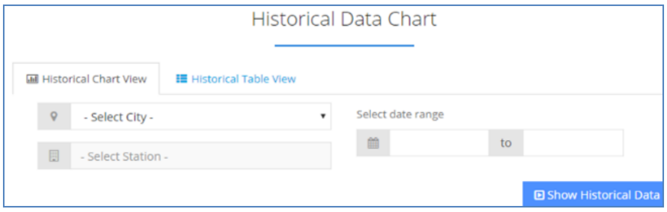

Figure 30. Online Data Logger Display

IV. RESULTS AND CONSLUSION

In this study a remote data logger device platform was created. The platform created allows input with input voltage or current input which is then processed and forwarded to the server via the internet. There are several types of internet connections that can be selected, namely ethernet cable connection, WIFI, or using a cellular USB modem. Provision of input with a standard voltage (0-5V) and current (4-20mA) provides the degree of freedom of the type of input sensor used. So that the platform created can be used for a variety of applications. On the other hand, internet connection options provide benefits when deployment in areas with various existing internet infrastructure alternatives.

REFERENCES

[1] Siew, Z. W., Wong, C. H. , Tan, S. E., Yoong, H. P. and Kin Teo, K. T. 2012. *Design and development of a tablet based real time wireless data logger*. IEEE Global High Tech Congress on Electronics, Shenzhen, 2012, pp. 111-116. doi: 10.1109/GHTCE.2012.6490135

[2] Wahyuni, R. T. dan Wijaya, Y. P. 2016. *Solar panel performance monitoring system in sensor node*. International Symposium on Electronics and Smart Devices (ISESD), Bandung, 2016, pp. 92-96. doi: 10.1109/ISESD.2016.7886699

[3] Kovacs, A., Nicolcioiu, A., Arhip, J., dan Cașu, G. 2014. *Design and implementation of a GPRS remote data logger for weather forecasting.* 2014 10th International Conference on Communications (COMM), Bucharest,, pp. 1-4.doi: 10.1109/ICComm.2014.6866708

[4] Munandar D. dan Syamsi, D. 2014.*Data logger management software design for maintenance and utility in remote*.The 1st International Conference on Information Technology, Computer, and Electrical Engineering, Semarang, 2014, pp. 74-78. doi: 10.1109/ICITACEE.2014.7065717

[5] Marr, B. 2016.*What Everyone Must Know About Industry 4.0. Majalah Forbes*.https://www.forbes.com/sites/bernardmarr/2016/06/20/what-everyone-must-know-about-industry-4-0/#71d43f90795f

[6] Lee, J., Bagheri, B., Kao, H. 2014. *A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems*. Jurnal Society of Manufacturing Engineers (SME) 2014.http://dx.doi.org/10.1016/j.mfglet.2014.12.001

[7] Mohanraj, I., Ashokumar, K., Naren, J. 2016. *Field Monitoring and Automation Using IOT in Agriculture Domain*, Procedia ComputerScience, Volume 93, 2016, Pages 931-939, http://www.sciencedirect.com/science/article/pii/S1877050916315216

[8] Prathibha, S. R., Hongal, A. dan Jyothi, M. P. 2017. *IOT Based Monitoring System in Smart Agriculture*, 2017 International Conference on Recent Advances in Electronics and Communication Technology (ICRAECT), Bangalore, 2017, pp. 81-84. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8081906&isnumber=8081873

[9] D. Dath, D. dan Panicker , J. V. 2017. *Enhancing adaptive huffman coding through word by word compression for textual data*. 2017 International Conference on Communication and Signal Processing (ICCSP), Chennai, pp. 1048-1051.

[10] Pimylifeup. (2015). What is a Raspberry Pi: All you need to know about the Pi. [Online] https://pimylifeup.com/what-is-raspberry-pi/

[11] Omega. (2017). RTD Temperature vs. Resistance Table. [Online].https://www.omega.com/techref/pdf/z252-254.pdf.

Harnan Malik Abdullah received the degree in electrical engineering from University of Brawijaya, in 2006 and master degree from Department Communication Engineering The Sirindhorn International Thai German Graduate of School Engineering (TGGS), King Mongkut's Technology of North Bangkok (KMUTNB), in 2012. His interests are in Internet of Things, Networking and Data Analytics.

Fridy Mandita received the degree in computer science from Gadjah Mada University, in 2007 and master degree from Department Software Systems Engineering (SSE) The Sirindhorn International Thai German Graduate of School Engineering (TGGS), King Mongkut's Technology of North Bangkok (KMUTNB), in 2014. His interests are in Natural Language, Machine Learning, Digital Image Processing, Automatic Differentiation, and Information Systems.

How to Cite this Article:

Abdullah, H. M. & Mandita, F. (2019). IoT Remote Data Logger Platform for Multiple Sensors and Robust Data Connection. International Journal of Science and Engineering Investigations (IJSEI), 8(85), 77-85. http://www.ijsei.com/papers/ijsei-88519-13.pdf