Raspberry Pi SCADA Zonal based System for Agricultural Plant Monitoring

Ata Jahangir Moshayedi
School of Information Engineering
Jiangxi University of Science and Technology
86 Hongqi Ave, Ganzhou, Jiangxi 341000, China
e-mail: *ajm@jxust.edu.cn

Atanu Shuvam Roy
School of Information Engineering
Jiangxi University of Science and Technology
86 Hongqi Ave, Ganzhou, Jiangxi 341000, China
e-mail: atanuroy911@gmail.com

Liefa Liao
School of Information Engineering,
Jiangxi University of Science and Technology,
No86, Hongqi Ave, Ganzhou, Jiangxi, 341000, China.
liaolf@xjtu.edu.cn

Shuai Li
Department of Computing, Hong Kong Polytechnic
University, Hung Home, Kowloon, Hong Kong,
shuaili@polyu.edu.hk

***Abstract*— SCADA (Supervisory Control and Data Acquisition) and DCS (Distributed Control System) are both famous terms in automation and they both have critical rules. There is rarely an industry in the world which are not auditing their outcome based on one of these terms. But IOT developed in recent years, mixing the functionality of these two terms has been going on too with various security challenges involved. Even some previous issues like the price of software and easy extendibility of them make the way for small industries to have their benefits. In this research paper along with the designing and comparison, two small SCADA systems for agricultural applications tries to reduce the size as well as address a zonal method to investigate the security problems. Till these days' researchers have shown lots of development to reduce the SCADA cost in different manner but, having the compact, powerful and small size host for SCADA system has not been reported before. One of the novel aspects in this design is hiring the Raspberry Pi as the smallest computer instead of PC or laptop which along with reducing the system cost can also reduce the whole design system. Besides, Web based SCADA system has been successfully designed here and used. This paper discusses the context of local as well as remote monitoring of an agricultural plant environment with the help of a Raspberry Pi network through which a SCADA System was established. This paper aims to prove that a far more efficient and cost-effective along a secure SCADA System for monitoring is possible with the lowest of efforts. So, having zonal idea will help system to achieve the aim of secure sensor monitoring without any kind of outside interference. The result shows the successful design and implementation of zonal idea as well as the perfect usage of Raspberry Pi as a small and reliable host.**

Keywords- ***Raspberry Pi, esp. module, web database, Arduino, internet of things (IoT), agriculture, zonal web-based SCADA***

I. Introduction

SCADA, or supervisory control and data acquisition system, is one of the systems that has been used since the 1970s. The SCADA system is used to monitor and control factories or equipment to acquire and monitor data. Initially there was a big gap between this term and another controlling system called DCS (Distributed control system). But nowadays, both these terms' features have become near in the way that the big wall between them is collapsed in many applications as they act under the same rule. In brief SCADA major part are: SENSORS to communicating with the outside environment, CONVERSION UNITS which consists of RTUs (Remote Terminal Units and PLC as the Electronic devices controlled by a microprocessor with the aim of interfacing to the sensors and send data by wire or wireless communication, MASTER UNITS: Used as the supervisory computer system which serve as the SCADA systems central processor. This part offers a human interface (HMI) and automatically regulates the system based on information from the sensors. the final section COMMUNICATIONS NETWORKS: uses some type of communications technology like: wired (telephone lines, WAN circuits) and wireless (radio, cellular, satellite) to connect the conversion units to the SCADA master unit[1]. Except the sensor types, which are mainly fixed for different applications, researchers try to introduce and use different methods in the rest of the three major parts to reduce cost, power usage and increase the system efficiency[2]. Especially after the Industrial IOT was introduced, the web based SCADA are more focused.In today's world, monitoring in the plant and farming field have increased significantly and having the automation in monitoring system helps farms and agricultural plants to have better and more qualified products.Mahir Kaya et al[3] proposes a system for Complex event processing-based software framework for smart plants with the help of low cost sensor nodes based on Arduino. Each node in their design collects information, and post processes the data for later evaluation. The Service Oriented Architectures (SOA) Architecture (EDA) that is triggered by events like AI, in the mentioned design, an ESP8266 card for communication. Java socket is used to collect data on the server side and on the central server the rules are defined by a rule engine. C.R Rocha et al[4] introduce a conceptual design to an open source SCADA system that can run on low power consuming computers in automation applications for plants. Their designed system consists of several packages like comm package, HMI package, AUX package each doing their individual job such as app package, In their proposed system they use remote

978-1-7281-5712-2/19/$31.00 ©2019 IEEE
DOI 10.1109/ICISCE48695.2019.00092

user interface for supervision and control. Prosanjeet J. Sarkar et al[5] in their survey on IoT based Agriculture Monitoring System proposes a system where they set up sensors(temperature, moisture ,humidity ,PH value,etc) with a micro controller set up, that transmits digital data to the cloud through a ESP8266 module which also works as an internet gateway to monitor agricultural parameters. This module serves as a component for wireless data transfer while the receiver uses serial protocol i.e. UART. Hence, remote data monitoring of the environmental parameters easily allows maximum yield by providing farmers/workers necessary information. Pastor et al[6] uses MQTT(open messaging protocol) and RESTful API on cloud services to establish a SCADA System and the design aimed to monitor situation in precision agriculture. In the proposed system three embedded devices used to control a total of three sensor networks each consist of various sensors like PH, EC, temperature, radiance, luminosity, soil, etc. Besides, each embedded device has MQTT broker to take in data while the REST API helps in exchanging data in between devices and central server on cloud. They used ESP8266 and Raspberry Pi in the different embedded device setup and connected all of them to the cloud hosted by Ubidots IoT Platform and to each other via a router. As the outcome mentioned, there are no dependencies on proprietary technologies. Thus, new sensor, nodes and control devices can easily be added to the system and hence the system can be expanded with new algorithms set by the agronomist; So, because of this interoperability data aggregation truly helps to improve control over the plants. Gondchawar et al [7] shows an IoT based agricultural smart environment monitoring system with the different nodes to have the remote sensing on robot platform and controlled via commands to the RPI computer which in turn sends command to the micro-controller itself. Their design SCADA system comprises of the components such as AVR Micro-Controller, ZigBee Module, Sensors, and the Raspberry Pi computer. Their system uses the wireless communication to access the various Nodes. Krishna et al [8] have shown a smart IoT based agricultural system that can be used in SCADA. Their system has mobile robots with various sensors types such as humidity sensor, pH sensor, thermo-hydro sensor, CO2 sensor, soil moisture sensor, and obstacle sensor. By using Raspberry Pi implemented on the robot platform and controlled via Wi-Fi they are sending data via wireless system to database. They conclude that the robot is hence used to reduce labor costs and increase efficiency. The data is sent from the robot platform using a ZigBee receiver to receive data and sent to the computer that they use as a server and use HTML and PhP page to display stored data as GUI. Prosanjeet J. Sarkar et al [9] shows a design system that follows a way of monitoring through data mining. In a targeted environment, the sensor nodes collect various data through the combination of sensors and Arduino Uno and then, through an Ethernet shield, sends the data to an IoT extension service called Blynk. They monitor the system through the Blynk app and conclude that their system allows multidisciplinary model to observe and interact with the physical environment through the cloud. Hence improving information exchange that helps in the overall yield, reliability and effectiveness of the plants. As the mentioned reviews shown above, the Arduino, pi and ESP based module like NodeMCU as the CPU and controller are more in demand in the recent low-cost SCADA systems. The main problem of the prevailing SCADA system is security. This issue force most of the current SCADA used in industry to work based on the belief that because SCADA is usually disconnected from the internet it is safe. But, still there are many ways to access and hack the disconnected system from the internet via virus/malware, etc. Vinay M. Igure, 2006, et al [10] pointed the PROFIBUS Protocol which has slow data transfer rate. Access control, firewall and intrusion detection systems as the conclusion suggests should be upgraded along with having better cryptographic key management, upgradation of device and OS security. Rodrigo Chandia, 2008 et al [11] proposes three key management methods named Hash Based key management, PKI based key management, and symmetric key management solutions for SCADA. They also mention strategies like forensic solution which presents a forensic architecture to support the capture, storage and analysis of SCADA network traffic, or introduce some network topology like grid topology and random topology [12]. Another difficulty which has been reported for the current existing SCADA System is the SCADA software which depends on OS (operating system) to acquire data have industry level pricing. To mitigate this problem, Web-based SCADA evolved. Web based SCADA simply uses a database server and a webhost server which his very easy to accomplish. Moreover, the software to be used on it is completely open source and for this less prone to run into issues. The main advantage of web-based SCADA system is that it can run as a silent application without the hassle of installing extra software and drivers and other dependencies like specific HMI. Both systems are built on Zonal idea accessing data in the field of agriculture and farm. The zonal model of SCADA with the limited accessing in each Zone can help to create a more secure impenetrable system. The zonal idea or Multi DCS/SCADA implemented in the Agricultural plants based on emergence of different types of data and level of monitoring. The other novel work designed system is hiring the Pi as the small host which can rectify the use of big computer, pc etc. The rest of this paper is organized as follows: in the Initially, designed SCADA System section the zonal Idea or Multi DCS is described. Then the Designed System and component in order to introduce both systems hardware and software with all its related parts have elaborated in third section. Finally, the paper has concluded with the achieved result and future work in section 4.

II. DESIGNED SCADA SYSTEM

A. Zonal idea or Multi DCS/SCADA

As it is mentioned before, the term ‘DCS’ describes controlling the system from various place and distribute it in different Zones, but monitoring and accessing from and in different Zones’ issue remain and is reflected in Zonal idea or Multi DCS/SCADA idea. The Zonal idea is basically indicating the level of accessing data in the hierarchical way to have secure monitoring. This idea tries to sort the security problem in SCADA Systems by using zonal idea. It goes to that each area based on accessing parameters into different Zone and send data or command with respect to their current Zone. The zonal idea or Multi DCS/SCADA has shown in Figure (1). As the Figure (1) shows, Zonal idea or Multi DCS/SCADA is basically the idea of accessing levels in local and global. The zonal idea can be recalled the multi DCS (Distributed control system). In the local Zone1 the controller which can be any controller, can acquire the data as the first window. the data in this stage just can be monitored in local place with any HMI and send it to Local Zone 2.

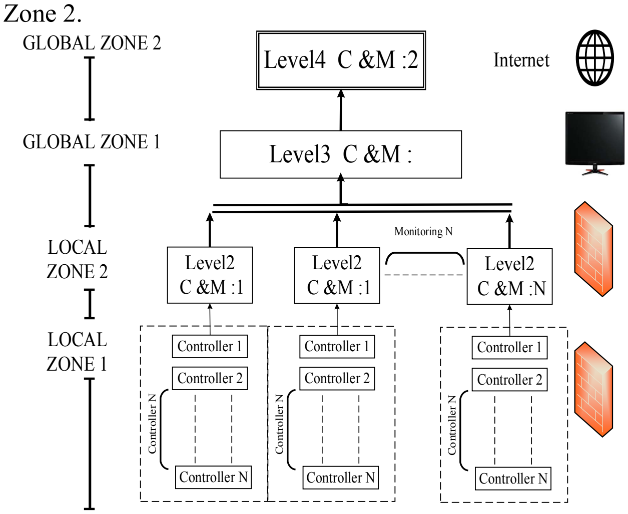

Figure 1. Fig 1:ZonalModel for SCADA Network

As the second stage “local Zone 2” data from each group can be observed and control but each part form this level is isolated from each other’s, the third stage called “Global Zone1” the data from local Zone 2 can be shown and controlled, then the secure data or filtered data can we access by “Global Zone2” which is based on internet and can be access in any place. Functional zoning is very beneficial once the firewall side comes into play because the access rules and restrictions should be largely similar across the specific Zone. Then Hackers will have a pretty hard time trying to get in such a secure place. More than the security side, control of the system, becomes very easy. And even if one Zone fails to function, other Zones keep on working just fine. Moreover, in the case of data analyzation and big data observation, the speed and decision will be faster.

III. DESIGNED SYSTEM AND COMPONENT

As stated, the zonal idea or Multi DCS/SCADA is being implemented in proposed SCADA system for agriculture application. Based on review paper two different system based on Arduino and NodeMCU are proposed. Then, performance compared. But instead of using any external local Host for local Zones and global Zone 1 the Raspberry Pi used. The implemented designs are Wire based and Wi-Fi based in local Zone 1 and are spliced into two parts of hardware by using :(a) Raspberry Pi, (b) Arduino, (c) ESP8266, (d) Sensors which include, DHT11, BMP180, SHT1X, Gas Sensor, PIR Sensor, Motion Sensor etc. And software’s like (a) Raspbian Operating System (b) PhP-MySQL (c) Python IDE (d) Arduino IDE (e) Apache2 with the help of programming LANGUAGES of (a) HTML (b) PhP (c) MySQL (d) Python (e) C+ which are described as following

A. Design System Hardware

- **Arduino Micro-Controller Boards***:* An open source platform for building electronic projects which has both hardware micro-controllers and Software IDE to upload code in this design. In order to interact with the sensors, two type of Arduino Board (Arduino UNO and Arduino MEGA 2560) is used [13].
- **Raspberry Pi3 model B/ B**+: In this design, the Raspberry Pi, as the small PC hired to implement all the scripts and related server hosting. Also, as the OS (Operating System) Raspbian Stretch (version 9.4 and Buster 10) has been used for Raspberry Pi boards (Raspberry Pi3 model B: ARM Cortex-A53 1.2GHz, 1GB SRAM, 1200MHZ, Raspberry Pi3 model B+: ARM Cortex-A53 1.4GHz, 1GB SRAM, 1400MHZ) [14]. The Raspberry Pi 3B/3B+ has 4 USB ports. According to the official standards of the USB or Universal Serial Bus, up to 127 devices can connect to a single USB network or port. Hence as the RPI (Raspberry Pi) has a total port of 4, a total of 127*4=508 devices can be connected to the PI theoretically. But there is also the factor of power. Here the reason for using 2 different versions of Raspberry and different OS are just the cause of availability. Even it should be noted that the Raspberry Pi b+ is more powerful that the B, and the software availability like Ubuntu Mate is limited for the B+ hardware [14].
- ***NODEMCU 12E***: In order to have the small and Wi-Fi based system, the NODEMCU 12E as the ESP8266 based module has been used. This module is a perfect combination of both the Wi-Fi shield and the Arduino board. The NodeMCU 12E module is capable of either hosting an application or webpage within its own coding. The module is cheap and great for prototype applications

and small projects and at the same time. Also, can be program with the Arduino IDE [15].

- ***Sensor***: As mentioned before, various sensors based on agriculture application are selected. Mainly, in agricultural farm the parameters like, pressure [16], motion detection [17], level of gas sensor [18], temperature [19], temperature and humidity [20], light and are attached to both systems. The important note is shown in TABLE (1).

TABLE 1： selected Sensor important point

NO	Sensor Type	Sensor/unit	Accuracy	Range
1	Humidity, Temperature	SHT10	RH: ±4.5 Temp: ±0.5	Operating Range: -40 to 123.8 °C
2	Pressure, Temperature, Altitude Sensor	BMP180	Pressure:±0.12	300-1100 hPa
			Temp: ±0.5, Resolution: -40 to +85°C,+-2°C	
3	Light sensor	LDR/ (Lux)	±5	10 Lux to 1000 Lux
4	Gas Detection like H2, LPG, CH4, CO, Alcohol, Smoke	MQ2/ ppm		300-10000 PPM
5	Motion Sensing	PIR / (meter)		Distance: Stationary: Up to 4M Moving: Up to 7 Meters Horizontal FOV: 110° Vertical FOV: 90°
6	Temperature	MAX 6675K/°C	~0.25°C	Detection: 0-1024°C Operating : -20 ~85°C

B. Software

In both designs the below software with the mentioned task are used;

- **Raspbian Operating System:** an open source OS based on the Debian operating system
- **PhP-MySQL**, a program from the default Raspbian repo that manages and integrates PhP and MySQL.
- **Python IDE,** a program that allows us to program and run Python Coded programs in real-time.
- **Arduino IDE:** an interactive development environment that allows user to write programs and write them to microcontroller boards.
- **Apache2:** a software that hosts a virtual webserver on local machine
- **HTML** hypertext mark-up language, a web design language
- **PHP:** a hypertext pre-processor programming language for web design used mostly for extensive data collaboration from server and client communicating between hardware and software.
- **MySQL: MySQL** is an open-source relational database management system.
- **Python:** an interpreted, high-level, general-purpose programming language.
- **C++:** a high-level programming language used to program the micro-controller boards.

C. wire based system – Arduino to PI System(AP)

The system 1 is based on zonal idea or Multi DCS/SCADA and wire connection from local Zone 1 to local Zone 2 are shown in Figure 2.

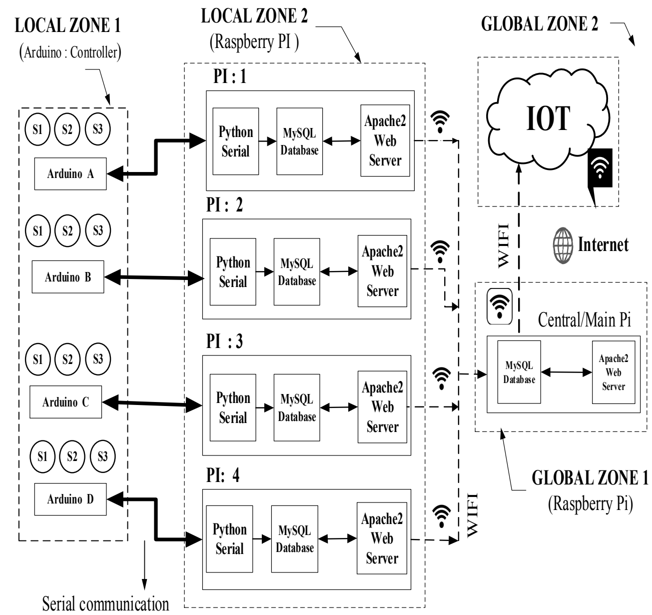

Figure 2:The System 1

As it is illustrated in Figure 2, the designed system which segmented into local and Global Zones. The system 1 design is to capture the output of the Arduino board through the serial monitor (Serial micro-USB to USB-A Cable). The whole system consists of four Arduino controllers in Local Zone 1 and four Raspberry Pi in local Zone 2 and one Raspberry Pi in global Zone 1.

D. Wifi based System 2 Esp to PI System(EP)

The system 2 as a Wi-Fi provision and zonal idea or Multi DCS/SCADA are shown in Figure 2. In this design NodeMCU 12E ESP8266 Module as the local Zone 1 connected to local Zone 2 with the Wi-Fi facility.

E. WEB page section

As it is illustrated in Figure 2 and Figure 3, the designed system is segmented into local and Global Zones. The system 1 design is to capture the output of the Arduino board through the serial monitor (Serial micro-USB to USB-A Cable). And the system2 use the benefit of Wi-Fi by NodeMCU 12E. for both systems the sensors are connected to the controller and the board is connected to the Raspberry Pi after pre-processing the byte-like data into valid integer format, sending it to the server for the user.

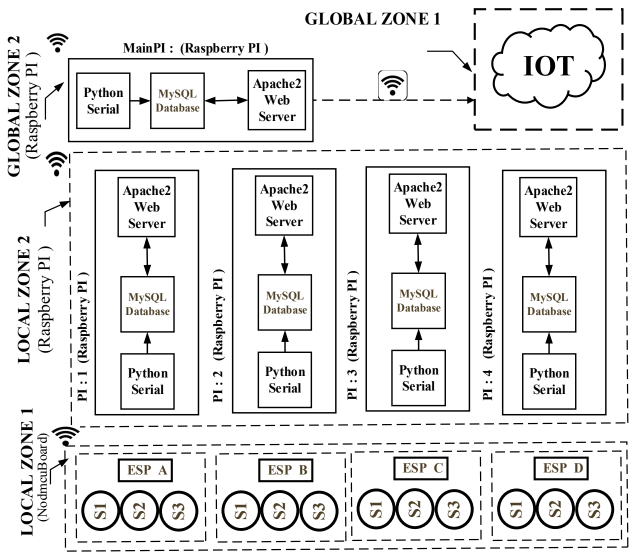

Figure 3. ESP to PI SCADA over Wi-Fi

The designed system splits the single string byte-like data into two or more (as per the number of outputs sent by the sensors) sections of individual data, which the python script does and sends them to the MySQL database. While for System 2 the same procedure happens only difference is the data type is string and the data doesn't need to be pre-processed hence it is directly posted through URL inline to a PhP post script on the server and the script saves the data in the database. As it is mentioned, in this design with the aim of compact and small host the Raspberry Pi is used, and the HTML pages as the GUI (Graphical User Interface) is used inside the Raspberry Pi.

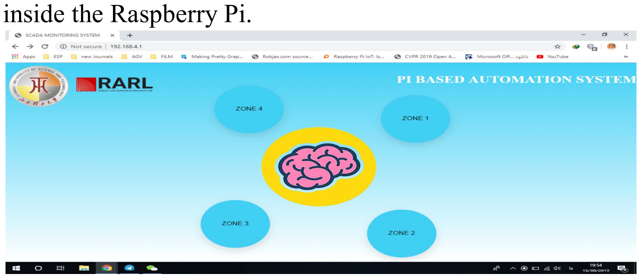

Figure 4: the web-based GUI

The Figure (4) shows, the main greeting page is offering the choice of selecting each local Zone from one to four. The user by selecting each Zone can enter and observe Zones. The Figure (5A) shows the web page designed inside the Raspberry Pi. The whole website is hosted inside the Raspberry Pi computer. Each Zones have their own hosting service running in the background. This allows the users to monitor the webpages locally without the need of any 3rd party services like outside hosting or IoT cloud services. When the user opens the ip address of the specific Zones the user wants to monitor, the direct monitoring and controlling interface of the sensors established in that Zone comes up on screen in the browser. The data and chart display on the webpages directly correspond to the data uploaded from the sensors to the database working via python script. The website has in built PhP functions to export existing data from SQL format into excel sheet which is not shown on the website considering security as well as functions to truncate the TABLEs of data if necessary. CGI-Scripting is still at work to provide further control over hardware from webpage. Once, the user logs into the main PI after connecting to the internal Wi-Fi network also hosted by the main PI itself, the user is greeted with the choice page offering the user to select one of the four Zones that have been set up in this project. Each zonal webpage has its own database and clicking on any one of the Zones will bring up that specific Zone's control and monitoring interface. The Figure 5 shows the Webpage sections inside each Zones. On the screen's leftmost side, the user will see a short small summarized interactive chart consisting a graph of the last five data entries of that Zone's sensors. On top of the chart there are some buttons having the functions of refreshing the page, showing full chart or go back. As the same Raspberry Pi systems are used for both systems named AP (Arduino-Pi) and EP (ESP-Pi) system a special button has been added to go to the webpage that displays the data from the database where the NodeMCU system (which is ESP8266 based module) sent the data to. The user will see a very similar webpage to the AP system webpage and hence the navigation here is also the same. On the bottom left corner under the summarized chart there is a small box that right at this moment is only capable of sending small numerical commands to the micro-controllers itself. So, the basic flow of webpage is as follows: Connect to the Network, Open the URL/IP of the Main PI -, then Select Zone and finally View Data on Webpage

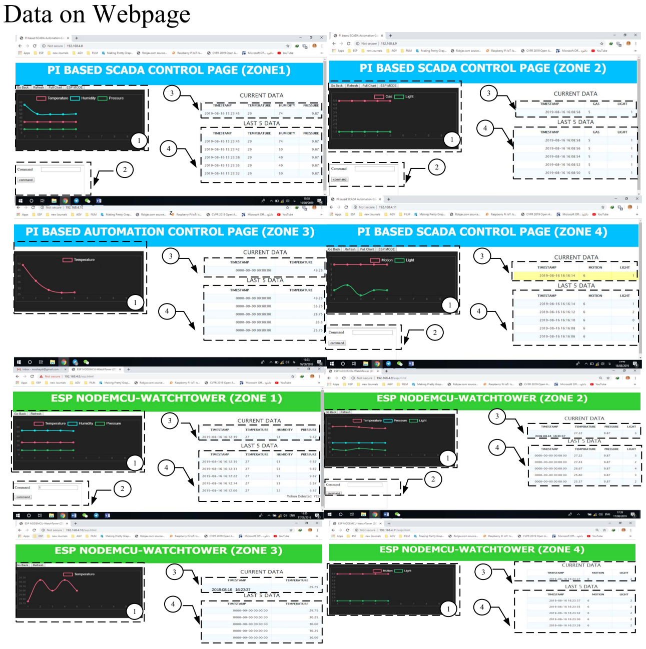

Figure 5: 1: Summarized chart and Navigation Buttons, 2: Command Prompt Box,3: Latest Data Tab,4: Last 5 Data Tab,5: Zone Selection Buttons

IV. EXPERIMENTATION AND RESULTS

To test the both design system, the systems are install in various parts of Golden campus yard, Jiangxi university of science and technology (120 x 100 Meter) and power by the portable power bank (Lithium ion Polymer Battery,12000 mAh) and Wi-Fi range is from 10 to 15 meters with limitation considering the for the Raspberry Pi Wi-Fi which can be extended by the external USB Wi-Fi antenna [33]. The real design system installation is shown in Figure 6. with respect to Zonal idea (the Figure (2) and (3)), the local Zone and the global Zone are connected together with the Raspberry Pi. All the Raspberry Pi systems have static ip addresses so that on the same network no extra variable is found lying around. Once all was set, we just set up the main and final pi system that is used to monitor all the other pi systems. All the pi systems can be connected via a single WIFI or interconnected Ethernet hub. The main pi has a single database and an apache2 web server running to just host and view all the other pi systems' work. If the main pi has internet connection, the system can set up a static open DNS service on the pi and then control and monitor it from anywhere in the world.

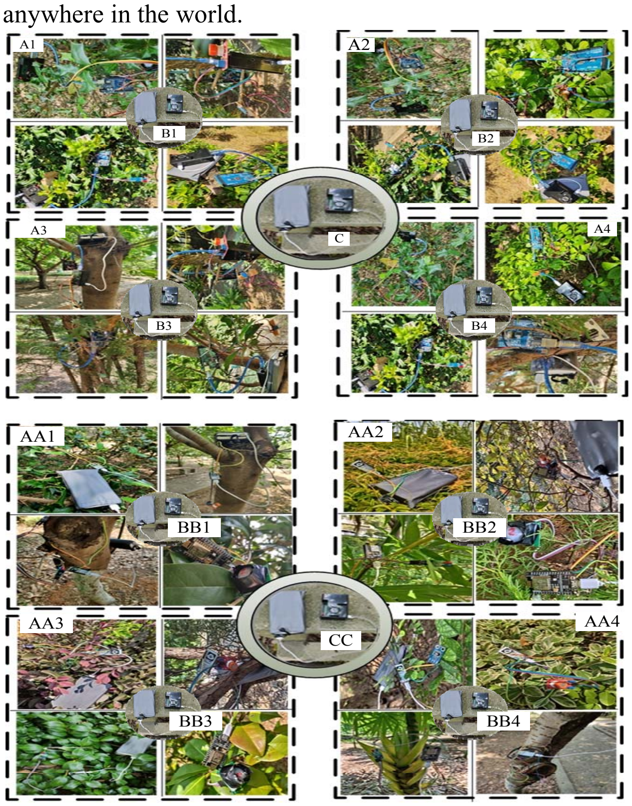

Figure 6: Setting up the sensors in actual natural environment

With respect to Security concern in both design, Security Protocols of the System shows that, the system is designed as such that each Raspberry Pi unit is guarded by its own password and firewall. Database password for each database is different and as well as the port is changed for further protection giving potential network penetrators much more work than usual. Each Zone is under its own encryption and password system. REST API is enabled with self-generating tokens for every session and the server generates zero cookies to save sessions making the environment even more secure. On the IoT end only monitoring is enabled hence only alteration can be done from within the local intranet only. Only the final or main pi is enabled internet connection. Despite compare and evaluate both proposed systems, its detected that The NodeMCU 12E, ESP based module built as the extension of Arduino board to have the WI-FI and it has its SOC (System on Chip) with integrated TCP/IP stack. Even the NodeMCU have some advantages but the it has some cons and limitations like, when the workload increases, and more data needs to be processed and managed it falls short. Moreover, experiment suggests that the supplying big amounts of data to large number of clients at the same time has been near fruitless for the ESP Module. The main features and comparison between NodeMCU 12E and Arduino are shown in TABLE (2).

TABLE 2: the comparison features

	Features	NodeMCU 12E	Arduino
1	Data Transfer Protocol	Wi-Fi 802.11 b/g/n	Serial, UART, I2C, SPI
2	Speed	80Mhz	16Mhz
3	Size (mm)	24.0 × 16.0	68.6 × 53.3: [UNO], 101.6×53.3: MEGA2560]
4	ADC	1 pin	6pin
5	Digital	9 Pins	14 Pins
6	Memory	4MB	32KB
7	Price	~$3 | ~$10NodeMCU)	~$22

On the other side, the system hardware comparison shows except the NodeMCU module problem both systems are working with the sensor data and can monitor data on webpage. The system hardware comparison is shown in TABLE (3).

TABLE 32: systems comparison

No	ITEM	Arduino-PI System	ESP-PI System
1	Connections Possible	32	infinite
2	Bi-Directional Communication	Yes	No
3	Communication Method	Serial	Wi-Fi
4	Database Logging	Fast	Moderate
5	Data Collision	<1%	>20%
6	Module Failure	Rare	Often
7	Cost	More	Less
8	Reprogramming	Easy	Difficult

V. CONCLUSION

In this research paper the two designs with the UART (communication (COM) ports) and Wi-Fi based on zonal idea or Multi DCS/SCADA and web-based GUI system is introduced. The main purpose of this research work along with security consideration in SCADA system and compare two systems is having the local host inside the Raspberry Pi as the smallest and powerful board. The both design system shown the successful implementations of zonal idea or Multi DCS/SCADA idea as the secure SCADA in agricultural application. In zonal idea or Multi DCS/SCADA based on its nature none of the unit can peek into another Zone and interact at all. Each level can have its own security measures unique to itself and everyone else. Using this idea, one can have the benefits like: security, fast accessing and monitoring in each stage. On the other hand, the idea of using Raspberry Pi as a micro-computer in this design for both systems is having system all system in the small and portable mode in agriculture arena. The survey on previous works shows that the researcher area more interested on PI and Arduino as the controller and the Wi-Fi specially in the agriculture or plant application branch is more demanding. The comparison between both designed systems shows, in spite of the ESP8266 based module NodeMCU is facilitated with Wi-Fi but the data handing during the system test have a few limitations like slow to program, and difficult access to have full control over the ESP modules programming. Besides, as it is independent of serial connectivity to do the work, the remotely programing is not possible then for every programming it should be connected to computer. Remote programming for the PI and using serial port over just Wi-Fi gives better response times and control Other than that the result of both designed systems show Raspberry Pi based on zonal idea or Multi DCS/SCADA can be applicable in various application and its Web based GUI can give the simple feature of remote data accessing over the internet in the safe manner without the hassle of installing extra software and drivers and other dependencies.

REFERENCES

1. i-SCOOP. (2019). HMI/SCADA software in the age of Industrial IoT and Industry 4.0. [online] Available at: https://www.i-scoop.eu/industry-4-0/hmi-SCADA-software/ [Accessed 3 Sep. 2019].
2. Instrumentation and Control Engineering. (2019). Components of SCADA: [online] Available at: https://automationforum.co/components-of-SCADA/ [Accessed 3 Sep. 2019].
3. M. Kaya, Y. Cetin-Kaya, and Y. Çetin-Kaya, “COMPLEX EVENT PROCESSING USING IOT DEVICES BASED ON ARDUINO,” Int. J. Cloud Comput. Serv. Archit., vol. 7, no. 6, pp. 13–24, 2017.
4. C. R. Rocha and M. Scholl, “Conceptual design of an open source SCADA system,” in Proceedings of the 25th International Conference on Flexible Automation and Intelligent Manufacturing-FAIM 2015, 2015, vol. 2, pp. 182–189.
5. Sarkar, Prosanjeet J., and Satyanarayana Chanagala. "A survey on iot based digital agriculture monitoring system and their impact on optimal utilization of resources." Journal of Electronics and Communication Engineering (IOSR-JECE) 11.1 (2016): 1-4
6. Ferrández-Pastor, Francisco, et al. "Developing ubiquitous sensor network platform using internet of things: Application in precision agriculture." Sensors 16.7 (2016): 1141
7. Gondchawar, Nikesh, and R. S. Kawitkar. "IoT based smart agriculture." International Journal of advanced research in Computer and Communication Engineering 5.6 (2016): 838-842
8. Krishna, K. L., Silver, O., Malende, W. F., & Anuradha, K. (2017). Internet of Things application for implementation of smart agriculture system. 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC). doi:10.1109/i-smac.2017.8058236
9. P. J. Sarkar and S. Chanagala, “IoT based Digital Agriculture Monitoring System & Their Impact on Optimal Utilization of Resources,” J. Res. Vol., vol. 4, no. 01, pp. 1–4, 2018
10. V. M. Igure, S. A. Laughter, and R. D. Williams, “Security issues in SCADA networks,” Comput. Secur., vol. 25, no. 7, pp. 498–506, 2006.
11. R. Chandia, J. Gonzalez, T. Kilpatrick, M. Papa, and S. Shenoi, “Security Strategies for SCADA Networks BT - Critical Infrastructure Protection,” vol. 253, pp. 117–131, 2008.
12. Keshtgary, Manijeh, and Amene Deljoo. "An efficient wireless sensor network for precision agriculture." Canadian Journal on Multimedia and Wireless Networks 3.1 (2012): 1-5.
13. “Arduino - Home”. [Online]. Available: https://www.arduino.cc/. [Accessed: 05-Sep.-2019].
14. “Teach, Learn, and Make with Raspberry Pi – Raspberry Pi”. [Online]. Available: https://www.raspberrypi.org/. [Accessed: 05-Sep.-2019].
15. “dev-esp32”. [Online]. Available: https://nodemcu.readthedocs.io/en/dev-esp32/. [Accessed: 05-Sep.-2019].
16. Bosch Sensortech, “Datasheet BMP180 Digital pressure sensor,” Bst-Bmp180-Ds000-09, no. April, p. 28, 2013.
17. P. Description, “Datasheet HC-SR501 PIR Motion Detector,” pp. 3–5, 2012.
18. T. Data, “MQ-2 Semiconductor Sensor for Combustible Gas,” Pololu, p. 2, 2016.
19. G. Description, O. Information, P. Configuration, and T. A. Circuit, “Cold-Junction-Compensated K-Thermocouple- to-Digital Converter (0°C to +1024°C),” Source, pp. 1–8.
20. P. Summary, “Datasheet SHT1x (SHT10, SHT11, SHT15),” 5, no. December, pp. 1–12, 2011.