The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM)
ISSN: 2602-3199

The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM), 2024

Volume 28, Pages 1-13

ICBASET 2024: International Conference on Basic Sciences, Engineering and Technology

Web Server Support of Multiple Controllers Data Transfer in IoT Network

Maja Kukusheva-Paneva
Goce Delcev University

Todor Chekerovski
Goce Delcev University

Sara Stefanova
Damilah, Skopje

Goce Stefanov
Goce Delcev University

Abstract: In this paper a solution with a method for enabling data transfer among three remote industrial plants is represented. In particular, it examines the connectivity of three remote industrial plants within IoT network. Each plant is equipped with a WIFI controller along with corresponding hardware components including sensors and actuators. The sensors monitor conditions and signals when change occurs, while an actuator receives a signal and performs an action. Data that is generated by these sensors and actuators is transmitted via the controllers and consolidated onto a common single channel hosted in a designated WEB server. The setup enables the data that is collected from two controllers, designated as MASTER 1 and MASTER 2 stations, to be consolidated and accessed through a single controller referred to as SLAVE station. Within the SLAVE station, real- time visualization of the data is enabled and also, the results can be presented on SCADA screen.

Keywords: Multi controller, Transfer data, WEB Server, IoT network

Introduction

To enhance production capacities, industrial companies must follow the modern trends of technological and technical achievements. This entails the incorporation of automation systems into their production process. For newly established production companies, this requirement is critical for ensuring their successful operation. However, a significant challenge arises in upgrading automated systems in existing companies where such systems are either absent or are operating at low level. Nevertheless, implementing automation systems of the production process in a company promises both increased production efficiency and improved performance of working machines. This improvement is reflected in terms of energy saving, as well as in timely and accurate monitoring of production device parameters, (Bennett, 1982; Hor, 2005; Automation Community, 2023).

In today's industrial companies, machinery and devices are typically located within a single geographical location. However, some devices there are working machines and devices that are located in one territorial location, but also devices that are spatially distant from the main plant and are an inseparable part of the production process. Often these remote plants are located far from intra and internet network of the production companies. Therefore, there arises a necessity to automate and connect these plants into the company's intranet and more widely into the Internet of Things (IoT) network.

Automation, control, management, and monitoring processes are of exceptional importance, perticulary in industrial use. Efforts are made to ensure more reliable and simpler work, especially for operators who directly oversee the industrial processes' functionality. This approach to work is made possible by the so-called SCADA (**S**upervisory **C**ontrol and **D**ata **A**cquisition) system, (Myomron, n.d.). Moreover, a modern industrial process control system of an industrial process is fully rounded if, in addition to SCADA, it is also connected to an IoT network. Such a concept enables process data to be transferred to any location, real-time visualizatio, and storege on a cloud-based platform.

In many instances, standalone industrial processes may function as separate entities, often located far from the intranet and Internet networks of the manufacturing companies. Therefore, data distribution from sensors and actuators, monitoring various parameters such as voltage, current, pressure, flow, temperature, LEDs, relay, etc., must be made from these remote entities to the master station via wireless communication, (Stefanov, 2021), most likely a radio frequency (RF) connection, (Hosseini, 2021; Stefanov, 2023).

In this paper, a prototype of an automated system that integrates appropriate hardware components from three different production processes located at different spatial locations is represented.

Design on Web Server Support Multiple Controller Data Transfer in IoT Network

The primary tool utilized in the solution enables data transfer to the three remote hardware components through Web Server. Each of the three hardware components is connected to the same channel on the Web Server. Figure 1 illustrates a block diagram of an implementation of a Web Server Support Multiple Controller in IoT network.

Web Server Support system enables data transfer between MASTER A, MASTER B and SLAVE stations. These three stations are installed in three geographically distant locations and communicate with each other through common channel on the Web Server. The solution uses Web Server Blynk cloud. In the designed prototype, switches and LEDs serve as sensors and actuators to verify the correct operation of the solution. In all three stations nodeMCU 8266-12E microcomputers units along with appropriate hardwer componets are installed, (ESP8266). Additionally, the microcontroller in MASTER A station is connected to an AtMega 328P microcontroller on nano Arduino board, (Microchip, n.d.). This microcontroller facilitates the connection to a SCADA screen supported by the CX-Supervisor SCADA system developed by Omron, (Myomron, n.d.). Furthermore, the MASTER A station incorporates a humidity and temperature sensor, implemented with DHT22 module, and is used to generate a digital signal. A 0-5 V voltage sensor signal is used to generate an analog signal, implemented through a potentiometer. LED D6 simulates a digital output signal in the MASTER A station.

In the MASTER B station NodeMCU 8266-12E is also used. Hardware components include a digital signal from the humidity and temperature sensor DHT22 and an analog signal from MQ7 sensor for measuring CO concentration. These signal values are displayed on both serial monitor and on an LCD display. In the SLAVE station, all signals from MASTER A and MASTER B stations are accumulated. The values of these signals are displayed on both a serial monitor and on an LCD display. To verify the proper connection between Blynk cloud SCADA and the SCADA on MASTER A, built-in LEDs are incorporated on outputs D3 and D5 in the SLAVE station. The LED on output D3 verifies the connection with a switch from the SCADA screen of the MASTER A station, while the D5 LED verifies the connection with the Blink switch from the Blynk cloud SCADA and the IoT smart device.

The Blynk Cloud Web server, as previously mentioned, serves as the central hub for the entire system. Figure 1 illustrates the connection of SCADA Blynk Cloud and an IoT smart device to the Web server. The SCADA Blynk Cloud allows for real-time visualization of the measured values from the three measuring points on its screen, as well as logging of these values, (Dashboard). Additionally, the IoT smart device enables access and management of the system from any location with an IoT network signal using mobile device.

The final conclusion drawn from the New Server support system is that the measured values from the three measuring points at different locations are effectively collected, visualized and stored on two SCADA systems: SCADA screen on CX-Supervisor and SCADA Blynk cloud. Furthermore, the system enables control over its signal from any measuring point, (Santos, 2019; Lightbluecrab, 2019).

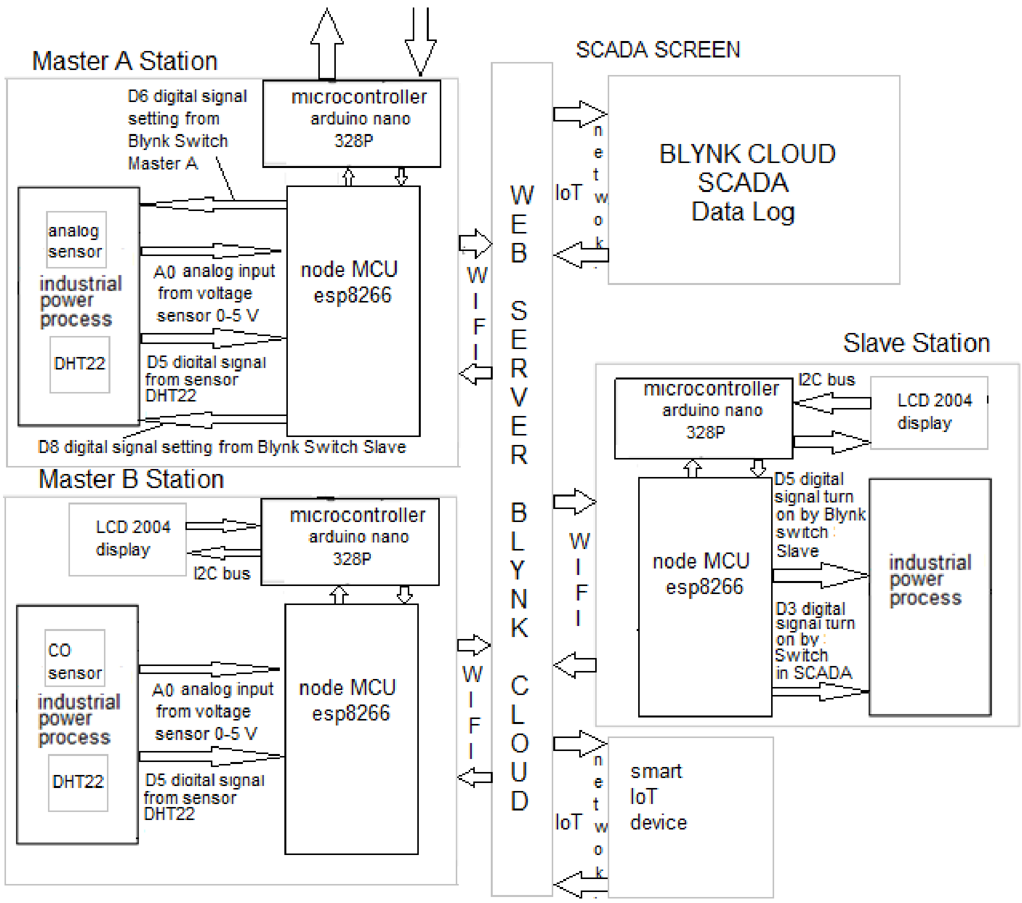


Figure 1. Block diagram of web server support of multiple controller in IoT network

The prototype system design consists of design of the MASTER A, MASTER B, SLAVE stations along with the design of the SCADA CX-Supervisor as and SCADA Blynk cloud. The design of the MASTER A station includes a microcomputer NodeMCU ESP8266-12E, ATmega 328P on an Arduino Nano board, a DHT22 sensor, an analog signal sensor (in this case, a signal from a potentiometer) and SCADA CX Supervisor, along with appropriate hardware components:

Hardwar components connected on Node MCU 8266-12E:

- *potentiometer as sensors on analog signal 0-5 V, connected on analog input A0.*
- *D6 digital output signal controlling from Blynk switch in Blynk SCADA cloud.*
- *LEDs as actuators connected on D6 digital output.*
- *DHT22 temperature and humidity sensor connected on digital pin D5.*

Hardwar components connected on Arduino nano board:

- *potentiometer as analog signal sensors 0-5 V, connected on analog input A0.*
- *DHT22 temperature and humidity sensor connected on digital pin D3.*
- *SCADA CX-Supervisor connected on UART pin on Arduino nano.*

The design of the MASTER B station includes a microcomputer Node MCU ESP8266-12E, DHT22 sensor, an analog signal sensor (in this case a signal from a MQ7 module):

Hardwar components connected on Node MCU 8266-12E:

- *analog signal sensor 0-5 V from MQ 7 module, connected on analog input A1*
- *DHT22 temperature and humidity sensor connected on digital pin D4*
- *LCD 2004 display connected by I2C bus on digital input D1 and D2 on node MCU 8266-12E*

The design of the SLAVE station consists of a microcomputer Node MCU ESP8266-12E, (Hosseini, 2021) and appropriate hardware components:

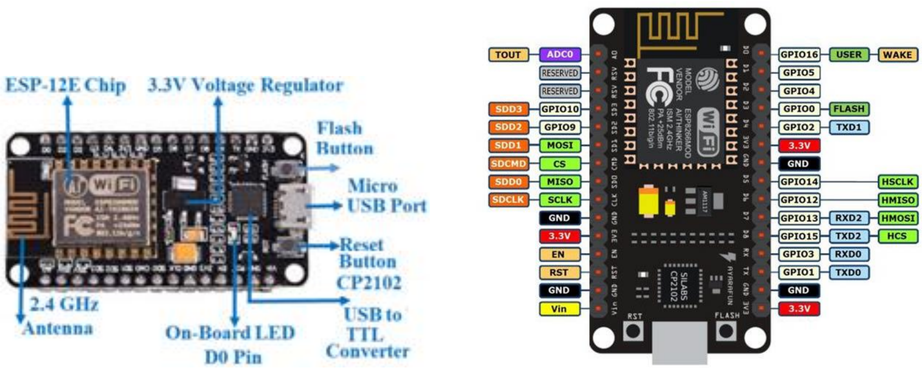
Hardwar components connected on node MCU 8266-12E:

- *D3 digital output signal controlling from SCADA screen switch*
- *LEDs as actuators connected on digital output D3*
- *D6 digital output signal controlling from Blynk switch in Blynk SCADA cloud*
- *LEDs as actuators connected on digital output D6*
- *LCD 2004 display connected by I2C bus on digital input D1 and D2 on node MCU 8266-12E*

2.1 Features of the used Hardware

a.) Microcomputer NodeMCU ESP8266-12e

The NodeMCU ESP8266 development board is equipped with the ESP-12E module which contains ESP8266 chip featuring Tensilica Xtensa 32-bit LX106 RISC microprocessor, (ESP8266). This microprocessor supports RTOS and operates at an adjustable clocj frequenct of 80MHz to 160 MHz. The NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power, along with in-built WiFi/ Bluetooth and Deep Sleep Operating features make it ideal for IoT projects. NodeMCU can be powered using Micro USB jack and VIN pin (External Supply Pin) and supports UART, SPI, and I2C interface. In Figure 2 NodeMCU ESP8266 and its pinout are shown.



a.) b.)

Figure 2. a.) NodeMCU ESP8266 and b.) pinout

NodeMCU is an open- source firmware and development board specially designed for IoT applications. It incorporates firmware that operates on the ESP8266 WiFi SoC from Espress Systems, along with hardware based on ESP-12 module.

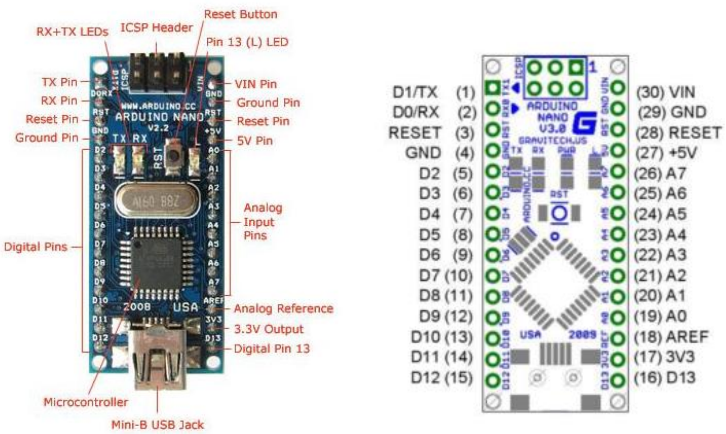
NodeMCU ESP8266 specifications and features include:

- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB
- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

The NodeMCU ESP8266 board can be easily programmed with Arduino IDE since it is easy to use.

b.) Microcomputer ATmega 328P

The Arduino Nano is an open- source microcontroller board based on the Microchip ATmega328P microcomputer developed by Arduino. The board is equipped with sets of digital and analog input/output (I/O) pins that can interface with various expansion boards (shields) and other circuits. The board features 14 digital I/O pins (six of which are capable of PWM output), along with 6 analog I/O pins and are programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered either by the USB cable or an external 9-volt battery, though it accepts voltages between 7 and 20 volts. The Nano board represents the first in a series of USB-based Arduino boards; Initially, it and version 1.0 of the Arduino IDE were the reference versions of Arduino, which have since evolved in newer releases. The ATmega328P on the board comes preprogrammed with a bootloader that allows uploading of new code without the need for an external hardware programmer, (ESP8266). In Figure 3a is shown electronic board of the Arduino Nano with build Atmega 328P microcomputer, while Figure 3b illustrates its pinouts, (ATMEGA328P).



a.) b.)

Figure 3. a.) Arduino Nano and b.) its pinout

c.) DHT11 Temperature and HUMIDITY SENSOR

The DHT22 sensor is a widely utilized temperature and humidity sensors. The sensor comes with a dedicated NTC for temperature measurment and an 8-bit microcontroller for outputing temperature and humidity values as serial data, (DHT11).The connection diagram for this sensor is shown in Figure 4.

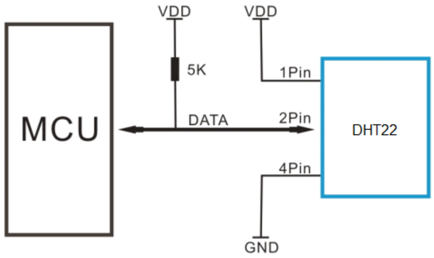


Figure 4. *Connection diagram for DHT11 sensor*

The sensor is factory calibrated simplifying its interface with other microcontrollers. It can measure temperature ranging from 0°C to 50°C and humidity from 20% to 90% with an accuracy of ±1°C and ±1%. From Figure 4 can be seen that the data pin is connected to an I/O pin of the MCU and a 5K pull-up resistor is used. The data pin outputs both temperature and humidity values as serial data. For interfacing the DHT22 with Arduino there are ready-made available libraries for quick setup and start. The output from data pin is sent as 8-bit humidity integer data , 8-bit humidity decimal data, 8- bit temperature integer data, 8-bit fractional temperature data and 8- bit parity bit. The duration of each host signal is detailly explained in DHT11 datasheet with and illustrative timing diagrams. This sensor can be used in applications for temperature and humidity measurement, local weather stations, automatic climate control and environment monitoring. In Figure 5 is shown actual size DHT22 sensor along with its pinout.

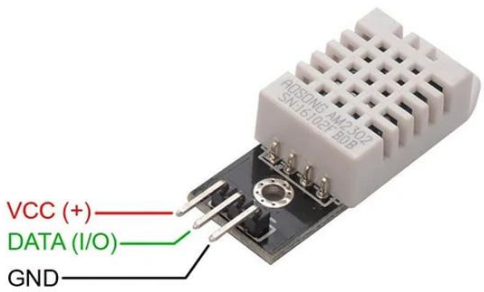


Figure 5. DHT11 sensor and pinout

DHT22 Specifications:

- Operating Voltage: 3.5V to 5.5V
- Operating current: 0.3mA (measuring) 60µA (standby)
- Output: Serial data
- Temperature Range: 0°C to 100°C
- Humidity Range: 20% to 100%
- Resolution: Temperature and Humidity both are 16-bit
- Accuracy: ±1°C and ±1%

The DHT22 sensor is available for purchase either as a standalone sensor or as a module. In both cases, the performance of the sensor is same. The sensor typically comes as a 4-pin package, althoug only three pins are utilized. The only difference between the sensor and module is that the module’s filtering capacitor and inbuilt pull-up resistor. For standalone sensor, these components are externally utilized if needed.

d.) MQ-7 CO Gas Sensor

In Figure 6 is shown MQ-7 gas sensor. TheMQ-7 gas sensor utilizes SnO_2 as a gas- sensing material, which has a lower conductivity in the clear air. However, in the atmosphere containing carbon monoxide at a certain concentration, the conductivity of the gas sensor increases as the concentration of carbon monoxide increases, (MQ-9B Co.)

Figure 6. MQ-7 gas sensor

This module can be applied to household and industrial gas leakage alarm, portable gas detecting device, etc. Detecting range is 0 ppm-2000 ppm (ppm is concentration in million pieces) carbon monoxide. In Table 1 is given pins connection on MQ-7 sensor.

Table 2. Pins connection on MQ-7 sensor.

Pin No.	Symbol	Description
1	DOUT	Digital out
2	AOUT	Analog out
3	GND	Ground
4	Vcc	+Power supply (2.5-5)V

2.1 Features of the used Software

a.) *Brief description of SCADA CX-Supervisor*

The SCADA system was created using Omron CX-Supervisor software, (Myomron, n.d.). CX-Supervisor is designed for development and operation of PC- based visualization and machine control systems. It is not only user- friendly for small supervisory and control tasks, but it also provides robust capabilities for designing highly sophisticated applications. CX-Supervisor offers powerful functions for a wide range of PC- based HMI requirements. Simple applicationscan be created rapidly with the aid of many predefined functions and libraries, simple applications can be quickly developed, while even complex applications can be created using powerful programming language or VBScript. CX-Supervisor runs on standard PC desktop computers running Microsoft Windows. CX-Supervisor is intuitive and easy to use, facilitates rapid configuration, testing and debugging of projects enabling developers to efficiently create and deploy applications. CX-Supervisor consists of two separate executable Windows programs: CX-Supervisor Development environment and CX-Supervisor Runtime environment. Applications are created and tested using the development environment and then delivered as a final customer application with the runtime environment. The runtime-only environment may only be used for executing an application previously generated using the development environment. It cannot be used to generate a new runtime application using the runtime environment. Figure 7 illustrates the connection between sensor hardware, microcontroller and CX-Supervisor SCADA.

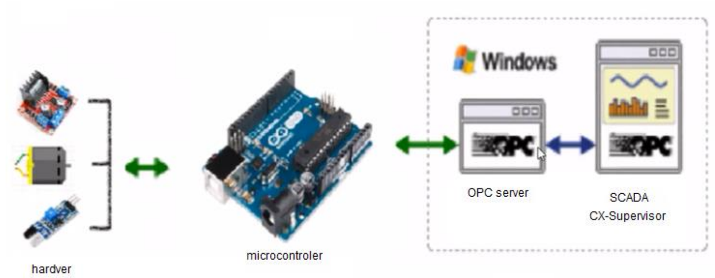
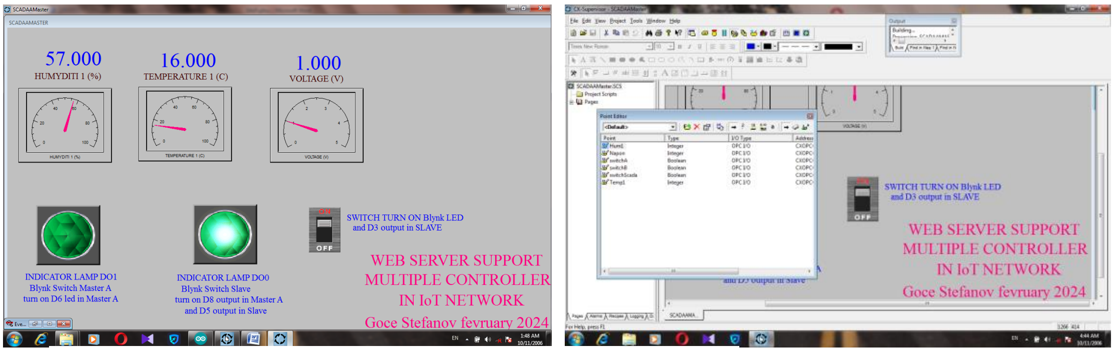


Figure 7. Connection between sensor hardware, microcontroller and CX-Supervisor SCADA

Setting up Graphic Symbols

The initial step is to set up graphic symbols. Once the project has been created with its own page, the graphic objects can be constructed and added to the page, as shown in Figure 8a.



a.) b.)

Figure 8. a.) Setting graphic symbols, b.) Setting variables for graphical symbols

The graphics editor uses a Graphic Object toolbar and a floating window called the Palette to construct and control objects on the page. These tools are very easy to use. The Graphic Object Toolbar Several feauters several small icons each representing a graphical object that can be used to construct the application. Some of the objects are graphical primitives - straight lines, ellipses, rectangles, while others are more advanced, such as the gauge object, which has built-in functionality.

Setting up Variables for Graphical Symbols

For each of the graphic symbols, a point variable with the appropriate size and unit is created, as shown in Figure 8b. These variables correspond to the variables in the Arduino code.

Setting Grafical Symbols with Variables

Finally, each graphic symbol is associated with a corresponding variable. In Figure 9 is shown SCADA screen with defines variables.

Figure 9. SCADA data system screen with defines variables

In solution represented in this paper for the MASTER A station, the humidity is displayed digitally using a display and analogically with an instrument. The same applies to temperature measurement at this point. Additionally, two lamps are installed to control and monitor the operation of the switches in the Blynk SCADA cloud. These lamps, labeled as Lamp DO1 and Lamp DO0, respectively, control the Blynk switches for MASTER A and SLAVE station. Lamp DO0, on the other hand, controls the Blynk switch for SLAVE, activating outputs D8 in MASTER A and D5 in SLAVE.

b.) *Brief Description of Blynk SCADA Cloud*

Blynk is a comprehensive software suite that facilitates the prototyping, deployment, and remote management of connected electronic devices at any scale.

Whether it's personal IoT projects or commercial connected products in the millions, Blynk empowers users to connect their hardware to the cloud and create iOS, Android, and web applications. With Blynk, users can analyze real-time and historical data from devices, remotely control them from anywhere, receive important notifications, and much more.

Blynk.Console is a feature-rich web application catering to different types on user. Its key functionalities include:

-Configuration of connected devices on the platform, including application settings.
-Device, data, user, organization, and location management.
-Remote monitoring and control of devices shown on Figure 10.

Blynk.Apps

Figure 10. Blynk apps device

Applications created with Blynk are designed for end-users, whether they are family members, employees, or product purchasers. User can easily download the application, connect their devices, and begin using them.

Blynk Edgent

Blynk Edgent is a packaged solution designed to simplify the connection of supported devices to the Blynk platform. It offers access to all its advanced features without requiring extensive coding. Some of the key features of Blynk Edgent are:

-Device claiming and Wi-Fi provisioning (bringing device online and authenticating them with a certain user).
-Data transfer between device and the cloud.
-API integration with Blynk.Apps and Blynk SCADA Cloud features.
-Over-the-air firmware updates for select hardware models.

Blynk Cloud

The Blynk Cloud is a server infrastructure for Blynk SCADA IoT platform binding all its components together. The compatible boards that operate with the Blynk cloud are given in Table 2.

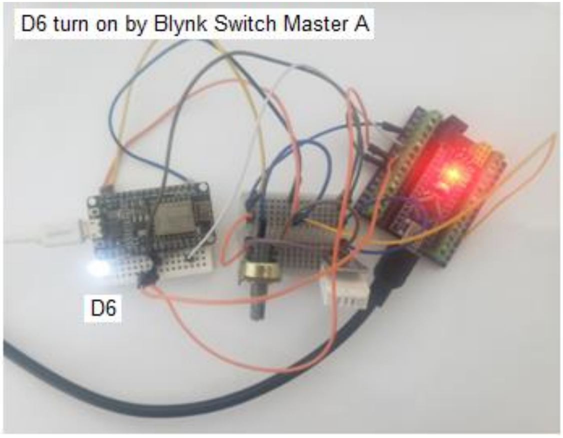
Table 2. Hardware components for Blynk cloud

Board	Provisioning	Secure connection	Blynk.Air
ESP32	✓	✓	✓
ESP8266	✓	✓	✓
Seeed Wio Terminal	✓	✓	✓
TI CC3220	✓	✓	✓

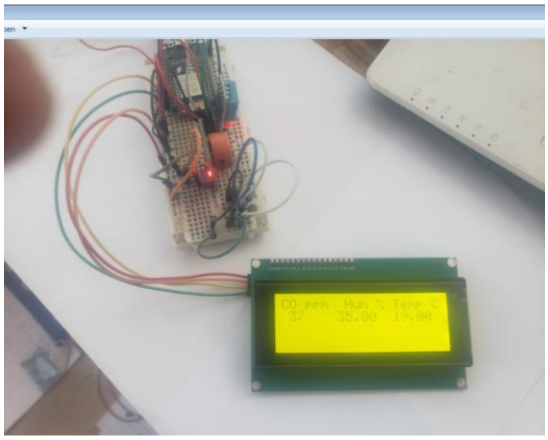
Hardware that works wit...
Arduino connection types
Made by Community
Troubleshooting
Need support for any ot...

3. Experimental Results

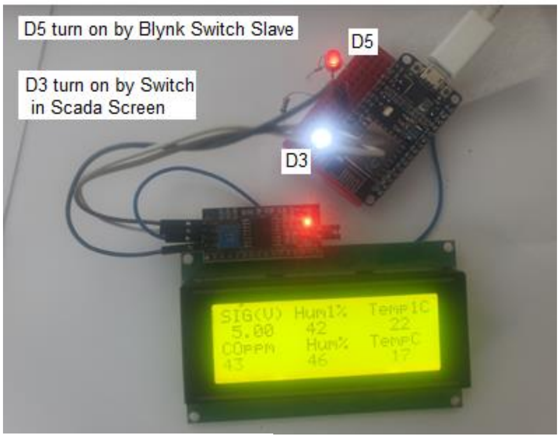
The designed prototype consists of MASTER A station (Figure 11.a), MASTER B station (Figure 11.b) and SLAVE station (Figure 11.c).



a.)



b.)



c.)

Figure 11. Prototype of designed Web Server Support data system: a) prototype on the MASTER A station, prototype on the MASTER B station and c) prototype on the SLAVE station.

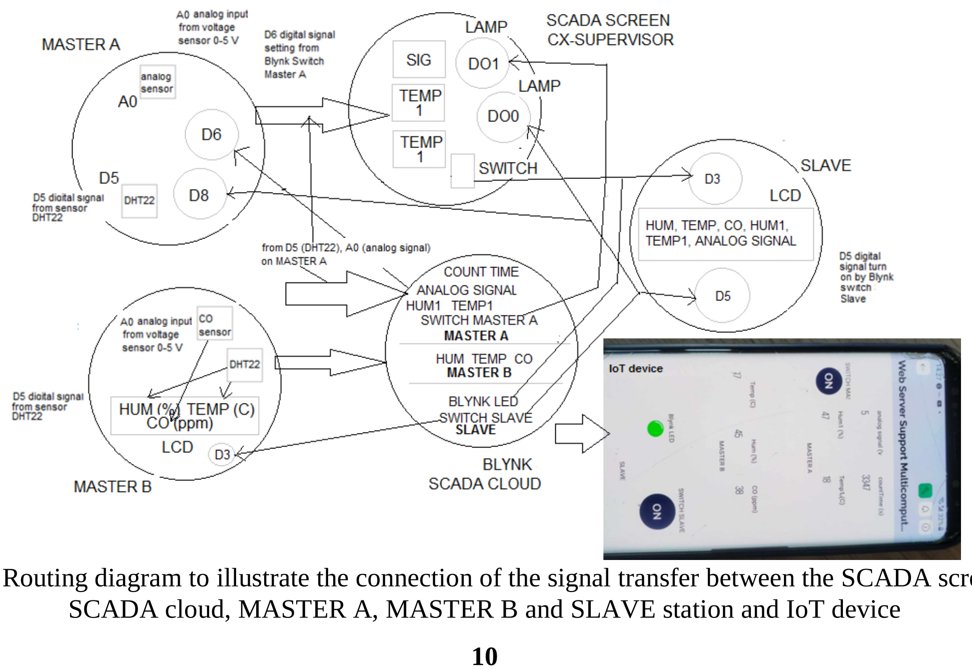
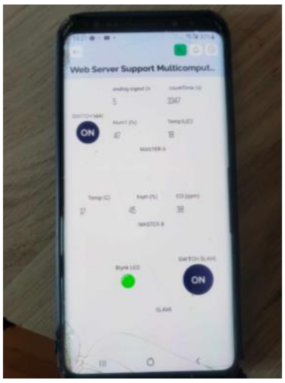
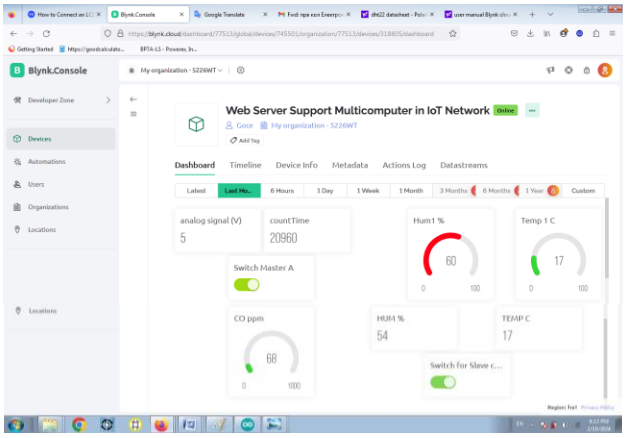


Figure 12. Routing diagram to illustrate the connection of the signal transfer between the SCADA screen, Blynk SCADA cloud, MASTER A, MASTER B and SLAVE station and IoT device

For a clearer overview and understanding of the signal flow within the system, Figure 12 shows the signal routing diagram between the blocks from which the designed solution is built. In Figure 13a, a data screen is displayed on a mobile device, showcasing data transferred through Web server data system in the IoT network, while in Figure 13b is shown a screen on the Blynk SCADA cloud.

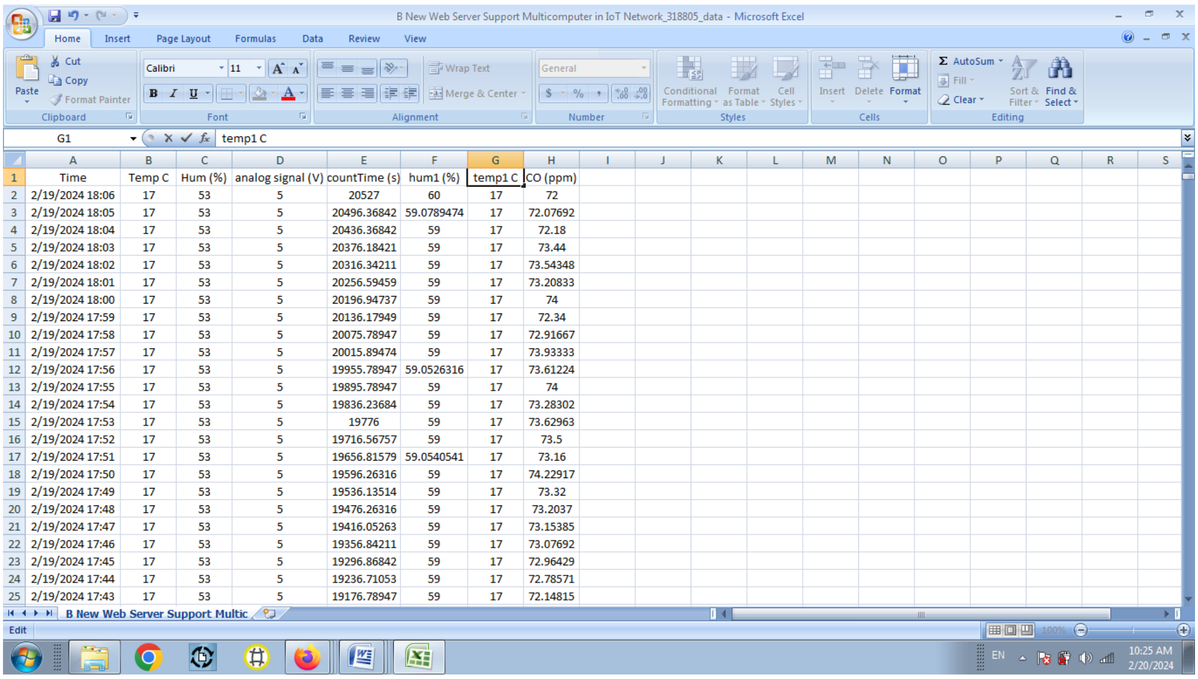


a.)



b.)

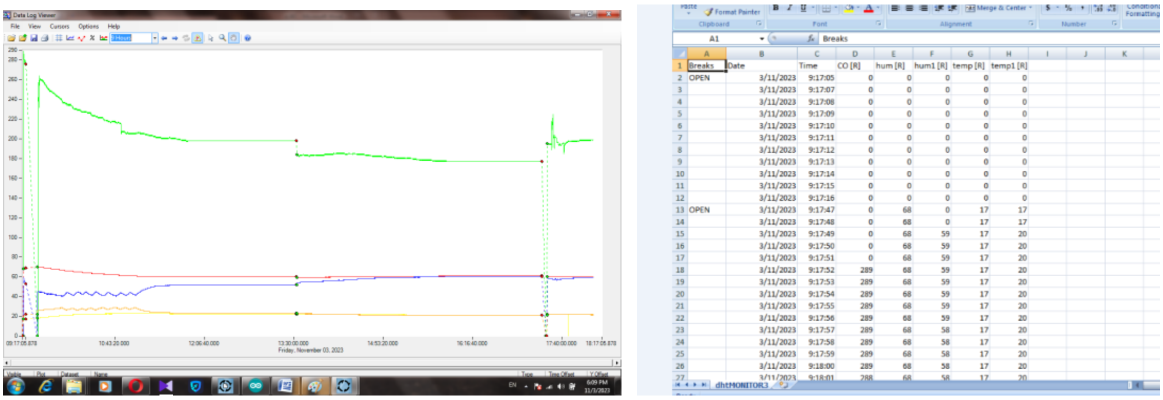
Figure 12. The screen on the transfer Web server data system in the IoT network: a) screen on mobile device, b) screen on Blynk SCADA cloud



	A	B	C	D	E	F	G	H
1	Time	Temp C	Hum (%)	analog signal (V)	countTime (s)	hum1 (%)	temp1 C	CO (ppm)
2	2/19/2024 18:06	17	53	5	20527	60	17	72
3	2/19/2024 18:05	17	53	5	20496.36842	59.0789474	17	72.07692
4	2/19/2024 18:04	17	53	5	20436.36842	59	17	72.18
5	2/19/2024 18:03	17	53	5	20376.18421	59	17	73.44
6	2/19/2024 18:02	17	53	5	20316.34211	59	17	73.54348
7	2/19/2024 18:01	17	53	5	20256.59459	59	17	73.20833
8	2/19/2024 18:00	17	53	5	20196.94737	59	17	74
9	2/19/2024 17:59	17	53	5	20136.17949	59	17	72.34
10	2/19/2024 17:58	17	53	5	20075.78947	59	17	72.91667
11	2/19/2024 17:57	17	53	5	20015.89474	59	17	73.93333
12	2/19/2024 17:56	17	53	5	19955.78947	59.0526316	17	73.61224
13	2/19/2024 17:55	17	53	5	19895.78947	59	17	74
14	2/19/2024 17:54	17	53	5	19836.23684	59	17	73.28302
15	2/19/2024 17:53	17	53	5	19776	59	17	73.62963
16	2/19/2024 17:52	17	53	5	19716.56757	59	17	73.5
17	2/19/2024 17:51	17	53	5	19656.81579	59.0540541	17	73.16
18	2/19/2024 17:50	17	53	5	19596.26316	59	17	74.22917
19	2/19/2024 17:49	17	53	5	19536.13514	59	17	73.32
20	2/19/2024 17:48	17	53	5	19476.26316	59	17	73.2037
21	2/19/2024 17:47	17	53	5	19416.05263	59	17	73.15385
22	2/19/2024 17:46	17	53	5	19356.84211	59	17	73.07692
23	2/19/2024 17:45	17	53	5	19296.86842	59	17	72.96429
24	2/19/2024 17:44	17	53	5	19236.71053	59	17	72.78571
25	2/19/2024 17:43	17	53	5	19176.78947	59	17	72.14815

Figure 13. Excel® data log file

The Blynk SCADA cloud provides the option to create a data log file for measurement data. This function generates a file compatible with Excel®. In the Figure 13 is an example of Excel® data log file. In addition to integrating Blynk SCADA into the IoT network within the designed Web server system, another option for monitoring measurement data in real- time and generating a data log file is available. This was achieved by installing the SCADA CX-Supervisor on the MASTER A station. In Figure 14a is shown real- time waveform for variables CO, Humidity, Temperature, Humidity1 and Temperature1, while in Figures 14b is shown data log file for same variables.



	Breaks	Date	Time	CO [R]	hum [R]	hum1 [R]	temp [R]	temp1 [R]
2	OPEN	3/11/2023	9:17:05	0	0	0	0	0
3		3/11/2023	9:17:07	0	0	0	0	0
4		3/11/2023	9:17:08	0	0	0	0	0
5		3/11/2023	9:17:09	0	0	0	0	0
6		3/11/2023	9:17:10	0	0	0	0	0
7		3/11/2023	9:17:11	0	0	0	0	0
8		3/11/2023	9:17:12	0	0	0	0	0
9		3/11/2023	9:17:13	0	0	0	0	0
10		3/11/2023	9:17:14	0	0	0	0	0
11		3/11/2023	9:17:15	0	0	0	0	0
12		3/11/2023	9:17:16	0	0	0	0	0
13	OPEN	3/11/2023	9:17:47	0	68	0	17	17
14		3/11/2023	9:17:48	0	68	0	17	17
15		3/11/2023	9:17:49	0	68	59	17	20
16		3/11/2023	9:17:50	0	68	59	17	20
17		3/11/2023	9:17:51	0	68	59	17	20
18		3/11/2023	9:17:52	289	68	59	17	20
19		3/11/2023	9:17:53	289	68	59	17	20
20		3/11/2023	9:17:54	289	68	59	17	20
21		3/11/2023	9:17:55	289	68	59	17	20
22		3/11/2023	9:17:56	289	68	59	17	20
23		3/11/2023	9:17:57	289	68	58	17	20
24		3/11/2023	9:17:58	289	68	58	17	20
25		3/11/2023	9:17:59	289	68	58	17	20
26		3/11/2023	9:18:00	289	68	58	17	20
27		3/11/2023	9:18:01	288	68	58	17	20

Figure 14. a.) Real- time waveform for variables CO, Humidity, Temperature, Humidity1, Temperature1, and in the Figures b.) Data log file for same variables

Conclusion

The paper presents the design and experimental implementation of a prototype supporting multi- controller data transfer via a Web server. The solution provides the option of communication among hardware components connected to process variables located at three remote locations. The medium through which the communication is carried is a Web server installed in the IoT network. All three microcontrollers are connected to the same channel in the Web server. Microcontrollers distribute the measurement data to the Web server. Consequently, the data transfer between the three remote industrial processes is achieved. Each measuring point's data can be locally visualized on a serial monitor or on the built-in LCD displays. In one of the measuring points, the MASTER A station, a SCADA system is installed. Distributed data in Web server is displayed in SCADA cloud and IoT mobile device. The built-in SCADA systems, SCADA CD-supervisor and SCADA cloud enable real-time visualization of measurement data as well as a data logging function.

Scientific Ethics Declaration

The authors declare that the scientific ethical and legal responsibility of this article published in EPSTEM journal belongs to the authors.

Acknowledgements or Notes

* This article was presented as an oral presentation at the International Conference on Basic Sciences, Engineering and Technology (www.icbaset.net) held in Alanya/Turkey on May 02-05, 2024.

* The Authors gratefully acknowledge support by Goce Delcev University Stip, North Macedonia through individual scientific fund.

References

Adafruit. (2024, May 8). *DHT11, dht22 and AM2302 sensors*. Retrieved from https://cdn-learn.adafruit.com/downloads/pdf/dht.pdf

Automation Community. (2023, December 30). *Difference between plc and SCADA*. Retrieved from https://automationcommunity.com/difference-between-plc-and-scada/

Bennett, S., & Linkens, S. (1982). *Computer control of industrial processes*, D.A. (Eds.), IEEE.

Espressif. (2020). ESP8266 technical reference. https://www.espressif.com/sites/default/files/documentation/esp8266-technical_reference_en.pdf

Hor, C. H., & Crossley, P.A. (2005). *Knowledge extraction from intelligent electronic devices, lecture notes in computer science* (pp. 82-111). United Kingdom, Loughborough University.

Hosseini, W. (2021, November 15). *NRF24L01 interfacing with Arduino: Wireless communication. electropeak*. Retrieved from https://electropeak.com/learn/wireless-communication-w-arduino-and-nrf24l01/

José, S. R., Arbour, Y., McPhalen, J., Briggs, M., Griswold, S., …& Kiran, S. (2019, April 2). *Sending data from an Arduino to the ESP8266 via serial. random nerd tutorials.* Retrieved from https://randomnerdtutorials.com/sending-data-from-an-arduino-to-the-esp8266-via-serial/

Lightbluecrab, & Instructables. (2019, March 22). *Blynk with ESP8266*. Retrieved from https://www.instructables.com

Microchip.(n.d.). *ATMEGA328P*. Retrieved from https://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf

Myomron. (n.d.). *CX supervisior getting started*. Retrieved from https://www.myomron.com/downloads/1.Manuals/Software/CX-Supervisor/

Stefanov, G., Kukuseva - Paneva, M. & Stefanova, S. (2023). 3-phase smart power meter implemented in RF network. *Balkan Journal of Applied Mathematics and Informatics*, *6*(1), 25-37.

Stefanov, G., Kukuseva- Paneva, M., & Stefanova, E. (2021) Design of an intelligent Wi-Fi sensor network. *Balkan Journal of Applied Mathematics and Informatics*, *4*(1), 17-26.

Winsen. (n.d.).*MQ-9B Co & CH4 detection sensor*. Retrieved from https://www.winsen-sensor.com/sensors/co-sensor/mq-9b.html

Author Information

Maja Kukusheva Paneva
Faculty of Electrical Engenering
Goce Delcev University Stip
Krste Misirkov, 10A, 2000 Stip
North Macedonia
Contact e-mail: *maja.kukuseva@ugd.edu.mk*

Todor Chekerovski
Faculty of Electrical Engenering
Goce Delcev University Stip
Krste Misirkov, 10A, 2000 Stip
North Macedonia

Sara Stefanova
Faculty of Mechanical Engineering - Goce Delcev University, Damilah, Skopje
North Macedonia

Goce Stefanov
Faculty of Electrical Engenering
Goce Delcev University Stip
Krste Misirkov, 10A, 2000 Stip
North Macedonia

To cite this article:

Kukusheva -Paneva, M., Chekerovski, T., Stefanova, S., & Stefanov, G. (2024). Web server support of multiple controllers data transfer in IoT network. *The Eurasia Proceedings of Science, Technology, Engineering & Mathematics (EPSTEM), 28*, 1-13.