A Scheme for efficient and equitable use of public utilities through supervisory and distributed control

Shreenivasa K.*, M. S. Bhat† and Rekha S.‡
Department of Electronics and Communication Engineering,
National Institute of Technology Karnataka, Surathkal, India - 575025
Email: *shreenivasakonaje@gmail.com, †msbhat@ieee.org, ‡rsbhat_99@yahoo.com

***Abstract*—Clean water and electricity are the two major public utilities for any society and it is the responsibility of each consumer to use these resources efficiently and minimize the wastage. This paper presents a distributed control scheme to address these issues for each utility. In this scheme, wireless sensor network is used for on-line monitoring and automated control of water in overhead tanks. In a similar way, using energy meters and intelligent circuit breakers, the power consumption is remotely monitored and power delivery is automated. A cloud based user application is developed through which authorized operators can view the complete data of water flow as well as energy consumption at desired locations on a single graphical user interface (GUI). The distributed control is developed using Internet enabled embedded boards along with sensors and supporting network nodes. IBM's Watson IoT platform is used for data acquisition, analysis and control.**

***Index Terms*- Data acquisition through sensors, energy meter, IoT platform, utility management through cloud.**

I. Introduction

Clean water is the basic need of all living beings and electricity is a very important resource for any modern society. It is the responsibility of every person to use them judiciously and conserve the precious resources to the next generation. Thus there is a need to device a scheme to monitor and control the resources for equitable use and efficient distribution with minimal wastage.

Today, in most cities and campuses in India, water supply system is old fashioned, complex and very inefficient. The poor distribution system is mostly due to an absolute lack of sensitivity by both the state and the consumer towards conserving the precious resource. Most urban areas depend on several sources of water as there is no integrated approach in place. There is also a lack of understanding on the part of the state and the consumer about how much water is required for a household per day. This, coupled with inefficient distribution network has resulted in inequitable supply among the consumers. This situation can lead to many social problems and a discussion of that is beyond the scope of this paper. Thus, undoubtedly there is an urgent need to, both sensitize the people and also to upgrade the system without disturbing the present pipeline network and routine operations. To alleviate the problem, a suitable and cost effective solution need to be developed. This paper presents an architecture for equitable water distribution in large campuses and urban areas without requiring the replacement of the distribution network. Similarly, a parallel can be drawn about the state owned electricity distribution system, which is both inefficient and inequitable. Figure 1 shows a simplified scheme of the proposed service oriented architecture [1].

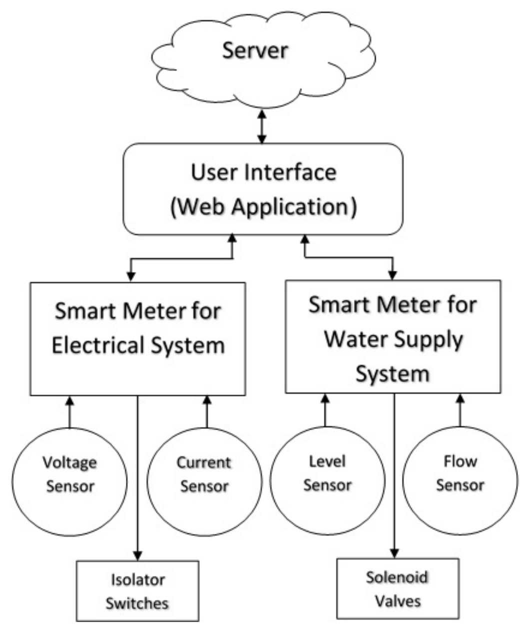


Fig. 1. Architecture of Distributed Control System

The scheme described in this paper looks at the development of a overall control system including software and hardware. The system is expected to perform the following functions [2].

1) Real-time acquisition of water and electricity parameters.
2) Visualization of actual water flow, water level and energy consumption.
3) Analysis /evaluation of consumption of these utilities at different places.
4) Providing alternatives for manual control of utilities.
5) Reduction in water and electricity wastage.

The architecture shown in figure 1 uses embedded boards, servers and internet connectivity. Further, the hardware required to implement the scheme includes,

978-1-5386-5323-4/18/$31.00 ©2018 IEEE

1) smart meters, voltage and current sensors, relays etc. for electricity and
2) flow sensors, level sensors, solenoid valves, motors etc for water

The scheme involves the installation of appropriate sensors, control valves/relays and embedded (controller) boards at appropriate places in the electricity and water distribution network. All sensors and actuators are connected to controller boards placed at appropriate places in the network. The communication between different controller boards and the centralized server housing the utility database (such as MySQL database) is established over the internet. Data acquisition is done at regular intervals of time and is programmed in the controller. Data obtained from various controllers is uploaded to the server. Server stores multiple parameters belonging to specific region/building. The parameters include power consumption, average and peak demands, water flow rate, water quantity, valve actuation timings etc. This data is then utilized for evaluation and analysis purposes. The overall system for water and energy monitoring can either made to operate in automated mode or the operator can intervene if required to manually control the selected utility/distribution as per the requirement [3].

II. Methodology

ESP8266 microchip controller is used to gather sensor information and to control actuators. In the water control system, flow and ultrasonic level sensors are interfaced to the controller to get water flow and water level information, respectively. A solenoid valve is used to control the water flow into water tank. In the electrical energy control system, voltage sensor and current sensors are connected to smart energy meter to gather energy consumption information. Electrical isolator switches are used to cut-off or connect electrical supply to loads. Wi-Fi communication technology is adopted to communicate between controller and the local server. Server and web applications are developed with the help of programming languages like PHP, HTML, Java and MySQL. ESP8266 is programmed using Arduino IDE. All hardware modules are powered by single electric board. As per cloud IoT paradigm [4], back-up servers are maintained within local area network and also in cloud such as IBM's Watson IoT platform service (in this case). Following parameters are considered while implementing both the control systems. They include,

1) Accurate Measurement of Voltage, Current, Water Flow & Level in tanks.
2) Smart Control: Switching of pumps, isolator switches, water Flow direction & level control.
3) Security: Allow operation by authorized persons only.
4) Automatic, Semi-automatic & Manual control: Programmed to operate automatically as well as through human intervention (App based).
5) Instant Notification: Status updates of water & electric power data and warning messages to authorized persons.
6) Un-interrupted communication: Seamless data packet transmission, 24×7 connection to server [5].

III. Design Methodology

Design and implementation of this strategy is divided into several tasks comprising hardware and software blocks including multiple layers [1]. Each block consists of a couple of logic controllers. One controller performs acquisition of energy consumption data and performs necessary load shedding when required, while another controller reads water data and controls the water flow. Separate controllers are connected to respective sensors and actuators.

Block diagram of Smart energy meter is shown in fig.2. Each block has an LEM voltage Hall effect transducer (LV-25P) and an LEM current Hall effect transducer (LA-25P) connected to logic controller (Arduino Ethernet Shield). These transducers also provide isolation between the power network and the signal level circuits. They convert single phase power quantities (voltages and currents) to low-level voltage signals of around 2.5 V range, which are generally proportional to the quantity being measured. Hall effect voltage transducer has conversion ratio (CR_v) of 2500:1000 [6]. From the measured data, the output current can be calculated as,

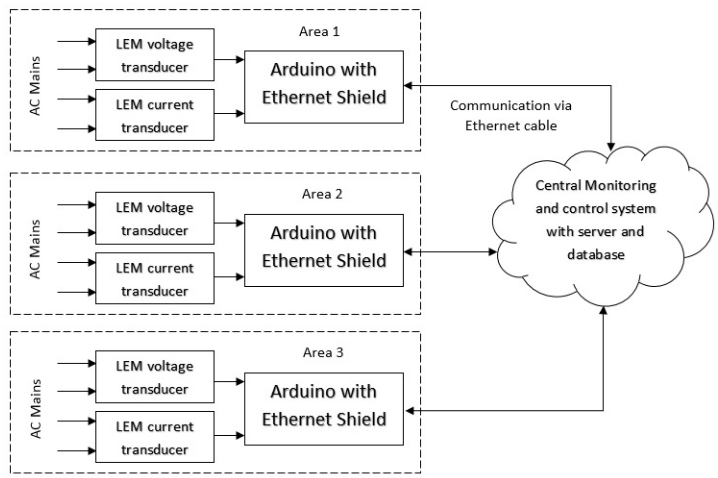


Fig. 2. Block diagram of typical smart meter for electricity supply system

$$I_o = (I_i * CR_v) \tag{1}$$

$$I_o = (2.3\ mA * 2:5) = 5.75\ mA \tag{2}$$

The output across the measuring resistance needs to be chosen in such a way that the output voltage is below 2.5 V corresponding to an input voltage of 230 V. As given in the data sheet, measuring resistance (R_m) can be chosen in the range of 30-190 Ω. In the present module, measuring resistance (R_m) is chosen to be 100 Ω. Thus the output voltage (rms) can be calculated as follows.

$$V_{rms} = (I_o * R_m) \tag{3}$$

$$V_{rms} = (5.75\ mA * 100\ \Omega) = 575\ mV \tag{4}$$

So, the peak to peak voltage (V_{pp}) can be calculated as,

$$V_{pp} = (V_{rms} * 2\sqrt{2}) \tag{5}$$

$$V_{pp} = (5.75\ mV * 2\sqrt{2}) = 1.62\ V \tag{6}$$

The conversion ratio (CR_i) for current transducer is 1:1000 [6]. Primary current can be calculated as,

$$I_i = (I_o * CR_i) \tag{7}$$

$$I_i = (25\ mA * 1000) = 25\ A \tag{8}$$

The low-level bipolar voltage signals from the hall effect voltage and current transducers are less than 5 V, hence they are compatible with the analog channels of the logic controller. Signal conditioning circuits are used where ever necessary [6].

Figure 3 shows the block diagram of smart water distribution system. In this case, a water flow sensor (YF-S201) and normally closed solenoid valves are connected to the incoming pipeline of water storage unit. Water flow rate and quantity are computed from flow sensor data. A non-invasive type of ultrasonic sensor (HC-SR04) is used to measure the water level in the tank.

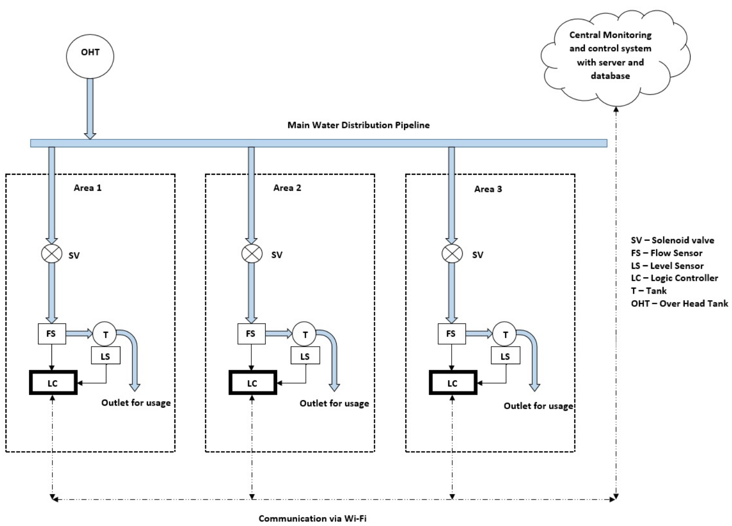


Fig. 3. Smart Water Supply System

All smart metering controllers are connected to a centralized server. Node MCU (ESP8266) is used as logic controller to communicate between server hosting the MySQL database and hardware installed in the field via Wi-Fi communication technology.

A. Coding methodology:

To implement an IoT based supervisory and distributed control system, HTML, PHP and Java scripts are used to develop web application. MySQL Server is maintained to keep the record of all data received from smart energy and water meter modules. ESP8266 is used as main controller unit that is programmed using Arduino software environment tool. Two different coding methods (shown in flow charts in fig. 4 and fig. 5) are used to develop distributed supervisory control system.

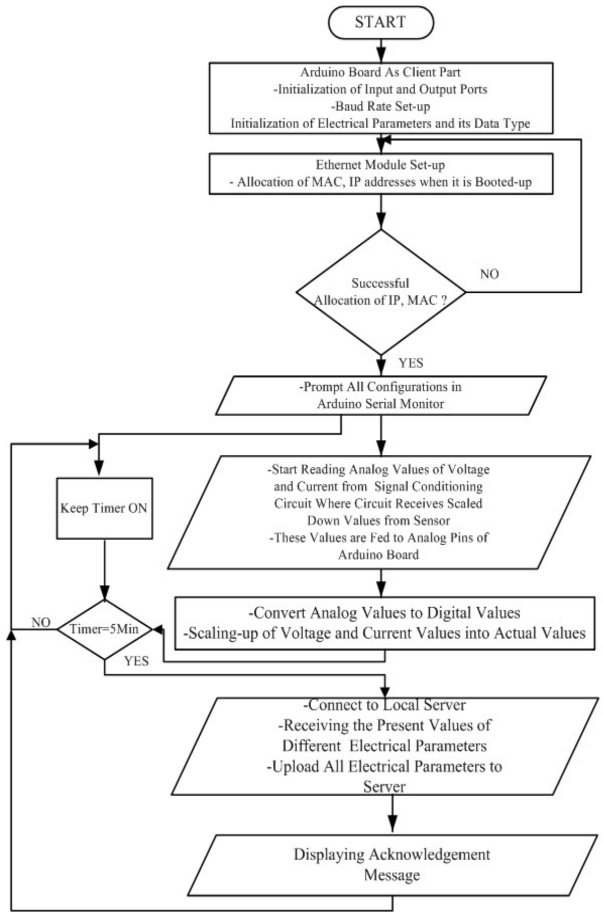


Fig. 4. Flow chart for smart energy management

B. Logic Controller (Node-MCU/Arduino Ethernet Shield):

Node-MCU includes firmware, which runs on the ESP8266 Wi-Fi SoC from *M/s Espressif Systems* and hardware, which is based on the ESP-12 module. The term “Node-MCU”, by default, refers to the firmware. Arduino Ethernet shield-2 allows an Arduino board to connect to the internet using the Ethernet library and to read and write in an SD card using the SD library. When logic controller is powered up, it does the initializations such as connecting to the nearest Wi-Fi router by putting required credentials, assigning static IP, declaring sensor value parameters etc [5]. HTTP request is sent from web application to respective boards (logic controllers) from remote PC. After validating the HTTP request by the controllers, it does corresponding operations. HTTP request could be of turning the water pump on/off, opening/closing of solenoid valves, isolator switches etc. or to know the present information regarding water level, water quantity, voltage, current etc. Logic controllers update the sensor data in the server database at regular time intervals as specified by the user.

Acquisition of Water Flow Data: A water flow sensor (YF-S201) is connected to ESP8266 board to record real-time water flow to the over-head tank when the motor is switched

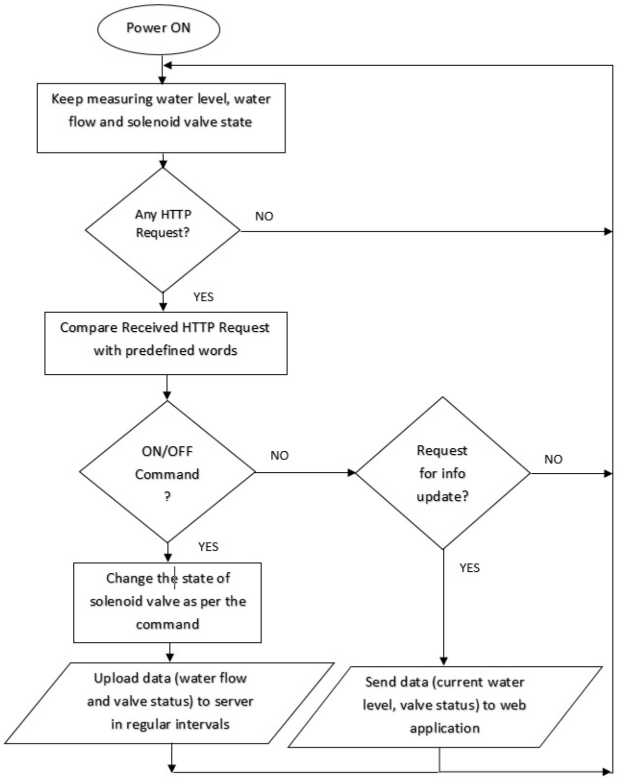


Fig. 5. Flow chart for smart water management

ON. The readings obtained from the sensor are then used to calculate the quantity of water supplied into the tank. The flow sensor can handle upto a maximum of 30 Ltr/min and each pulse is approximately 2.25 mL. These values are then calibrated by ESP8266 to obtain water flow rate per hour and total water quantity. Data line of water flow sensor is connected to D6 pin of Node MCU and the sensor is powered with 5 V supply.

Ultrasonic sensor (HC-SR04) operating at 40 kHz frequency is used to measure the water level in the tank. The sensor starts its measurement when the $Trig$ pin receives a high pulse (5 V) for at least 10 μs. This will initiate the sensor and transmits 8 cycles of ultrasonic burst at 40 kHz and wait for the reflected ultrasonic burst. When the sensor detects ultrasonic echo at the receiver, it will set the Echo pin to high (5 V) and delay it for a period (width) proportional to the distance. To obtain the distance, the width (T_{on}) of the echo is measured. Ultrasonic sensor has a range from 2 cm to 400 cm with a resolution of 0.3 cm.

$$Time = width\ of\ echo\ pulse\ in\ \mu S(microsecond) \quad (9)$$

$$Distance\ in\ cm = Time/58 \quad (10)$$

where 58 is the normalization factor for converting time of flight to distance. A fixed distance is programmed in ESP8266 controller. Thus whenever echo signal value reaches to that specified distance, it indicates that the water level has reached the threshold. Once it reaches the threshold value, necessary action will be initiated automatically by ESP8266 to start/stop the motor without waiting for the input from the server.

Water Flow control: Solenoid valves (normally closed closed configuration) of 2 inch diameter and operating voltages of 230 V AC are actuated by ESP8266 through a pair of relays. When ESP8266 sends logic high signal through GPIO ports, solenoid valves get 230 V and open the valve for water to flow. 1 HP submersible pump is used in this work to pump the water from water reservoir to overhead tank. Pump gets 230 V, 50 Hz when switched ON by the command from ESP8266 via relay.

C. Network Configuration

To supervise and control remotely via intranet, all the logic controllers (ESP8266 and Arduino ethernet shields) are assigned with static IPs. Username and password are preprogrammed onto the controller to connect to local area network. When the board is powered up, it tries to connect to local network and sends acknowledge signal to remote server that is accessed through the intranet. Web applications are developed to operate the solenoid valves and electrical relays from remote places to control and monitor the two utilities. Whenever any area is selected in web application, it pings to the corresponding controller having the specific static IP address. After pinging, either retrieval of sensor data or activating the relays is done by sending specific command to the selected IP address (logic controller). Sending data or receiving commands between logic controllers, server and web application are done through HTTP requests.

D. Graphical User Interface

For building the IoT based water distribution system, a web application is developed as shown in fig. 6 for the user to interact with the hardware modules. Using the web application, the operator is required to select the area from where the information needs to be collected. Here area names are mapped to IP addresses of development board. By clicking on the button provided in the web page, it pops-up the animation showing present values of sensors and status of solenoid valve in the selected area. Frequent updates on status of the valve and water level are obtained from board after sending HTTP request to the selected logic controller [7]. The slide switch in the web application sends ON/OFF commands to the controller to perform desired actions.

Since GUI is running on the server, operator can retrieve past data too. Based on range of date and time entered by the operator, web page will display information on water flow rate, quantity, status of valve, voltage, current, power and energy consumed during the period of interest. The GUI plots the graphical representation of water quantity supplied to the selected tanks. All these information can also be downloaded for off-line data analysis. Local server named *phpmyadmin* is created and maintained to store information regarding these two utilities. The server updates sensor data and web application's operation using SQL queries [7].

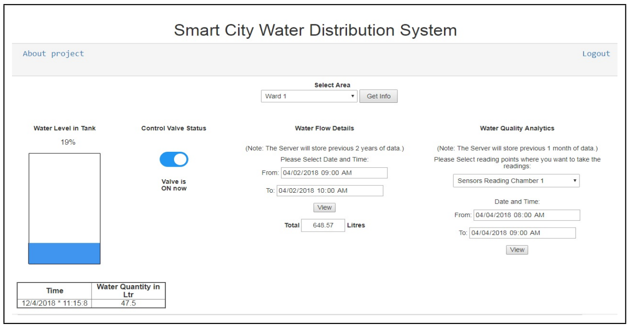
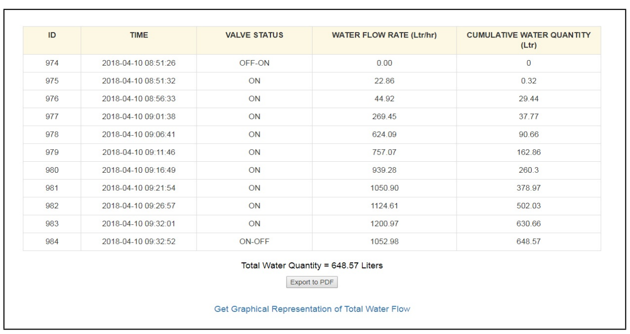


Fig. 6. Graphical User Interface for smart water distribution system

The web application developed and MySQL server have the following capabilities.

1) Access to view current water level, water quantity, valve status, energy consumption etc in a given utility location to an authorized person.
2) Access to control the supply of water and electricity in a given location/area.
3) Access to the past information on utilization of two utilities with time stamp.
4) Analysis of total water quantity or energy consumption over a specific time period.
5) Provision to take a back-up of data from the server for off-line processing.

Figure 7 shows a sample of such data stored and fetched from database for the period of interest.



ID	TIME	VALVE STATUS	WATER FLOW RATE (Ltr/hr)	CUMULATIVE WATER QUANTITY (Ltr)
974	2018-04-10 08:51:26	OFF-ON	0.00	0
975	2018-04-10 08:51:32	ON	22.86	0.32
976	2018-04-10 08:56:33	ON	44.92	29.44
977	2018-04-10 09:01:38	ON	269.45	37.77
978	2018-04-10 09:06:41	ON	624.09	90.66
979	2018-04-10 09:11:46	ON	757.07	162.86
980	2018-04-10 09:16:49	ON	939.28	260.3
981	2018-04-10 09:21:54	ON	1050.90	378.97
982	2018-04-10 09:26:57	ON	1124.61	502.03
983	2018-04-10 09:32:01	ON	1200.97	630.66
984	2018-04-10 09:32:52	ON-OFF	1052.98	648.57

Total Water Quantity = 648.57 Liters

Export to PDF

Get Graphical Representation of Total Water Flow

Fig. 7. Water flow data over specified time range

The figure gives information about the valve changed timings, water flow rate and the quantity of water flown into the water tank. Whenever pump and solenoid valve are switched ON, the time is noted in the database along with the flow rate and quantity at that time interval. The average water flow and cumulative sum of water quantity is recorded at every one minute interval. Switching status of valve is mentioned in table as ON-OFF or OFF-ON indicating valve is switching from OFF to ON state or vice-versa. Same information can be fetched from the database for off-line processing by authorized person. Note that in fig. 7 and 8, changes in solenoid valve status (ON/OFF) and pump ON/OFF action makes the abrupt transition in water flow rates.

The collective data (both water quantity and power drawn) obtained from database for the specified period is shown in graphical format as in fig. 8 and 9. In fig. 9, initially a constant load was connected to the supply and then changed to a variable load. A PHP code is written in JSON format to retrieve filtered values from database [7]. Plotting a graph from fetched data, animation of pointer movement on graph screen, downloading options etc are written in Java script.

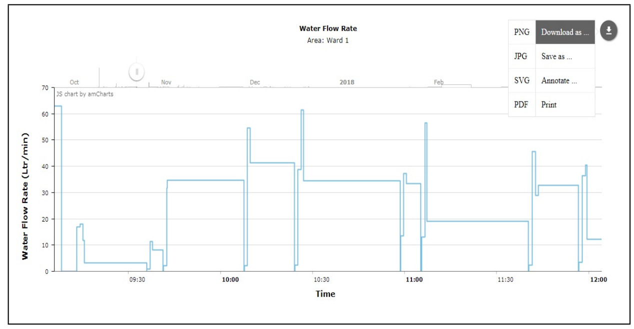


Fig. 8. Graphical representation of water quantity supplied over a time

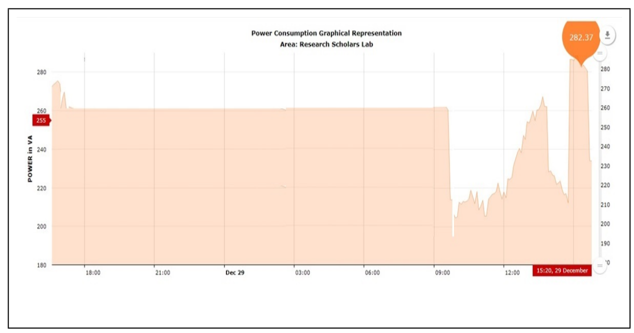


Fig. 9. Graphical representation of power drawn by the load over a specified time range

E. Validation of Controller and Interface Hardware

Arduino and Node-MCU (ESP8266) embedded boards are used as logic controllers in this work. Total memory consumption by the final sketch is summarized in table I. As far as size, reliability, performance and memory consumption are concerned, Node-MCU is found to be better suited for this particular application. Baud rate of 250000 is set to operate ESP8266 for faster communication. The ultrasonic sensor (HC-SR04) used in this work provides accurate water level with an error of less than 0.5% and a standard deviation of 0.2 to 0.6 cm.

IoT Application is developed to supervise and control water distribution system using IBM's Node-RED editor (Fig. 10). Data on water flow, level and quantity are sent to IBM watson IoT as specific events. These are loaded to Node-RED application and the operator can use this application from a remote place using the internet.

Fig. 11 shows the GUI of web based water supply monitoring and control in real time.

TABLE I
MEMORY UTILIZATION IN DIFFERENT LOGIC CONTROLLER BOARDS

Memory Utilization in Logic Controllers		
Storage Parameters	**Node-MCU (ESP8266)**	**Arduino-Ethernet Shield**
Built-in ROM	1.04 MB	32 KB
ROM utilization	230 KB	20 KB
Percentage of utilization	23 %	63 %
Built-in RAM	82 KB	2 KB
RAM utilization by global variables	34 KB	1.4 KB
Percentage of utilization	40 %	70 %

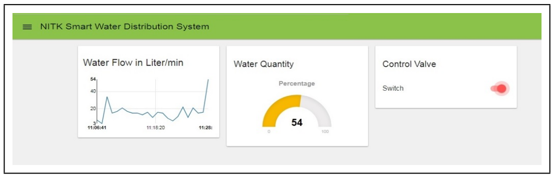


Fig. 10. IBM Node-RED application development for smart water management system

Fig. 11. Water flow and water quality display on GUI in real time map

IV. CONCLUSION

Based on the survey of two utility systems in different places, major issues were identified and studied. Lack of monitoring and control of the utilities is found to be the main reason for wastage and inequitable distribution of the utilities. An automated system is proposed to reduce the man power, reduce leakages, improve efficiency and ensure equitable distribution of the resources.

A. Benefits of Supervisory and Distributed Control System

The proposed scheme of smart water and electricity management system has the following benefits.

1) Ensures efficient and equitable energy and water distribution.
2) Automatic control - Lesser manpower and lesser human intervention.
3) Detect and prevent water wastage with sensor deployment.
4) Lesser maintenance cost.
5) Life time of pumps will increase by avoiding dry run.
6) Preference based auto turning-off of electricity supply during heavy energy consumption.
7) Remote control & management via intranet/internet.
8) Data Analytics on water flow, energy consumption during different time-lines. Availability of on-line information on water level, quantity of water in tanks, load voltage and current etc.

V. FUTURE WORK

The supervisory and distributed control system can be augmented to include the following.

1) Addition of on-line water quality check for treated water.
2) Developing smart grid technology and shifting of isolator switches to solar powers during power failures.
3) Water and electricity billing.
4) Real-time water leakage sensing and preventive maintenance.
5) User friendly android application development for different modes of communication - GSM/GPRS, LoRA, Xbee etc.
6) Displaying real time water quality, water flow and energy consumption data on Dashboards map as shown in fig. 11.
7) Solar Power based system to power up hardware modules in remote locations.
8) Interfacing present system with rainwater harvesting. Extend the scope to include smart irrigation system. Auto shut-off sprinklers when level of water goes below threshold. Auto turn-on when the moisture content is low and periodic On-Off.

REFERENCES

[1] L. Da Xu, W. He, and S. Li, "Internet of things in industries: A survey," *IEEE Transactions on industrial informatics*, vol. 10, no. 4, pp. 2233–2243, 2014.
[2] M. J. Mudumbe and A. M. Abu-Mahfouz, "Smart water meter system for user-centric consumption measurement," in *Industrial Informatics (INDIN), 2015 IEEE 13th International Conference on*. IEEE, 2015, pp. 993–998.
[3] M. M. Rathore, A. Ahmad, A. Paul, and S. Rho, "Urban planning and building smart cities based on the internet of things using big data analytics," *Computer Networks*, vol. 101, pp. 63–80, 2016.
[4] A. Botta, W. De Donato, V. Persico, and A. Pescapé, "Integration of cloud computing and internet of things: a survey," *Future Generation Computer Systems*, vol. 56, pp. 684–700, 2016.
[5] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "Smart grid technologies: Communication technologies and standards," *IEEE transactions on Industrial informatics*, vol. 7, no. 4, pp. 529–539, 2011.
[6] M. K. Mishra, K. Karthikeyan, G. Vincent, and S. Sasitharan, "A dsp-based integrated hardware set-up for a dstatcom: design, development, and implementation issues," *IETE Journal of Research*, vol. 56, no. 1, pp. 11–21, 2010.
[7] S. D. T. Kelly, N. K. Suryadevara, and S. C. Mukhopadhyay, "Towards the implementation of iot for environmental condition monitoring in homes," *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3846–3853, 2013.