IoT leak detection system for building hydronic pipes


**Saeed, Nagham ORCID logoORCID: https://orcid.org/0000-0002-5124-7973 and Urbonaviciusa, Audrius (2019) IoT leak detection system for building hydronic pipes. International Journal of Engineering and Manufacturing(IJEM), 9 (5). pp. 1-21. ISSN 2305-3631**

**http://dx.doi.org/10.5815/ijem.2019.05.01**

*I.J. Engineering and Manufacturing,* **2019, 5, 1-21**
Published Online September 2019 in MECS (http://www.mecs-press.net)
DOI: 10.5815/ijem.2019.05.01

*Available online at* **http://www.mecs-press.net/ijem**

# IoT Leak Detection System for Building Hydronic Pipes

Audrius Urbonavicius[a], Nagham Saeed[a]

[a] *School of Computing and Engineering, University of West London, W5 5RF, United Kingdom*

Received: 06 July 2019; Accepted: 10 August 2019; Published: 08 September 2019

**Abstract**

Building's Air Conditioning systems require moving liquids for dweller comfort. Clogged pipes, system degradation can cause pressure buildups, leaks and other faults which leads to damage to the building. Most of the leaks in the commercial building occur due to poor maintenance and/or material degradation. Visual inspection is most predominantly used to solve this problem in the industry. This paper introduces the Internet of Things technology to detect leakage in building's hydronic pipes with the support of sensors, fault detection method and mechanical control. The system consists of: Microcontroller, Windows application and website application. Internet of Things technology was used to monitor and control the hydronics using microcontroller's capability of connecting to main server which is used to transmit the data to the cloud. The prototype was successfully built and tested. Promising results show that leaks above 2ml/s could be detected after 4 seconds specifically for the built small-scale system while control and monitor feature could be implemented with Internet of Things technology.

**Index Terms:** Pipeline automation, Leak detection, Building services, Supervisory Control and Data Acquisition (SCADA) and Internet of Things (IoT), Industry IOT (IIOT).



* Corresponding author.
E-mail address: audriusurbo@gmail.com

## 1. Introduction

Building design was always important engineering field, which embraces many different branches for great number of specialty engineers to work on. Mechanical, electrical and plumbing fields are the most important of them all. Fluid transportation, in this case – water, glycol, mineral oil or other coolant, is fault-sensitive section that needs attention. [1]

Faults in the pipes result in leaks, which causes building damage and financial cost. In the older buildings and in some current infrastructure, leaks are being detected by visually inspecting the pipes, or noticing the apparent symptoms of leakage. These symptoms can range from higher energy usage, damaged walls, ceilings or higher water usage. Besides visual inspection, devices that shuts the water if it reaches the limit are used to detect the leaks too. [2]

This paper proposes a system that could be used to monitor and control the building hydronic and other pipes. This aim requires several different objectives to be completed, which will be stated in the following section. In the building engineering field, this would closely relate to Building Management System (BMS) and Supervisory Control and Data Acquisition system (SCADA). This project was partially inspired by these systems and their current flaws. Connectivity is very important in these systems as BMS is critically important in managing energy usage. This can vary from lights left on, where no motion-detector switch is installed, to hot water leaks which would increase energy usage more substantially [3]. Current systems in market lack connectivity and automation, some systems are offering just supervisory monitoring, while others feature some automation. [4]

The system proposed was designed to monitor and control the building pipelines and it also can be modified to use other sensors to save more energy and resources. The small-scale prototype was implemented to prove the methodology. IoT technology was used to create a monitor and control software. Interface with server alongside web application was developed to offer supervisory control features for remote and local users.

The aim of this research is to create a solution for rapid control and monitoring of building hydronic pipelines. The system was required to be 4th generation of SCADA system. This means that Internet of Things technology needs to be integrated into the system, which improves interoperability [5]. The prototype (practical implementation) is to be done with these functions in mind:

1. Demonstration of working leak detection technique
   a. Receive data from sensors continuously
   b. Process data
   c. Send data to database
2. Demonstration of control of the pipeline
   a. By processing data act by changing position of valves or other control components
   b. Act with user intervention by changing position of valves or other control components
3. Demonstration of the state of the system in real-time
   a. Visualization:
   (1) Sensor activity
   (2) Control system states
   (3) Other component states and activities
   b. Create a graphical user interface in server supervisory and web applications

The three parts of the system will be designed to work together to create a software and hardware solution for hydronic pipe monitoring and control system. This system will be practically applied and tested.

Aims and objectives are set to be completed by selecting best approach with aid of previous research. This will be discussed in the following section.

The rest of the paper is organized as follows: section II dedicated for related work, section III presents the methodology behind the system. Detailed explanation for the system implementation was in section IV whereas section V showing the testing and calibration for the IoT Leak Detection System under normal conditions. Results were shown in section VI for uncontrolled and controlled leak. Section VII contains an analysis for the results achieved while section VIII summarize the finding. Finally, future work was provided in section IX.

## 2. **Literature review**

Detection of pipeline faults are also time-critical as most of the pipes in the buildings are in cavities hardly accessible. Leaks in such areas can cause great damage to the building, from damaged ceiling tiles, plasterboard, to other hydrophobic building materials such as wooden parts of the building. Furthermore, effects of leaks can cause illness as slow leaks cause humidity increase and it leads to mold formation. Also if the leak occurs in AC systems, it can cause higher energy consumption, especially in heating. All effects mentioned previously leads to higher maintenance cost [6]

Most of the leaks in the commercial building occur due to poor maintenance and/or material degradation. Copper pipes are projected to last for 20-25 years [7], but failure can occur in the first years. Therefore, they require constant maintenance, but as some pipes are in hardly accessible places, it is hard to do so.

Different pipelines in different industries require different solutions as one solution does not exist for all. Water Supply and Sewerage Utilities uses wider pipes than hydronic [8], but the principle is the same. Hitachi has developed the IoT gateway to control and monitor sewage pipes. [9]. A method uses optical fiber to transmit power and data to/from the sensors inside the pipes. This method is not applicable to hydronics as the pipe diameter is usually smaller than sewage and the high/low temperatures, tight bends not going to work inside the pipe. Sewage piping, on the other hand, has more area to work in. The cross-section diameters vary from 3 inches to 5 inches providing space to insert optical fiber without creating blockages or any other disruptions. This method can be adapted by using optical fiber instead of copper wires, inserted in the insulation (lagging) of the pipes, creating power and data links between the server and the sensors. This would increase reliability, since optical fiber is nonconductive and leaks/condensation in the lagging will not create right conditions to cause short circuit.

One of the solutions for detecting leaks that could be applied in the building services is mass flowrate comparisons. [10] Not to be confused with volumetric flowrate comparison, mass flowrate is measured in units of mass per units of time. This method can be used for gas transportation as gases tend to expand when heated, meaning the volume increases while mass does not depend on pressure or temperature. Mass flow rate can be measured with Coriolis, thermal mass, and other flow meters. These sensors are generally expensive, meaning they are not suitable for the currently affordable practical experiments. Volumetric flow meters can be used as liquid in hydronic pipes changing temperature predictably. If the AC works as the district heating/cooling (cold/hot water is being pumped from outside facilities) temperature inlet and outlet of coil unit depends on the liquid temperature and room temperature. The temperature difference can be predicted and flow meter sensors calibrated to take this difference in account if it is high enough.

Other field worth mentioning is oil/gas industry. It has been around for over a hundred years and the research on it promised big financial gains. Therefore, huge amounts of research on all aspects have been conducted. One of the subfields is oil pipelines, where Z. Jia and et al. [11] proposed a novel hoop strain based negative pressure wave (NPW) approach was used to detect and localize pipeline leakages in a 180 ft PVC pipeline equipped with five manually controllable leakage points. By having pressure and tank volumes sensors can over time compared to identify the leak [12]. This method can be implemented in many areas where system medium is travelling between tanks

One of the monitoring system's purpose is to identify the state of the flow. Many ways of acknowledging the flow has been researched and invented, one of them is vibration sensor [13]. Cracks in the operating pipe, therefore leak, causes vibration waves travelling through the pipe and surrounding medium. These mechanical vibrations are very small power signals requiring high signal-to-noise ratio hydrophones, acoustic emission or other sensors [14]. This method is preferably used in underground pipes as the medium (gravel, sand, concrete) between the sensor and the pipe helps the vibrations reach sensor in ground level and even pinpoint the exact location of the vibration source. This method is advanced, but hydronic pipes in the building create huge amounts of noise from compressors, valves and other AC components [15]

IoT technology was discussed and developed as early as 1982 with vending machine being first physical every-day appliance connected to the internet. [16] Since then, IoT technology was broadly researched and expanded. As of 2019 it is common to have home appliances with internet connectivity feature such as fridges, toasters and doorbells. IoT technology is applied to numerous engineering fields: automotive, commercial devices, infrastructure, industrial etc. It is also constantly expanding as the costs are decreasing and the research of improvement is still rising. By 2020 Korean government made plan implies that more than 10000 plants to be optimized with the help of IoT and other information technology software.[17] Several issues rise due to the rapid development on IoT technology. Security flaws let the IoT devices to be controlled without authorization due to lack of encryption in communication. [18,19] General device safety is also concern as software bugs can cause unexpected results. If the sensor of the smart-door entry system malfunctions, it can enable access, creating dangerous situations. [20]

## 3. **Methodology**

Technique based on mass balance concept offers solution that can be implemented and calibrated to work on most of the systems where total quantity of fluid entering and leaving a network is the same [21]. Mass flowrate is great method as the liquid or gas inside expands and contracts due to the temperature change, meaning it will change in volume. On the other hand, it is complicated and expensive to implement mass flowrate in the system. Instead of this, volumetric flowrate was used instead, as the sensors are widely available and easy to implement. This, however, creates a problem if the liquid temperature in the pipeline is varying substantially. This Volumetric flow comparison method compares the flowrate and multiple points and checks invariances over the time. These invariances can be liquid slow drift which happens naturally or a leak between the sensors. Equation 1 illustrates the relationship between the inlet and outlet volume differences over time (flowrates) in theory. Variable *Total Leak* is the leakage amount in volume over time. It is defined by the difference of inlet and outlet flowrate. If the system is balanced (inlet flowrate is same as outlet flowrate under normal conditions), *Total Leak* will be zero. But the reality does not represent the theory and this equation can not be taken for granted, but it can be relied upon for future work.

$$Total\ Leak(t) = \sum_{t=0}^{t}\left[ flowrate_2(t) - flowrate_1(t) \right] \tag{1}$$

Creating a viable system ready to be fitted commercially requires careful research and testing. A prototype was chosen to be built with most commonly used pipe in AC industry – 15mm copper pipe. As this is the most widely used transfer pipe, the results are to be expected to reflect the real-world situations. The methodology section contains theory and practical aspects of leak detection, mechanical control and software implementation. Multiple tests were done, proving the sensor validity and software functionality.

SCADA systems uses wide range of components to unify and monitor and control the physical infrastructures or buildings. Sensors and controls are first level of physical layer. Sensors at the monitor point sends the signal to specific devices, where signal is processed and control signal is either sent or not depending on the monitor point signal. Those specific devices can be Remote Terminal Unit (RTU), Programmable Logic Controller (PLC), Intelligent Endpoint Device (IED), and all of those devices have common function: to send, receive signals and use control components. These controllers are connected in network which unifies the system via Master Terminal Unit (MTU). MTU acts as the server and receives all data and sends it to databases or the last layer: interface. Interface is the layer connecting user and the system. Figure 1 shows the layout of SCADA system.

For PLC/RTU/IED microcontroller was used. It is less expensive, easier to use, but less robust, not designed for SCADA systems. MTU, Interface and Database was used on one device, therefore simplifying the system.

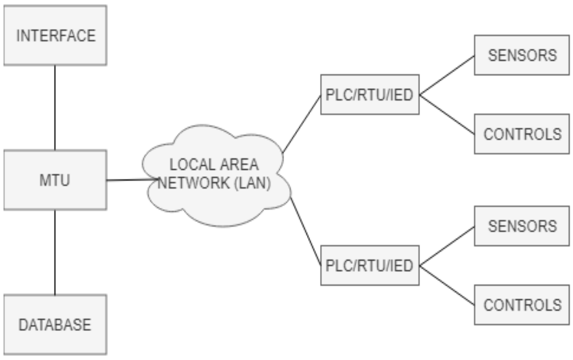



Fig. 1. SCADA system layout

## 4. Implementation

Prototype was built to reassemble building district heating/cooling system, where main flow and return pipes are common. These systems work by supplying building with chilled/heated water and running it through Fan Coil Units (FCU) which consist of coil where heat is exchanged and a fan. Prototype does not consist of FCU, but the piping is similar to commonly found layout. Figure 2 shows the physical layer of the system with bottom pipe as flow and top as return with bypass pipe for emergencies in the middle.

Fig. 2. Physical system layout

*4.1 Leak detection.*

Leak detection was achieved by employing volumetric flow comparison technique to compare the flow at two or more different points in the pipeline. Simple line flow imbalances or average imbalances in pipe content over period of time can aid to detect a leak. By having two flow meters in series, a constant or repeated variance of flow means a possible leak. This technique does not detect the leak immediately as the flow meters have slight variances constantly and this needs to be averaged out and computed by cancelling out other parameters.

The sensor itself was chosen to be turbine sensor due to low cost and easy operation as the prototype needs to be made within a set budget. Models of the flow sensors are YF-S201 [22] and YF-B1 [23]. They operate in the same principle, but calibration indices are different: 7 and 11 respectively.

The flow was measured in liters per minute and later converted to milliliters per second (ml/s). Due to invariances in different flow meter calibration factors, normalized values were used for visual representation. Normalization was done by calibrating the sensors in the no-leak condition and taking the average value as normal, following sensor measurements were compiled as difference in ratio between normal and real time value from the average calibrated value.

Initial testing concluded that, further calibration is required. Several following experiments were conducted, and calibration values got changed in trial and error method. After each test, calibration indices were changed to represent theoretical values as close as possible.

*4.2 Mechanical control*

Mechanical control was achieved by placing SOVs into the pipeline. These valves can be normally-open (NO) or normally-closed (NC), the usage of them depends on the usage of the pipe. For example, if pipe is constantly used, it is recommended to use NO valve to save energy and vice versa. It is important to have multiple SOVs to isolate the leak as the pressure affects the water from incoming and outgoing pipe directions. When leak occurs, the pressure affects the water from inlet and outlet, therefore, to minimize the leak amount, both directions need to be cut-off.

DC pump was used to move the liquid across the system. The 2.5W pump reached maximum of 8L/min was proven to be within acceptable range. The real-world implementation would be more sophisticated. District heating uses primary pumps, secondary and set of boosters for bigger buildings.

*4.3 Monitoring and Control software*

The software for monitoring and control consist of three parts: Microcontroller, Web application, Server and server interface. Microcontroller is used for signal processing and transmission between main server and the prototype. The server is the main interface and control unit, while web application is a smaller version of the main server, hosted in the cloud accessible with internet connection. The following sections include further details about each part

*4.4 Microcontroller*

Microcontroller uses relays for switching component states and serial Universal Asynchronous Receiver/Transmitter (UART) communication protocol. Arduino Nano was used as microcontroller as it is adequate for this operation. It was used for communications between server and the prototype. The microcontroller reads analogue voltages and converts it to digital data; therefore, it can be stored as in the program as integers, double, string or other datatypes. The data then can be manipulated and sent via the serial communication to the server. The main functions of microcontroller are:

- Send current SOVs and pump states to the server
- Calculate the flow
- Send current flow meter readings
- Send signals to relay module in order to control the flow

Microcontroller was set up to count the negative edges of the pulses per each second which gives frequency of the waveform. The frequency was calculated by multiplying it by calibration factor which gives liters per minute. In the meantime, control data was read from the server and if it mismatched (current different than read), the states of the components (SOVs, Pump) were changed.

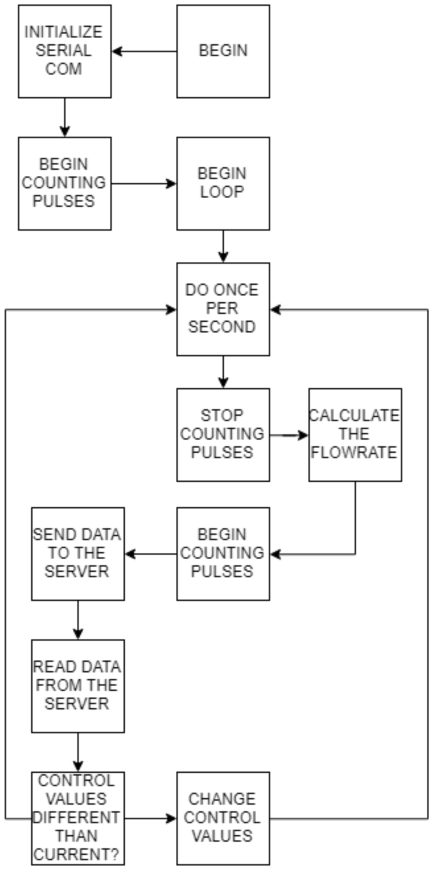



Fig. 3. Microcontroller activity flowchart

Figure 3 illustrate the Microcontroller process flow. The Begin condition simply turns the power to integrated circuits and sets starts running the code. The condition can be triggered by turning on the power. After all variables are initialized and pins set up correctly, it starts counting pulses from sensor signal. More precisely it counts negative edges from square wave outputted from the sensors. These steps are considered as a set-up, before the loop. The loop consists of arguments which are to be executed once. In following steps, pulse counting function is stopped and flowrate is calculated by dividing frequency by calibration factor. After the calculation is completed, the pulse counting process is started again. This was disabled because the variables used in calculation can change at the same time the flowrate is being calculated, causing errors. Sensor data and current component states are sent to the server via serial port. Right after that, data retrieved from the server is processed while changing the component states if they are found to be different from current states. As this is the last function block in the loop, the process is starting again.

Data was sent using serial communications with server. The component states were in integer datatype while sensor data was kept in double datatype, because it was real number. All this data was then converted to string datatype and packed together with space as the separator. The process is visualized in figure 4.

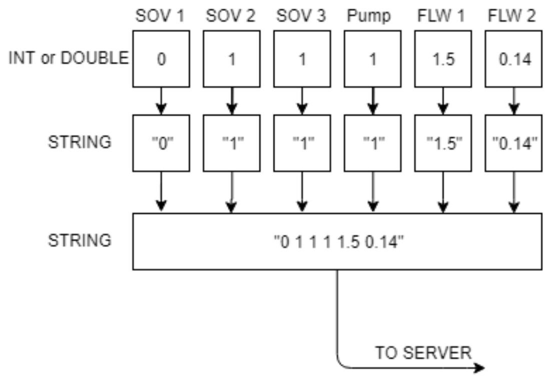



Fig. 4. Data sent from microcontroller to server

## 5. Server and Server Interface

Server was developed for local monitoring and control features of the system. The system gathers data from the controller and visualizes it on the interface. It also stores the values in local and cloud databases, therefore enabling IoT technology to use its potential remotely. This server application and interface was developed to have user friendly way of gaining full control over the system.

Application is to be executed once and should run as long as the pipeline is active. When launching the executable, the communication link must be established between the software and the microcontroller. Before starting feeding the information to the system, multithreading process is started. One thread reads/writes data from/to microcontroller and updates the graphs and databases while other thread waits for user input commands. Data is also being changed in the first thread, because of this some drawbacks are created. These drawbacks (bugs) which occur due to closing the program and changing component states. While one thread wants to change the values through serial port, other thread also uses the same values, creating exceptions while crashing the program. These needs more time to resolve since multithreading software is found to be difficult to develop. This process is visualized in figure 5.

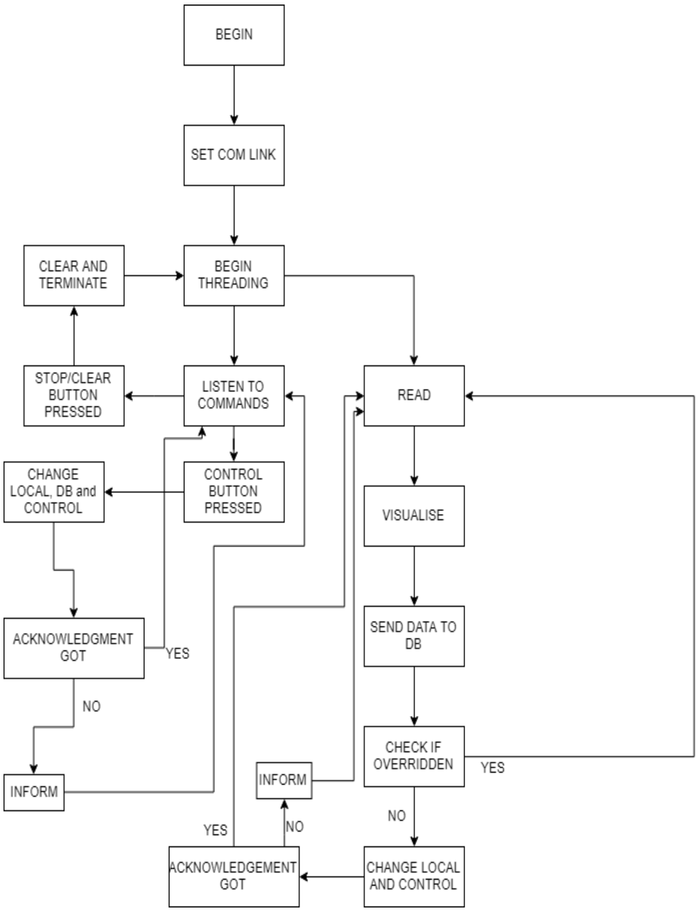



Fig. 5. Server-side interface flowchart

The program for the server and interface was written in C# using Microsoft's Windows Forms Graphical User Interface (GUI) class library and Microsoft Visual Studio 2017 as Integrated Development Environment (IDE). One form was used in the program and the components list in the form includes "Labels", "Graphs", "Buttons" and "TextBoxes".

Data from microcontroller was read using serial communication, in string lines. Since the lines were read constantly, it is possible the line read will be not timed correctly and values incorrect, therefore string data was read correctly if the length of it fell between 18 and 20 characters. Since smallest possible string while everything is 0 is 18 and largest is 20. The string read was split with space as splitter symbol. String split values were converted to integer or double datatypes. This process was visualized in figure 6.

Process of data sending to the microcontroller with component state commands differs from data reading pro-cess due to the code limitation. Function in the microcontroller can read string line data until specific symbol. That specific symbol was chosen to be '$'. Integer and double datatype values were converted to string and put together with space as separator and symbol '$' was added to the end of the string that was about to be sent. Full string sent data includes component states to be changed to and symbol $' at the end.

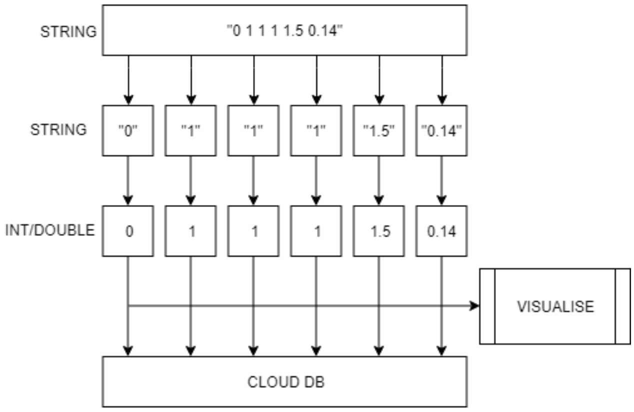



Fig. 6. Server reading data from microcontroller process

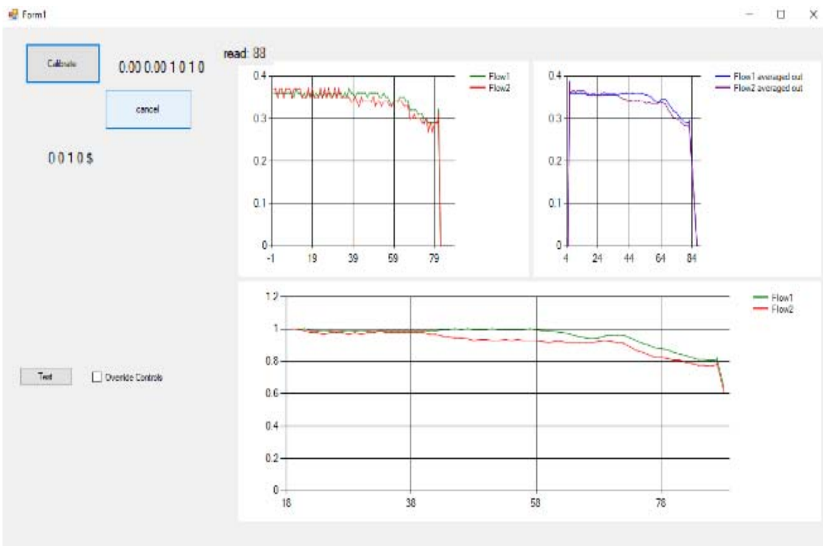



Fig.7. Snapshot of the server monitoring interface

Figure 6 shows the snapshot of the server-side interface application. Values on the right of 'Calibrate' button shows the values sent from the controller and the bottom ones show what is to be sent back to the controller. This was done for debug purposes as real-time debug technique is more efficient than built-in to IDE debugger.

Button 'Calibrate' is self-explanatory, it calibrates and normalizes values for third graph when the system is in no-leak condition. During the calibration, the values read are being taken as 'normal' values and any deviation from it are graphed as ratio between 'normal' and averaged out in second graph. This was done due to different pressures at those points which leads to different amount of flow at the same time, but because it is in no-leak condition, the total amount of the flow is technically the same.

"Override Controls" tick box changes the feed where the commands are being taken from. If it is not ticked, is being controlled from cloud, else it will be controlled by the server.

### *5.1 Web Application*

Web-application purpose is to control and monitor the system over long distances. The server features connection with database and can control the components from the database, while web-application has ability to change those values in the database from which components are controlled from. The web application also able to visualize the data just like the server does, except is not yet implemented to update the graphs with in real-time. Figure 8 shows the design of the web application and the snapshot was taken during one of the experiments.

It was developed with WebForms class library and it was hosted in Microsoft Azure servers. Most of the written code was also C# with some web development languages for website design.

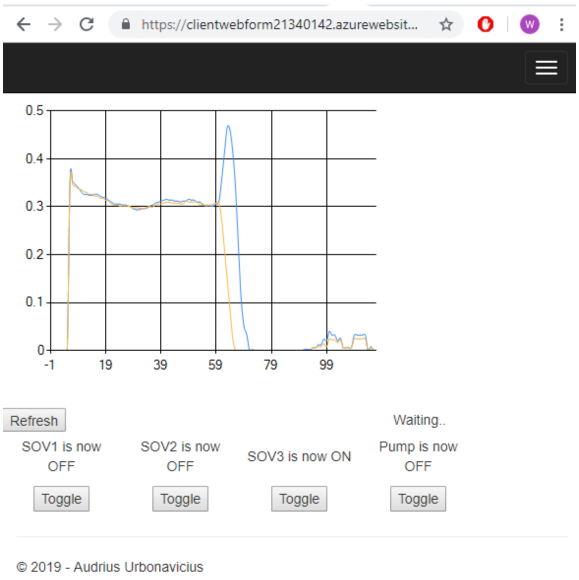



Fig.8. Snapshot of the web application screen

### *5.2 Prototype initial set-up*

The practical prototype tests were conducted to simulate the system as close to real life scenarios as possible. The Table 1 consists of initial set-up parameters and components in which tests were completed. Distance between flow meters was measured to be 40cm and the total length of the prototype equals to 1m.

Table 1. Initial set-up parameters and components

| Component | Parameters |
|---|---|
| Medium | Water at 20 °C |
| Pipes | Diameter: 15mm<br>Material: Copper |
| Pump | Nominal power: 2.5W<br>Maximum flow rate: 8000ml/min<br>Maximum pressure: 10Bar |
| Flowrate meters | Operating voltage: 5V<br>Pulse Duty: 50% ±10%<br>Calibration factors: 11 (YF-B1)<br>7 (YF-201) |
| SOV | Operating voltage: 12V<br>Default position (at stand-by state): closed<br>Pressure drop: not given |

Figure 9 shows the full layout of the prototype system. It consists of three parts: software (IoT), hardware (controls) and plumbing (piping). IoT aspect features local and cloud database for storing sensor data. Physical layer, most importantly plumbing includes tank, two routes, flowrate sensors, SOVs and manual valve for simulating leak. Normal route is longer and pressure drop is higher due to SOV physical properties. Bypass route was made for leak minimization. During normal operation, liquid is flowing through normal route while SOV1 is closed and SOV2 and SOV3 are open, and in leak conditions liquid is flowing through bypass route, where SOV1 is open and rest of the SOVs are closed.

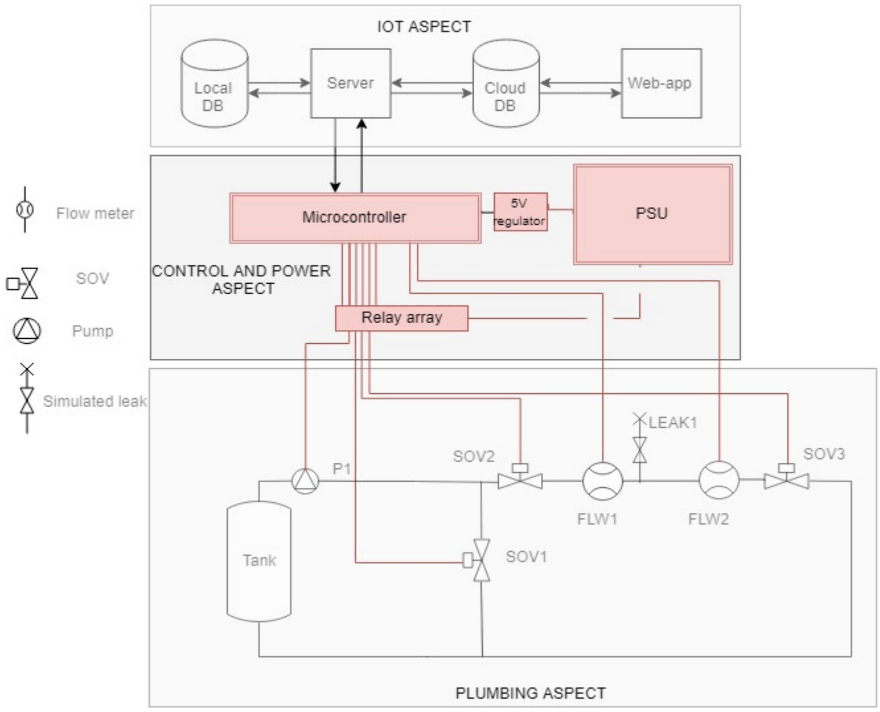



Fig.9. The built prototype schematics

## 6. Testing and Calibration

Several scenarios of testing were done to prove the effectiveness of the prototype. The scenarios created for the experiments were designed to reflect the real-world experience. (System ***Calibration Under Normal Conditions)***

This section analyses leak detection testing response by creating normal conditions without leak, leaks without intervention and leaks with automated response. During the testing the leak amount was differing from each other experiment, aiming to provide better accuracy results.

During the leak, leakage amount was recorded manually in terms of volume and then calculated from the sensor output data differences. The leakage process time was recorded in the software, so that sensor data could be compared to it.

***First test*** were done without leak, which means the difference between flowrates should be zero. During first experiment, the flowrate of first sensor was constantly lower than second flowrate. This would indicate two things: system is not calibrated, and calibration factors are wrong, or liquid is being added in-between the sensors. Under normal conditions no liquid is being added, which means calibration needs to be done.

Figure 10 shows the sensor data from very first experiment. The calibration values were 11 and 7.5 for first and second sensor accordingly. Leak data graphed was calculated with equation 1. The flowrates were originally measured in liters per minute, but it was converted to milliliters per second because time x axis is in seconds.

The same test revealed that pump voltage can drop, lowering the flowrate of the system. This occurs at around T = 50 s. This leaves for further work to be completed to diagnose the issue. It is possible that the motor is faulty, or the heat due to the stress lowers the voltage for self-protection. The voltage rises gradually up again to the normal amount.

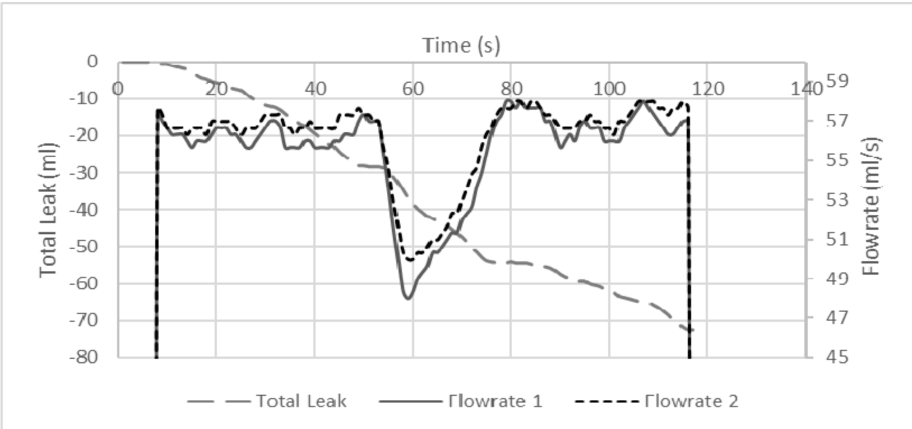



Fig. 10. First test with normal conditions without calibration

***Second test*** was also done under normal no-leak conditions. All parameters were left untouched, except the calibration factors. Assuming that from the first test visualized in the figure 10, flowrate 2 is correct, flowrate 1 sensor can be calibrated. The calibration factor for the first sensor was lowered to 10.5 by 0.5. This would increase the first sensor flowrate slightly giving better results. The results were graphed in figure 11.

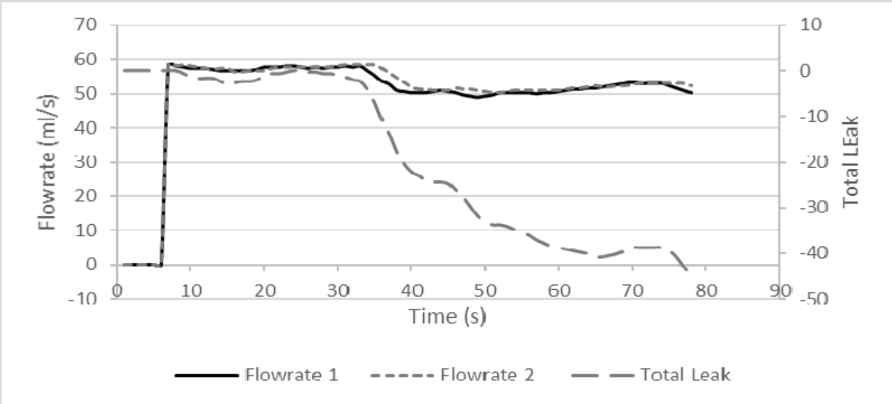



Fig. 11. Second test with normal conditions and mild calibration

Once again, the first flowrate is being slightly smaller than second flowrate. Up until T = 32 s mark, flowrates were at around the same level suggesting that calibration was successful. From T = 32s the voltage dropped once again, lowering the flowrates and making the difference between flowrates larger. The total leak was calculated to be around -45ml, which indicates further calibration is needed.

***Third test*** was done in the same manners, and the calibration value for first sensor was set to be 10.2. The measured values were visualized in figure 12.

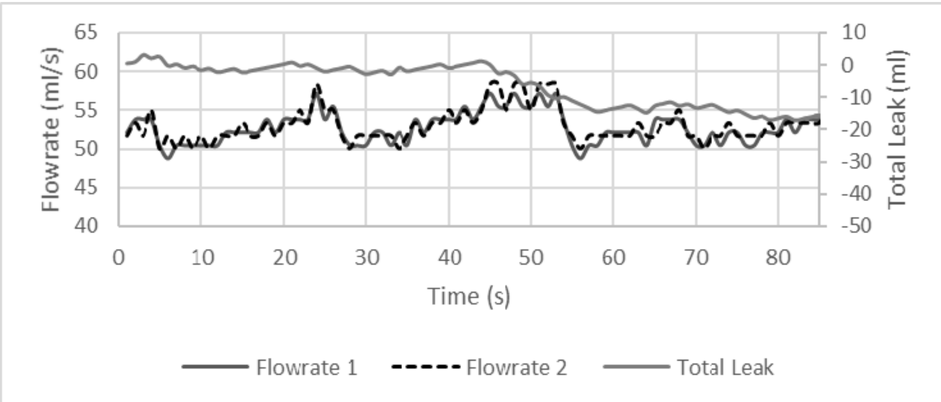



Fig. 12. Third test in normal conditions and further calibration

This test results looks more promising. The difference between flowrates is very small. In fact, it only reaches 15ml over 80 seconds, which is much more accurate compared to previous testing, proving that calibration helped. On the other hand, it was noted, that difference of flowrates goes up in the positive axis, meaning calibration values needs more adjusting.

It was noticed, that absolute flowrate difference never went above 2ml. Since the tests were conducted under normal conditions, it can be assumed that if the difference is below 2ml, it will not be taken in account as the leak.

With this assumption fourth test was done under normal conditions with the calibration values changed to 9.9 and 7.52 for sensor one and two accordingly. The sensor data was graphed in figure 13.

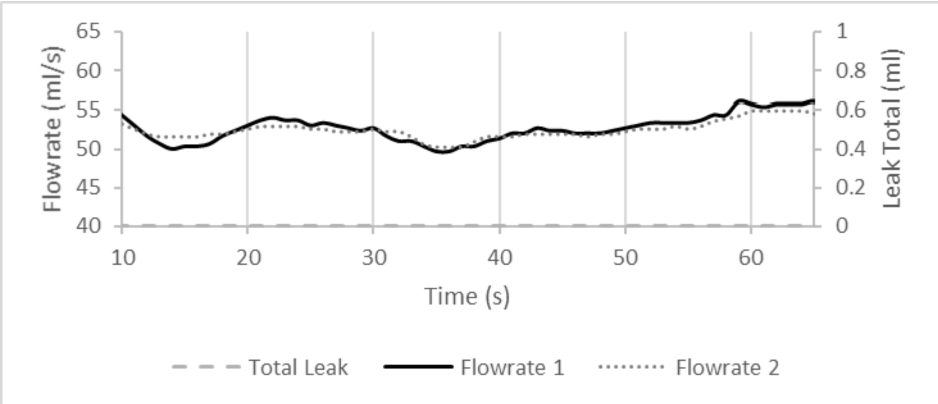



Fig.13. Third test with normal conditions and improved calibration

The test duration was about 70 seconds, and in that time, no difference was recorded in flowrates, with the assumptions made previously. The flowrate during this experiment varied from 50ml/s to 55ml/s. This is normal as the system is not perfect, pump is not working in theoretical conditions, giving dips in voltage leading to dips to flowrate.

This experiment sums up the performance under no-leak scenario. The difference between flowrates are constantly zero because of the successful calibration and assumptions made. At this point further experiments with introduced leaks can be done with accurate measurements.

## 7. Results

This section includes the results for uncontrolled leak scenario compared with controlled leak by intervention

### *7.1 Uncontrolled leak*

During the testing of uncontrolled leak, the system was not intervened or controlled in any way. The leak continued up until the storage tank was empty. Figure 14 shows the uncontrolled leak sensor graph where leak occurs at T = 60 s. Couple of things can be observed. The second flowrate drops immediately while first flowrate increases. Since the leak occurs between the sensors, the drop at second sensor is because liquid chooses easier path, the leakage direction as the following pipe has much higher physical resistance than leakage pipe which is completely open. The first flowrate also increases due to the same reasons. Since the resistance is much lower, the liquid doesn't have to go through the normal route, it can be pushed at the faster rate. (Refer to figure 9).

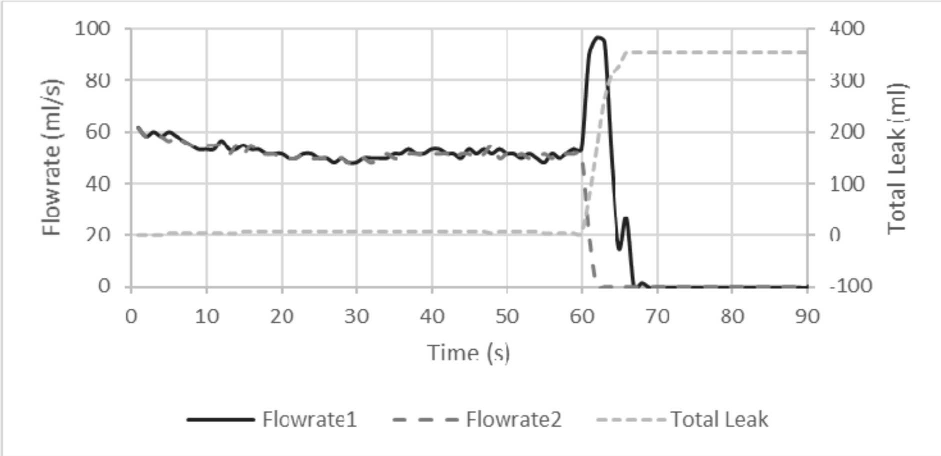



Fig.14. Uncontrolled leak

During normal conditions from T = 0 s to T = 60 s the leakage amount is totaling to 0, therefore calibration and leakage assumptions made earlier are proven to be correct. The total leakage amount measured equals to 365ml while calculated from sensors it is around 350ml, which is in acceptable variation range. This data can also be visualized in terms of volume passed through the sensor. Figure 15 shows the same experiment visualized in a different way. This shows clearer picture when leak occurred and how much did it leak in terms of volume. The steeper the curve, the faster the flow. During the leak first sensor volume increases visibly, indicating the increased volumetric flowrate.

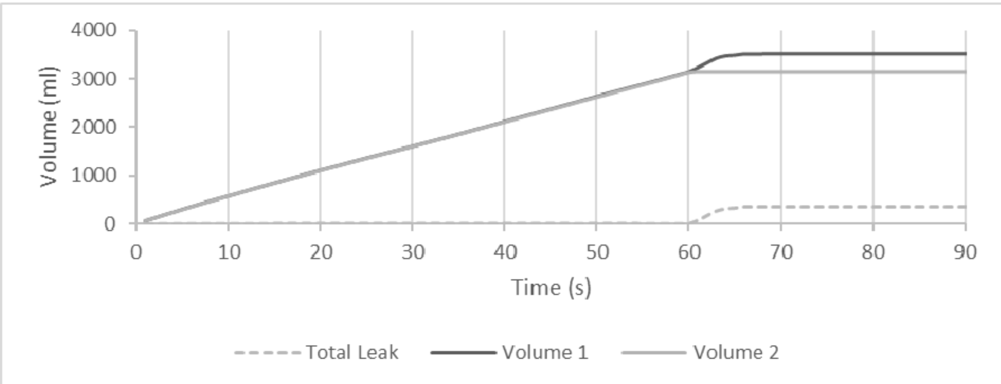



Fig. 15. Uncontrolled leak in terms of volume passed through sensors

### *7.2 Controlled leak by intervention*

In the final set of tests, the leak was introduced during the normal operation. Figure 16 shows the first test which was done by introducing leak at T = 40 s which was running up until T=82 s. The totaling leak reached 88 ml where real measured value was 90 ml, therefore once again the theory came close to practical values, implying that the sensors are calibrated correctly and assumptions made are helping. The intervention of switching the system off during this test occurred at T=83 s.

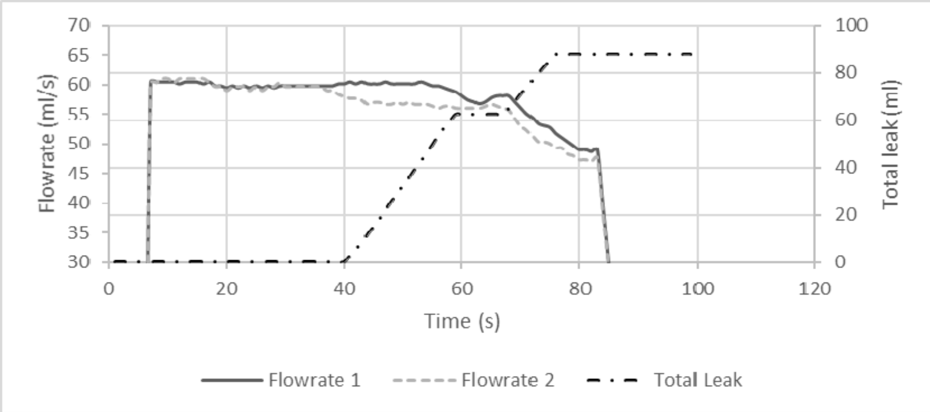



Fig.16. First test with controlled leak of 135ml/min

Second test was done using the same principle but with faster leakage rate. Figure 17 shows the sensor data where leak occurs at T=62 s and goes on till T=90 s. The totaling leak amount was measured to be 126 ml. Calculated value was off by 25ml of total leak - reaching 101ml. Intervention occurred at around T=85 s.

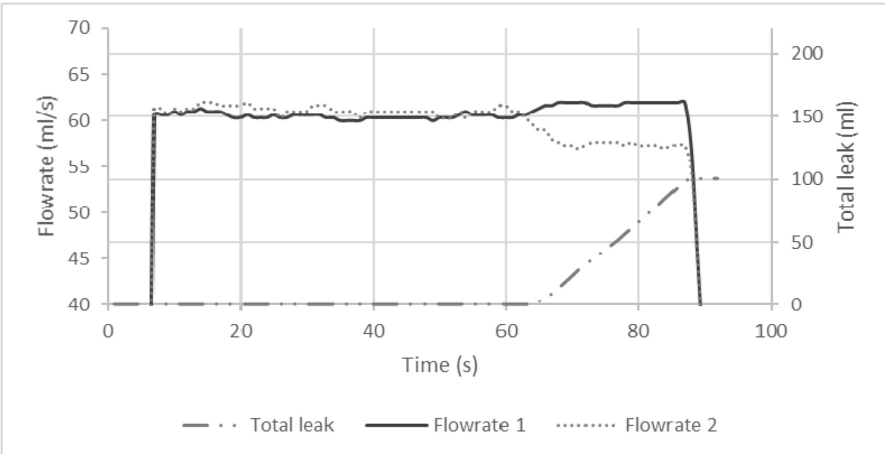



Fig.17. Second test with controlled leak of 250ml/min

Final test was done by implementing software leak detection system. The difference in flowrates was registered if calibrated values were differing more than user defined percentage. During this particular test, it was set to be 3%. If difference was higher than ~2ml per second, and if it persisted longer than 5 seconds, flow was redirected via bypass route. Leak was simulated at T = 45 s, which is clearly visible in Figure 18.

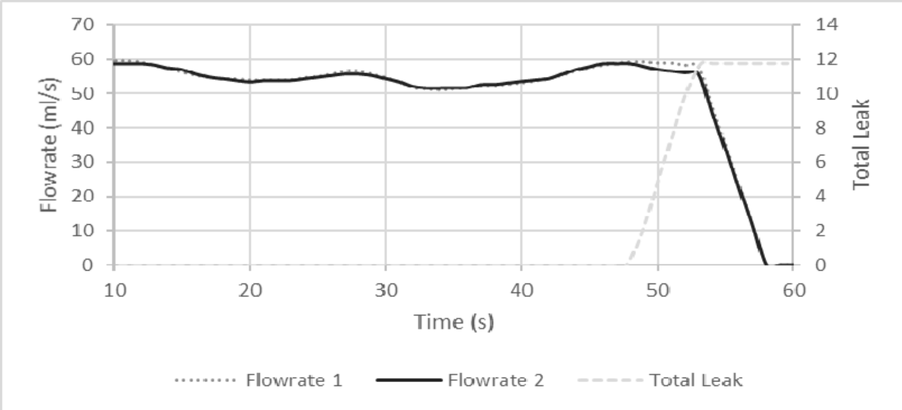



Fig.18. Simulated leak with software intervention

Server information dialogue also was programmed to display if the leak occurred and at what time did it occur. The leak scenario function sends the data to the microcontroller with commands that close SOV2 and SOV3 and opens SOV1 (refer to Fig. 8). It can also be programmed to shut the pump off if the leakage happens.

## 8. Analysis

System was designed and manufactured successfully. Local and remote terminals were developed and tested with working prototype. It was proven to be functional and useful as it remotely or locally monitors and controls the system built with help of physical layer including sensors and control elements. Cloud database was successfully created, and it was used by server and web application.

The leak detection technique system got validated by comparing sensor values. Three sets of tests were conducted to confirm the accuracy of the system. First set of tests shows how sensor data behaves in normal (no-leak) condition and how important was to calibrate the system. Second set was done with large amounts of leak introduced at specific time and finally a controlled leak with intervention of SOV control. The sensors were also calibrated to improve the accuracy and couple of assumptions made to decrease discrepancies in the calculations.

First set of tests show that leak amount in normal conditions can be negative as much as 1ml, indicating that liquid is added. Therefore, assumption that, leak amount cannot be negative. Other assumption was made that due to frequency measurement technique, pulse can fall in in current or following second resulting in small frequency spikes or drops. Assuming that these spikes result in 2ml/s of flowrate rise, calculated leaks lesser than 2ml will not be taken in calculations.

In second set of tests, the leak was introduced and no control was used. The sensor data shows that the leakage rate increased up until it reached the flowrate of first sensor, indicating that all the flow directly went to the leak. This means that second flowrate should be 0, and the experiment show that this is correct.

Final experiments were done by introducing leak and control. Different amounts of leak rate ranging from 100ml/min to 300ml/min were graphed and total amounts of leaks were compared to real measured values. The values compared changed from 2% to 20% indicating that it is precise, but future work is required to improve the accuracy. Software intervention was also implemented, featuring leak minimization using bypass route or system shutdown.

## 9. **Conclusion**

A system was designed, that could be used to monitor and control the building hydronic and other pipe systems. The system designed was closely related to SCADA system which is widely used in road infrastructure, factories, hospitals, educational and other buildings. System consists of multiple physical and software layers: sensors, microcontrollers, networking, servers, databases, cloud computing.

Proposed system was built and utilized with pipe system prototype. It was built using widely available copper tubing and compression fittings. Prototype was used to prove the methodology and to closely reproduce building pipe works. Pipeline feature control and monitor sensors: Solenoid Operated Valves (SOV) and Flowrate sensors. Wide variety of sensors and valves are available, but these were chosen based on cost, availability and reliability. SOVs are widely available and with right amount of current, it can control the flow sufficiently.

Microcontroller with relay module was used to control the components and receive the sensor data. Microcontroller was programmed to calculate the flowrate from raw sensor data in real-time and send it to the main server. Microcontroller can be adjusted to use different type of sensors such as pressure or temperature. It can be used for different purposes in SCADA system such as energy saving, or it can be used in the pipe system to improve the leak detection method accuracy.

Server and interface were developed to monitor and control the microcontroller. It receives and sends data to the microcontroller with control commands and receives the sensor data. Server or MTU was also sending data to cloud database which allows other devices to control and monitor the system. Database is protected by login password and username. Interface visualizes data in real-time, and allows user to select control source, either locally or remotely. Security concern over data transmission was not taken seriously as this is an early prototype, but for future work it can be improved as database password is a bare minimum-security integration.

Web application used in the SCADA system was configured to be remote user interface, with optional control features. This feature was programmed to be set locally by the main server interface. Since it is web application, all devices connected to the internet or local network can be used as remote supervisory and control unit.

## 10. Further work

There are more problems that needs to be solved for implementing the solution in the industry. Volumetric flow comparison technique needs to be improved for longer length pipelines. The liquid flowing through the pipes can have variations in flowrate, and the ‘wave’ of the flowrate will hit next sensor after long enough period of time and the software will consider it a leak. One of the solutions to this problem is multiple types of sensors. Pressure sensors would also improve accuracy and add other type of leak detection technique based on pressure wave detection. Rapid pressure drop is created when leak occurs in the pipeline, which travels upstream and downstream. The pressure against time can be compared between the sensors and the leak can be localized. This could be combined with volumetric flow comparison technique given more accurate values and more.

System can be improved and commercialized. The meters used were turbine type, meaning they are intrusive. Non-intrusive type of meters can be added to older buildings as they do not require change of the pipeline. Pressure sensors can be attached to the system providing more types of leak and other faults such as pressure drops and spikes. These sensors can also be combined and analyzed with machine learning for full automated solution with great accuracy.

Leak detection can be improved by adding software elements. Instead of checking the difference in flowrates and how much it is higher than user defined limit, dynamic index can be used. It can be programmed to set external value dependent on difference amount, difference length in time, and how fast is it changing. This external value could tell if there is a leak depending on how it could be calibrated. An example would be if the pipe burst and algorithm in use identifies the leak after set period of time this would prove inefficient way of detecting these leaks. But with external value, which would increase rapidly, the action could be committed earlier. This could lead to much quicker intervention on pipe bursting and it would improve the accuracy with not calibrated sensors.

## References


[1] "Water Leaks in Multi Storey Buildings - A Problem Bigger Than Panama Leaks!", Linkedin.com, 2019. [Online]. Available: https://www.linkedin.com/pulse/water-leaks-multi-storey-buildings-problem-bigger-than-hamid-khan. [Accessed: 25- Jun- 2019].

[2] Www.aqualeak.com, 2019. [Online]. Aqua leak water detection product list Available: https://www.aqualeak.com/32/flow-based-water-leak-detection/. [Accessed: 24- Jun- 2019].

[3] Www1.eere.energy.gov, 2019. [Online]. Available: https://www1.eere.energy.gov/buildings/publications/pdfs/corporate/pnnl-15149_market_assessment.pdf. [Accessed: 24- Jun- 2019].

[4] Energycontrolsonline.co.uk, 2019. 963 datasheet: [Online]. Available: https://energycontrolsonline.co.uk/pdfs/en-ta200636-uk0yr1115q.pdf. [Accessed: 24- Jun- 2019].

[5] V. Nguyen, Q. Tran and Y. Besanger, "SCADA as a service approach for interoperability of micro-grid plat-forms", Sustainable Energy, Grids and Networks, vol. 8, pp. 26-36, 2016. Available: 10.1016/j.segan.2016.08.001.

[6] "Damage Caused by Water Leaks", The Balance Small Business, 2019. [Online]. Available: https://www.thebalancesmb.com/damage-caused-by-water-leaks-462680. [Accessed: 24- Jun- 2019].

[7] "The Lifespan of Residential Plumbing Pipes", Santhoff Plumbing Company, 2019. [Online]. Available: https://santhoffplumbingco.com/the-lifespan-of-residential-plumbing-pipes. [Accessed: 24- Jun- 2019].

[8] "Boiler Piping Flow & Pipe Sizing | HVAC Hydronics", High Performance HVAC Heating & Cooling Reviews, 2019. [Online]. Available: https://highperformancehvac.com/boiler-piping-flow-pipe-sizing/. [Accessed: 24- Jun- 2019].

[9] Hitachi.com, 2019. Monitoring and Control Systems for the IoT in the Water Supply and Sewerage Utilities Hitachi Re-view [Online]. Available: https://www.hitachi.com/rev/archive/2017/r2017_07/pdf/p52-59_R7A03.pdf. [Accessed: 24- Jun- 2019].

[10] T. Chis: "Pipeline Leak Detection Techniques", Annals. Computer Science Series, 5th Tome, 1st Fasc. – 2007.

[11] Z. Jia, L. Ren, H. Li, S. Ho and G. Song, "Experimental study of pipeline leak detection based on hoop strain measure-ment", Structural Control and Health Monitoring, vol. 22, no. 5, pp. 799-812, 2014. Available: 10.1002/stc.1718.

[12] N. Saeed and M. Abbod: "Modelling Oil Pipelines Grid: Neuro-fuzzy Supervision System", International Journal of Intelligent Systems and Applications, vol. 9, no. 10, pp. 1-11, 2017.

[13] H. Fuchs and R. Riehle, "Ten years of experience with leak detection by acoustic signal analysis", Applied Acoustics, vol. 33, no. 1, pp. 1-19, 1991. Available: 10.1016/0003-682x(91)90062-j.

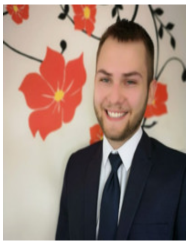
[14] K. Marmarokopos, D. Doukakis, G. Frantziskonis and M. Avlonitis, "Leak Detection in Plastic Water Supply Pipes with a High Signal-to-Noise Ratio Accelerometer", Measurement and Control, vol. 51, no. 1-2, pp. 27-37, 2018. Available: 10.1177/0020294018758526.

[15] J. Muggleton, M. Brennan and Y. Gao, "Determining the location of buried plastic water pipes from measurements of ground surface vibration", Journal of Applied Geophysics, vol. 75, no. 1, pp. 54-61, 2011. Available: 10.1016/j.jappgeo.2011.06.030.

[16] IoT history: Cs.cmu.edu, 2019. [Online]. Available: https://www.cs.cmu.edu/~coke/history_long.txt. [Accessed: 25- Jun- 2019].

[17] Y. Kang, M. Han, K. Han and J. Kim, "A Study on the Internet of Things (IoT) Applications", International Journal of Software Engineering and Its Applications, vol. 9, no. 9, pp. 117-126, 2015. Available: 10.14257/ijseia.2015.9.9.10.
[18] D. Bastos, M. Shackleton, and F. El-Moussa, "Internet of Things: A survey of technologies and security risks in smart home and city environments", Living in the Internet of Things: Cybersecurity of the IoT 2018 , IET, 2018.
[19] Z. Bakhshi, A. Balador and J. Mustafa, "Industrial IoT security threats and concerns by considering Cisco and Microsoft IoT reference models," 2018 IEEE Wireless Communications and Networking Conference Workshops, WCNCW 2018, 2018.
[20] D. Nguyen, C. Song, Z. Qian, S. V. Krishnamurthy, E. J. M. Colbert and P. McDaniel, "IoTSan: Fortifying the safety of IoT systems", International Conference on Emerging Networking Experiments and Technologies (CoNEXT '18), Greece, 2018.
[21] M. Turkowski, A. Bratek and M. Slowikowski, "Methods and Systems of Leak Detection in Long Range Pipelines", Journal of Automation, Mobile Robotics & Intelligent Systems vol. 1 no. 3, 2007.
[22] YF-S201 sensor datasheet, Hobbytronics.co.uk, 2019. [Online]. Available: https://www.hobbytronics.co.uk/datasheets/sensors/YF-S201.pdf. [Accessed: 25- Jun- 2019].
[23] YF-B1 Sensor Datasheet, Media.digikey.com, 2019. [Online]. Available: https://media.digikey.com/pdf/Data%20Sheets/Seeed%20Technology/114991171_Web.pdf. [Accessed: 25- Jun- 2019].

**Authors' Profiles**



**Audrius Urbonavicius** received BEng degree in Electrical and Electronic engineering from University of West London. He is currently working as Operations Engineer in video analytics and IT engineering sector.

**Dr. Nagham Saeed** received her Ph.D. degree in optimised wireless communication networks from Brunel University, U.K., in 2012. She is currently a Senior Lecturer in Electrical Engineering with the School of Computer and Engineering, University of West London, UK. She has authored a respectable number of papers in journals and refereed conferences. Her current research interests include intelligent systems for modelling and classification to control industry operations or communication networks such as Mobile Ad Hoc NETworks (MANET) or Vehicular Ad hoc NETworks (VANETs) with the support of Artificial Intelligence (AI) , Machine learning (ML), and/or Industry Internet of Things (IIoT) in the process. She is a member Intelligent Sensing and Vision (IntSaV) and the Sensing, Localisation and Contextual Data Analytics (SENLOCDA) research groups.

**How to cite this paper:** Audrius Urbonavicius, Nagham Saeed. " IoT Leak Detection System for Building Hydronic Pipes ", International Journal of Engineering and Manufacturing(IJEM), Vol.9, No.5, pp.1-21, 2019. DOI: 10.5815/ijem.2019.05.01