Parallel Redundancy System for Critical Conditions Monitoring and Alerting

Original Scientific Paper

Goran Horvat
Josip Juraj Strossmayer University of Osijek,
Faculty of Electrical Engineering, Department of Communications
KnezaTrpimira 2b, Osijek, Croatia
goran.horvat@etfos.hr

Drago Žagar
Josip Juraj Strossmayer University of Osijek,
Faculty of Electrical Engineering, Department of Communications
KnezaTrpimira 2b, Osijek, Croatia
drago@etfos.hr

Davor Bogdanović
Josip Juraj Strossmayer University of Osijek,
Faculty of Electrical Engineering, Department of Communications
KnezaTrpimira 2b, Osijek, Croatia
davor.bogdanovic@etfos.hr

Abstract *– Monitoring critical conditions is of outmost importance in any system for achieving long life and stability. In this process, various parameters can be classified as critical and their values must be kept within a bounded interval by means of monitoring and acting upon a change in the value. A practical example of critical conditions monitoring is temperature monitoring in data centers (server rooms) where the temperature value must be kept below a certain threshold in order to achieve long life and stability of equipment. This paper presents a system designed for monitoring temperatures and alerting of their critical values is proposed - PRSMA. With a parallel redundancy feature that guarantees high reliability of the proposed solution, this approach achieves timely alerting upon critical condition, real-time supervision of temperature values and forecasting of critical conditions. The redundancy aspect is realized by using a mobile operator link alongside with the Internet-based landline connection to a cloud-based service – the Internet of Things concept. The proposed architecture is tested in laboratory conditions and the advantages of this approach are shown through measurement and testing.*

Keywords *– alerting, cloud computing, critical conditions monitoring, embedded system, Internet of Things, temperature monitoring*

1. INTRODUCTION

With the rapid development of microelectronics and component cost reduction, today's consumer electronics is becoming omnipresent in the sense that in 2011 the number of devices on the planet exceeded the number of people on the planet [1]. With this rising trend of consumer electronics, the supporting systems such as data centers are growing as well, as the need to interconnect these devices is becoming intensely expressed. This yields another important issue, i.e., energy efficiency. With an increase in computational power and functionality of data centers, the overall energy consumption is increasing as well resulting in the need to constantly cool the rooms where this equipment is located. Also, from the fact that cooling accounts for 30%- 50% of total energy consumption of data centers [2], it is safe to assume that hazardous conditions are likely to occur in case of cooling system failure. This can result in component/system failure, or in the worst case scenario, a fire [3]! On the other hand, it must be noted that modern and advanced data centers have already incorporated safety measures in the form of system shutdown, backup cooling and fire control systems; however, the topic of this paper is oriented towards a large number of small data centers (server rooms, equipment cabinets, etc.), often not equipped with high reliability and advanced climate control (cooling) systems. Furthermore, the proposed system could be applied to various other domains such as monitoring of wine production [4], measurement or radioactive material leakage [5], and similar applications where monitoring of various parameters is of outmost importance [9].

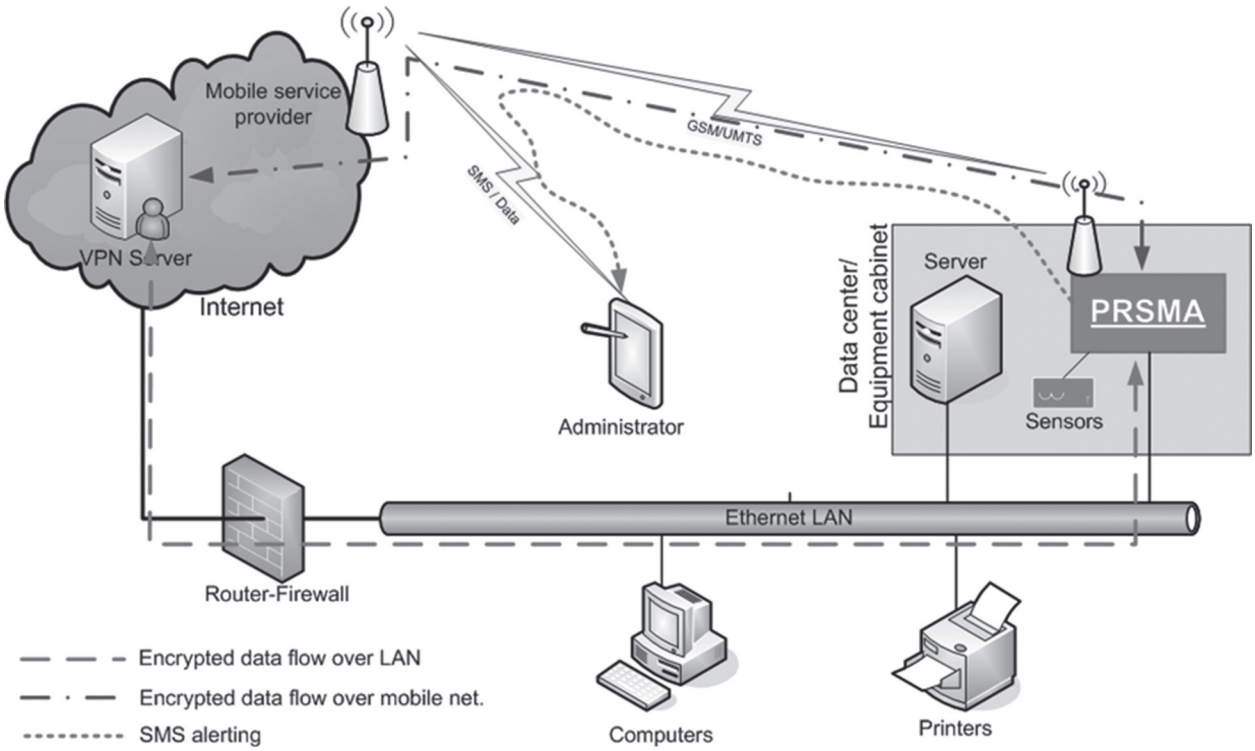

Fig. 1. Architecture of the proposed parallel redundancy system for monitoring and alerting.

Within this scope, this paper presents the development of a parallel redundancy system for critical conditions monitoring and alerting (PRSMA) designed as a low-cost but high-reliability device, intended for use in small server rooms, data centers, equipment cabinets, etc. The main reason behind the parallel redundancy alerting system is the fact that upon critical conditions such as high temperature or high humidity the communication infrastructure becomes unstable and it is impossible to establish a reliable communication channel through an Ethernet-based link. Consequently, a hot reserve system is proposed to establish communication in any condition. The system is based on a GSM communication module, alongside with Internet-based LAN communication. By proposing this parallel redundancy architecture the probability of a single point communication failure (SPOF) is reduced. Furthermore, the ability of using a GSM mobile network enables instantaneous alerting via SMS or phone call to the operator in charge of the equipment at hand [10].

The paper is organized as follows: Section 2 describes the basic system architecture of the proposed system, placing emphasis on the parallel redundancy aspect in general. Section 3 shows the developed prototype alongside with measurements and Section 4 gives the conclusions.

2. ARCHITECTURE OF THE PROPOSED SYSTEM

The proposed Parallel Redundancy System for Monitoring and Alerting (PRSMA) is based on two communication technologies, i.e., landline Internet connection (via DSL, Cable, Fiber, etc.) and wireless mobile services (GSM, UMTS), to achieve high data delivery reliability towards the end user (system administrator or person in charge). The architecture of the proposed system is shown in Figure 1.

As seen in Figure 1, the PRSMA device is located within the observed area (area of interest) alongside with dedicated sensors. The main task of the PRSMA is to monitor various parameters (temperature, humidity, water level, smoke, etc.) from dedicated sensors and process sampled data. The processed data is then compared to a set of predefined rules (constraints that can be modified) in the process of estimating critical conditions. Upon the detection of a critical condition (e.g., the temperature value exceeds the maximum predefined value), the administrator is notified of a critical condition occurring within the observed area. Also, processed parameter values are continuously delivered to the Virtual Private Server (VPS) where real-time monitoring can be performed [11].

In order to achieve high reliability of the proposed PRSMA, the overall architecture incorporates three data flows:

- Encrypted data flow over LAN,
- Encrypted data flow over mobile network, and
- SMS alerting.

Encrypted data flow over LAN is used for data exchange between the Virtual Private Server (VPS) Cloud service and the PRSMA using landline communication, where the existing infrastructure is used for data communication. In the event of a landline network failure, a hot swap is realized using the second data flow (over mobile network) and the connection towards the VPS

server is preserved. In the case of a VPS failure or Internet inaccessibility, an SMS based alerting system is activated and the end user is notified of current conditions within the area of interest.

2.1. PRSMA DEVICE

The PRSMA device is a cost-effective solution based on a low-power microcontroller and communication modules. A block diagram of the PRSMA device is shown in Figure 2.

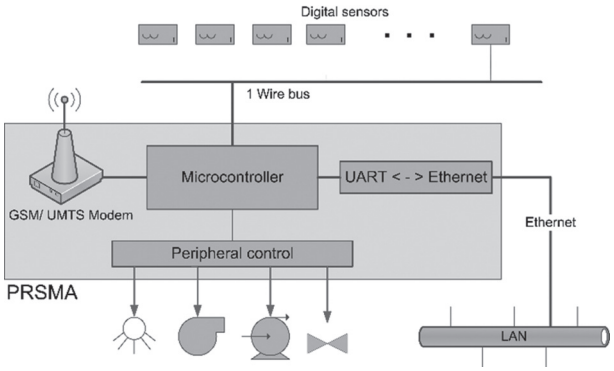

Fig. 2. Block diagram of the PRSMA device.

As seen in Figure 2, the main processing unit of the PRSMA is an 8-bit microcontroller capable of handling all tasks related to processing of data and communication. The most important task of the PRSMA is data acquisition from various sensors to provide the full picture on the monitored environment.

Due to the fact that data transmission from sensors to the PRSMA must be highly reliable, a wired approach was chosen using a 1-Wire communication bus. The advantage of using the 1-Wire communication bus is the ability of interconnecting a large number of sensors using only 3 conductors. As all devices are uniquely identified by a 64bit serial number, there is no need to assign sensor ID numbers and the sensors can be easily identified [6]. Consequently, today a large number of sensors use 1-Wire communication as a standard (e.g., a DS18B20 temperature sensor) enabling omnipresent interoperability [7]. Finally, with the availability of 1-Wire ADC and 1-Wire GPIO port expander modules it is possible to design any custom sensor, having ease of implementation in mind.

Due to the fact that the PRSMA has the ability to send values of 10 sensors simultaneously, the amount of data transmitted is constant for 1 or 10 sensors, as the unused sensor slots are occupied with blank values. Also, due to the fact that sensors are located on the 1-Wire bus, all ADC conversion and sampling is performed simultaneously (distributed among sensors), thus the received sensor value is a true instantaneous value of the dedicated sensor.

Next, a very important aspect of the PRSMA is the parallel redundancy communication system composed of two separate communication modules, i.e., the GSM/UMTS Modem and the TCP/IP Ethernet module (UART – Ethernet). Each module has a required protocol stack for communication integrated, thus there is no need to integrate TCP/IP or GSM/UMTS stacks within the microcontroller. This simplifies system design as the only communication protocol needed is a UART. Both modules also handle connection to VPS by means of the TCP socket or HTTPD connection [11]. Out of the aforementioned protocols, we chose to use the TCP socket due to the fact that by having a constantly opened TCP tunnel it is possible to monitor the functionality of the device by using keep-alive settings on the TCP tunnel [12].

Finally, in order to act upon any critical condition, the PRSMA incorporates a peripheral control unit with the task of controlling the resulting critical conditions. These include A/C control in the form of IR or relay signals, fan or turbine control for ventilation, pump control for water draining, etc.

2.2. COMMUNICATION PROTOCOL

As stated before, the means of transmitting data to the VPS server is achieved by using a TCP socket, both from the Ethernet network and the GSM/UMTS network simultaneously. This establishes a hot reserve in case one of the communication channels is severed. Due to the fact that TCP sockets are kept open via keep-alive packets, it is possible to constantly monitor for broken connections or device failures. In the process of establishing the TCP tunnel, the PRSMA acts as a TCP client and connects to a static IP address of the VPS server. In this way, the need for a dynamic DNS or other services for identifying the IP address of the PRSMA is avoided. This also reduces the possibility for malicious users to connect to the PRSMA and manipulate peripheral devices. However, additional security aspects must be considered on the VPS, as exposed sockets are available publicly on the Internet.

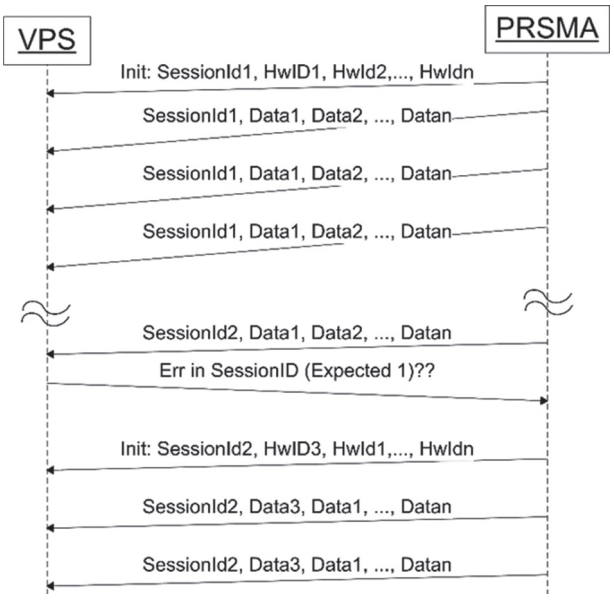

Fig. 3. TCP Socket data flow – handshaking and exchanging of hardware IDs.

In order to achieve communication between the PRSMA and the VPS, the approach of using sessions for data exchange is used. The main reason for using sessions is the need to map sensor data with its hardware serial number (Hardware ID). Due to the fact that hardware ID is a 64bit unique number, continuous transmission of hardware ID numbers together with the data would result in the generation of unnecessary network traffic. The proposed approach uses the technique of mapping hardware ID numbers with positions of data in each session, thus eliminating the need to send hardware IDs each time. The proposed method is shown in Figure 3.

As seen in Figure 3, upon device initialization the PRSMA sends an *Init* packet containing the session ID (incremented from the last session) and the associated hardware IDs of each sensor connected to the PRSMA. Accordingly, in the forthcoming data packets the position of each data sample is mapped to the position of the hardware ID within the *Init* packet. As long as the Session ID is unchanged, the received data samples will be mapped to the *Init* packet with the same Session ID. For the sake of the argument (Figure 3, a lower portion of the graph), if during normal operation an event causes the device to reset (due to power failure, a faulty sensor, etc.) and the Session ID changes, the VPS will receive a data packet with the Session ID that differs from the saved Session ID of the last *Init* packet. If this occurs, the VPS will transmit an error message to the PRSMA requesting an update on the *Init* packet. The PRSMA will respond with a new *Init* packet, mapping new positions of sensors within the forthcoming data packets. In this manner, mapping of the data to sensor Hardware IDs is guaranteed.

2.3. SECURITY ASPECTS AND DATA ENCRYPTION

A very important aspect of establishing communication over the Internet is the security aspect. As the Internet is a public domain, all devices connected to the Internet are vulnerable to attacks from malicious users. Consequently, to maximally reduce the possibility of unauthorized access, the PRSMA uses several mechanisms: first of all, the chances of direct intrusion are minimized by implementing a TCP client connecting only to a designated IP address (VPS). Next, each packet is time-stamped and any duplicate packet received at either side is automatically discarded. This ensures protection against replay attacks. Furthermore, each packet is controlled by a CRC checksum in the footer, ensuring the integrity of data.

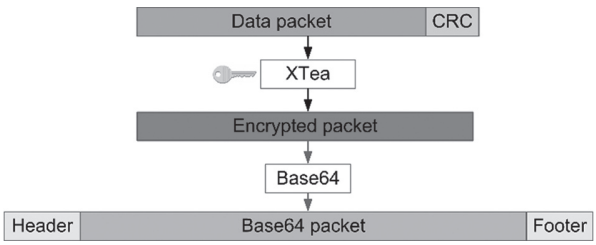

Fig. 4. Data packet encryption and encapsulation before transmission via the TCP socket.

Finally, to ensure data confidentiality over the Internet, all data packets are encrypted by a lightweight encryption cipher. The process of preparing data packets for transmission is shown in Figure 4. After the packet is time-stamped and the CRC checksum is calculated, the packet is encrypted by using a lightweight XTEA (eXtended Tiny Encryption Algorithm) cipher. The best cryptanalysis for XTEA is a related key differential attack that can break 32 out of 64 rounds of XTEA, requiring $2^{20.5}$ chosen plaintexts and time complexity of $2^{115.15}$, which presents a cryptographically safe algorithm [8]. Due to the fact that after encryption the data stream can contain all values [0, 255] for each byte (not just ASCII text), the obtained data are encapsulated by using the Base64 algorithm, resulting in a standard textual string, with a header and footer added. After the packet is received at the VPS, the Base64 is converted to raw byte data and decryption is preferred using the same key. The decrypted data is checked for checksum validity and if the CRC checksum is verified, the data is forwarded as a valid data packet to higher layers for processing. The process of sending data from the VPS is reverse.

The aforementioned ensures confidentiality, integrity and timeliness of the data as the TCP socket incorporates retransmissions and flow control, together with the parallel redundancy channel.

3. TESTBED AND MEASUREMENT OF THE PROPOSED SYSTEM

The proposed system was developed as a standalone solution using a low-cost microcontroller, Ethernet LAN and GPRS communication modules and an LCD display for a user interface. A microcontroller chosen for the function of a processing unit was the ATMega32 microcontroller from Atmel running at 11.0952 MHz to ensure synchronous work with the UART standard. The choice of the aforementioned microcontroller is omnipresence and a low cost of the integrated circuit at hand. For external sensors, a simple digital temperature sensor DS18B20 was used to measure a single temperature value and a relay for A/C control. The developed PRSMA device is shown in Figure 5.

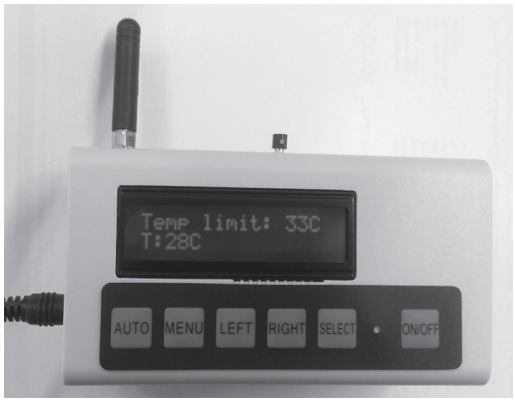

Fig. 5. The developed PRSMA device in the laboratory testing environment.

Experiment methodology is as follows: The testing of the proposed system was performed in laboratory conditions in order to estimate the ability of the PRSMA to act upon critical conditions. The PRSMA device was stationed in a server room with climate control. The device samples a temperature value from a temperature sensor. The data is sampled every 2 seconds and an average of 10 measurements is calculated. The average value is sent from the PRSMA device towards the VPS every 20 seconds. The measured parameter (temperature) was monitored for a time period of 36 hours. The temperature data shown at the VPS can be seen in Figure 6.

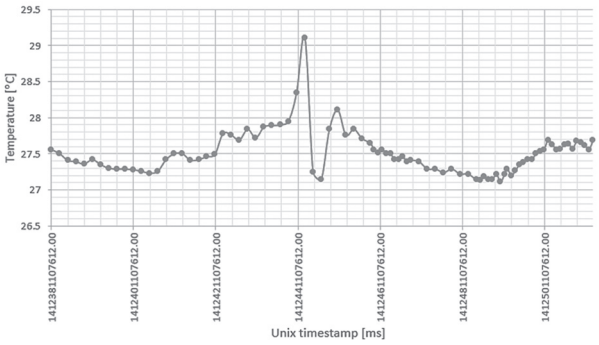

Fig. 6. Example of temperature measurement in a period of 36 hours

In the observed time period one disturbance was introduced upon which the Air Conditioning device was turned off – the simulated failure. When the temperature value exceeds the maximum value (predefined rule), the critical condition event is initiated upon which the administrator is notified by SMS and the AC system is turned back on by using a relay. The temperature is normalized after a certain period of time. The temperature curve depends on the type of the AC system, accounting for the present temperature undershoot (Figure 6).

Based upon the performed measurements it can be concluded that the real time monitoring component of the system performs as expected. Furthermore, the experiment also demonstrates the simulated system failure when the cooling system of a server room malfunctions. The PRSMA acts by controlling the auxiliary cooling system (AC system) and returns the atmospheric conditions to normal.

3.2. NETWORK COMMUNICATION ANALYSIS

A network communication segment was investigated by analyzing network traffic intensity. Figure 7 depicts the measurement data throughput of the PRSMA device in normal operating conditions. The data throughput was measured on a LAN communication segment, with the use of a network tap. The tap forwarded the data onto a PC, where a network packet analyzer (Wireshark) was used to sort and sniff the packet stream. The aforementioned resulted in a data throughput diagram shown in Figure 7.

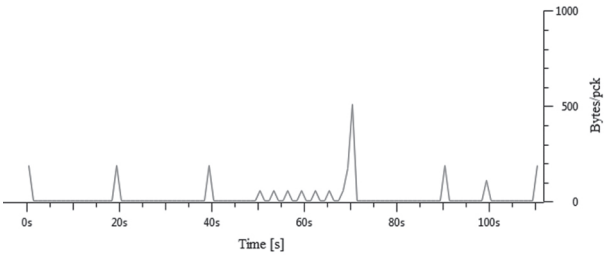

Fig. 7. Measuring data throughput – IO diagram (*NO – normal operation, RST – reset, SID – session initialization*)

Figure 7 shows an IO graph for normal operation (0s – 40s) - NO, reset condition of a device (55s – 65s) - RST, retransmission of *Init* packet with a new Session ID (65s) - SID and further normal operation - NO.

In the aforementioned diagram it is visible that upon normal data transmission (each Δt = 20s) for each data transmission approximately 200 bytes are transmitted. This measurement is based on transmitting the data from 10 sensors on one PRSMA device (one active sensor and nine dummy sensors). This results in an average throughput of 80 bit/s, representing insignificant network load compared to the maximum data rate of the LAN network (100 Mbit/s or higher). Furthermore, it is possible to calculate the monthly data usage of the PRSMA system based on the sampling time (Δt):

$$I[month] = \frac{518.4}{\Delta t} \quad (1)$$

If calculated on a monthly basis, the total data usage of 26 MB is expected from one PRSMA, representing the negligible cost of data.

In the case of the reset condition and the need to retransmit *Init* packet with Hardware IDs, the amount of data transmitted increases approximately 3 times or more (700 bytes), which is expected due to the fact that all 64-bit hardware IDs need to be retransmitted. This can be seen in Figure 7, data spike on 70s. Afterwards, normal operation resumes and data sample packets are transmitted every Δt.

4. CONCLUSION

This paper proposes an architecture for establishing a highly reliable and cost-effective system for monitoring critical conditions in various applications, i.e., the PRSMA. The system is based on achieving parallel redundancy using a landline channel to the Internet (DSL, Cable, etc.) and a mobile operator (GSM/UMTS) communication channel, together with the ability to send critical event alerts via SMS. The system uses a cloud-based Virtual Private Server (VPS) in order to concentrate data flows from both communication channels and enable real-time data presentation and overview. To achieve data confidentiality over the public domain such as the Internet, a security mechanism is implemented in form of encryption and reply attack

protection. The need to install expensive server solution is avoided by using a VPS solution.

Furthermore, the proposed PRSMA system has the ability of sampling standard 1-Wire sensors on a single bus (up to 10 sensors), thus enabling interoperability with various sensors available on the market. Also, by incorporating peripheral control, the ability of controlling systems such as Air Conditioning, fans, pumps, etc. is enabled. By using cost-effective components, a low-power microcontroller, and standardized communication modules, the PRSMA device is a low-power and low-cost solution for use in various applications.

Testing and measurements of the PRSMA device performed in laboratory conditions (testbed of a single temperature sensor) shows that the system performs as expected upon the occurrence of the critical condition, and by actuating the Air Conditioning unit, the rise in the temperature beyond the maximum value (a predefined set of rules) is controlled. Finally, by investigating network traffic intensity it was concluded that sending one data sample from the PRSMA uses approximately 200 bytes, which results in monthly data usage of 25 Mbytes for normal operation (for sampling time of $\Delta t = 20$ s) . In the case of the need to retransmit the *Init* package with associated Hardware IDs of sensors (in the case of device reset), the amount of data is tripled per one request. However, due to small frequency of the *Init* packet, this approach reduces overall data consumption (compared to the approach of sending Hardware IDs together with sensor sample data) and enables a cost-effective, secure and very reliable communication system.

Future work on this topic includes the investigation of transients between mobile and landline communication in case of communication failure. This is required to estimate the reliability of the communication itself and present reliability analysis of the system as a whole. Furthermore, in case of mobile communication failure, the administrator must be informed of a possible critical event by alternative means (other than SMS). A timely notice on critical events via push notifications or a similar technology will also be investigated, alongside with reliability analysis of using alternative means. Integration of the PRSMA system with mobile based applications (iOS, Android, etc.) is also a good approach to achieving a user-friendly interface for the PRSMA architecture.

5. REFERENCES

[1] J. Gubbi, R. Buyya, S. Marusic, M. Palaniswami, Internet of Things (IoT): A vision, architectural elements, and future directions, Future Generation Comp. Syst. Vol 29, No. 7, 2013, pp. 1645-1660

[2] H. Zhang, S. Shao, H. Xu, H. Zou, C. Tian, Free cooling of data centers: A review, Renewable and Sustainable Energy Reviews, Vol. 35, July 2014, pp. 171-182, ISSN 1364-0321.

[3] T. Lu, X. Lü, M. Remes, M. Viljanen, Investigation of air management and energy performance in a data center in Finland: Case study, Energy and Buildings, Vol. 43, No. 12, December 2011, pp. 3360-3372, ISSN 0378-7788.

[4] L. Boquete, R. Cambralla, J.M. Rodríguez-Ascariz, J.M. Miguel-Jiménez, J.J. Cantos-Frontela, J. Dongil, Portable system for temperature monitoring in all phases of wine production, ISA Transactions, Vol. 49, No. 3, July 2010, pp. 270-276, ISSN 0019-0578.

[5] F. Ding, G. Song, K. Yin, J. Li, A. Song, A GPS-enabled wireless sensor network for monitoring radioactive materials, Sensors and Actuators A: Physical, Vol. 155, No. 1, October 2009, pp. 210-215, ISSN 0924-4247.

[6] D. Eisenreich, B. DeMuth, Chapter 10 - 1-Wire Basics for TINI, In: Embedded Technology, edited by Dan Eisenreich and Brian DeMuth, Newnes, Burlington, 2002, pp. 345-431, Designing Embedded Internet Devices, ISBN 9781878707987.

[7] DS18B20 1-Wire Digital Temperature Sensor, accessed October 2014, available at: www.datasheets.maximintegrated.com/en/ds/DS18B20.pdf

[8] G. Horvat, D. Zagar, G. Martinovic, STFTP: Secure TFTP Protocol for Embedded Multi-Agent Systems Communication, Advances in Electrical and Computer Engineering, Vol. 13, No. 2, 2013, pp. 23-32.

[9] N.M.S. Hassan, M.M.K. Khan, M.G. Rasul, Temperature Monitoring and CFD Analysis of Data Centre, Procedia Engineering, Vol. 56, 2013, pp. 551-559, ISSN 1877-7058.

[10] T. S. Ueng, Z. D. Tsai and J. C. Chang, SMS alert system at NSRRC, 2007 IEEE Particle Accelerator Conference (PAC), Albuquerque, NM, 2007, pp. 401-403.

[11] B. Cheng, X. Cheng, J. Chen, Lightweight monitoring and control system for coal mine safety using REST style, ISA Transactions, Vol. 54, January 2015, pp. 229-239,

[12] R.O. Ocaya, A framework for collaborative remote experimentation for a physical laboratory using a low cost embedded web server, Journal of Network and Computer Applications, Vol. 34, No. 4, July 2011, pp. 1408-1415, ISSN 1084-8045.