Uniport Journal of Engineering and Scientific Research (UJESR)

Vol. 6, Issue 1, 2021, Page 01-10
ISSN: 2616-1192

(www.ujesr.org)

Development of State Transition Table Based Remote Monitoring of Process Control System

Chiagunye, T*[1], Okozor, N[1], Okey, D.O[2] and Ilo, S.F[3]
[1]Department of Electronic and Computer Engineering, Nnamdi Azikiwe University, Awka
[2,3]Department of Computer Engineering, Michael Okpara University of Agriculture, Umudike
*Corresponding author's email: theodore.tochukwu@mouau.edu.ng

Abstract
There is urgent need for industrial managers to monitor a real – time process without being there especially in this age of high-tech mobility. In ancient times, there was mobility but the difference today is the tech advancement. This paper develops a novel method of remote monitoring and process control based on state transition table (STT). An algorithmic state machine (ASM) chart implemented in a microcontroller was integrated with a monitoring software developed to enable a graphics user interface (GUI) visualization of the process status. The proposed system enables users to control a process and remotely monitor the controlled process. In addition, the microcontroller sends a short test message (SMS) through a Global System for Mobile Communication GSM modem to any desired remote site. The system was evaluated using a real-time incubator and was found to be cost effective and offers easy usability to the end users compared to existing systems achieving an accuracy of about 78% in performance. Hence, the developed system finds application in industrial process monitoring.

Keywords: *ASM, GSM, GUI, Microcontroller, State transition*

Received: 26th May, 2021 *Accepted: 6th August, 2021*

1. Introduction

The trend these days is for industrial control system to be made as user friendly as possible. Industrial process control managers like to see an integration of user- friendliness and intelligence (Hawryszkiewyez, 1989). These would enable them monitor process progress, material usage and down –time from any location. The process control industry is therefore moving towards the development of user centric intelligent systems that put managers in firm control of their processes (Chivilikhin, 2019). Industrial managers of real-time systems need to monitor the state of the process under control remotely especially in this age of mobility (Ayad et al., 2013). Industries today engage real – time technologies to observe exact events in the process control systems without on-site presence. This requires a graphic user interface (GUI) that can adequately represent the process under control at the remote site where the real – time event is displayed (Fränzle, 2019).

In the work of Alghamdi (2019), it was stated that everyone wanted things to happen at the click of a button noting that all organizations are moving out of the traditional processes to more scientific processes of monitoring. This transition includes elimination of most of the manual processes to get more accurate results and the job done in more efficient manner (Czerwinski and Kania, 2016). With real time monitoring one can determine what is happing in system with respect to performance issues or problem reports. Real – time monitoring includes statistics that represent the current activity on the system that can help determine usage patterns and resource allocation and identify problems areas (Salauyou and Bulatowa, 2018).

According to Gajalakshmi et al. (2020), an optimized ASM routing algorithm for cognitive radio networks based on the parity detection algorithm for cognitive radio networks (CRN) was implemented. The developed system uses ASM technique to determine a user-defined parity check between CRNs. Chivilikhin et al. (2020) proposed a framework based on automatic synthesis methods that learns the behaviour of an existing legacy programmable logic controller (PLC) and generates state machines that can be incorporated into IEC 61499 function blocks. The system accepts error-free traces and synthesizes a modular controller that may be distributed across several physical devices but has high cost of implementation.

Among all the literature, no one considered the cost effectiveness of the developed system and the user-friendliness except the use of GUIs. Also, the developed systems in the reviewed works (Ayad et al., 2013) could not accommodate new systems through a plug and play mechanism. In this paper, a system that has plug and play feature maintaining cost effectiveness has been developed, which will enable it to be customized for any industrial process control systems that can be represented as an Algorithmic State Machine (ASM) chart.

2. Materials and methods

The Algorithmic State machine method used comprised of the following steps:

- The drawing of ASM chart for the process
- Transforming the ASM chart into state transition table (STT) Expanding the state transition table fully
- Burning the location address and the content address of fully expanded STT into microcontroller.

2.1 Design approach

The top – down design approach for system development was employed. The system was broken into subsystems from the topmost level to the lowest level. Each subsystem was translated into a program module and the various program modules into sub-routines, functions and simple programming statement. The state transition table based remote monitoring of process control system is a supra – system which was further broken down into component systems as illustrated in Fig. 1.

Fig. 1: Supra System Component

Control unit

This was made up of microcontroller – based module that receives its instruction from different input unit, examines the signals and processes the information after which it sends the information to the mobile station.

Mobile station

In this work, two different GSM modems were used. The first GSM modem is based on the specific area where the control system is located. The modem receives the SMS message which it stores in the SIM memory. The microcontroller extracts it and processes accordingly to carry out specific operation, while the second modem is used to receive SMS message for the GUI unit.

GUI unit

This section displays the information by the mobile station graphically for clear understanding of the user. In order words the status condition of the process under control are being display at this section.

2.2 Block diagram of the envisaged system.

The block diagram of the developed system is shown Fig. 2. The data acquisition system monitors the plant under control and sends the output pattern as an input to the microcontroller. The microcontroller analyzes the data and send the information in form of SMS to the modem attached to it which then forwards the same message to another modem that display the information to the GUI remotely. The microcontroller as well activates the output port that affect the control of the plant under control. The analysis and electronic control of the system was achieved using the intelligent agent embedded into the control software in the microcontroller.

2.3 System design

2.3.1 The system algorithm

Here a given environment is being monitored to see that temperature is within a certain range. For example, if the temperature is low a heater is turned ON to heat up the environment to a desired level, but if the temperature is high a cooler is then turned ON to cool the environment. In a situation where it happens that the temperature is outside the range, an error signal in form of alarm will occur and a reset button will be used to bring it to normal.

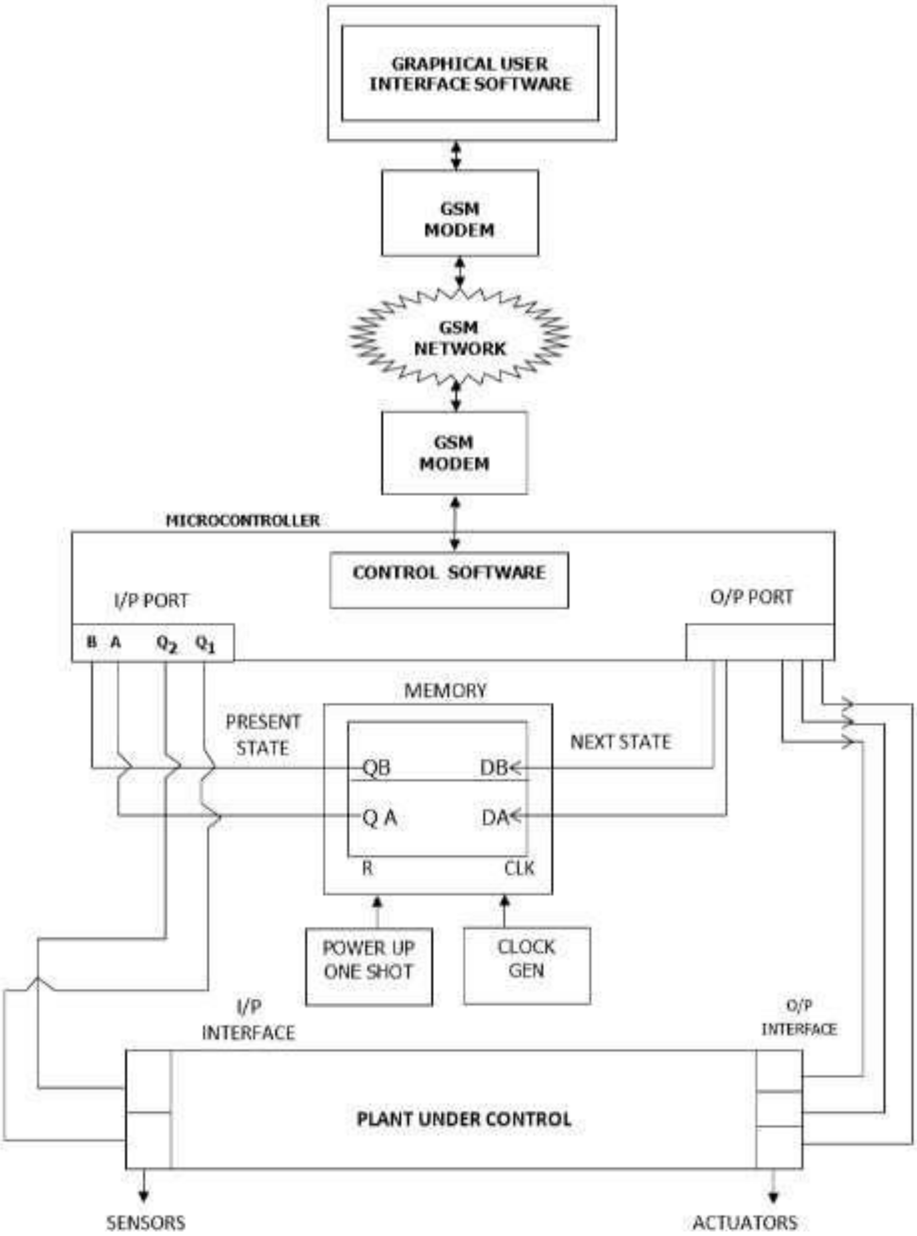

Fig. 2: Block diagram of the remote monitoring process control system

2.3.2 The ASM chart

Fig. 3 shows the ASM chart of the temperature process control system. The ASM chart comprises two symbols, namely the rectangular boxes or state and the decision boxes or qualifiers. The labels or names inside the rectangular boxes are the state outputs, while the labels inside the decision boxes are regarded as the input qualifiers. For this example, the ASM chart has two outputs which are HHEATER (when the temperature is high) and HERROR. The logic level of the output signal is high or active when the control system is in that state. The bit pattern at the top right end of the state box is its state code. The letter B, A of the ASM chart signify the two flip flops B and A that are used to represent the various states of the machine.

The state code is the logical levels at the Q outputs of these two flip flops respectively. Each rectangular box in the ASM charts is a state box. The symbol ST0 enclosed in a circle at the bottom left-hand corner of a state box is the state name. Here ST0 stands for state 0, and similar interpretations apply to the other states. Hence, ST1 means state 1 and so on. Each decision box has one entry path and two exit paths. The exact value of an input qualifier determines which exit path is followed out of a decision box. In the ASM chart of Fig. 3, L_{th}, U_{th} and RST are the input qualifiers. With the help of a K-map of Table 1 in which the state names are inserted serially in an adjacent cell and which is called a state map, the state codes are chosen such that only one-bit changes level as one moves from one state of the control system to another.

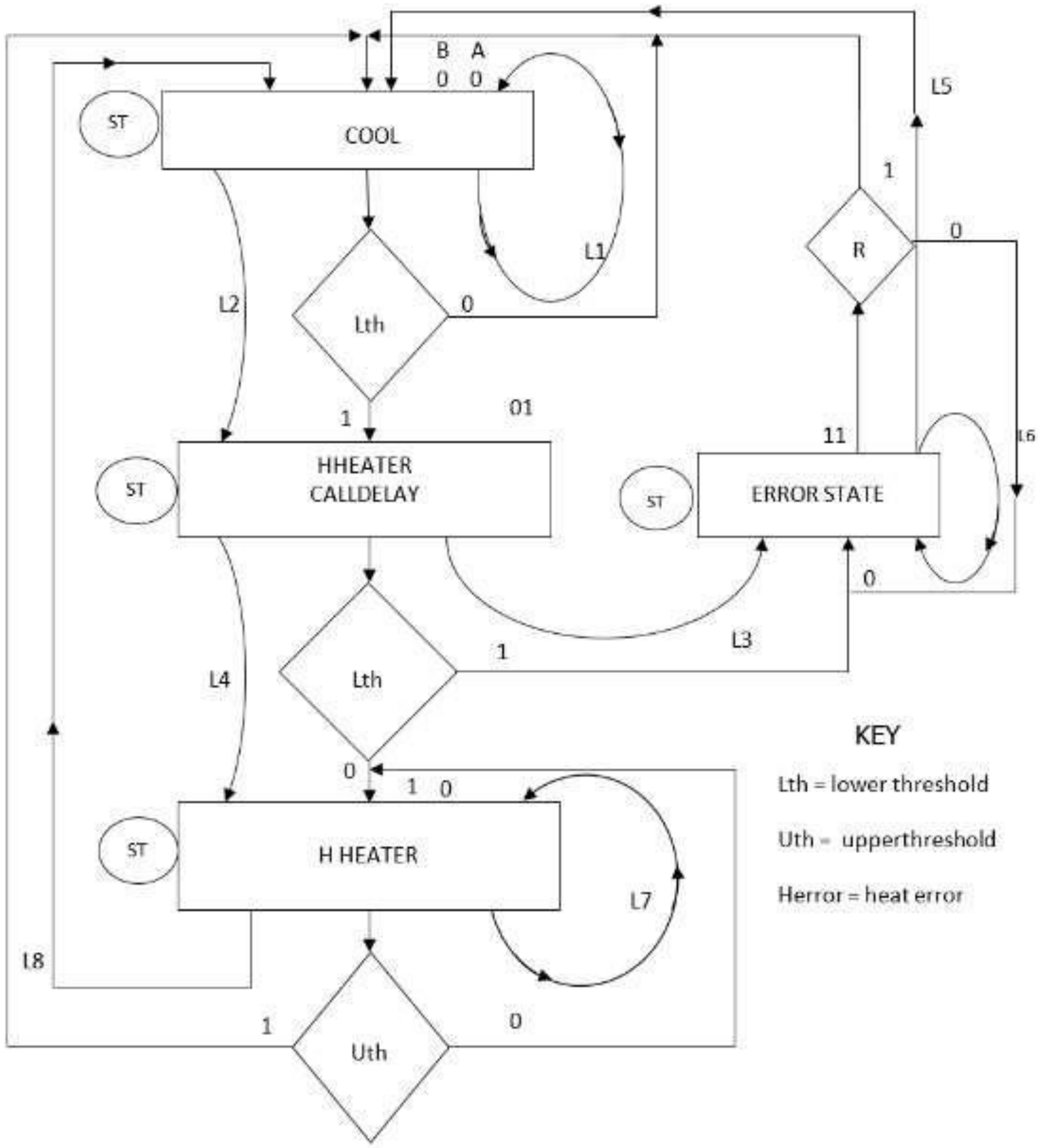

Fig. 3: ASM chart of the temperature process control system

2.3.3 State transition table (STT) for temperature control system

The ASM chart of the temperature control system has an equivalent tabular representation known as a state transition table (STT) shown in Table 1. An ASM chart can be fully described in terms of the link paths comprising it. A state machine transits from the present state to the next when a clock pulse occurs. A link path is a path that follows from the present state to itself or to another state when the clock pulse arrives. When there is an input qualifier between the present state and another, the logic level of the qualifier determines the next state the machine goes to at the clock pulse. If there is no qualifier between the present state and the next, the machine must conditionally transit from its present state to the adjacent state in forward direction when a clock pulse arrives. The ASM chart has eight link paths labelled L1 to L8. L1 is the link path from state 0 back to itself when the input qualifier is 0. Also, L2 is the link path from state 0 to state 1 when the input qualifier is 1. Similarly, L3 represents the transition from state one (ST1) to state two when the qualifier Lth is 0. L4 is the transition from state two back to itself when the input qualifier U_{th} is 0 and L5 is the transition from state one to state three which is the error state where the system stays until a button is pressed to return it to state 0 through link path L7.

In the STT table a number of dashes appear under the column heading input qualifiers (Lth, Uth, and RST). The dash (-) implies that the input qualifiers are not relevant to the transition that is being made by the link path. An Input qualifier that is relevant to the link path will either be 0 or 1, rather than dash. A dash in the STT also means that the input qualifiers that appear in column heading may be at logic 0 without affecting the control process.

Table 1: State transition table of temperature process control system

Link path	Input qualifiers (Lth, Uth, Rst)	Present state name	Reesent state code	Next state name	Next state code	Output (Hhtr, Herror)
L1	0 - -	ST0	0 0	ST0	0 0	0 0
L2	1 - -	ST0	0 0	ST1	0 1	0 0
L3	1 - -	ST1	0 1	ST2	1 1	1 0
L4	0 - -	ST1	0 1	ST3	1 0	1 0
L5	- - 1	ST2	1 1	ST0	0 0	0 1
L6	- - 0	ST2	1 1	ST2	1 1	0 1
L7	- 0 -	ST3	1 0	ST3	1 0	1 0
L8	- 1 -	ST3	1 0	ST0	0 0	1 0

Table 2 shows the fully expanded state transition table of the temperature control system. An STT is said to be fully expanded when all the dashes on each row are given all the possible combination of logic values, leading to new rows in the state transition table, one for each combination of the values for the dashes on that row. In this respect, an STT data row with one dash becomes two rows, one row when that (dashed) qualifier is given the logic value 0 and the other when the qualifier is given logic level 1. Similarly, an STT data row with two dashes expands into four STT rows. Assume the dashed qualifiers are represented by q1, q2. Then the first STT row in the expansion will be q1q2=0, 0, the second value is q1, q2= 0, 1, the third row q1, q2=1, 0, and the fourth row q1 q2 1, 1. As shown in table 1 above. Two dashes on STT data row would in like manner lead to four rows in fully expanded STT.

2.3.4 System flow chart

The flow chart shown in Fig. 4 is used to develop the control software of the incubator at the remote site. The program was written in assembly language using macro integrated development environment (MIDE) editor. There the program was debugged and complied before it was transferred to the microcontroller chip (89C51) using a WELLON programmer. The software that runs on the GUI is implemented using the flow chart of Fig. 5. MIDE was also used to develop the program codes which are thereafter loaded in the virtual controller.

Table 2: Fully expanded STT table for temperature control system

Link path	Present state code	Input qulifiers (Lth, Uth, Rst)	Next state code	State output (Hhtr, Heror)	Location address (Hex)	Location content (Hex)
L1	0 0	0 0 0	0 0	0 0	0 0	0 0
	0 0	0 1 0	0 0	0 0	0 2	0 0
	0 0	0 0 1	0 0	0 0	0 1	0 0
	0 0	0 1 1	0 0	0 0	0 3	0 0
L2	0 0	1 0 0	0 1	0 0	0 4	0 4
	0 0	1 1 0	0 1	0 0	0 6	0 4
	0 0	1 0 1	0 1	0 0	0 5	0 4
	0 0	1 1 1	0 1	0 0	0 7	0 4
L3	0 1	0 0 0	1 1	1 0	0 C	0 E
	0 1	1 1 0	1 1	1 0	0 E	0 E
	0 1	1 0 1	1 1	1 0	0 D	0 E
	0 1	1 1 1	1 1	1 0	0 F	0 E
L4	0 1	0 0 0	1 0	1 0	0 8	0 A
	0 1	0 1 0	1 0	1 0	0 A	0 A
	0 1	0 0 1	1 0	1 0	0 9	0 A
	0 1	0 1 1	1 0	1 0	0 B	0 A
L5	1 1	0 0 1	0 0	0 1	1 9	0 1
	1 1	1 0 1	0 0	0 1	1 D	0 1
	1 1	0 1 1	0 0	0 1	1 B	0 1
	1 1	1 1 1	0 0	0 1	1 F	0 1

L6	1 1 1 1 1 1 1 1	0 0 0 1 0 0 0 1 0 1 1 0	1 1 1 1 1 1 1 1	0 1 0 1 0 1 0 1	1 8 1 C 1 A 1 E	0 D 0 D 0 D 0 D
L7	1 0 1 0 1 0 1 0	0 0 0 1 0 0 0 0 1 1 0 1	1 0 1 0 1 0 1 0	1 0 1 0 1 0 1 0	1 0 1 4 1 1 1 5	0 A 0 A 0 A 0 A
L8	1 0 1 0 1 0 1 0	0 1 0 1 1 0 0 1 1 1 1 1	0 0 0 0 0 0 0 0	1 0 1 0 1 0 1 0	1 2 1 6 1 3 1 7	0 2 0 2 0 2 0 2

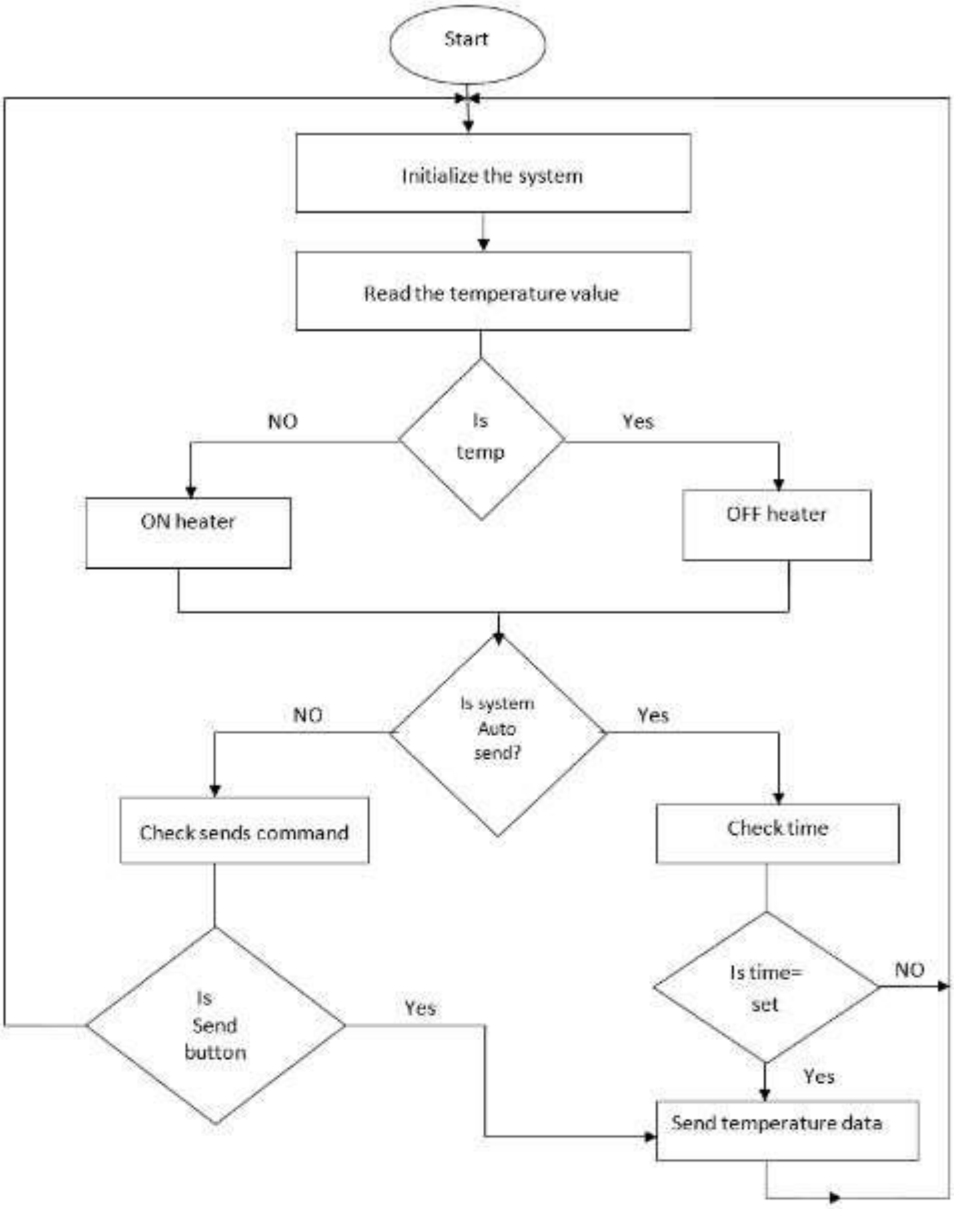

Fig. 4: Flow chart of control software at remote site

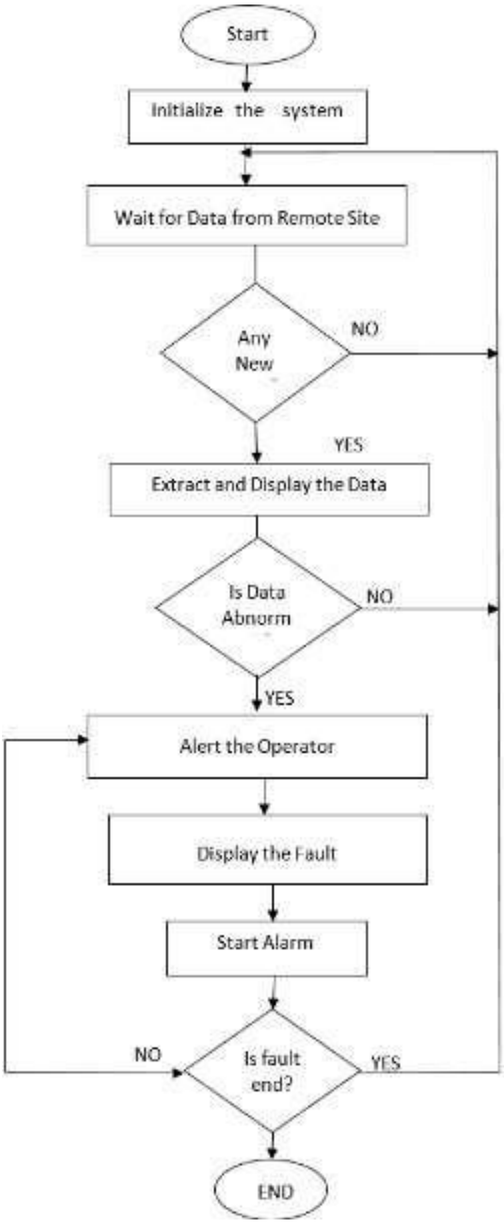

Fig. 5: Flowchart for the GUI implementation

3. Results and discussion

3.1 Hardware subsystem implementation

Hardware subsystem implementation

The hardware subsystem is in two phases: the hardware subsystem at the remote site and hardware subsystem at the operator's end. Each of the subsystem is made up of the input, control and output interfaces the subsystem is made up of the input, control and output interfaces.

The input interface

Microcontroller based oven shown in Fig. 4 is implemented at the remote site. The input elements include the temperature sensor (Lm35), analog to digital converter (ADC0804), crystal oscillator (11.0592MHz), auto/manual button and "send command" button. The sensor senses the temperature of the incubator and transfers the same to the microcontroller through ADC which converts the analog temperature value to the digital equivalent. The auto/manual button is used to determine whether the controller will send the processed temperature value at the press of a button or automatically. When the button is activated, the controller sends data after every 30secs automatically, otherwise the stored data is sent at the press of send command button. The hardware subsystem at the host (operator's) site is made up of the GSM modem, the signal level converter and the host computer. The GSM modem receives the temperature data sent from the remote and transfer the same to the host computer via a level converter, max232 as shown in Fig. 6.

The output interface

The output subsystem at the remote site is made up the liquid crystal display (LCD), the heater and a GSM modem. The controller displays the temperature on LCD in real time while the heater is responsible for supplying heat energy to the environment. The microcontroller sends temperature data to the host computer through the GSM modem. The output subsystem at operator's site shown in Fig. 7 is a virtual graphical user interface (GUI) comprising LED indicators, LCD, buzzer and Central Server. Proteus virtual simulation module was used to develop the output subsystem.

The central server receives the data sent from the remote site, interprets it and displays the information on the virtual LCD. The server also compares the received data with predefined thresholds and activates alarm when the temperature value goes beyond these set limits. Amber LED shows that system is operating normal while green LED shows that the server is ready to receive new data. Red LED and Alarm come on when a fault occurs. The output interface at operator's site is shown in Fig. 7.

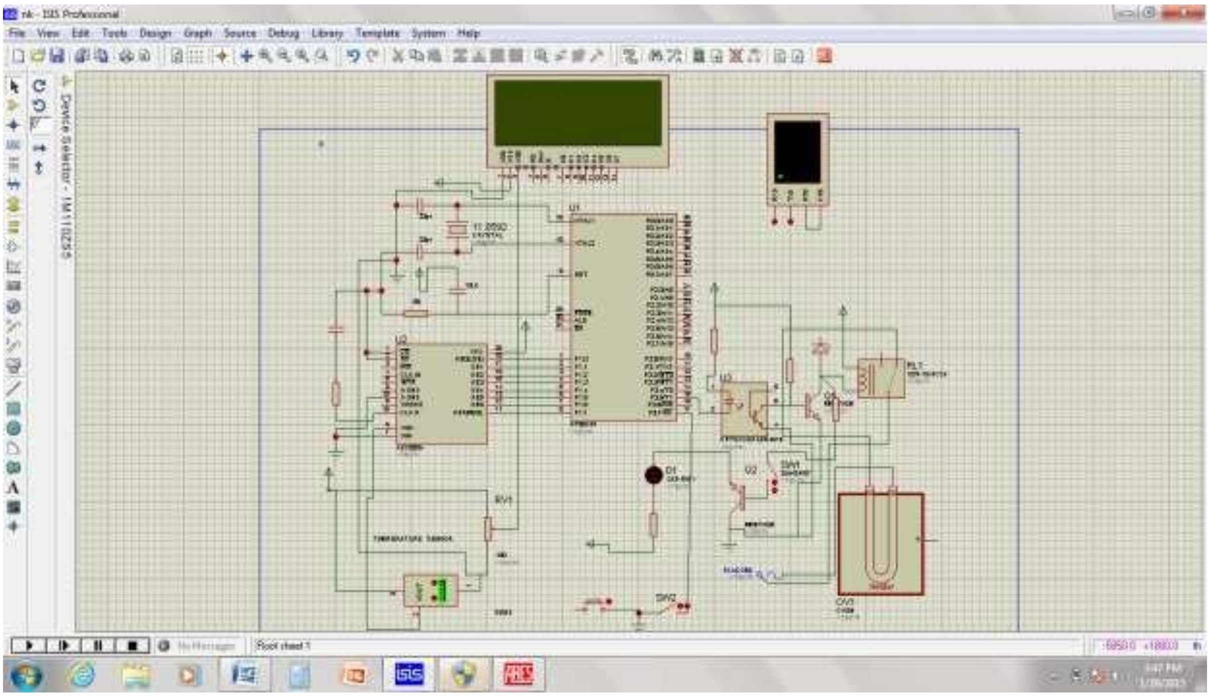

Fig. 6: Microcontroller-based oven system (remote site).

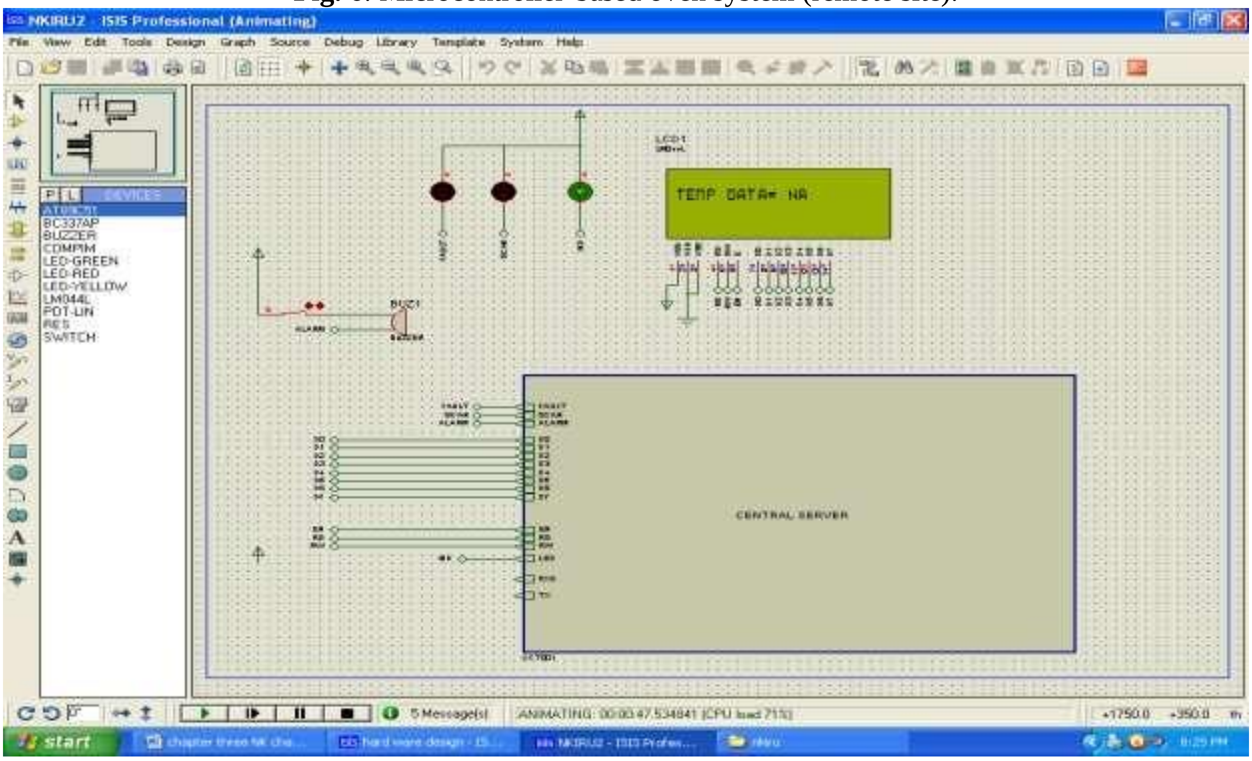

Fig. 7: Output interface at operator's site

3.2 The control subsystem implementation

The control subsystem is implemented using AT89c51 microcontroller. At the remote site, the controller performs the dual function of controlling the temperature of the environment and emulating the Attention (AT) Command of GSM modem using the result to send processed data to host computer. When compared to the work of Ayad et al. (2013), it was observed that the transmission time taken to send the processed data form to the host computer is minimal.

3.3 The communication subsystem implementation

The communication Subsystem was implemented using GSM modem, which was interfaced to the microcontroller for short message service (SMS). My x-5 SAGEM phone which is used in this paper has a number of facilities that make it suitable to use. Not only that the baud rate can be selected automatically by the phone, one can also set the baud rate manually, a feature which other phones do not have. The serial port of the phone is compatible with that of the microcontrollers and can accept Attention Commands (AT) in a text mode thereby making the programming aspect of the work less cumbersome, less time consuming and easy to understand and implement. My x-5 SAGEM phone was chosen due to its operational compatibility with the microcontroller implemented in this work.

Performance evaluation

Fig. 8 shows the graphical representation of the monitoring system with the temperature range of 33<=T<35 degree centigrade. Before the control system is switch ON the incubator was at room temperature of 25°C. The control system is now turned ON and the incubator temperature rises steadily as the heater is turned on by the control system. When the temperature of the plant gets to 35°C (upper threshold) the incubator heater is turned OFF and the incubator temperature slightly overshot the upper threshold and then begin to come down. When it falls down to 33°C (lower threshold) the incubator heater is turned ON once again. The temperature of the incubator begins to rise again after a slight undershot. As shown in Fig. 8, the control system keeps the temperature of incubator in the range between 33°C and 35°C, except for the slight overshot at 35°C and undershot at 33°C. When the control system is turned OFF the incubator temperature fall steadily down to room temperature of 25°C. In comparison with the work of Czerwinski and Kania (2016), it was observed that this new system maintained a more steady, reliable and accurate temperature, hence becomes an improved version of the previous works.

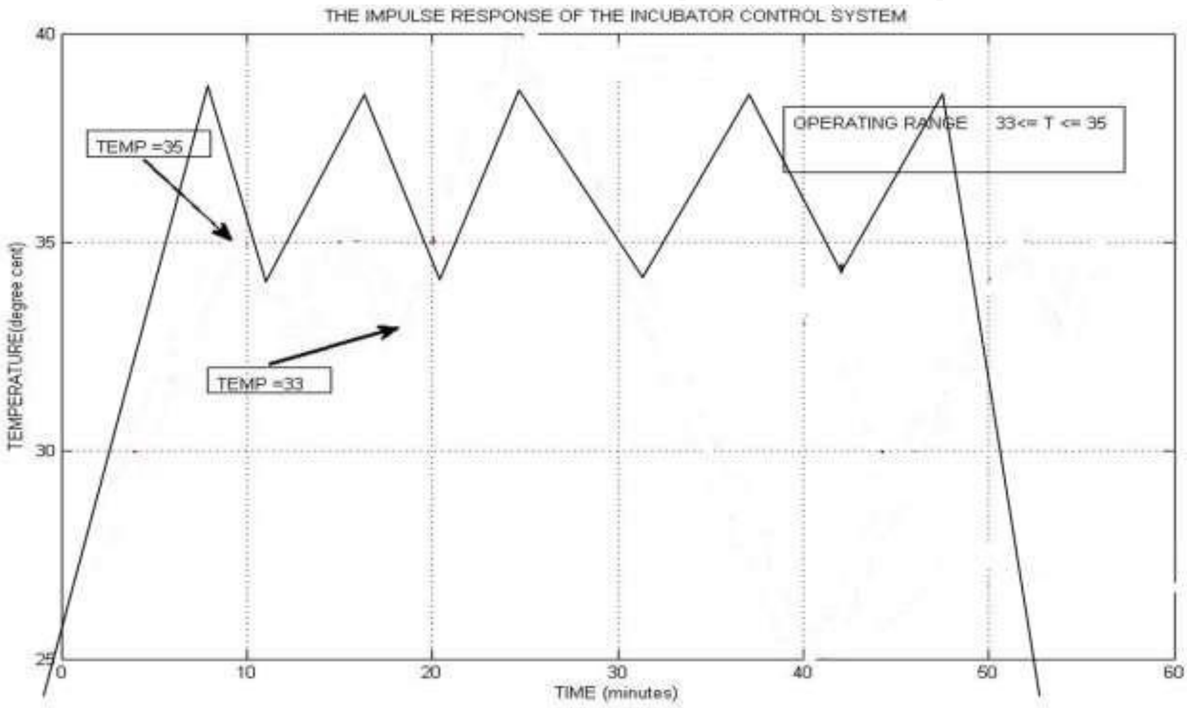

Fig. 8: Impulse response of the incubator control system

4. Conclusions

A generic approach to wireless access for remote monitoring has been designed and implemented. It is adaptable for use in a wide range of applications and yet it is simple, cost effective and does not need a complicated process of customization. The novelty in this proposed system is that a new application can use the software simply by replacing the STT table with that of the new application. As a result, the system can be plugged into many applications that require remote mobile access. It serves as an effective front-end communication interface that provides wireless access to data in real time from anywhere. The system was designed to be user friendly by providing all the facilities that will enable users monitor an on-going process.

References

Alghamdi, S. (2019) Design a Universal Remote-

Controlled Thermostat Based on FPGA, 2019 6th International Conference on Control, Decision and Information Technologies, pp. 918-921.

Fränzle, M., Chen, M., Kröger, P. (2019) In Memory of Oded Maler: Automatic Reachability Analysis of Hybrid-State Automata. ACM SIGLOG News 6, 19-39.

Ayad, G. I., Raghad, Z.Y. and Essa, F.A. (2013) Based Automatic Remote Control of Gas Reduction System using PIC Microcontroller. IRACST–Engineering Science and Technology: An International Journal, 3(2):217- 227.

Chivilikhin, D., Patil, S., Chukharev, K., Cordonnier, A. and Vyatkin, V. (2020) Automatic State Machine Reconstruction from Legacy Programmable Logic Controller Using Data Collection and SAT Solver, IEEE Transactions on Industrial Informatics, 16(12):7821-7831.

Chivilikhin, D., Patil, S., Cordonnier, A. and Vyatkin, V. (2019) Towards automatic state machine reconstruction from legacy PLC using data collection, Proceeding of 17th IEEE International Conference Industrial Information, 147-151.

Czerwinski, R. and Kania, D. (2016) State Assignment and Optimization of Ultra- High-Speed FSMs Utilizing Tristate Buffers. ACM Transactions on Design Automation of Electronic Systems, 22(1):1-25.

Gajalakshmi, R.K., Ananthkumar, T., Manjubala, P. and Rajmohan, R. (2020) An Optimized ASM based Routing Algorithm for Cognitive Radio Networks. 2020 International Conference on System, Computation, Automation and Networking (ICSCAN), pp. 1-6.

Hawryszkiewyez, I.T (1989) Introduction to System Analysis and Design, Prentice Hall India, New Delhi. Salauyou, V. and Bulatowa, I. (2018) Performance Targeted Synthesis of ASM Controllers on FPGA, Measurement Automation Monitoring, 64(2):31-33.