path addressable ROM patterns programmed into the microcontroller forms the control software for the control process. The address inputs to the address decoder part of the ROM device is the input linking the input port of the microcontroller and is used to locate the corresponding 'next state' and 'state outputs'. At the occurrence of a clock pulse, the next state pattern becomes the present state pattern which will join the input qualifiers to form the next address input to be used by the microcontroller. AT89C51 microcontroller is used in the prototype phase of VHTR design.

Table 2: Assignment of I/O lines to microcontroller ports

Process Variables	I/P Interface Lines	O/P Interface Lines
Temperature Control System	UTH is connected to I/P port bit 2, LTH is connected to I/P port bit 0	HTR is connected to O/P Port bit 1, ALARM is connected to O/P Port bit 0

Table 3: Fully Expanded STT Table for Temperature Control System

LINK PATH	PSB1 PSB2	PRESENT STATE CODE	INPUT QULIFIER LTH UTH RST	NEXT STATE CODE	STATE O/P HHT HER0R	LOCATION ADDRESS (HEX)	LOCATION CONTENT (HEX)
L1	0 0	0 0	0 0 0	0 0	0 0		0 0
	0 0	0 0	0 1 0	0 0	0 0	0 0	0 0
	0 0	0 0	0 0 1	0 0	0 0	0 2	0 0
	0 0	0 0	0 1 1	0 0	0 0	0 1	0 0
						0 3	
L2	0 0	0 0	1 0 0	0 1	0 0	0 4	0 4
	0 0	0 0	1 1 0	0 1	0 0	0 6	0 4
	0 0	0 0	1 0 1	0 1	0 0	0 5	0 4
	0 0	0 0	1 1 1	0 1	0 0	0 7	0 4
L3	0 0	0 1	1 0 0	1 0	1 0	0 C	0 A
	0 0	0 1	1 1 0	1 0	1 0	0 E	0 A
	0 0	0 1	1 0 1	1 0	1 0	0 D	0 A
	0 0	0 1	1 1 1	1 0	1 0	0 F	0 A
L4	0 0	0 1	0 0 0	1 1	1 0	0 8	0 E
	0 0	0 1	1 1 0	1 1	1 0	0 A	0 E
	0 0	0 1	0 0 1	1 1	1 0	0 9	0 E
	0 0	0 1	0 1 1	1 1	1 0	0 B	0 E
L5	0 0	1 0	0 0 1	0 0	0 1	1 1	0 1
	0 0	1 0	1 0 1	0 0	0 1	1 5	0 1
	0 0	1 0	0 1 1	0 0	0 1	1 3	0 1
	0 0	1 0	1 1 1	0 0	0 1	1 7	0 1
L6	0 0	1 0	0 0 0	1 0	0 1	1 0	0 9
	0 0	1 0	1 0 0	1 0	0 1	1 4	0 9
	0 0	1 0	0 1 0	1 0	0 1	1 2	0 9
	0 0	1 0	1 1 0	1 0	0 1	1 6	0 9
L7	0 0	1 1	0 0 0	1 1	1 0	1 8	0 E
	0 0	1 1	1 0 0	1 1	1 0	1 C	0 E
	0 0	1 1	0 0 1	1 1	1 0	1 9	0 E
	0 0	1 1	1 0 1	1 1	1 0	1 D	0 E
L8	0 0	1 1	0 1 0	0 0	1 0	1 A	0 2
	0 0	1 1	1 1 0	0 0	1 0	1 E	0 2
	0 0	1 1	0 1 1	0 0	1 0	1 B	0 2
	0 0	1 1	1 1 1	0 0	1 0	1 F	0 2

Communication Interface

The communication subsystem is used to interface the microcontroller to the GSM modem. To achieve this integration, a serial F-bus cable and a voltage level converter MAX 232 was introduced. The Universal Asynchronous Receiver/Transmitter (UART) of the microcontroller was synchronized with that of the phone by sending a string of 'U' 128 times. The message sent was converted to hexadecimal code for the phone to understand. This conversion was done by the control software using AT commands. In this regard, AT command was used to establish the communication between the microcontroller and the modem. Figure 2 shows the communication interface of the system.

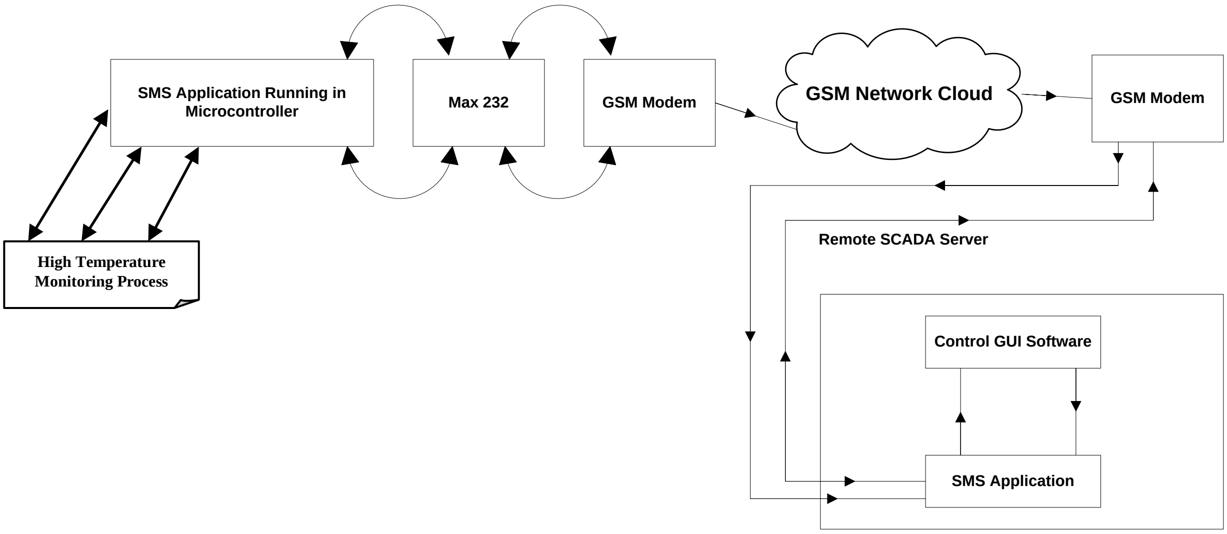


Figure 2: Temperature process control communication interface.

Graphic User Interface (GUI)

As seen from Figure 2, the SMS message from the microcontroller reaches the remote site via the communication interface (GUI). In this regard, the remote GSM modem receives the transmissions of the microcontroller and makes its data available to the GUI software. When the GSM modem interfaced to the microcontroller gets SMS, it transfers the same message to the second phone that is interfaced to the remote PC/server. The data sent by the microcontroller has two parts: the address part (made up of the state code and qualifiers) and the output part (made up of the next state code and the state outputs). The input to the PC/SCADA server system at the remote end is the normal test message from the GSM modem or handset. This SMS based system involves usage of the AT command. The system SMS software uses “AT Command” features at microcontroller end (sending end) and at remote end (receiving end). In this regard, “AT COMMAND” was used to establish communication with the modem. The following AT commands were used in context, viz:

i. “AT+CGMS”- Sending SMS Messages

ii. “AT+CMGD” – to clear the SMS receiving memory location in the GSM modem or message storage.

iii. “AT+CPMS”- preferred message storage

iv. “AT+CMGS” –Signals the modem to operate in test mode.

To ascertain whether a GSM/GPRS modem or mobile phone support the sending of SMS message through AT commands, the following were carried out:

i. Use the “AT command +CSMS” (command name in text: select Message Service) to check whether mobile-originated SMS message are supported.

ii. Perform test operations to check whether +CMGS (command name in text: SEND message Indications to TE), + CMGL (command name in text: list Message) and/ or +CMSS (command name in text: send Message from Storage) are supported.

For the message reception (in Figure 2), the receiving of SMS Messages and the reading SMS messages from message storage is achieved via the communication module. To ascertain whether a GSM/GPRS modem or mobile phone supports the receiving and reading of SMS message through AT commands:

1. Use the AT command +CSMS (command name in text: Select Message service) to check whether mobile – terminated SMS message are supported.

2. Perform test operations to check whether+ CNMI (command name in text: New Message Indications to TE), +CMGL (command name in text: list Message) and/or + CMGR (command name in text: Read Message) are supported.

If the GSM/ GPRS modem or mobile phone supports the +CNMI AT command, it can send a notification or directly forward the message to the PC whenever a new SMS message arrives. If the GSM/GPRS modem or mobile phone does not support +CNMI but support +CMGL and /or + CMGR, the PC has to poll the GSM/GPRS modem or mobile phone repeatedly in order to know if any new SMS message have arrived. For instance, consider Algorithm 1 for AT Command procedure:

Algorithm 1:

```
AT

OK

AT+CMGF =1
```

```
OK

AT+CMGL =”ALL”

+CMGL: 1,’’REC READ’’, “+2348035841219” „ “06/11/11, 00:30:29 +32”

+CMGL: 2,”REC READ “, + 23480358411219„” “06/11/,00:32:20 +32”

OK
```

Below is a description of what is done in Algorithm 1:

- Line 1: “AT” is sent to the GSM/GPRS modem to test the connection. The GSM/GPRS modem sends back the result code “OK” (line2), which means the connection between the HyperTerminal Program and the GSM /GPRS modem work fine.
- Line 3: The AT command +CMGF is used to instruct the GSM /GPRS modem to operate in SMS text mode. The result code “OK” is returned (line 4), which indicates the command line “AT+CMGF=”has been executed successfully. If the result code “ERROR” is returned, it is likely that the GSM/GPRS modem does not support the SMS text mode. To confirm, type “AT +CMGF=?” in the HyperTerminal program. If the response is “+CMGF(0,1)” (0=PDU mode and 1 =text mode), then SMS text mode is supported. If the response is “+CMGF: (0)“, then SMS text mode is not supported.
- Line 5-9: the AT command +CMGL is used to list all SMS text messages in the message storage of the GSM/GPRS modem. There are two SMS text massages in the message storage: “.”avio –chasis -12345 “ A simple demo of SMS text messaging” “+ 234803541219” is sender mobile phone number. “06/11/11, 00:30:29+32”and “06/11/11, 00:32:20+32” tell us when the SMS text messages were received by the SMSC”+32” is the time zone. Note that the unit is a quarter of an hour.” REC READ” indicates both of the SMS text messages have been read before.
- Line 11: The result code “OK” indicates the execution of the AT command +CMGL is successful.

The GUI software unpacks the received pattern and depending on the display setting set by the user, displays information either for the process manager, or for maintenance personnel or for the engineer. Supposed the user specified that the information should be displayed for the process manager (who merely wants to know if the system is working normally or having any kind of trouble), colors are used to convey this status information for the process manager.

System Control Flowchart

In this section, the control flow chart for the whole VHTR control system is described (see ***Appendix 3***). In the control regulation design, the system software keeps checking for feedback from the process until this requirement is satisfied. If feedback is gotten, the byte 1 and byte 2 patterns are broken down into state code and qualifier and next state code and outputs respectively. The software tests the radio button to know the type. Only one button is activated at a time and could be either the manager, maintenance personnel or engineer. The feedback information is compared with the pre-stored value to know if there is mismatch. If there is a mismatch, the officer using the GUI is alerted of a fault condition in the process control system. The data sent by the microcontroller has two parts; the address part (made up of state code and the state outputs) and the output part (made up of the next state code and the state outputs). Depending on the display setting of the user, the GUI software unpacks the received pattern and displays information either for the process manager, maintenance personnel of the engineer. To ensure user-friendliness, it is possible to change the setting on the display by the use of three radio buttons, of which only one is active at a time. Tables 4, 5, and 6, depict the displayed information pattern for the manager, maintenance man, and engineer respectively.

Table 4. Display Data for Temperature Control System Manager

State Name	Colors	Description
ST0	Green	The system is okay
ST1	Amber	The system is doing some testing
ST2	Red	The system is at error state
ST3	Green	Normal state

Table 5. Display Data for the Temperature Control System Maintenance

Feedbacks	Fault Rectification
0010001	Error state , Reset the system and press the button

Table 6. Display the Temperature Control Engineer

Feedback	B A HHEATER HERROR
0010001	0 0 0 1 Expected output
	0 0 1 0 Actual output

PERFORMANCE EVALUATION OF THE TEMPERATURE REACTOR SYSTEM

C++ is used to drive the automation process model in Proteus ISIS 8.5[13]. In order to properly evaluate the performance of temperature control in a reactor system as described in this work, an analytical model of the system was developed and simulated with MATLAB Simulink [14]. The three main parameters of the model design which also form the parameters for the system evaluation include: the temperature, the disturbance, and the control signal. The Simulink model showing the design feedback and feed forward compensators to regulate the temperature of the oven reactor through a heat exchanger is shown in Figure 3. This is similar to a typical automated heat exchanger process. The process temperature is kept constant by varying and monitoring the amount of steam supplied to the heat exchanger lower pipe via its control valve. Variations in the temperature of the inlet flow are the main source of disturbances in this process. Let disturbance = *d*, Temperature = *T* and Control Signal = *V*. The temperature set point and the temperature disturbance T is controlled using a control feedback element *d*. the transfer function G_d represent how a change in *V* affects *T*. To regulate *T* around a set point, feedback architecture is used to control the valve opening (voltage *V*). The feedback is implemented with proportional-integral (PI) controller. Feedback control achieves set point tracking while feed forward control help to reject measured disturbances.

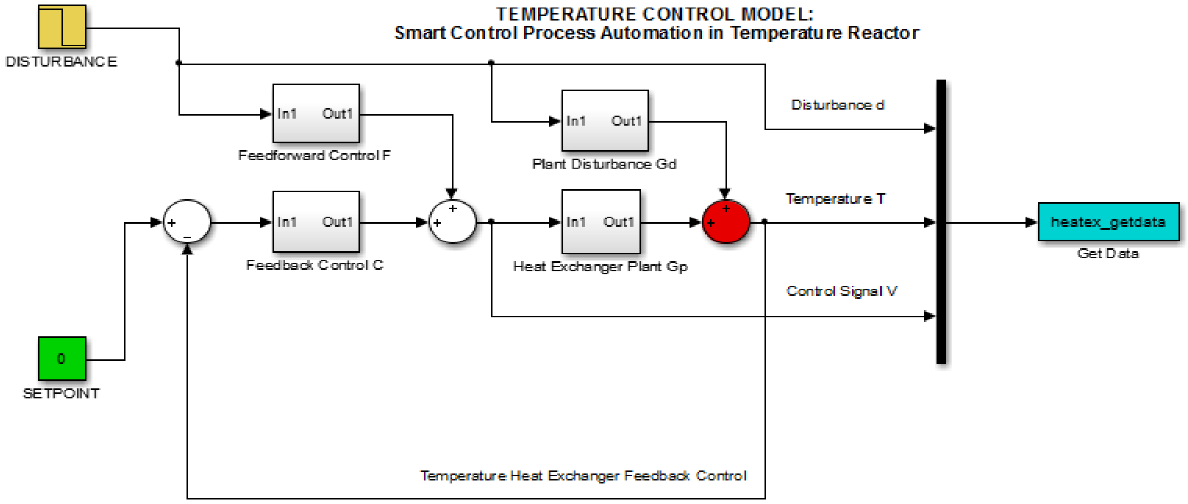


Figure 3. VHTR process automation model.

Figure 4 shows the response of the system disturbance, temperature and the control signal from the simulation of the model in Figure 3. It shows the relationship between the control signal, the disturbance and the temperature of the reactor system. As the control signal is varied, the disturbance and the temperature are regulated optimally.

From the graph, it was observed that there was no disturbance and variation in the temperature in the first 50 seconds. But after that, as the disturbance rises, the temperature decreases. This response to step change in temperature set point is shown in Figure 5. It is possible to have variations in temperature as a result of process changes in in-flow fluctuations. To reject such disturbances, alternative feed-forward architecture was realized. Hence, by introducing the feedback and feed-forward compensators to regulate the temperature of the oven reactor through a heat exchanger, Figure 5 was observed. Optimality response was obtained resulting from the effect of combining both feedback and feed-forward controls. The magnitude of deviation is also shown accordingly.

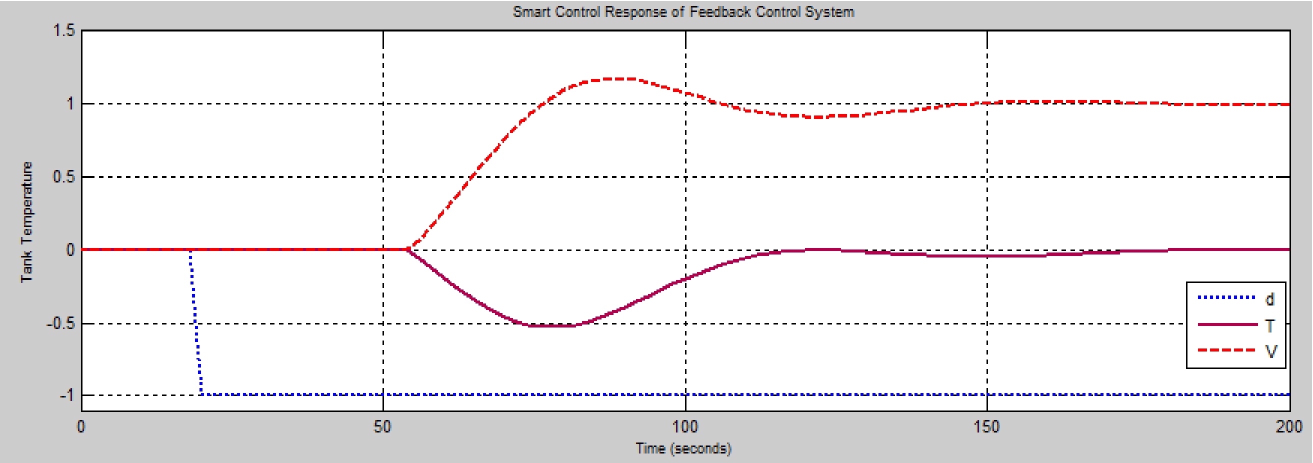


Figure 4. Temperature exchange reactor control response.

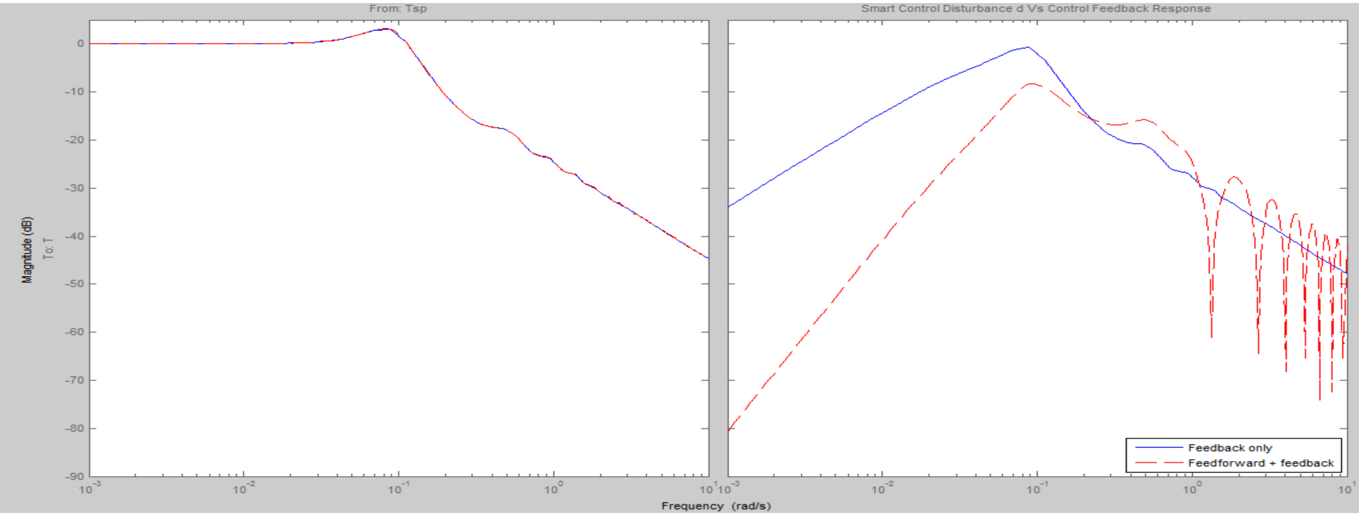


Figure 5. Feedback and feed-forward system response.

SYSTEM IMPLEMENTATION

Figure 6 shows a circuit diagram of the proposed system designed using Proteus ISIS 8.5 electronic design automation platform. The design schematics demonstrate microcontroller-based automation of temperature control system. The Arrhenius virtual simulation for temperature monitoring, demonstrating the input and output interfaces was realized in the design. In this case, 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 incubator environment. The microcontroller sends temperature data to the host computer through the GSM modem. The output subsystem at operator's site shown in Figure 7 is a virtual GUI comprising LED indicators, LCD, buzzer and Central Server. Proteus Virtual Simultion Module was used to develop the output subsystem. Figure 8a,b show the completed system deployed for real-life temperature monitoring.

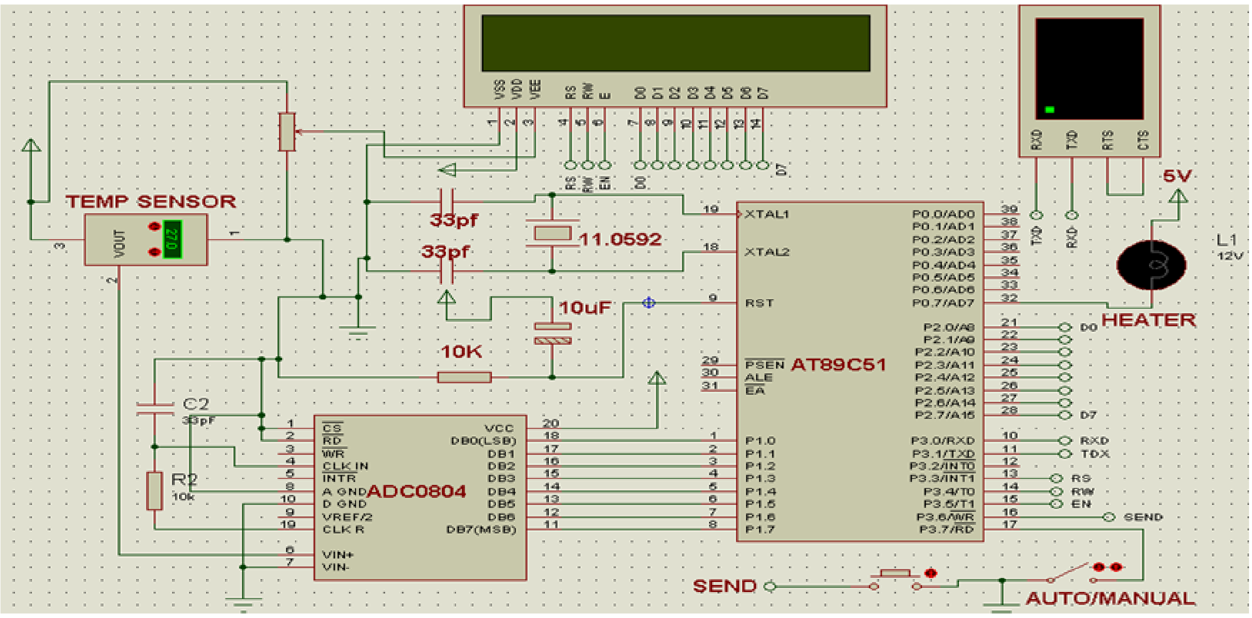


Figure 6. Arrhenius Virtual simulation for Temperature monitoring.

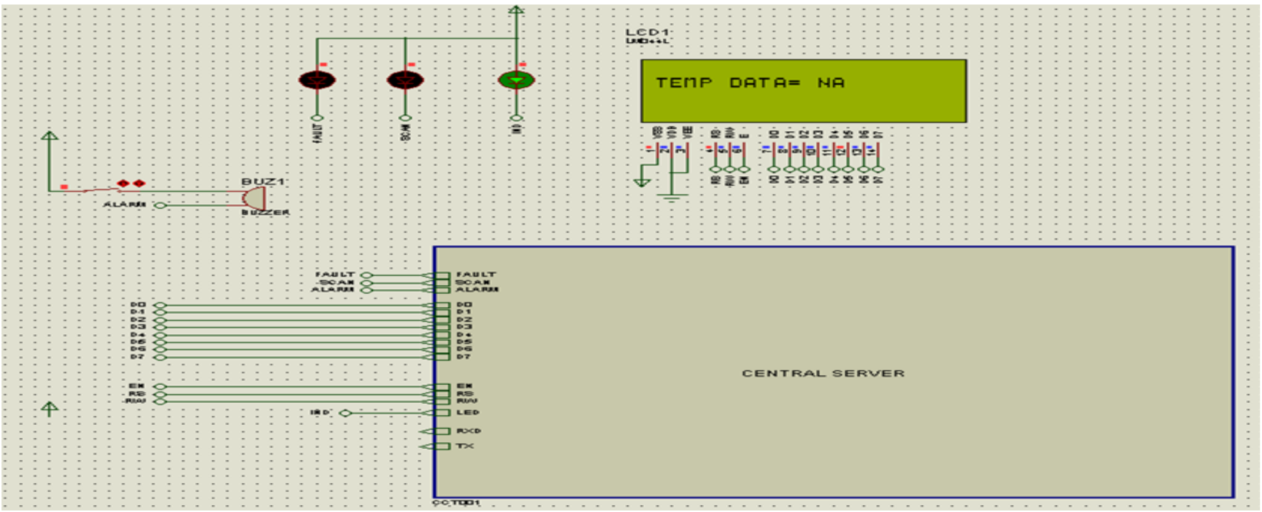


Figure 7. Remote monitoring output interface at operator's site.

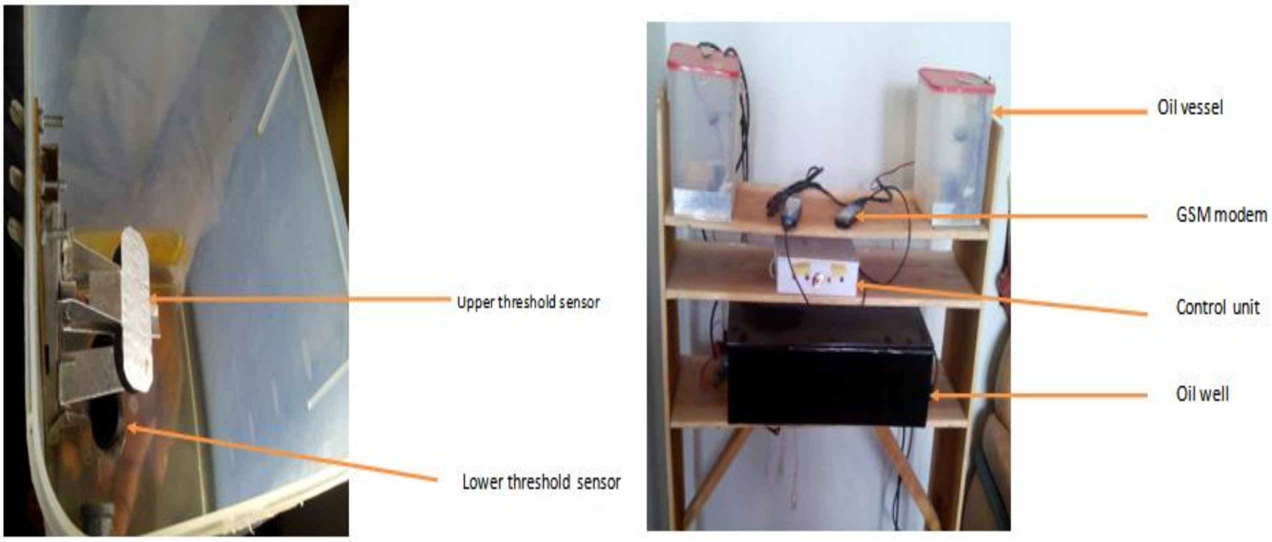


Figure 8a. Real-life VHTR monitoring and control Prototype with GSM modem.

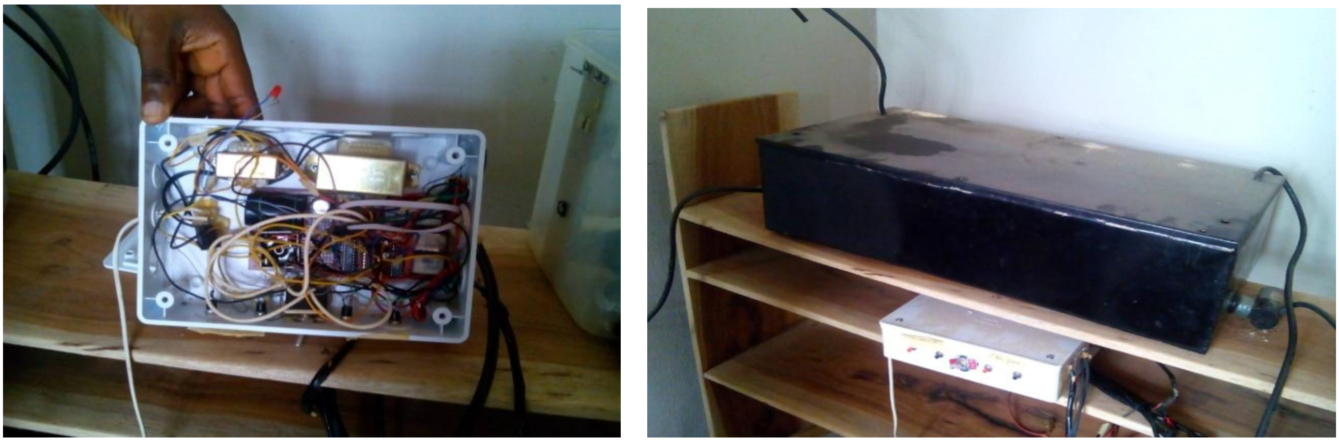


Figure 8b. Packaged real-life VHTR monitoring and control Prototype.

CONCLUSION

This research has presented a smart process digital control in a temperature reactor system (VHTR). The system features an efficient process control in L3 to avoid industrial disaster. A fully expanded state transition table was derived to provide an advanced design perspective to a smart digital temperature process control system while showing a contextual proof-of-concept. A developed temperature process control system was discussed. Its process control bit pattern generation, the process control microcontroller with the communication interfaces is presented. The system GUI alongside the control flow chart was highlighted. The performance evaluation of the temperature reactor system was discussed while detailing the full scale implementation. It is observed that the optimality of the system is feasible through the combination of the feedback and the feed-forward control elements. The design validation showed satisfactorily control states in real-time. This system can be applied in remote monitoring of a mission critical process control of VHTR systems. Future work will focus on VHTR system reliability based on interference theory and reliability optimization of the digital temperature reactor system.

REFERENCES

[1] Temperature Sensors for Thermal Reactors - Williamson Corporation. Available [Online]: www.williamsonir.com/temp-sensor-for-thermal-reactor, Retrieved 12/05/2018

[2] Georg Hackenberg, Christoph Richter, Michael F. Zäh, "A multi-disciplinary modeling technique for requirements management in mechatronic systems engineering- 2nd International Conference on System-Integrated Intelligence: Challenges for Product and Production Engineering; *Procedia Technology* 15 (2014) 5-16.

[3] Cesar Vandevelde, Francis Wyffels, Bram Vanderborght , Jelle Saldien, "Do-It-Yourself Design for Social Robots: An Open-Source Hardware Platform to Encourage Innovation", *IEEE Robotics & Automation Magazine*, 24(1), Pp. 86 – 94, 2017. DOI: 10.1109/MRA.2016.2639059

[4] Brenna Sniderman, Monika Mahto, Mark J. Cotteleer-" Industry 4.0 and manufacturing ecosystems-Exploring the world of connected enterprises", *Deloitte University press*", 2016.

[5] C. C.Udeze, U. Godwin, H. C Inyiama, C. C. Okezie, "A Cost-Effective Approach to the Design and Implementation of Microcontroller-based Universal Process Control Trainer, *International Journal of Advanced Computer Science and Applications, (IJACSA) Vol. 3, No. 1, 2012, Pp.* 142-147.

[6] Inyiama H. C, Okezie C.C, Designing microcontroller-based universal process control systems, (Electroscope), *Dept. Electrical and Electronics Engineering, Nnamdi Azikiwe University Awka*. 2(2), Pp.11-26, 2007.

[7] S.Rajesh Kumar , S.Rameshkumar, "Industrial Temperature Monitoring And Control System Through Ethernet LAN", *International Journal Of Engineering and Computer Science* ISSN:2319-7242 Volume 2 Issue 6 June, 2013 Page No. 1988-1991

[8] Huseyin Bulgurcu, " Development of multi process control trainer for Mechatronics Education", *http://deneysan.com/Content/images/documents/h-bulgurcu-development-of-multi-process-control-trainer-for-mechatronic-education-technics-technologies-education-management-volume-6-no-3-pages-783-791-2011_36113954.pdf.*

[9] Sayan Samanta , Subhodip Maulik and Srismrita Basu, "Development of Temperature Monitoring And Control System Using Temperature Sensor", *International Journal of Engineering Research & Technology (IJERT*), 1(8), 2012 Pp.1-8.

[10] Doug Chapin, Scott Kiffer, Jim Nestell, "Very High Temperature Reactor: A Technical Summary", June 2004. Available [Online]:*large.stanford.edu/courses/2013/ph241/kallman1/docs/chapin.pdf, Retrived 12/05/2018.*

[11] A preliminary research on software reliability model of Reactor Protection System of High Temperature Gas-Cooled Reactor-Pebble bed Module, *12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Zhangjiajie, China 2015, 15-17 Aug. 2015, **DOI:** 10.1109/FSKD.2015.7382302.

[12] D. J. Groebel, A. Mettas, F.B Sun, "Determination and interpretation of activation energy using accelerated-test data", *In Proc. Annual Reliability and Maintainability Symposium*, 22-25 Jan. 2001, Philadelphia, PA, USA, USA, DOI: 10.1109/RAMS.2001.902442.

[13] Professional Protus ISIS Proteus Professional Full 8.5 SP1 Build 22252 İndir 2017.

[14] https://www.fullprogramlarindir.com/proteus-professional-full-8-5-sp1-build-22252-indir-2017.html

[15] R2018a - Updates to the MATLAB and Simulink product families. Available [online]:
https://www.mathworks.com/products/new_products/latest_features.html, Retrieved 29/05/2018.

APPENDIX 1 - Arrhenius equation Derivation

At time *t*, Arrhenius equation specifies that

$$X = X_0 \, exp\left(\frac{-Q}{RT}\right) \qquad 1$$

Where

$X =$ Reaction Rate (S^{-1}) or diffusion co − efficient (CM^2/s)

X_0 = Pre − exponential factor $(S^{-1}$ *or* (CM^2/s)

$Q =$ Activation Energy for Reaction or diffusion (KJ/Mol)

R = Ideal gas constant (8.314J/Mol k)

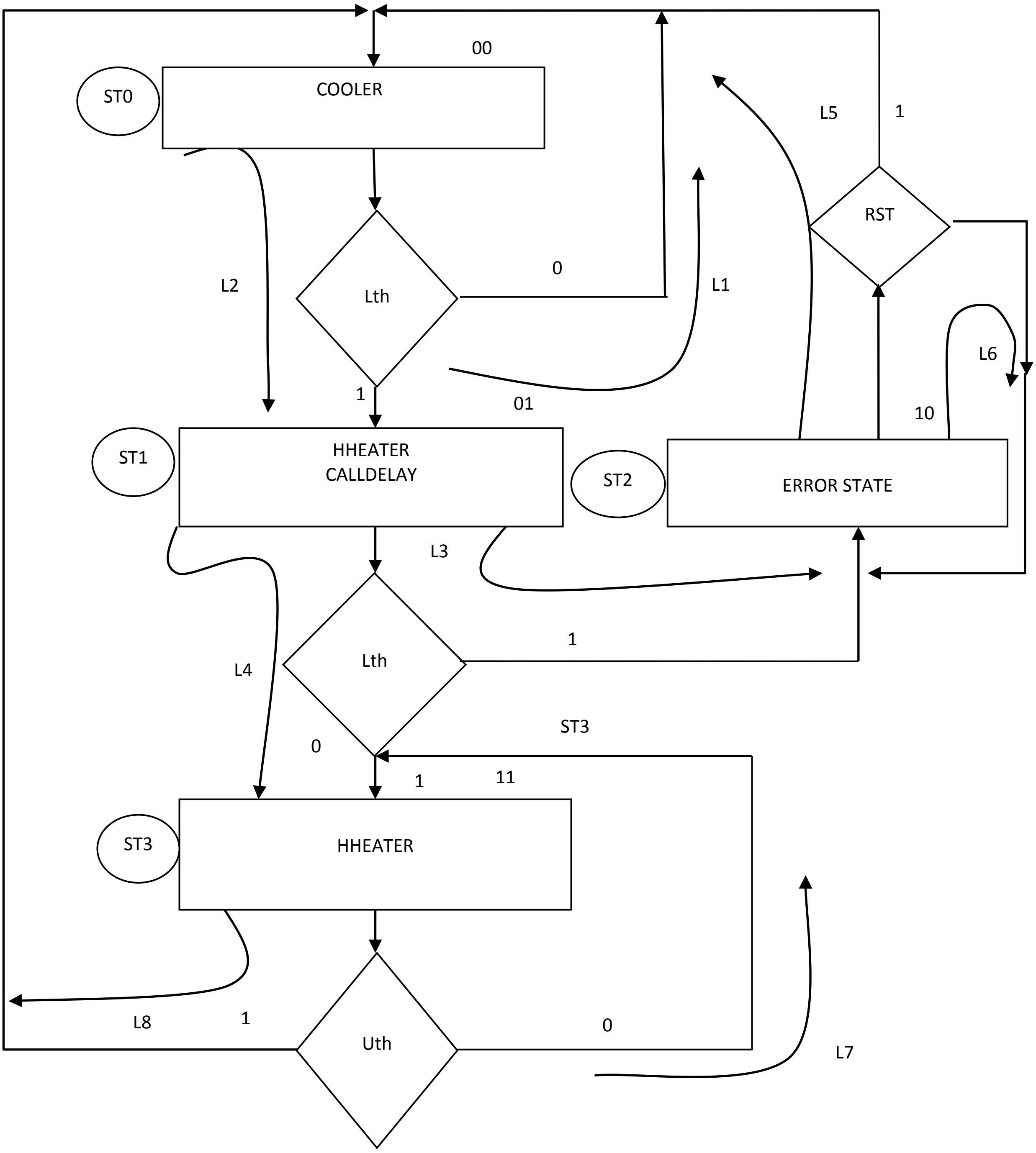
T = Temperature (K)

Ant time *t,* temperature transits from T_c to T_H, the reaction rates and diffusion rates increases, thus,

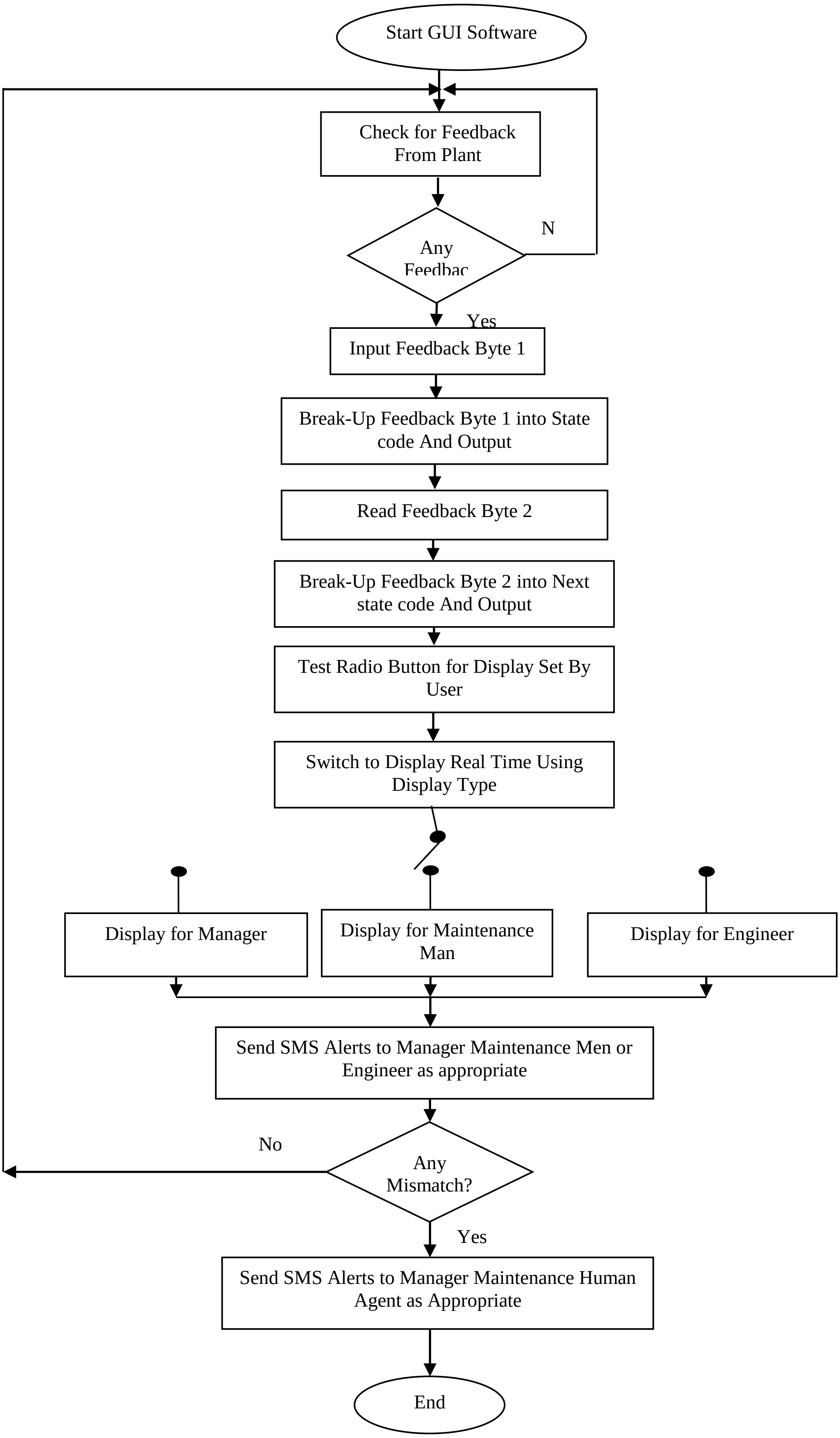
$$\frac{X_H}{X_c} = exp\left(\frac{E}{R}\left(\frac{1}{T_c} - \frac{1}{T_H}\right)\right) = \left(\exp\left(\frac{E}{R}\right)\right)^{\left(\frac{1}{T_c} - \frac{1}{T_H}\right)} \qquad \textbf{2}$$

Aggregation $\frac{X_H}{X_c}$ depends on the required activation energy; hence, slower the process with higher activation energies will be aggravated most by an increase in temperature. With smart digital automation, X and T are regulated for optimality.

APPENDIX 2: ASM chart for Temperature Control System:



APPENDIX 3: Process Control Flow chart Algorithm:


Start GUI Software
Check for Feedback From Plant
Any Feedbac
N
Yes
Input Feedback Byte 1
Break-Up Feedback Byte 1 into State code And Output
Read Feedback Byte 2
Break-Up Feedback Byte 2 into Next state code And Output
Test Radio Button for Display Set By User
Switch to Display Real Time Using Display Type
Display for Manager
Display for Maintenance Man
Display for Engineer
Send SMS Alerts to Manager Maintenance Men or Engineer as appropriate
No
Any Mismatch?
Yes
Send SMS Alerts to Manager Maintenance Human Agent as Appropriate
End