Quick Prototyping of Real Time Monitoring System Using Proteus Development Tool: A Case for Self Diagnostic Traffic Light System

P.I. Okwu, E.S. Mbonu, C.G. Ezekwe & C.U. Ajuzie
Electronics Development Institute
National agency for Science and Engineering Infrastructure (NASENI)
Awka, Anambra State, Nigeria
okwupi@gmail.com, mbo_eke_sam@yahoo.com, norakingchi@yahoo.com, ajuziasoftbiz@yahoo.com

ABSTRACT
In this paper, a self diagnostic system for maintenance of traffic light system is developed. The quality of light produced by the light emitting element of traffic light system at remote site is monitored via graphical user interface (GUI). The remote system’s status report obtained from the GUI helps the maintenance operator(s) to change or repair the traffic light head at appropriate time. The remote site consists of a prototype T-Junction traffic light, controlled by an intelligent processor (microcontroller) which also monitors and diagnoses fault(s) in the system and alerts the central server via GSM modem interfaced to it. The microcontroller emulates the attention (AT) command of the modem and uses it to send fault alert in form of short message service (SMS) to the host server. The server has GSM receiver interface to it through a smart level converter (max 232). The tools used in the development of the prototype T-junction traffic light include proteus Virtual Simulation Module Software Development Kit (VSMSDK) used to develop the GUI and the host server, macro integrated development environment(MIDE) used to edit and assembly the software program. Wellon programmer was used to transfer the assembled program to the microcontroller chip. The results showed that proteus professional, though primarily a virtual development tool can also be used for real time development of a prototype embedded system, reducing design-market time.

Keywords: Self diagnostic, graphical user interface, proteus VSM, real time development, prototype embedded system

1. INTRODUCTION
The word prototype comes from the Latin words proto, meaning original, and typus, meaning form or model. In a non-technical context, a prototype is an especially representative example of a given category. In software development, a prototype is a rudimentary working model of a product or information system, usually built for demonstration purposes or as part of the development process. In the systems development life cycle (SDLC) prototyping model, a basic version of the system is built, tested, and then reworked as necessary until an acceptable prototype is finally achieved from which the complete system or product can now be developed. In prototype-based programming, a prototype is an original object; new objects are created by copying the prototype. In hardware design, a prototype is a "hand-built" model that represents a manufactured (easily replicable) product sufficiently for designers to visualize and test the design [10]. The term ‘maintenance’ means to keep the equipment in operational condition or repair it to its operational mode. Maintenance has majorly been applied in production system where the main objective is to have increased availability of production systems, with increased safety and optimized cost [1].

According to [1], the maintenance strategies that are commonly applied in plants include break down maintenance, preventive or scheduled maintenance, predictive or condition based maintenance, opportunity maintenance and design out maintenance. The equipment under breakdown maintenance is allowed to run until it breaks down before repairing it. This strategy is suitable for equipments that are not critical and have spare capacity or redundancy available. In predictive maintenance, the condition of the system is monitored. Condition monitoring (CM) detects and diagnoses faults and it helps in planned maintenance based on equipment condition. This condition based maintenance strategy or predictive maintenance is preferred for critical systems and for such systems, breakdown maintenance is to be avoided. A number of CM techniques such as vibration, temperature, oil analysis, etc., have been developed, which guide the users in planned maintenance [1]. None of these techniques can be applied directly to traffic light system whose failure can be as critical as production plant since it can lead to loss of life, job, even property. Just imagine a fire fighter squad being trapped in a traffic hold up as a result of traffic light malfunction or failure.
Maintenance culture has been very poor with respect to traffic light system, evidenced by lack of or scanty publications and conference proceedings on the subject matter. This paper therefore seeks to develop a self diagnostic system that will monitor the state of traffic light system and send situation report to maintenance operators. Traffic light head is used as a case study.

Self-diagnosis, a term traditionally used in medical science to mean the process of diagnosing or identifying medical conditions in oneself, has in recent years been applied in science and engineering for fault identification and correction. Authors of [3] developed low hardware overhead self-diagnosis technique using reed-solomon codes for self-repairing chips. The circuit under diagnosis is assumed to be composed of a large number of field repairable units (FRUs), which can be replaced with spares when they are found to be defective. Jen-Chieh Yeh et al in [4] proposed a systematic approach in testing flash memories, including the development of march-like test algorithms, cost-effective fault diagnosis methodology, and built-in self-test (BIST) scheme. A novel generalized-comparison-based self-diagnosis algorithm for multiprocessor and multicomputer systems using a multilayered neural network is described by [5].

The authors considered the system-level self-diagnosis of multiprocessor and multicomputer systems under the generalized comparison model (GCM). In this diagnosis model, a set of tasks is assigned to pairs of nodes and their outcomes are compared by neighboring nodes. The collections of all comparison outcomes, agreements and disagreements among the nodes, are used to identify the set of faulty nodes. In this work, the comparison-self approach proposed by [5] is applied in traffic light control system. Specific tasks are assigned to sensor nodes that monitor the state of traffic lamp heads. An Intelligent controller in return evaluates the state of each lamp head by comparing the real outputs of the sensor nodes with the expected outputs stored in the data base residing in the memory of the controller. How proteus professional, though primarily a virtual development tool, can be used for real time development of a prototype monitoring system is show cased in this work.

2. OVERVIEW OF SOFTWARE DEVELOPMENT TOOL FOR EMBEDDED SYSTEM DEVELOPMENT

Embedded system is usually comprised of hardware and firmware. The firmware is the application specific software (ASS) running inside a microcontroller which controls the hardware interfaced to it. A number of software development tools are available for embedded systems. The list includes but not limited to electronic work bench, livewire, multism, edison and eagle [6], [7] [8]. The major draw back of the mentioned tools is that none of them has virtual com port which can be used to interface external microcontroller hardware for real life simulation and data acquisition. Besides, their component library is limited when it comes to high level intelligent applications (HLIA). Although TINA, also a development tool, has real time measurement capability, it does not provide virtual port for real life embedded system interface [9]. Proteus Virtual Simulation Module (VSM) Software Development Kit (SDK) overcomes these limitations. Proteus is software for microcontroller simulation, schematic capture, and printed circuit board (PCB) design. It is developed by Labcenter Electronics. Figure 1 shows the virtual DB9 com port which can be used to transfer and receive data serially to and from an external microcontroller based system while figure 2 shows how to interface the port to a microcontroller. The com port can accept baud rates in the range of 2400bps to 57600bps.

Figure 1: virtual com port

Figure 2: Interfacing com port to microcontroller
Figure 3: conceptual architecture model of the Traffic monitoring System

3. DESCRIPTION OF THE SELF DIAGNOSTIC TRAFFIC LIGHT SYSTEM

Figure 3 shows the conceptual architecture model of the traffic light monitoring system with low cost graphical user interface. Below is the description of the model.

- The traffic light apartment houses the lamp head which is made up of six ac lamp bulbs arranged in the North-East direction. The North direction is composed of three bulbs (red, amber green). Each bulb is housed in a separate apartment, see figure 4. The East direction is a replica of the North direction. Each bulb apartment has a light sensor embedded it.

- The intelligent controller has three major software components: control program (CP), intelligent agent (IA) and data base (DB). The control program controls the switching pattern of the bulbs. It also put the system on hold if a fault occurs until the fault is cleared. The data base is a look up table that contains the input patterns to the bulbs and expected output patterns. The intelligent agent takes an input pattern from the control program and then matches the real output pattern against the expected output pattern and reports back any mismatch or error to the control program. The IA also uses rule base reasoning to identify fault(s) and then prompts and copies the fault(s) to transmission subroutine which uses attention command (AT) protocol to send short message service (SMS) fault alert to the host computer.

- Embedded application programming interface (API) is used to achieve a SMART interfacing and communication between the intelligent controller and the GSM modem.

- The GSM modem acts as a radio frequency transmitter which sends the fault alert to the manager (host computer) and manager’s mobile phone.
The host computer contains the application program (AP) for graphical user interface (GUI) containing the virtual model (VM) of the traffic light system. The VM has a liquid crystal display (LCD) for virtual display of the state of the system at the remote site. When a fault occurs, remote site alerts the manager via the GUI and his mobile station. VM alarm is also activated. The manager upon receiving the alert instructs the technician to go and rectify the faults(s) at the remote site. After clearing the fault(s), the technician or operator resets the system enabling the intelligent controller to send situation report to the manager before continuing its operation.

4. TECHNICAL SOLUTION AND PROJECT DETAILS

4.1 Implementation of traffic light system at remote site.

![Figure 4 (a & b): prototype traffic light system](image)

![Figure 5: the schematic design of the traffic light system.](image)

Figure 4 (a & b): prototype traffic light system

Figure 5: the schematic design of the traffic light system.

Figure 5 shows the schematic design for implementation of the traffic light system at remote site and is made up of the following components.

**The Intelligent Controller:** Atmel 89c51 microcontroller is used. It has 4 input/output ports, 32 input/output pins, serial communication port, 4Kb of internal ROM, 128 kb of internal RAM among other features [11]. Pin 9 is the RESET pin. It is an input and is active high (normally low). Upon applying a high pulse to this pin, the microcontroller will reset and terminate all activities. This is often referred to as a power on reset. In order for the RESET input to be effective, it must have a minimum duration of 2 machines cycles [12].

Now the voltage charge in an RC circuit is given by...
Vc = V (1 – e^(-t/RC)) \[19\] \[13\]  

Where  
RC = time constant  
t = 2 machines cycles  
now V = Vcc = 5v  

Vc = voltage across C3 which is usually less than Vcc because of discharging action of the capacitor through resistor R1.  

Machine cycle = 1/value of crystal/12MHz \[2\]  
1/machine cycle = 11.0592 x 10^6 / 12 x 10^6 = 921.6 KHz  
Machine cycle = 1/921.6 KHz = 1.085µsec  
2machiones = 2 x 1.085 = 2.17µs  

From (1) Vc/V = 1 – e^(-t/RC)  
e^(-t/RC) = (1 – Vc/V)  
\ln e^(-t/RC) = \ln (1 – Vc/V)  
c = -(t/RC)(1 – Vc/V)  

Let R1 = 10k, then c3 = 10µf, t = 2.17µs  
The crystal oscillator (X1) provides operating frequency of the microcontroller. The 11.0592MHz value was chosen so that SMART interface between the controller and standard modems can be achieved \[12\]. Capacitors C1 and C2 provide stability to the crystal during operation.  

The relay Interface IC, ULN2003: This is used to achieve a neat and smooth interface between the controller and the ac relays. More information about the chip can be gotten from \[14\].  

The traffic Bulbs: 220vac, 40W bulbs are used as lamp heads. They are interfaced to the controllers through the relays.  

GSM modem: Although there are GSM modems customized for embedded applications, Sagem phone (My-X5 ) was used in this work because of its low cost and other advantages enumerated in \[15\].  

The feedback Element: Figure 6 shows the expanded circuit diagram of the feedback element.  

![Figure 6: expanded circuit diagram of feedback element.](image)

It is made up of resistors (R2-R7, each 10 kilo ohms) connected in series with a light dependent resistor (LDR). This arrangement forms a voltage divider and the output is fed into the input of an operational amplifier (op-amp). The other input of the amplifier is referenced to 2V. The value of the LDR on total darkness is about 20 kilo ohms. So the input to the op amp at total darkness is  

Vop = (20/ (20+10))*5 = 3.33v.  

This will force the op-amp output to high. Now when light from the traffic lamp head falls on the LDR, its resistance reduces forcing the op amp input to drop below 2v thereby making the output low. It is these changes that the intelligent agent monitors and makes decision based on the rule embedded in the microcontroller.  

4.2 Fault Identification  

Each time there is a mismatch of input pattern and expected output feedback pattern, the controller sends sms to the host computer and also to the mobile phone of the manager. The sms contains the state code of the pattern in question, the input pattern and real-time feedback pattern. The host computer also has data base containing the vectors of input pattern and expected output feed back pattern. Fault identification is done by complementing the real-time feedback pattern and using the result to activate the virtual traffic light. The host computer uses the input pattern to extract the expected output result and display it on LCD. Thus, by reading the information on the LCD and observing the virtual traffic light, the manager can easily identify the fault(s).
4.3 Software Implementation

The flow chart that implements the program control, monitoring and sms alert is shown in figure 9. Note that feedback is connected to port1 (P1) and output bulbs to port 0 (P0) of the microcontroller as shown in figure 3. N is used to represent states, i.e, N=000 – 005.

4.4 Implementation of Graphical User Interface

Figure 7 shows the GUI at the host computer. It comprises LCD display unit, the virtual traffic light heads, GSM modem receiver, buzzer alarm, all interfaced to intelligent microcontroller. The controller receives the sms fault alert through the receiver, extracts and interprets it and then, displays the result on the LCD. The present state of the traffic light heads at remote site is replicated by the virtual heads. Figure 8 shows the flow chart for fault diagnosis at manager’s host computer.

Figure 7: GUI for Traffic Light Monitoring

Figure 8: flow chart for fault diagnosis at host computer
4.4 Test and Result.
The completed traffic light system was tested and the result tabulated as shown in Table 1.

Table 1: Summary of test and result

<table>
<thead>
<tr>
<th>Test</th>
<th>Expected Result</th>
<th>Result obtained</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open and run the GUI at host computer</td>
<td>The GUI should display ‘Traffic light monitoring system’ with Red and Green lights showing as depicted in figure 7.</td>
<td>The GUI displayed ‘Traffic light monitoring system’ after 2 seconds. The Red and Green lights also came on.</td>
</tr>
<tr>
<td>Power on the traffic light control system at the remote end.</td>
<td>The system should initialize and send sms status report to host computer. Thereafter system should start working according to state transition table 1.</td>
<td>The GUI did receive the status report after 10 seconds as shown in figure 10. The system started working according to table 1.</td>
</tr>
<tr>
<td>Disenable RED bulb in the NORTH direction using simulator switch, SW1</td>
<td>The system at remote end should send error report to the host computer and mobile phone of the manager, ‘+2348068633798’, and put the traffic light on hold.</td>
<td>The host computer and the phone with the mobile number ‘+2348068633798’ received sms error report. All the traffic lights went off even though there was still power in the system.</td>
</tr>
<tr>
<td>Deactivate the switch SW1 and reset the system at remote site</td>
<td>The remote system should send status report to host computer and manager’s mobile phone. Thereafter, the system should start working normal.</td>
<td>The manager and the host computer received sms status report 15 seconds after resetting the system at remote site. The remote site also started working normal.</td>
</tr>
</tbody>
</table>

5. CONCLUSION

Quick prototyping helps to reduce design-market time. Proteus VSM has inbuilt capabilities that enable fast embedded system prototyping. These capabilities can be explored. The com port and the approach used in this work can be employed in other system development.
REFERENCES

[10] [http://searchcio-midmarket.techtarget.com/definition/prototype
[14] [http://pdf.datasheetcatalog.com/datasheet2/0f/06fx6a46ig46dxfj32qsa1i8o3y.pdf

Authors’ Bio

Dr. Okwu Patrick Ikechukwu is a Deputy Director Research and Development at Electronics Development Institute, Awka. He has B. Eng and M.Eng in Electronics Engineering and PhD in Electronics productions. He is a member of IEEE, NSE and COREN. Email: okwupi@gmail.com

Engr. Mbonu Ekene Samuel is a senior Research Engineer at Electronic Development Institute, Awka. He has B.Eng in Electronic Engineering, M.Eng in Computer Engineering. Currently, he pursues PhD at Nnamdi Azikiwe University, Nigeria. His research interests include Embedded Systems, mechatronics Systems, fault tolerant and self diagnostic systems; integrated systems: comprising computer, control, communication and cognition, flexible automation, and wireless Sensor Network. He is a member of IEEE, NSE, COREN, and ISA. Email: mbo_ekte_sam@yahoo.com

Mrs. Ezekwe Chinwe Genevra is a Senior Scientific Officer at Electronics Development Institutes, ICT

Engr. Ajuzie Uchechukwu Chiemezuwo a Senior Research Engineer with Electronics Development Institute (ELDI) Awka, Anambra State, a Federal government Owned research institute, under the National Agency for Science and Engineering Infrastructure (NASENI). He obtained a B.Eng in Electrical/Electronics and Computer Engineering at Nnamdi Azikiwe University Awka, Anambra State, Nigeria in 2004, M.Eng in Electronics and Computer Engineering at the same school in 2009 and he is currently running his PhD program in Electronic and Computer engineering in the same University. His research interest are Smart grid systems, security system automation, embedded control and communication systems, using GSM technology to solve engineering and national problems, inverter system development and solar application systems, fuzzy logic control systems, web application development e.t.c. His contacts are phone no: +2348030745870 and E-mails: ajuziasoftbiz@yahoo.com or ajuziasoftbiz@gmail.com