Development of Web Based SCADA like Application using Arduino Platform

Mrs. Archana A. Hatkar[1], Mr. Prashant S. Bibave[2], and Mr. Liladhar P. Bhamare[3]
[1]*ME Electronics in Digital Systems, Sir Visveswaraya Institute of Technology, Nashik*
[2]*ME Electronics in Communication, Sir Visveswaraya Institute of Technology, Nashik*
[3]*ME Electronics in Microwave, Sir Visveswaraya Institute of Technology, Nashik*

***Abstract*—Nowadays SCADA systems are used for home automation, Greenhouse automation, E-agriculture, Power generation and distribution etc. SCADA is nothing but Supervisory Control and Data Acquisition System. Basically, these SCADA applications include Level Monitoring, Light Climate Control, Security Surveillance, control and manage spatially separated utility sites and Control of Shutters Doors and so on. With the arrival of new hardware and software technologies here a system is proposed which can perform similar to SCADA applications at lower cost and lower maintenance. This project proposes a viable solution for SCADA like applications which include monitoring and controlling of Water level, Oil level, Temperature, Voltage, Humidity Displacement etc. This system can not only perform these industrial applications but also proposes a fine web-based solution to access all these acquired data and equipment. Here a remote based application is used which will allow the user to access the inter-organizational data/equipment in industries via internet, it also overcomes the problem of weak encryption used by the SCADA. Arduino Platform is the new technology used for supervisory control purpose. Alarm handling, Access Control, Automation, Logging, Archiving, Report generation, Interfaces to hardware and software etc are some features provided by the application. In future this system using .NET platform may replace the whole SCADA solution.**

***Index Terms*—SCADA, .NET, Temperature sensor, Humidity Sensor, Logging.**

I. INTRODUCTION

Web-Based SCADA is the area of research nowadays and it is a topic of research. There is lots of research is going on in this area. SCADA has wide range of applications. Here we are going to develop a small module with five types of sensors which are sensing the real time parameters from environment and web-based monitoring is provided to it with the help of interfacing to wireless computer/Laptops by using real time server. Control mechanism is also developed to keep these environmental parameters within set point.

1.1 Basics of SCADA Systems

SCADA is Supervisory Control and Data Acquisition Systems. SCADA programs are used in industrial process control applications for centralized monitoring and recording of pumps, tank levels, switches, temperature etc. in power plants, electrical power grids, and water treatment [1]. SCADA systems are also referred to as HMI i.e. Human Machine Interface. SCADA (supervisory control and data acquisition) is a type of Industrial Control Systems (ICS). The task of supervision of machinery and industrial processes on a routine basis can be an excruciatingly tiresome job. Always being by the side, a machine or being on a 24x7 patrol duty around the assembly line equipment checking the temperature levels, water levels, oil level and performing other checks would be considered a wastage of the expertise of the technician on trivial tasks [2]. But, to get rid of this burdensome task, the engineers devised equipment and sensors that would prevent or at least reduce the frequency of these routine checks. As a result of that, control systems and it's various off springs like SCADA systems were formed [2].

Computers offered flexibility in programming and communicating with field data acquisition units that was previously being done by hard wired equipment. SCADA monitors, controls and alarms the plant and/or regional facilities' operating systems from a centralized location. It includes the communication of information between a SCADA central host computer, many scattered units and/or Programmable Logic Controllers. For example, in a water filtration plant, the remote units measure the pressure in pipes and report the readings to the central computer located somewhere in the control tower. In case of any anomaly, the SCADA system would alert the main station of the problem apprising it of other details like the severity of the anomaly and measurement values in an organized fashion [1]. The systems may vary from simple, like temperature reporting in a building to complex like monitoring the traffic on many traffic lights.

Elements of SCADA Systems

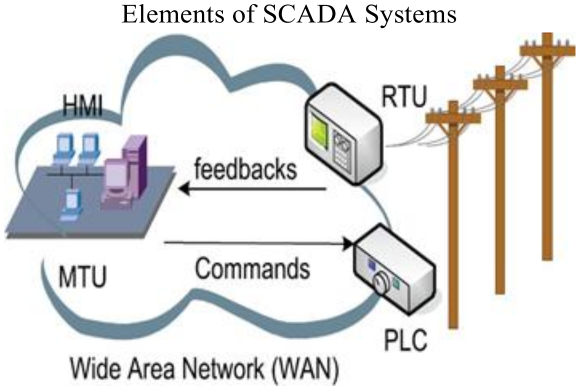

Figure 1.1: Elements of SCADA system

1.1.1 SCADA Master Station Computer Systems

It is the repository of the real-time or near real-time reported data collected from the remote terminal units connected to it. It is generally standard computer hardware equipment and very few SCADA system suppliers have ventured out to make their own computer equipment [3]. A few companies like IBM and CDC did try making hardware for it, but the effort was short lived and commercial off-the-shelf computer products continue to be the main stay. The back-end SCADA software must be able to repeatedly poll the RTUs for data values, should have software for their retrieval, storage and processing [4]. The processing may include unit conversion, cataloguing into tables etc.

1.1.2 Human-Machine Interface

This is the eye candy part on the host station. The values that have been stored in the host computers are presented to the human operator in an understandable and comprehensible form using HMIs. These may provide trending, diagnostic or management information and detailed schematics and animations representing the current states of the machines under its control [3]. Pictorial representation being more understandable to humans is the preferred form in SCADA HMIs.

1.1.3 Remote Terminal Units (RTUs)

An RTU is a normally a transducer or a sensor which allows the electrical circuitry to interface with the process instrumentation and control equipment. The physical parameters like pressure, temperature, humidity etc. are measured through a change in electrical property of some component in the transducer which is indicative of the physical change [3]. A single RTU may measure many different types of parameters. Depending on the values of the measurements, the Input/Output circuitry of a RTU can be analog or digital. Analog corresponds to measurements with a numeric range of continuous values which are later converted using an ADC, like a temperature scale, while digital have limited number of states (generally two) mainly used for flagging. Specific signals can be generated to control process equipment. These days, RTUs are microprocessor-based devices and these conversions are primarily internal to them [3].

1.1.4 Programmable Logic Controllers

The use of microprocessors on RTUs has helped RTUs become smarter with in- creased functionality. PLCs have been built around the philosophy of automation. Reprogram ability being the biggest asset, PLC based RTUs can be debugged and fixed on the field itself along with adding new features like support for multiple polling, exception re- porting, time-tagging etc. This also enables them to execute simple logical processes without involving the master station [3]. Vendors using different type of communication and coding on this equipment has led to standardization of protocols and languages for RTUs too, for example standardized control programming language, IEC 61131-3. These languages require very less training and are based on intuitive approach unlike procedural languages like C and FORTRAN.

1.1.5 SCADA Communication

The conveying of data from an RTU to the master station and commands from the host to the RTU need to be done over a communication system. Also, since a SCADA system might not be localized to just a single plant, the vastness of the network also has to be catered to along with speed, accuracy, security & performance being among other important issues. Before the computer networking solutions were made available, most systems for communication were voice communication based. SCADA communication systems were also built using the same infrastructure and had the same bandwidth limitations. But, with the corporate now wanting to include the SCADA information network into their core networks over security concerns, SCADA systems have also embraced LANs and WANs for seamless integration with everyday office computer networks. This has an advantage for the corporate users that they would not need a separate parallel network for SCADA system.

II. LITERATURE SURVEY

2.1 Architectures

2.1.1 First generation: “Monolithic”

In the first generation, computing was done by mainframe computers. Networks did not exist at the time SCADA was developed. Thus, SCADA systems were independent systems with no connectivity to other systems. Wide Area Networks were later designed by RTU vendors to communicate with the RTU. The communication protocols used were often proprietary at that time. The first-generation SCADA system was redundant since a back-up mainframe system was connected at the bus level and was used in the event of failure of the primary mainframe system [3].

2.1.2 Second generation: “Distributed”

The processing was distributed across multiple stations which were connected through a LAN and they shared information in real time. Each station was responsible for a particular task thus making the size and cost of each station less than the one used in First Generation [3]. The network protocols used were still mostly proprietary, which led to significant security problems for any SCADA system that received attention from a hacker [8]. Since the protocols were proprietary, very few people beyond the developers and hackers knew enough to determine how secure a SCADA installation was. Since both parties had invested interests in keeping security issues quiet, the security of a SCADA installation was often badly overestimated, if it was considered at all.

2.1.3 Third Generation “Networked”

These are the current generation SCADA systems which use open system architecture rather than a vendor-controlled proprietary environment. The SCADA system utilizes open standards and protocols, thus distributing functionality across a WAN rather than a LAN. It is easier to connect third party peripheral devices like printers, disk drives, and tape drives due to the use of open architecture. WAN protocols such as Internet Protocol (IP) are used for communication between the master station and communications equipment [3]. Due to the usage of standard protocols and the fact that many networked SCADA systems are accessible from the Internet; the systems are potentially vulnerable to remote cyber-attacks. On the other hand, the usage of standard protocols and security techniques means that standard security improvements are applicable to the SCADA systems, assuming they receive timely maintenance and updates.

2.2 SCADA Software Features

2.2.1 Database Builder

The Database Builder allows the configuration of data sources to be recorded and displayed in real time. Up to 64 analogue and 128 digital values can be acquired. Each analogue channel can be assigned two alarm set points which can be adjusted during runtime both locally and remotely via web access. Digital channels and logging are also configured using the Database Builder [9].

2.2.2 Local Web Real Time Displays
Analogue channels are automatically displayed as bar graphs. Channels which have exceeded the alarm set points are displayed in red. A separate alarm/event summary page is also available with user configurable on/off state text (e.g. open/closed, up/down). These are fixed displays requiring the user only to complete the database configuration. This allows the system to be up and running in a fraction of the time of similar products.

2.2.3 Historic Trend Viewer
Recorded data can be viewed or exported as a CSV file for use with a standard spreadsheet program. The Trend Viewer allows up to 6 user selectable channels to be displayed. Multiple trend displays can be opened simultaneously. Operator comments can be added to the graph data.

2.2.4 Function Builder
Select functions from a library to perform logical and mathematical operations on acquired data, in real time. The derived output values can also be displayed and recorded. The Function Builder also includes a sequencing language that can run one or more tasks, each executing a series of operations, controlling connected I/O if required.

2.2.5 Graphics Builder
Unlimited number of custom mimic animated displays. It supports layering to reveal increased levels of detail when zooming in.

2.2.6 Graphic Viewer
It allows display of custom mimics. When installed on a remote PC, custom mimics can be viewed via web access.

III. SYSTEM DEVELOPMENT

System architecture is the conceptual model that defines the structure, behavior, and more views of a system. An architecture description is a formal description and representation of a system, organized in a way that supports reasoning about the structures of the system.
System architecture can comprise system components, the externally visible proper- ties of those components, the relationships (e.g. the behavior) between them. It can provide a plan from which products can be procured, and systems developed, that will work together to implement the overall system.
The basic concept was to create a system which can perform like supervisory control and data acquisition system for monitoring and controlling the Industrial equipment. This system implements two ways for application remote monitoring and command, the first is based on SMS/call functionalities provided by GPRS network and the second one is accomplished by using a web page interface provided when using a web server [18].

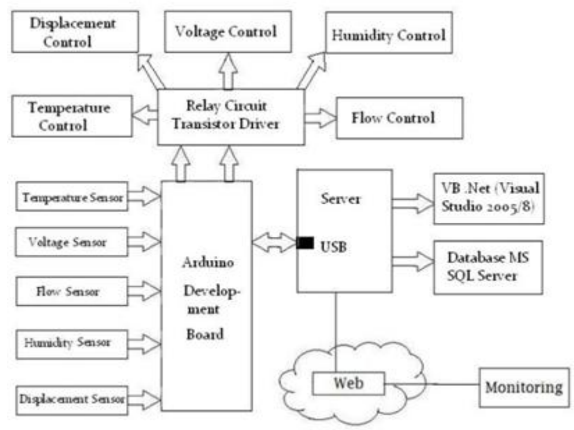

Figure 3.1: Proposed System

IV. SOFTWARE DEVELOPMENT

The .NET Framework is the infrastructure for the new Microsoft .NET Platform. The .NET Framework is a common environment for building, deploying, and running Web applications and Web Services. The .NET Framework contains a common language run- time and common class libraries like ADO .NET, ASP .NET and Windows Forms to provide advanced standard services that can be integrated into a variety of computer systems. The .NET Framework provides a feature-rich application environment, simplified development and easy integration between a number of different development languages. The .NET Framework is language neutral. Currently it supports C++, C, Visual Basic, and Script.
The Arduino software is also open-source. The source code for the Java environment is released under the GPL and the C/C++ microcontroller

libraries are under the LGPL. The Arduino language is merely a set of C/C++ functions that can be called from your code. Your sketch undergoes minor changes (e.g. automatic generation of function prototypes) and then is passed directly to a C/C++ compiler. All standard C and C++ constructs supported avr g++ should work in Arduino.

IIS (Internet Information Server) is a group of Internet servers (including a Web or Hypertext Transfer Protocol server and a File Transfer Protocol server) with additional capabilities for Microsoft's Windows NT and Windows 2000 Server operating systems [23]. IIS is Microsoft's entry to compete in the Internet server market that is also addressed by Apache, Sun Microsystems, O'Reilly, and others. With IIS, Microsoft includes a set of programs for building and administering Web sites, a search engine, and support for writing Web-based applications that access databases Microsoft points out that IIS is tightly integrated with the Windows NT and 2000 Servers in a number of ways, resulting in faster Web page serving.

V. ALGORITHM

A. Monitoring Module

1. Start.
2. Enter the User ID and Password given to Administrator.
3. Monitor the screen.
4. Check whether a temperature sensor value is changing according to temperature varies. Check same for other sensors also.
5. Different values will be displayed according to the status of that parameter.
6. If you want to know about the history, click on view logs button. The whole database will be displayed along with the Sensor values along with the date and time of parameter change.
7. Logout of your account by clicking exit button.
8. Stop.

B. Online Monitoring Module

1. Start.
2. Enter the URL/link.
3. Enter user ID and Password.
4. Monitor the screen.
5. Different values will be displayed according to the status of that parameter.
6. If the value goes above set point, proper control is fired.
7. Send SMS to authenticated person using web API.
8. Stop

VI. SOFTWARE TESTING

A. Monitoring Module

1. Login Form

Login form is used for accessing the personal account of the administrator using his own username and password. It appears as shown in Figure 4.1.

Figure 4.1 Login Form

2. Monitoring Page

After an administrative person login with his own username and password his account will be opened where he can see the various parameters values at that time. He can see previous logs also. It appears as shown in Figure 4.2.

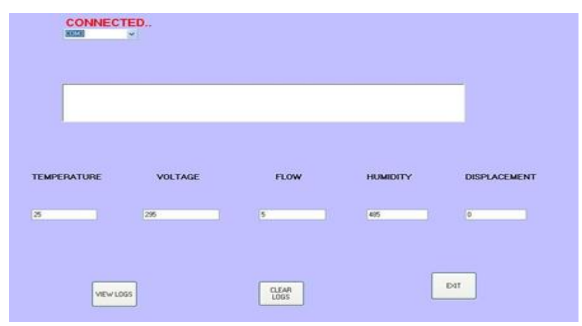

Figure 4.2 Monitoring Form

3. View logs Page

Administrative person can see previous logs by clicking view logs button. The whole database will be displayed along with the Sensor values along with the date and time of parameter change. It appears as shown in Figure 4.3.

TEMPERATURE	VOLTAGE	FLOW	HUMIDITY	DISPLACEMENT	EVENT_OCCUR
28	274	5	493	0	7/16/2015 7:25 ...
27	275	5	493	0	7/16/2015 7:25 ...
30	274	5	487	0	7/16/2015 7:25 ...
28	275	5	490	0	7/16/2015 7:25 ...
26	274	6	491	0	7/16/2015 7:25 ...
25	274	5	494	0	7/16/2015 7:25 ...
25	274	5	496	0	7/16/2015 7:25 ...
26	274	5	496	0	7/16/2015 7:25 ...
36	274	5	496	0	7/16/2015 7:25 ...
32	274	6	495	0	7/16/2015 7:25 ...
26	274	5	492	0	7/16/2015 7:25 ...
28	274	5	489	0	7/16/2015 7:25 ...
27	274	5	485	0	7/16/2015 7:25 ...
28	274	6	488	0	7/16/2015 7:25 ...
26	274	5	490	0	7/16/2015 7:25 ...
32	274	5	494	0	7/16/2015 7:25 ...
26	274	5	497	0	7/16/2015 7:25 ...
26	274	5	497	0	7/16/2015 7:25 ...
31	274	5	496	0	7/16/2015 7:25 ...
25	274	6	495	0	7/16/2015 7:25 ...
27	274	5	492	0	7/16/2015 7:25 ...
30	274	5	489	0	7/16/2015 7:25

Figure 4.3 View logs Page

B. Website Module

1. Login Form

Login form is used for accessing the website using a unique username and password only by the Administrator. This website cannot be accessed by using static IP. It appears as shown in Figure 4.4.

WELCOME TO ONLINE SCADA

Figure 4.4. Web Login Form

2. In this form the status of the various parameters is monitored i.e. it has value below set point. In addition, with this administrator can see whether the control action is fired or not when value exceeds set point. It appears as shown in Figure 4.5

WEB-BASED SCADA GRAPHICAL VIEW PANEL

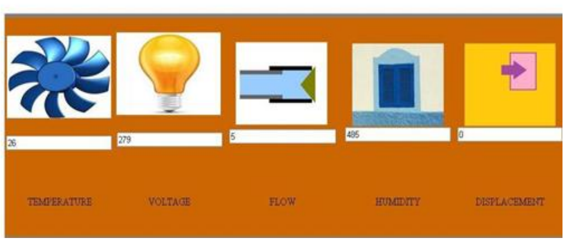

Figure 4.5: Web Monitoring Page

VII. RESULTS

1. Extraction of sensed parameters from the various sensors.
2. Continuous monitoring on the sensor values.
3. Proper supervisory control action depending on set points for a particular parameter.
4. Accurate graphical display on web-based environment.
5. SMS sending facility to authenticated person.

VIII. CONCLUSION

The Web-based SCADA is the area of research nowadays. Here we developed a small SCADA like simple automated application which senses the real time environmental parameters and comparing the values with set point it fires a control action automatically. On the field side all the sensors lie which continuously monitors on real time environment. All these sensor values are given to the server with the new technology Arduino! Arduino is the open-source hardware as well open-source software platform. As it is having analog as well as digital input and output pins feeding the input of sensor values from analog pins is much easier. It is also having a USB port. So, communication with the computer system is done easily.

REFERENCES

[1] Antonio M. Grilo, Jaime Chen, Manuel Daz, Daniel Garrido, and Augusto Casaca, “An In-targeted WSAN and SCADA System for Monitoring a Critical Infrastructure”, IEEE Trans- actions On Industrial Informatics, Vol. 10, No. 3, pp. 1755-1764, August 2024.

[2] B. Galloway and G. P. Hancke, “Introduction to industrial control networks”, IEEE Commun. Surveys Tuts., vol. 15, no. 2, pp. 860-880, May 2013.

[3] S. A. Boyer, “SCADA Supervisory Control and Data Acquisition”, Englewood, CO, USA: ISA-International Society of Automation, Jun. 15, 2009. ISBN: 978-1-936007-09.

[4] B. Marchi, A. Grilo, and M. Nunes, “DTSN: Distributed transport for sensor networks”, in Proc. IEEE Symp. Compute. Commun. (ISCC’07), Aveiro, Portugal, Jul. 2007, pp. 165-172.