GUI BASED INDUSTRIAL ANNUNCIATING SYSTEM WITH MODBUS INTERFACE

Sudhanshuvinayak Puranik, Ankit Dhyani, Hardik Ahmedabadkar, Rahul Shaiwale

BITS edu campus, Vadodara

ABSTRACT

The annunciating system is the backbone of any process industry as it allows technicians to remotely monitor the ongoing processes and keep vital stats of the system in check. The purpose of this project is to improve upon currently available annunciating systems using mass-produced low-cost ARM assemblies like Raspberry Pi, BeagleBone, etc. These assemblies will help us in addressing our needs of annunciation. The annunciating systems we plan to make will have data-logging facility and remote data acquisition for SCADA systems. These systems will be capable of receiving analog inputs along with digital ones. The system would be designed in such a way that it could be reconfigured by the user as per his needs after the initial setup. So, our designed system will have all the functionalities of the current systems along with many improvements on current drawbacks.
Keywords: Annunciating System, SCADA, Data Logger, Analog Graphing

1. INTRODUCTION

In any industry, there are certain device parameters that have to be constantly monitored to ensure the smooth functioning of the machines and safety of the workers. This data is fed to a centralized station where a technician keeps a watch and notifies the subordinates in case of any emergency or malfunctioning. These systems are interfaced with different sensors to monitor different parameters like pressure, temperature, RPM, etc.

2. ANNUNCIATING SYSTEM

An annunciating system is basically a set of labels that provide information on the state or condition of devices, machineries, processes, etc. which are being used in a plant.

As mentioned in the abstract, an annunciator is a vital part of any process industry or an automated plant, as it allows the maintenance engineer to remotely monitor the condition of currently ongoing processes. As annunciating system is being used in the plant, the maintenance engineer does not have to go to each and every machine or device to monitor its vital stats like temperature, pressure, speed, etc. He can monitor the machinery of the plant sitting at one place.

These systems are widely used in power plants (thermal, nuclear, hydel), steel plants and aircraft.

2.1   Drawbacks of Current Annunciating System

1) Use of LEDs and Translucent Film as Alarm Labels makes these systems Non-Reconfigurable
2) No Data Logging or Remote Data Acquisition Facility
3) Lack of Analog Inputs
4) High Cost of Imported Annunciating System

2.2   Solutions to Current Problems

1) Use of GUI display for alarm labels
2) Making Data-Logging and Remote Data Acquisition available.
3) Providing of Analog Inputs on the system
4) Use of mass-produced assemblies to keep the cost low

3. OBJECTIVES

- The aim of this project is to ease out the Control and Instrumentation process by making better Annunciating Systems which have Analog Inputs, Data-Logging and Data- Acquisition facilities.
- The system should be easy to use and modify, it should also be compatible with currently used industrial standards like its MODBUS. So, it can replace them easily.
- The price of the system should be low, so it can serve as a viable low cost alternative to SCADA system.
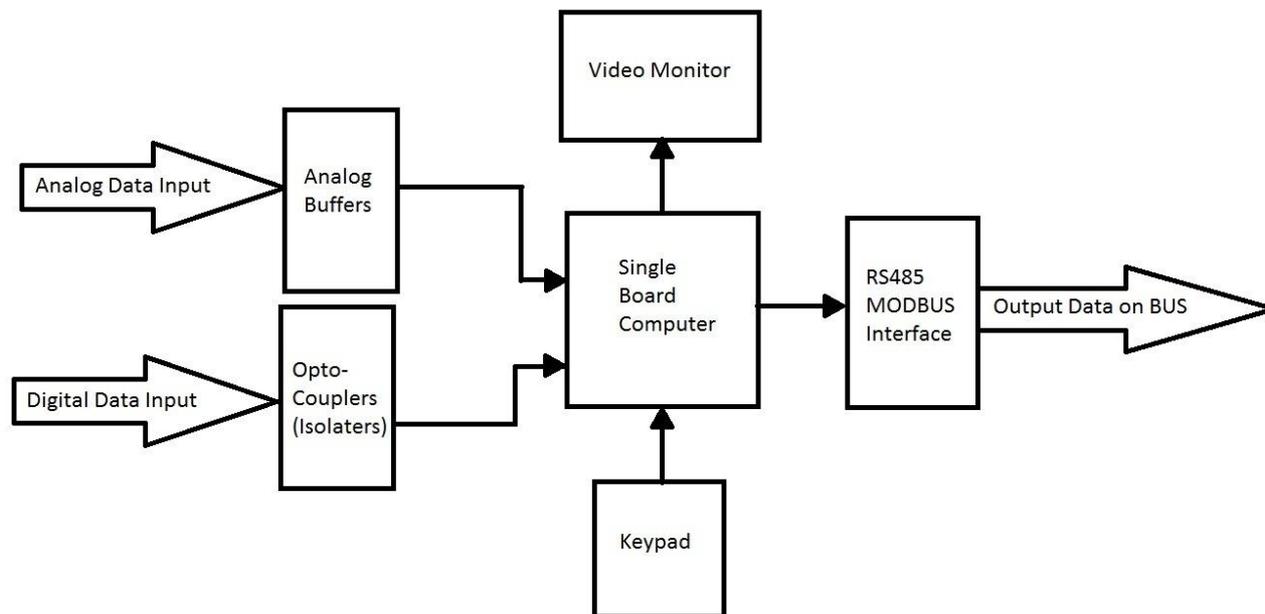
4. SYSTEM DESCRIPTION

Figure 1: Block Diagram

1) Analog Data Input

The data from transducers like temperature sensors, pressure sensors, flow sensor, etc.

2) Digital Data Input

Data from digital inputs like switches, relays, gates, micro controllers, etc.

3) Opto-Couplers (Isolators)

The optocouplers provide electrical isolation which protects the circuit from damage. It is mainly used for digital inputs.

4) Analog Buffer

These can be either op-amps providing isolation or A/D converter which would provide digital data for analog quantity. It also serves as a buffer which does not allow sensors to get loaded by other circuits.

5) Single Board Computer (SBC)

This is the brain of the system, which controls all the peripherals and processes the input data; like Raspberry Pi, BeagleBone Black, etc.

6) Video Monitor

This is the display which provides us with a visual medium through which we interact with the system like LED display with HDMI interface, etc.

7) Keypad

This is the device through which the user interacts with the system.

8) RS485 MODBUS Interface

The processed data can be provided to other parallel running systems like SCADA or other Master Units, etc. for Process/Control on the BUS.

The price breakdown of system

Table 1: Cost of System

Component	Price (Rs)
Raspberry Pi	2500
A/D Converter	1000
Video Monitor	5000
Digital I/Os	500
RS485 Cable	100
Miscellaneous	500
Total	**9600 /-**

International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org
Volume 4, Issue 9, September 2015   **ISSN 2319 - 4847**

5. OVERALL WORKING OF SYSTEM

As shown in flowchart the system is of event driven type. The GUI of the system is designed in Python which runs on Linux which is ported on Raspberry Pi. Analog inputs come from A/D converter in this Arduino is used for this purpose, it communicates with Raspberry Pi serially. GPIO's on Raspberry Pi are programmed as inputs. In this way the alarm inputs are both analog and digital type.
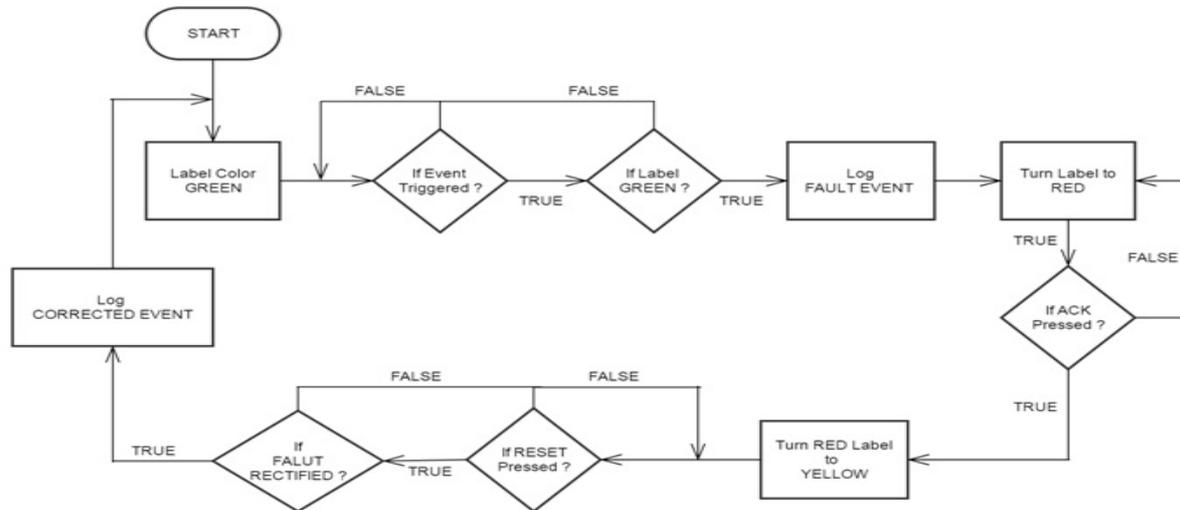

Figure 2: Flow Chart

One can see from the flowchart whenever an alarm is triggered (event) meaning either analog value has crossed the threshold or digital input has become high, a chain of events take place till the fault is solved. In this way the whole annunciating system works.

Screenshot of GUI designed for this shown in Figure 3. The GUI is color coded and has tabs which indicate the current state of ongoing functions. If the value of a particular parameter exceed s (or falls below) a pre-defined limit, the alarm is triggered off and the color of the tab allocated to that function changes from green to red. At the same time the time stamp of the event is saved and displayed on the screen to help the technician understand the chain of events. The buzzer can be muted and the alarm acknowledged in which case the color of the tab changes from red to yellow. When the problem is rectified, the color of the tab changes back to green.
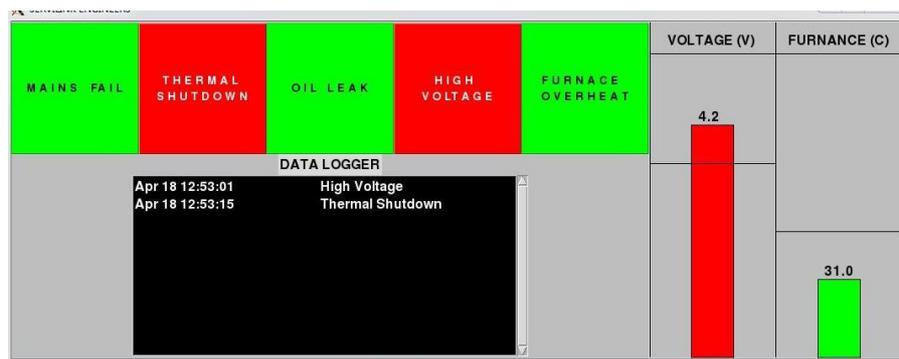

Figure 3: GUI of the System

6. Conclusion

Through this project we have successfully developed an annunciating system with HMI (Human Machine Interface). This system overcomes all the major drawbacks of currently used annunciators like lack of reconfigurablity, analog inputs and data logging. These systems have been made by keeping in mind back compatibility so it's easy to replace them with current systems. The overall expenditure on the development is only a fraction of the costs of current systems.

References

[1] Gary W. Scheer ,"Roles of Annunciators in Modern Electrical Substations", 13th Annual Western Power Delivery Automation Conference, Spokane, Washington,March-2011

[2] "Modicon Modbus Protocol Reference Guide", Modicon Inc.,1996
[3] Zed Shaw, "Learning Python the Hard Way 3rd Edition", Addison-Weasley[TM]
[4] Donat Wolfram, "Learn Raspberry Pi Programming with Python", Technology in Action[TM]
[5] Matt Richardson & Shawn Wallace, "Getting Started with Raspberry Pi", O'Reilly[TM]
[6] Lambert Kenneth, "Easy GUI Programming in Python ", Amazon Digital[TM]
[7] Massimo Banzi , "Getting Started with Arduino 2nd Edition", O'Reilly[TM]