

Large Scale Industrial Temperature Monitoring and Control System using Labview Through USB (Manual and Automatic Modes)

Ponnaganti Naganjaneyulu Balla Pavan Kumar Naga Harikrishna Vadisala
 Prof. B. Karthikeyan
 Sense School, Vellore Institute of Technology,
 Vellore.

Abstract: The temperature monitoring and control system can be done in computer (PC) using Laboratory Virtual Instrumentation Engineering Workbench (LABVIEW) software. To obtain the quality of service in the industries, data acquisition plays an important role. The main purpose of using LABVIEW is for data acquisition. The temperature is measured by the temperature sensor (LM-35) and the obtained analog signal is processed by the microcontroller (LPC-1768). The LABVIEW acquires data from the microcontroller through USB port for further processing of the data. The data is processed and displayed in the LABVIEW. The system control and automation can be done through the help of control circuitry in the LABVIEW. There are two modes, "automatic" and "manual", in which either one can be selected in the LABVIEW. The temperature values obtained can be stored into the excel sheet through the LABVIEW.

Keywords: LABVIEW, LPC-1768, VISA 5.2 driver, LM35 temperature sensor, L293D motor driver, automatic mode, manual mode.

1. INTRODUCTION

In the industries, several techniques were used for the temperature monitoring. Ethernet, Wireless LAN etc are the previous methods used for the industrial application. But they require data acquisition cards which increase the cost to the system. Hence we are opting for USB interface which does not require acquisition cards which reduces of cost to the system. The software gives the flexibility to program as per the requirements. Software programs are used to change the data process. The LABVIEW software is used to process the data that is obtained from the external devices such as sensors. This paper is about temperature monitoring and control system in industrial applications in two modes (automatic and manual) along with data storage(excel sheet).

METHODOLOGY:

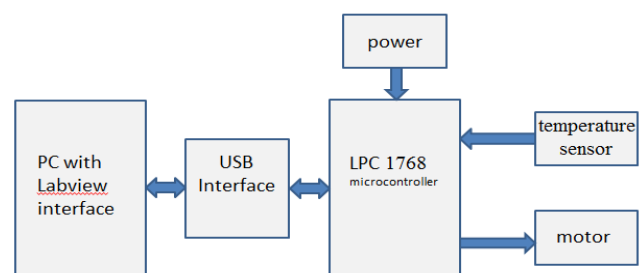


Figure1. Block Diagram.

The above figure shows the block diagram of the temperature monitoring system using LABVIEW software. Firstly the LPC-1768 microcontroller detects the millivolt value of temperature from the sensor LM35. Then LPC 1768 converts the analog values to digital values by ADC converter which is present in pin 15 to pin 20. Then the obtained digital signal is send to PC with LABVIEW through USB interface. In LABVIEW the temperature conversion is done. Based on the obtained temperature the control signals are generated and sent to LPC-1768. Further controlling actions is done by LPC-1768 to control the temperature. VISA 5.2 driver is used to establish the communication between LABVIEW and LPC-1768. The digital values are sent from LPC-1768 to LABVIEW by using some graphical interface in the LABVIEW. That graphical interface is called as READ block.

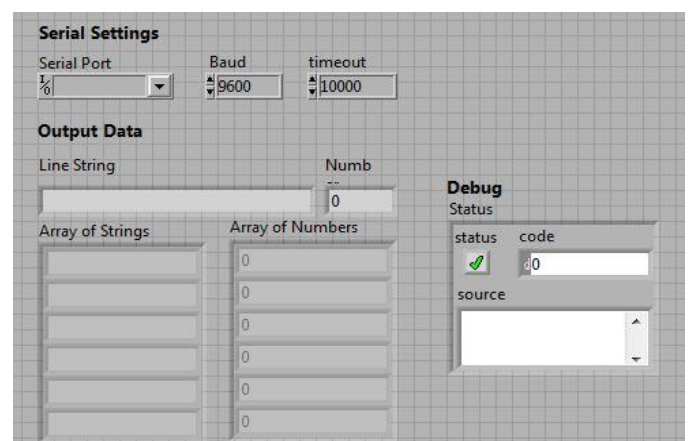


Figure2. front panel of LPC 1768 read to LABVIEW.

In the front panel of LABVIEW READ, select the serial COM port. Set the baud rate and timeout value. The value sent by connected device (LPC1768) is displayed in the line string, array of strings, array of numbers and number indicator (based on the type of value).

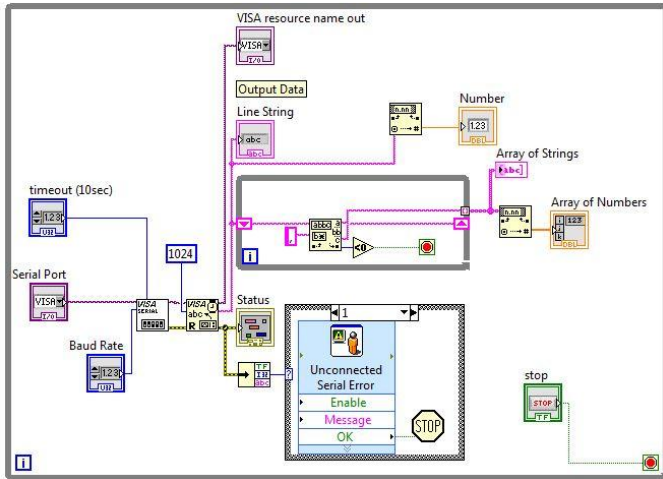


Figure3. Block Diagram of LPC 1768 read to LABVIEW.

In the block diagram of LABVIEW READ, the connections are done as shown in above for the read operation from LPC-1768 to LABVIEW. The entire read block is made into SUB-VI for the user flexibility.

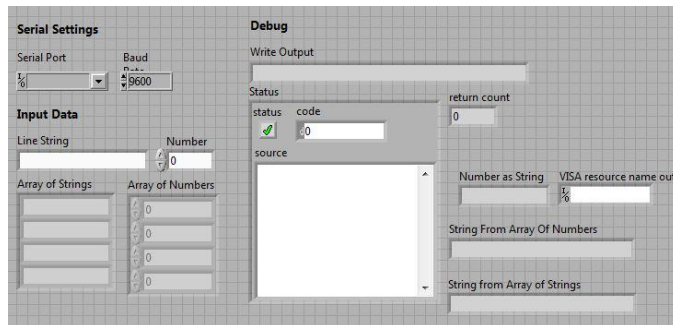


Figure4. front panel of LPC 1768 write to LABVIEW.

In the front panel of LABVIEW WRITE, select the serial COM port. Set the baud rate. Write to the connected device(LPC 1768) by typing the desired values on the string, array of numbers and number indicator (based on the need).

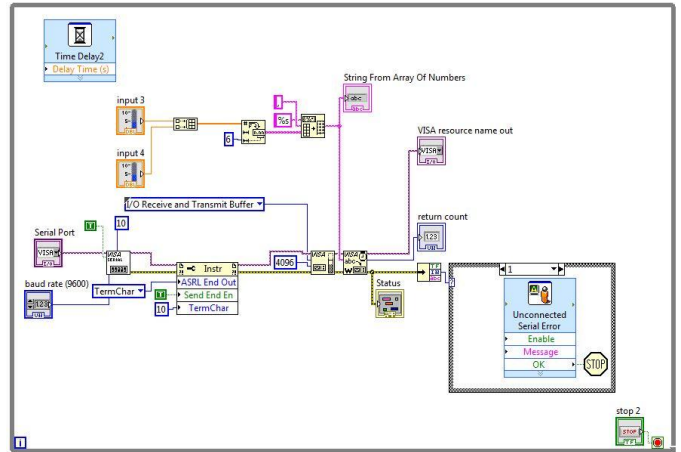


Figure5. Block Diagram of LPC 1768 write to LABVIEW.

In the block diagram of LABVIEW write, the connections are done as shown above for the LABVIEW write to LPC-1768. The entire write block is made into SUB-VI for the user flexibility[10].

In this paper there are two modes of operation. They are:

1. Manual mode
2. Automatic mode

These modes provide the flexibility to the user according to the needs. The mode can be selected in the front panel (manual / automatic) by using enum variable.

MANUAL MODE OF OPERATION:

In the front panel we select manual mode from enum variable. Select the serial COM ports (one is for read purpose and other is for write).In the manual mode of operation, the user provides the control signal from the LABVIEW. Control signal is nothing but LED values which control the hardware connected to LPC-1768.

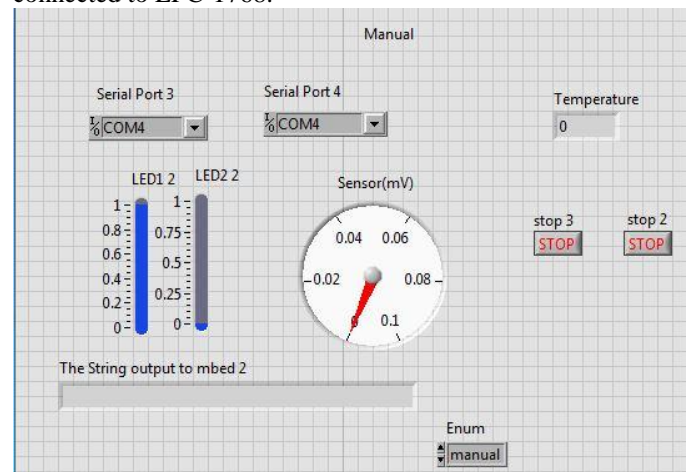


Figure6. front panel of manual mode of operation.

In the front panel of the manual mode, select the connected device for read and write purposes. The gauge displays the sensor value (in mV) that is obtained from the sensor LM35 through LPC-1768. The sensor value is converted into temperature in degree centigrade by:

$$\text{Temperature} = (\text{sensor value} / 3.3) * 1000;$$

Based on the obtained temperature, the control signals are generated by user (1,0) and written to LPC-1768. LPC-1768 controls the motor by the obtained write values from LABVIEW. If (0,0) is send from LABVIEW then motor is in stop condition. If (1,0) the motor runs.

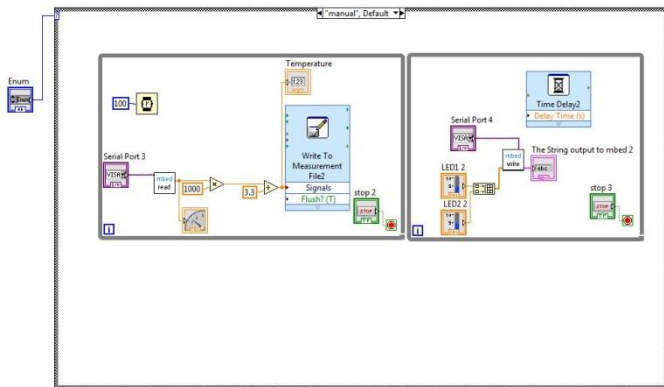


Figure7. Block Diagram of manual mode of operation.

In the block diagram of manual mode, the connections are done for reading from and writing to LPC-1768 by using read and write SUB-VIs. The obtained temperature values are stored in the excel sheet by “write measurement” block.

AUTOMATIC MODE OF OPERATION:

In the front panel, we select automatic mode from enum variable. Select the serial COM ports (one is for read purpose and other is for write).In the automatic mode of operation, the LABVIEW itself provides the control signal to LPC-1768.

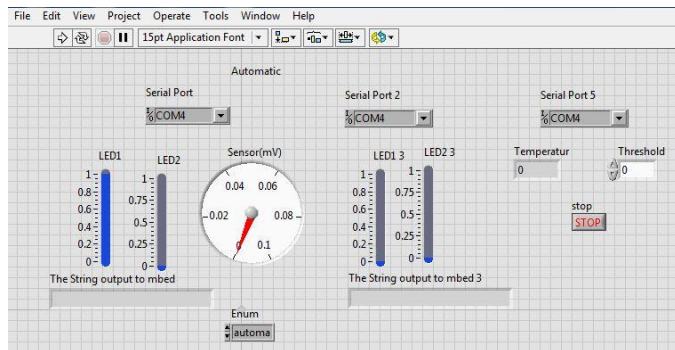


Figure8. Front panel of automatic mode of operation.

In the front panel of automatic mode, select the connected device for read and write purposes. The temperature conversion is same as in the manual mode. The threshold value is set in front panel. The temperature is compared to the threshold and based on that control signals are generated.

CASE1:

If the temperature is greater than the threshold, then control signals (1,0) are written to LPC-1768 for driving the hardware connected to LPC-1768.

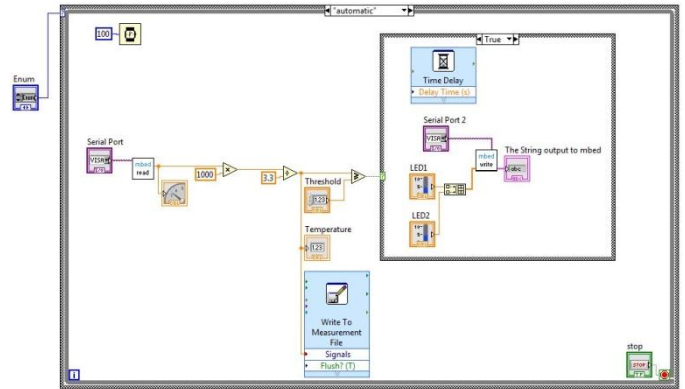


Figure9. Block Diagram of automatic mode of operation for case1.

The connections are done as shown above for the case1 in automatic mode. The obtained temperature values are stored in the excel sheet by “write measurement” block.

CASE2:

If the temperature is less than the threshold, then control signals (0,0) are written to LPC-1768 for the motor to stop/no motion.

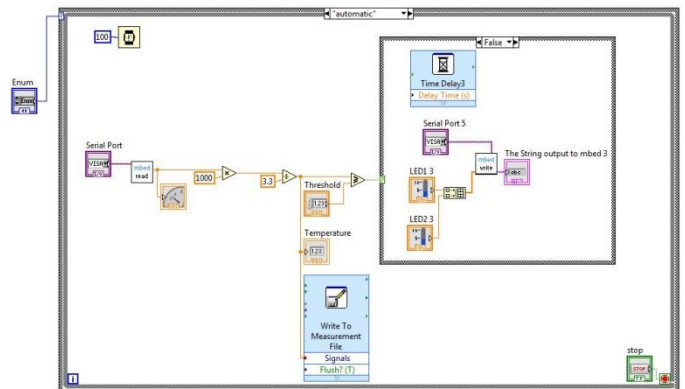


Figure10. Block Diagram of automatic mode of operation for false case.

The connections are done as shown above for the case2 in automatic mode. The obtained temperature values are stored in the excel sheet by "write measurement" block.

RESULTS:

The temperature monitoring is done with LABVIEW software in two modes (manual and automatic). In LABVIEW, temperature in centigrade is obtained after processing the sensor values (in mV). The temperature values are stored in an excel sheet (one value per 250 mS).

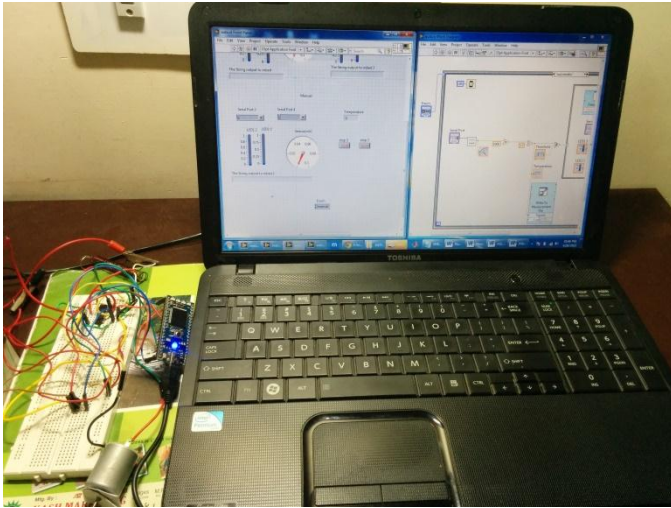


Figure9. LABVIEW block diagram design

The inputs to the motor is provided by LABVIEW through LPC-1768. The motor is rotated when LABVIEW sends LED values of 0,1.

CONCLUSION:

At first we have to establish the communication between LABVIEW and LPC-1768 through USB interface with the help of VISA 5.2 driver for LABVIEW and windows mbed serial driver for LPC-1768. Reading from temperature sensor LM35 which is connected to LPC-1768, to the LABVIEW is done. The temperature values are stored in excel sheet. Writing from LABVIEW to hardware (motor) connected to LPC-1768 is done. In manual mode, user manually control the hardware (motor) connected to LPC-1768, based on temperature values. In automatic mode, LABVIEW itself control the hardware (motor) connected to LPC-1768. In future this project can be extended with Bluetooth module so that user flexibility will be increased.

REFERENCES:

- [1] S. Rajesh Kumar and S. Ramesh kumar, "INDUSTRIAL TEMPERATURE MONITORING AND CONTROL SYSTEM THROUGH ETHERNET LAN", M.E Communication System, PGP College of Engineering and Technology, India
- [2] Nie Chun-yan¹, Xu Shan-shan², and Ji Shu-jiao¹, "Data Acquisition and Realization of Communication Transmission Based on LabVIEW", 2012 International Conference on Computer Science and Electronics Engineering, DOI 10.1109/ICCSEE.2012.191.
- [3] Shyam.R.Nair, Venkatesh Thalaimalayan, "Design of a LabVIEW Based Real Time Simulator for ABS Tester", International Journal of Scientific and Research Publications, Volume 2, Issue 5, May 2012.
- [4] Subhransu Padhee, Yaduvir Singh, "Data Logging and Supervisory Control of Process Using Lab VIEW", Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January, 2011, IIT Kharagpur.
- [5] Guangling Guo, Zhiqiang Li, Fan Yang, "Design of High Speed Pulse Data Acquisition System Based on FPGA and USB", 2011 International Conference on Multimedia Technology (ICMT).
- [6] Nader N. Barsoum, Pin Rui Chin, "Ethernet Control AC Motor via PLC Using LabVIEW", Intelligent Control and Automation, 2011, 2, 330-339, Published Online November 2011 (<http://www.SciRP.org/journal/ica>).
- [7] Piotr Bilski and Wieslaw Winiecki, "Virtual Spectrum Analyzer Based on Data Acquisition Card", IEEE transactions on instrumentation and measurement, vol. 51, no. 1, february 2002.
- [8] A. Ferrero, L. Cristaldi and V. Piuri, "Programmable Instruments, Virtual Instruments, and Distributed Measurement Systems: What Is Really Useful, Innovative, and Technically Sound", IEEE Instrum. Meas. Mag., vol. 2, issue 3, pp. 20-27, Sept. 1999.
- [9] Gary W. Johnson, Richard Jennings, "LabVIEW Graphical Programming", 4th ed. McGraw-Hill.
- [10] www.ni.com