A WEB-BASED APPROACH TO AUTOMATED INSPECTION AND QUALITY CONTROL OF MANUFACTURED PARTS

Immanuel Edinbarough, Manian Ramkumar, Karthik Soundararajan

The University of Texas at Brownsville/Rochester Institute of Technology/Alliance Automation Systems

Abstract

This paper presents an approach for the automated inspection and quality control of manufactured parts from a remote site. A web-based control system was developed to establish communication among the various components of the experimental manufacturing cell, consisting of an IBM 7535 robot, a measuring station and a part feed station. The various cell components are interfaced with the computer through digital and analog interface modules from National Instruments and is controlled using a graphical programming language called LabVIEW. A dedicated web page was developed for the cell that allows remote users to perform online quality tests. Details pertaining to the cell and the control architecture developed to automate the quality control task are presented in this paper. The feasibility of providing non-site based hands-on engineering and technology education is also explored through this project.

I.   Introduction

The advent of the world-wide-web has tremendously influenced the way in which industry and academia perform various activities. More and more academic institutions are starting to recognize the vital link that the web provides between the faculty and the students.[1] The motivation for this work comes from the awareness of the Internet and its innumerable applications in the manufacturing industry. There are several applications already developed or under development, to harness the power of the Internet for manufacturing applications. One such application that we have explored is the use of the Internet to perform basic dimensional measurements on components and do quality checks, from a remote site. This approach to quality control will eliminate the mandatory presence of an operator near the automated manufacturing cell, but still be able to monitor the production and quality of the parts. Academic institutions can use this technique to offer quality control and related courses on-line for distance delivery. In the following sections we will be presenting details pertaining to the necessary tools and a methodology for developing a web-based quality control system.

II.   Background:

The World Wide Web is currently a firmly established, though virtual, reality. A few years after its impressive breakthrough, from limited professional circles to everyone's working and social life, the web constitutes an additional space in which people can communicate, work, trade, or spend leisure time. And increasingly, too, it is a place to learn.[2,3]

Educators (teachers, developers, researchers, students), well aware of the potential of web technology, have adopted it for creating new learning environments, thus yielding a huge repertoire of educational web sites. The rationale behind this creative endeavor is the expectation that unique features of the technology (e.g., powerful information manipulation tools and communication means) will substantially contribute to the teaching and learning processes.[4]

The use of the web as a training medium for lab-based courses still faces a myriad of challenges. Several educational institutions are currently working to address this problem and are still in the developmental stages. Web-based training programs which address the issue of controlling CNC, robot and metrology equipment, through the internet, are still considered new ideas and need consistent effort to make it a viable solution. The work presented in this paper will address some of the critical issues and some basic tools currently available for implementing this technology. New and emerging tools will provide incredible opportunities to expand this technology to new levels, in the near future.

III.   Experimental Manufacturing Cell

The experimental cell consists of a robot, a measuring station, a gravity feeder, a computer and field input/output (I/O) interface modules (Figure 1). The measuring station consists of two pneumatic cylinders, one to clamp the part and the other attached to an LVDT to take the measurement of the part feature. There is also an optical sensor, which senses the presence or absence of the part on the measuring station. The field I/O interface consists of three modules, the network module, the input module and the output module. These modules act as the interface between the computer and the other components of the experimental cell.

IV.   Hardware Aspects of the Cell

Material Handling System

An IBM 7535 robot is used for material handling. The task includes picking parts from the feeder and placing it on the measuring station and subsequently moving it to the storage bin. The digital I/O of the robot is interfaced with the computer through the field I/O modules. The end of arm tooling is a gripper actuated by solenoid valves. The robot controller and program continuously maintains the robot in a ready state, to perform its task sequence immediately upon receiving the START signal from the field I/O modules.

Linear Variable Differential Transformer (LVDT)

An LVDT is used to measure the dimension of the manufactured part. This is mounted on the measuring station of the cell. It is interfaced with the computer through the input module of the field point device.
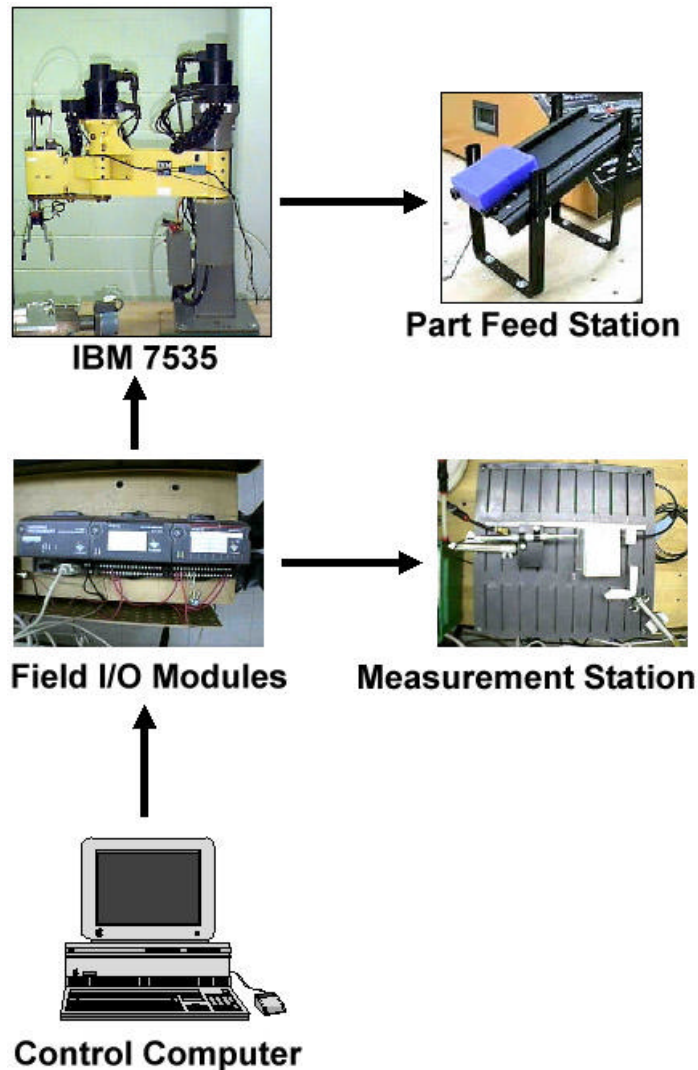

Figure 1. Components of the Experimental Cell

Fixture

The part holding fixture is located on the measuring station. The fixture is operated by two pneumatic cylinders, which receive the actuation signal through the field output module.

V. System Integration Aspects of the Cell

The various hardware components of the cell, discussed in the previous section, are integrated by the use of sensors and digital I/O signals. These signals are monitored by the data acquisition system, which takes appropriate action based on the user program. The action is in the form of field output module activation or deactivation. This provides control signal to the field output devices and signal to the robot. The user program, developed using LabVIEW, controls the various activities of the cell and manages the overall integration of the components of the system.

VI. Web Based Control Architecture

The user can access the experimental cell and the inspection system, from a remote site, through the Internet. The web based control architecture designed for this application is shown in Figure 2. The control computer runs the web server and LabVIEW software. The field I/O network module, which enables the I/O interface, is connected to the control computer through serial communication. The input module of the field I/O module receives signals from the part present sensor and measurement voltage from the LVDT. The output module sends signals to control the part clamping fixture, the robot and the pneumatic cylinder. This cylinder provides linear displacement to the LVDT probe.
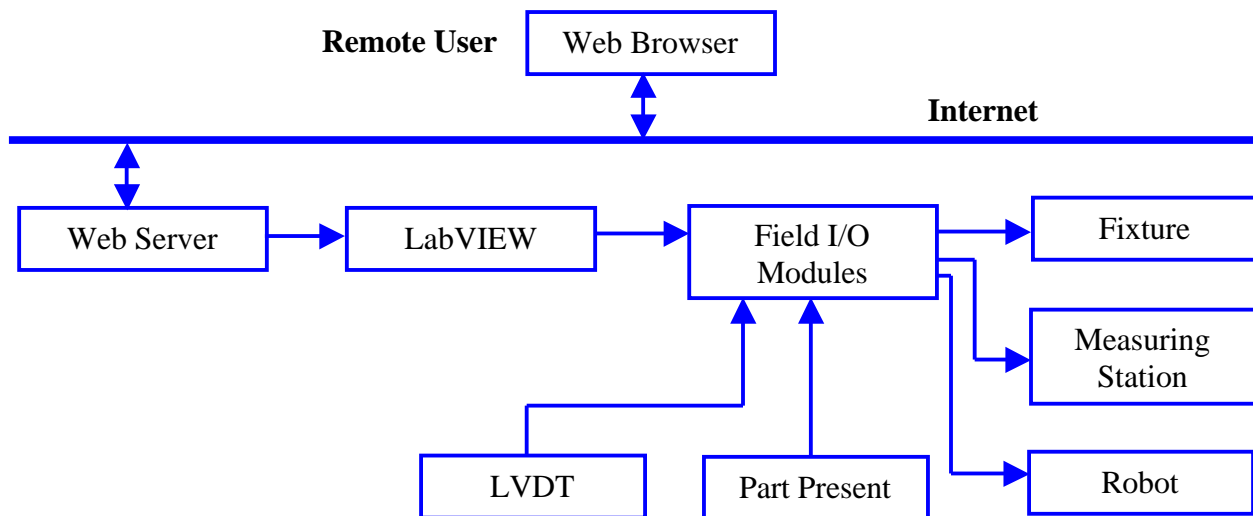

Figure 2. Web Based Control Architecture

The system integration enables the cell control computer to perform the sequence of tasks outlined in Figure 3, automatically, when the user starts the experiment from a remote site.

VII. Programming and Control Aspects of the Cell

The cell control is primarily accomplished by the use of Virtual Instruments (VIs) created using LabVIEW. These LabVIEW VI's along with the AppletVIEW VIs and robot programs enable information exchange between the various levels of the control architecture.
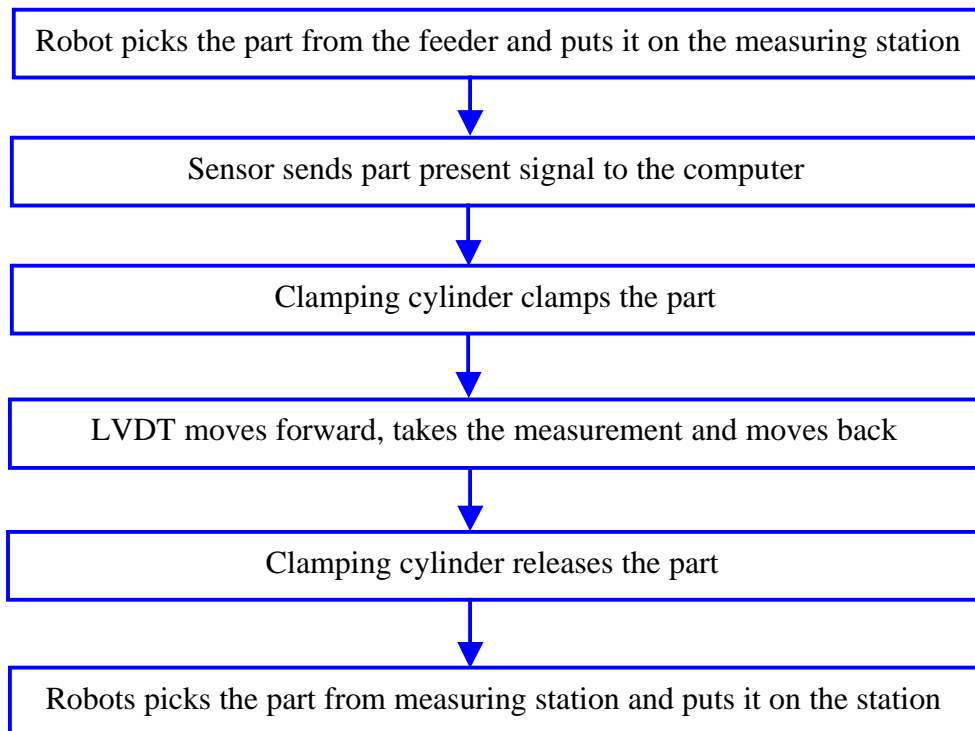
```
┌─────────────────────────────────────────────────────────────────────┐
│  Robot picks the part from the feeder and puts it on the measuring station  │
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼
┌─────────────────────────────────────────────────────────────────────┐
│           Sensor sends part present signal to the computer           │
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼
┌─────────────────────────────────────────────────────────────────────┐
│                   Clamping cylinder clamps the part                  │
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼
┌─────────────────────────────────────────────────────────────────────┐
│          LVDT moves forward, takes the measurement and moves back    │
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼
┌─────────────────────────────────────────────────────────────────────┐
│                  Clamping cylinder releases the part                 │
└─────────────────────────────────────────────────────────────────────┘
                                    │
                                    ▼
┌─────────────────────────────────────────────────────────────────────┐
│   Robots picks the part from measuring station and puts it on the station   │
└─────────────────────────────────────────────────────────────────────┘
```

Figure 3. Task Sequence

This forms the software framework for the cell (Figure 4). This section explains various software used and the information exchange strategy.

Level 3

This is the highest level of the software framework and it controls the entire cell and also provides a link to user, through the web. It consists of the following components:

- LabVIEW

LabVIEW is the abbreviation for Laboratory Virtual Instrument Engineering Workbench. It is powerful and flexible instrumentation and analysis software developed by National Instruments Corporation. LabVIEW programs are called Virtual Instruments or VIs. LabVIEW is different from text-based programming languages in that it uses a graphical programming language, known as the "G" programming language. The principle that governs this type of programming is known as data flow. This is to say that the executable elements of a VI execute only when they have received all the required input data. In other words, data flows out of the executable element only after the code is finished executing. LabVIEW also has an extensive library of virtual instruments and functions for common control systems such as PLCs to aid programming.

- AppletVIEW

In order to operate and control the experiment from the Internet, communications need to be established between the control computer and the remote user. This is done using software called

AppletVIEW. AppletVIEW is an application kit for creating Java applets that communicate with LabVIEW. Using AppletVIEW, LabVIEW programmers can easily create Java applets, which will interact over the web with the LabVIEW VIs. AppletVIEW provides two way communications between the LabVIEW program and the Java applet on a web client. Using AppletVIEW, an instrumentation system connected to a PC and to the Internet can now be made accessible from any computer in the world with a web browser and network connectivity (Figure 5). AppletVIEW contains three main modules, the Applet Builder, AppletVIEW.jar and AppletVIEW VIs
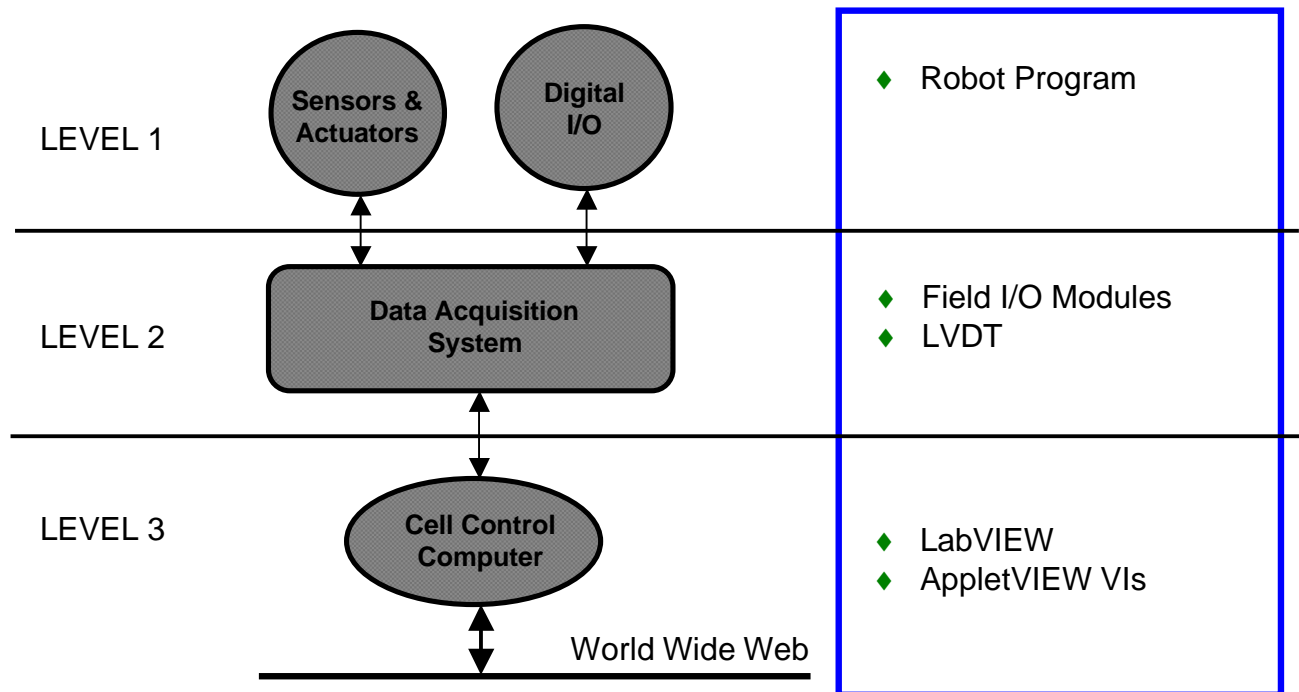
Figure 4. Software Framework for the Cell

- Applet Builder

This is a graphical tool for building Java applets that simulate the front panel of LabVIEW on the web page. The web page will look similar to the LabVIEW front panel. The Applet builder file is saved as a ".jvi" file. It is also automatically saved as a html file.

- AppletVIEW.jar

These are the Java class files for the web browser. The applet created using Applet builder is pushed to the web client at run time and reconstructed with the Java class files included with LabVIEW.

- AppletVIEW VIs

These VI's are used for transmitting data back and forth between LabVIEW and the Java applets. Two examples of these VI's are ReadApplet.vi and WriteApplet.vi. ReadApplet.vi will read data from the applet on the web page, into LabVIEW and WriteApplet.vi will write data from LabVIEW to the applet. In order for AppletVIEW to function, a computer server should be running LabVIEW continuously.
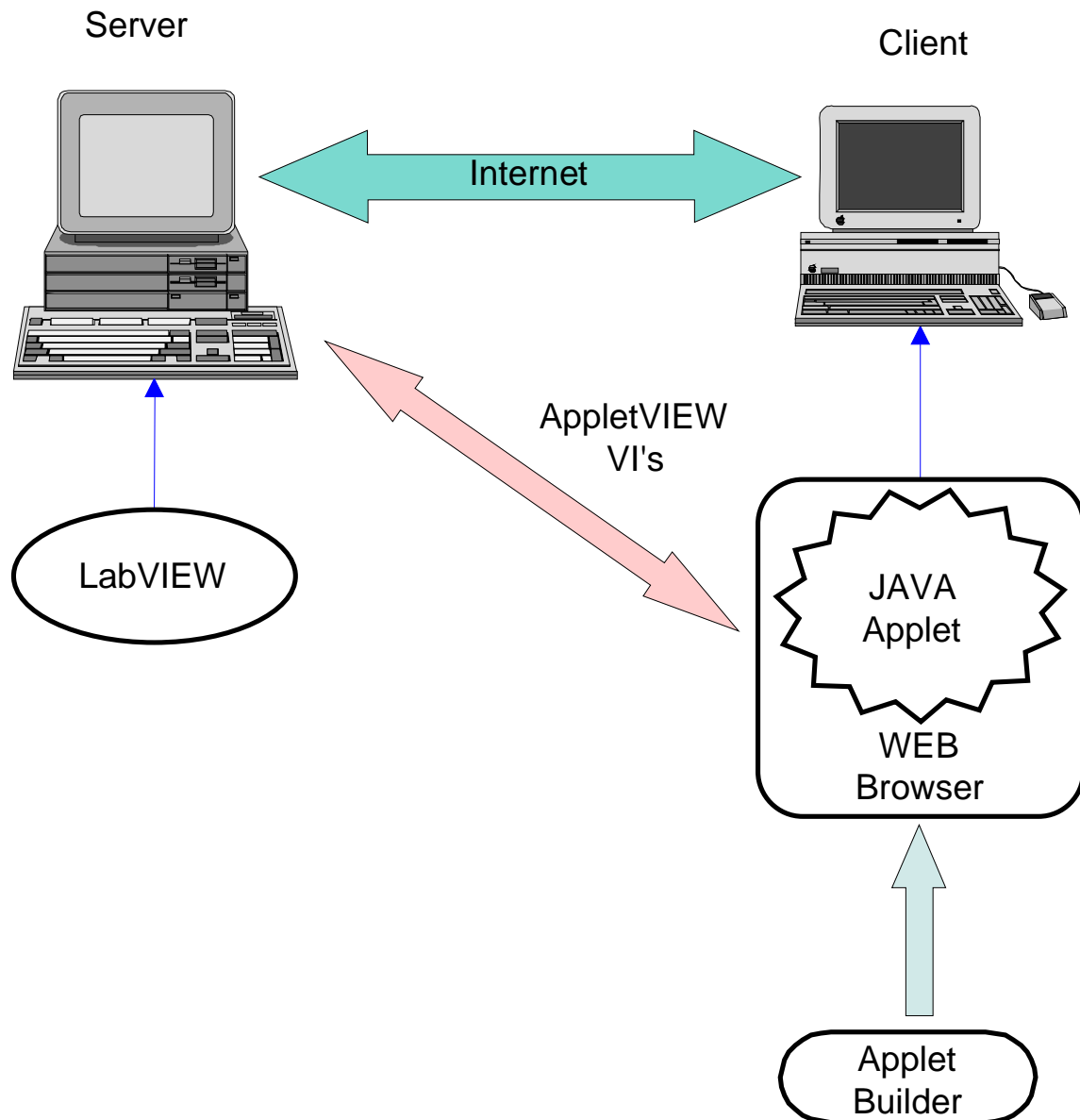
System Communications

Figure 5. System Communications Architecture

Level 2

This is the intermediate level and is responsible for data acquisition. The field output module triggers the required outputs and the input module receives the voltage input from the LVDT and feeds it to LabVIEW.

- Server

This is a computer that accepts requests from the user. The server used for this experiment is Microsoft's Personal Web Server. The server has a root directory and all the files that are supposed to be displayed to the user are stored in this directory. The user can log on to the web site and request the files from the server. The .jvi and .jvi html files created using Applet Builder are stored in the root directory. When the user logs on to this web site, and requests the html page, the web server pushes up the .jvi html file. Once the page is loaded, the html specifies that Java applets have to be loaded and the browser's Java virtual machine requests the server for the Java class files in the AppletVIEW.jar. Once the server pushes this file to the web page, the applet begins running. The applet then requests the .jvi file, which is then sent to the web page by the server. In order for all this to happen, LabVIEW has to be running on the server. Once the connection between LabVIEW and the browser has been established, both are ready to receive and send data. The AppletID identifies the web browser and LabVIEW VI. The connection stays open as long as the browser is open. If one browser is connected to LabVIEW and some other client tries to connect to LabVIEW, it will not connect till the previous client has disconnected.

- Description of a Virtual Instrument (VI)

The VIs created to control this experiment have a lot of sub VIs. Sub VIs are responsible for receiving inputs, perform the required operations and sending the output. The CreateAppletListener.vi is the sub VI that looks for connection from a web browser through a specified port. Only when this VI gives an output signal, the rest of the components of the VI are executed. This is, only when a client connects to LabVIEW from the web browser, they can operate the experiment. The next VI is WaitOnAppletConnection.vi. In this VI you can specify how long it has to wait for a connection. The default is to wait indefinitely. Once data has passed these two VIs, the VI is ready to accept data from the browser to start the experiment. When the user presses the start button on the browser, the case structure in the VI becomes true and the VI starts execution. The data from the browser is read into the VI using the ReadApplet.vi. The components in the front panel and the browser are identified by the component IDs that were created in the Applet Builder. The ReadApplet.vi gives out the component ID from which the data is being received and the value to be written.

Once the program starts running and the part measurements are taken from the LVDT, calculations are performed in the VI to find out the statistical measures such as mean, standard deviation and Cpk are displayed on the front panel. These data are then written to the browser using the WriteApplet.vi. This VI gives out the value to be written and to what component it is to be written. The part number being measured and the current measurement are also displayed in the browser.

VIII. Benefits and Issues of Non-site based Hands-on Courses:

The following benefits will play a major role in sustaining the student's interest and their retention in non-site based hands-on courses:
1. Greater communication between faculty and students
2. Partnership between instructors on site and instructors at other institutions
3. Latest information is communicated at faster rate
4. Course delivery becomes more easy and less expensive
5. Duplication of effort is eliminated
6. Learning materials and other resources can be cross-referenced with hyperlinks

1. Technical support requirement is reduced
2. Inclusion of graphics and animation will give depth to the content and material

Even though the above list of benefits is impressive, web-based delivery of hands-on courses requires a thorough understanding of the following issues:
1. Student motivation
2. Training and adaptation time required with new technology
3. Problems with new technology implementation (should be addressed effectively and on time)
4. Limited instructional methods because of current hardware and software limitations
5. Equipment operational safety
6. Security issues

IX.   Conclusion and Future work

This project helped us successfully accomplish the following:
1. Development of a remotely controlled cell that can be used for web based quality control.
2. The integration of robot and other various input and output devices with a computer based data acquisition and control system in developing an intelligent control architecture.
3. The successful interface of LabVIEW with the experimental setup to control the entire experiment.
4. A platform for future developments.

Some suggestions for future work:
1. Integrate a milling machine with the setup to machine the parts and then check for dimensional accuracy.
2. Transmit live video of the operation of the cell to the web page.
3. Incorporate security in the system in terms of giving access to the users to operate the cell.

X.   Bibliography

1. Edinbarough I and Ramkumar M., A feasibility study for the implementation of non-site based hands-on curriculum for engineering technology education, ASEE Conference, June 2001
2. Berenfeld, B. (1996). Linking students to the infosphere. T.H.E. Journal, 4(96), 76-83.
3. Khan, B. (Ed.) (1997). Web-based instruction. Englewood Cliffs, NJ: Educational Technology Publications.
4. Mioduser, David; Nachmias Rafi; et.al., Web-Based Learning Environments: Current Pedagogical and Technological State, Journal of Research on Computing in Education, Volume 33, N0. 1, 2000
5. LabVIEW Student edition 5.0, Robert H. Bishop
6. Field Point user manual, National Instruments
7. AppletVIEW user manual, Nacimiento Corporation
8. IBM 7535 Users Manual

IMMANUEL EDINBAROUGH

Immanuel Edinbarough is a Professor in the department of Engineering Technology at the University of Texas at Brownsville. He has 6 years of industrial experience in the field of machine tools manufacturing. He has been in the teaching profession for the past 13 years, 3 years as a visiting professor at the Rochester Institute of Technology. His teaching and research interests are in the areas of Automation, Robotics, Machine Vision, and CAD/CAM/CIM. He has published several papers, in these areas, in various national & international conferences and journals.

MANIAN RAMKUMAR

Manian Ramkumar is a Professor in the Computer Integrated Manufacturing Engineering Technology Department at the Rochester Institute of Technology, Rochester, NY. He teaches courses in CAD, CAM, CIM, FMS, robotics, surface mount electronics manufacturing, assembly automation, and controls for manufacturing automation. He was instrumental in developing the CIM and Surface Mount Electronics Manufacturing laboratory at RIT. These laboratories are equipped with production scale equipment that is used for hands-on training and conducting applied research projects for companies.

KARTHIK SOUNDARARAJAN

Karthik Soundararajan is a Controls Design Engineer at Alliance Automation Systems, Rochester, NY. He graduated from the Rochester Institute of Technology with a Master of Science degree in Computer Integrated Manufacturing. His areas of expertise are in the field of Programmable logic controllers (PLC's), Robotics, Vision systems, I/O networking and communications.