NETWORK-BASED REMOTE CLOSED-LOOP FORCE CONTROL USING LABVIEW

Richard Chiou[1] & Yongjin (James) Kwon[2]

[1]Applied Engineering Technology
Goodwin College of Professional Studies
Drexel University
Philadelphia, PA 19104, USA

[2]Industrial and Information Systems Engineering
College of Engineering
Ajou University
Suwon, South Korea, Zip 443-749

Abstract

Remote monitoring and controlling of manufacturing equipment has many applications. One of the applications is to observe the different types of forces being applied by a robotic gripper at the remote work cell. For this reason, the “sense of touch” feedback is necessary to effectively control the gripping operation. The actuator used for the gripper was a permanent magnet DC motor. The input to the motor was a voltage and the output was the gripping force measured by a force sensor. LabVIEW was used to create the VIs for remote controlling of the gripper. The VIs were constructed using client-server architecture and the communication protocol was UDP multicast. This allowed effective real-time control of the gripper over the Ethernet network. The server VI was used for data acquisition and client VI was the control application where the user provided the control method, parameters and monitored the gripper input-output signals. The presented development was incorporated in the course, MET 205 Robotics and Mechatronics in the spring and fall of 2008 at Drexel University, and the students in the class conducted remote experiments. The students found the setup very interesting and practical, especially in the area of information integrated production systems.

Introduction

Ethernet is a computer networking technology for local area network (LAN). The devices connected to the Ethernet network communicate with each other using different types of languages called protocols [1-3]. Every protocol transport data in the form of unit blocks of information called packets. These protocols are universally accepted. Therefore, Ethernet provides a flexible platform for automation system, a high data bandwidth and a promise to support higher bandwidth requirements in the future [4-6]. It has a straightforward integration with Internet. Broadly speaking, two types of communication protocols are mostly used for networking: TCP and UDP. TCP (Transmission Control Protocol) is the most commonly used protocol on the Internet. The reason for this is because TCP offers error correction. When the TCP protocol is used there is a "guaranteed delivery." This is due largely in part to a method called "flow control." Flow control determines when data needs to be re-sent, and stops the flow of data until previous packets are successfully transferred. This works because if a packet of data is sent, a collision may occur. When this happens, the client re-requests the packet from the server until the whole packet is complete and is identical to its original.

UDP (User Datagram Protocol) is anther commonly used protocol on the Internet. However, UDP is never used to send important data such as webpages, database information, etc; UDP is commonly used for streaming audio and video. Streaming media such as Windows Media audio files (.WMA), Real Player (.RM), and others use UDP because it offers speed! The reason UDP is faster than TCP is because there is no form of flow control or error correction. The data sent over the Internet is affected by collisions, and errors will be present. Remember that UDP is **only** concerned with speed. In particular, the network delay has a tremendous effect on the stability of Internet-based feedback control. This is the main reason why UDP must be used for the remote feedback force control via the Internet [3]. LabVIEW provides methods for using these protocols for data transfer over the Internet. The data are transferred between two or more PC connected via the Ethernet network. The PC where the data originates is a server and the PC where the data is transmitted to is the client. Such connection is called the client-server architecture.

As an effort to utilize the network-based control system using LabVIEW, a gripper force control process has been described in this paper. A gripper is a device used for holding and manipulating objects. In robotic environment, grippers represent the end-effectors. Robots use them for material handling. The movement of the gripper fingers is controlled by a small permanent magnet DC motor, a servo motor or a pneumatic actuation system. As the gripper comes in contact with the object to be picked, the reaction force between the surface of the object and the gripper fingers increases. The maximum value of this force is limited by the force provided by the actuating motor as well as the type of parts being handled. Changing the force provided by the motor, the gripping force can be controlled according to the part types. As the gripping process progresses, there can be disturbances from unforeseen elements like vibration, which may reduce the effectiveness of the gripper [7, 8]. Therefore, a force feedback can be used in a control loop to ensure that the gripping force is maintained at a required level. The client VI developed using LabVIEW performs the control operation using PI Control to minimize the response overshoot and ensure quick response [9-14]. Figure 1 depicts the overall system setup.

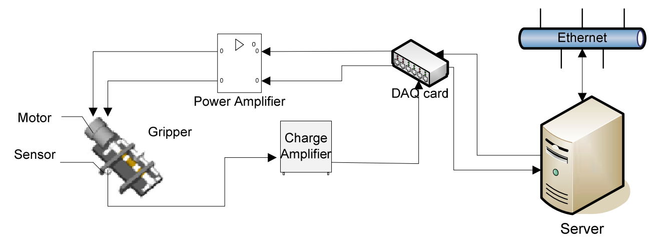
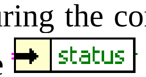


Figure 1: Block diagram showing the server side setup.

LabVIEW computer program was selected because of its ability to easily interface with physical devices through the Data Acquisition Card (DAQ), it provides functions and methods for network communication programming and ease of incorporating mathematical functions. It also provides functions to control the data transfer rate, which is essential for real-time control applications. Quanser’s WINCON is MATLAB Simulink based software. It’s not compatible with LabVIEW unless National Instrument develops the driver for WINCON. For LabVIEW based control environments, the hardware, such as DAQ-Pad or Compact-RIO, developed by National Instrument is recommended. This paper describes the design procedure of server VI, client VI and highlights on the communication protocol used.

SERVER VI

Server VI performs the data acquisition processes. For this, a DAQ card is used. A DAQ card is an on-board or external device that connects to the computer. DAQ card consist of a 12-16 bit analog to digital converter (ADC) with 8 -16 channel input (AI) and 1-2 channel (AO). The DAQ card used for this study was the NI DAQpad-6015, which has 12 bit ADC, 8 input channels and 2 output channels. This card was connected to the server using USB port. The force sensor mounted on the gripper was connected to the DAQ card through the input channel (AI) and the DC motor was connected to the output channel (AO). The maximum allowable channel depends on which type of A/D board is being used. For boards that have both single ended and differential inputs, the maximum allowable channel number also depends on how the board is configured. For example, a PCI-DAS6025 has eight channels for differential, 16 for single-ended input mode. The application to read multiple signals is either using 2 DAQ-assistant in the block diagram or just use the end user GUI called signal express for LabVIEW [15]. The sampling rate (Hz) in the DAQ device can be assigned by either "Measurement & Automation" module or DAQ assistant in the block diagram.

Three processes are performed on the server VI for the closed loop force feedback control via Ethernet network. First, the VI reads the sensor response for force measurement second, it sends the data to the Ethernet network and third, it writes data to the gripper motor for actuation. The server side application (Figure 2) to read the data performs two processes. Figure 3 shows the G-code for the first process, which is the first part of the server side application to read sensor data. This code shows a sequence structure, which handles the UDP connection. It opens a UDP multicast socket on the *port* for which the value is provided by the variable *Local Port*. When this VI is run, the UDP polymorphic VI opens the connection. If an error occurs during the connection, it throws an "error" and the 

returns "true" and no connection is established. The connection is established when there is no error, which is shown by the Boolean "Connected". This sequence structure also opens a data acquisition channel on input port "ai0" using the DAQmx polymorphic VI. This channel is used to read the data from the force sensor connected to the gripper.

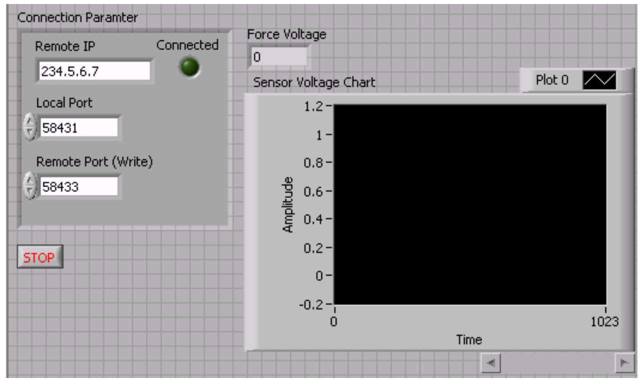


Figure 2: Front panel for server side application to read the sensor data.

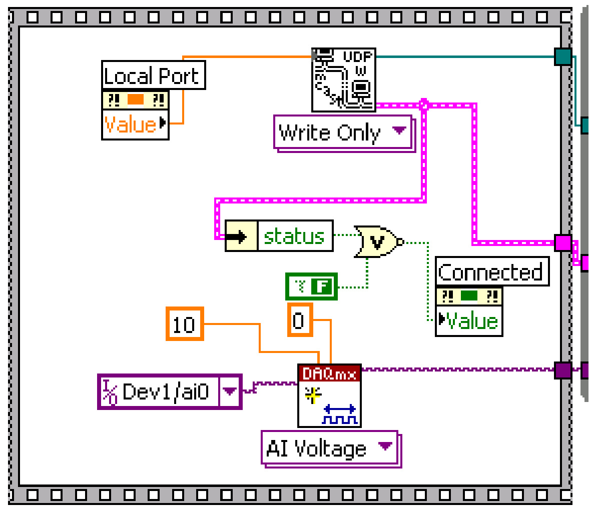
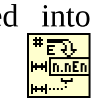
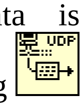


Figure 3: G-code for the server side application to connect to the Ethernet network using UDP protocol.

Figure 4 shows the G-code for reading the sensor data and sending it to the network. The data read from the sensor is numeric. The data packet sent in the network consists of string data type. The numeric data are converted into exponential string data ype using  polymorphic function. Once the data is converted, it is sent to the network using 

(UDP write). The connection id for this "UDP Write" is received from the connection opened in previous sequence structure shown in Figure 3. The socket parameters (IP and port addresses) for "UDP write" is provided by the variables Remote IP and Remote Port. Any client in the network can receive these data by establishing connection to this IP and Port addresses.

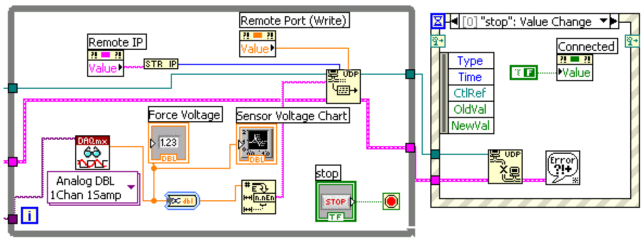


Figure 4: G-code for data acquisition and sending the data to network using UDP protocol.

Second operation performed in the server side application is to write the data received from client to the gripper actuation motor. The data acquisition for reading and writing the data is a parallel process. Thus, these two processes run independently. Figure 5 shows the front panel for the server application for writing data to the motor.

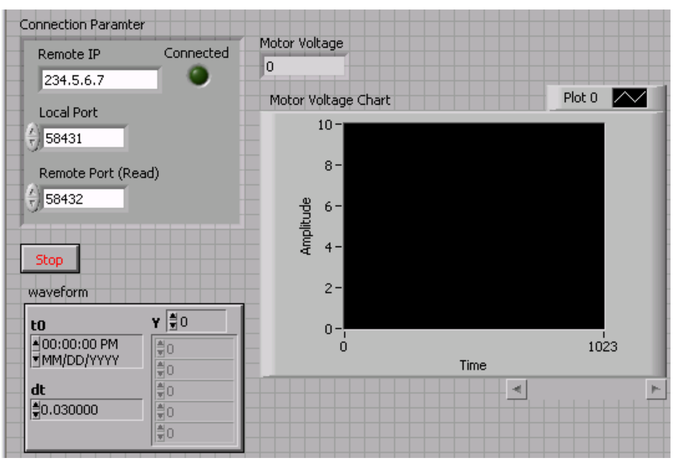


Figure 5: Front panel for server application to write data to gripper motor.

The server application for writing data to the motor runs independently and has no connection to the server application for reading data from the sensor. As in the earlier case, this application also connects to the network. The connection parameter (IP and port addresses) needs to be the same as provided by the client application (this will be discussed later). Figure 6 shows the first sequence structure for establishing network connection to read the data sent by the client. As explained earlier for the case of server application to read the data, this application also connects to the network when there is no error. Figure 7 provides the G-code structure.

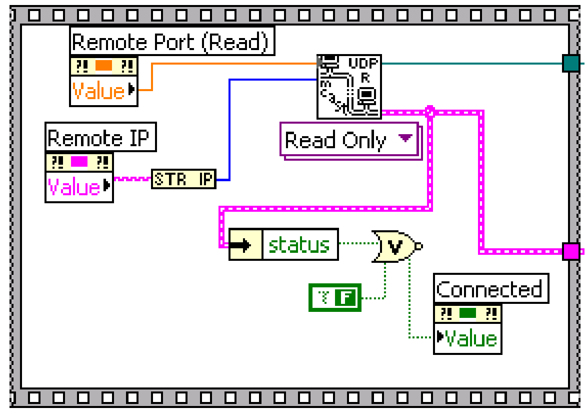


Figure: 6: G-Code for the first sequence structure of server application.

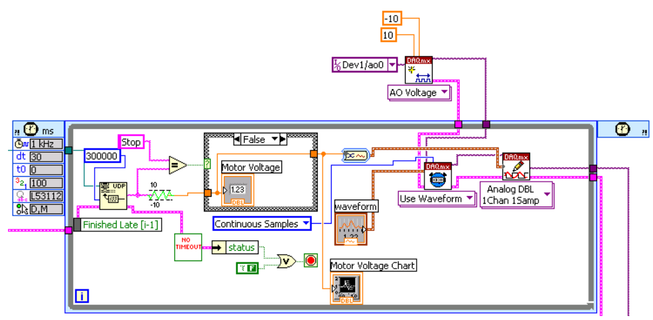
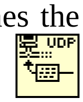
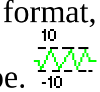
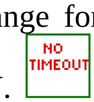


Figure 7: G-Code for reading the data from the network and writing to the motor.

This application uses a timed "WHILE" loop to ensure that the data writing rate matches the output rate capacity of the DAQ card.  (UDP read) is used to read the data from the network. This "data read" is in string format, which is converted into numeric data type.  is a subVI (Figure 8) used to filter the output voltage data greater than +10V and less than -10V as the output range for the DAQ card is from +10V to -10V.  subVI (Figure 9) ensures that the application does not exit because of the TIMEOUT error, which occurs when there is no data flow. For each data received, it is written to the DAQ card output through “ao0” channel.

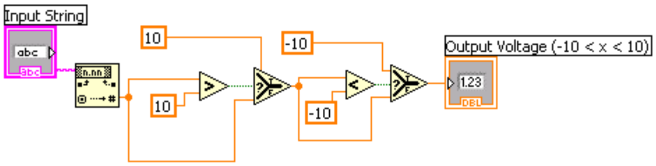
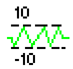


Figure 8: G-code for the  subVI.

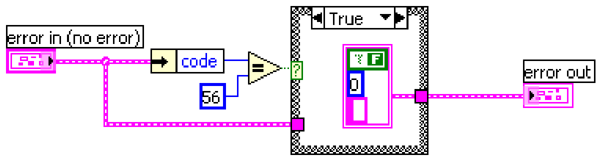
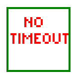


Figure 9: G-code for  subVI.

CLIENT VI

Client application is the control application that receives sensor data from the server, performs necessary control calculations and sends the output data back to the server to be written to the actuation motor. Figure 10 shows the front panel for the client application. When the client VI runs, the first operation is to reset all the values. Second operation is to establish the connection with the network. The connection parameter (IP and Port addresses) should match the ones provided in the server application. Once the connection is established, the user is provided with the options to select the type of control system (open loop or closed loop control) to use and provide the control parameters. For the open loop control, the user provides the required operating voltage for the motor which ranges from +10V and -10V. For closed loop control, the user needs to provide control parameters (Integral, derivative and proportional constants) depending on type of controller selected. Once these parameters are set, the control algorithm takes over the process and monitors the gripper. Figures 11 and 12 provide the sequence structure and the control G-code for the client application.

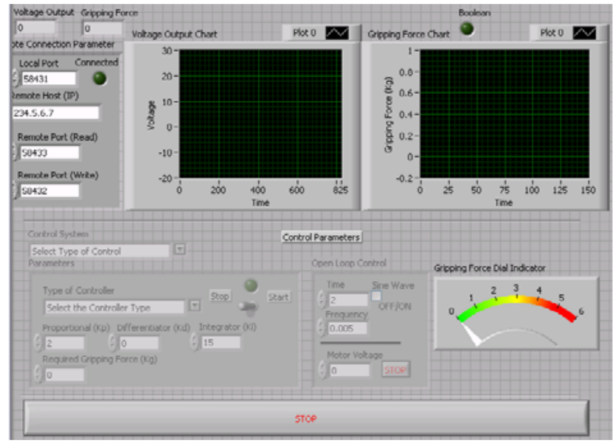


Figure 10: Front panel for the client application.

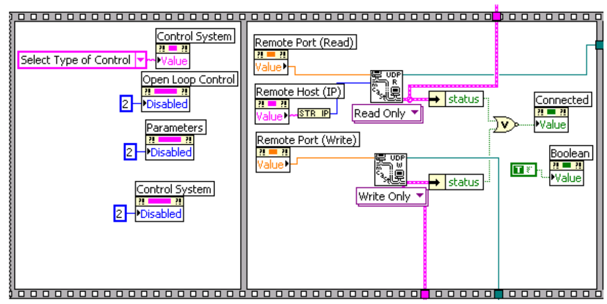


Figure 11: First sequence structure for the client VI.

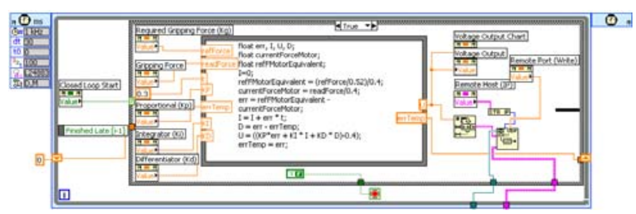
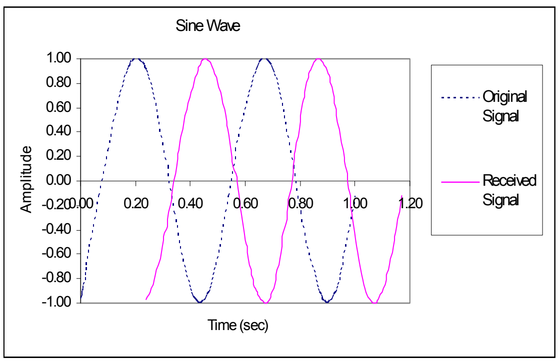


Figure 12: G-code for closed loop control in client VI.

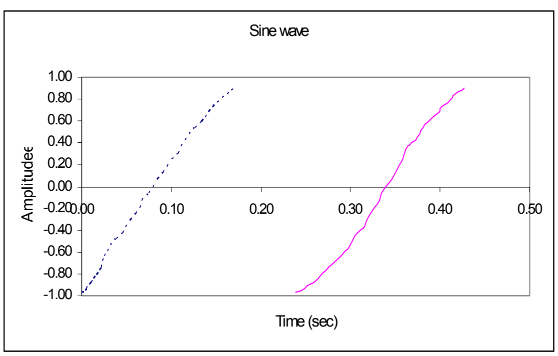
The control parameters are taken as the input for the “formula node” that computes the control calculations and output the signal as the variable “U”. The data are converted into string data type and sent to the network using the UDP write polymorphic VI . The output data are also displayed in a chart for monitoring and can also be saved in an excel file for further analysis.

RESULT

Data transportation through Ethernet or Internet network inadvertently suffers from time delay. Using remote connection procedure described in the earlier section, this delay was monitored between the client and server PCs by sending a 1.9 Hz sine wave signal from a server to the client, which bounces it back to the server. The total time for each data to travel from the server to the client was recorded. The received sine wave signal had a phase lag of 0.25 sec. Figure 13 shows the two sine waves plotted against the time. Figure 14 illustrates the transport delay by the network system.



(a)



(b)

Figure 13(a): Phase lag occurrence in the sine wave due to the network delay. Figure 13(b) shows the observation made for first 0.5 sec. This observation is done when the delay was close to 0.25 sec.

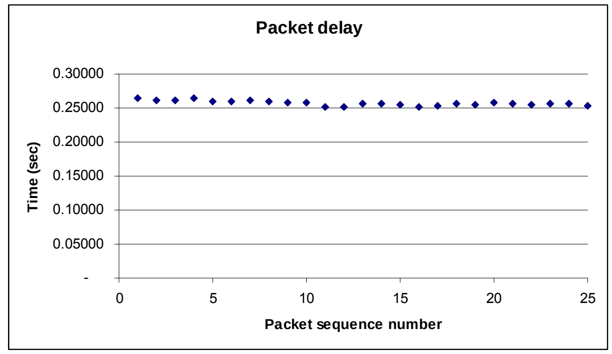
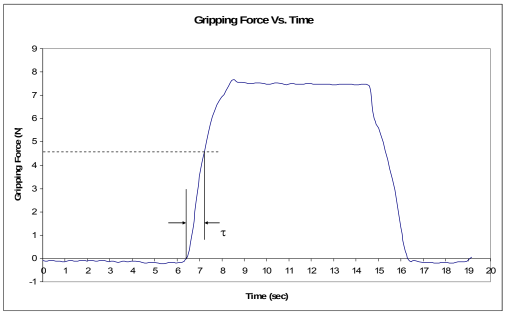
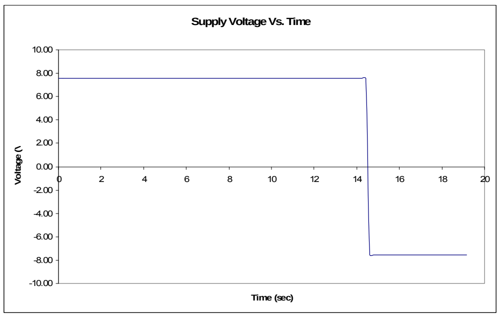


Figure 14: Transport delay caused by the network for randomly selected 25 data packets.

The objective of determining the transport delay was to tune the control system to minimize the effect of the delay. Gripper control process was performed to observe the open loop and closed loop control. First, a fixed step voltage of 7.59V was supplied to the motor. This generated gripping force of 7.84 N. Second observation was done using a closed loop control. The reference gripping force was set at 7.84 N. The results of these processes can be seen in Figures 15 and Figure 16. Figure 15(a) shows the open loop response plot of gripping force vs. time. When the step input voltage was given, the motor of the gripper started spinning, causing the gripper finger to move forward at a constant speed until it came in contact with the object that was required to be gripped. When the gripper came in contact with the object, the gripper sensor sensed force. The nature of rise in this gripping force is shown in Figure 15(a). When an input voltage of opposite polarity was given (as shown in Figure 15(b)), the gripper retracts. This caused the drop in the force experienced by the gripper. The nature of this drop in the gripping force due to the change in the input voltage polarity can be seen in Figure 15. Figure 16 shows the open loop response for step voltage increase from 7V to 10 V. The initial gripping force was set at 13N, which increased to 26N when the voltage input was increased from 7V to 10V. The time constant observed in both cases (Figure 15 and Figure 16) were equal to 0.7 sec.

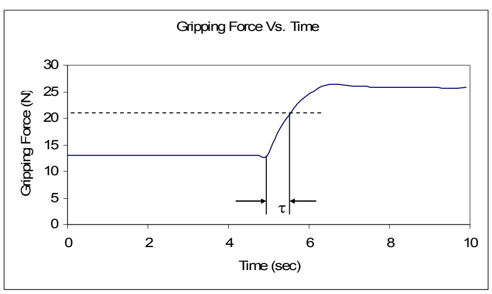


(a)

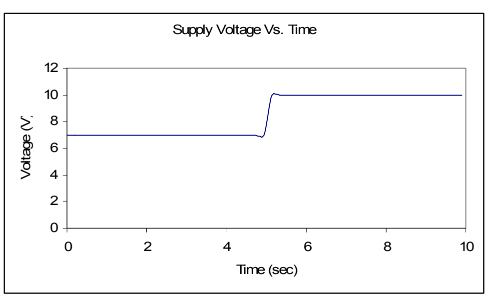


(b)

Figure 15: Gripper operation open loop response in (a) when the step voltage was 7.59V (b). Time constant observed is close to 0.7 sec.

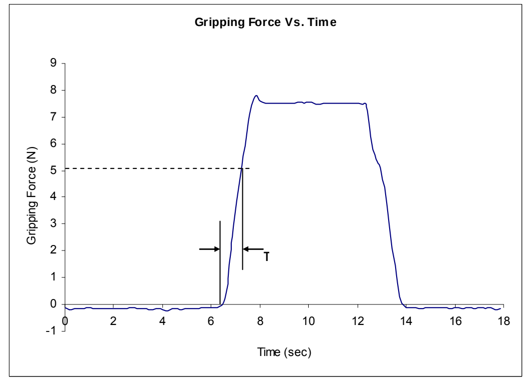


(a)

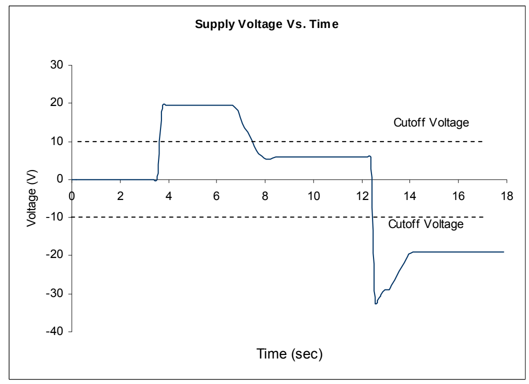


(b)

Figure 16: Open loop gripping response when the initial gripping force was 13N and final gripping force was 26N. The time constant (τ) = 0.7 sec.



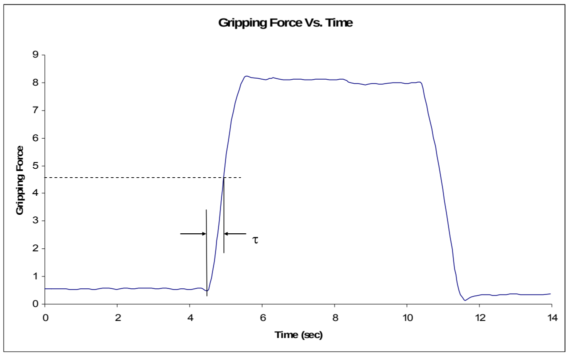
(a)



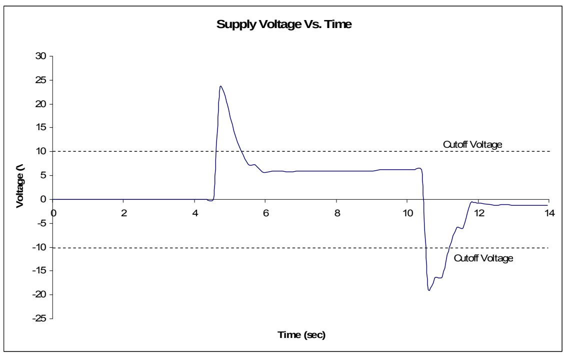
(b)

Figure 17: Proportional closed loop response when the gripper force was set at 7.4 N. The proportional constant (K_p) was 5. The observed time constant is 0.85 sec.

Figures 17(a) and 18(a) show the P and PI closed loop response plot of gripping force vs. time, respectively. A reference gripping force was entered and the control program handled the supply voltage. Figures 17(b) and 18(b) show the variation in supply voltage during the gripping operation. When the required gripping force was small, the supply voltage was high and vice versa. When the open loop gripping was performed, the time required for the gripping force to reach the required value was 2.188 sec. When the P closed loop gripping was performed, the time required for the gripping force to reach the reference value was 1.494 sec. When the PI closed loop gripping was performed, the time required for the gripping force to reach the reference value was 0.984 sec, which was quicker than other cases.



(a)



(b)

Figure 18: PI Closed Loop response for gripping operation when the gripping force was set at 7.4 N. Observed time constant is 0.5 sec.

The developed work has been used as a part of the course, MET 205 Robotics and Mechatronics at Drexel University in 2008. Overall, about 40 students have taken the course. In the class, students need to understand the principles of PID (proportional, integral, derivative) controllers. By using the robotic force feedback control system, the students were able to directly apply their knowledge and theory of PID control systems and observe the characteristics of each PID component (i.e.,. P, I, D and also the combination of PI, PD, etc.). Such testing bed allowed the students to better understand the seemingly complex mathematical equations in the PID control system by experimenting with the robotic system as they watch the changes in the system behavior. The graphical interface of LabVIEW allowed the easy control of the robotic gripper, and the effect of change was instantly displayed in the form of force vs. time graph in real-time.

Figure 19 illustrates the test bed for the students. The detailed list of equipment for the project is shown in Table 1. Figure 20 depicts a group of students working on a remotely located PC to control the robotic gripper. The web publishing tool in LabVIEW can be used to publish a server-side program and GUI through any web browser (ex. Internet Explore). It can also be accessed anywhere in the world through Ethernet. The web publishing tool is able to prevent multiple clients to login. The first user has to release the control of the program so that the other users are able to log in. Therefore, it is impossible for two students to control the grip force experiment simultaneously.

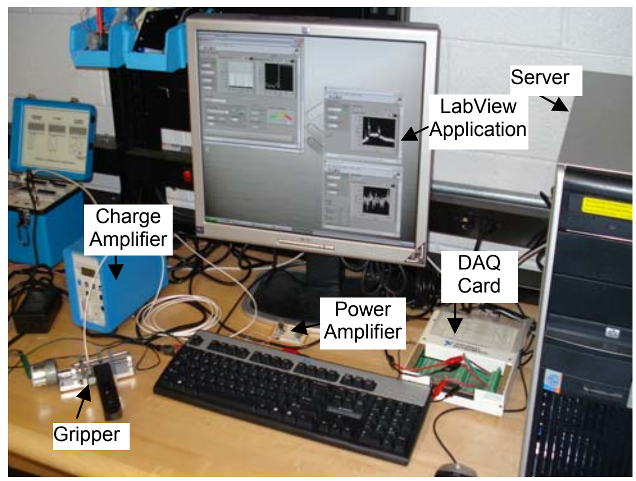


Figure 19: Experimental setup.

Table 1. List of Equipment.

	Equipment
1.	NI LabVIEW for Windows
2.	NI DAQPad-6015 (for USB), Screw-Terminal Connectors, US (120 V), 200 kS/s, 16-Bit, 16 Analog Input Multifunction DAQ
3.	Function Generator HP 3310A Function Generator
4.	Tektronix 2465 DVS 300 MHz 4 Trace Oscilloscope
5.	Piezo-electric loadcell (Kistler 9039)
6.	Charge Amplifier Kistler 5010
Others	A DC motor, a power amplifier a NPN power transistor, gripper, and screws.

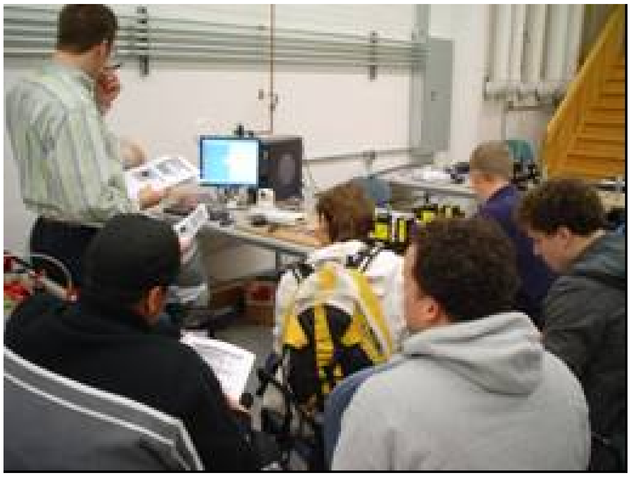


Figure 20: Students in MET 205 conducting a remote experiment.

CONCLUSION

This experiment for controlling the gripper has shown the effective use of labVIEW for remote controlling and monitoring through an Ethernet Network. The required data can be effectively transported through the network using UDP protocol by converting the data into string format. For numerical operations, this data can be converted into numeric data type using conversion functions provided by labVIEW. This study also presented the feasibility of using Ethernet for real-time control. The data transport delay can be determined and modeled into the system transfer function and its effect can be minimized by properly tuning the control system parameters, which was seen from the results of closed loop control, which proved to be more effective than an open loop control. A visual feedback using video stream from a camera can also be incorporated into the client application for a better monitoring of the remote work cell. From educational point of view, such remote testing facility allows the students to work on the experiment from anywhere, at anytime, which is especially beneficial to part-time, working students with scheduling conflicts, and place-bound students.

Acknowledgement

The authors would like to thank the National Science Foundation (Grant No. NSF-DUE-CCLI- 0618665) for its financial support of the project.

References

1. Bicchi, A., Caiti, A. and Prattichizzo, D., "Dynamic force/torque sensors: theory and experiments," Advanced Robotics," ICAR '97. Proceedings, 8th International Conference, pp. 727-732, July 1997

2. Brown, M., "Computer simulation of controlled impedance robot hand," Robotics and Automation. Proceedings. 1984 IEEE International Conference, pp. 442- 450, Mar 1984

3. Cao, J., Cleveland, W., Lin, D. and Sun, D., "Internet traffic tends toward Poisson and independent as the load increases," in Nonlinear Estimation and Classification, D. Denison, M. Hansen, C. Holmes, B. Mallick, and B. Yu, Eds. New York: Springer-Verlag, 2002, pp. 83–110

4. Castro, D., Marques, L., Nunes, U. and Almeida. A.T. de, "Tactile Force Control Feedback in a Parallel Jaw Gripper," In Proc. IEEE International Symposium on Industrial Electronics ISIE'97, Guimarbs, 1997

5. Ewald, H. and Page, G. F., "Client-Server and Gateway-Systems for Remote Control," IMTC 2003 – Instrumentation and Measurement Technology Conference, Vail, CO, USA, May 2003.

6. Fite, K.B., Liang Shao and Goldfarb, M., 2004, "Loop shaping for transparency and stability robustness in bilateral telemanipulation,", IEEE Transactions on Robotics and Automation, vol. 20, no. 3, pp. 620- 624

7. Jagannathan, S. and Galan, G., 2004, "gripper,", IEEE Transactions on Neural Networks, vol. 15, no. 2, pp. 395- 407

8. Kim, W. S., Hannaford, B. and Bejczy, A. K., 1992, “Force-Reflection and Shared Compliant Control in Operating Telemanipulators with Time Delay,” IEEE Transactions on Robotics and Automation, pp. 176-185

9. Ljung, L., System Identification - theory for the user, Prentice Hall, Englewood Cliffs, N.J., USA, 1987

10. Lorenz, R.D., Zik, J.J. and Sykora, D.J., 1991, "A direct-drive, robot parts, and tooling gripper with high-performance force feedback control," IEEE Transactions on Industry Applications, vol. 27, no. 2, pp. 275-281

11. Park J., Kim S., Kim D.H., Kim B., Kwon S.J., Park J.O. and Lee K., "Identification and control of a sensorized microgripper for micromanipulation," IEEE/ASME Transactions on Mechatronics, vol. 10, no. 5, pp. 601- 606, Oct. 2005

12. Pohjola, Mikael, “PID Controller Design in Networked Control Systems,” Master’s thesis, Department of Automation and Systems Technology, Helsinki University of Technology, Finland, 2006.

13. Ranaweera, A., Bamieh, B. and Parmenter V., 2005, "Sensors, actuators, and computer interfacing laboratory course at the University of California at Santa Barbara, Mechatronics", vol. 15, Issue 6, pp. 639-650

14. Shen, Y., Xi, N., Lai, K. W. C., Li and W. J., 2004, "Internet-based remote assembly of micro-electro-mechanical systems (MEMS)," Assembly Automation, vol. 24, no. 3, pp. 289-96

15. http://zone.ni.com/wv/app/doc/p/id/wv-531/nextonly/y

Biographical Information

Dr. Richard Chiou’s background is in mechanical engineering with an emphasis on manufacturing. Dr. Chiou is currently an associate professor in the Goodwin College of Professional Studies at Drexel University. His areas of research include machining, mechatronics, and internet based robotics and automation. He has secured many research and education grants from the NSF, the SME Education Foundation, and industries.

Dr. Yongjin (James) Kwon, with more than 12 years of experience in both academic and industrial settings, has extensive and practical knowledge concerning contemporary issues in design, manufacturing, and quality control. He is affiliated with Drexel University and is currently a professor of industrial and information systems engineering at Ajou University, South Korea. Since 2004 his research has been supported by Yamaha Robotics Co., a private company, and the National Science Foundation and the U.S. Department of Education.