REMOTE ACCESS OF CONTROL LOOP TRAINER KIT USING GNU Radio in REAL TIME

Shiva Agrawal, Pushkar Bhave, Nishigandh Vartak, Chitra Chopde, #Vishal Pande, *Prafulla Patil

Instrumentation, Vidyavardhini's College of Engineering and Technology
#Assistant Professor, Instrumentation, Vidyavardhini's College of Engineering and Technology
Address
shiva.agrawal1990@gmail.com
megalotus@gmail.com

ABSTRACT
In this paper we have described the concept of Remote Access Real Time Laboratory developed by us using GNU Radio which is a FOSS(Free and Open Source Software).
The main motivation behind our persistence is the knowledge of the facts which either discourage or infliction of reluctance among creative students who have very limited access time and who can hardly afford the very costly trainer kits available in the market. Moreover concept of remote access real time lab were implemented earlier but using proprietary softwares like LabView, etc. which again adds to the cost for implementation. The only solution to this is the use of free softwares.

For this purpose we are using GNU Radio which is a free and open source software(FOSS) having powerful benefits as easy coding using C++ and Python (Free and Open source softwares), simplicity in hardware interfacing, ease in implementation of various soft controllers, etc.

The above stated platform allows development of various controller algorithms such as P, PI, PID and other custom algorithms where end user can get real time process outputs. Moreover GNU Radio allow data acquisition during the experiment, storage, processing and online transmission of data to multiple users logged on to their respective web browsers. Control of the experimental process parameters (e.g. liquid flow rate) from one (or more) remote stations over the web in real time is also incorporated

This will enable trainees to access the kit without any strict time constraints and free of cost experimentation which will also help the technical personnel to study and implement various controller algorithms in real time. It also greatly facilitates joint research and development activities between academia and the industrial community.

KEYWORDS
Remote Access, RART lab(Remote Access Real Time laboratory), GNU Radio, web based engineering application.

INTRODUCTION
The web has become a vital element of many human activities and continuing to one of the strong core of a successful information-technology-based society. The web is used not only to gain visibility, share information, sell products and conduct business, but also to improve the way we design engineering systems, manufacture them and test the final products. A balanced usage of internet facilities is reducing the length of time and design cycles with improved quality.

The implications of such a networking technology can be far reaching. For example, using network technologies in measurement solutions, we can govern the input to and output from a hardware platform located on a production floor or test bench, which can avoid additional processing power for in-depth analysis in the control center via software support, log and store post-analysis information on a corporate database and display key processed information to clients or consultants around the world via a standard web browser.Computer-based data acquisition, web-based experiments and virtual instrumentation and control applications have been an active area of interest in recent years.In this context, the work presented here may help to meet the needs of higher education, where usage of interactive computer-based e-learning is on the rise. The proposed remote-access real-time laboratory

(RART-Lab) allows, for example, university clients situated in remote locations to take advantage of experimental facilities that cannot readily be duplicated (e.g. for reasons of cost). But our main focus in this context is not just to develop Remote – access real time lab but to develop the same by using FOSS.

So that there won't be any limitations due to Proprietary softwares.

For this a flow control loop is selected from the trainer kit of our college. The loop consists of mainly one flow meter and one final control element to regulate and control the flow through loop. To make the process real time work and remotely operable, signals from transducer and output to control valve is connected to computer via. Input Output module. On software part **GNUradio** is used for development of various control algorithms such as PID, PI, PD, model based, etc. Moreover we can experiment step change, ramp change response of the process.
To make the process remotely accessible, client server connectivity is used through GNUradio. In final stage of the project, we can further develop a website which will help students to access the trainer kit for experiment and study purpose through internet with the given time slots.

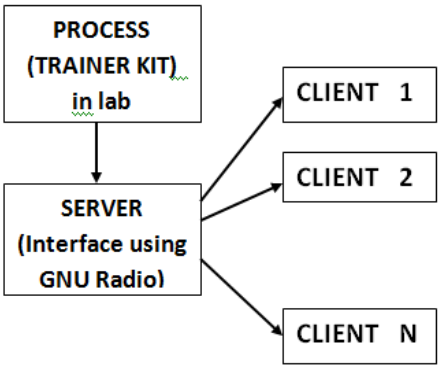

Fig. 1 Basic Layout

As shown in fig 1, main parts of the context are process loop of trainer kit, I/O card (microcontroller circuit) for data acquisition, GNU radio and Client server connectivity.

PROCESS LOOP (FLOW CONTROL LOOP)

The loop consists of mainly one flow meter and one final control element to regulate and control the flow through loop. To make the process real time work and remotely operable, signals from transducer and output to control valve is connected to computer via. Input Output card. Above mentioned loop is a part of trainer kit (shown in fig. 2)

Fig. 2 Trainer kit

I/O CARD

As shown in fig. 3, output from the flow sensor i.e current flow rate (4 to 20 mA) is converted to voltage (1 to 5 V) i.e. V to I convertion. This voltage is given to microcontroller (ATmega16) which is then send to computer i.e. to GNU
radio via serial communication. From computer, control signal in voltage is send back to microcontroller which is converted to analog voltage via PWM. The voltage signal is converted to current signal (4 to 20 mA) and given to final control element i.e. control valve for regulation of the flow.

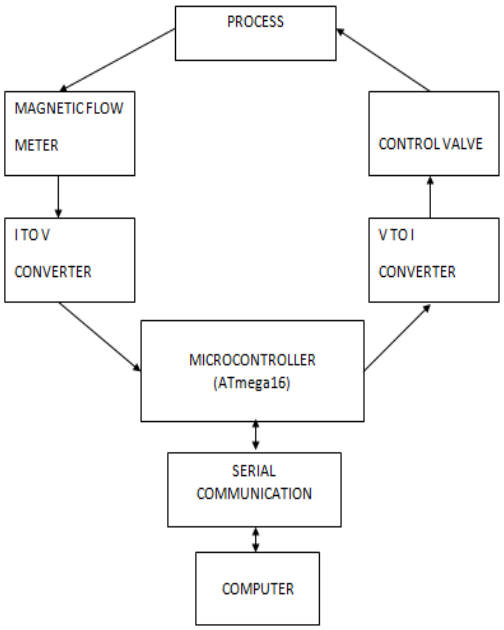

Fig 3 Block diagram of I/O card

CIRCUIT DIAGRAM OF I/O CARD

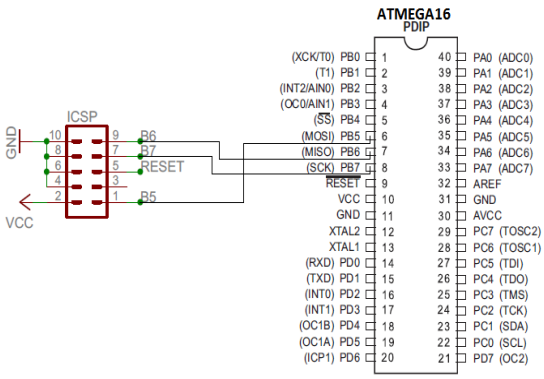

Fig. 4 a) Atmega16 with ICSP socket

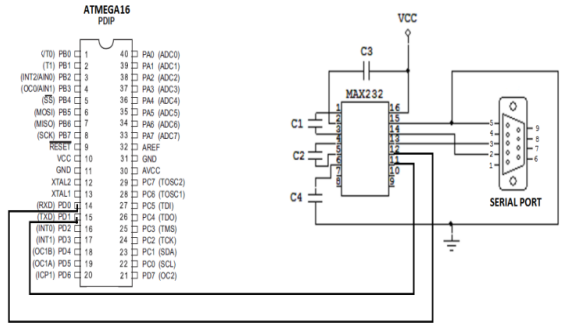

Fig 4 b) Atmega16 with MAX232 and DB9

GNU RADIO

GNU Radio is a free software toolkit for learning, building, and developing software defined radio systems. GNU Radio is released under the GPL version 3 license. GNURadio is an API for creating signal blocks using C++/Python languages.

GNU Radio is a collection of software that when combined with minimal hardware, allows the construction of radios where the actual waveforms transmitted and received are defined by software that is it turns the digital modulation schemes used in today's high performance wireless devices into software problems. GNU Radio provides a library of signal processing blocks and the glue to tie it all together. The programmer builds a radio by creating a graph (as in graph theory) where the vertices are signal processing blocks and the edges represent the data flow between them. The signal processing blocks are implemented in C++. Conceptually, blocks process infinite streams of data flowing from their input ports to their output ports. Block's attributes include the number of input and output ports they have as well as the type of data that flows through each. The most frequently used types are short, float and complex.

Some blocks have only output ports or input ports. These serve as data sources and sinks in the graph. Writing new blocks is not difficult. Graphs are constructed and run in Python. Graphical interfaces for GNU Radio applications are built in Python. Interfaces can be built using any toolkit we can access from Python. GNU Radio provides blocks that use interprocess communication to transfer chunks of data from the real-time C++ flow graph to Python-land. GNU Radio Companion (GRC) is a graphical user interface that allows you to build GNU Radio.

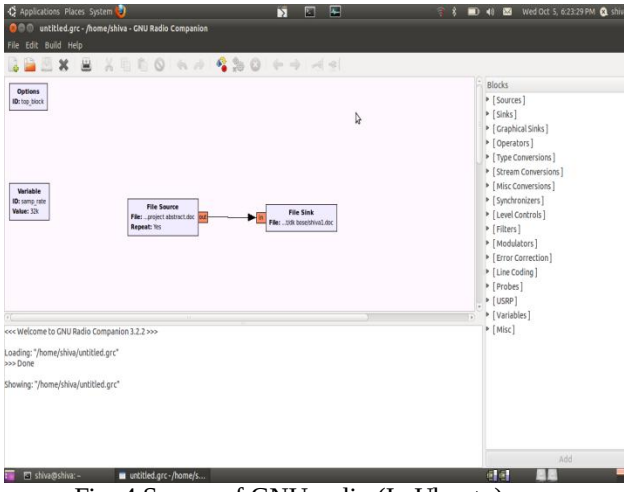

Fig. 4 Screen of GNU radio (In Ubuntu)

SERVER – CLIENT CONNECTIVITY

By using GNU radio and suitable server – client protocols remote process in real time is to be implemented. For the same, website is also be developed which will provide user interface for clients. Access of kit to clients is going to be schedule as per slot booking system so that at a time only one client will be able to access the kit in real time and others can view the output.

CONCLUSION

Blocks of some control algorithms are developed in GNU Radio by using C++ and Python platform and other blocks are in process. IO card is developed by using Atmega16 microcontroller along with serial communication circuit.IO card is developed and interfaced successfully with GNU Radio. Work of website and client server connectivity is under development.

FUTURE SCOPE

It is believed that a generic system such as that demonstrated in this work not only will make academic interactions and real-time data sharing more fruitful, but also can facilitate collaborative research and development activities. The system is also suited for India as well as other developing nations, where there are typically many distributed networks of engineering colleges/institutions, but all such institutions cannot afford to install large hardware intensive experimental set-ups.

In the present context few of the control algorithms have been implemented in flow control loop. Using the same free, open source and powerful platform of GNU Radio similar process trainer kits of educational institutes and engineering applications can be developed and access remotely in real time.

REFERENCES

[1] Paper entitled as "Remote-access real-time laboratory: process monitoring and control through the internet protocol" By *K. K. SoundraPandian, Manoj Rao and Sameer Khandekar (corresponding author)Department of Mechanical Engineering, Indian Institute of Technology Kanpur, Kanpur , International Journal of Mechanical Engineering Education 36/2.*

[2] Paper entitled as "Serial Port Programming in Windows and Linux" by *Maxwell Walter November 7, 2003*

[3] Paper entitled as "Challenges in Remote Laboratory Sustainability" by *Christophe Salzmann and Denis Gillet Ecole Polytechnique Fédérale de Lausanne, Switzerland Coimbra, Portugal September 3 – 7, 2007 International Conference on Engineering Education – ICEE 2007*

[4] Paper entitled as "Web enabled remote laboratory (r-lab) framework" by *ji hua, aura ganz, associate professor, director, multimedia networks laboratory, department of electrical & computer engineering, university of Massachusetts amherst, Amherst. 7803-7961-6/03/$17.00 © 2003 ieee november 5-8, 2003, boulder, co 33rd asee/ieee frontiers in education conference.*

[5] White paper entitled as "The Advantages Of Remote Labs In Engineering Education" by *Doru Popescu, Barry Odbert*

[6] Book entitled as "8051 Microcontroller" by *majdi* to learn serial communication circuit for IO cards.

[7] Book entitled as "Digital control" by Kannan M. Moudgalya *Indian Institute of Technology, Bombay*

[8] Site entitled "www.2.tech.purdue.edu/ecet/courses/reference material/atmel" for register and other material for ATmega16 algorithm development.

[9] Site entitled as "www.python.org" to get basics of python language.

[10] Site entitled as "*www.spoken-tutorial.org/wiki*" to get video tutorials of GNUradio , Linux.

[11] Site entitled as "*www.gnuradio.org*" official site for learning GNUradio.

[12] Datasheet of *ATmega16 and Max232.*