

Remote Laboratory



M. Saravana Mohan, P. Karthikeyan, D. Ram Kumar, M. Rupesh

ABSTRACT--- *This paper presents a web-based remote access real-time laboratory using IOT (Internet of Things) control. The suggested architecture helps to improve safety for user to work in hazardous environment or the handling of hazardous chemical substances at industry. This system also creates new opportunities for distance learning among students and researches, particularly within in engineering disciplines where hands-on experience is regarded as essential to acquire knowledge and reinforce professional skills. The remote laboratory presented here offers an economical solution for multi-users to access real instruments in real industry via internet. This remote laboratory system uses real instruments such as Robotic arm, DC pump, Water flow sensor and Cameras and the developer user interface was designed using python ad tkinter module which allows users to access the lab and information about the conducted experiment.*

Keywords - remote laboratory; IOT,robotic arm, python ad tkinter, Water flow sensor

I. INTRODUCTION

An innovative real-time remote-access control engineering teaching laboratory was developed and demonstrated at Oregon State University in 1998^[1]. A remote laboratory called VLAB on an oscilloscope experiment was setup at the National University of Singapore in 1999^[2]. Later, a web-based control experiment on a coupled tank apparatus was further developed ^[3]. These systems were developed to monitor the working process remotely through internet from anywhere around the work zone. The goal of this type of remote operation is to put the user in the control room with full and immediate access to all important system parameters and process control in industries instead of putting labours into hazardous work environment ^[4]. This contribution also presents a remote laboratory system with a new business model that aims at binging physical experimentation back into the learning arena, where remotely operable experiments used in education and training schemes are made available to a global market ^[5]. Even quite rigorous simulators do not fully demonstrate the many practical problems related to operation of real-world equipment, and the trend towards the

increased use of simulators in learning programs can only partially compensate for the decline in the use of physical experiments.

The control process which is performing in the industries need absolute time constraint, perfection in the work otherwise the work machine may get damaged and most importantly hazardous environment in industries affect the labours. In the education and research field, learning is accomplished through a diversified set of activities like lectures, paper and pencil exercises, simulators, pilot experiments, discussions, and report writing. Nevertheless, it is the strong believer of the authors that physical experiments constitute an important ingredient in order to carry out real hands-on learning activities ^[6]. As mentioned in the introduction, there are several reasons for the decline in the use of physical experiments. One important reason is the fact that physical experiments are costly to develop, maintain and operate. Another reason has been a strong belief that simulators can fully replace physical experiments. Some disadvantages of traditional labs include the need for a qualified person/teacher, the demand on institutional space, time constraints on the students and on the institution, crowded classes, safety problems, recurring maintenance, etc [7,8] . However, technological developments in the supervisory systems now make it possible to access process systems on the Internet. The idea designing the prototype of remote laboratory for access control of electrical machines through client – server management is illustrated by [9-15].

Web server, video server and Laboratory Virtual Instrument Engineering Workbench (LabVIEW) controller server are designed based on a client–server structure. In the technical design of front end user interface differs from previous papers which includes the graphical representation of the output and current scenario of work environment through video live streaming, and everything working under the web server and LabVIEW [16]. This idea helps to implement remote laboratory system in the same way to let the user understand the work environment through the live streaming video.

The Remote laboratory concept ideas was proposed in different ways by different people around the world. Most of the research in this field revolves around the use of simulation type or kind using the virtual environment with real instrument based using SCADA application. Ferrolho and Crisóstomo [17] designed a collection of software tools that allow the development of industrial applications of monitoring and controlling robotic networks and computer numerical control (CNC) machines inserted in flexible manufacturing cells. In [18], the trajectory tracking of a car-like mobile robot is performed wirelessly and through the Internet.

Manuscript published on 30 August 2019.

* Correspondence Author (s)

M. Saravana Mohan, Department of Mechatronics Engineering, Kumaraguru college of Technology, India. (E-mail: saravanamohan.m.mce@kct.ac.in, saravana.moha@gmail.com)

P. Karthikeyan, Department of Mechatronics Engineering, Kumaraguru college of Technology, India. (E-mail: karthikeyan.15mc@kct.ac.in)

D. Ram Kumar, Department of Mechatronics Engineering, Kumaraguru college of Technology, India.

M. Rupesh, Department of Mechatronics Engineering, Kumaraguru college of Technology, India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](https://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

The time delay transmission is tackled by a quality-of-service module.

An Internet-based teleoperation system for the control of an autonomous mobile robot with a two-layer fuzzy controller is implemented in [19]. To robustly transmit the surroundings and control information of the robot, packet-type data are employed. A system architecture of the Internet-based tele laboratory allows the remote control and programming of two educational online robots [20]. The delay associated with Internet communications is solved by using predictive-display techniques. Real-time control of a swinging load through the Internet was proposed earlier by [21]

This paper presents the development of a remote-access control system which allows users to perform control experiments through Internet. A dc motor control module is used as an example to illustrate our design. The system is composed of an internal distributed system and an application system linked by a data acquisition (DAQ) interface card. The main aim of this proposed remote laboratory system shown in Fig. 1 is to design a computer-based system for controlling the process remotely over the internet from anywhere in the world. The computer system can be either an embedded computer or a personal computer. The handling of hazardous chemical substances around the industry is used as an example to demonstrate the effectiveness of this remote laboratory in this paper, using real instruments such as Robotic arm, DC pump, water flow sensor and cameras. The proposed prototype design should be able to control from internet through developed application. As mentioned in the earlier, the demonstration is based on handling the hazardous substance around the plant, so the system should have to full-fill all the necessary user needs. The described user needs such as the proposing the end work environment to the user through visual live feed, proper sequential control of hardware device algorithm to avoid the collateral damage and continuous feeding of the hazardous chemical flowing status to the user. All these definitions bind together into the single application should have to be developed and it can able to work through internet. The design should be low cost compared to the cost of development of the multiple instruments in different location and making a travelling issue for users.

II. MATERIALS AND METHOD

The Proposed idea is to develop a system which can able to access through remotely from anywhere in the world through network. Therefore, whole thing works on the basis of the Internet of things. Remote laboratory name itself derive it meaning by defining that entire system has more number of laboratory interconnected through the network. These remotely connected devices can be controlled through network using any application or websites. The proposed system which can be used in any areas of interaction such as take the On-Field observation for study of astronomy requires specific mount location to monitor the changes happen related to the study of astronomy, also in the education field. The concept of having the remote laboratory is more like virtual laboratory where things actuating not in real time instead in happens in the simulation manner which may difficult for the users to understand the concept,

problems like this can be resolved by the developed system. The proposed concept of remote laboratory architecture given in fig 2 which can be accessed by the users without any limitation so that it helps them to understand the concepts very easily and let them feel the hands on experience with lost cost, it is more similar commercial platform for education. The proposed system can even use in the process management area where several process applications like industries, power plants or hazardous areas where human were limited by the physical strength and safety aspects these system comes handy solve these industrial issue using remote devices. In the development of remote laboratory prototype involves the assumption of end scenario which is major concept in developing the prototype and respective remote device will be designed. In this prototype system, end scenario condition assumed as manipulating the hazardous chemicals in the laboratory. There are few details is necessary to work with remote device from the remote location those are end environment, remote device control architecture, data acquisition and data presentation, and user interface. The entire system network management maintained using web server and client based interconnections.

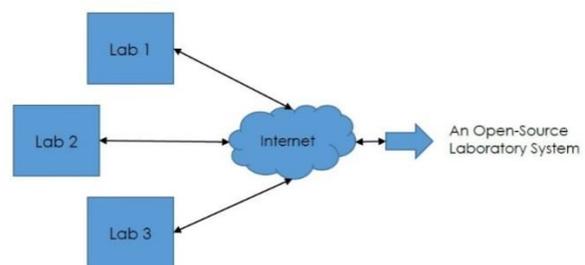


Fig 1 Proposed Model for Remote Laboratory

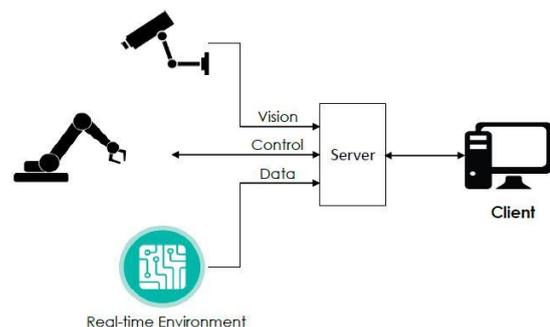
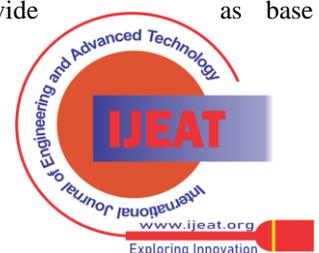


Fig 2 Basic Architecture for Remote Laboratory

Materials and software used in this system are Raspberry PI, USB camera, Servo motor, DC gripper, DC pump, Water flow sensor, Web server – Flask, OpenCV, and Tkinter. A Raspberry PI of model 3B which has 1.4 GHz processor and 2.4/5 GHz dual-band Wi-Fi (100 Mbit/s) is used. A simple USB camera is connected to the main processor and stream the camera into the network. This system is provided with 2 DC powered servo motor of one which has the capacity of 6kg/cm used to carry the end point of the arm and another of 12kg/cm which is used to provide as base support for the arm.



A DC gripper is used as an end effector for the robotic arm to manipulate the water container around the laboratory. A DC pump is used for refilling operation of the hazardous chemicals. A simple water flow sensor (5-24V DC power) is connected to the DC pump which is used to sense the amount of water flow pass between input and output. The web server is created using flask framework plugin. The plugin is added into the python library and coded in python with resources of front end as python application script using tkinter and backend as python script using flask. The python OpenCV (Open Source Computer Vision Library) package is used to process the sequential frames of images get manipulated and send it to network through IP address using web server. Tkinter is most commonly used GUI method. Python with tkinter outputs is the fastest and easiest way to create GUI applications.

III. CIRCUIT DESIGN AND WORKING RESULTS

The Hardware design shown in fig 3 for remote laboratory involves with hardware such as Raspberry Pi 3 model B, DC pump, YFS-201 flow sensor, cameras and user defined robotic arm. In the design of prototype, USB camera interfaced to raspberry pi through USB communication, using that feed from the camera is streamed live in the network so that cameras will be acting as user defined IP cameras [9].

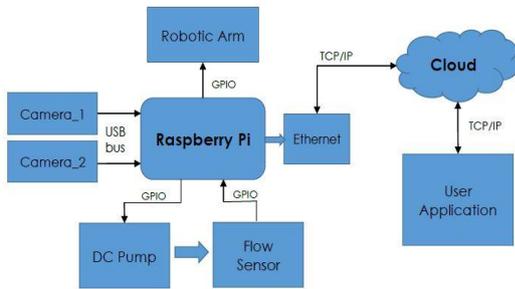


Fig 3 Hardware Layout for Remote Laboratory

The End device is essentially robotic arm, which will be controlled through raspberry pi GPIO pins. Servo motors are end actuators of the robotic arm, which will be operated in duty cycle pulses. The PWM will be generated through the raspberry pi GPIO pins and passed to controlled the sensor to actuate the position of the end effector. End effector is also the made of specified motor either it will be servo or dc motor which will be controlled either through microcontroller or motor drive depending upon the limitation PWM GPIO pins. The DC pump will be controlled through the motor driver and motor driver will be controlled by raspberry pi through GPIO pins. The Flow motor will be monitor only after the DC pump gets turned ON, using signals reflected by the hall effect flow meter water flow will be obtained after certain calculation through values of number of pulses and constants pulses from the flow meter datasheet. Raspberry Pi will be connected to the internet through Ethernet.

IV. NODE - VISUAL FEED

Two cameras get connected through the raspberry pi through the USB bus. In the python OpenCV module is used

in the python script, which actually requests the frame from the OpenCV module, which will be defined with the port location using video capture syntax. The main purpose of using two cameras in this system is to make the user understand the full visualization of the end environment shown in the fig 4. Two camera will be connected to the processor, script will be capture the frames of the camera and post it to the network. The working procedure of the vision camera feed will be checking of the hardware port connections, and configuring the camera port location into the python script. A JVC color CCD camera with the MATROX . METEOR2 PCI frame grabber is used to take the video in real-time. InetCAM video capture software is used as the video server interface. It is simple and requires no plug-ins [9]. The video server is located in the controller PC to save computing resources. The idea to implement the video server to help the user or customer or student to understand the end work environment is described briefly in a remote lab SCADA, where the end environment streamed through the video server by CCD camera.

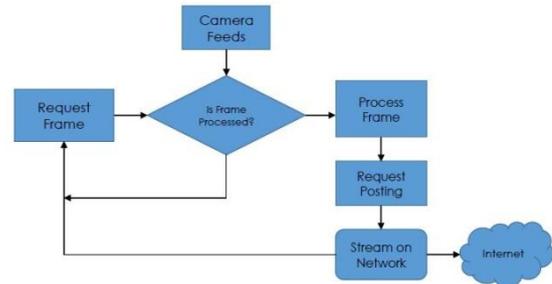


Fig 4 Visual Feed Flow Chart

First, the script will check for whether the previous frame get processed or not, if not it will wait until the previous frame get processed and once the previous process gets finished then resultant frame retrieved from the OpenCV request and frame gets processed and the frame will be get ready for posting it into the network. Network processing involves image stabilization technique to reduce the size of the images or frames. The first process involves the video property setting function and second involves the flipping of video frame with respect to certain axis and blurring of images result the lowering of the size of the frame this makes the video server streaming as fast as possible. Once the network processing gets finished the frame will be request the network for opening the channel to pass the sequentially processed frames. Once the request gets opened the flask server will post the data into the network to the certain IP address and port number.

V. NODE- HARDWARE CONTROL

The Hardware control of remote laboratory involves controlling of output devices such as Water Flow meter, DC pump, Servo Motors, and Motor Driver. The end device will be controlled through GPIO pins of the raspberry pi. The motor driver will be used to control the DC pump and driver will be controlled using PWM signal as mentioned in the Fig 6.

The raspberry pi can only able to pass two PWM signals at a time. For controlling of robotic arm, it requires the three PWM signal because robotic arm is made up of servo motor. Servo motor works on PWM (Pulse width modulation) principle, means its angle of rotation is controlled by the duration of applied pulse to its Control PIN.

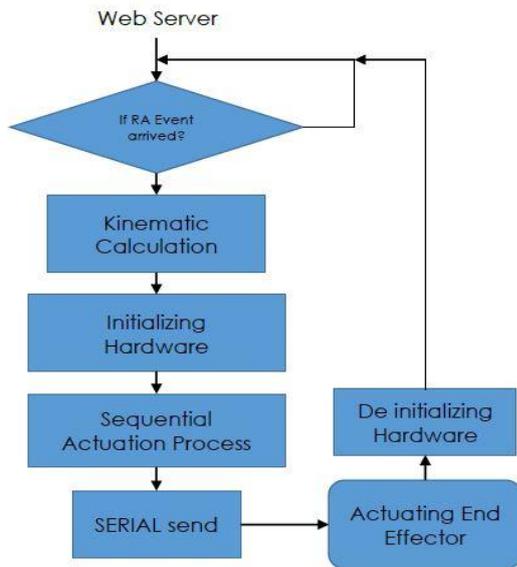


Fig 5 Flow chart for controlling Robotic Arm

Basically servo motor is made up of DC motor which is controlled by a variable resistor and some gears. The robotic arm has two servo motor one for base motor and another one for the shift of shoulder link and the end effector made up of dc motor or even available with servo motor for precise control of the end gripper fingers. It may possible for raspberry pi to fail to process the all the servo motors, grippers and driver to control the output to resolve that we can use Arduino to process the PWM function by communicating between the raspberry pi and arduino through serial communication. From the web server, the control event will be called from the user application. Web Serve will be managing the entire process in the hardware device which is used to bind every function into single IP address.

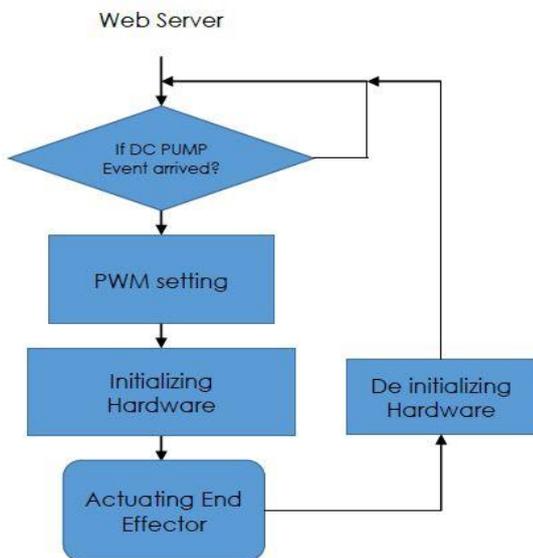


Fig 6 Flow chart for controlling DC Pump

From the application, the web server events will be called, for controlling the hardware devices as there are two main events, event Robotic Arm and event DC pump. When event hit the Robotic Arm, the two parameters, angles for each actuator and sequential order will get passed to the called event. Sequential actuation function is a safety function to manipulate the robotic arm in sequential order for safety aspects to control the arm without any collision. The flowchat of the control is given in the Fig 5. In case if the event hit the DC pump event, the raspberry pi will pass the PWM pulse into the motor driver to turn on the DC pump and control it into lowest speed for water flow. Once the pump gets start, python will start the multithreading to calculate the water flow for each moment and passes the value as return type. There is little difference between the controlling the robotic arm and dc pump, controlling center gets change for both operations. Since the raspberry pi can't able to pass more than two PWM pins, then uses arduino gets into purpose, which is used to control the robotic arm through microcontroller pins and DC pump will be controlled using the GPIO of the raspberry pi. Once the arduino receives the end angle for each actuator, the angle gets passed to each actuator in certain order, and once everything done it return a parameter to the source raspberry pi to intimate the finished position.

VI. NODE-DATA UPDATING

Once the DC pump gets turned ON it will start pumping the water to the water container through flow meter. Flow meter is connected to the raspberry pi through PWM pin as it requires the PWM to watch input given by the flow meter. Flow meter gives the input one whenever the hall effect rotameter type of flow meter, whenever it gives zero it indicates that it hasn't passed the one cycle yet. By finding the number of one's per certain second with multiplication with constant diameter of the rotameter cycle we can have determined the amount of water flow passes through the passage. Fig 8 shows the water flow Data Updating Flow Chart in web Server This signal will be identified in the raspberry pi by setting GPIO 13 to listen to the interrupt value of one after each sec it will multiplied with the rotameter cycle constant (circumference of the vane) and gives updated output of amount of flow which will be constantly updated in the internet by posting it using python's request module. Until the pump in the ON state this python flow meter thread will run, and keep the flow meter value update for each 0.5 secs. By using web server shown in the Fig 9, this updated value retrieved by the python application script and posting the new value continuously on the graphical user interface. So that, the user can monitor the flow pass through water container in real time using the flow meter thread and can control the DC pump ON and OFF using the Hardware control server. The controlling of GPIO through network will be verified with raspberry pi and controlling of hardware becomes more friendly and fast as expected.

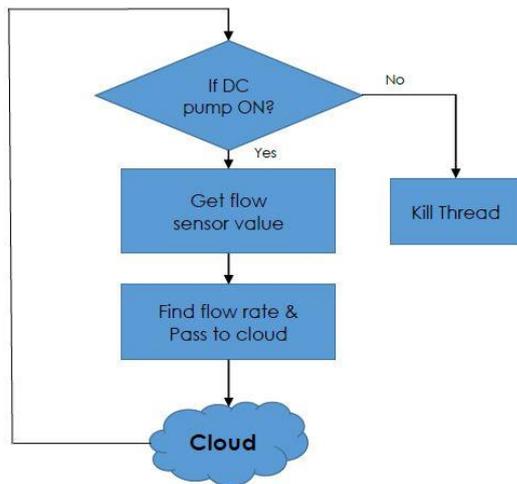


Fig 8 Water Flow Data Updating Flow Chart in web Server

Web Server is just responder to every request, acting on a central server, in this research work it acts on a computer. Web Server works in a way that in establish a web page on the network, if any user access it and make a request, web server undertakes that request and works on it, then respond to the user. This part of the work does two things, first one is create a web server and make it live and second one (message handler) is to update the data received from the flow meter into the cloud. The web server works on flask framework, which is used to develop a web server similar to the socket connection, but it only responds to the request. Flask framework is python plug-in which is added to the python library through which it can be accessed. Other frameworks to develop web server requires a web page as front end, but here it works as posting machine and back end scripts, here front end was developed using the pythons tkinter module to develop a graphical user interface and back end was developed using python script. The Visual feed web server generates an IP address and streams all the frames into the network, and other flow meter data will be continuously posted into the network in data control web server. All these IP address were bind together into the single user accessible application by using tkinter module. So it means that, to make the entire system work it is necessary to run the web servers, otherwise the whole system will fail.

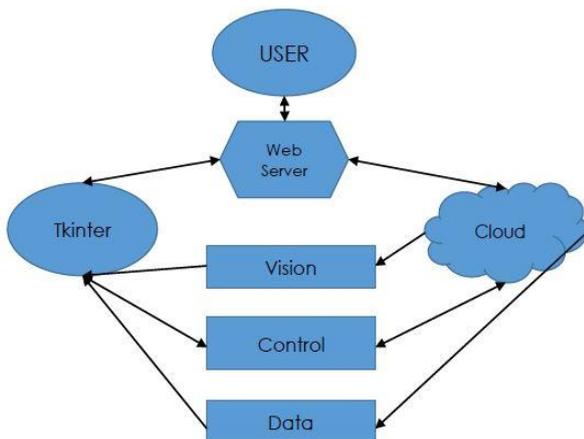


Fig 9 Web Server Flow Chart

VII. USER APPLICATION

In the end the entire system will be bind into the single user accessible application, it initially planned to be designed in workable webpage but it fails to interface so many things into the single web page. The problem faced with web page interfacing is that video streams, when video is streamed into the server, control lock instantly occurs. The control lock is nothing but an issue occurred due to continuous stream of video frames into the network, the entire control lock into that mount location in the web server, then resolving that using splitting up the operation into the individual IP address results, each individual must run on their own web page can't able to bind into the single web page. Finally, control lock and multiple web page issue is resolved using the application based user accessing system, which has been developed using the python script. The Tkinter module is python module used to develop user's own user interface based on the requirement. The entire script requires the three different end point, by using that it will access the visual feed, hardware control access and data posting algorithm. But it can only be able to operate only if the web servers get turned by the central access, otherwise it won't be possible to access the whole proposed system. By the applying the request to the access to remote laboratory, let the management to generate the customer id and generates the link and make the server up and send the link to the user. Thus, the user receives the link and thereby feeding those links into tkinter application, user can able to control the remote laboratory.

VIII. CONCLUSION

This innovative work is to provide solution to the low scale companies, students and independent researchers by developing a remotely controllable device or remote laboratory which can be accessed by anyone from anywhere with some little investment. Thereby the proposed system avoids the necessity of full time investment for the researches in the field of education and in industries to handle the equipment remotely around the plant. The current proof of concept design of remote laboratory is to provide the safest and advanced way to handle the hazardous chemical substance in the industries such as power plants, chemical laboratories and where ever the hazardous chemical substances were mostly handled. The developed prototype design of remote laboratory allows the worker to control or manipulate the hazardous chemical substance in a highly efficient and safest way.

REFERENCES

1. Bhandari, M.H. Shor (1998), 'Access to an instructional control laboratory experiment through the World Wide Web', *American Control Conference, Philadelphia*, pp. 1319-1325.
2. C.C. Ko, B.M. Chen, S.H. Chen, V. Ramakrishnan, R. Chen, S.Y. Hu, and Y. Zhuang (2000), 'A large scale web-based virtual oscilloscope laboratory experiment', *IEE Engineering Science and Education Journal*, pp. 69-76.

3. C.C. Ko, B.M. Chen, J. Chen, Y. Zhuang, K.C. Tan (2001), 'Development of a web-based laboratory for control experiments on a coupled tank apparatus', *IEEE Transactions on Education*.
4. Ingvar Gustavsson (2003), 'A Remote Access Laboratory for Electrical Circuit Experiments', *Int. J. Engng Ed.* Vol. 19, No. 3, pp. 409-419.
5. Christian Schmid, Tor Ivar Eikaas, Bjarne Foss, Denis Gillet (2001), 'A Remote Laboratory Experimentation Network', *IFAC Telematics Applications In Automation And Robotics*.
6. Ingvar Gustavsson (2002), 'Remote Laboratory Experiments in Electrical Engineering Education', *Fourth IEEE International Caracas Conference On Devices, Circuits And Systems*.
7. Zafer Aydogmus, And Omur Aydogmus (2009), 'A Web-Based Remote Access Laboratory Using Scada', *IEEE Transactions On Education*, Vol. 52, No. 1.
8. Houcine Hassan, Carlos Dominguez, Juan-Miguel Martinez, Angel Perles , And Jose Albaladejo (2007), 'Remote Laboratory Architecture for the Validation of Industrial Control Applications', *IEEE Transactions On Industrial Electronics*, Vol. 54, No. 6.34
9. Kin Yeung, Jie Huang (2003), 'Development of a Remote-Access Laboratory: A DC Motor Control Experiment', *Computers in Industry* 52 pp 305–311.
10. E.L. Gomez (2016), 'Robotic Telescopes in Education', *Edith Cowan Institute for Education Research, Perth, Australia*.
11. Claudiu Chiculita, Laurentiu Frangu , 'A Web Based Remote Control Laboratory'.
12. Baki Koyuncu, And Mehmet Güzel (2007), 'Software Development for the Kinematic Analysis of a Lynx 6 Robot Arm', *International Journal Of Computer, Electrical, Automation, Control And Information Engineering* Vol:1, No:6.
13. Mahidzal Dahari, Jian-Ding Tan (2011), 'Forward and Inverse Kinematics Model for Robotic Welding Process Using KR-16KS KUKA Robot', *Centre of Product Design and Manufacture*.
14. Mustafa Jabbar Hayawi (2011), 'Analytical Inverse Kinematics Algorithm of a 5-DOF Robot Arm', *Journal Of Education Of College*, No.4 Vol.1.
15. Juyi Park, Jung-Min Kim, Hee-Hwan Park, Jin-Wook Kim, Gye-Hyung Kang, And Soo-Ho Kim(2012), 'An Iterative Algorithm for Inverse Kinematics of 5-DOF Manipulator with Offset Wrist', *International Journal Of Mechanical, Aerospace, Industrial, Mechatronic And Manufacturing Engineering*, Vol:6, No:12.
16. K. Singh, S. Chatterji, And S. L. Shimi (2015), 'Remote Lab in Instrumentation and Control Engineering Using Labview', *International Journal Of Electronics And Electrical Engineering*, Vol. 3, No. 4.
17. Ferrolho and M. Crisóstomo (2007), "Intelligent control and integration software for flexible manufacturing cells," *IEEE Trans. Ind. Informat.*, vol. 3, no. 1, pp. 3–11.
18. H. Chih-Lyang, C. Li-Jui, and Y. Yuan-Sheng (2007), "Network-based fuzzy decentralized sliding-mode control for car-like mobile robots," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 574–585.
19. S. Kwee-Bo, B. Kwang-Sub, and H. Fumio (2006), "Internet-based teleoperation of an intelligent robot with optimal two-layer fuzzy controller," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1362–1372.
20. R. Marín, P. J. Sanz, P. Nebot, and R. Wirz (2005), "A multimodal interface to control a robot arm via the web: A case study on remote programming," *IEEE Trans. Ind. Electron.*, vol. 52, no. 6, pp. 1506–1520.
21. T. Chang, P. Jaroonsiriphan, M. Bernhardt, and P. Ludden (2006), "Webbased command shaping of Cobra 600 robot with a swinging load," *IEEE Trans. Ind. Informat.*, vol. 2, no. 1, pp. 59–69.