A WEB BASED ROBOT MANIPULATOR CONTROLLER

This section presents the realization of a simple and flexible microcontroller based robot manipulator controller. A web control/server module has been developed for the networked access of this robotic system in a laboratory. The robot actuator control is designed around a microcontroller, which senses the motor speed, position and direction with the help of a quadrature decoder and generates PWM signals for the motor control H-bridge. The robot controller developed is suitable for SCARA/XR4 robot arms from Rhino Robotics Inc., USA. The robot manipulators, which are equipped with permanent magnet DC motors with inbuilt incremental optical encoders for driving the links, and the end-effector are considered for the study. The controller is capable of accepting the commands from a host computer through an RS232 serial port. The control software in the host computer has a user friendly graphical interface with teach-pendant like keys.
and functions to control the manipulator arm. The computer can be replaced with a small web server implemented in another microcontroller, which is having a serial port to interact with the robot controller and an Ethernet interface for web access. The web pages stored in the program memory area are very similar to the host computer Graphical User Interface (GUI), through which the user can control and monitor the robot. Without any additional hardware or software any networked system can be programmed (teach the robot). Some of the laboratory exercises using the controller and manipulator are also discussed.

A1.1 Introduction

The capability and simplicity of web based automation systems are stupendously expanding their use in industrial, household and robotic applications. Robotic control is an exciting and highly challenging research work in recent years. Several solutions to the implementation of digital control systems for robot manipulators and mobile robots are proposed in various literatures [152, 153]. In order to meet the demands of a flexible robot controller, the amazingly increasing computational capabilities of personal computer are envisaged. By utilizing the inbuilt standard interfaces and resources one can easily design a practical robot controller, which can be upgraded even in a day-by-day manner. The compatibility allows the user to work offline and online. The use of PC and existing communication interfaces help the efficient co-ordination or synchronization of multiple robots in a large workspace in an industrial environment. The analog control schemes like PI or PID loop can be easily implemented using the host computer. These flexibilities are very useful in a laboratory environment to study the characteristics of the robot and its controller.

The Robotic workstation described here has been developed in the Department of Electronics, Cochin University of Science and Technology, India by
utilizing the existing Rhino XR4 and SCARA Robot arms from Rhino Robotics Inc., USA. In 1988, the robotics laboratory was equipped with Rhino XR4 and Rhino SCARA robots with their own controller and RoboTalk, a high level robotic programming language, which is capable of establishing a serial communication with the host computer [154, 156]. In order to gain more control and low level access over the control strategies as well as to utilize the state of the art technologies available, the company supplied controllers have been replaced with the in house controllers designed in Cochin University of Science and Technology. The controller design has the same interface standard that is compatible with the existing mechanical arm. So the robot controller developed is suitable for SCARA/XR4 robots. It can also be used as a general purpose controller for any robot having compatible driving power for the actuator, by changing the interface connector and configuration settings like maximum and minimum count values for various motors etc.. The entire controller hardware is simplified by using more integrated components and the inbuilt teach pendant interface is also eliminated. The controller is capable of establishing communication with a standard PC through an RS232C Port and can receive the commands for various link actuators to manage its position, speed, direction and count increments. The controller is also capable of executing the hard home command and detecting the end points and stall conditions for self protection of the system. The H-bridge drivers for the motors are designed around an integrated motor control chip from National Instruments [104]. A fast serial link of 28.8 kbps has been established from the computer to the controller. The computer is provided with a very user friendly interface to support the teach pendant operations and a high level language to learn the robotic systems. For obtaining more flexibility, another module has been developed for interfacing the controller to the web through an Ethernet link. The control GUI stored in this single chip module can be accessed from anywhere in the
network. The technical details of the system and its implementation and use in the laboratory are described.

A1.2 Robotic Workcell

The functional diagram of the setup is shown in figure A1.1. The Robotic Workcell consists of Rhino XR4 and SCARA Robot manipulators and a conveyor platform for robot to increase the freedom and flexibility of the robotic system. The XR4 is an educational robot having five axis articulated mechanical manipulator with an end effector or gripper. The robotic arm requires six motor drives for control and additional similar drives for actuating the moving conveyor robot platform and tilt rotator tables which necessitates auxiliary drive ports. The joint actuators are permanent magnet DC servomotors with inbuilt gear boxes and incremental optical encoders. The optical encoders mounted on the shaft of the drive motors result in high resolution position measurements. The shaft encoder pulses A and B (from Ch_A and Ch_B) are fed to the microcontroller to measure the position, speed and direction of rotation of the motor.

![functional_diagram]

*Figure A1.1  Functional diagram of the setup*
Rhino SCARA robots are also equipped with the same types of servo motor drives having the same control interface. So the controller can be used with any one of the Rhino robots with appropriate configuration. The SCARA robot is a four axis servo controlled arm with an end effector or tool. The main characteristic of SCARA is its high precision, repeatability and speed. The basic SCARA geometry is a parallel structure and it incorporates one prismatic joint for vertical up and down movements.

A1.3 Robot Controller Design

The robot controller uses PIC18F452 microcontroller, which can incorporate five link controllers. Each link controller consists of one direction control bit, one bit PWM signal and one bit brake signal. The link controller also uses two quadrature encoder inputs for finding the position, speed, direction of movement and stall condition of the link. The functional block diagram of the robot controller is shown in figure A1.2. It uses LMD18200 H-bridge driver IC from National Instruments to drive and control the link actuator. The LMD18200 is a 3A H-bridge designed for motion control applications. It accommodates peak output currents up to 6A and operating supply voltages up to 55V. The chip is also having a thermal shutdown (outputs off) at 170°C which is very helpful in short circuit and overload conditions. The driver chip having a current sense output is not utilized in this design as one can easily identify the stall condition of the robot from the PWM status and the encoder inputs.
Figure A1.2. Block diagram of the robot controller

The optical incremental encoder attached to the motor shaft is capable of generating pulses A and B. The low level information provided in the form of two pulse trains A and B are 90° out of phase (quadrature) and depending on the direction of rotation, one of these pulses will lead or lag the other. From these signals the actual position, speed and acceleration information needed for the controller are computed. So a link control requires three output signals to control the motor and two input signals from the optical encoder and one input from a micro switch to sense the hard home position. The robot can achieve hard home position with the help of micro switch sensing. A 20x4 LCD module is interfaced for monitoring the control status of the robot arm. This will display the current position of the links and stall conditions if any. The control card is designed in such a way that if the system needs more than five control/auxiliary ports, an additional card can be stacked to incorporate more links/Aux ports. A serial communication link has been established using a MAX 232 level converter. The power supply requirements of the motors is +18V, 10A and that of the rest of the circuitry is +5V, 1A.
A1.4 Control Program and User Interface

The user interface developed for the control program in the host computer is very flexible and modular and has three modes of operation. They are play, online control and file/offline control, which can be selected from the Control menu. From the Settings menu the hard home can be executed and the speed (PWM duty cycle) for the current step or program also can be set. The Status menu is helpful in monitoring the current motor count value and gripper status of the robot arm. The count values from the optical encoders are used for representing the position or angle of the link with respect to the reference or home position. The angle corresponding to the count value depends upon the number of pulses per revolution of the encoder and the gear ratio of the motor shaft. For a particular motor the number of pulses per revolution of the encoder and the gear ratio are constants, so the count value is a measure of the position or angle of the link.

![Screen shot in play mode](image)

**Figure A1.3 Screen shot in play mode**

In the play mode most of the teach pendant play mode functions of the Rhino Mark IV Controller are implemented. By using this GUI, all the motors can
be actuated individually and the movements can be saved by pressing the save button. The screen shot of the play mode GUI is shown in figure A1.3.

**ONLINE PROGRAMMING**

![Online Control Mode Screen](image)

**Figure A1.4 Details of the online control mode**

Figure A1.4 shows the Online Control mode screen. In this mode the user can select the appropriate motors by clicking the corresponding check boxes. The operator write the destination incremental count value for the motors and pressing the move button key will guide the robot arm to the respective positions with the selected speed by displaying the current position values of each motors at the current value location of the GUI. By clicking the Save button the user can store these movement steps in a file. The advantage of this mode is that the user can fine turn the movements by feeding small count values and establishing up or down
movements by visualizing the robot arm actions. The previous position key is also very useful in developing the control program or teaching the robot.

The third mode is the file/offline control mode. In this mode the system can load a file containing several movements and other commands like PWM value, delay, home(hard home), soft home, switch sense etc. The program can be run by clicking the run button and editing is also possible with this mode. A single execution or repeated operation can be achieved by selecting the single/continuous check box. This mode is used for offline editing of the program as well as file execution of the robotic system. The GUI for this mode is shown in figure A1.5. The control file format is in plain ASCII text and the user can edit or merge the files using an editor like notepad. A typical file is shown in figure A1.6.
Appendix 1

Figure A1.6 A typical file structure of the control program

A1.5 Web Control Module

The dedicated PC interface can be replaced with any networked computer by using a web control module. This increases the flexibility of the system and the micro-web control/server module [159, 160] is designed around a PIC18F67J60—with inbuilt Ethernet peripheral module having 128k flash program memory, 8k Ethernet Tx/Rx buffer and more than 3k data memory [158].

Figure A1.7 Block diagram of the Ethernet web control/server module
The functional block diagram of the system is shown in figure A1.7. The microcontroller is having all the features to implement the Ethernet interface. The only additional components required are the pulse transformer, associated resistors and capacitors. For establishing a communication link to the robot controller an RS232C interface is designed by using MAX232 level converter chip and the inbuilt serial peripheral module of the microcontroller. The program memory stores the code and web pages to interact with the user and control the robot arm. An IP number must be assigned to the control module during configuration of the system. Power supply requirements for the module are 3.3V for microcontroller and 5V for level converter.

All the control interfaces are implemented in web pages and is very similar to the dedicated GUI. The screen shot of the Control Web page for the file/offline control mode is shown in figure A1.8. The small web control/server module will generate appropriate commands to the controller. User login and password protections etc. are also implemented for security. The host computer or micro-web control module can send commands/parameters like hard home, position increments/decrement values, and duty ratio of PWM etc. to the robot controller.

![Figure A1.8 The Control Web page for the file/offline control mode](image-url)
A1.6 Exercises

In order to learn the basics of robot manipulation and the fundamentals of robot motion the students must practice certain laboratory exercises like accuracy, precision and repeatability study, pick and place objects, manipulator characteristics study by learning the actuator response etc [155,157].

The motor A is used as gripper or hand actuator for the SCARA/XR4 robots. From the full gripper opening to closing the controller receives more than 50 pulse counts from the corresponding encoder. So the pulse count value can be used to sense whether the object or work piece is held by the gripper or not. If the gripper driver is actuated and the number of pulses received by the controller is less than that required for a close means that the object is held by the gripper. In practice by applying 5 to 10 percentage of PWM signal voltage, while the gripper holds the object, the tendency of the object to slip can be eliminated.

For the repeatability study the gripper holds a felt tipped pen above a drawing sheet on the work table. The robot is moved to draw arcs on the paper in such a way that we obtain multiple arc crossing points on the sheet. Then the arm is moved in a random fashion and the same movements are repeated to obtain another set of arc crossing point that may overlap the old points or slightly deviate from the earlier ones. The repeatability of the robotic system can be computed from the deviations by repeating the process.

To perform the pick and place exercise, students use a number of small wooden blocks of various sizes. The robot can be taught or programmed to pick and place the objects in the desired way. Executing the same program by changing the speed, students can learn the speed and acceleration problems associated with the arm. Trapezoidal and triangular velocity schemes can be studied. Analog control schemes like PID or PI can be implemented by properly setting the
parameters like delays, PWM values and increment counts for the respective motors. The photograph in figure A1.9 shows the finished view and inside view of the controller and a typical workcell.
A1.7 Summary

The robot controller presented in this annexure replaces the existing XR4/SCARA robot controller from Rhino Robots Inc, USA and is capable of operating from a computer or micro-web controller system through Internet or Intranet. This setup is ideal for a laboratory to familiarize with the various control schemes and techniques associated with robotics. Solving the inverse kinematic equations and finding the pose of the arm is difficult to implement in the web control scheme. But in PC based controller approach, one can easily utilize the high level tools like Matlab for solving the same.