T 000

EDUCATION IN ELECTRONICS AND ELECTRICAL ENGINEERING
─────────────────────────────────────
MOKYMAS ELEKTRONIKOJE IR ELETROTECHNIKOJE

A New Approach to 3-Axis Cylindrical and Cartesian Type Robot Manipulators in Mechatronics Education

A. Altintas

Department of Electronics and Computer Education, Technical Education Faculty, Dumlupinar University,
43500, Simav-Kutahya, Turkey, phone: +90 274 5137917, fax: +90 274 5137914, e-mail: a_altintas@dumlupinar.edu.tr

Introduction

Robotics is an applied engineering science that has been referred to as a combination of machine tool technology, electrical, electronics and computer science. It includes such seemingly diverse fields as machine design, control theory, micro-electronics, computer programming, artificial intelligence, human factors, and production theory. Research and developments are proceeding in all of these areas. Advancements in technology will enlarge the scope of the industrial applications of robots. Industrial robots are available in a wide variety of sizes, shapes, and physical configurations. The vast majority of today's commercially available robots possess one of four basic configurations: Cartesian, Cylindrical, Polar (Spherical) and Jointed-arm configurations [1].

The 3-dimensional (3D) cartesian coordinate system provides the three physical dimensions of space. The relevant coordinates are of the form $P(x, y, z)$. The cartesian coordinate robot, illustrated in Fig.1a, uses three perpendicular slides to construct the x, y and z axes. By moving the three slides relative to one another, the robot is capable of operating within a rectangular work envelope. The notation for cylindrical coordinate system is $P(\rho, \varphi, z)$. The cylindrical coordinate robot, illustrated in Fig.1b, uses a vertical column and a slide that can be moved up or down along the column. The robot arm is attached to the slide so that it can be moved radially with respect to the column. By rotating the column, the robot is capable of achieving a work space that approximates a cylinder.

Robotics course is essential for engineering education and is included in the undergraduate-level syllabus. Also, robotics is a unique educational tool for many reasons including its ability to inspire students and motivate them to be creative. Robotics is a very interdisciplinary field; this situation makes the robotics course quite hard. In literature, some studies focusing on the robotics education have been reported [2–4].
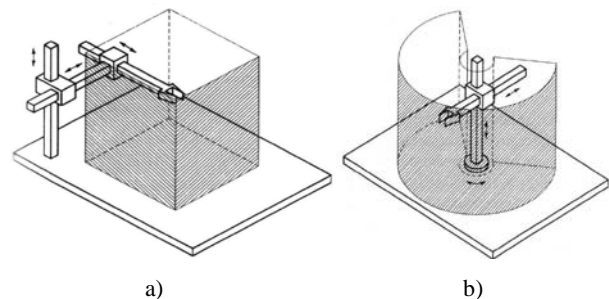

Fig. 1. 3D robot manipulators, a) Cartesian, b) Cylindrical

In this paper both the 3-axis cylindrical and cartesian coordinate robot manipulators actuated with stepper motors are designed and implemented; and a GUI (Graphical User Interface) with a programming page driven from Matlab program are developed for programming and controlling the designed robot manipulators. Stepper motors, unipolar type with two phases, can be run at different drive modes. Linear motions of the robots in question are performed with the ball-screw nut systems. Rotary motion in cylindrical robot design is realized with a timing belt. The GUI is built with benefiting from the facilities offered by the Matlab programming language. Matlab implements GUIs through GUIDE (Graphical User Interface Development Environment). Motion control of the robots is carried out with the parallel port of a Personal Computer (PC). 12-bit of the parallel port is used to drive 3-stepper motor. By the aid of the developed GUI and robots, the user will be able to control robot manipulators in individual and complex motion generation. As an exemplary application for the educational robotic system, a dc motor attached to the third axis of the robot manipulator is used to drill hardboard programmatically. The presented platform, user-friendly learning and teaching equipment, has been used for one semester by students attending courses on Robotics at the Technical Education Faculty of Dumlupinar University. The results from student feedbacks have been extremely positive.

Design Procedures of the Presented Robotic System

Matlab is matrix-based software for scientific and engineering numeric computation and visualization. Matlab is chosen as the programming tool primarily because of simple GUIs, immediate graphics facilities, built-in functions, the possibility of adding user-written functions, interactive mode of work, simple programming and its wide availability on computing platforms. These factors make Matlab a powerful tool for research and practical problem solving. Matlab implements GUI through GUIDE. A GUI is a user interface program built with graphical objects such as buttons, text fields, sliders and menus. In fact, these definitions are well known for almost all computer users. Applications that provide GUIs are generally easier to learn and use since the person using the application does not need to know what commands are available or how to use them.

The parallel port of a PC is used to drive stepper motors by using the DB-25 parallel port connector. The PC's parallel port is a simple and inexpensive tool for building computer controlled devices and projects. It has an 8-bit bidirectional and 9-bit unidirectional (4-output, 5-input) interface designed to transmit data synchronously using TTL signal level. TTL outputs can source 2.6mA and sink 24mA. Thus, small-size stepper motors can be driven from a PC's parallel port with a simple interface circuit containing a transistor. In this study, totally 12-bit excitation pulses (8-bit data register and 4-bit control register) are employed in robot motion control (4-bit for each stepper motor). Almost all programming languages allow programmers to access parallel port using some library functions. Matlab program also has some useful library functions such as digitalio, addline, putvalue.

A stepper motor can be a good choice whenever controlled movement is required. They can be used to advantage in applications where rotation angle, speed, position and synchronism need to be controlled. Stepper motors have found their place in many different applications. Because of these inherent advantages, stepper motors were chosen as actuators for robot-motion generation in this study. Since a hybrid stepper motor combines the best features of both the permanent magnet and variable-reluctance type stepper motors, this further increases the detent, holding and dynamic torque characteristics of the motor when compared with others. Hence, hybrid stepper motors with two-phase are used in robot-motion control [5–7].
As commonly known, there are three common types of drive for stepper motors: Wave, full step, and half step drive. Half step drive combines both wave and full step drive modes; every second step only one phase is energized; during the other steps two phases are energized. This results in angular movements that are half of those in wave or full drive modes. The hybrid stepper motors used in practice are MARCA MINEBEA Co. Ltd. ASTROSYN, Type 23LM-C355-P6V. Load-less maximum speed of the motor is 15.7 rad/s (~150 rpm). The stepper motor can maintain a stepper angle accuracy within ±0.45° at half step drive, ±0.9° at wave and full step drive.

The rotary motion of a stepper motor can be converted into linear motion by several mechanical means. These include rack-pinion systems, screw-nut systems, ball screw-nut systems, chains, belt and pulleys and other mechanical linkages. A ball screw with a threaded shaft provides a spiral raceway for ball bearings which act as a precision screw. As well as being able to apply or withstand high thrust loads they can operate with minimum internal friction. They are made to close tolerances and are therefore suitable for use in situations in which high precision is necessary. While reducing friction, ball screws can operate with some preload, effectively eliminating backlash between input (rotation) and output (linear motion). This feature is essential when they are used in computer-controlled motion-control systems. Therefore, rotary motion of the stepper motor is converted into linear motion by using a ball screw-nut system in this work.

The linear travel for per-step of the stepper motor is determined by the motor's rotary step angle and the pitch of the ball screw. The ball screw-nut system used in this study is 16mm diameter with a pitch of 5mm; thus, linear motion for per-step is 0.025mm (5/200) for wave and full step drive modes, and is 0.0125mm (5/400) for half step drive mode. As a result of this, the stepper motors can maintain a position accuracy within ±0.0125mm at wave and full step drive, ±0.00625mm at half step drive. In cylindrical robot design, rotary motion of the base is realized with a timing belt. The timing belt transfers torque from a stepper motor with a gear of 25mm diameter to a tambour of 96.5mm diameter. The overall system has a stepper angle accuracy within ±0.2331° at wave and full step drive, ±0.1165° at half step drive. Maximum linear travel of the cartesian robot axes $P(x, y, z)$ forming the working volume of the robot are 30cm, 30cm, 40cm, respectively. Maximum travel of the cylindrical robot axes $P(\rho, \varphi, z)$ are 30cm, 170°, 50cm, respectively.

GUI Overview

The main GUI windows of the cartesian and cylindrical coordinate robot control programs are given in Fig.2. These similar windows mainly consist of four-frame. The first frame entitled 'MAXIMUM RATINGS AND POSITION ERRORS' is the part for selecting drive mode of stepper motor, and displaying hardware limitations and position errors. The position errors vary according to the drive modes.

The second frame entitled 'PC CONNECTION & PROGRAMMING' is the section for setting up and closing computer connection, opening the programming page and etc. With the aid of the programming page, the user is capable of writing, editing, saving and running the operating codes. In order to use operating modes as shortened codes in the programming page, all of the operating modes are defined as built-in functions separately. Naturally, input parameters of the built-in functions will be different from each other. By using these built-in functions, the developed robots have been used to drill hardboard programmatically. Names of the some built-in functions of cartesian robot and detailed descriptions of those are presented in Table 1. Cylindrical robot has similar built-in functions, too. In fact, the developed robots can be employed at all sorts of position

and motion control by means of defining new built-in functions. These functions should be written in the programming page, given short-meaningful names, and saved with .m extension.
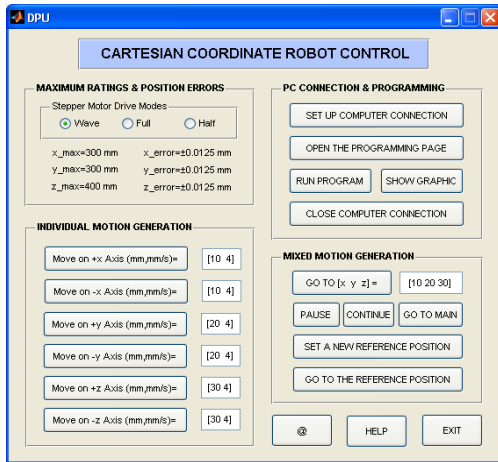
a)
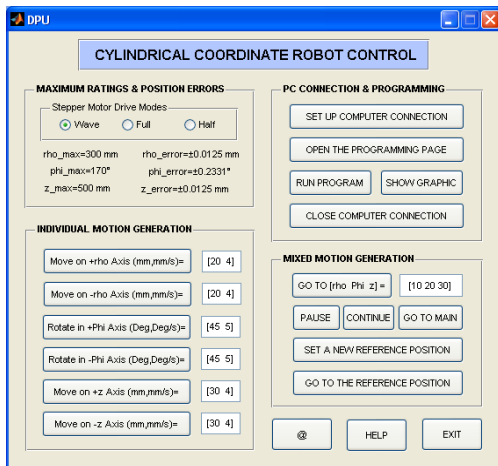
b)

Fig. 2. Main GUI windows of: a – cartesian robot manipulator; b – cylindrical robot manipulator

Table 1. Names of the built-in functions and descriptions on cartesian robot manipulator

Function	Description
Ccc	Closing computer connection
Cont	Continuing the action
GoRefPos	Going to the reference position set up before
GoMainRef	Going to the main reference position
Pause	Pausing the action
SetRefPos	Setting up a reference position
Movxp(x,n)	Moving in +x direction with a speed of n
Movxn(x,n)	Moving in –x direction with a speed of n
Movyp(x,n)	Moving in +y direction with a speed of n
Movyn(x,n)	Moving in -y direction with a speed of n
Movzp(x,n)	Moving in +z direction with a speed of n
Movzn(x,n)	Moving in -z direction with a speed of n
Mov(x,y,z)	Moving to (x,y,z) coordinates
Wait(n)	Waiting for n seconds

The third frame entitled 'INDIVIDUAL MOTION GENERATION' is the part of the moving the robot axes individually by pushing the related buttons. The buttons,

named with their own motion axes, take the content of edit boxes positioned on the left-side of the buttons, and send driving pulses to the related stepper motors. Edit boxes contains two values: the amount of movement and speed. If the amount of movement and speed are chosen from outside of the robot's hardware limitations, a warning message will be given to the user. Maximum speed of the linear motion is determined as 12.5 mm/s nearly. The buttons in third and fourth frames can be activated regardless of programming page.

The fourth frame entitled 'MIXED MOTION GENERATION' is the section for moving the 3 robot-axis jointly, running-pausing-continuing the movement, setting a reference position, and sending the robot actuator to the main reference position or to the defined reference position. By inputting the required 3 data related to the 3-axis into the edit box and pressing the button labeled with 'GO TO [x y z]', the robot manipulator can be positioned to these coordinates. At any time, the robot movements can be paused and maintained with the buttons 'PAUSE' and 'CONTINUE', respectively. Setting a reference position, done with the button 'SET A NEW REFERENCE POSITION', is a very important event because the robots perform all movements according to this reference position.

Fig. 3. A photograph showing the cartesian and cylindrical robot manipulators

An experimental contrivance which shows the cartesian and cylindrical robot manipulators is given in Fig. 3. Writing a computer program for robot manipulators is, naturally, a practical and wise way when production capacity, efficiency, flexibility and etc. are taken into account. When the user pushes the push button labeled with 'OPEN THE PROGRAMMING PAGE' in the second frame, the programming page will pop-up and offer the user writing the program code for controlling the robot manipulator. The program page takes the system parameters (stepper motor drive mode, hardware limitations and etc.) from the main GUI window, and defines them as global variables in order to access at any time. Functionally, the standard editor-page of the Matlab program has been used for the programming page.

An exemplary view of the programming page related to the cartesian robot is given in Fig. 4. The menus and toolbar located to top of the window can be used for different operations such as saving, debugging, running, printing and etc. The first seven-line in Fig.4 is the part of the defined global parameters, and set up the computer connection; this essential part must be added to all

computer programs. The written program can be run from editor-page directly or with the button named with 'RUN PROGRAM'. After running the program, the PC provides a connection via the parallel port, and controls the robot. At the finish, 2D (x-y axes) graphics of the drilled hardboards in accordance with the program can be displayed with the button 'SHOW GRAPHIC' placed in the second panel. Fig.5, obtained from the cylindrical coordinate robot program, shows the hardboard graphic drilled by a drill of 2mm diameter with intervals of 5mm and 10°.

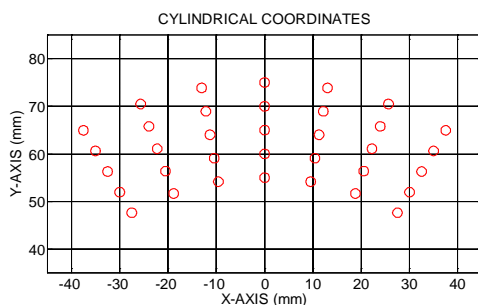**Fig. 4.** An exemplary view of the programming page related to the cartesian robot

Fig. 5. The drilled hardboard graphic obtained from cylindrical robot program

Conclusions

Robotics is a very interdisciplinary field, among others, involving areas from engineering, computer science, artificial intelligence, cognition, logics and mathematics. This situation makes the robotics courses quite hard to understand. Therefore, there have been complaints from students in classroom sessions. To overcome this problem, 3-axis cartesian and cylindrical coordinate robot manipulators are designed and realized; and a user-friendly GUI, built with Matlab program, are developed for controlling the robots. The presented platform, making the course more interesting, has greatly improved the effectiveness and efficiency of the learning process on robotics courses. Due to special abilities to provide accurate control over speed and position, stepper motors have a wide range of applications. Motion control of the robots is fully carried out with stepper motors in this work. Therefore, the platform can be, also, considered as a learning aid system for students to get a comprehensive understanding of stepper motors.

References

1. **Groover M. P., Weiss M., Nagel R. N., Odrey N. G.** Industrial Robotics: Technology, Programming and Applications. – McGraw–Hill, Inc., 1986.
2. **Mansour M. Schaufelberger W.** Software and Laboratory Experiments Using Computers in Control Education // IEEE Control Systems Magazine, 1989. – Vol. 9. – No. 3. – P. 19–24.
3. **Verner I. M., Ahlgren D. J.** Conceptualising educational approaches in introductory robotics // IJEEE, 2004. – Vol. 41. – No. 3. – P. 183–201.
4. **Manesis S., Koutri S.** A student–friendly approach to undergraduate teaching of experimental industrial control systems // IJEEE, 2003. – Vol. 40. – No. 4. – P. 255–269.
5. **Kenjo T., Sugawara A.** Stepping Motors and Their Microprocessor Controls, 2.ed. – Oxford University Press, England, 1994.
6. **Soltanpour M. R., Shafiei S. E.** Robust Backstepping Control of Robot Manipulator in Task Space with Uncertainties in Kinematics and Dynamics // Electronics and Electrical Engineering. – Kaunas: Technologija, 2009. – No. 8(96). – P. 75–80.
7. **Amann, R. Geiger F.** Simulation and Code Generation for a Parallel Kinematic Manipulator with three Degrees of Freedom // Electronics and Electrical Engineering. – Kaunas: Technologija, 2007. – No. 1(81). – P. 27–30.

A. Altintas. A New Approach to 3-Axis Cylindrical and Cartesian Type Robot Manipulators in Mechatronics Education // Electronics and Electrical Engineering. – Kaunas: Technologija, 2010. – No. 10(106). – P. 151–154.

In this study both the 3-axis cylindrical and cartesian coordinate robot manipulators actuated with stepper motors were designed and implemented. A GUI with a programming page driven from Matlab software was developed for programming and controlling the designed robot manipulators. The presented platform here has been intended to be used for educational purposes. The designed and mounted exemplary equipment greatly can improve the effectiveness and efficiency learning processes on robotics and related courses. Ill. 5, bibl. 7, tabl. 1 (in English; abstracts in English and Lithuanian).

A. Altintas. Roboto manipuliatoriaus galimybių tyrimas trimatėje ir polinėje koordinačių sistemose ir vaidmuo mechatronikos studijose // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2010. – Nr. 10(106). – P. 151–154.

Sukurti ir patobulinti robotai manipuliatoriai trimatėje ir polinėje koordinačių sistemose veikia su žingsniniais varikliais. Naudojant „Matlab" programų paketą sukurta robotą manipuliatorių valdanti ir kontroliuojanti sąsaja, skirta mokymuisi. Il. 5, bibl. 7, lent. 1 (anglų kalba; santraukos anglų ir lietuvių k.).