PROGRAMMING AND CONTROLLING OF RPP ROBOT BY USING A PLC

BOGDAN Laurean
University "Lucian Blaga" of Sibiu, e-mail: laurean.bogdan@ulbsibiu.ro

Keywords: Robots, programmable logic controller, programming, ladder diagram

ABSTRACT: The paper is presenting a simple and cheaper solution for programming and controlling of RPP robot by using a programmable logic controller (PLC). The RPP robot is designed and manufactured within the Faculty of Engineering from "Lucian Blaga" University of Sibiu as a patent no. 112418 CI6.B25J 18/02. Classical systems for programming and control of industrial robots is based on numerical control equipment developed around a computer structure. Programmable Logic Controllers have proven to be viable alternatives to driving machine tools, of industrial systems and Robots. PLC is well-suited for industrial environment. The paper highlights the simplicity and ease of configuration control and programming of the robot arm using a PLC. The paper proposes a method of programming and controlling a robot.

1. INTRODUCTION

Control and programming movements of a robot's mobile parts is a matter of great importance especially when the application requires constructive simplicity and is easy to implement. Achievements in the field of robot control and programming show various ways of solving the problem. The most common achievements are relying on systems configured around the structures of computer. Number of degrees of freedom of movement, as well as linking the rules of these movements, determine the complexity of control and programming of the robot [3,4,5]. The development of low cost computer has brought the most recent revolution, the Programmable Logic Controller (PLC). The advent of the PLC began in the 1970s, and has become the most common choice for manufacturing controls. PLCs have been gaining popularity in the domain of factories and will probably remain predominant for some time to come. Most of this is due to the advantages they offer: cost effective for controlling complex and flexible systems, where PLCs can be reapplied to control other systems quickly and easily, computational abilities allow more sophisticated control, trouble shooting aids make programming easier and reduce downtime, reliable components make these likely to operate for years before failure. A Programmable Logic Controller is a specialized computer, designed to be used for industrial control [4]. Programmable Logic Controllers are sometimes called programmable controllers (PC) but are more commonly called PLCs.

Ladder logic is the main programming method used for PLCs. As mentioned before, ladder logic has been developed to mimic relay logic. The decision to use the relay logic diagrams was a strategic one. By selecting ladder logic as the main programming method, the amount of retraining needed for engineers and technicians was greatly reduced.

The first PLCs were programmed with a technique that was based on relay logic wiring schematics. This eliminated the need to teach the electricians, technicians and engineers how to program a computer - but, this method has stuck and it is the most common technique for programming PLCs today [3]. There are other methods for programming PLCs. One of the earliest techniques involved mnemonic instructions. These instructions can be derived directly from the ladder logic diagrams and entered into the PLC through a simple programming terminal. Sequential Function Charts (SFCs) have been developed to accommodate the programming of more advanced systems. These are similar to flowcharts, but much more powerful. Structured Text programming has been developed as a more modern programming language. It is quite similar to languages such as BASIC.

2. PLC versus PC

The architecture of a PLC's CPU is basically the same as that of a general purpose computer; however, some important characteristics set them apart. First, unlike computers, PLCs are specifically designed to survive the harsh conditions of the industrial environment. A well-designed PLC can be placed in an area with substantial amounts of electrical noise, electromagnetic interference, mechanical vibration, and non-condensing humidity. A second distinction of PLCs is that their hardware and software are designed for easy use by plant electricians and technicians.

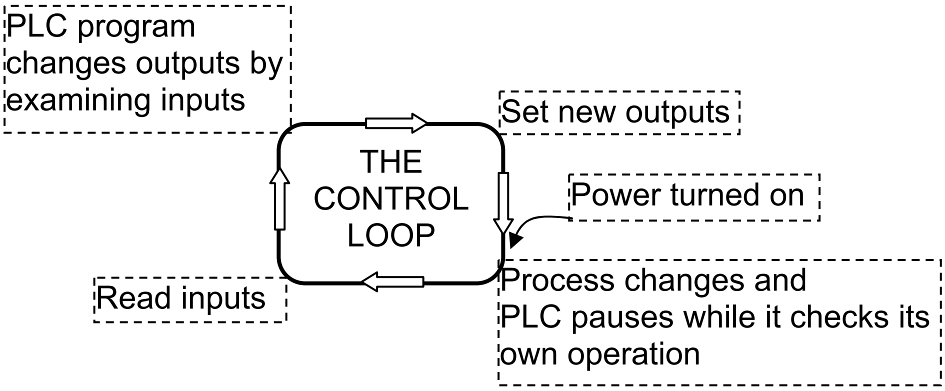

Fig. 1. Steps in execution operations of a PLC

The hardware interfaces for connecting field devices are actually part of the PLC itself and are easily connected. The modular and self-diagnosing interface circuits are able to pinpoint malfunctions and, moreover, are easily removed and replaced. Also, the software programming uses conventional relay ladder symbols, or other easily learned languages, which are familiar to plant personnel. Whereas computers are complex computing machines capable of executing several programs or tasks simultaneously and in any order, the standard PLC executes a single program in an orderly, sequential fashion from first to last instruction, (Fig. 1). Bear in mind, however, that PLCs as a system continue to become more intelligent. Complex PLC systems now provide multiprocessor and multitasking capabilities, where one PLC may control several programs in a single CPU enclosure with several processors. With the proliferation of the personal computer (PC), many engineers have found that the personal computer is not a direct competitor of the PLC in control applications. Rather, it is an ally in the implementation of the control solution. The personal computer and the PLC possess similar CPU architecture; however, they distinctively differ in the way they connect field devices. While new, rugged, industrial personal computers can sometimes sustain midrange industrial environments, their interconnection to field devices still presents difficulties. These computers must communicate with I/O interfaces not necessarily designed for them, and their programming languages may not meet the standards of ladder diagram programming. The personal computer is, however, being used as the programming device of choice for PLCs in the market, where PLC manufacturers and third-party PLC support developers come up with programming and documentation systems for their PLC product lines. Personal computers are also being employed to gather process data from PLCs and to display information about the process or machine (i.e., they are being used as graphic user interfaces, or GUIs). Because of their number-crunching capabilities, personal computers are also well suited to complement programmable controllers and to bridge the communication gap, through a network, between a PLC system and other mainframe computers.

Some control software manufacturers, however, utilize PCs as CPU hardware to implement a PLC-like environment. The language they use is based on the International Electrotechnical Commission (IEC) 1131-3 standard, which is a graphic representation language (sequential function charts) that includes ladder diagrams, functional blocks,

instruction lists, and structured text. These software manufacturers generally do not provide I/O hardware interfaces; but with the use of internal PC communication cards, these systems can communicate with other PLC manufacturers' I/O hardware modules [5]. Communication and transmission of the signal within the sensing system are generally processed in digital form after digitization of the analog input signal. The analog transmission of the sensed signal prior to digitization requires special care, as the quality of the signal transmission directly influences the quality of sensing. The analog signal is easily deteriorated by the noise signal surrounding the transducers/sensors and the signal transmission cables.
The high-frequency noise signals coming from the power circuits including the motors, the digital devices, etc., as well as those coming from the power supply can be major sources of noise signals. The signal transmission requires special techniques when the signal is to be transmitted via relatively moving interfaces without contact. The slip ring, wireless transmission with use of radio waves and optical methods are generally employed in such cases.

3. Application

At "Lucian Blaga" University of Sibiu, in field of robots, there is experience in cinematic structure of robots, work space, as well as in terms of programming and control movements. The model presented in this work, that is the programming and control of movements using a PLC, is the result of a patent of invention. Experimental robot model is manufactured using a structure with cylindrical coordinates (Fig. 2). Cylindrical coordinates allow robot arm to move in space under coordinates.

The Robot can make a rotation and two translations (rotation, translation, translation - RPP). Mathematical matrix of reference transformation $[A]_{0,n}$ is (OZYX)0 in reference system (OZYX)n, results three operators $[P]_{z0,z}$, $[R]_{z0,\alpha}$ and $[P]_{x,r}$:

$$[A]_{0,n} = [P]_{z0,z}[R]_{z0,\alpha}[P]_{x,r} \qquad (1)$$

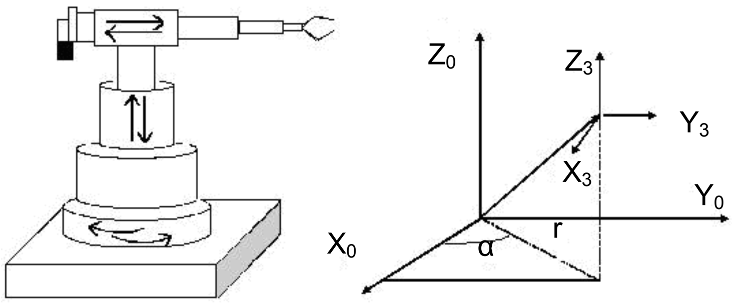

Fig. 2. RPP robot

The robot is relying on cinematic structure with three movements: Rotation-Translation-Translation. Experimental model of robot was designed and manufactured at the Faculty of Engineering, (patent no. 112418 CI6.B25J 18/02), [2].

3.1. ROBOT STRUCTURE

The robot is based on a modular structure. Making movements by elastic elements: belts with cable, allows displacement on distance with high dynamic performance. The kinematics structure allows the robot to realize movements based on wire mechanism. For each axis movement is necessary a DC servomotors. The motor type is EVP DC 12V, 120 Watts. Switching between rotation sense is made by an intermediary relays which support 10 Amp on contacts.
The displacement of robot arm, can be achieved independently or simultaneously depending on software and hardware used. Telescopic mechanism of the translation module is the degree of mobility equal to 1 (Fig. 3).

S_A represents input of mechanism, F_A represents mechanical force. Number of external links is equal to 2: an input S_A and an output S_B. The function of transmitting forces depends of S_B and F_B. S_B depends of S_A. To determine the movement's velocity of the robot arm, we have two situations, when the C is fixed, respectively when the D is fixed. Due to construction, module translation movements are fine and can reach high speeds. Because of the elastic structure of movement's transmission, telescopic modules are recommended to be used at works that do not exceed 30N.

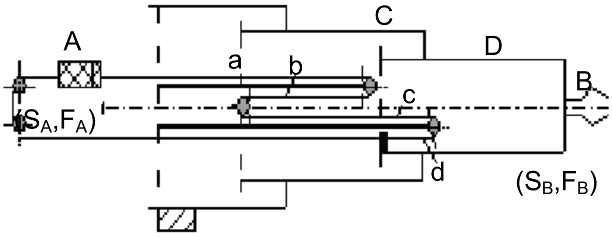

Fig. 3. Telescopic mechanism from robot structure

3.2. Hardware configuration

The PLC used for application is Moeller Easy 512 DC RC and programming is possible in diagram DIN IEC based on relays wiring as electrical diagram or in Ladder diagram. Output Q01 is for the right displacement, Q02 is for the left displacement. The external relays will be K1 and K2. One of the relays will serve output Q01 and other relay for output Q02. The inputs/outputs connections for the Moeller Easy, is presented in figure 4. Power supply is provided by external source 24 DC Voltage and protection is provided by F1. The elements connected to the PLC as digital inputs are: start switch and encoder 1 I02. The elements connected to the PLC as digital outputs will be: relay K1 and K2. Relays K1 and K2 provide change of rotation sense of DC motor. The two relays K1 and K2 form an H bridge for DC motor supply and for sense of rotation change Fig. 4).

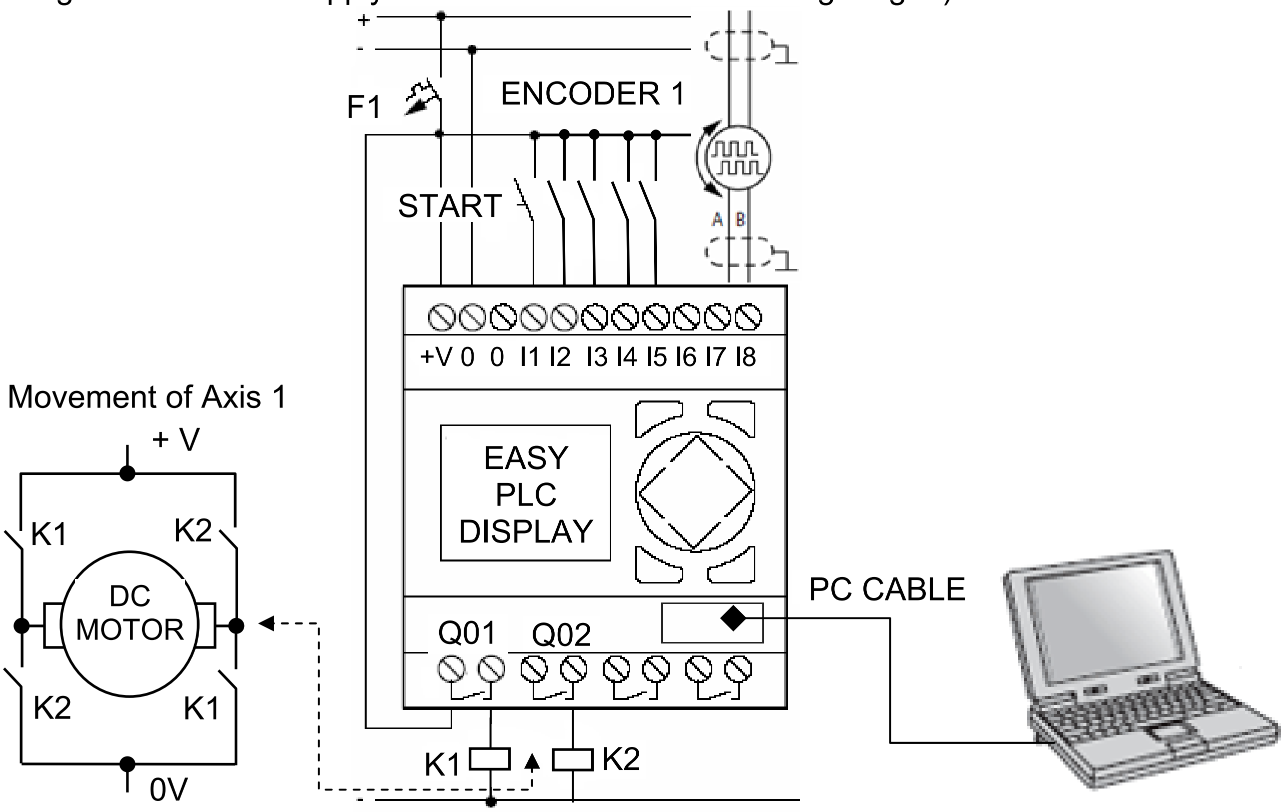

Fig. 4. PLC, inputs, outputs and PC connections

The connection between PLC and PC uses a serial cable whereas programming is made in EASY-SOFT6 Pro software [5].

3.3. PLC programming in Ladder diagram

The operands, the lines of program and the function block from ladder diagram are presented in figure 5. The second column shows rung and function block corresponding to field area from ladder diagram. On the third column we have the role of each operator.

Operand	Rung/function block	Comment
I01	002A	START
I02	001A, 007A	ENCODER1
Q01	005G	RIGHT
Q02	011G	LEFT
M01	002G, 003A, 005A, 007C	MEMORY1
M02	003E, 009G, 010A, 011A	MEMORY2
C01	001G, 003C, 006A, 006G, 009A	COUNTER1
C02	004A, 007G, 008A, 008G, 010C	COUNTER2

Fig. 5. Operands, function blocks and role of inputs and outputs

The program contains 11 lines. The pulses from encoder are received by Counter 1. The right displacement of robot arm is provided by Q01, the left one is provided by Q02. Internal memories M01 and M02 ensure function block for logical ladder diagram. The program begins with reading status of each input. When I01 (START) will become active, the robot arm will move and the encoder will send pulses to counter 1. Counter 1 will determine changes in the next lines of program (Fig. 6).

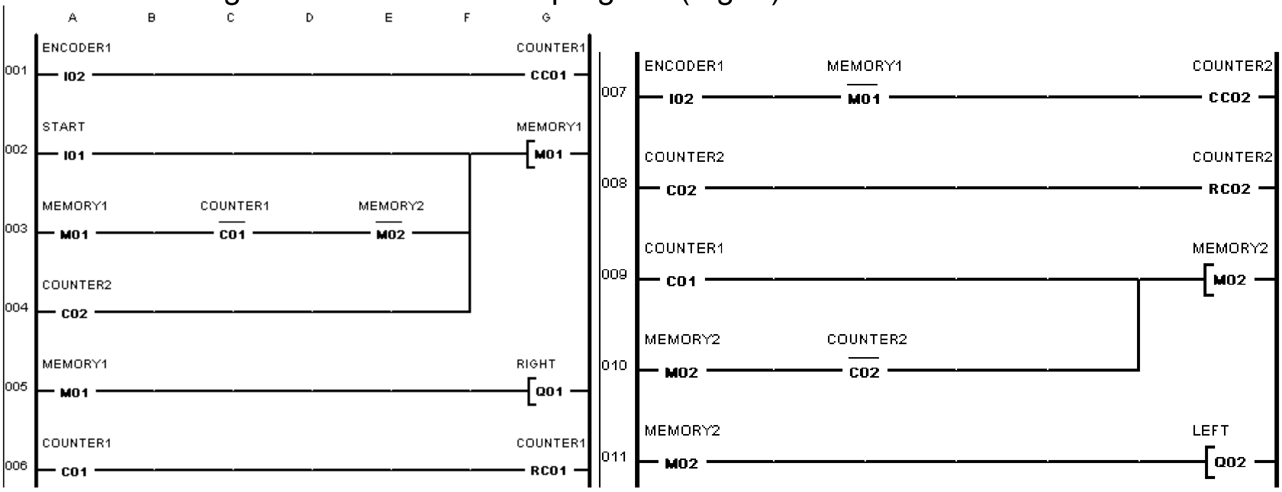

Fig. 6. Program of an axis in Ladder diagram

3.4. Condition monitoring inputs and outputs of PLC

The number of programmed pulses from PLC depends on the value of arm displacement. When the number of pulses from encoder becomes equal with pulses from PLC the status of outputs will change. The status of outputs Q1 and Q2 are changing after each cycle of pulses received from encoder, input I2. In figure 7 is presented status of input I2 and outputs for two pulses and for ten pulses. The status of inputs and outputs is monitored using the oscilloscope from PLC software.

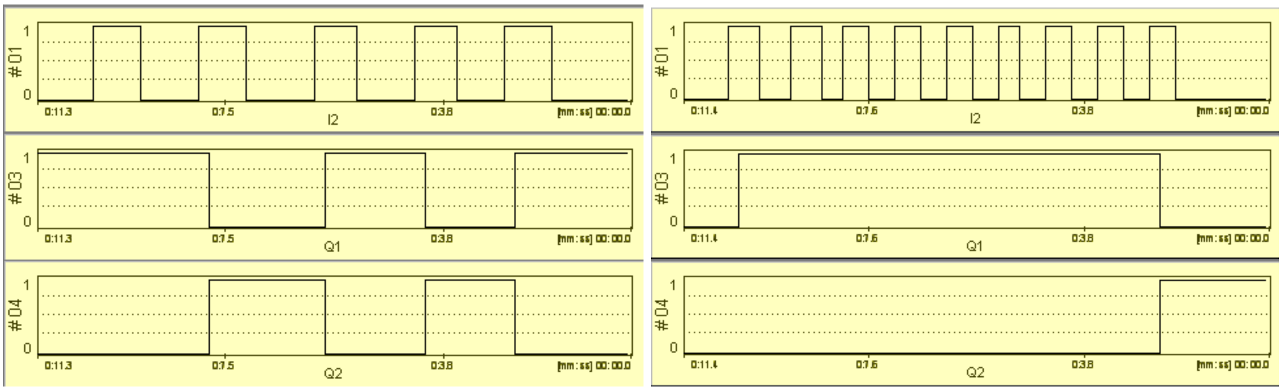

Fig.7. The status of inputs and outputs

The internal oscilloscope from software is running in simulation modus. Simulation cycle time can be set between 30 and 3000 milliseconds. For the simulation here being presented one used a simulation cycle time of 100 milliseconds. For each 100 milliseconds the program in ladder is executed cyclical.
The time for changing any inputs is much higher than 100 milliseconds. The changing of any inputs consists in START switch. The counter CC01, counter CC02, Memory M01 and Memory M02 are internal virtual image of inputs and outputs. This is the advantage of PLC as compared with hard relay controller.
Counters and memory status are not to be seen in simulations of inputs' and outputs' status. The result of counters and memory status change will be Q1 and Q2 change (Fig. 7). Q1 and Q2 are physical internal PLC relay that determines command of DC motor for robot arm movement.

4. CONCLUSION

Using PLC for programming and control of movements of industrial robots is possible, easy to apply and cheaper than classical numerical control. The PLC is capable of being reprogrammed and used for other tasks. The most important capability of it is the one to be programmed and maintained easily. PLC Moeller has cycle time between 30 and 300 milliseconds. Ladder diagram offers a simple method for robot programming, whereas language is simple and easy to use. The number of instructions for programming movements of an axis is small, while flexibility in program changing is high. The system response time is very good.

REFERENCES

[1]. Ben-Zion, Sandier (2000). *Robotics: Designing the mechanisms for automated machinery*-2nd ed. Academic Press 24-28 Oval Road, London NW1 7DX, UK. http://www.hbuk.co.uk/ap/, accessed: 2009-01-12.

[2]. Bogdan, L. Analog control and programming of a RPP robot with a PLC, *Proceedings of the International Conference on Manufacturing Systems – ICMaS*, Vol. 4, 2009, ISSN 1842-3183, University POLITEHNICA of Bucharest, Romania.

[3] Bryan, L.,A., Bryan, E., A. (1997). *Programmable Controllers, Theory and Implementation.* Second Edition. An Industrial Text Company Publication, Atlanta, Georgia, USA.

[4] Hugh, Jack (2005). *Automating Manufacturing Systems with PLCs,* Version 4.7. http://claymore.engineer.gvsu.edu/~jackh/books.html, accessed: 2009-02-10.

[5] ****Easy500, Easy700, Control Relay*-User Manual, 1st published 2004, edition date 05/04, © Moeller GmbH, 53105 Bonn.