DESING OF A KINEMATIC CONTROL MODEL, FOR AN ANTHROPOMORPHIC ROBOT ARM, APPLIED TO THE TEACHING OF INDUSTRIAL ROBOTICS.

Rodrigo Escandon C.[1], *Member, IEEE,*rescandon@ieee.org., Marco Carpio A.[2], mcarpio@ups.edu.ec.

Universidad Politecnica Salesiana del Ecuador, 2013. Carrera de Ingenieria Electronica.

***Abstract*—The current work presents a preliminary study on the types and models of industrial programming, These techniqies allowed us to desing a controller for positioning and trajectory functions of a didactic anthropomorphic robot arm. This design is based on a real kinematic model; furthermore, the system of the controller and robot ilustrate the basics of industrial robotics and become teaching tools at the undergraduate level.**

***Index Terms*—Industrial Robotics, degrees of Freedom, IDE systems, DAQ systems, path functions, articulate variable.**

INTRODUCTION

INDUSTRIAL robotics studies all things related to robotic manipulators, as specified by the International Standards Organization (ISO). An industrial robot is "A multifunctional manipulator, reprogrammable, with several degrees of freedom, capable of moving parts, operates tools, or other special devices, as paths programmed to perform various tasks" [1].Due to the current impact that industrial robotics has in regard to the industry and its high social impact, Industrial robotics is considered as an integral matter, covering many subject areas among which are: Theory of Mechanisms, Mechanical Design, Electronic Control, Electric Power Systems and Object-Oriented Programming. The project's main objectives are to develop an academic software for monitoring and controling articulated robots arms and to provide better services to educational institutions to impart these skills as part of vocational training of their students.

PREVIOUS WORK.

Currently it is estimated to be at least one and a half million robots in industrial areas worldwide, mostly in Asia, followed by Europe and America. There are also statistics in regards to industrial growth, specifically, some findings show that there is an industrial robot for every seven human workers. The industrial impact of robotic devices is reflected also in the social sphere, if industry did not have the number of robots it has today, it could not continue with its current production. For these reasons modern concepts of industrial robotics are an indisputable part of engineering training.[2]

Specialized academic computer systems for controlling electromechanical or electronic plants often use a program based on the objectives of the particular course in addition to considering environments that are familiar for the users, in this case engineering students. In consequence, by presenting our work with the controller and robot we aim that students learn how to make changes to the manipulator control IDE system , and we provide a flowchart for graphical programming environment. This project was designed on Matlab and Simulink platforms, which meet all the software parameters for this work.

Academic software used for electro mechanical systems.

Many electromechanic devices are used for studying engineering and are also used for coming near and practice certain concepts that are taught in specific classes. Also, it is important for a high quality education that the monitoring and controlling programs satisfy both professionals and students' needs. This is the reason why enterprises such as Feedback, National Instruments, etc.; that construct and distribute these devices program the software by utilizing platforms that are familiar to the students and that also allow the integration of additional systems to the original controller.

Software for integrated development that exist in the market and that are used for controlling academic devices, for instance Labview, C++, Java, etc.; are not regularly used during teaching engineering classes but are rather used in advanced specialty courses. This is why we opted for utilizing MathWorks as an academic platform (to develop our Project) that offers an interactive environment with external devices and also offers tolos for processing and interpreting signals where engineering students feel much more familiar.

Rodrigo Escandon, Carrera de Ingenieria Electronica, Universidad Politecnica Salesiana, Cuenca, Ecuador, e-mail: rescandon@ieee.org.

978-1-4799-2722-7/13/$31.00 © 2013 IEEE

Industrial programming levels.

The industrial robot programming, can be classified into three levels. These levels are restricted by the sequence of commands that the robot has to execute. The task given to the robot is defined around an object. We call Robot level when you need to specify each movement that the robot needs to do so that it accomplishes the task. Object level is when programming is done to describe the actions that the robot needs to do to manipulate the object. Task level uniquely describes the overall objective that the robotic system intends to perform.

The current state of the robot programming is maintained at the first level because of theneed to generate code from the robot at the higher levels, this is required for the robot-movement and to specify its movements, all this is tipically necessary in industrial application.[3]

Industrial robot programming methods.

The best approach to conceptualize the methods of industrial programming is to refer to the system used to indicate the sequence of actions to be performed by the robot, this approach classifies programing in two types: programming known as guided or assisted learning and textual method. Guided or assisted learning is a direct manipulation of the actuators of the robot for it to follow a sequence of actions and to keep a memory of this sequence so the robot is able to repeat them. In different ways for its different works. The textual method is an alternative method where the user programs or gives guidelines to each one of the functions that the robot is to accomplish.[4]

Interface Development.

The process for the design of the interface first requires to establish certain electrical, electronic and mechanical features specific for the robot type to be used. These features need to be taken into consideration at the time of developing the interface. The robot type that we used is known as anthropomorphic because of its resemblance to a human arm. The main characteristic of this type of robot specifies a minimum of five degrees of freedom where three define the position of the manipulator, while the remaining two define the orientation of the end effector. Other components that are part of the sensory systems are both internal and external parameters that will be used in particular for programming control algorithms provided in the interface. Once finished the conceptualization of all the attributes for the robot arm, it is necessary to calculate the kinematic equation which is implicit to the anthropomorphic robot. The kinematic equation will be programmed in the interface benefiting more to the function of the system Remarkably, besides utilizing this project to teach industrial robotics, the use of the interface can be expanded and innovated, allowing changes in the functions and tasks that the robot can do.

The project involves the installation of a physical link using standard network protocols to manipulate the robot from an external computer. This network may communicate with the IDE software interface, allowing its use also for this type of robot.

Anthropomorphic industrial robot arms.

An anthropomorphic robot arm is made only by rotational joints, allowing them to have a range of motion for its moving parts for the effector end of the arm to reach a position.Taking into consideration that the actuators used are DC motors (which have internal position sensors allowing to control the manipulator orientation and position as desired). The anthropomorphic robot has limitations which have to be programmed in the interface so the user is able to control further changes in the system.

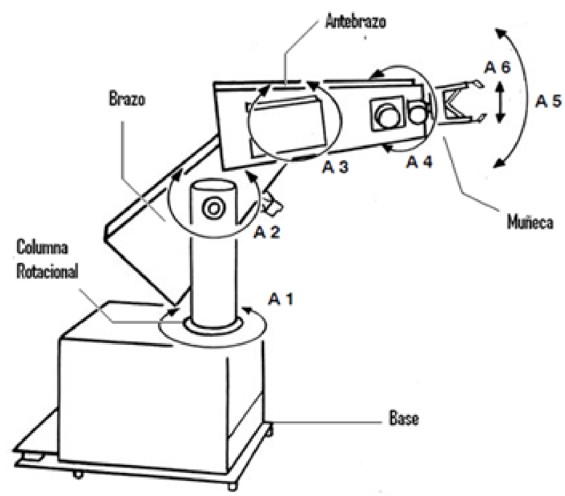

Figure 1. Mechanical Structure, Mentor anthropomorphic robot arm[5]

The image shows the movement diagram of anthropomorphic robot arm "Mentor". This device could be utilized in academia for laboratory tests and the implementation of the monitoring and control programmed interface. The physical characteristics of the robot allow a practical approach to understanting the its transmitting mechanisms, such as gears and couplings.

According to the objectives of the project this feature provide higher performance because allows the student a better approach to the system. Reducing systems, gearboxes, motion transmission systems and sensor systems among others are some features that the robot has, this features represent mechanical considerations that need to be taken in account at the time of the application. for their proper use.

ITEM	DESCRIPTION	VALUE
Loads	Standar Load	1Kg.
Working Area	Max. Range	428 mm
Other Data	Axles	6
	Repeatability	± 2cm
	Mounting positions.	Floor.

Table I
Specifications Mentor anthropomorphic robot arm

The "Control System" is the most important part because it will be responsible for interpreting the signals given by the interface and robot arm movements. The system name comes from the need to allow users handling end control algorithms, and is necessary the full knowledge of the robot, as well as the different components. The components of the robot arm control system include: an electronic circuit consisting of a power amplifier and a data acquisition circuit; these components interact with the interface that is set up in the computer.

Interface Design.

The principal objective of the interface is the control of the robot motion which is based on trajectory functions, or motion parameters defined by the user. Considering the implementation of this software in academia intends to identify robotic plants, to allow programming of diverse types of contols and to allow better performance in beginner academic levels.

Simulink, a IDE graphical environment, was utilized for enhancing the performance of the interface and to allow flexibility of programming of the robot by this means the student is able to change the programing according to the student's control algorithms.

The interface is formed by libraries which make up the "toolbox", akin the interface. There are two libraries, one that has to do with the movement of the robot and one that has to do with the communication to and from the robot; these libraries feedback to each other to orchestrate the function of the robot and respond to the parameters and trajectory functions that the student designs.

The image shows a specific example that uses the libraries created for control and monitoring of the "Mentor" robot arm. The example shows a simple model of a fuzzy positioning controller. In this case the user can determine the desired joint variables for the robot movement. Further, the rules set in the controller sends the data to the robot moving its joints to the required point. This model also allows assessment of the robot's actual response and physical position based on a computer simulated model of the robot.

The findings considered for the analysis of the results were obtained based on the control model of the previous example.

METHODOLOGY.

The robotic manipulator was subjected to hardware and software tests, because this electromechanical system needs these two parameters for its proper functioning. These tests allowed assessing the parameters of control and the physical responses of the robot; this allowed a proper evaluation of the robot communicating attributes that were programed in the interface.

The initial tests were specific to the software developed, these tests were developed based on the design of test cases, evaluation of the test cases helped to improve the functions of the interface by detecting and reducing programming errors prior to testing laboratory cases. Laboratory tests are taken on a controlled environment allowing to verify that the results are correct. These tests provide an evaluation of the software and also allow an evaluation the physical performance of the robot, in this way we can assess determinant values of the robot such as position and accuracy.[6]

Software Tests.

The tests that were conducted were based on the previously defined "test cases", two test cases go together and are intended to find mistakes in the software, the test is considered to be successful if any mistake is indeed found. To emphasize, passing the test does not guarantee the absence of errors, it can only show the software defects that have to do with these tests; in this way we can make changes to the program on the specific parameters that were tested. The test cases were obtained based on standard parameters known as black box and white box sequences that aim to raise test parameters for a specific software.[7]

- White box testing: Test cases designed by this method perform a thorough examination of the procedural details, checks sequence algorithms and examines the state of the program at various points; for this reason the tests conducted under the provisions of white box are supposed to have perfect programs, due to their level of testing, defining all logical paths, develop tests for all these paths and evaluating the results. White box tests assess the software algorithms in two ways:: Basic road tests and Loop tests. The former tests give a measures the complexity of the algorithm using a notation of the program flow graph and the cyclomatic complexity calculation. Loop tests take care of the loops that are repetitive sequences. These tests ensure that all paths of the algorithm are executed at least once.
- Black Box Testing: Test cases designed in this way, unlike the white box testing,disregard the internal behavior and structure of the program; it focuses on the analysis of the software interface, this is why the studingtest cases greatly contributes to the project. The parameters used for Black Box Testing are presented in equivalence classes, these ensure the proper functioning of the interface by only considering the input and outputs to compare them with the optimal results.

Hardware Tests.

Hardware tests assess the robot physical responses to the signals emitted by the computer (programmed in the parameters of the interface); they also allow to clarify the precision and accuracy of the parameters of the system. The methodology used for hardware testing consists on evaluating the robotic actuator sequence of movements, i.e. measuring the range of motion, timing and precision of movement from a starting to an ending point. In order to obtain conclusive results about the quality of the software, we conducted two experiments and various processes. The experiments consisted in repeating sequential movements where we measured the degrees of freedom and did a physical analysis of the robot response that we will discuss below.

Data from the position sensors after performing a movement was stored in a database for further analysis. One hundred samples were obtained from each sensor; we obtained an average which was used as a reference for each of the 20 experiments, after repeating this acquisition for 10 processes, with a total of 200 movements. Statistical analysis of the data

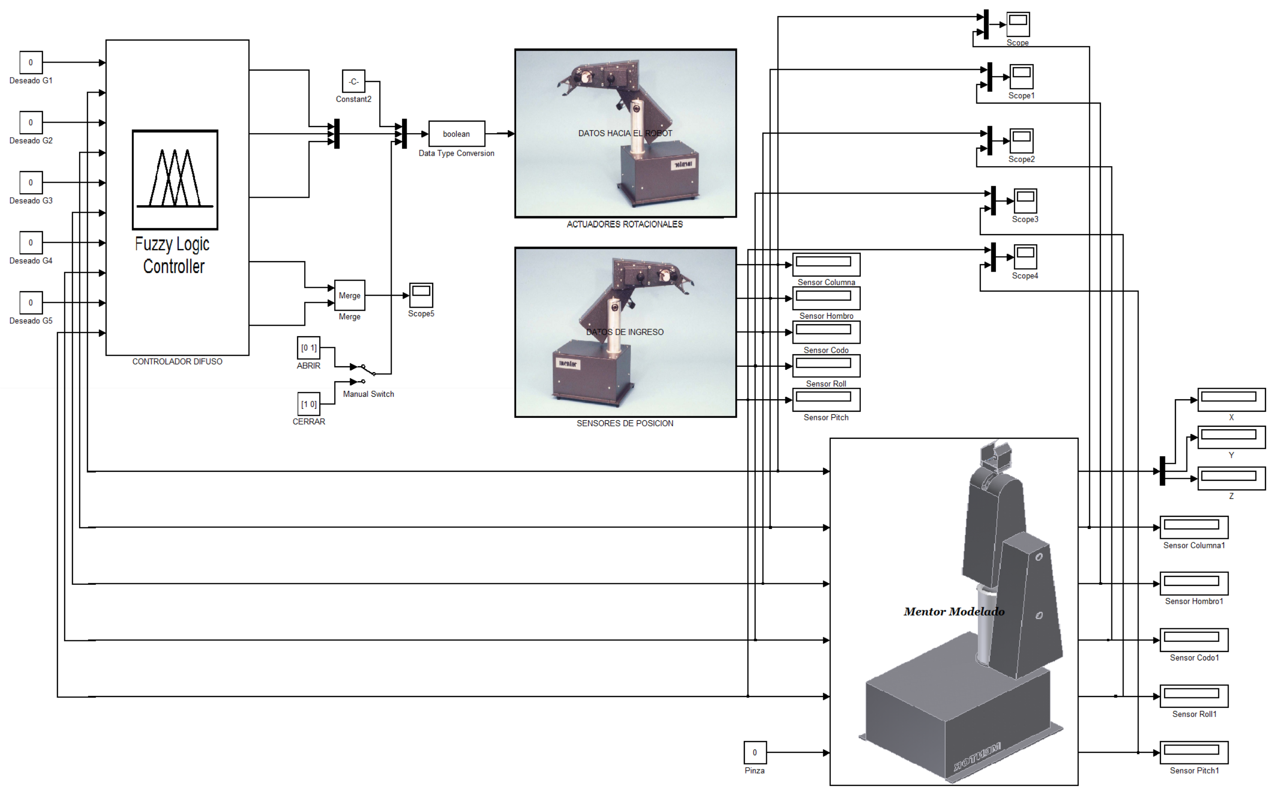

Figure 2. Example of using the toolbox for the Mentor robot arm control.

Figure 3. Motion picture used for experimentation.

Repetition	A1	A2	A3	A4	A5
1	-43,1455	77,5089	56,7902	9,44877	29,34688
2	-41,3667	80,1042	59,1765	9,33039	30,3510
3	-42,4839	83,2536	59,3538	9,06405	30,1350
4	-39,7595	79,1819	57,9713	10,0111	30,8273
5	-39,5176	81,3794	59,1605	9,59674	29,9357
6	-40,3983	80,4408	59,4076	9,15283	29,1396
7	-42,0286	79,3352	59,6582	9,09364	30,1607
8	-39,7528	80,1477	57,2841	9,47837	30,4203
9	-38,255	82,2587	59,4545	9,47837	30,7644
10	-42,8339	78,1565	56,9672	10,4846	30,5241
11	-39,6203	77,6124	63,766	9,86309	29,4165
12	-40,7435	79,2497	59,5739	10,0702	30,8356
13	-38,7248	77,7714	54,1685	9,3008	29,2607
14	-39,6344	79,6134	60,9187	10,3366	29,9537
15	-39,1918	79,1335	62,1449	10,8693	30,5587
16	-41,6211	79,208	57,4062	10,0702	29,6588
17	-37,4107	81,4106	60,8233	10,3662	29,7972
18	-42,6379	79,3558	58,0605	9,77431	30,9568
19	-42,9515	82,3350	60,8372	9,3008	29,6242
20	-41,3126	77,6096	58,5783	10,1886	29,5549

Table II
SAMPLES ACQUIRED THE 10TH EXPERIMENT OF THE 5TH TESTING PROCESS.

was performed to assess the validity of the results. We were able to compare the experiments and draw conclusions in regards to certain parameters, especially those involving the robot mechanical part. A sample of the data is shown here, this is data from a simple experiment, where the end point determined the degree of freedom. A1 = 40, A2 = 80, A3= 60, A4 =10, A5=30, the robotic manipulator.

With these data we were able to perform statistical analysis and calculate the parameters of precision and accuracy of the interface. To note, each part of the data results from an average of statistical samples obtained from a position sensor linearized by a mathematical function. This average can be affected by driver errors and should be considered a rounding error of the ecuation.Figure 4 illustrates the samples acquired in the 10th experiment of the 5th testing process.

RESULTS.

First of all, statistical parameters of precision and accuracy were calculated based on analysis of all the samples and testing processes performed. We utilized probability distribution, which allows to model different phenomena of a general nature without not even knowing the phenomenon as such. We utilized a normal distribution of Gaussian distribution; this is most often used and is considered optimal for real phenomena. Its density function has a flared shape and expresses the result in values between 0 and 1 allowing to establish statistical criteria based on the behavior of the samples regardless of their nature; the distribution satisfies the following equation.

$$f(x) = \frac{1}{\sigma\sqrt{2\Pi}} e^{-\frac{1}{2}\left(\frac{x-u}{\sigma}\right)^2} \qquad (1)$$

Where u (mu) is the mean, σ is the standar deviation, and σ^2 is the variance.

Precision and accuracy can be determined with this distribution. Precision is calculated as the standard deviation since it determines the distribution of samples, accuracy is determined as the difference between the mean of the signal and the reference value given.[8]

Calculation of each of these parameters will be made after organizing and acquiring the data, defining the samples acquired in a graph. As seen in the figure the distribution of samples of a specific experiment for a defined degree of freedom, by using the Histogram tool allows observing a relationship similar to the samples calculated from the normal distribution.

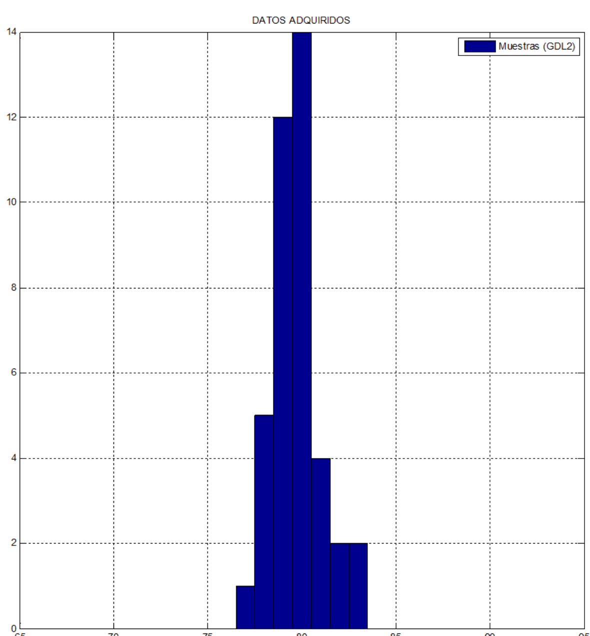

Figure 4. Histogram of the samples of the second degree of freedom of the fifth test process.

The result of repeating the process and calculating statistics for each of the degrees of freedom in the processes and experiments produce a global estimation of the entire system parameters. The data allow us to have a valid project because it was obtained based on the calculation of the average of all processes and experiments thus allowing a more real and accurate relationship to the programmed interface.

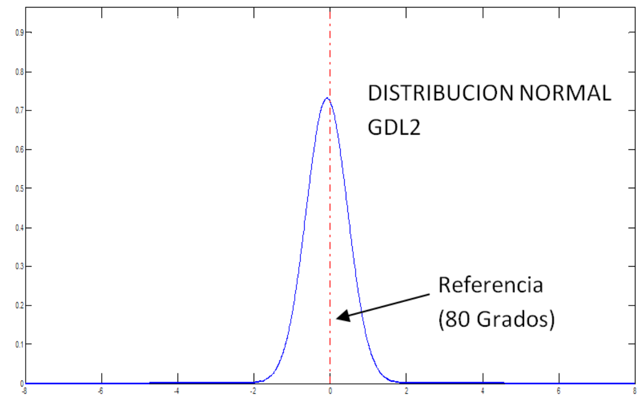

Figure 5. Normal distribution of the second degree of freedom of the fifth test process.

The following table summarizes the data mentioned above and defines the subsequently obtained parameters to physical evidence of the interface. It should be noted that these parameters were obtained after completing a complete path for the robot, which is why these data do not provide information about the behavior of the robot during the path, ie their response before reaching steady state.

GDL	Accuracy	Precision
1	0,70938	1,50246
2	0,76142	1,65288
3	1,37594	2,75858
4	0,51172	1,28808
5	0,27804	1,02548

Table III
AVERAGE VALUES OF ACCURACY AND PRECISION OF THE SYSTEM.

The subsequently analysis from the physical response through the use of the response to an abrupt change in each degree of freedom, thus allowing obtaining data to report on the behavior of the robot while performing sequential movement.

The data acquired for this structure unlike physical response steady state data are acquired in real time similarly to the position sensors, the same as mentioned above were linearized in order to specify a value in tension but no equivalent in degrees, thus allowing the analysis of each degree of freedom separately as was done with testing and statistical analysis above.

$$\theta = \left(\frac{\theta_{max} - \theta_{min}}{V_{max} - V_{min}}\right)(V - V_{min}) + \theta_{min} \qquad (2)$$

Where θ is the value of the articulate variable in degrees, $\theta_{max}, \theta_{min}$, are the limits of the work volume for each degree of freedom, in degrees, V_{max}, V_{min}, are the voltage limits that the sensor can provide and, V is the value that the sensor has at any time or status change, measured in volts.

The equation allows to build a signal as it is illustrated on Figure 6. With the signal we can define its own control

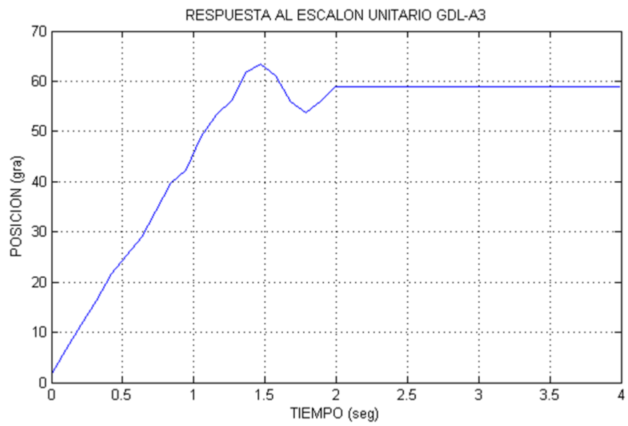

Figure 6. Physical response to a movement, GDL3

parameters that are referred to the physical response of the system. In addition this response shows the real movement of the robot in each one of its joints. This information is relevant for teaching control systems.

GDL	Settling time	Real value reached
1	5.670 seg.	89.29062
2	5.024 seg.	90.76142
3	2.042 seg.	58.62406
4	1.030 seg	90.57172
5	1.213 seg.	44.72196

Table IV
PARAMETERS FOR THE MECHANICAL MOVEMENT RESPONSE.

The parameters shown in the table refer to a movement that starts at a point of zero degrees for all degrees of freedom, resulting in a point defined by: A1 = 90, A2 = 90, A3 = 60, A4 = 90 and A5 = 45. As you can see, the position error is related to the previous calculation and also allows us to observe that in spite of the establishment time is lower in the degrees of freedom which are defined by links physically smaller. The physical response could vary according to the mechanical inertia of the device; this creates a more oscillating physical response before reaching a steady state, as shown in the figure above.[2]

CONCLUSIONS.

The analysis in regards to the types of programming and to the current prototypes and applications allow to better understand the interface, giving attributes and designing the prototype that help the theoretical teaching, also, it will allow in the future, the innovation of programming and will allow users to improve the algorithms for controlling robots.

The follow-up study, allowed us to establish the main recommendations of the project after its implementation, mainly because the next stage in the study of the subject should entirely focus on the programming of robotic manipulators and consider specific industrial standards. To note, the platform that we propose is able to implement real time systems and to control external devices. In this way, this platform can be utilized in higher levels of education such as masters and doctorates that have to do with industrial robotics.

Statistical analysis showed a physical response of the robot to the control parameters of the interface. The robot showed very acceptable response to the control programming and the feasibility to accomplish specific tasks.

The robotic manipulator physical responses were affected by an error of inertia. The robot is a kinematic chain where every link depends on the previous one, even though they should be independent from each other the robot depends on its mechanical attributes.

The implemented control and monitoring software, have a role in teaching of great value. It also is warranted to create a manual of manual of practice or a protocol book, in concordance to the curriculum of the subjects where the interface and robotic devices can be used.

Considering the physical response and their condition due to the physical structure of the robot, it is necessary to undertake a process of updating the various systems that make up the robotic manipulator, enabling improved physical performance of the robot.

REFERENCES

[1] I. I. O. for Standardization), *8373, Robots and robotic devices.* 2012.
[2] M. T. Torriti, *Manipuladores Robóticos*. Pontificia Universidad Católica de Chile, Departamento de Ingeniería Eléctrica, 2006.
[3] K. D. D. P. R. Escandón, P. Ramón, *Historia y evolución de la robótica Industrial*. Universidad Politecnica Salesiana, 2012.
[4] C. L. K.S. FU, R.C. González, *Robótica: Control, detección, visión e Inteligencia*. McGraw-Hill Espana, 1997.
[5] F. Instruments, *Mentor desktop robot.* 1958.
[6] M. I. M. R. P. Martínez, P. Asmar, *Aprender a investigar: El proyecto de investigación. Instituto Colombiano para el fomento de la Educación Superior*. Instituto Colombiano para el fomento de la Educación Superior, Santa Fe de Bogotó, 2009.
[7] It-Mentor, *Capacitación y guía para desarrollo de software*. 2005.
[8] W. Ronald, *Probabilidad y Estadística para Ingenieros*. Prentice Hall Hispanoamérica SA, 1999.
[9] C. B. R. A. A. Barrientos, L. Penin, *Fundamentos de Róbotica*. McGraw-Hill Espana, 1997.
[10] U. Eco, *Técnicas y Procedimientos de investigación, estudio y escritura.* Gedisa, S.A., 1988.