BRACON: control system for a robotic arm with 6 degrees of freedom for education systems

David Rivas, Marcelo Alvarez, Patricio Velasco, Javier Mamarandi, Jose Luis Carrillo-Medina, Víctor Bautista, Omar Galarza, Patricio Reyes, Mayra Erazo, Miltón Pérez, Mónica Huerta.
Departamento de Eléctrica y Electrónica
Universidad de las Fuerzas Armadas ESPE
Av. General Rumiñahui s/n, Sangolquí, Ecuador P.O.BOX: 171-5-31
{drrivas, rmalvarez,epvelasco, jamamarandi, jlcarrillo, vrbautista, ovgalarza, wpreyes, mjerazo, mfperez3, mhuerta}@espe.edu.ec

Abstract—**This article focuses on the design and development of a control system for a robotic arm designed at the Universidad de las Fuerzas Armadas, Latacunga extension, by using Dynamixel servomotors. The use of Python software, with advantages and features of being a free programming language, provides the project with reliability and ease of communication with a computer arm. The use of these techniques allow to obtain solutions much cheaper than the current ones by using open source software.**

Keywords-component; Robotic Arm; Dynamixel; Python; Servomotor; Free software; RS-48.

I. INTRODUCTION

One of the main problems of education at technical level in developing countries is the high cost of training equipment in the area of automation and robotics for teaching, but the systems are prohibitive and not manipulated. In a work performed at the University Obafemi Awolowo of Nigeria, the authors present instructional Robots oriented low cost system in Africa, thus responding to the need for training in the technical area using concepts "low cost", since in this region had important technological limitations [1], thus responding to the need for training in the technical area using concepts "Low cost" since in this region faced major technological limitations; secondly, there are studies on the development of robots for educational purposes and present lessons learned in the development of a robotic arm "low cost" and compare their capabilities with a commercial robotic arm [2], obtaining good results in design mechanical but having problems with the control system.

The University of the Armed Forces in recent years has been working on the development of arms and later educational environments for industrial environments. During these investigations, the need to develop a basic software to control these devices, which would aim to act as the interface between the operator and actuators are presented.

The BRACON ("BRAzo CONtrolado", arm controlled) application developed in Python platform allows network control actuators that make robotic arm. This application allows the machining features and palletizing based on angle control positioning of each of the engines under the criteria of direct and inverse kinematics.

This program allowed based control of robotic arms 5 and 6 degrees of freedom, giving the possibility to the University of the Armed Forces of the implementation of low-cost equipment for teaching robotics [3].

The article continues in the next section showing the design of the robotic arm in the third section, the objective of the project described in the fourth section of the base software structure is shown to control servomotors with RS-485 communication, the fifth section the results of applying the test cases built on model, finally in the last section the conclusions obtained during the investigation is shown.

II. ROBOTIC ARMS

Industrial robots are mechanical devices programmable multi-function, designed to move material, parts, tools or specialized through variable programmed motions for multitasking devices. An industrial robotic system includes not only industrial robots but also to all devices and / or sensors needed for the robot to perform its tasks, as well as sequencing or monitor communication interfaces.

Robotic systems are generally used to perform unsafe, dangerous and even repetitive operator tasks. They have many different functions, such as material handling, assembly, welding, loading and unloading of a machine or tool, and features such as: painting, spraying, etc. Most robots are configured for operation by teaching technique and repetition.

A robotic arm is a type normally programmable mechanical arm, similar to the functions of a human arm; this may be the total sum of the device or may be part of a more complex robot [4,5]. The parts of these manipulators or arms are interconnected by joints which allow both a rotational movement (such as an articulated robot) and a translational movement or linear movement.

978-1-4799-6466-6/15/$31.00 ©2015 IEEE

Then the actuator movement of the robotic arm, and the programming language to be used for communication between the elements described.

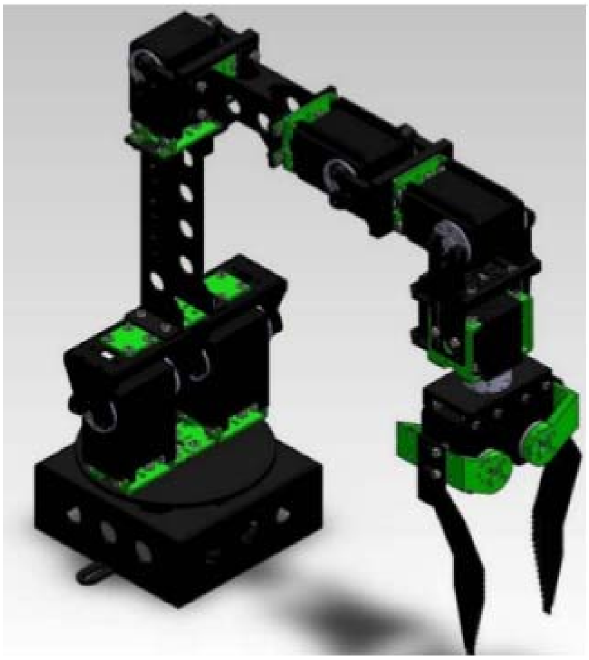


Figure 1. Final design of the structure of the robotic arm.

A. Dynamixel Servomotors

The Dynamixel actuators have been used by universities, laboratories civil and military research [6,7]. Each smart actuator has a built-in microprocessor to facilitate communication bus, position feedback, temperature and load monitoring. The housing of each servo is built specifically for robotics, providing ease of mounting rails and a comprehensive system of support available for the construction of robotic limbs.

The serial communications TTL and RS-485 connections allow bus transfers chain with 1-3 Mbps.

In addition, the MCU (Microcontroller Unit) on board Dynamixel servos have a set of user-customizable features, allowing users to adjust the servos specific functionality for the required applications [8,9].

B. Python Programming Languaje

Python is a multi-paradigm programming language, this means that rather than forcing programmers to adopt a particular style of programming, it permits several styles: object-oriented programming, functional programming and imperative programming. Other paradigms are supported using extensions.

In recent years the Python programming language has become very popular due to various reasons like: The number of libraries that contains data types and functions built into the language itself, which help to perform many common tasks without having you program them from scratch.

In recent years the Python programming language has become very popular due to various reasons like: The number of libraries that contains data types and functions built into the language itself, which help to perform many common tasks without having you program them from scratch.

The simplicity and speed with which programs are created. A Python program can have 3-5 lines of code less than the equivalent in Java or C. The number of platforms that can develop, such as Unix, Windows, OS / 2, Mac and others. Moreover, Python is free, even for business purposes [10,11].

Currently, Python is being increasingly used in this type of robotic arms for their versatility and array of tools, available in open source environment, which allows from a base and improving software continue appending operation with input whether individual or collective [12,13].

III. OBJETIVE OF THE RESEARCH

In the field of robotics may notice great progress over time, see how robots perform tasks that humans have become very difficult to make and need these to enter hazardous or inaccessible environments in the medical field need for robotic prostheses for disabled patients [14]; this type of research and development requires great efforts and financial investments, is for that reason that software have become owners and difficult to acquire because the costs are very high for educational institutions, because of this it was decided to investigate projects based on free software which reduces costs and gives the opening to access these programs [15,16].

Therefore, the present research, following that ideology, aims to develop the necessary libraries using the Units (the programming language Python platform) for a network servo of a robotic arm to perform the functions of machining and palletizing be the primary objective project reduce software acquisition costs.

To achieve this goal, will describe how to perform the robotic arm assembly and communication protocols were used.

A. ***Assembling the Robot Arm***

The characteristics possessed by Dynamixel servo motors makes them ideal for the development of various robotic applications. Control of these systems through a free software application will develop projects where knowledge of their control more easily understood and does not require purchasing licenses for future research. Additionally, these actuators allow to have an open system and is coupled to the same requirements.

Having the actuators, it is only required that the structure as the robot arm. For this it is essential that the pieces which form the union and support for all the robotic system must be strong enough to support the weight and force of the actuators, but also should be light so that the total weight of the robot arm is optimal and not waste power trying to move its own weight.

The parts and joints were designed at the University of the Armed Forces ESPE (Latacunga) and manufactured in the city of Quito-Ecuador. Figure 1 shows the final design of the robotic arm with the parts manufactured by the University.

B. Communication USB2Dynamixel

The USB2Dynamixel interface is a device manufactured by the same company where the robotic arm kits used to operate Dynamixel devices are purchased.

Figure 2. USB2Dynamixel interface and connection ports.

USB2Dynamixel is connected to the PC USB port and has two 3P and 4P connectors that are installed so that multiple Dynamixel devices can be connected to these ports.

As the robotic arm consists of devices in this case servo motors Dynamixel DX / RX serial communication protocol is established that the RS-485 port connection is the 4P connector shown in Figure 2 is needed.

The USB2Dynamixel interface must be configured for this type of communication, for which the only thing to be done is to select the mode with the selector switch located on the east.

Communication is established between the computer and the network of actuators is the RS-485 (Standard EIA-485) is an improvement over RS-422 as it increases the number of devices that can be connected (10 to 32) and defines the characteristics necessary to ensure the appropriate voltage values when it is fully loaded. With this capability, you can create networks connected to a single RS-485 devices.

This capability, and high immunity to noise, make this type of serial transmission is the choice of many industrial applications requiring distributed network connected to a PC or other controller for data collection, HMI, or other operations devices. RS-485 is a set covering RS-422, so that all devices that communicate using RS-422 can be controlled by RS-485. Hardware RS-485 can be used in serial communications distances up to 4000 feet of cable. The connector pin arrangement is the same as the RS-232 communication [17].

C. Communication and Network Setup Servo

Once have the model of the robotic arm assembly and subject to a strong and stable enough base, it is necessary to power the servo network, these operate with a voltage between 14.8V and 11,1V to a current consumption 1.7 A, 5.2 A and 6.3 A for the MX-28, MX-64 and MX-106 models respectively.

The connection mode network servo to a PC using the USB2Dynamixel interface is shown in Figure 3.

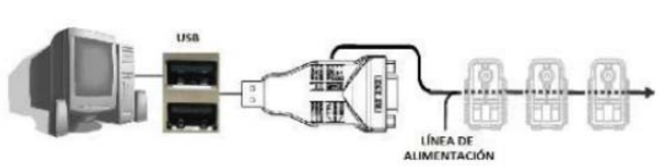


Figure 3. Connecting a PC to the network using the servo interface USB2Dynamixel.

IV. PROGRAMMING INTERFACE AND CONTROL SYSTEM

In developing the control system based on free software starts by making a diagram of basic and general flow to establish a communication between the system, in this case a robot arm and the controller is a desktop computer, this process is it can be seen in Figure 4.

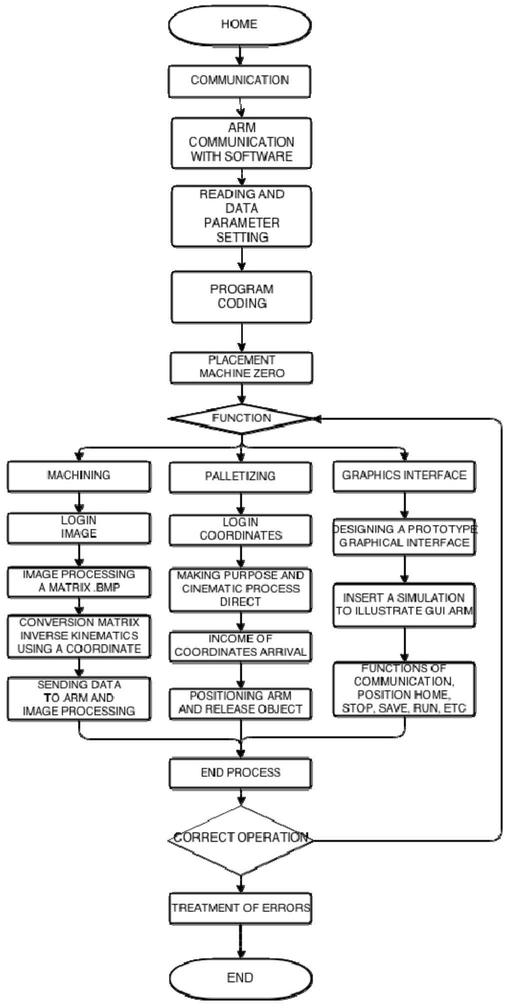


Figure 4. General flowchart programming.

As in any programming language starts importing libraries if they are needed, declaring variables, calling functions including; Because this program does not follow any method of control is said to be a heuristic programming as the robot arm follow the instructions line by line program.

The whole process was conducted in two parts, one of which was the development of the program responsible for network communication Servo, linearize the total number of data from a motor revolution to a range of values can be interpreted by a user and that it can perform multiple calculations and processes; also where you can read the data delivered by the actuator as the ID (identifier of each servomotor), value temperature, torque, position, speed and other parameters depending on the type of servomotor Dynamixel being used can be read.

So in this first part of reading and writing functions, where you can directly manipulate and configure the network one by one servo.

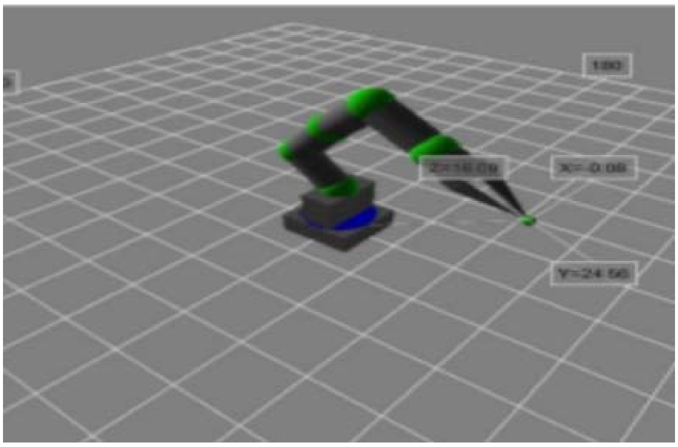
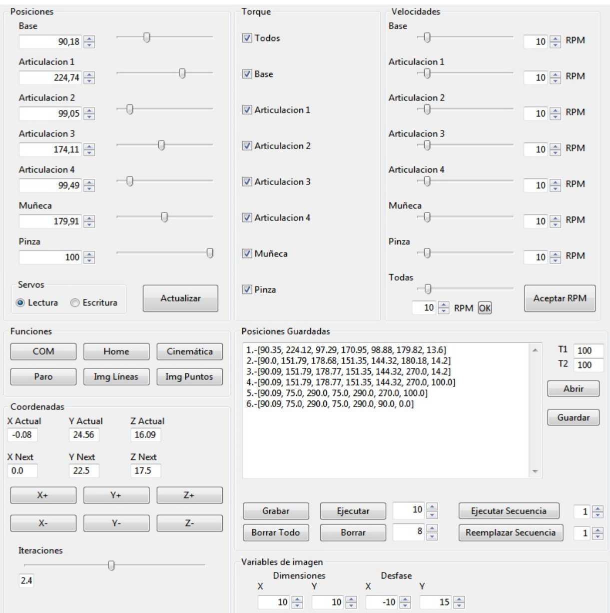


Figure 5. GUI animated robot arm.



(a)

Simulacion	Coordenadas	Grados	

Servos Dynamixel Caracteristicas

	ID	Temperatura(°C)	Voltaje(V)	Carga(%)
Servo	1	30	12.0	1
Servo	2	33	11.8	2
Servo	3	35	11.7	3
Servo	4	36	11.5	8
Servo	5	38	11.4	7
Servo	6	35	11.5	4
Servo	7	34	11.3	0
Servo	8	36	11.1	7
Servo	9	33	11.2	6

b)

Figure 6. (a)Sliders. (b) Robotic Arm in communication with GUI.

The second part of the development is scheduled in the Python extension called Vidle Python, where you have the option of creating windows, buttons, animated figures, among others. The \need for a graphical interface is extremely important because it requires a user-friendly system for users who do not have programming skills also required that systems are intuitive, then this interface anyone familiar with a computer can easily handle the robotic arm using sliders, buttons, text boxes, etc., and observe changes in both simulation shown in the interface (see Figure 5, 6) and in the same physical robotic system [18,19].

Figure 7. Servo Control System.

In order to fit the hardware and software, it starts importing all functions programmed to schedule file GUI, as if it were a library. And then sends signals to the actuators are placed individually to zero machine, sending its proper functioning and positioning then continue machining processes or palletized as required from the operator. The coupled system and running can be seen in Figure 7.

V. RESULTS

The main results obtained are shown, the construction of the robotic arm using software described:

- Servo motors with serial communication have great advantages to owning monitoring and internal control.
- With programming on both functions and graphical interface was required the study of other fields such as inverse kinematics, mathematical vector, linearization ranges, image processing and compression mechanical robots to relate all as one system and work together.
- Functions performed in each servomotor, of: movement, setting torque, speed, reading and recording of data, depending on all these parameters could read the servomotor.
- To observe the behavior of the robot before handling it required a simulation function, because this can be avoided several accidents because if a user enters values moving at high speed while executing the risk of strokes you run the people close to the arm.
- It worked in what is the parameterization of coordinates because the robotic arm to manipulate only the top values for each servomotor is entered and manipulating their degrees of freedom one by one until you reach the desired position in space; but it is not optimum, thus mobilizing the arm to a desired position it has to work together as a system synchronization and using mathematical reasoning of the vector sum is obtained that the arm reaches a position only by entering the values on the Cartesian axes X, Y and Z.
- To function palletizing he also worked in polar coordinates, where the robotic arm will send data to the coordinates in radius values, swivel and height, expected results and taking objects that position was obtained and placed elsewhere also designated by the user.
- To function machining had to conduct the study and investigation of inverse kinematics, to understand this topic the fundamental idea was to read the image was draw and transform it to an array of coordinates to the robotic arm scroll through these items and trace figure charged to the program. For this is what was done in Python image processing and scaling ratio of the image matrix obtained ones and zeros in a matrix of coordinates where the robotic arm would travel in space.
- The tests indicate favorable and satisfactory results as expected, each of the functions programmed to fulfill their duties fully, one of the main roles in Programming for travel, this is a technique in which the user making the programming has physical contact with the robot arm and actually gains control and takes the arm of the robot through the desired workspace within the positions, that is widely used in industry to make a robot work routine.

Therefore, can conclude that thanks to the built model has achieved the initial objectives, especially in the significant improvement in the control of the robotic arms inexpensive servomotor Dynamixel that by performing correct positions preprogrammed decrease significantly their mistakes.

VI. CONCLUSIONS

From research, have obtained the following conclusions:

- It was possible to make palletizing and machining of a robotic arm with 6 degrees of freedom, using a graphical interface designed in free software as it is in this case Python.
- The RS-485 protocol communication allows to communicate the nine devices (actuators) that make up the serial robotic arm ensuring the voltages required for the normal functioning of the robotic system when they work the most.
- Free software offers possibilities to tailor programs to the specific needs of the project. It also offers the possibility of improving the software code to provide greater functionality to the robotic arm.
- The software used Python has great advantages and multiple extensions that allowed the development of the GUI in this project, but also gives its applicability in various fields of engineering.
- Python allows the management of Dynamixel servomotors, and libraries created by means of a communication through the USB port and USB2Dynamixel converter.

It is intended to work later in the evolution of the robotic arm control based Fuzzy Logic systems [20,21] applied in engineering in which alternatives are contemplated to solve more complex and difficult to solve problems.

ACKNOWLEDGMENTS

Thanks to Drs José Luis Rojo Alvarez and Luis Enrique Sanchez Crespo, Prometheus Researchers in the Department of Electrical and Electronics Engineering (funded by the Ministry of Higher Education, Science, Technology and Innovation of the Republic of Ecuador) for their kind support.

REFERENCES

[1] Adebola, S. O., Odejobi, O. A., & Koya, O. A. (2013, September). Design and implementation of a locally-sourced robotic arm. In *AFRICON, 2013* (pp. 1-4). IEEE.

[2] Krasnansky, P., Toth, F., Huertas, V. V., & Rohal'-Ilkiv, B. (2013, June). Basic laboratory experiments with an educational robotic arm. In *Process Control (PC), 2013 International Conference on* (pp. 510-515). IEEE.

[3] Augusta R. V. M., Valencia J. H. F., “Construction of prototype robotic arm controlled by a computer”, Escuela Politécnica Nacional, Quito, 2012. unpublished.

[4] Jamshidi M., Anderson J.N., "Parallel and multi-arm robots. Robotics and Autonomous Systems", 1994 unpublished..

[5] Jamshidi, M., "Multi-arm robots. Robotics and Autonomous Systems", 1989. 5(4): p. 305-306.

[6] Gomez, J. V., & Sandnes, F. E. (2012). RoboGuideDog: Guiding blind users through physical environments with laser range scanners. *Procedia Computer Science*, *14*, 218-225.

[7] Tanaka, F., & Takahashi, T. (2012, March). A tricycle-style teleoperational interface that remotely controls a robot for classroom children. In *Human-Robot Interaction (HRI), 2012 7th ACM/IEEE International Conference on* (pp. 255-256). IEEE.

[8] Herías F. A. C., "Servomotores", 20 September 2007. \url{http://www.aurova.ua.es:8080/proyectos/dpi2005/docs/publicaciones/pub09-ServoMotores/servos.pdf}

[9] Robotics, "Trossen Robotics Dynamixel Guide", 2006. \url{http://www.trossenrobotics.com/p/MX-64R-dynamixel-robot-actuator.aspx.}

[10] Zhenli L., "A brief survey of commercial robotic arms for research on manipulation. in Robotics and Applications (ISRA)", 2012 IEEE Symposium on. 2012.

[11] Bohren, J., Rusu, R. B., Jones, E. G., Marder-Eppstein, E., Pantofaru, C., Wise, M., ... & Holzer, S. (2011, May). Towards autonomous robotic butlers: Lessons learned with the pr2. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on* (pp. 5568-5575). IEEE.

[12] Basile, F., Caccavale, F., Chiacchio, P., Coppola, J., & Curatella, C. (2012). Task-oriented motion planning for multi-arm robotic systems. *Robotics and Computer-Integrated Manufacturing*, *28*(5), 569-582.

[13] Ruiz-del-Solar, J., Palma-Amestoy, R., Marchant, R., Parra-Tsunekawa, I., & Zegers, P. (2009). Learning to fall: Designing low damage fall sequences for humanoid soccer robots. *Robotics and Autonomous Systems*, *57*(8), 796-807.

[14] Ben-Tzvi, P., Charifa, S., & Shick, M. (2010, October). Extraction of 3D images using pitch-actuated 2D laser range finder for robotic vision. In *Robotic and Sensors Environments (ROSE), 2010 IEEE International Workshop on* (pp. 1-6). IEEE.

[15] OSHA, "Industrial robots and robot system safety", 1 Enero 1999. \url{https://www.osha.gov/dts/osta/otm/otm_ivotm_iv_4.html.}

[16] Qassem, M. A., Abuhadrous, I., & Elaydi, H. (2010, March). Modeling and Simulation of 5 DOF educational robot arm. In *Advanced Computer Control (ICACC), 2010 2nd International Conference on* (Vol. 5, pp. 569-574). IEEE.

[17] N. Instruments, "Comunicación Serial: Conceptos Generales", 6 Junio 2006. http://digital.ni.com/public.nsf/allkb/039001258CEF8FB686256E0F005888D1.

[18] C. Tecnologías, \emph{Main features of the Python language}, September 1 2011. http://www.cuatrorios.org/index.php?option=com$_$content$\&$\\view=article$\&$id=161:principales-caracteristicas-del-lenguaje-python$\&$catid=39:blogsfeeds.

[19] Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T., Leibs, J., ... & Ng, A. Y. (2009, May). ROS: an open-source Robot Operating System. In *ICRA workshop on open source software* (Vol. 3, No. 3.2, p. 5).

[20] Hacioglu, Y., Arslan, Y. Z., & Yagiz, N. (2011). MIMO fuzzy sliding mode controlled dual arm robot in load transportation. *Journal of the Franklin Institute*, *348*(8), 1886-1902.

[21] Li, Y. J., Wang, G. C., Zhang, J., & Jia, Z. Y. (2012). Dynamic characteristics of piezoelectric six-dimensional heavy force/moment sensor for large-load robotic manipulator. *Measurement*, *45*(5), 1114-1125.