JÖKULL JOURNAL
JOKULL ISSN: 0449-0576
Vol ---, No. --- ;

DEVELOPMENT OF AN ELECTRONIC CONTROLLER APPLIED TO A ROBOTIZED MANIPULATOR THROUGH THE SPI PROTOCOL

CLAUDIO URREA

Grupo de Automática, Departamento de Ingeniería Eléctrica, Universidad de Santiago de Chile, Avenida Ecuador 3519, Estación Central
Santiago, cl 9170124, Chile
claudio.urrea@usach.cl

JOHN KERN

Grupo de Automática, Departamento de Ingeniería Eléctrica, Universidad de Santiago de Chile, Avenida Ecuador 3519, Estación Central
Santiago, cl 9170124, Chile
john.kern@usach.cl

Abstract

Purpose - In this paper the design and implementation of an embedded control system, using DSCs (Digital Signal Controller) devices of dsPIC type, applied on a four-DOF (Degrees Of Freedom) model OWI-535 robotic manipulator is presented.

Design/Methodology/Approach - The mathematical model of manipulator is elaborated obtaining joint trajectories to achieve the tracking of a test Cartesian trajectory. A control structure based on a Master/Slave configuration, between two dsPIC30F4013 through SPI (Serial Peripheral Interface) protocol is developed. In such a system, algorithms are implemented: PID (Proportional Integral Derivative) with anti-windup and, generation PWM (Pulse-Width Modulation) and data acquisition, in the DSCs master and slave, respectively.

Findings - Taking advantage of the high transmission speed of SPI communications and of the quick response of the dsPIC30F4013 DSC, which makes up the embedded system, it is possible to reduce the error correction time of the robot's joint position and to eliminate the vibrations of the actuators, achieving settlement of the desired position quickly and smoothly. This is achieved through two work zones to produce PWM signals: in the first one PWM signals are generated with high duty cycle for a calculated time, and in the second one PWM signals are produced in an exponentially decreasing duty cycle to achieve the desired position.

Practical implications - The embedded control system that is developed makes it possible to increase the response speed and accuracy, and to provide autonomy to a four-DOF model OEI-535 robotic manipulator, ignoring, for example, as a control element, the use of a

general purpose computer. This system can be extended, with an appropriate power interface, to the control of manipulator robots with higher load capacities.
Originality/value - This paper uses dsPIC30F4013 Digital Signal controllers communicated via the SPI protocol, which have not been widely explored in the control of robot manipulators, getting increase speed of joint position error correction, eliminate vibration of the actuators when approach their detention, and provide autonomy a four-DOF robotic manipulator. The proposed methodology, greatly simplifies the development of an autonomous control system based on DSCs controllers that communicate with each other.

Keywords Controllers, embedded systems, dsPIC, SPI, robotics, programming

Paper type Research paper

1 Introduction

Embedded systems, thanks to the currently available large variety of hardware and software development tools, have become a fundamental part in a vast range of industrial activities like automobiles, avionics, telecommunications, aerospace, automation and robotics, among many others (Bareno, 2011). As an example of this, it can be estimated that 98% of the 32-bit processors made is meant for the embedded systems market (Jackson and Caspi, 2005). From an economic point of view, according to a BBC Research report massive investments in research are being made around the world, forecasting that the global market for embedded systems will go from US$92 billion in 2008 to approximately US$112.5 billion by the end of 2013 (BCC Reserch, 2009).

Embedded systems are heterogeneous systems whose hardware and software is designed to solve a specific problem within a larger system in which they are integrated. Those systems are mainly composed of programmable or configurable chips such as microprocessors, microcontrollers, DSPs (Digital Signal Processors), DSCs, FPGAs (Field-Programmable Gate Array), and others (Andrade et al., 2011; Basmage Pinheiro et al., 2011; Borges et al., 2011; Galeano, 2009; Melgarejo et al., 2011; Moreno, 2012; Valvano, 2004, 2009).

In the field of robotics, and according to the specifications of each robot, the control systems used in robotic manipulators are implemented by means of embedded systems, as is the case of the IRC5 Compact, a controller developed by ABB enterprises, having trajectory control abilities, friendly programming unit, flexible RAPID language, and vast communications possibilities as major features, all this packed in a small size, as shown in Figure 1.

Figure 1 IRC5 Compact: controller developed by ABB enterprises

The present work is organized as follows: in section 2 we introduce the main features and the kinematic and dynamic models of the manipulator robot OWI-535. Section 3 refers to the Cartesian trajectory test applied to the robot. Section 4 explains the design of the embedded control system, including the hardware configuration. Section 5 describes the control algorithm used and the of SPI communications form that was selected. Finally, results and conclusions are presented in Sections 6 and 7, respectively.

2 Four DOF Robotic Manipulator

The robotic manipulator considered in this study includes 4 revolving joints, plus the actuator in the gripper. For analysis, design and implementation purposes, we include the first three joints and the gripper's actuator. Figure 2 shows a picture of the model OWI-535 robot, along with its schematic representation.

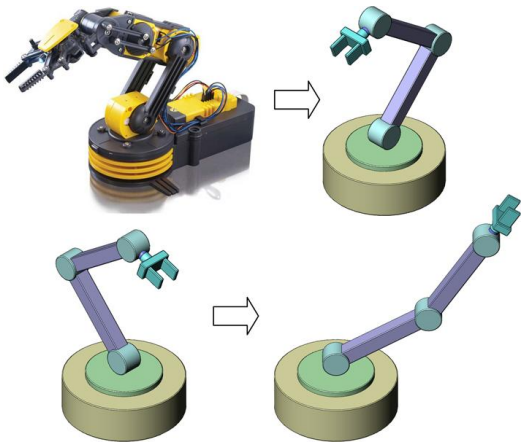


Figure 2 Robot model OWI-535 and its schematic representation

In Figure 3 we can see the robot's schematic diagram, considering for its kinematic and dynamic modeling, the arrangement of the systems of coordinate axes and the centroids, where q_1, q_2, q_3 and l_1, l_2, l_3, represent the generalized coordinates and the length of the links: first, second and third, respectively. l_{c1}, l_{c2} and l_{c3} express the length from the origin to the centroid of the links: first, second and third, respectively.

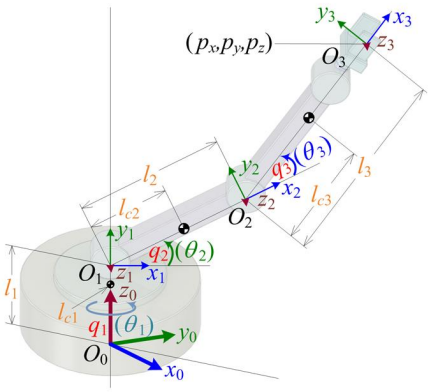


Figure 3 Schematic diagram of the model OWI-535 robot considering the arrangement of the systems of coordinate axes and the centroids

Direct and inverse kinematic models are obtained by applying Denavit-Hartenberg and geometric methods, respectively (Angeles, 2006; Craig, 2004; Spong et al., 2005). Results are presented in Eqns. (1) to (4):

$$\mathbf{T}=\begin{bmatrix} c_{23}\cdot c_1 & -s_{23}\cdot c_1 & s_1 & (l_2 c_2+l_3 c_{23})c_1 \\ c_{23}\cdot s_1 & \dfrac{c_{123}-c_{-123}}{2} & -c_1 & (l_2 c_2+l_3 c_{23})s_1 \\ s_{23} & c_{23} & 0 & l_1+l_2 s_2+l_3 s_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1}$$

$$\theta_1=\arctan\left(p_y/p_x\right) \tag{2}$$

$$\theta_3=\arctan\left(\frac{\pm\sqrt{1-c_3^2}}{c_3}\right);\ c_3=\frac{p_x^2+p_y^2+(p_z-l_1)^2-l_2^2-l_3^2}{2l_2l_3} \tag{3}$$

$$\theta_2=\arctan\left(\frac{p_z-l_1}{\pm\sqrt{p_x{}^2+p_y{}^2}}\right)-\arctan\left(\frac{l_3 s_3}{l_2+l_3 c_3}\right) \tag{4}$$

where: $s_1 = \sin\theta_1$, $s_2 = \sin\theta_2$, $c_1 = \cos\theta_1$, $c_2 = \cos\theta_2$, $s_{23} = \sin(\theta_2+\theta_3)$, $c_{23} = \cos(\theta_2+\theta_3)$, $c_{123} = \cos(\theta_1+\theta_2+\theta_3)$ y $c_{-123} = \cos(-\theta_1+\theta_2+\theta_3)$.

The dynamic model for the robotic manipulator is obtained from the Lagrange-Euler formulation according to expression (5) (Angeles, 2006; Iñigo and Vidal, 2004; Ollero, 2006; Siciliano and Khatib, 2008):

$$\boldsymbol{\tau} = \mathbf{M}(q)\ddot{\mathbf{q}} + \mathbf{C}(q,\dot{q}) + \mathbf{G}(q) + \mathbf{F}(\dot{q}) \tag{5}$$

where $\boldsymbol{\tau}$ represents the vector of generalized forces (of $n\times1$dimension), $\mathbf{M}$ is the inertia matrix (of $n\times n$ dimension), $\mathbf{C}$ is the vector of centrifugal and Coriolis forces (of $n\times1$dimension), q represent the components of the joint position vector, $\dot{q}$ the components of the joint speed vector, $\mathbf{G}$ is the vector of gravitational forces (of $n\times1$dimension), $\ddot{\mathbf{q}}$ is the vector of joint acceleration (of $n\times1$dimension), $\mathbf{F}$ is the vector of friction forces (of $n\times1$dimension), and n is the number of degrees of freedom of the robot. Therefore, using Eqn. (5), the dynamic model can be expressed through Eqns. (6) to (21):

$$\mathbf{M} = \left[M_{11}\ M_{12}\ M_{13}\ ;\ M_{21}\ M_{22}\ M_{23}\ ;\ M_{31}\ M_{32}\ M_{33}\right] \tag{6}$$

$$M_{11} = \tfrac{1}{2} m_2 l_{c2}^2 \left(\mathrm{c}_{2\cdot2}+1\right) + m_3\left(l_2\mathrm{c}_2 + l_{c3}\mathrm{c}_{23}\right)^2 + I_{1zz} + I_{2zz} + I_{3zz} \tag{7}$$

$$M_{12} = M_{21} = I_{2zz} + I_{3zz} \tag{8}$$

$$M_{13} = M_{31} = I_{3zz} \tag{9}$$

$$M_{22} = m_2 l_{c2}^2 + m_3\left(l_2^2 + 2l_2 l_{c3} + l_{c3}^2\mathrm{c}_3\right) + I_{2zz} + I_{3zz} \tag{10}$$

$$M_{23} = M_{32} = I_{3zz} + l_{c3} m_3\left(l_{c3} + l_2\mathrm{c}_3\right) \tag{11}$$

$$M_{33} = m_3 l_{c3}^2 + I_{3zz} \tag{12}$$

$$\mathbf{C} = \left[C_{11} \quad C_{21} \quad C_{31}\right]^{\mathrm{T}} \tag{13}$$

$$C_{11} = -\left(l_{c2}^2 m_2 \mathrm{s}_{2\cdot2} + 2m_3\left(l_2\mathrm{c}_2 + l_{c3}\mathrm{c}_{23}\right)\left(l_2\mathrm{s}_2 + l_{c3}\mathrm{s}_{23}\right)\right)\dot{\theta}_1\dot{\theta}_2 - 2l_{c3} m_3 \mathrm{s}_{23}\left(l_2\mathrm{c}_2 + l_{c3}\mathrm{c}_{23}\right)\dot{\theta}_1\dot{\theta}_3 \tag{14}$$

$$C_{21} = \left(\tfrac{1}{2} l_{c2}^2 m_2 \mathrm{s}_{2\cdot2} + m_3\left(l_2\mathrm{s}_2 + l_{c3}\mathrm{s}_{23}\right)\left(l_2\mathrm{c}_2 + l_{c3}\mathrm{c}_{23}\right)\right)\dot{\theta}_1^2 - l_2 l_{c3} m_3 \mathrm{s}_3 \dot{\theta}_3^{\,2} - 2l_2 l_{c3} m_3 \mathrm{s}_3 \dot{\theta}_2\dot{\theta}_3 \tag{15}$$

$$C_{31} = l_{c3} m_3 \mathrm{s}_{23}\left(l_2\mathrm{c}_2 + l_{c3}\mathrm{c}_{23}\right)\dot{\theta}_1^2 + l_2 l_{c3} m_3 \mathrm{s}_3 \dot{\theta}_2^2 \tag{16}$$

$$\mathbf{G} = \left[G_{11} \quad G_{21} \quad G_{31}\right]^{\mathrm{T}} \tag{17}$$

$$G_{11} = 0 \tag{18}$$

$$G_{21} = \left(l_{c2} m_2 \mathrm{c}_2 + m_3\left(l_2 \mathrm{c}_2 + l_{c3} \mathrm{c}_{23}\right)\right) g_z \tag{19}$$

$$G_{31} = l_{c3} m_3 \mathrm{c}_{23} g_z \tag{20}$$

$$\mathbf{F} = \begin{bmatrix} F_{11} & F_{21} & F_{31} \end{bmatrix}^{\mathrm{T}} \tag{21}$$

where: $c_{2\cdot2} = \cos(2\theta_2)$; $s_{2\cdot2} = \sin(2\theta_2)$; m_1, m_2, and m_3, represent the masses of first, second and third links, respectively; I_{1zz}, I_{2zz} and I_{3zz} represent the momentum of inertia of the first, second and third link with respect to the first z axis of their joint, respectively.

3 Test Cartesian Trajectory

After getting the mathematical model for the manipulator, we established a Cartesian test trajectory to be followed by the robot's end effector, composed of 7 curve segments in the xyz space, as shown in Figure 4.

At the moment of starting the movement, the manipulator's end is placed in the rest position P_0: here the gripper must lift a light object, a die in this case, moving it along the defined segments, and then putting it again in its original position.

After that, by means of inverse kinematics and using the selected Cartesian trajectory, we get the joint trajectories that must be entered as references in the embedded control system.

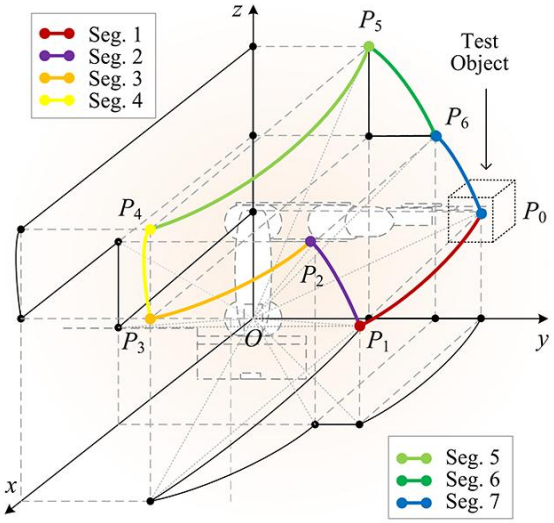


Figure 4 Scheme of the Cartesian test trajectory in the *xyz* space

4 Embedded Control System

The embedded control system is composed mainly by two DSCs dsPIC30F4013 arranged in a Master/Slave configuration, communicated through the SPI protocol, as pointed in Figure 5.

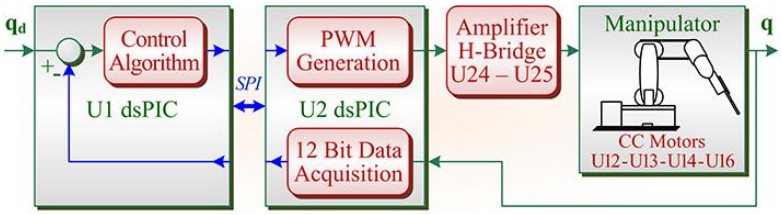


Figure 5 Block diagram of the embedded control system applied in the robotic manipulator

In Figure 6 we present a block diagram showing the general distribution of peripherals in the robotic system. In this system, the control algorithm resides in the Master U1, which communicates the correction actions of position error into the Slave U2. The latter addresses the task of providing the PWM (Pulse Width Modulation) signals to the electric actuators, corresponding to DC motors (U12, U13, U14 and U16), through H type amplifier bridges (U24 and U25) (Abdalrahman et al., 2013; Ozturk et al., 2013). The device U19, also governed by U2, represents an LCD display of 4 lines by 20 characters, intended to show the joint's position and error data.

In order to obtain the position feedback signals of the joints, we attach potentiometers in the center of each rotor, conditioning the voltage signals and making the proper calibrations to reach the adequate voltage levels required by the internal ADC (Analog Digital Converter) in the dsPIC U2. This converter is configured with a 12 bits resolution, allowing to acquire position data in a range from 0 to 4095 (decimal base).

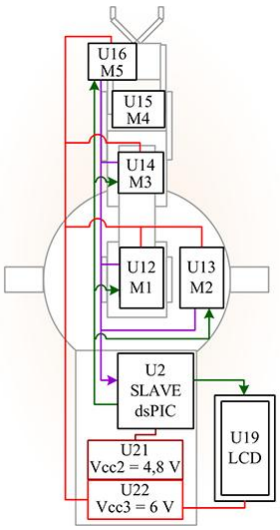


Figure 6 Diagram showing the distribution of peripherals in the robot

By means of a coding algorithm, executed in U2, feedback data are packed and informed in SPI to the Master using properly arranged frames. In Table I we indicate the hardware elements used in the embedded control system.

Table I Description of hardware in the embedded control system

Peripheral number	Description of Hardware
U1	Master (Microcontroller dsPIC30F4013)
U2	Slave (Microcontroller dsPIC30F4013)
U12	DC Motor 1 - Hip
U13	DC Motor 2 - Shoulder
U14	DC Motor 3 – Elbow
U15	DC Motor 4 - Wrist (fixed)
U16	DC Motor 5 - Gripper
U17	Infrared Sensor QRD1114
U18	Light - Gripper
U19	LCD 4·20 Characters (Hitachi 44780)
U20	Vcc1 = 4,8 V (4 AA 1.2 V Batteries)
U21	Vcc2 = 4,8 V (4 AA 1.2 V batteries)
U22	Vcc3 = 6 V (4 C 1.2 V Batteries)
U23	Vcc4 = 18.6 V (8 1.2 V Batt.+ 1 9 V Batt.)
U24	IC H Bridge (SN754410)
U25	IC H Bridge (SN754410)

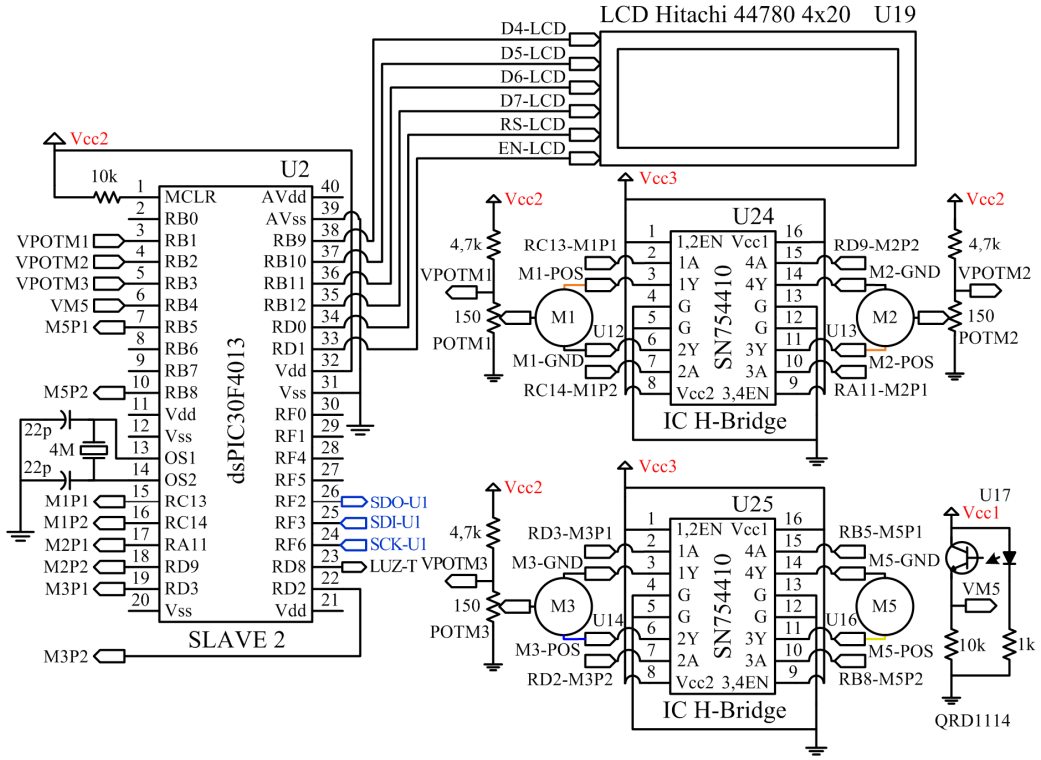


Figure 7 Schematic diagram of the disposition of the Slave U2 with its associated peripherals

In Figure 7 we can see the schematic diagram of the arrangement of Slave U2, with its associated peripherals.

Table II shows a description of the data bus of Slave U2 in terms of its commanded peripherals.

Table II Description of the data bus of U2 with its peripherals

U2 Pins	Description - Comment
RC13 - M1P1	Signal to U24: rotation to the right, U12
RC14 - M1P2	Signal to U24: rotation to the left, U12
RA11 - M2P1	Signal to U24: rotation to the right, U13
RD9 - M2P1	Signal to U24: rotation to the left, U13
RD2 - M3P2	Signal to U25: rotation to the right, U14
RD3 - M3P1	Signal to U25: rotation to the left, U14
RB8 - M5P2	Signal to U25: rotation to the right, U16
RB5 - M5P1	Signal to U25: rotation to the left, U16
RB1 - VPOTM1	Feedback voltage from rotor in U12
RB2 - VPOTM2	Feedback voltage from rotor in U13
RB3 - VPOTM3	Feedback voltage from rotor in U14
RB4 - VM5	Feedback voltage from rotor in U17
RB9 - LCDD4	Data Bus to U19: 4bits
RB10 - LCDD5	Data Bus to U19: 4bits
RB11 - LCDD6	Data Bus to U19: 4bits
RB12 - LCDD7	Data Bus to U19: 4bits
RD0 - LCDRS	Register select signal to U19
RD1 - LCDEN	Enabling signal to U19
RD8 - LUZ-T	Light of the robotic arm, for handling objects

In order to reduce the position error's correction time, we designated work zones A and B for the generation of PWM signals. In zone A, 5 V DC signals are generated during a time period t_A, and in zone B the PWM signals are produced during a time t_B to get the settlement of the desired position in a smooth way, as pointed in Figure 8.

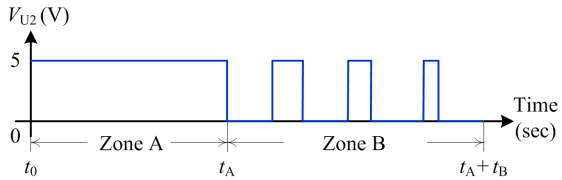
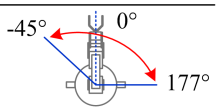
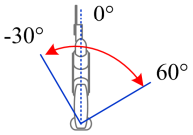
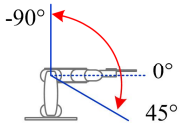
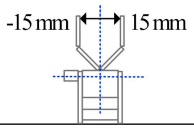


Figure 8 Work zones involved in the generation of PWM signals

The joints' spinning ranges considered for the movements of this manipulator, measured in degrees, are shown in Table III.

Table III Joints' movement range

Joint	Operating	Limits	Scheme
Hip	222°	-45° and 177°	
Shoulder	90°	-30° and 60°	
Elbow	135°	-90° and 45°	
Gripper	30 mm	-15 mm and 15 mm	

5 Control Algorithm and SPI Communications

The control algorithm for this robotic system is programmed in the Master U1, and it corresponds to a PID with antiwindup for each joint (Ang et al., 2005; Astrom and Hagglund, 2006; Astrom and Murray, 2009; Franklin et al., 1998). The tuning of controller parameters is made through successive empirical adjustments, obtaining a response with a minimum error and an adequate compromise between speed and accuracy in the correction of the error. In Figure 9 we show the block diagram of the PID controller with antiwindup.

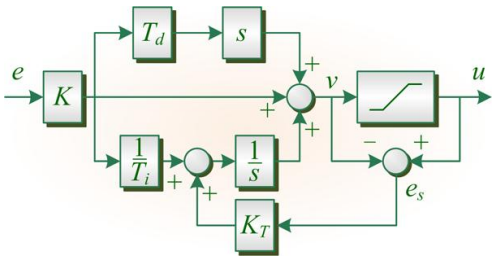


Figure 9 Block diagram of the PID controller with antiwindup

In this scheme, the output u of the controller in the discrete time k is expressed by the algebraic equation of expression (22):

$$u(k)=u(k-1)+K\left[\left(1+T/T_i+T_d/T\right)e(k)\right]+\ldots$$
$$K\left[-\left(1+2T_d/T\right)e(k-1)+T_d/T\,e(k-2)\right]+K_T T e_s(k) \qquad (22)$$

where K represents the proportional gain, T_i is the integral time constant, T_d is the derivative time constant, T is the sampling period, K_T is the integral correction constant, e

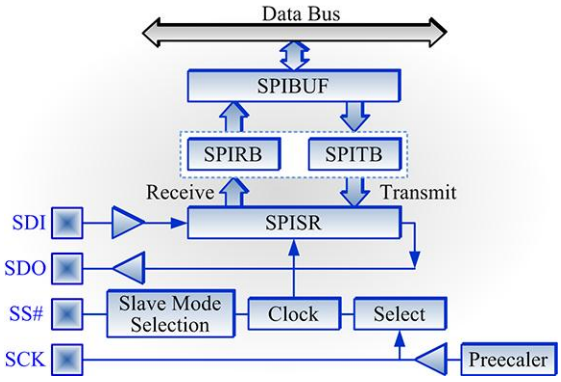
is the error between position reference and position feedback, and e_s is the error between the saturated and non-saturated outputs of the controller. In Table IV we show the values of the constants selected for the PID controller with antiwindup.

Table IV Gains considered in the controllers

Joint	K	T_i	T_d	K_T
Hip	0.4056	0.10	0.025	4.056
Shoulder	0.4045	0.12	0.030	3.371
Elbow	0.4911	0.12	0.030	4.092
Gripper	0.4911	0.12	0.030	4.092

In order to provide the control actions to the Slave U2 we used the SPI protocol. This kind of communications corresponds to a synchronous standard of serial data currently used to establish communications between diverse devices such as microcontrollers, DSCs, EEPROM memories, ADC and DAC converters, DSPs, etc. (Szadkowski, 2014). The SPI protocol allows enabling *full duplex* links, that is, we can send and receive information simultaneously, raising the data transference rate. This protocol has 3 operation modes for transmitting data in 8 and 16 bits formats: slave mode, master mode, and finally, frame mode (Angulo, 2006; Angulo et al., 2006; Huddleston, 2007). Figure 10 show the block diagram for the SPI module used to connect the U1 and U2 dsPICs.

Figure 10 Block diagram of the SPI module employed in the dsPICs



According to the diagram (Figure 10), we usually use the following lines for establish links: SDI (Serial Data Input), SDO (Serial Data Output), SCK (Synchronization Clock) and SS# (Slave Selection by low level); however, after carrying out different communication tests, we decided to use signaling by the byte address of the U2 slave circuit, avoiding having to implement the SS# line, in this way simplifying the bus lines, as shown in Figure 11.

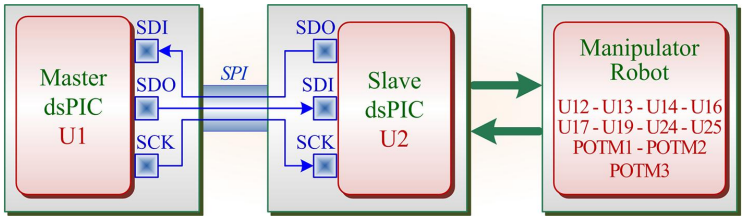


Figure 11 Block diagram showing the format of SPI communications applied in the embedded system

Table V give a description of the control bus for the Master U1 as a function of the Slave U2.

Table V Description of Control Bus of Master U1 used with U2

U1 Pins	Description – Comments
RF2 - SDI	Input Data from the Slave U2
RF3 - SDO	Output Data to the Slave U2
RF6 - SCK	Clock Signal for synchronization with U2

The information arranged in the SPI bus corresponds to frames containing data from the joint position sensors and PWM signal data for the actuators. The size of the frames, both for reception and emission with respect to the Master, is set in 10 bytes, from which 2 bytes correspond to the frame header and the remaining 8 bytes represent the data from sensors or actuators. Since the resolution selected in the two dsPICs is 12 bits, the frames are produced using 2 bytes per sensor or actuator, the first one corresponding to the most significant data, and the second one to the less significant data, as seen in Figure 12, where E1 and E2 are the *bytes* of the frame header, POS *n*A and POS *n*B are the most significant and less significant position *bytes* of sensor *n*, respectively, and PWM *m*A and PWM *m*B represent the most significant and less significant PWM signal *bytes* of the actuator *m*, respectively.

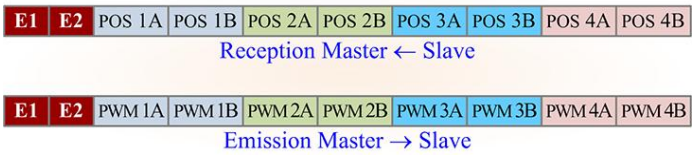


Figure 12 Scheme showing the reception frame (Master ← Slave) and the emission frame (Master → Slave)

In Figure 13 we see a simplified flow diagram representing the programming routines, functions and general configurations used in the control embedded system. The operation code of the program is developed in C language, using the MikroC compiler from MikroElektronica Company.

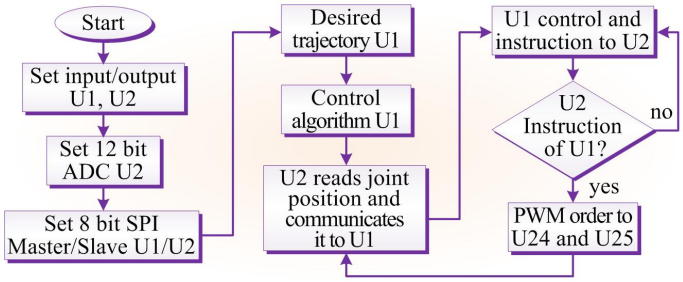


Figure 13 General Flow Diagram of the programming algorithm for the embedded control system

The actual implementation of the robotic manipulator with the embedded control system (marked with a red circle) can be seen in Figure 14.

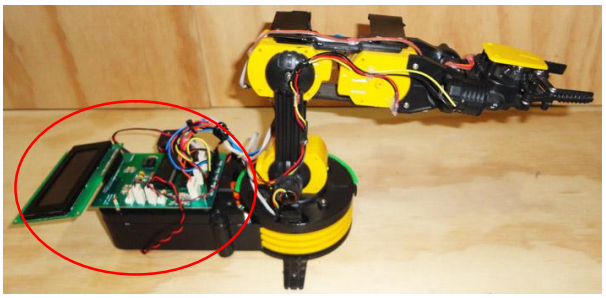


Figure 14 Implementation of the robotic manipulator showing the embedded control system

6 Results

In Figure 15 we indicate the serial communications results obtained experimentally, in the form of the signals delivered by the SPI module when the two dsPICs are interlaced. We can see the construction of the serial configuration word that is composed of an address byte and a data byte. For this purpose we used a digital oscilloscope that gets high frequency data packet measurements.

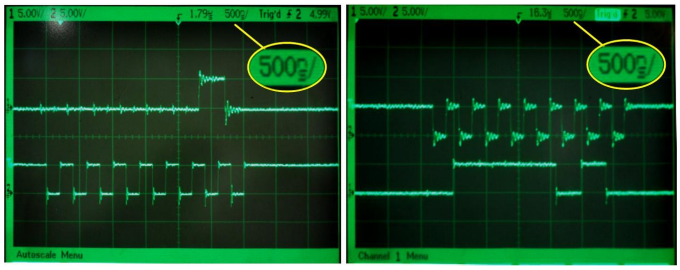


Figure 15 Results of Master/Slave SPI communications: data *byte* and address *byte*

In Figure 16 we display the curves corresponding to desired and real joint trajectories using the PID controller with antiwindup, where q_{dn} and q_n represent the real and desired joint trajectories, respectively (n indicates joints 1 to 3).

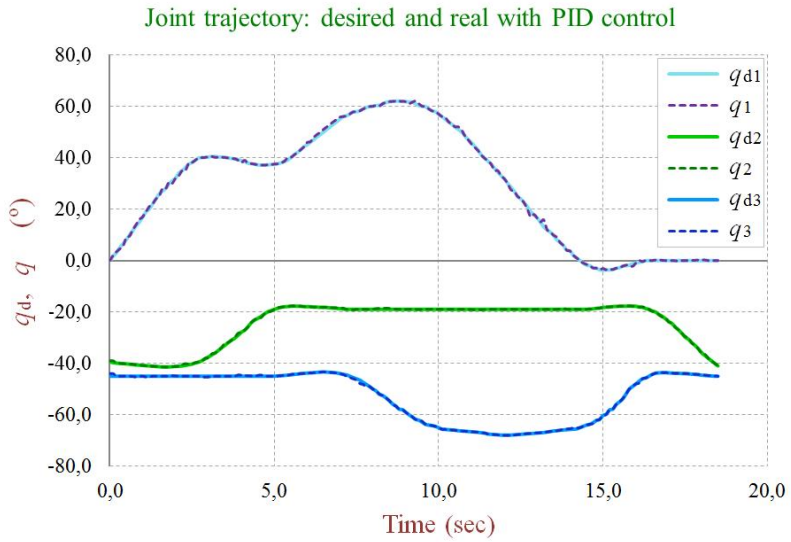


Figure 16 Comparison of desired and real joint trajectories employing the PID controller with antiwindup

Figure 17 shows the chart of the errors obtained from desired and real joint trajectories, using the PID controller with antiwindup, where e_n express the errors in joint trajectories (n represents joints 1 to 3).

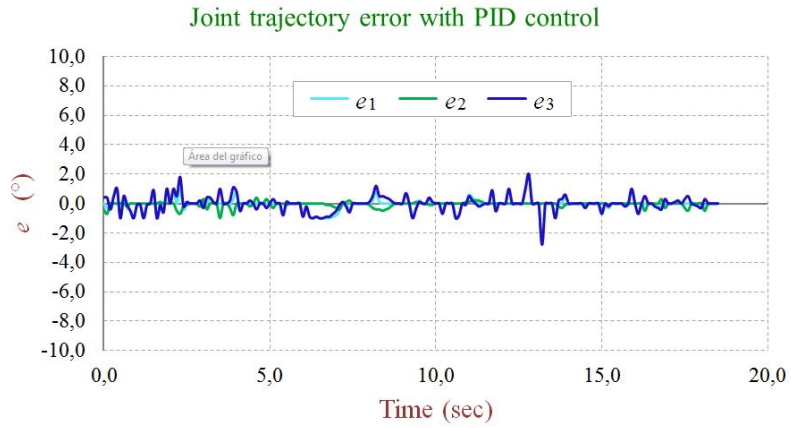


Figure 17 Error of joint trajectories, using the PID controller with antiwindup

7 Conclusions

In this work we presented the design and implementation of an embedded control system applied to a manipulator robot with 4 DOFs. We programmed algorithms for control, coding and decoding, and PWM acquisition and actuation, in DSC devices of the dsPIC30F4013 kind, arranged in a Master/Slave configuration, communicated by means of the SPI protocol.

The implemented embedded controller allowed the robotic manipulator to achieve the execution of a test trajectory in Cartesian space with a minimum position error.

The utilization of a frame header or startup has been essential to establish a correct SPI communication between the two master and slave dsPICs: data is sent in a specific sequence after this header, determining the order and identification of each data byte.

According to the results obtained from the implementation essays of the robotic manipulator, along with its embedded controller, we can start a new stage in the study, design, analysis and implementation of embedded control systems applied to robotic automation, leading to the implementation of a robotic system composed by a manipulator mounted on a mobile robot, as seen in Figure 18.

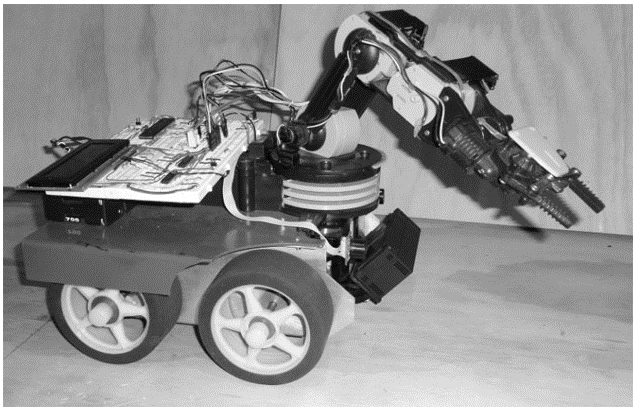


Figure 18 Prototype of a robotic system composed of a manipulator mounted in a mobile robot

Acknowledgements

This work was supported by the Mecesup USA1298 Project of the Universidad de Santiago de Chile, Chile.

References

Abdalrahman, A. and Zekry, A. (2013), "Control of the grid-connected inverter using dsPIC microcontroller", *2013 Japan-Egypt International Conference on Electronics, Communications and Computers (JEC-ECC)*, Presented at the 2013 Japan-Egypt International Conference on Electronics, Communications and Computers (JEC-ECC), pp. 159–164.

Andrade, J.D.S., Sodre, M.V.S., Dominguez, D.S., Milian, F.M. and Torres, M. (2011), "Remote Device Control and Data Acquisition Using Embedded Systems", *IEEE Latin America Transactions*, Vol. 9 No. 6, pp. 938 –943.

Ang, K.H., Chong, G. and Li, Y. (2005), "PID control system analysis, design, and technology", *IEEE Transactions on Control Systems Technology*, Vol. 13 No. 4, pp. 559–576.

Angeles, J. (2006), *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*, Springer, New York, 3rd ed.

Angulo, J.M. (2006), *dsPIC: Diseño Práctico de Aplicaciones*, McGraw-Hill Interamericana de España S.L.

Angulo, J.M., García, B. and Angulo, I. (2006), *Microcontroladores Avanzados dsPIC: Controladores Digitales de Señales : Arquitectura, Programación y Aplicaciones*, Thomson-Paraninfo.

Astrom, K.J. and Murray, R.M. (2009), *Feedback Systems: An Introduction for Scientists and Engineers*, Princeton University Press.

Astrom, K.J. and Hagglund, T. (2006), *Advanced Pid Control*, ISA-The Instrumentation, Systems, and Automation Society, 1sted.

Bareno, C. (2011), "Teching/Learning Methods for Embedded Systems Using Copyleft Hardware", *IEEE Latin America Transactions*, Vol. 9 No. 4, pp. 503 –509.

Basmage Pinheiro, L., Batista, E.A., Cambraia Leite, L. and Palmiro, F. (2011), "Development of an embedded module using FPGA technology and Fuzzy technique", *IEEE Latin America Transactions*, Vol. 9 No. 5, pp. 753 –760.

BCC Research. (2009), "Embedded Systems: Technologies and Markets", *Information Technology Report Code: IFT016C*, available at: http://www.bccresearch.com/report/embedded-systems-technologies-markets-ift016d.html.

Borges, P.S., Carvalho, T.O.C., Pires, T., Torres, M. and Milian, F.M. (2011), "Embedded System for Detecting and Georeferencing Holes in Roads", *IEEE Latin America Transactions*, Vol. 9 No. 6, pp. 921 –925.

Craig, J.J. (2004), *Introduction to Robotics: Mechanics and Control*, Prentice Hall, New Jersey, 3rd ed.

Franklin, G.F., Powell, J.D. and Workman, M.L. (1998), *Digital Control of Dynamic Systems*, Addison-Wesley, 3rd ed.

Galeano, G. (2009), *Programación de sistemas embebidos en C*, Alfaomega, 1st ed.

Huddleston, C. (2007), *Intelligent Sensor Design: Using the Microchip DsPIC*, Newnes.

Iñigo, R. and Vidal, E. (2004), *Robots industriales manipuladores*, Alfaomega, México.

Jackson, D.J. and Caspi, P. (2005), "Embedded systems education: future directions, initiatives, and cooperation", *ACM SIGBED Review*, Vol. 2 No. 4, pp. 1–4.

Melgarejo, M., Bulla, J. and Sierra, G. (2011), "An Embedded Type-2 Fuzzy Processor For The Inverted Pendulum Control Problem", *IEEE Latin America Transactions*, Vol. 9 No. 3, pp. 240–246.

Moreno, O. (2012), *Diseño y Programación de Sistemas Embebidos con el Núcleo Microblaze: Fundamentos, Conceptos y Métodos del Cómputo de Propósito Específico*, Eae Editorial Académica Española, España, 1sted.

Ollero, A. (2006), *Robótica: Manipuladores y Robots Móviles*, Marcombo, Barcelona, Spain, 1sted.

Ozturk, S. and Cadirci, I. (2013), "DSPIC microcontroller based implementation of a flyback PV microinverter using Direct Digital Synthesis", *2013 IEEE Energy Conversion Congress and Exposition (ECCE)*, Presented at the 2013 IEEE Energy Conversion Congress and Exposition (ECCE), pp. 3426–3433.

Siciliano, B. and Khatib, O. (2008), *Springer Handbook of Robotics*, Springer, Berlin, Heidelberg, 1st ed.

Spong, M.W., Hutchinson, S. and Vidyasagar, M. (2005), *Robot Modeling and Control*, Wiley, New York, 1st ed.

Szadkowski, Z. (2014), "A Prototype of Readout Electronics Based on an Unified Altera Platform for Underground Muon Counters Triggered by Surface Detectors", *IEEE Transactions on Nuclear Science*, Vol. Early Access Online, doi:10.1109/TNS.2014.2300446.

Valvano, J.W. (2009), *Introduction to Embedded Systems: Interfacing to the Freescale 9S12*, CL Engineering, Canada, 1st ed.

Valvano, J.W. (2004), *Introducción a Los Sistemas de Microcomputadora Embebidos*, Cengage Learning Editores, 1st ed.