PREDICTIVE CONTROL APPLIED TO ROBOTIC JOINTS USING FPGA

Edna Rodrigues Cassemiro
UNICAMP-State University of Campinas/Faculty of Mechanical Engineering /13083-970/Campinas-SP/Brazil
ednarc@fem.unicamp.br

Joao Mauricio Rosario
UNICAMP-State University of Campinas/Faculty of Mechanical Engineering/13083-970/Campinas-SP/Brazil
rosario@fem.unicamp.br

Didier Dumur
SUPELEC/ Plateau de Moulon/ 91192 /Gif- sur-Yvette/France
Didier.dumur@supelec.fr

Abstract. The i*ndustrial robots that carry out tasks with accuracy and rapidity are request more and more. The most industrial robots controllers does not consider the dynamics, and they use the PID with constant parameters, it becomes it rigidit with low speed operation. The study and applications of new structure control, that possibiltes good performance in relation to the parameter variations, is studied here. The development of techniques where controlling classics of type PID can be substituted efficiently is the aim this article. Generalized Predictive Control (GPC) has shown to be an effective strategy in many application fields, with good temporal and frequency properties (small overshoot, cancellation of disturbances, good stability and robustness margins), able to cope with important parameters variations. This kind of controller was studied on a robotic platform (PRR) design with 3 DOF's with the model implementation, simulation and a reprogrammable logic components design (FPGA).*

***Keywords**: predictive control, robot control, reconfigurable logic*

Introduction

In industrial environments, the robots perform tasks with rapidity and accuracy in order to produce goods and services with minimal production time. In most industrial applications, the robot tasks are programmed by learning without the need of a geometrical model. In this way, its trajectory is defined through a set of angles associated to the angular movement of each degree of freedom of the robot, which after interpolation, will act as reference signal for positioning controllers located at each joint that compare the signals deriving from the position sensors of the joints (David, 1998). For many operations, the operator defines the tasks, or reference movements, with respect to a coordinate system that is fixed to the end-effector of the robot (in the Cartesian space). The most used controller is the classical PID , however, the quick flexible manipulators are essential to achieve the performance leading to a minor production time and small energy consumption (Pimenta, 2001a,b), more resourceful control algorithms must be implemented, which can cope with important parameters variations, such as inertia. For that, the predictive control has proved to be an efficient control strategy, the maintenance of a high level of performances may be impossible to reach with a fixed predictive controller in case of important parameters variations.

The figure 1 shows the workflow to implementation the GPC (Generalized Predictive Control) using FPGA *(Field Programmable Gate Array*). Each step for this system construction is presented here, but this paper focuses only the model design, the control design, and function prototyping with FPGA. The calibration and final implementation is for a further work.

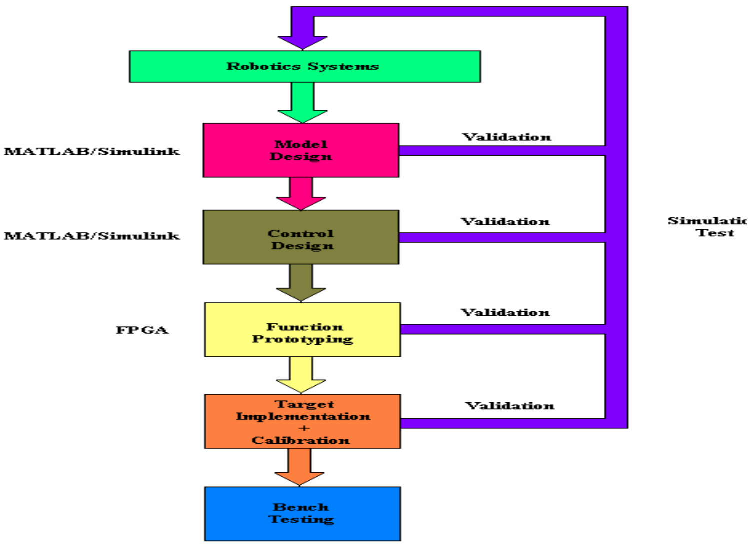


Figure 1: Workflow from model design to implementation

The implementation of this system we consider a PRR robot with 3 DOF's (fig 2). The paper is organized as follows: section 2 provides a description of this robot, including kinematics, dynamic and actuator modeling. The section 3 presents the advanced predictive axis control structure implemented under the RST formalism. Section 4 is dedicated to the test results and section 5 proposes the position controller using FPGA. Finally Section 6 has some conclusions.

2. Robot Modeling

A mechatronics device with 3 DOF (PRR robot) developed for assist in tasks for which conventional 6R manipulators have difficulties to do. This table uses 3 DC motors; each one is connected to a gear. The basis axis is connected to a ball screw to possible translational movement.

The table position control can be implemented through the control feedback of each isolated joint (Craig, 1989). Each joint model is necessary, but all the joints must be coordinated as shown on fig. 3, so that the dynamic model of the structure must also be defined. The control system, which consists essentially of three position control loops, is built with SIMULINK blocks. The inner speed and torque control loops are already included in the drive model. The rest of the manipulator and its load are represented by three SIMULINK nonlinear models, one for each motor drive.

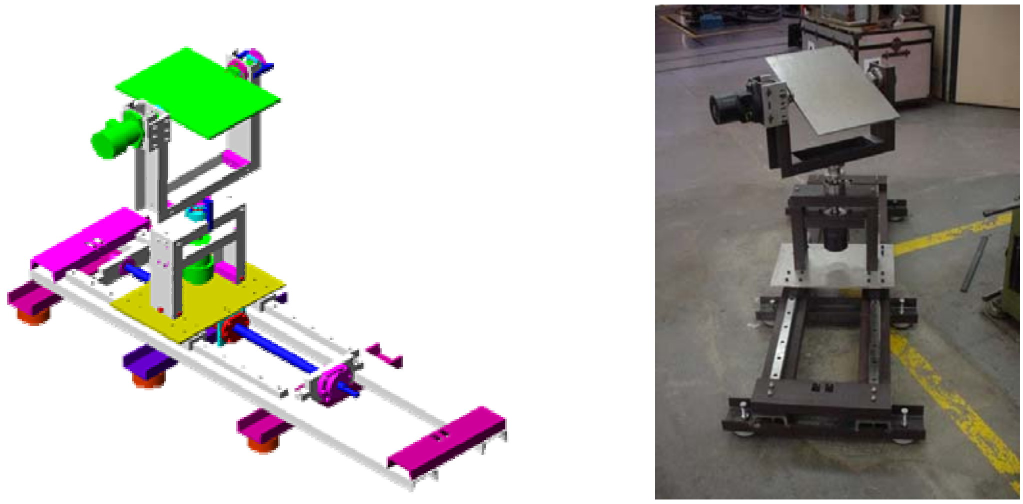


Figure 2: Robot 3 axis.

The desired movements (expressed in angular coordinates) and the control laws are in different coordinate systems, requiring the implementation of fast algorithms for the inversion of the geometrical model and generation of the reference trajectory in angular coordinates (Fig. 3).

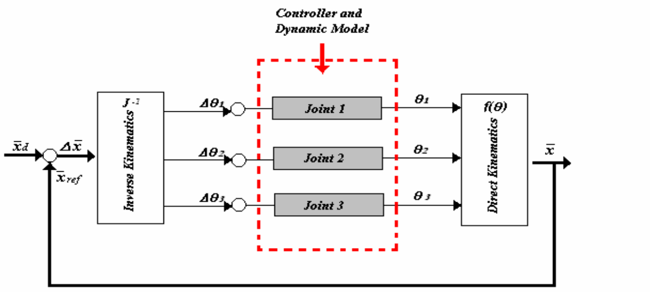


Figure 3: Inverse Kinematics Model.

2.1. Mathematical model

The geometrical model of a robot expresses the position (X, Y, Z) and orientation (ψ, θ, ϕ) of the end-effector with respect to a coordinate system fixed in the base of the robot, in function of its generalized coordinates (angular coordinates in the case of rotational joints), that is:

$$\underline{x} = f(\underline{\theta}) \qquad (1)$$

where: $\underline{\theta} = (\theta_1, \theta_2, \ldots, \theta_n)$: Angular Position and $\underline{x} = (X, Y, Z, \psi, \theta, \phi)$: Position vector.

This relation may be expressed mathematically by a matrix that relates the system of coordinates fixed in the base of the robot with a system of coordinates associated to its end effector. This matrix is designed homogeneous passage matrix and is obtained from the product of the homogeneous transformations matrix, $A_{i,\ i\text{-}1}$, that relates the system of coordinates of an element *i* with the system of the previous element *i*-1, that is

$$T_n = A_{0.1}*A_{1,2}*........*A_{n\text{-}1,n} = [\ \underline{n}\ \underline{s}\ \underline{a}\ \underline{p}\ ] \tag{2}$$

where

$\underline{p} = [\ p_x\ ,\ p_y\ ,\ p_z\ ]$: position vector;

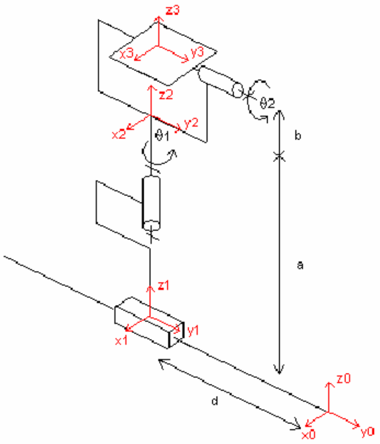
$\underline{n} = [\ n_x\ n_y\ n_z\ ]$, $\underline{s} = [\ s_x\ s_y\ s_z\ ]$ and $\underline{a} = [\ a_x\ a_y\ a_z\ ]$: orthonormal vector that describes the orientation.

The description of the transformation matrix is done through the usage of the Denavit-Hartenberg procedure, after the obtaining of the four parameters θ_i, a_i, d_i and α_i. The need for finding references in angular coordinates referring to the tasks defined in the Cartesian space is expressed mathematically by the inversion of the geometrical model, that is:

$$\underline{\theta} = f^{-1}\ (\underline{x}) \tag{3}$$

Through the function *f* it is possible to calculate the movement the end effector resulting from the movement of the joints. This function is nonlinear and has no nontrivial analytical solution.

The robot system has 3 DOF's and the kinematics model of a robotic joint can be derived through the Denavit-Hartenberg parameters formulation (fig. 4). This coefficients permit to obtain the different positions relative the parties of this mechanical device.



(a) Kinematics structure

j	α_j	d_j	θ_j	r_j
1	0	d	0	0
2	0	0	θ_1	a
3	θ_2	0	0	b

(b) Denavit-Hartenberg parameters.

Figure 4: Three axis manipulator

The position and orientation equation in the center point of the table associate for this robot are:

Position:

$$p_x = ((D6 + D4)*s23 + A2*c2)*c1 \quad p_y = (c5*D6 + D4)*s23 \quad p_z = (D6 + D4)*c23 - A2*s2 + D1 \tag{4}$$

Orientation vector:

$$\begin{array}{lll} n_x = (c1*c23 - s1\ ) & n_y = s1*c23 & n_z = -s23 \\ s_x = s1 & s_y = -\ s1*c23\ +\ c1 & s_z = s23 \\ a_x = c1*c23 + c1*s23 & a_y = s1*c23 + s1*s23 & a_z = c23 \end{array} \tag{5}$$

with $c_i = \cos(\theta_i)$, $s_i = \sin(\theta_i)$, $c_{i\,j} = \cos(\theta_i + \theta_j)$ and $s_{i\,j} = \sin(\theta_i + \theta_j)$

2.2 Dynamic model

The dynamical and control systems were studied after to solve the trajectory problem. The control of system was made for each joint (decoupling joint). The dynamic control involves the determination of the inputs, so that, the drive of each joint moves its links to the position values with required speed. The dynamic model of a robotic joint can be derived through the Euler-Lagrange formulation that expresses the generalized torque (Spong, 1995). The manipulator dynamic behavior can be described by a group of differential equations called dynamic motion equations. For a rigid manipulator with three degrees of freedom the equations are:

$$\tau_i(t)= J_i(\theta(t))\,\ddot{\theta}_i(t)+C_i(\theta(t),\dot{\theta}(t))+ Q_i(\theta(t)) \tag{6}$$

with i = 1, .. ,3
where $\tau_i(t)$ is the generalized torque vector, $\theta_i(t)$ the generalized frame vector (joints), $Ji(t)$ the inertial matrix, $C_i(\theta,\dot{\theta})$ the non-linear forces (for example centrifugal) matrix, $Q_i(\theta)$ the gravity force matrix.

The input reference is obtained in angular coordinates by trajectory interpolator, after that, it compares the reference signals with angular positions sensor of each joint (encoder incremental). The controller will make the corrections being taken into account the robot's dynamic model studied.

.

2.2.1 Actuator model

Each robotic joint commonly includes a DC Motor, a gear transmission and an encoder. Considering the DC Motor, the three classical equations are the following:

$$\begin{aligned} u(t) &= L\frac{\mathrm{d}\,i(t)}{\mathrm{d}\,t} + R\,i(t) + K_E\frac{\mathrm{d}\,\theta(t)}{\mathrm{d}\,t} \\ T_m(t) &= J_m\frac{\mathrm{d}^2\theta(t)}{\mathrm{d}\,t^2} + B_m\frac{\mathrm{d}\,\theta(t)}{\mathrm{d}\,t} \\ T_m(t) &= K_T\,i(t) \end{aligned} \tag{7}$$

where, $T_m(t)$ is the motor torque, $\theta(t)$ the angular position of the motor, $i(t)$ the motor current, L, R, J_m respectively the inductance, resistance and motor inertia.

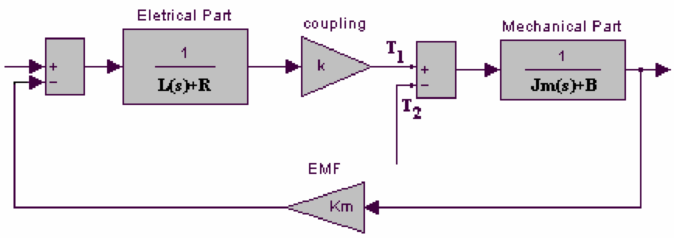


Figure. 5: DC Motor block diagram

The block diagram, showed in Fig. 5, uses of the Laplace formulation, with the electric and mechanical equations for each joint of the manipulator (eq. 7).

The inputs for these blocks are the motor speeds, and the outputs are the torques from the low-speed sides, which are applied to the robot structure model. The gears are characterized by their ratio and inertia and the stiffness and damping of input and output shafts. The gears output shafts are connected to the T1 and T2 inputs of a Robot block that represents the rest of the robot structure. This block calculates the effective torque reflected to each joint. For each three joints (numbered i), we can consider globally the other links effects as a single load reflecting to the joint a torque that is composed of three terms (eq. 6).

3. Generalized Predictive Control (GPC)

Predictive Control philosophy, aiming at creating an anticipative effect using the explicit knowledge of the trajectory in the future, can be summarized as follows (Aström, 1986) and (Boucher, 1996):

- Definition of a numerical model of the system, to predict the future system behavior,
- Minimization of a quadratic cost function over a finite future horizon, using future predicted errors,
- Elaboration of a sequence of future control values, only the first value is applied both on the system and on the model,
- Repetition of the whole procedure at the next sampling period according to the receding horizon strategy.

CARIMA model:

The CARIMA (Controlled AutoRegressive Integrated Moving Average Model) form is used as numeric model system with the aim to cancel every static error.

$$A(q^{-1})y(t)=B(q^{-1})u(t-1)+C(q^{-1})\frac{\xi(t)}{\Delta(q^{-1})} \tag{8}$$

where, u is the control signal applied to the system, y the output of the system, $\Delta(q^{-1})=1-q^{-1}$ the difference operator, *A*, *B and C* polynomials in the backward shift operator q^{-1}, of respective order n_a and n_b, ξ an uncorrelated zero-mean random sequence.

Prediction equation:

The polynomial prediction (equation 8) is obtained from input-output model. The definition of an optimal j-step ahead predictor which to anticipate the behavior of the process in the future.

$$\hat{y}(t+j)=\underbrace{F_j(q^{-1})y(t)+H_j(q^{-1})\Delta u(t-1)}_{\text{free response}}+\quad+\underbrace{G_j(q^{-1})\Delta u(t+j-1)+J_j(q^{-1})\xi(t+j)}_{\text{forced response}} \tag{9}$$

F_j, G_j, H_j are unknown polynomials and J_j are derived solving Diophantine equations.

Cost function:

The GPC strategy minimizes a weighted sum of square predicted future errors and square control signal increments:

$$J=\sum_{j=N_1}^{N_2}\left(\hat{y}(t+j)-w(t+j)\right)^2+\lambda\sum_{j=1}^{N_u}\Delta u(t+j-1)^2 \tag{10}$$

Assuming: $\Delta u(t+j)=0$ for $j \geq N_u$ four tuning parameters are required: N_1 the minimum prediction horizon, N_2 the maximum prediction horizon, N_u the control horizon and λ the control weighting factor.

RST form of the controller:

An algorithm which calculates the CPC in RST form was applied (fig. 6).

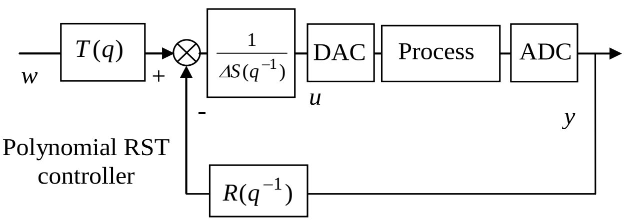


Figure 6: GPC in RST form.

$$\Delta u(t)=T(q)\,w(t)-R(q^{-1})\,y(t)-S^{*}(q^{-1})\,\Delta u(t) \tag{11}$$

The main feature of this RST controller is the non causal form of the *T* polynomial, creating the anticipative effect of this control law. The degrees of the 3 polynomials are as follows:
The polynomials are:

$$\begin{aligned}&\text{degree}\left(R(q^{-1})\right)=\text{degree}\left(R(q^{-1})\right)\\&\text{degree}\left(S(q^{-1})\right)=\text{degree}\left(B(q^{-1})\right)\\&\text{degree}\left(T(q)\right)=N_2\end{aligned} \tag{12}$$

4. Application on Robotic axis control

The purpose of the following experiments is to implement a simulator robot position control, based in Kinematics trajectories, developed in SIMULINK, considered the motor position axis of 3 DOF's industrial robot included joints dynamics is application is to test the GPC. The considered system (David, 1998) and (Isermann, 1992), developed at

UNICAMP, Brazil, included three DC motors, a 1:100 gear, a ball screen transmission (only axis 1) and incremental encoders, used for supervision and control with the respective parameters (Table 1a). An algorithm which calculates the CPC in RST form showed at section 3.4, and the results are presented. The system simulation uses three different axes (different inertias) employing the same motor. It uses an independent RST regulator for each joint, and their values were found using the tunings in the table 1b.

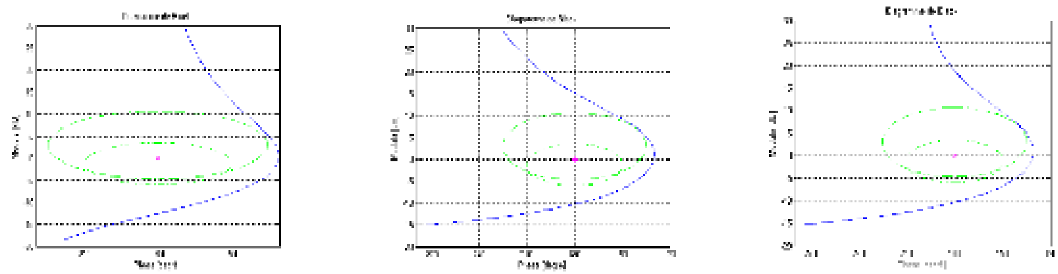
Motor Siemens 1FK6032	
Inertia (kgm^2)	0.71x10^{-3}
Weight (kg)	3
Mechanical Time Constant (ms)	2.4
Voltage constant (V/rad/s)	0.23
Torque constant (Nm/A)	0.67
Inductance (mH)	13
Resistance (Ω)	5.7

a) Motor Parameters.

Axis	N1	N2	Nu	λ
1	1	11	1	508
2	1	19	1	0.07
3	1	17	1	238

b) Tuning Parameters for each joint

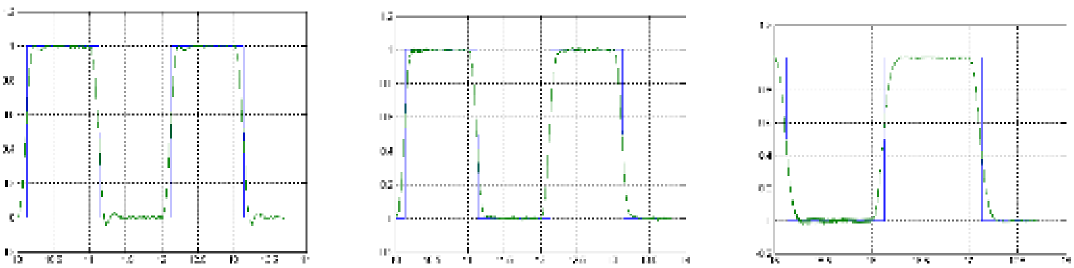
Table 1: Parameters



(a) axis-1 (b) axis-2 (c) axis-3

Figure 7: Black diagram

This tuning provides the required stability margins, as illustrated in the Black diagram Fig. 7. This diagram also shows the direct and complementary sensitivity functions, which area is avoided by the controlled system frequency response, thus providing good robustness towards disturbance (e.g. influence of a measurement noise on the control signal, influence of an output disturbance on the output (Boucher, 1996).



(a) (b) (c)

Figure 8: GPC – system response (a) axis-1 (b) axis-2 (c) axis-3

Each axis presents different inertias. The figure 8 shows the simulation results for each joint using GPC. If the inertia variation increases it is observed the GPC can regulate the system in order to become it steady.

5. Position Controller using FPGA

An alternative to controllers implemented by software is reconfigurable logic. The controller proposed has as objective the control of robotics joints. This programmable controller is capable to process the digital signals originating from a sensor coupled to the motor (ENCODER) and of a digital signal of a trajectory control (TRAJETORY). The controller's output is a digital signal for the PWM potency block. The control of the robot's joints can be implemented through FPGA, which are valuable tools in hardware projects, with integration capacity, high speed; flexibility and easy programming also present an attractive solution in project of controllers (Rosario, 2003). The controller has as objective the control of a motor of continuous current through the PWM (pulse width modulation) potency block.

For FPGA, a digital controller is implemented in RST form. This RST has its gain parameters fitting through external programming. The controller's output is a digital signal for the PWM potency block. Digital-analog or analog-digital converters are not necessary in the control loop (fig. 9). Four main blocks can be observed:

- *Error Detecting Block:* This block is used for the comparison of the signals ENCODER and TRAJETORY, generation of a proportional binary word to the error among the periods.
- *Control Register Block*: it implements the control registers, responsible for the programming of parameters in FPGA.
- *Power Interface Block:* It converts the binary word supplied by controller in a pattern of digital signals to control of the PWM block.

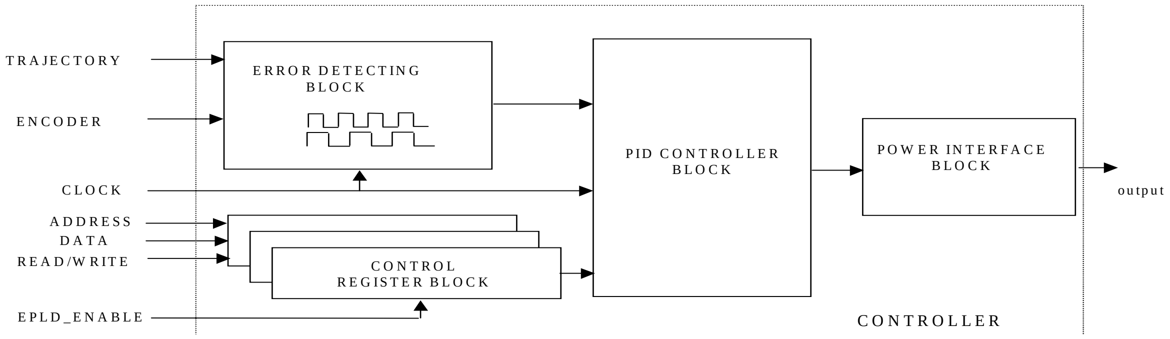


Figure 9: Control system implemented in FPGA.

The following input digital signals are considered:

- *Trajectory:* (train of pulses) responsible by the system command. The period of this signal is associated with the speed with the trajectory is executed. The number of periods it is associated to the displacement executed by the trajectory.
- *Encoder:* (train of pulses) coming of a position encoder coupled in the axis of the motor (encoder incremental). In the same way, this signal has been associating to its frequency of the rotation of the motor.
- *Clock:* responsible for the rate of increment of the FPGA internal elements.
- *Address:* It allows defining which register will be programming. It is possible to program up to 16 registers, contends information as: gains of the controller, number of bits used in internal operations or habilitation flags state.
- *Data:* Bi-directional signal of 16 bits that allows the writing and reading of binary words in the FPGA internal registers.
- *FPGA :* Habilitation signal for writing or reading in the FPGA internal registers.
- *Read/Write:* Select a write/read cycle in the FPGA internal register.

The block diagram, the FPGA graphical language using dedicated libraries (Altera, 2005) is presented in the Fig. 10. The register error block stores values of e[k], e[k-1] and e[k-2], and makes shift operations (e[k-1] = e[k] and e[k-2] = e[k-1]). The output register block stores u[k] and u[k-1].

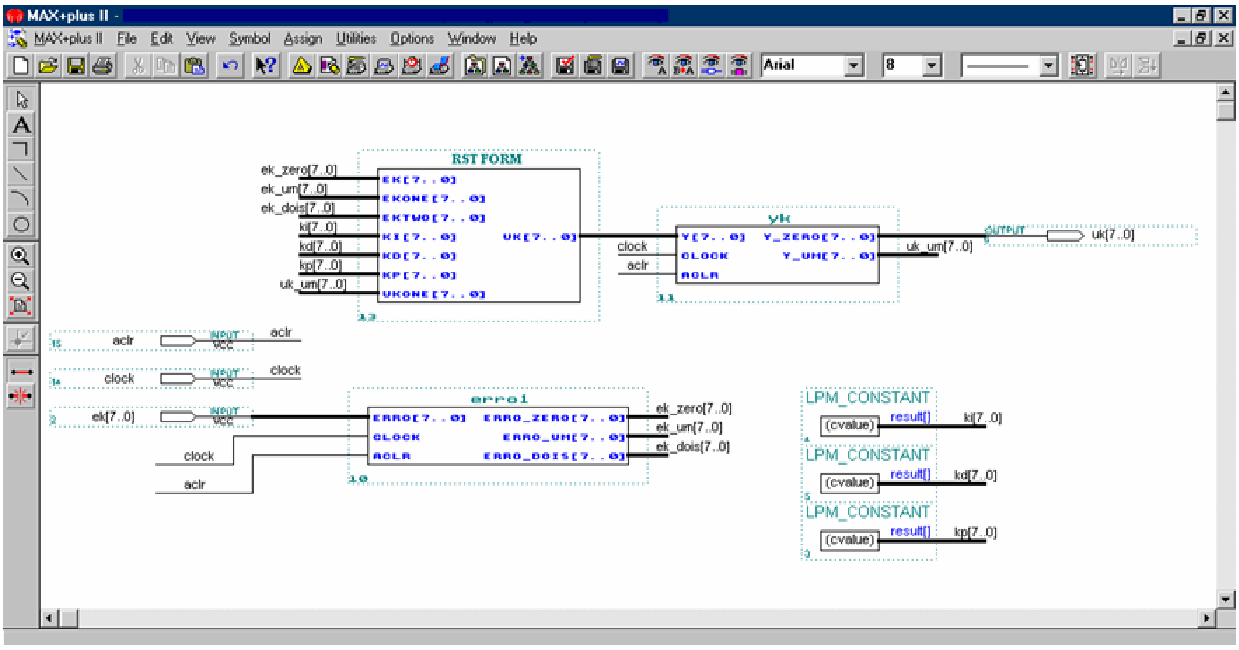


Figure 10: A digital RST regulator implemented in Graphical language using reconfigurable logic

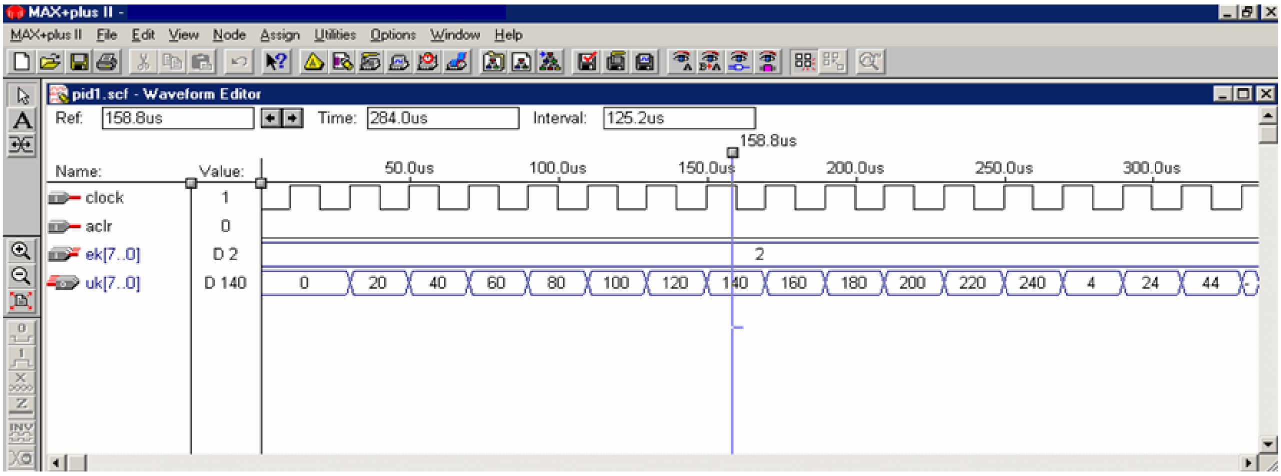


Figure 11: Digital RST regulator Simulation

6. Conclusions

This paper has presented the implementation of GPC controllers under a RST form for the control of robotic joints of robots. The GPC has a good performance in relation to the system variations. The use of FPGA allows a low cost and fast implementation of the desire controller. Other positive characteristics of FPGA can be considered: high integration capacity and it is easy maintenance and correction.

Through simulations, using the parameters of the system, the viability of the project was verified and with supported experimental, described previously, it will be possible the validation of the obtained data.

7. Acknowledgement

This paper was developed by support of the “Conselho Nacional de Desenvolvimento Científico e Tecnológico”, CNPq, and “Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – CAPES”- Brazil, through a collaboration work between the Control Department of SUPELEC, France, and the ‘Laboratory of Automation and Robotics of UNICAMP, Brazil.

8. References

Altera Corporation; Internet: http://www.altera.com, 2005.

Aström, K., Wittenmark, B., “Adaptive Control”, *Addison-Wesley Editions*, 1989.

Boucher, P., Dumur, D., “Predictive Motion Control”, *J. Syst. Eng., Special Issue on Motion Control Systems*, Vol.5, pp.148-162, *Springer-Verlag* London, 1995.

Boucher, P., Dumur, D., “La Commande Prédictive“, Collection Méthodes et Pratiques de l’Ingénieur, *Éditions Technip*, Paris, 1996.

Craig, J. J., “Introduction to Robotics: Mechanics and Control”, Second edition, *Addison-Wesley Publishing Company*, 1989.

David, S., Rosário, J. M., “Modeling, Simulation and Control of Flexible Robots”, *CONTROLO’98*, pp. 532-539, Coimbra, Portugal, 1998.

Isermann, R., Lachmann, K.H, Matko, D., “Adaptive control systems”, *Prentice Hall*, 1992

Pimenta, K. B., Souza, J. P., Rosário, J. M., Dumur, D. “Control of Robotic Joints with Generalized Predictive Control (GPC)”, *Radd’2001*, Vienne, Austria, 2001a.

Pimenta, K. B., Rosário, J. M. “The Using Techniques of Generalized Predictive Control (GPC) Applied on Control of Robotic Joints”, *Asiar*, Bangkok, Thailand, 2001b.

Rosário, J. M., Cassemiro, E. R., “Modelling and Control of Anthropomorphic Gripper in Industrial Robot”, Proc. IEEE 12th International Conference on Advanced Robotics, Coimbra, Portugal, 2003.

9. Responsibility notice

The authors are the only responsible for the printed material included in this paper.