

# Experiencias de Laboratorio de Bajo Costo para el Aprendizaje de Sistemas de Control Embebido en Tiempo Real

Patricia Baldini<sup>1,2</sup>, Guillermo Calandrini<sup>2</sup>, Néstor Campos<sup>1</sup> y Matías Frusto<sup>1</sup>

<sup>1</sup> Departamento de Electrónica, Universidad Tecnológica Nacional, FRBB, 11 de Abril 461 Bahía Blanca, Buenos Aires, Argentina

<sup>2</sup> Departamento de Ingeniería Eléctrica y de Computadoras, Universidad Nacional del Sur, Av. Alem 1253, Bahía Blanca, Buenos Aires, Argentina  
{[pnbaldi](mailto:pnbaldi@criba.edu.ar), [calandri](mailto:calandri@criba.edu.ar)}@criba.edu.ar

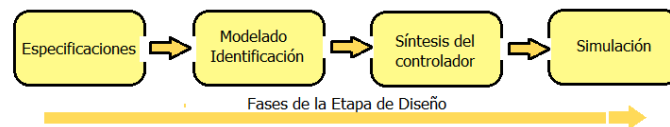
**Resumen.** La mayoría de los cursos introductorios de sistemas de control presenta un enfoque esencialmente teórico con un contenido matemático dominante. El recurso didáctico más utilizado son los laboratorios virtuales orientados a la visualización y análisis de resultados de diseños de papel y lápiz. Por otra parte, aún cuando se contemplen experiencias sobre sistemas reales, es frecuente que se trabaje solo sobre sistemas de tiempo continuo. En este trabajo se presenta un prototipo de bajo costo que permite diseñar una secuencia de experiencias de laboratorio para ser integradas en la formación en sistemas de control para ingeniería electrónica. Las experiencias, orientadas al diseño e implementación del control digital en tiempo real de un sistema dinámico, son ideadas para lograr una mejor comprensión e integración de los diversos conceptos involucrados, tales como sistemas de tiempo real y control realimentado, con la aplicación de las tecnologías actuales asociadas a sistemas embebidos.

**Palabras Clave:** Educación en control, control en tiempo real, procesador Cortex M-4, software embebido, CASCD.

## 1 Introducción

Comúnmente el ciclo de desarrollo de un sistema de control consta de dos grandes etapas bien diferenciadas: el diseño del controlador y su posterior implementación. El enfoque tradicional de la enseñanza para cursos introductorios de sistemas de control realimentado suele restringirse a la primera etapa, de modo que resultan bastante abstractos y con un contenido de matemática intensivo (Fig. 1). Frecuentemente se deja de lado la posibilidad de introducir a los estudiantes en la problemática de la implementación del sistema de control que, por naturaleza es un sistema de tiempo real. La integración y la interacción natural entre estas dos disciplinas son a menudo descuidados debido a la tendencia compartimentada de la enseñanza en carreras de ingeniería. Esto se opone al tipo de competencias con que debe formarse al futuro profesional. Por otro lado, si bien las simulaciones proporcionan una herramienta

didáctica importante que refuerza el aprendizaje de la teoría de control, las experiencias con sistemas reales resultan irremplazables. La riqueza de comportamiento y las situaciones inesperadas son imposibles de incluir en un modelo práctico; resultan más estimulantes y, en última instancia, permiten validar desde la prueba directa las ventajas y limitaciones del uso de los modelos matemáticos desarrollados [1], [2].



**Fig. 1.** Fases sucesivas de la etapa de diseño de un sistema de control

En este trabajo se presenta una propuesta pensada para proporcionar a los estudiantes de ingeniería electrónica las competencias y facilidades necesarias para implementar soluciones de control en tiempo real contemplando ambas etapas. El objetivo se logra por medio de una breve introducción a los principios básicos que sustentan el control en tiempo real. Luego, combinando un software ampliamente utilizado en control como es el entorno Simulink® [3] junto a una plataforma embebida con núcleo de procesamiento ARM® Cortex-M4 [4], se constituye un ambiente de trabajo integrado para prototipos de soluciones en control. Los algoritmos de control, descritos como modelos para Simulink® (funciones S) y traducidos a código ejecutable en el microcontrolador mediante el blockset Real Time Workshop, son validados sobre procesos o sistemas a escala de laboratorio. Se crea un ambiente de aprendizaje constructivista basado en el paradigma “learning by doing” [5] que contempla plantear a los alumnos un desafío relacionado con situaciones reales para resolver en pequeños grupos y que tiene como producto final el que producirían en su desempeño profesional.

## 2 Objetivos de Aprendizaje y Selección del Sistema

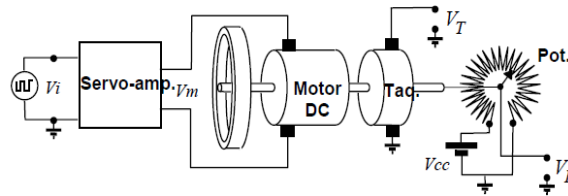
Un curso inicial de sistemas de control cubre los tópicos de sistemas lineales (ecuaciones diferenciales, función transferencia, respuesta en frecuencia) y el diseño clásico de lazos simples de control en el dominio del tiempo y de la frecuencia (Lugar de las raíces, Nyquist, Bode, Nichols, técnicas de diseño en frecuencia) para el caso de sistemas analógicos [6], [7].

Una formación práctica actual en control requiere del desarrollo de habilidades de modelado, manejo de restricciones, empleo de software para simulaciones, selección y diseño del controlador, así como conocimiento de la tecnología de sistemas embebidos para la implementación del controlador digital y la adquisición de datos para el intercambio de información en tiempo real con sensores y actuadores.

El sistema presentado en este artículo pretende proporcionar a los estudiantes una experiencia integral comenzando desde la selección de los componentes básicos para construir un sistema de control y la evaluación de las limitaciones que esos componentes imponen a los objetivos esperables. El sistema consiste en un motor de

corriente continua, de imanes permanentes, (Liotton C23, 24 volts nominales), que cuenta con un tacómetro integrado [8], [9]. Este puede ser utilizado como sensor de velocidad, para realizar un lazo interno de realimentación, y también genera una señal medible para la identificación de los parámetros. La finalidad es posicionar un volante de inercia, que actúa como carga. La acción de control es convertida en una señal PWM. Como sensor de posición se utiliza un potenciómetro lineal acoplado directamente al eje del motor (Fig. 2).

Si bien no se pide que los alumnos armen el sistema, esto les permite familiarizarse con los problemas iniciales implicados en la estructuración del mismo. Por otra parte, el control del motor presenta el equilibrio ideal entre la simplicidad de un sistema de fácil comprensión y modelado, y la riqueza de la experiencia que resulta transferible a sistemas diversos. También se contempla la posibilidad de rediseñar las experiencias incluyéndolas como parte del estudio de sistemas más complejos tales como péndulos invertidos, robots móviles, ball-beam.



**Fig.2.** Esquema del sistema de control de posición a lazo abierto

Los objetivos desde el punto de vista didáctico son dobles. Por un lado, el desarrollo de habilidades profesionales y de un conocimiento experto en control, mostrando la teoría como soporte para entender y resolver problemas reales. Se fomenta la observación reflexiva en base a una experiencia práctica experimentando activamente sobre conceptualizaciones abstractas. Por otro lado, la inclusión de controladores digitales, implementados mediante sistemas embebidos para completar la formación. El control embebido es de naturaleza multidisciplinaria requiriendo la integración de conocimientos transversales y previos. La característica digital de la acción de control generada en tiempo real implica compatibilizarla con la dinámica de la planta analógica mediante el uso de los consecuentes conversores y la correcta relación entre del tiempo de ejecución del algoritmo de control y el periodo de muestreo seleccionado.

### 3 Fundamentos del Control en Tiempo Real

Típicamente un controlador digital se implementa como un algoritmo numérico, diseñado para proporcionar un factor de corrección que modifique convenientemente el comportamiento dinámico de un sistema físico. La noción de tiempo real en control implica operaciones de cómputo que deben realizarse acorde a la evolución del sistema, interactuando con el estado del mismo a través de dispositivos de interfaz tales como conversores, sensores, y actuadores. De modo que se realizan ciclos de operación continuos, en cada uno de los cuales solo un dato muestreado es adquirido

y procesado para producir una acción sobre el sistema. La Fig. 3 muestra la estructura estándar de un lazo de control realimentado simple donde, en cada ciclo se debe cumplir una secuencia básica de operaciones: tomar una medida, convertirla en un valor numérico que se compara con un valor de referencia resultando en una señal de error que es usada dentro del algoritmo de control para computar un valor de salida. Esta salida resultante se convierte en una señal analógica que actúa sobre el sistema físico para modificar su comportamiento acorde a lo especificado. La secuencia de operaciones representa una tarea en tiempo real (RTT) que se repite cada ciclo de tiempo o período de muestreo. El período de muestreo se supone constante y debe adaptarse a la dinámica del sistema. Un período muy grande conduce a la pérdida de efectividad del control mientras que si resulta muy corto puede producir problemas de cuantización [10].

Aplicaciones más avanzadas pueden requerir la implementación de diferentes RTTs, con distintos períodos de muestreo y prioridades de ejecución.

Para sistemas de control en tiempo real de laboratorio las exigencias de tiempo son reducidas en comparación con las aplicaciones industriales. Esto es debido a que, para adquirir experiencia práctica sobre los distintos aspectos teórico de interés, es suficiente controlar una sola variable de modo que se ejecuta una única RTT.

La aproximación clásica del aprendizaje de control contempla un proceso de prueba y error que implica sucesivos cambios en el algoritmo de control antes de lograr la performance deseada, de modo que el ciclo diseño- implementación-verificación debe estar conformado para que resulte lo más corto e integrado posible. En este proceso, la generación de código ejecutable implementando el controlador digital requiere de habilidades de programación no siempre previstas en la formación y que pueden requerir una cantidad de tiempo no disponible por parte del alumno, por lo que se recurre al generador de código de Simulink, Real-Time Workshop (RTW), que crea un ejecutable que puede descargarse automáticamente en el procesador de destino (target) seleccionado [11]-[16].

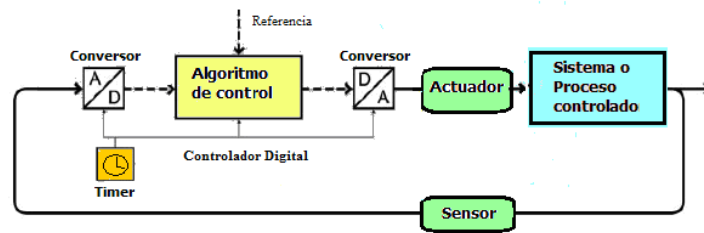


Fig. 3. Estructura de un sistema realimentado con controlador digital

#### 4 Selección de la Plataforma de Procesamiento

Esta propuesta está orientada a alumnos que no están fuertemente familiarizados con programación, tiempo real y sistemas embebidos y que deben enfocarse en experimentar con la problemática del diseño de controladores y el análisis de sus

efectos. De modo que se consideró una herramienta que deje en un segundo plano la complejidad asociada al software y hardware utilizados. También se contempló el requerimiento adicional de bajo costo. Si bien se basa en un software comercial, este constituye la herramienta de diseño de uso convencional en cualquier curso de sistemas de control, se halla disponible en la cátedra y los alumnos están familiarizados con su funcionamiento.

En este punto, se deben considerar dos aspectos. Por un lado, el firmware a cargar en el microcontrolador y un software que permita visualizar los resultados y por otro, la configuración del hardware necesario.

Simulink® permite conceptualizar soluciones gráficamente en un ambiente de modelado basado en diagrama de bloques, mientras que el blockset RTW complementado con el de Waijung [18] facilitan la generación eficiente de código C adaptado al microcontrolador adoptado (Fig. 4). Se genera código que podrá ser utilizado rápidamente sin necesidad de una larga codificación y depuración manual. RTW recompila, enlaza y descarga automáticamente ejecutables en el microcontrolador de destino, proporcionando un acceso rápido y sencillo. Al sistematizar el proceso de creación de ejecutables, estas herramientas proporcionan una manera eficaz y fiable de comprobar, evaluar y repetir diseños en un entorno de tiempo real. También provee un Profiler que permite comprobar los tiempos de ejecución del sistema embebido

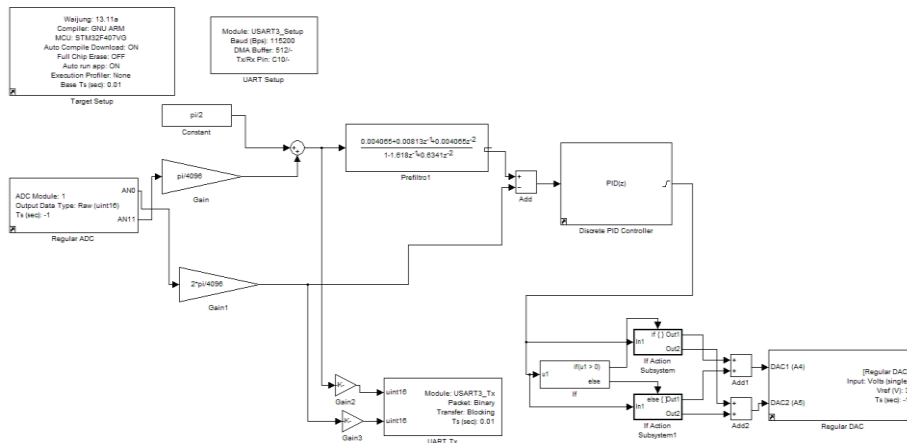


Fig. 4. Diagrama en bloques del algoritmo de control y el generador de código de Simulink

Para visualizar en tiempo real la posición de referencia y la posición del eje del motor y almacenar dicha información en memoria para su posterior análisis se utilizó el blockset de instrumentación de Simulink® (Fig. 5) y se creó un puerto serie virtual en la PC empleando el circuito integrado FT-232 con su respectivo controlador, conectado mediante un cable USB a la computadora. De esta manera, resulta muy sencillo iniciar un enlace con cualquier lenguaje de programación. En el otro extremo, el FT-232 posee una interfaz UART para la comunicación con el microcontrolador.

La placa de desarrollo, elegida por su bajo costo, alta velocidad de procesamiento y variedad de periféricos incluidos, es una STM32F4 Discovery [19] con núcleo ARM

Cortex-M4 de 32 bits. Se programa fácilmente mediante un cable USB y un programador on-board con debugger en tiempo real.

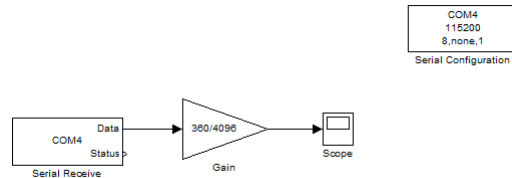


Fig. 5 . Diagrama en bloques del software de visualización

Resulta bastante sencillo poner en funcionamiento un sistema de control con muy poco hardware externo al kit (Fig. 6).

En cuanto al hardware, previo a implementar el algoritmo de control, se deben configurar al microcontrolador y sus periféricos. La frecuencia máxima que el dispositivo permite es de 168 MHz. Se utilizan dos canales uno para la referencia y otro para el sensor de posición del motor, con un timer adaptado a la frecuencia de muestreo adoptada. Para permitir el giro en ambos sentidos del motor es necesario contemplar voltajes negativos para lo cual se aprovecha la entrada diferencial de la etapa de potencia utilizando dos DAC's disponible en el microcontrolador uno conectado a la entrada positiva y el otro a la negativa.

La UART se utiliza a una velocidad de 192500 baudios con un bit de start, un bit de stop y sin paridad [20].

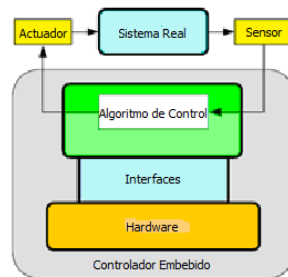


Fig. 6. Esquema del sistema de control embebido

## 5 Descripción de las Experiencias Propuestas

El objetivo final de las experiencias es diseñar un sistema para controlar la posición angular de una carga utilizando un motor DC. Se contempla una secuencia de actividades de laboratorios que permita el desarrollo de todas las etapas propias de un proyecto en control: Reconocimiento de componentes, Modelado, Identificación, Diseño del controlador analógico, Discretización e Implementación del controlador diseñado, Validación de resultados en tiempo real.

Cada experiencia contempla objetivos teóricos y prácticos que involucran tres tareas:

- Diseño y cálculos que deben realizarse antes de concurrir al Laboratorio de Control.
- Desarrollo de la práctica en el laboratorio: implementación, conexiones, mediciones y registro de datos experimentales.
- Presentación de resultado mediante un informe. En el mismo se describe la práctica desarrollada (cálculos, diseño, simulaciones, evaluación y comparación de resultados, inconvenientes, etc) y se elaboran las conclusiones. Debe entregarse en las fechas preestablecidas.

-Primera experiencia: el objetivo es describir cada uno de los componentes necesarios para implementar el sistema, indicar su funcionamiento dentro del lazo de control y obtener un modelo matemático que lo represente (Función transferencia) incluyendo las limitaciones implicadas en la elección de este modelo lineal.

Se presenta al alumno la planta y demás componentes del sistema: motor DC y carga (volante de inercia); los sensores de posición (potenciómetro) y velocidad (taquímetro); actuador (servo-amplificador); controlador (sistema embebido); comparador (etapa diferencial en servo-amplificador). El alumno debe reconocer la función que cumple cada bloque dentro de un esquema de control realimentado. Se realizan los ensayos de sensores, conexiones, rangos admisibles, limitaciones, y la comunicación de las señales de entradas y salidas para su visualización mediante Simulink. Se detecta la existencia de no linealidades, como por ejemplo “zona muerta” o saturación, se analizan sus efectos sobre la respuesta del sistema y la posibilidad de compensarlas mediante software.

-Segunda experiencia: el objetivo es estimar en base a experimentos los coeficientes de las funciones transferencia para la planta y los sensores, es decir, se realiza la identificación de los parámetros del modelo desarrollado en la experiencia anterior. Se propone utilizar la respuesta al escalón y también la respuesta en frecuencia, registrar las mediciones de entradas y salidas del sistema y procesarlas numéricamente. Considerando ciertas simplificaciones, esto implica en el caso de la planta determinar una constante de tiempo y una ganancia. Los alumnos comparan resultados y observan la incertidumbre siempre presente en la identificación.

- Tercera experiencia: el objetivo es analizar distintas estructuras o acciones de control que permitan verificar todas las especificaciones, usando técnicas de diseño en el dominio de la frecuencia. Se fijan ciertos requerimientos sobre la velocidad de respuesta, sobrepico, rechazo a perturbaciones y se sintonizan los parámetros del controlador con estructura proporcional, derivativa e integrativa (PID) en base a la respuesta en frecuencia del sistema. Se compara cualitativamente con otras opciones de control obtenidas variando las ganancias del PID: proporcional, proporcional y derivativa, proporcional e integrativa, y se observan las ventajas potenciales de cada una de estas acciones de control mediante simulación.

-Cuarta experiencia: el objetivo es la implementación del control embebido en tiempo real y la comprobación de su efectividad sobre las características deseadas de la respuesta del sistema físico. Se selecciona el período de muestreo en base a la respuesta en frecuencia del sistema y se discretiza el controlador contemplando la necesidad de usar un esquema anti windup para contrarrestar los efectos negativos de la saturación del actuador en relación a la acción integrativa del control. También se

considera la necesidad del uso de filtros anti aliasing como consecuencia de la discretización. Se programa el microcontrolador a partir de un modelo en Simulink y el generador de código automático específico. Finalmente se evalúa el tiempo de ejecución del algoritmo de control empleando el Profiler del Blockset de Waijung para garantizar que se complete la RTT en un tiempo menor al período de muestreo seleccionado. Finalmente se compara los resultados obtenidos con las simulaciones realizadas en la experiencia 3.

## 6 Conclusiones

En este trabajo, se desarrolla y presenta un ambiente de trabajo basado en Matlab para proyectos con controladores embebidos aprovechando las facilidades provistas por Simulink, los Blockset de tiempo real y dispositivos embebidos, y las capacidades de comunicación serie de Matlab con el microcontrolador seleccionado. Se ilustra su potencial formativo en un curso inicial de control realimentado para ingeniería mediante la propuesta de una secuencia de experiencias didácticas sobre una estructura basada en un motor de corriente continua con una plataforma de procesamiento fácilmente configurable. Esta secuencia contempla la generación automáticamente de código para programar el microcontrolador a partir del algoritmo de control diseñado, variando tanto los parámetros como la estructura del controlador, la adquisición de señales provenientes de sensores, el procesamiento de datos en línea, la visualización en tiempo real de los resultados utilizando instrumentos virtuales y el almacenamiento de datos pertinentes para el posterior análisis. El objetivo principal es poner en evidencia la naturaleza interdisciplinaria de todo sistema de control integrando dentro de lo que permite el tiempo disponible, las etapas de diseño e implementación (electrónica, sistemas en tiempo real, teoría de control ,sistemas embebidos).

Se remarca que el uso del sistema embebido tanto para la adquisición de datos como para generar la acción de control representa una solución de bajo costo que resulta particularmente beneficiosa para cursos numerosos. El sistema adoptado puede ser integrado fácilmente a sistemas mas complejos para generar actividades complementarias en cursos avanzados.

## Referencias

1. Feisel, L.D. and Rosa, A.J.: The Role of the laboratory in Undergraduate Engineering Education. *Journal of Engineering Education*, Vol. 94 , pp. 121-130 (2005)
2. Welsh, J.S., Daredia, T., Sobora, F., Vlacic, L., Goodwin, G.C.: Simulated versus Hardware Laboratories for Control Education: A Critical Appraisal. In: *Proc. of the 17<sup>th</sup> World Congress IFAC . Korea . pp. 8129-8135 (2008)*
3. Mathworks : <http://www.mathworks.com/products/simulink/>
4. ARM processor. Cortex-M Series: <http://www.arm.com/products/processors/cortex-m/cortex-m4-processor.php>
5. Franklin, G., Powell, J. and Emami-Naeini, A.: *Feedback Control of Dynamic Systems*. 7<sup>th</sup> Ed. Prentice Hall. New York (2015)



6. Schank, R.C., Berman, T.R. and Macpherson, K.A.: Learning by Doing. In :Reigeluth, C.M.(eds.) Instructional-design Theories and Models: A New Paradigm of Instructional Theory, Vol. 2 , Ch. 9. pp. 183-214. Taylor and Francis (2009)
7. Ogata, K.: Modern Control Engineering. 5<sup>th</sup> Ed. Prentice Hall. New York (2010)
8. Gunasekaran, M., Potluri, R.: Low-Cost Undergraduate Control Systems Experiments Using Microcontroller\_Based Control of a DC Motor. IEEE Trans. Education. Vol. 55, no. 4, pp. 508-516 (2012)
9. Siller-Alcalá, I.I., Jaimes-Ponce, J., Alcántara-Ramirez, R., Rubio-Avila, J.J.: A Hands-on Process Control Laboratory for Undergraduate Students: Using Low Cost Experiment Kits. Proc. Od the 7<sup>th</sup> WSEAS Int. Conf. Education and Educational Technology . pp.167-172 (2008)
10. Salzmann, Ch., Gillet, D., Huguenin, P.: Introduction to Real-time Control using LabVIEW™ with an Application to Distance Learning. International Jour. Eng. Education, vol. 16, no. 3 (1999). Available online: <http://www.ijee.ie/articles/Vol16-3/ijee1123.pdf>. Access: mayo 2014.
11. Lim, D. J.: An Undergraduate Laboratory Course in Real-Time Dynamic Control. IEEE Trans. Education. Vol. 48, no. 1, pp. 105-110 (2005)
12. Martí, P., Velasco, M., Fuertes, J.M., Camacho, A., Buttazzo, G.: Design of an Embedded Control Systems Laboratory Experiment. IEEE Transactions on Industrial Electronics, Vol. 57, no. 10, pp.1-10 (2010)
13. Quaranta, G., Mantegazza, P.: Using Matlab-Simulink RTW to Build Real Time Control Applications in user Space with RTAI-LXRT. Real time Linux Workshop Milano, Italy. pp.19-26 (2001).
14. Rodríguez, A.A., Cifdaloz, O., Dhirasakdanon, T.: Description of a Modeling, Simulation, Animation, and Real-Time Control (MoSART) Environment for a Class of Electromechanical Systems. IEEE Trans. Education. Vol. 48, no. 3 , pp. 359-373 (2005)
15. Shiakolas, P.S.: Development of a Real-Time Digital Control System with a Hardware in the loop Magnetic Levitation Device for Reinforcement of Control Education. IEEE Trans. on Education, Vol. 46, nro. 1, 79-87 (2003)
16. Velasco, M., Martí, P. Castañé, R., Villá, R., Fuertes, J.M. Key Aspects for Co-Designing Real-Time and Control Systems In: Proc. Int. Workshop Real-Time and Control. RTC 05 (2005).
17. Teng, F.C.: Real-Time Control Using Matlab Simulink. Proc. of Systems, Man and Cybernetics .Vol. 4 , pp. 2697-2702 (2000).
18. Waijung Blockset: <http://waijung.aimagin.com/>
19. ST Microelectronic. Reference Manual RM0090: *STM32F407xx Advanced ARM-based 32-bits MCU*.
20. Campos, N.D., Frusto, M., Baldini, P.N.: Controlador PID diseñado bajo control robusto QFT implementado sobre Cortex M-4. En: 5<sup>to</sup> Congreso de Microelectrónica Aplicada.  $\mu$ EA 14. Córdoba. (2014)