

Una Revisión Sistemática a las Aplicaciones en Prototipado Rápido para Control

J. Hoyos-Gutiérrez, J. Cardona-Aristizabal, P. Muñoz-Gutiérrez and D. Ramirez-Jimenez

Title— A Systematic Literature Review on Rapid Control Prototyping Applications

Abstract— In this paper, a systematic literature review of rapid control prototyping issue is presented, which facilitates the education in modeling and control of dynamic systems, because it reduces the effort programming. Six questions were proposed that were used in the analyses of one hundred references among conference and journal papers. The used questions are, for example, the dates and geographic regions of the publication, software and hardware platforms, differences, similarities and application areas. Several applications were reviewed in electrical systems, transport vehicles, nonlinear systems, power electronics, process control, industrial automation and robotics. It concludes in which geographic regions are more employed, advantages and problems of his use, what differences exist between the commercial and free software design.

Index Terms— Control engineering education, Engineering design tools, Computer aided software engineering, Rapid control prototyping.

I. INTRODUCCIÓN

El prototipado rápido para control (RCP de sus siglas en inglés Rapid Control Prototyping) [1], trata sobre las técnicas que permiten el desarrollo de la estrategia de control para una aplicación o planta de laboratorio de manera rápida y sencilla, para lo anterior se aprovechan elementos de hardware y software previamente desarrollados. En RCP no es necesario codificar programas en lenguajes como 'c', ni programar directamente los puertos del procesador que posea la plataforma o tarjeta de desarrollo, sino que de manera gráfica se realiza un esquema y luego, con dicho esquema, se programa directamente la plataforma de desarrollo.

Además del control de sistemas dinámicos, existen otras áreas que también emplean la denominación de prototipado rápido, por ejemplo en mecánica [2], programación de computadores [3] y electrónica digital orientada a mecatrónica [4]. Este artículo se centra solamente en la revisión de técnicas de prototipado rápido para el control.

En RCP el hardware se puede definir como una plataforma física que se compone del microprocesador, memoria y los puertos de entrada y salida, generalmente: i) un computador personal o PC, ii) un microprocesador o microcontrolador, iii) un DSP (del inglés Digital Signal Processing) y iv) lógica

programada. El software son los programas que permiten la programación en entornos gráficos (Ej. Simulink, Labview y Scicos) de los cuales se genera el código para el hardware. Una descripción del proceso RCP para un DSP, fue descrito por Hong y otros en 2002 [5].

En la literatura sobre RCP es común encontrar el término "hardware en el lazo" (del inglés Hardware In the Loop, abreviado como HIL). El HIL emplea un controlador en hardware o físico con una planta simulada en un computador, donde las entradas y salidas están acopladas a un sistema de señales externo, como por ejemplo una tarjeta de adquisición de datos [6]; en este caso el modelo simulado debe calcularse en tiempo real [7]. Otro término es el de Software en el lazo (SIL), que trata sobre simular en tiempo real la planta y el controlador, SIL tiene una ventaja sobre HIL en que, al no requerirse entradas y salidas, la integridad de las señales se preservan [8].

Ejemplos de los primeros trabajos en sistemas de prototipado rápido en control se describen en [9]- [10]. En 1992 [9], fue presentado un sistema compuesto de varios procesadores, siendo el principal un Motorola 68020. El sistema era programado en un lenguaje de texto específico llamado RPL (del inglés Rapid Prototyping Language), para el que desarrollaron un compilador. El sistema consta también de un procesador de señales DSP TMS32010 que ejecuta partes del programa de aplicación. En 1993 [11] se emplearon los programas MATRIXx y SystemBuild para la programación de un microprocesador de 16 bits. El programa MATRIXx permitía la programación gráfica por bloques y con el SystemBuild se generaba código en lenguaje Ada a partir del esquema, estos dos programas fueron aplicados al control de vuelo de un satélite.

Entre los años 1992 y 1993 la empresa dSpace integró a sus tarjetas de desarrollo la generación automática de código desde programas Matlab/Simulink. Lo que permitió que por ejemplo, en 1995 [12] programaran un DSP a partir del código escrito en Matlab y en 1997 [13] se empleó programación gráfica con Simulink.

En 1997 [10] implementaron el control de un rectificador monofásico controlado por señales PWM con Matlab/Simulink y la plataforma dSpace DS1102, en este caso emplearon tres técnicas: control PI (Proporcional Integral) de voltaje con bloque predictivo en cascada, basado en modelo de voltaje y bloque predictivo en cascada, PI de voltaje y control por vector de corrientes. Los autores

concluyen que el código que genera el Simulink es altamente optimizado y sólo les tomó una semana su desarrollo.

Algunos documentos que han tratado de manera general sobre RCP, han sido [1], [7], [14] y [15]. En [1], publicaron sobre la librería TargetLink para Matlab/Simulink y cómo esta genera código ejecutable para microcontroladores a partir de los diagramas de Simulink. Una revisión a las tecnologías de prototipado rápido en control, fue realizada en [7], inicialmente presentó una revisión breve de conceptos básicos de hardware y tiempo real, luego expuso los criterios para selección de plataformas como son: el tiempo de muestreo mínimo posible, tiempo de desarrollo del algoritmo de control, peso y tamaño (movilidad) y precio, con estos datos presentó una tabla en la cual se comparan diversas opciones de desarrollo con el paquete Matlab/Simulink. Finalmente ilustra equipos de laboratorio o casos de estudio como, por ejemplo, el mecanismo de aceleración de un automóvil.

En [14], realizaron una comparación entre las herramientas y plataformas RCP que existían antes del año 2000, tanto para el sistema operativo Windows como Linux, en este trabajo se concluye que el dSpace es la más completa en cuanto a opciones interactivas y bajo tiempo de muestreo. En el Capítulo siete del libro de Abel y Bolling en 2006 [15], se presentó un compendio de las herramientas hardware y software que existen para prototipado rápido, los autores listaron las principales ventajas de usar RCP:

- Permite un chequeo rápido y económico de controladores.
- Otorga experiencia en el control del sistema (empleo de HIL).
- Facilita ajustar y optimizar los parámetros del controlador.
- Posibilita la adquisición y visualización de los datos.
- Permite usar herramientas de análisis.

Actualmente se cuenta con múltiples opciones para realizar prototipado rápido en control, las comerciales como por ejemplo Matlab/Simulink, LabView y las no comerciales como Scilab/Scicos. En el caso del programa Simulink, existen varias posibilidades [16]:

- Empleando un PC. (Ej. Simulink Coder ó Real Time Workshop [17])
- Empleando un DSP. (Ej. Embedded Coder [18])
- Empleando microcontroladores. (Ej. TargetLink [1])

En el caso de Scilab/Scicos existen las opciones de RTAI (del inglés Real Time Application Interface) [19] y [20] para ejecutar desde un PC y la versión libre de X2c [21] para plataformas embebidas.

Una ventaja de Simulink sobre Scicos es el llamado modo externo, en el que una plataforma de cómputo es enlazada con el computador donde se diagrama el esquema de Simulink, el esquema de control es descargado en la plataforma de cómputo y mediante memorias compartidas entre las dos plataformas [22] se tiene comunicación de parámetros y señales. Una vez descargado el código generado es posible cambiar parámetros del controlador y observar las señales reales que se requieran, por ejemplo, en [23], emplearon una Raspberry PI o con la plataforma dSpace [24].

El aporte de este artículo es presentar un análisis de la literatura en aplicaciones de RCP, que sirva de ayuda en la selección de herramientas y plataformas de RCP para cursos

relacionados con el área de automática. En lo que resta del documento se presenta la metodología, resultados y conclusiones de la revisión.

II. METODOLOGÍA

Con el propósito general de realizar una revisión a las aplicaciones, se plantearon seis preguntas de investigación y se realizó una búsqueda intensiva en bases de datos de artículos mediante palabras claves descritas más adelante, se vincularon nuevas palabras clave a medida que se profundizaba en la revisión de la literatura. En el proceso se excluyeron los artículos que no daban respuesta a las preguntas de investigación que se plantearon.

A. Preguntas de Investigación

Para la revisión sistemática fueron planteadas las siguientes preguntas:

RQ 1: ¿Cuáles son las fechas de publicación, regiones y distribución geográfica de la filiación de los autores?

RQ 2: ¿Qué programas fueron utilizados para la creación y programación del Prototipado rápido?

RQ 3: ¿En qué plataformas fue implementado, por ejemplo, microcontrolador, DSP o Computador de escritorio?

RQ 4: ¿Se ve fácil o viable la posible replicación de los proyectos descritos por las publicaciones?

RQ 5: ¿Cuáles son las similitudes o diferencias entre sí, por ejemplo, es de bajo costo?

RQ 6: ¿En qué ha sido aplicado? plantas, sistemas, etc.

Estas preguntas se redactaron de manera heurística buscando identificar que tecnologías hardware y software de RCP han sido aplicadas a sistemas didácticos y los beneficios y desventajas de emplearlo.

B. Estrategia de búsqueda y selección

Se consultaron las bases bibliográficas:

- IEEE Xplorer.
- ScienceDirect.
- SpringerLink.
- Wiley online library.
- ACM.
- Jstor.

Con las cadenas de búsqueda en títulos de los artículos:

- “rapid control prototyping”.
- “rapid prototyping” y en alguna parte del título “control”, “real time”, “education”.
- “code generation” y en alguna otra parte del título “control”.
- “hardware in the loop”.
- “embedded” y en alguna otra parte del título “control”.

De varios de estos artículos se revisaron sus referencias, tomando de estas las que tuvieran relación directa con RCP. De 110 referencias consultadas se excluyeron 22 porque no

trataban el tema de prototipado rápido en control, si no, sobre la estrategia de control empleada o no daban detalles de la plataforma hardware empleada, con lo cual, sin dicha información no era posible responder las seis preguntas de investigación que se plantearon.

III. RESULTADOS

Se presentan respuestas a las seis preguntas planteadas anteriormente. Además, como respuesta a la pregunta 6, se realiza una revisión de aplicaciones en circuitos eléctricos y electrónicos, sistemas no lineales, electrónica de potencia, control de procesos continuos, automatización industrial, robótica y vehículos de transporte.

A. Respuestas a las Preguntas

RQ 1: ¿Cuáles son las fechas de publicación, regiones y distribución geográfica de la filiación de los autores?

En la Figura 1, se muestra el número de publicaciones por año, es posible observar que entre 2008 y 2012 se tuvo el mayor auge de publicaciones, para la obtención de la gráfica se emplearon 90 entre ponencias en seminarios o congresos y artículos en revistas. En la Tabla I, se presentan los porcentajes de publicaciones según las diversas regiones geográficas, Europa y Asia lideran la tabla, igualmente se emplearon 90 referencias.

RQ 2 y RQ 3: ¿Qué programas fueron utilizados para la creación y programación del Prototipado rápido?

¿En qué plataformas fue implementado, por ejemplo, microcontrolador, DSP o Computador de escritorio?

En las Tablas II y III que se encuentran al final del documento, se puede observar los programas y plataformas empleados para RCP. Según los artículos revisados en este documento, el programa más empleado es Matlab/Simulink y la plataforma más utilizada es el computador tipo PC, tanto con el sistema operativo Linux como Windows, con la librería RT de Simulink.

RQ 4: ¿Se ve fácil y/o viable la posible replicación de los proyectos descritos por las publicaciones?

Si se cuenta con la licencia del programa Matlab es fácil implementar el prototipado rápido, como puede verse por ejemplo en [25], [26] que emplearon los microcontroladores Arduino y Dspic respectivamente o en [27], [28] que emplean un PC ejecutando el código de RCP y un microcontrolador como tarjeta de adquisición, también es posible emplear dos PC como en [29], [30], donde un PC sirve para el diseño con Matlab/Simulink y un PC adicional ejecuta el código generado desde el Simulink.

El Labview también es una aproximación sencilla para implementar RCP, se puede emplear una tarjeta compatible con el fabricante National Instruments, por ejemplo en [31] y [32] o una plataforma de lógica programada FPGA (del inglés Field Programmable Gate Array) como en [33], [34]. Por el lado de software libre se puede emplear el Scilab/Scicos con una versión libre de la librería X2c como en [21] que emplea DSP o microcontroladores de alta gama.

RQ 5: ¿Cuáles son las similitudes o diferencias entre sí, por ejemplo, es de bajo costo?

La mayoría de aplicaciones presentan similitudes, y en lo que corresponde a las diferencias podrían estar por ejemplo, en el empleo de laboratorios remotos como en [35], [36], software libre [19], [21], pruebas de usabilidad de las

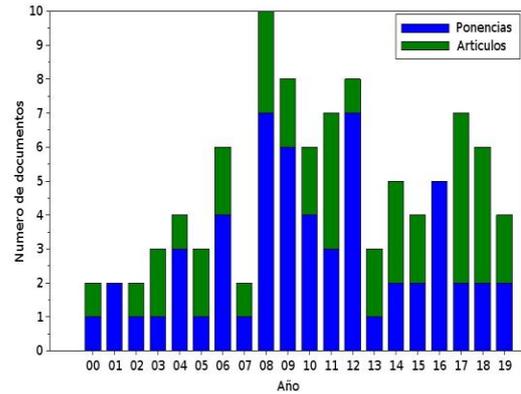


Fig. 1. Publicaciones vs años. (verde) artículos en revistas, (azul) ponencias en congresos.

plataformas desarrolladas como en [37], empleo de redes de comunicación como bus CAN (del inglés Controller Area Network) [38], problemas que se presentan en la generación de código [39], distribución del código en varios microprocesadores y multitarea [40].

En cuanto a plataformas RCP de bajo costo están las que emplean el Scilab/Scicos [21] o Modelica [40], las que emplean microcontroladores como [27], [28] como tarjeta de adquisición o [25], [26] como plataforma de descarga. Información adicional a esta pregunta podría ser encontrada en la sección de aplicaciones de este documento.

RQ 6: ¿En qué ha sido aplicado? plantas didácticas, sistemas dinámicos, etc.

En la Tabla IV, al final de este documento, se presenta un compendio por aplicaciones, de las referencias empleadas, siendo las más encontradas: electrónica de potencia y sistemas no lineales (en su mayoría aplicaciones para enseñanza en universidades). Información adicional a esta pregunta podría ser encontrada en la sección de aplicaciones.

B. Aplicaciones

Seguidamente, se revisan algunas aplicaciones en sistemas eléctricos, sistemas no lineales, electrónica de potencia, control de procesos continuos, automatización industrial, robótica y vehículos de transporte.

Circuitos Eléctricos y Electrónicos

El uso de RCP con Matlab/Simulink desde Windows de circuitos eléctricos fue presentado en [5], [17] y [41]. En [5], usaron un filtro FIR (del inglés Finite Impulse Response) simulado como modelo y el controlador es un DSP C6000 de Texas, los autores desarrollaron código de los bloques de Simulink para explotar el paralelismo y las ventajas de operar números reales del DSP.

TABLA I
PORCENTAJES DE PONENCIAS Y ARTÍCULOS PUBLICADOS
SEGÚN REGIÓN GEOGRÁFICA.

Región Geográfica	Porcentaje
Europa	51
Asia	26
Norteamérica	15
Suramérica	5
África	2
Australia	1

Con un circuito de segundo orden en [17], como sistema a controlar, un PC ejecutaba el Simulink con la librería RTW (del inglés Real Time Workshop), como control emplearon las técnicas de PID (Proporcional Integral Derivativo) y ubicación de polos, el prototipado rápido en control reduce costos y trabajo de ingeniería en las fases de diseño y pruebas, es entre otras, una de las conclusiones del artículo. En [41], emplearon un circuito RC, este sistema corre el RCP directamente en un solo PC gracias al RT-SIL (del inglés Real Time Software In the Loop), en el cual el tiempo de muestreo de la simulación del esquema Simulink es sincronizado a intervalos exactos, para que trabaje en tiempo real, los autores exponen que este tipo de implementación trabaja bien para plantas simples, dado que es tiempo real blando por la limitación del sistema operativo Windows.

En [19] emplearon un computador analógico como sistema dinámico y Scilab/Scicos con Linux RTAI en un PC, en este programaron técnicas de control PID y filtro adaptativo, concluyen que el sistema es de bajo costo y que su inconveniente es el proceso de compilar y modificar el sistema operativo para el PC con Linux, en [42] los mismos autores de [19] presentaron en 2011 un trabajo sobre la sintonización de dichos controladores. Un trabajo similar al anterior fue presentado en [43], probaron para un circuito integrador simple simulado, las técnicas de RCP empleando Matlab/Simulink, para lo cual usaron un PC con Linux en tiempo real (RTAI). Los autores reportan que el proyecto ha sido aplicado con éxito en reducción de vibraciones, péndulo invertido, celdas fotovoltaicas, estructuras flexibles. Como conclusiones, los autores proponen que se requieren pocas habilidades de programación y que la sobrecarga de cómputo, producto de la generación automática de código, no es obstáculo para lograr una frecuencia de muestreo de 10Khz.

Sistemas No lineales

El control implementado con RCP de un péndulo invertido simple ha sido presentado en [21], [44]-[48]. En [21] emplearon Scilab/Scicos y la librería X2c para programar un microcontrolador de 32 bits STM32F4 de Texas, emplearon un control por realimentación de estados adaptativo, entre sus conclusiones es que el sistema presentó buenas respuestas y el software y herramienta empleados son de uso libre. Con Matlab/Simulink, Grega y otros en 1998 [44], probaron la implementación de un controlador difuso corriendo en un PC. Expusieron las ventajas de emplear herramientas computacionales para diseño y análisis de sistemas de control, además, como la librería RTW de Matlab funciona para lograr el tiempo real. Los autores resaltan la importancia del empleo del PC como plataforma por su bajo costo, aunque no es confiable su uso en aplicaciones de control en la industria.

Spong y otros [45] en 2001, programaron un DSP TMS320C6711 con Matlab/Simulink y la técnica de control por balance de energía para balancear y erguir el péndulo, luego un control de balance lo estabiliza en su posición invertida, el artículo no presentó conclusiones. En [46], controlan un péndulo invertido simple rotacional mediante LQR (del inglés Linear Quadratic Regulator) y ubicación de polos con una tarjeta de desarrollo propio a partir del DSP F28335, el cual programaron en Matlab, los autores agregaron nuevos bloques para manejo de los puertos del

DSP, concluyen que gracias al RCP con Simulink y su extensa librería se tiene una forma de adaptar algoritmos de manera rápida y simple.

En [47] aplicaron control difuso usando la librería respectiva de Matlab/Simulink y un DSP TMS320F2812 de Texas para lograr el tiempo real. El principal aporte de este trabajo fue el desarrollo de un sistema de control difuso, concluyeron que el RCP con Matlab mejoró la eficiencia de desarrollo del experimento considerablemente. Un control de un doble péndulo invertido mediante una plataforma de bajo costo, fue presentado en 2010 [48], emplearon Matlab/Simulink y un DSP TMS320F2812, según los diseñadores de la plataforma el poder controlar una planta no lineal y multivariable muestra las bondades de ésta, con las ventajas de su bajo costo y que el diseñador de controladores se pueda enfocar más en la creatividad y resolver problemas de ingeniería, que en la parte de implementación de hardware y software.

En [27], presentaron un sistema de RCP con PC de un péndulo no invertido programado con Matlab/Simulink y un microcontrolador PIC como tarjeta de adquisición, aplicaron control PID y adelanto-atraso del ángulo del péndulo, el cual es movido por una hélice en un extremo, los autores realizaron encuestas a los estudiantes de forma anónima para conocer el sentir de estos ante su aprendizaje de conceptos de control y también sobre la implementación, uso y necesidad de asistencia que requirieron del docente sobre el proyecto, concluyen que el proyecto permite a alumnos de carreras diferentes a la ingeniería eléctrica familiarizarse con las técnicas de control debido a la facilidad para su realización y que es bien aceptado por los estudiantes.

Sistemas de levitación de bolas por electromagnetismo o aire han sido presentados en [25], [29] y [49]. En [25], emplearon Matlab/Simulink para un levitador por ventilador, como plataforma RCP emplearon el Arduino Uno, en la cual descargaron el código de control de una estrategia PI que genera la librería Arduino Target de Matlab, los autores resaltan el bajo costo y fácil consecución de la plataforma y que esta es adecuada para educación en control, de todos modos, es de considerar la dependencia de Matlab, cuya licencia si tiene un costo. En 2001 [29] y 2003 [49] emplearon dos PC que se comunican por el puerto serial, uno con Matlab/Simulink y otro con un sistema operativo llamado xPC, el cual tiene una tarjeta de adquisición que conecta el sistema, la conclusión principal es que la facilidad para implementar un compensador adelanto atraso, permitió a los estudiantes enfocarse en comprender sobre control y no en escribir código a bajo nivel, con lo que notaron más entusiasmo en los estudiantes.

Sistemas de modelos didácticos de helicópteros controlados por RCP fueron presentados en [31], [50], [51] y [52]. En [31], programaron en Labview un PC con una tarjeta de adquisición MyDAQ, el helicóptero puede ser portable y los estudiantes pueden llevarlo a casa y desarrollar sus actividades. En [50], emplearon Matlab/Simulink con la Dspace DS1104 para pruebas de SIL (control en el PC) y la descarga del algoritmo de control fue realizada en un microcontrolador de 32 bits TC1775, las estrategias de control fueron por realimentación de estados y linealización exacta, los autores en resumen concluyeron que al usar RCP los estudiantes pueden investigar el comportamiento al variar

los parámetros del controlador. La técnica de linealización por retroalimentación fue implementada en RCP [51], un microcontrolador Siemens SAB 80C166 fue programado con el Matlab/Simulink, el autor destaca que el tiempo de implementación del control se reduce drásticamente y además es un hardware más barato que soluciones comerciales.

En [52], emplearon la técnica de control LQG (del inglés Linear Quadratic Gaussian) con Labview y el modelo simulado de un drone usando HIL, del Labview emplearon las herramientas para diseño de controladores (CDSim) y robótica (Robotics), los autores destacan el bajo costo y la posibilidad de probar con diversas técnicas de control.

En [53], presentaron el control LQR realizado por RCP con Matlab/Simulink y la plataforma LEGO mindstorms NXT de la dirección y estabilidad vertical de una bicicleta. Uno de los inconvenientes es que los pocos sensores de la plataforma LEGO no permiten una realimentación de estados completa, por lo que requirieron agregar un estimador de Kalman.

Electrónica de Potencia y Control de Motores Eléctricos

El control de velocidad de un motor de inducción trifásico ha sido presentado en [54]- [55]. Por la técnica de control vectorial en [54], el cual implementaron con Scilab/Scicos y un DSP TMS320F28335 de Texas, usaron la librería X2c [21], los autores reportaron, como ventajas, que el X2c permite el Prototipado rápido y además es posible modificar parámetros de los bloques en línea y la implementación con el X2c para el control vectorial de motores, fue exitosa, siendo éste un software libre podría emplearse en pequeñas empresas. También usaron control vectorial en [56], emplearon como plataforma una FPGA conectada a una tarjeta de adquisición de datos PCI y como herramienta de programación el Matlab/Simulink, un PC con Linux se encarga de realizar la conexión de la tarjeta de adquisición y la tarjeta de la FPGA, que contiene el código generado por el RCP, como ventajas los autores manifiestan que se logran tiempos de muestreo de microsegundos.

Por PWM senoidal [57], usaron una FPGA y una tarjeta dSpace, pero controlaron el módulo de potencia de un motor de 15 fases, concluyen que con el Simulink y el dSpace es posible realizar pruebas simuladas (HIL) reduciendo costos. En un control por modos deslizantes neuronal [55], emplearon Matlab/Simulink y un DSP TMS320F28069 de Texas, una red neuronal recurrente de alto orden es usada para identificar el modelo dinámico del motor, el control se realiza por modos deslizantes y un observador para estimar los flujos magnéticos, los autores concluyeron que el RCP cumple los objetivos de reducir el tiempo de desarrollo, esto demostrado mediante resultados experimentales.

El control de motores DC, empleando RCP con Matlab/Simulink, para el caso de motores con escobillas, fue abordado en [18], [35], [58] y [59]. En [18] y [58], emplearon un DSP de Texas TMS320C32, en resumen concluyen que obtuvieron resultados satisfactorios en procesos educativos empleando la plataforma, ya que minimiza el distanciamiento entre teoría y práctica, también permite que el estudiante se familiarice rápidamente con las técnicas de control empleando RCP. En [59] se usó la plataforma dSpace 1005 y concluyen que el RCP permitió, mediante simulaciones del modelo de aplicación, encontrar errores antes de la compra del hardware. En 2011, Hercog y otros [35] ampliaron su

desarrollo del 2005 para que se pueda usar de manera remota mediante un PC con Labview y conectado a la internet. El modo remoto resuelve la carencia de equipos y los prolongados tiempos de uso por parte de los estudiantes.

Trabajos de RCP en control de motores DC sin escobillas, son por ejemplo: [42] y [60]. Un microcontrolador Stellaris LM3S800 fue programado con Matlab/Simulink [42], en [60] por su parte se usó un DSP TMS320F240 de Texas. En [42] concluyen que lograron la implementación de una librería para el microcontrolador de 32 bits citado y que este es adecuado para el control de motores y aplicaciones de comunicación. En [60] en resumen se concluye que el prototipado reduce el tiempo y esfuerzo en la programación del control de un motor sin escobillas.

Control con RCP y Matlab/Simulink de un servomotor DC, ha sido realizado en 2020 [26] y 1997 [61]. En [26] emplearon un Dspic de microchip 33FJ126MC802 y se concluye que el empleo de los Dspics permite comprobar de manera real, teorías de alto nivel a muy bajo costo. En [61] emplearon la plataforma dSpace DS1102, un control proporcional e integral fue usado y concluyeron que con el RCP los estudiantes tienen más tiempo para enfocarse en la sintonización del controlador y la verificación de las teorías de control sin involucrarse tanto en la programación.

En [39], realizaron con RCP el control “dead-beat” por la técnica de observar el torque y el flujo de un motor síncrono de imanes permanentes (SPMSM del inglés Surface Permanent Magnet Synchronous Motor). Emplearon Matlab/Simulink, pero no especifican el hardware. Una de las conclusiones es que existe un retardo grande de tiempo causado por el cómputo del observador debido al código auto-generado por el Simulink.

Plataformas educativas en electrónica de potencia han sido presentadas en [24], [62]- [63]. En [24] y [62] presentaron el empleo de una plataforma RCP para educación, un motor sin escobillas fue controlado con la plataforma dSpace DS1104 y Matlab/Simulink, los estudiantes aplicaron control PID y espacio de estados. Mediante una encuesta realizada a los estudiantes sobre las herramientas de laboratorio empleadas, midieron el grado de aceptación y comodidad de estas, todos respondieron positivamente. Los autores concluyeron que el empleo de la plataforma probablemente permitirá que los estudiantes se desempeñen satisfactoriamente en temas de control industrial. En [64] emplean Matlab/Simulink, pero el código generado es programado en un DSP de Texas TMS320F28035, concluyen que es una herramienta valiosa para el aprendizaje activo y que provee a los estudiantes un aprendizaje profundo en el diseño de controladores para la electrónica de potencia.

En [65], desarrollaron una interfaz gráfica para la simulación y generación de código en lenguaje 'c' de proyectos en electrónica de potencia, tanto del sistema o modelo como del controlador. La herramienta llamada VTB (del inglés Virtual Test Bed), la cual permite emplear como controlador un computador tipo PC con Linux-RTAI [66] o un DSP [63], su funcionamiento fue probado para un convertidor tipo boost y un inversor monofásico por puente H, los autores emplearon técnicas de control PI y por realimentación en espacio de estados, explican claramente el proceso para lograr instalar el RTAI y exponen las

limitaciones para lograr tiempos de muestreo menores a 300 μ s, que se tienen al emplear un PC.

En [67], realizaron el control de un convertidor DC-DC tipo Buck con una dSpace y Matlab/Simulink, emplearon una técnica de control por modos deslizantes llamada ZAD (del inglés Zero Average Dynamic) y FPIC (del inglés Fixed Point Inducting Control). Los autores reportan frecuencias de muestreo de 6kHz y concluyen que en este tipo de controladores es importante los efectos de la cuantificación, lo cual observaron de forma experimental.

Procesos Continuos

El control de la temperatura es uno de los más implementados con RCP, en [6] para un sistema de calentamiento de aire, en [68] para una secadora de ropa, en [69] de una habitación, [70] del plástico de una extrusora, planta didáctica [71]. La simulación en hardware (HIL) de un sistema de calentamiento de aire fue implementado mediante RCP en 1999 [6], el prototipado rápido de la planta fue implementado en Simulink y corre en una plataforma tipo PC con una tarjeta de adquisición de bajo costo, para el control emplearon un PLC de cuya implementación dan pocos detalles. Los autores concluyen que el empleo de la técnica de HIL es beneficioso en educación en control por el alto costo de las plantas de control de procesos, aunque esta no debe ser exclusivamente usada, recomiendan emplear plantas simples reales en fases iniciales de aprendizaje para evitar que el estudiante interprete incorrectamente el rol de los modelos de simulación.

Una simulación HIL y control implementado en físico de una secadora de ropa por bomba de calor, fue realizado con RCP mediante Matlab/Simulink y Labview [68], como plataforma emplearon un computador tipo PC con Windows programado con una máquina de estados. Los autores concluyen que las herramientas empleadas redujeron inmensamente el tiempo de desarrollo y que el HIL les permite realizar pruebas variando propiedades de los elementos de la secadora, además, que estas pruebas se realizan de manera segura y sin gastos de energía por ser un sistema simulado.

El control de la temperatura de una habitación fue implementado mediante RCP con Matlab/Simulink y la plataforma dSpace en [69], la técnica de control predictivo basado en modelo permite el uso óptimo de la energía empleada para la calefacción/enfriamiento de la habitación. Por último, se expone que las ventajas de emplear RCP son: por un lado, se genera el código de manera automática y por otro, la plataforma dSpace permite visualizar y modificar parámetros del controlador en línea.

En [70], realizaron el control de temperatura de un sistema de extrusión de plástico, inicialmente el sistema es simulado en HIL, donde emplearon un control PID difuso implementado en Matlab/Simulink corriendo en una plataforma tipo PC, para garantizar el tiempo real, el código generado desde Matlab se ejecuta en el PC sobre una plataforma software llamada SIT (del inglés Simulation Interface Toolkit) de National Instruments, se concluyó que su diseño, usando RCP, tuvo excelente desempeño en la extrusora real.

En [71], implementaron un control PID de temperatura a un sistema didáctico, para el RCP emplearon Scilab/Scicos

con RTAI-Lab y un PC de escritorio bajo Linux. Los autores exponen que la herramienta ha sido empleada con éxito por 120 estudiantes y que su solución es de bajo costo comparada con las comerciales.

Un control de nivel y flujos de un sistema de cuatro tanques fue presentado en [72], emplearon Matlab/Simulink y como plataforma HIL un PC, el cual comunicaron por OPC (del inglés Ole for Process Control) con un equipo industrial DCS (del inglés Distributed Control Systems) real. Varias técnicas de control fueron implementadas, por ejemplo: control PID multilazo, control predictivo basado en modelo. Se concluye que en los cuatro años en que han desarrollado experimentos con RCP y HIL, el empleo de un controlador DCS comercial y plantas reales o simuladas complejas ayudan a que la transición de los estudiantes de la academia a la industria no sea tan traumática.

Automatización Industrial

Ejemplos de aplicaciones que usan RCP en control de posición fueron presentadas en [20], [32], [73]- [74]. En [20], generaron el código con Scilab/Scicos y RTAICodeGen (del inglés RTAI code Generator), emplearon un sistema de servo control y controladores PID, difuso y PID con puertos analógicos inalámbricos para las pruebas con un actuador lineal por tornillo, además discutieron los inconvenientes y su solución al implementar el prototipado rápido para una plataforma con microprocesador ARM 920T de 32 bits para un control de movimiento. Una de sus conclusiones fue que el empleo de Scilab/Scicos en RCP podría crecer en un futuro.

En [32], emplearon Labview para control de movimiento de dos ejes, el control es implementado de forma jerárquica: el anfitrión es un PC con Labview encargado de enviar las trayectorias 'xy', y un controlador PXI de National Instruments programado con RCP el encargado de controlar cada uno de los ejes. Concluyen sobre RCP, que el Labview redujo el tiempo de desarrollo gracias a la combinación de simulación y ejecución con los sistemas reales. Otro control multi-eje fue presentado en [73], para el RCP emplearon también Labview y la plataforma Compact RIO de National Instruments, con una técnica para ajustar un controlador mediante aprendizaje iterativo llamada ILC (del inglés iterative learning control), se compara la respuesta contra otras como la técnica de control de acople cruzado CCC (del inglés cross coupling control).

En [75], se empleó Scilab/Scicos para programar un PC con Linux RTAI y una tarjeta de adquisición Computer Measurements PCI-1200 para la identificación y control del modelo de posición de precisión de un actuador lineal para desplazamientos de hasta 1mm, aunque es de bajo costo, requiere dos PC de escritorio y un tiempo para instalar y modificar el sistema operativo Linux, una de las conclusiones es que con pocos pasos, el mismo diagrama de Scicos usado para simulación puede ser transformado en un controlador corriendo en un PC estándar. Un sistema RCP fue presentado en [74], para el control PID y difuso de una máquina de control numérico computarizado CNC, los autores realizaron identificación de la planta y compararon las respuestas del control PID con las de un control difuso, concluyen que el funcionamiento exitoso de la máquina demuestra que se puede emplear RCP en problemas del mundo real.

Implementaciones con RCP vinculadas con autómatas programables o PLC (del inglés Programmable Logic Controller), fueron presentados en [76], [77], [78]. La librería RTW del programa Matlab/Simulink fue usada por [76], escribieron en 'c' los bloques Simulink para implementar Redes de Petri y realizaron pruebas en un sistema de bandas transportadoras, el código RCP es descargado en un PLC software, se expone que el empleo de lenguajes gráficos para RCP conlleva a menos errores.

En [77], desarrollaron una librería a partir de las ecuaciones dinámicas de componentes mecánicos e hidráulicos, la realizaron en texto estructurado para un controlador lógico programable o PLC, a partir de los bloques realizaron primero la simulación HIL y control de un motor sin escobillas, luego realizaron la implementación real. Expresan que también es posible desarrollar los bloques con el paquete de Simulink llamado PLC coder y que se puede utilizar de manera industrial las técnicas de HIL y RCP.

En [78], también emplearon Matlab/Simulink para un sistema de trefilado de alambre en frío, con una estrategia de control de calibre, el código generado es descargado en una tarjeta dSpace DS1006 que se conecta por redes de comunicación Profibus y CAN con un PLC. Se concluye que con RCP el paso de simular al experimento real es mínimo.

Robótica

Ejemplos de aplicaciones en robótica móvil con RCP fueron presentadas en [28], [79] y [80], que emplearon Matlab y Simulink y en [81] que emplea un programa de su propio desarrollo. Un control de trayectorias programadas a un robot móvil de dos ruedas de tracción, fue presentado en [28], emplearon el Simulink para la programación del controlador en un PC, emplearon la estrategia PI para la velocidad de las ruedas, un Arduino Due hace las veces de tarjeta de adquisición de datos, se concluye que, aunque no trabaja en tiempo real, es de bajo costo y suficiente para propósitos educativos. En [79], implementaron un robot móvil con un brazo robótico, para la programación emplearon el Matlab y una plataforma Arduino, el robot se puede dirigir mediante una tableta que se comunica por Bluetooth con el Arduino.

En [80], realizaron un robot móvil con LEGO Mindstorms, el programa encargado del tiempo real y la ejecución de tareas paralelas fue realizado en el lenguaje Ada para la CPU de LEGO, se concluye que en Ada es muy simple implementar los aspectos de tiempo real, también que el tiempo de aprendizaje de los estudiantes se reduce y que la plataforma empleada es de bajo costo. En [81], presentaron una aplicación de RCP para un robot móvil que juega fútbol, para la programación de la interfaz gráfica, que permite el diseño del RCP, emplearon una herramienta propia de su Universidad llamada SIMTool, los programas fueron realizados en un PC con Windows y probados en un simulador de fútbol robótico llamado SimuroSot, el uso de programación en bloques permite diseños rápidos y funcionales, concluye el artículo.

Ejemplos de RCP para robots articulados han sido implementados en [30], [36], [82] y [83]. Un robot simulado de seis ejes usando HIL fue presentado en [30], emplearon Matlab/Simulink y un PC con tarjeta de adquisición como plataforma de descarga del código, un control PI fue

empleado para las articulaciones y una máquina de estados como control supervisor, una de sus conclusiones es que ellos tuvieron que adaptarse de la programación de código en texto a la diagramación de bloques en RCP para el diseño de estrategias de control. En [36], desarrollaron el prototipado rápido para el control articular de dos de las cuatro articulaciones de un brazo robot didáctico, emplearon Matlab/Simulink con la plataforma dSpace y una interfaz gráfica realizada en C++. Este último permite el manejo del robot desde internet tanto en modo manual como en modo automático (ejecución de trayectoria articular programada).

En [82], expusieron un sistema de RCP con Matlab/Simulink y la plataforma dSpace para el control PI articular de un robot paralelo planar fue empleado, tanto el modelo dinámico del robot como el esquema RCP están bien descritos por lo que es posible su replicación, mediante las respuestas al seguimiento de una trayectoria cartesiana, comprobaron el correcto funcionamiento de las técnicas de control. En [83], aplicaron prototipado rápido en control y un script de Matlab para el control angular de un brazo robótico de seis grados de libertad, emplearon una técnica para ajustar un controlador mediante aprendizaje iterativo llamada ILC (del inglés Iterative Learning Control), los autores concluyen que se demostró la eficiencia del Matlab/Simulink para resolver problemas complejos y que el proyecto es una oportunidad para que los estudiantes puedan aprender la implementación real de sistemas que involucren programación híbrida y control por aprendizaje iterativo.

Vehículos de Transporte

Un control de la relación aire y combustible de un motor de combustión fue presentado en 2004 [84], emplearon Matlab y Simulink con los que realizaron el RCP para descargarlo en un microcontrolador Motorola MPC555, como estrategias de control emplearon PI y modos deslizantes. Los autores recomiendan emplear el mismo procesador programado por RCP, en las fases de diseño y producción en masa, porque, de lo contrario, podrían aparecer inconvenientes.

Un control inteligente de un sistema de suspensión activa de un vehículo empleando RCP fue presentado en [85], el control de la suspensión activa mediante amortiguadores que cambian su dureza en tiempo real por efecto de campos magnéticos, la realizaron en Matlab/Simulink y una librería para simulación de la dinámica de vehículos llamada veDYNA, además emplearon la plataforma dSpace DS2102 especial para experimentación en automóviles. Los autores concluyen que el empleo de las técnicas de RCP usando Matlab/Simulink pueden reducir el periodo y el costo de la investigación.

En [86], fue publicado un RCP para el control de un sistema no-lineal consistente en el mecanismo de clutch de un carro, implementaron en Matlab/Simulink un controlador de modos deslizantes dinámico y compararon sus respuestas con un PID y con modos deslizantes convencional. El código generado es descargado a la tarjeta dSpace de punto flotante logrando un tiempo de muestreo de 2ms. Los resultados logrados por los autores muestran un buen desempeño del controlador. El artículo está bien documentado por lo que es posible reproducirlo, aunque el equipo dSpace puede ser algo costoso.

TABLA II
PROGRAMAS DE DISEÑO EMPLEADOS PARA RCP.

Programas empleados	Referencias
Matlab/Simulink	[1], [5], [6], [10], [12], [17], [18], [23], [24] [25], [26], [27], [28], [29], [30], [35], [36] [37], [39], [87], [44], [45], [46], [47], [48] [49], [50], [51], [53], [54], [56], [57], [55] [59], [58], [42], [60], [64], [67], [68], [69] [70], [72], [74], [76], [78], [80], [79], [82] [83], [62], [88], [43], [41], [84], [86], [85], [61]
Scilab/Scicos	[19], [20], [21], [38], [75], [71]
Labview	[31], [32], [33], [34], [52], [73]
Modelica	[40]
Desarrollos propios	[66], [63], [77], [81]

En 2018 [87], usaron RCP con HIL del mecanismo de despliegue de llantas de un avión y del sistema eléctrico de potencia que lo alimenta, un dSpace es programado en Matlab/Simulink con el modelo del mecanismo y con el controlador FOC (del inglés Field Orientated control). Un módulo PXI de National Instruments contiene el HIL del Sistema de eléctrico de Potencia también programado en Matlab/Simulink. Los autores concluyen que el RCP permite realizar múltiples pruebas de sistemas de gran tamaño físico, así como también encontrar estrategias que permitan la regeneración de energía eléctrica de un avión.

La administración de un banco de baterías programada mediante RCP con Matlab/Simulink fue realizada en 2012 [88], emplearon un módulo dSpace especial para automóviles que se comunica mediante el protocolo CAN (del inglés Controller Area Network). El algoritmo se encarga del control de variables como voltaje, temperatura, corrientes, porcentaje de carga. El prototipado rápido también lo emplearon para Hardware en el lazo del modelo. Los autores concluyeron que la técnica de HIL es segura y permite la experimentación sin correr riesgos, al emplear las baterías reales, también les permitió realizar la implementación en un periodo muy corto de tiempo.

Otros

En 2016 Flores y otros [37], crearon un sistema hardware y software al que denominaron prototipado rápido de control avanzado. En el componente de programación, crearon una librería para Matlab/Simulink que permite la multitarea, soporte de ejecución con muestreo variable y sincronización de sistemas de control distribuido entre otras cosas. Como plataforma emplearon el microcontrolador de 32 bits STM32F4 de ST Microelectronics. A partir de una encuesta a 100 estudiantes se obtiene una aceptación en la usabilidad del sistema del 81%, por lo que los autores concluyen que es de gran facilidad de uso y versatilidad de programación, además que el sistema desarrollado les permite emplearlo con grupos multidisciplinarios que requieren formación en control y que este es de bajo costo con respecto a sistemas comerciales.

Bucher y Balemi en 2008 [38], desarrollaron librerías para Matlab y Scilab para el empleo del bus CAN con RCP para el control de diversas plantas. Una de estas plantas fue un dispositivo háptico que sirve para explorar la librería de

realidad virtual de Matlab. Los autores sustentan el empleo del bus CAN en que este permite integrar nuevos sensores y actuadores, dando flexibilidad del sistema de las librerías desarrolladas, además que es similar con las aplicaciones industriales.

Elmqvist y otros en 2009 [40], desarrollaron la librería de sistemas embebidos para el programa Modelica, la cual genera código a partir de diagramas de Modelica. Describen todas las posibilidades y configuraciones de la librería, como por ejemplo la parametrización de multi-tareas. Los autores en sus conclusiones exponen que todavía faltan algunos elementos de la librería por completar y agregar.

Problemas al Emplear RCP

En cuanto a problemas al emplear RCP, Lee y Lorenz en 2017 [39], reportaron algunas respuestas indeseables al emplearlo en el control de velocidad de un motor de imanes permanentes, estos comportamientos aparecen debido a la auto-generación de código desde el diagrama RCP, básicamente por el orden en que los códigos correspondientes a los bloques se ejecutan. Lo anterior fue mejorado gracias a que, desde el Simulink, existe la opción “sorted execution order” que permite cambiar el orden de ejecución de los bloques que conforman el diagrama. Luego mediante un código realizado a mano del mismo controlador, verifican el correcto funcionamiento del código auto-generado con el nuevo orden de ejecución.

IV. DISCUSIÓN Y CONCLUSIONES

Según las referencias consultadas en este documento, en Europa es donde más ha sido empleado el RCP, por ejemplo, la herramienta dSpace, una de las primeras para RCP muy utilizada, fue desarrollada en Alemania. Asia es el segundo continente donde se han publicado proyectos en RCP. Adicionalmente y a partir de la Tabla II la mayoría de aplicaciones son en electrónica de potencia, seguida por aplicaciones en sistemas no lineales que en buena parte son trabajos académicos.

El prototipado rápido en control para software libre (Scilab/Scicos) todavía está en fase de desarrollo y faltan librerías y características de los programas comerciales como Labview o Simulink, pero los trabajos que han sido presentados [19], [21] y [54] son prueba de que es posible emplearlo.

TABLA III
PLATAFORMAS EMPLEADAS PARA RCP.

Plataforma empleada	Referencias
tipo PC	[6], [17], [19], [27], [28], [29], [30], [31] [37], [38], [44], [49], [68], [70], [72], [74], [83], [43], [41], [75], [71], [66], [81]
DSpace	[10], [24], [36],* [87] [50], [57], [59], [67], [69], [78], [82], [62], [88] [86], [85], [61]
DSP	[5], [12], [18], [21], [35] [45], [46], [47] [48], [54], [55], [58], [60], [64], [63]
Microcontrolador	[1], [25], [26], [27], [51], [42], [79] [84]

TABLA IV
NÚMERO DE REFERENCIAS (NR) POR ÁREA DE APLICACIÓN.

Área de aplicación	NR	Referencias
Sistemas no lineales	15	[21], [25], [27], [29], [31], [44], [45], [46], [47], [48], [49], [50], [51], [53], [52]
Electrónica de potencia	22	[10], [18], [24], [26], [34], [35], [39], [54], [56], [57], [55], [59], [58], [42], [60], [64], [67], [62], [66], [63], [65], [61]
Control de procesos continuos	6	[6], [68], [69], [70], [72], [71]
Automatización Industrial	8	[20], [32], [74], [76], [78], [75] [73], [77]
Robótica	8	[28], [30], [36], [80], [79], [82], [83], [81]
Sistemas Eléctricos	5	[5], [17], [19], [43], [41]
Vehículos de Transporte	5	[87], [88], [84], [86], [85]

A partir de las referencias, es posible establecer que el prototipado rápido en control es una herramienta muy útil para la enseñanza de las técnicas de control de sistemas dinámicos simples y complejos, ya que permite que el estudiante se enfoque en las técnicas de control y no en los problemas de la programación en tiempo real y/o del hardware de la plataforma. En algunos pocos casos pueden surgir problemas debido al código auto-generado, como por ejemplo el reportado al usar RCP en electrónica de potencia [39].

Con las FPGA se pueden conseguir tiempos de muestreo del orden de microsegundos [56], con los DSP menos de un milisegundo, el computador personal generalmente está limitado a valores por encima de algunos milisegundos.

V. REFERENCIAS

- [1] H. Hanselmann, U. Kiffmeier, L. Koster, M. Meyer y A. Rukgauer, «Production quality code generation from Simulink block diagrams,» de *Proceedings of the 1999 IEEE International Symposium on Computer Aided Control System Design (Cat. No.99TH8404)*, 1999.
- [2] M. Gurr y R. Mühlaupt, «Rapid Prototyping,» de *Reference Module in Materials Science and Materials Engineering*, Elsevier, 2016.
- [3] J. Carter, M. Xu y W. B. Gardner, «Rapid prototyping of embedded software using selective formalism,» de *16th IEEE International Workshop on Rapid System Prototyping (RSP'05)*, 2005.
- [4] M. Glesner, A. Kirschbaum, F. M. Renner y B. Voss, «State-of-the-art in rapid prototyping for mechatronic systems,» *Mechatronics*, vol. 12, pp. 987-998, 2002.
- [5] K. H. Hong, W. S. Gan, Y. K. Chong, T. F. Cheong y S. H. Tan, «Rapid prototyping of DSP algorithms on VLIW TMS320C6701 DSP,» *Microprocessors and Microsystems*, vol. 26, p. 311-324, 2002.
- [6] W. Grega, «Hardware-in-the-loop simulation and its application in control education,» de *FIE'99 Frontiers in Education. 29th Annual Frontiers in Education Conference. Designing the Future of Science and Engineering Education. Conference Proceedings (IEEE Cat. No.99CH37011)*, 1999.
- [7] R. Grepl, «Real-Time Control Prototyping in MATLAB/Simulink: Review of tools for research and education in mechatronics,» de *2011 IEEE International Conference on Mechatronics*, 2011.
- [8] J. Bélanger, P. Venne y J. N. Paquin, «The what, where, and why of real-time simulation,» *Planet RT*, pp. 37-49, 01 2010.
- [9] H.-J. Herpel, N. Wehn y M. Glesner, «RAMSES-a rapid prototyping environment for embedded control applications,» de *Proceedings The Second International Workshop on Rapid System Prototyping*, 1992.
- [10] H. Mirab, «Rapid Prototyping for Real-Time Systems Design,» *Institution of Electrical Engineers IEE, London*, 1993.
- [11] S. Carabelli, c. Greco y F. Mannino, «MatDSP: a DSP-based Matlab Toolbox for Rapid Prototyping of Digital Control Systems,» de *IFAC*, 1995.
- [12] A. J. Bhatti, S. K. Spurgeon, R. Dorey y C. Edwards, «Rapid prototyping of a sliding mode controller for idle speed control,» de *IEEE Colloquium on System Control Integration and Rapid Prototyping in the Automotive Industry (Digest No. 1997/388)*, 1997.
- [13] I. Sadasiva, F. Flinders y W. Oghannasimo, «A graphical based automatic real time code generator for power electronic control applications,» de *ISIE '97 Proceeding of the IEEE International Symposium on Industrial Electronics*, 1997.
- [14] F. C. Teng, «Real-Time Control Using Matlab Simulink: Windows and Linux Based Systems,» *IFAC Proceedings Volumes*, vol. 33, pp. 199-204, 2000.
- [15] D. Abel y A. Bollig, *Rapid Control Prototyping - Methoden und Anwendungen*, Springer, Ed., Springer, 2006.
- [16] W. Fotso y X. Querol, "Evaluation of a Modeling and Automatic C Code Generation Toolset as an Open Source Alternative Solution", de *Embedded Real Time Software and Systems (ERTS2012)*, Toulouse, France, 2012.
- [17] V. Hölttä, L. Palmroth y L. Eriksson, «Rapid control prototyping tutorial with application examples,» 01 2004.
- [18] H. Darko y J. Karel, «Rapid Control Prototyping using MATLAB/Simulink and a DSP-based Motor Controller,» *Int. J. Engng Ed. Vol. 21, No. 4, pp. 596-605*, 2005.
- [19] D. Radu, D. Petru, M. Trusca, D. Dumitrache y I. V. Sita, «Rapid Control Prototyping Educational Toolbox for Scilab/Scicos,» de *Proceedings of the European Control Conference 2009 • Budapest, Hungary*, 2009.
- [20] G. Skiba, T. Žabiński y A. Božek, «Rapid Control Prototyping with Scilab/Scicos/RTAI for PC-based and ARM-based Platforms,» *Proceedings of the IMCSIT. Volume 3*, 2008.
- [21] G. Grabmair, S. Mayr, M. Hochwallner y M. Aigner, «Model based control design - A free tool-chain,» de *2014 European Control Conference (ECC)*, 2014.
- [22] Mathworks.
- [23] S. A. Zulkifli y A. K. A. Rahman, «Development of single phase active power filter for rapid control prototyping using low cost microcontroller,» de *4th IET Clean Energy and Technology Conference (CEAT 2016)*, 2016.
- [24] R. Ahmed y J. Johnson, «DSP-Based Real-Time Control Systems Design, Analysis, and Implementation For Reinforcement of Controls Education,» *American Society for Engineering Education*, 2006.
- [25] G. Berisch, H. Donath, F. Heinrich, I. Huebener, T. Lipke, T. Mende, M. Schriefer, H. Schulte y M. Zajac, «Design and Development of a Low Cost Rapid Control Prototyping System Applied to an Air Suspension System,» de *Proceedings of the 9th IFAC Symposium Advances in Control Education, The International Federation of Automatic Control Nizhny Novgorod*, 2012.
- [26] R. Delpoux, L. Kerhuel, V. Léchappé y A. Duvernois, «On Chip Rapid Control Prototyping for Education,» de *21st IFAC World Congress (IFAC 2020)*, 2020.
- [27] E. T. Enikov, V. Polyzoey y J. Gill, «Low-Cost Take-Home Experiment on Classical Control Using Matlab/Simulink Real-Time Windows Target,» de *Proceedings of the 2010 American Society for Engineering Education Zone IV Conference*, 2010.
- [28] Y. S. Lee, B. Jo y S. Han, «A Light-Weight Rapid Control Prototyping System Based on Open Source Hardware,» *IEEE Access*, vol. 5, pp. 11118-11130, 2017.
- [29] P. S. Shiakolas y D. Piyabongkarn, «On the Development of a Real-Time Digital Control System Using xPC-Target and a Magnetic Levitation Device,» de *Proceedings of the 40th IEEE Conference on Decision and Control*, 2001.
- [30] C. H. Chen, H.-L. Tsai y J. C. Tu, «Robot control system implementation with rapid control prototyping technique,» de *2004 IEEE International Conference on Robotics and Automation (IEEE Cat. No.04CH37508)*, 2004.
- [31] B. Taylor, P. Eastwood y B. L. Jones, «Development of a Low-cost, Portable Hardware Platform to Support Hands-on Learning in the

- Teaching of Control and Systems Theory,» *Engineering Education*, 9:1, 62-73, DOI: 10.11120/ened.2014.00023, 2014.
- [32] R. Rajagopal, S. Ramamoorthy, L. Wenzel y H. Andrade, «A Rapid Prototyping Tool for Embedded, Real-Time Hierarchical Control Systems,» *EURASIP Journal on Embedded Systems*, 2008.
- [33] S. Buso y T. Caldognetto, «Rapid Prototyping of Digital Controllers for Microgrid Inverters,» *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 3, pp. 440-450, 2015.
- [34] H. Nagy, M. Ruba, H. Hedesiu y C. Martis, «Rapid control prototyping of a speed control strategy for a switched reluctance machine,» de *2016 International Conference and Exposition on Electrical and Power Engineering (EPE)*, 2016.
- [35] D. Hercog, A. Rojko, M. Čurkovič, B. Gergič y K. Jezernik, «Embedded platform for rapid implementation of local and remote motion control experiments,» *Electrical Review*, 2011.
- [36] P. Kielan y G. Kłapyta, «The idea of laboratory stand for rapid prototyping of robot control systems for remote operation via Internet,» de *15th International Workshop on Research and Education in Mechatronics (REM)*, 2014.
- [37] A. F. Caballero, D. S. Copaci, A. Villoslada, D. Blanco y L. M. Lorente, «Sistema Avanzado de Protipado Rápido para Control en la Educación en Ingeniería para grupos Multidisciplinares,» *Revista Iberoamericana de Automática e Informática industrial*, vol. 1, p. 350-362, 2016.
- [38] R. Bucher y S. Balemi, «CAN-bus based rapid control prototyping system for education laboratories,» de *Proceedings of the 17th World Congress The International Federation of Automatic Control Seoul, Korea*, 2008.
- [39] S. Lee y R. D. Lorenz, «Implementing observer-based design methodology for deadbeat-direct torque and flux control with back-EMF self-sensing using rapid control prototyping,» de *2017 IEEE Energy Conversion Congress and Exposition (ECCE)*, 2017.
- [40] H. Elmqvist, M. Otter, D. Henriksson, B. Thiele y S. E. Mattsson, «Modelica for Embedded Systems,» de *Proceedings 7th Modelica Conference*, 2009.
- [41] Q.-C. Zhong, C. Matthews, P.-L. Nguyen y S. Clarke, «Low-cost Rapid Control Prototyping Paradigm,» de *IFAC Conference*, 2010.
- [42] R. Duma, P. Dobra, M. Trusca, D. Petreus y D. Moga, «Towards a Rapid Control Prototyping Toolbox for the Stellaris LM3S8000 Microcontrollers,» de *Proceedings of the 18th World Congress The International Federation of Automatic Control Milano (Italy)*, 2011.
- [43] G. Quaranta y P. Mantegazza, «Using Matlab-Simulink Rtw to Build Real Time Control Applications in User Space With RTAI-LXRT,» de *In Real Time Linux Workshop, Milano-Italy*, 2001.
- [44] W. Grega, K. Kolek y A. Turnau, «Rapid prototyping environment for real-time control education,» de *Proceedings Real-Time Systems Education III*, 1998.
- [45] M. W. Spong, D. J. Block y K. J. Astrom, «The Mechatronics Control Kit for Education and Research,» de *Proceedings of the 2001 IEEE International Conference on Control Applications, Mexico City, Mexico*, 2001.
- [46] N. Semenić, A. Sarjaš, R. Svečko y A. Chowdhury, «Design and Implementation of Rapid Control Prototyping System with Matlab/Simulink,» *Informacije MIDE M*, 2011.
- [47] Z. Fang, N. Song y L. Wang, «Implementing Design and implementation of a novel fuzzy controller with dsp for rotary inverted pendulum,» de *2009 Chinese Control and Decision Conference*, 2009.
- [48] X. M. Chen, X. L. Gong, H. X. Zhou, Z. B. Xu, Y. G. Xu y C. J. Kang, «An Economical Rapid Control Prototyping System Design with Matlab/Simulink and TMS320F2812 DSP,» de *Proceedings of the International MultiConference of Engineers and Computer Scientists Vol II, IMECS 2010*, 2010.
- [49] P. S. Shiakolas y D. Piyabongkarn, «Development of a real-time digital control system with a hardware-in-the-loop magnetic levitation device for reinforcement of controls education,» *IEEE Transactions on Education*, vol. 46, pp. 79-87, Feb 2003.
- [50] C. Busch y S. Lambeck, «Rapid Control Prototyping of a controller for a experimental helicopter set-up used in control engineering courses,» de *2010 IEEE International Conference on Control Applications*, 2010.
- [51] S. Rebeschies, «MIRCOS - microcontroller-based real time control system toolbox for use with Matlab/Simulink,» de *Proceedings of the 1999 IEEE International Symposium on Computer Aided Control System Design (Cat. No.99TH8404)*, 1999.
- [52] K. Benkhoud, S. Bouallègue y M. Ayadi, «Rapid control prototyping of a quad-tilt-wing unmanned aerial vehicle,» de *2017 International Conference on Control, Automation and Diagnosis (ICCAD)*, 2017.
- [53] M. Basso, G. Innocenti y A. Rosa, «Simulink meets Lego: Rapid controller prototyping of a stabilized bicycle model,» de *52nd IEEE Conference on Decision and Control*, 2013.
- [54] N. Lopac, G. Šegon y N. Bulić, «Application of a Model-Based Design Tool X2c in Induction Machine Vector Control,» *Engineering Review*, Vols. %1 de %2Vol. 39, Issue 1, 2019.
- [55] S. Simard, J.-G. Mailloux y R. Beguenane, «Prototyping Advanced Control Systems on FPGA,» *EURASIP Journal on Embedded Systems*, 2009.
- [56] L. Weichao, H. An, G. Shiguang y S. Chi, «Rapid control prototyping of fifteen-phase induction motor drives based on dSPACE,» de *2008 International Conference on Electrical Machines and Systems*, 2008.
- [57] E. Quintero-Manriquez, E. N. Sanchez, R. G. Harley, S. Li y R. A. Felix, «Neural Sliding Mode Control for Induction Motors Using Rapid Prototyping,» *IFAC paper on line*, 2017.
- [58] D. Hercog, M. Čurkovič y K. Jezernik, «DSP Based Rapid Control Prototyping Systems for Engineering Education and Research,» de *Proceedings of the 2006 IEEE Conference on Computer Aided Control Systems Design*, 2006.
- [59] J. Khan, «Rapid Control Prototyping (RCP) solutions for the validation of motor control applications,» de *2016 International Conference on Emerging Technological Trends [ICETT]*, 2016.
- [60] D. Potnuru, K. A. Mary y C. Saibabu, «Design and implementation methodology for rapid control prototyping of closed loop speed control for BLDC motor,» *Journal of Electrical Systems and Information Technology, ScienceDirect*, pp. 99-111, 2018.
- [61] J.-C. Juang y J.-S. Sheu, «Controller Rapid Prototyping and Its Incorporation in Control Education,» de *IFAC Advances in Control Education, Istanbul, Turkey*, 1997.
- [62] R. Ahmed y C. Ramesh, «Laboratory Experimentation and Real-Time Computing: An Integrated Environment,» *American Society for Engineering Education*, 2009.
- [63] S. Choi y M. Saedifard, «An Educational Laboratory for Digital Control and Rapid Prototyping of Power Electronic Circuits,» *IEEE Transactions on Education*, vol. 55, pp. 263-270, 2012.
- [64] A. Monti, E. Santi, R. A. Dougal y M. Riva, «Rapid Prototyping of Digital Controls for Power Electronics,» vol. 18, pp. 915-923, 2003.
- [65] B. Lu, X. Wu, H. Figueroa y A. Monti, «A Low-Cost Real-Time Hardware-in-the-Loop Testing Approach of Power Electronics Controls,» *IEEE Transactions on Industrial Electronics*, vol. 54, pp. 919-931, 2007.
- [66] G. Wang, D. Li, A. Monti y E. Santi, «Virtual Prototyping and Automatic Code Generation for DSP-based Digital Control of Power Electronics Equipment,» de *2006 IEEE Workshops on Computers in Power Electronics*, 2006.
- [67] F. E. Hoyos-Velasco, N. T. Garcia y F. A. García, «Rapid Control Prototyping of a permanent magnet DC motor using non-linear sliding control ZAD and FPIC,» de *2012 IEEE 3rd Latin American Symposium on Circuits and Systems (LASCAS)*, 2012.
- [68] J. Holtkötter, J. Michael, C. Henke, A. Trächtler y others, «Rapid Control Prototyping as part of Model-Based Development of Heat Pump Dryers,» *Procedia Manufacturing, ScienceDirect*, pp. 235-242, 2018.
- [69] c. Lapusan, r. Balan, o. Hancu y c. Rad, «Rapid Control Prototyping in the Development of Home Energy Management Systems,» *Applied Mechanics and Materials*, vol. 659, pp. 395-400, 2014.
- [70] J. Xu, G. Zuo, J. Chen y M. W. Ningbo, «A rapid control prototyping system design for temperature control of plastic extruder based on labview,» de *2011 International Conference on Electronics, Communications and Control (ICECC)*, 2011.
- [71] O. Jaramillo, L. D. Guerrero y J. F. Flórez, «Rapid Control Prototyping with Scilab/Scicos, RTAI-Lab for the mic955 training module of the feedback instruments,» *Revista Iberoamericana de Tecnologías del Aprendizaje ISSN: 1932-8540 ed: IEEE Education Society*, 2018.
- [72] R. Dittmar, «Use of an industrial distributed control system in conjunction with Matlab/Simulink for process control education,» de

Proceedings of the 9th IFAC Symposium Advances in Control Education, 2012.

- [73] W.XU, J.HOU, W. YANG y C. WANG, «A double-iterative learning and cross-coupling control design for high-precision motion control,» *Archives of Electrical Engineering*, 2019.
- [74] R. Bucher, «Rapid Controller Prototyping at the SUPSI Laboratory,» de *Scuola Universitaria Professionale della Svizzera Italiana*, 2006.
- [75] R. H. Guerra, J. Alique, R. H. Haber y P. Villena, «Diseño e implementación rápida de controladores inteligentes,» *Vol. 83, no 8: 459-466 DYNA*, 2008.
- [76] P. Orth, A. Bollig y D. Abel, «Rapid Prototyping of Sequential Controllers with Petri Nets,» de *16th Triennial World Congress IFAC*, 2005.
- [77] L. Simoni, M. Beschi, D. Colombo, A. Visioli y R. Adamini, «A Hardware-In-the-Loop setup for rapid control prototyping of mechatronic systems,» de *2015 IEEE 20th Conference on Emerging Technologies Factory Automation (ETFA)*, 2015.
- [78] M. Wehr, S. Stockert, D. Abel y G. Hirt, «Rapid Control Prototyping in Cold Rolling Using Piezoelectric Actuators,» de *IFAC-PapersOnLine*, 2016.
- [79] A. Soriano, L. Marin, M. Valles, A. Valera y P. Albertos, «Low Cost Platform for Automatic Control Education Based on Open Hardware,» de *Proceedings of the 19th World Congress The International Federation of Automatic Control*, 2014.
- [80] P. J. Bradley, J. A. de la Puente, J. Zamorano y D. Brosnan, «A Platform for Real-Time Control Education with LEGO MINDSTORMS,» de *Proceedings of the 9th IFAC Symposium Advances in Control Education*, 2012.
- [81] J. a. H. S. a. K. H. Jang y C. K. Ahn, «Rapid Control Prototyping for Robot Soccer,» de *Proceedings of the 17th World Congress The International Federation of Automatic Control Seoul, Korea*, 2008.
- [82] C. Lapusan, V. Matis, R. Balan, O. Hancu, S. Stan y R. Lates, «Rapid control prototyping using matlab and dSpace. application for a planar parallel robot,» de *2008 IEEE International Conference on Automation, Quality and Testing, Robotics*, 2008.
- [83] Y. Ye, A. Tayebi y X. Liu, «Rapid-prototyping of iterative learning control using MATLAB/Simlink hybrid-programming,» de *Canadian Conference on Electrical and Computer Engineering*, 2015.
- [84] W. Lee, M. Shin y M. Sunwoo, «Target-identical rapid control prototyping platform for model-based engine control,» *Automobile Engineering*, 2004.
- [85] X. Dong, M. Yu, C. Liao, W. Chen y Z. Li, «Rapid control prototyping development of intelligent control system of vehicle semi-active suspension,» de *2008 7th World Congress on Intelligent Control and Automation*, 2008.
- [86] Y. Zhao, Z. Liu, W. Yang y L. Cai, «Rapid Control Prototyping of an Automated Clutch Using dSPACE and Matlab/Simulink,» de *2010 International Conference on Computing, Control and Industrial Engineering*, 2010.
- [87] R. Subramanian, P. Venhovens y B. P. Keane, «Accelerated design and optimization of battery management systems using HIL simulation and Rapid Control Prototyping,» de *2012 IEEE International Electric Vehicle Conference*, 2012.
- [88] D. Hogan, L. Albiol-Tendillo, B. Kelleher, V. Valdivia-Guerrero y R. Foley, «Rapid-prototyping and hardware-in-the-loop laboratory platform for development and testing of electro-mechanical actuator controls,» de *The Journal of Engineering, IET*, 2018.