

# webPLC: una herramienta para las clases de automatización industrial

J.A. Rodríguez Mondéjar  
Dpto. de Electrónica y Automática ICAI UPCO  
Alberto Aguilera 23, 28015 Madrid  
mondejar@dea.icaei.upco.es

Oscar Neira García  
oscarneira@jazzfree.com

## Resumen

*En este artículo se describe la herramienta webPLC creada en el Departamento de Electrónica y Automática del ICAI (UPCO). webPLC es una aplicación de libre distribución que permite la simulación de autómatas programables, el control remoto de un autómata programable y el control real de una planta mediante un ordenador trabajando como si fuera un autómata. En la práctica, webPLC está demostrando ser una herramienta muy útil para las clases de automatización industrial, como complemento al autómata real. Además, su uso es también muy útil en las clases de informática industrial, porque es un compendio de toda la tecnología web que se puede encontrar en los sistemas industriales más modernos.*

**Palabras clave:** simulación de autómatas programables, control remoto vía Internet, tecnología web.

## 1 INTRODUCCIÓN

El desarrollo de la herramienta *webPLC* surgió de las siguientes necesidades planteadas en las clases de automatización industrial:

- Dificultad del alumno para poder trabajar en casa con el autómata programable. El coste del PLC o el coste de un simulador de PLC con un mínimo de prestaciones es inviable para un alumno. Los programas de libre distribución encontrados para simulación de autómatas fallaban en dos puntos: demasiado simples, o con limitaciones funcionales muy importantes (por ejemplo: no permitir salvar el programa de control).
- Dificultad del alumno para aprender el funcionamiento del ciclo de *scan* del PLC. Aunque la clase se imparta con el autómata real, el alumno tiene muchas dificultades para entender la filosofía del control continuo a partir

de la ejecución de un programa repetidas veces. Una de las razones está en que ningún autómata real permite la ejecución del programa de control paso a paso, o de ciclo en ciclo. Esto es lógico porque lo contrario es ir contra la propia filosofía de funcionamiento del autómata.

A estas dos necesidades hay que añadir que todos nuestros alumnos tienen ordenador en casa, conexión a Internet y que se decantan al 50% entre Windows y Linux.

Con todas estas necesidades y restricciones se planteó el desarrollo de la herramienta *webPLC* a través de un proyecto final de carrera para el curso 2000/2001.

Hoy *webPLC* es una aplicación real probada extensamente en las clases con los alumnos, que está disponible de forma gratuita para cualquier alumno o profesor, y que gran parte de ella (simulación principalmente) se ejecuta sobre un navegador que incorpore una Java Virtual Machine actualizada. Se ha conseguido que el simulador sea independiente de la máquina y del sistema operativo. *webPLC* se puede conseguir a través de las siguientes direcciones [www.dea.icaei.upco/jarm](http://www.dea.icaei.upco/jarm) o [go.to/oscarneira](http://go.to/oscarneira).

El resto de este artículo se organiza en los siguientes apartados: descripción de las funciones de *webPLC*, arquitectura interna de *webPLC*, filosofía del simulador y conclusiones.

## 2 DESCRIPCIÓN DE LAS FUNCIONES DE *webPLC*

*WebPLC* permite:

- **Modo Simulación:** simular el funcionamiento de un PLC. Como PLC base [1] se ha utilizado el IFM314 de la marca Siemens en modo lista de instrucciones y la norma IEC 1131 [2]. No obstante, está preparado para simular PLCs de

otras marcas con unos mínimos cambios. El simulador incluye las siguientes funcionalidades:

- Edición de programas.
- Utilización de instrucciones lógicas, temporizadores, contadores, etc.
- Depuración de programas amigable a nivel de sintaxis.
- Uso de símbolos en los programas.
- Simulación del funcionamiento de la planta mediante botones.
- Posibilidad de conectar con la planta a través del puerto paralelo.
- Posibilidad de mostrar la interfaz en varios idiomas.
- Simulación paso a paso de la planta.
- Simulación ciclo a ciclo.
- Batería de ejemplos integrados en la interfaz.

En la figura [1] se muestra el estilo de la interfaz del simulador con un ejemplo sobre un control de semáforo para paso de peatones. La figura muestra como el simulador admite símbolos, permite visualizar el resultado lógico (o aritmético) de las operaciones, etc.

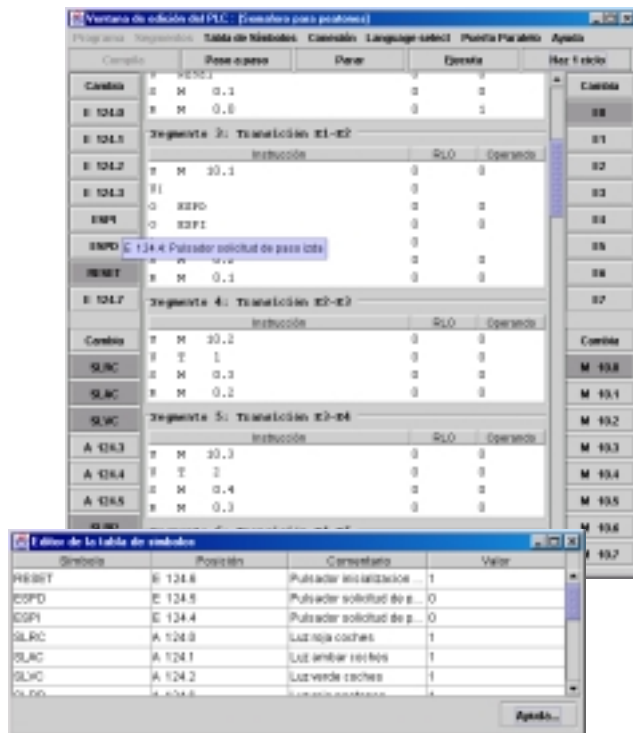


Figura 1 Interfaz de webPLC

El simulador es independiente del sistema operativo. Se puede ejecutar tanto como aplicación applet o como aplicación java en cualquier ordenador con un visualizador para internet o una JVM. Esto permite llegar tanto a la comunidad Windows como a la Linux.

- **Modo PLC.** Permite utilizar un ordenador personal como un PLC real. Básicamente es un PLC construido a partir de Linux con Real-Time Linux. Real-Time Linux se encarga de lanzar de forma periódica la tarea de control. Por eficiencia de tiempo, el programa de control en JAVA es traducido a C y de C a ensamblador. Por supuesto, la función de watch-dog está incorporada al PLC [3].
- **Modo conexión remota.** Permite controlar ese ordenador que funciona como PLC real de forma remota a través de Internet de manera integrada con el simulador: sobre el propio programa de control se indican los resultados reales. Es posible también ver el estado del ordenador que funciona como PLC a través de una página WEB o WAP.

### 3 ARQUITECTURA INTERNA DE webPLC

En la figura [2] se muestra la arquitectura interna de webPLC.

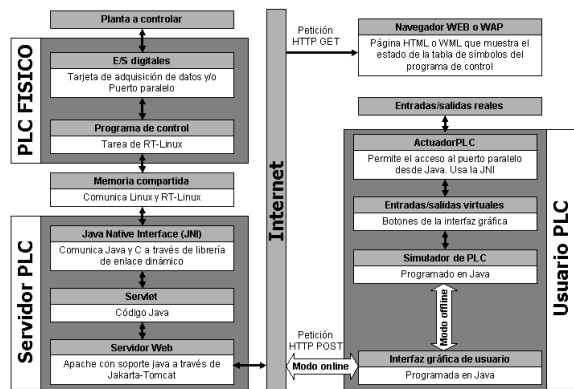


Figura 2 Arquitectura interna de webPLC

Desde el punto de vista físico webPLC tiene dos componente principales:

- Ordenador de control. Es el ordenador que controla la planta real y que por tanto está trabajando como un PLC.
- Ordenadores remotos. Son los ordenadores que permiten acceder a través de Internet al ordenador de control para programar y ver el estado del PLC, o que simplemente simulan el comportamiento del PLC real.

Desde el punto de vista lógico webPLC se organiza en tres grandes bloques:

- PLC Físico

- ServidorPLC
- UsuarioPLC

PLCFísico y ServidorPLC corren en el ordenador de control. PLCFísico es realmente el PLC y ServidorPLC permite el acceso a los datos del PLCFísico a través de Internet.

El UsuarioPLC es la interfaz de webPLC con el usuario que permite introducir el programa de control, simularlo y en su caso, cargarlo en el ordenador de control, a través de ServidorPLC. Tiene dos modos de funcionamiento:

- **Modo offline:** el UsuarioPLC permite sólo la edición y simulación del programa de control en el simulador de PLC que lleva integrado. En este modo no hay conexión con el ServidorPLC. Las entradas/salidas en este modo están simuladas como botones de la interfaz gráfica. Opcionalmente, se puede usar el módulo ActuadorPLC para que parte de esas entradas/salidas virtuales se conviertan en reales usando el puerto paralelo del propio ordenador.
- **Modo online:** el UsuarioPLC se conecta a través de Internet con el Servidor PLC. El programa de control se ejecuta en el PLCFísico. En este modo el UsuarioPLC puede programar el PLCFísico, mostrar el estado del PLCFísico o cambiar posiciones de memoria remotamente. Hay dos tipos de acceso: modo administrador y modo usuario. El primero permite la carga de programas y modificación de sus variables; el segundo, sólo la visualización.

El *PLCFísico* está constituido por un canal de entrada/salida de datos digital (a través de tarjeta de adquisición de datos y/o del puerto paralelo), el programa de control en lenguaje máquina (tarea de control), y Real-Time Linux como sistema operativo que permite la ejecución controlada de la tarea de control, y la ejecución de Linux como una tarea más.

El *ServidorPLC* está programado en Java y funciona sobre el Linux del ordenador de control. El núcleo principal de este bloque es el servlet, un pequeño programa en Java que funciona dentro de un servidor web y cuya función es gestionar todas las conexiones a través de Internet. Para comunicar el servlet con el PLCFísico se emplea una zona de memoria compartida, a la que se accede a través de la Java Native interface (JNI).

El UsuarioPLC está programado en JAVA, y se puede ejecutar como una aplicación independiente o mediante applet.

Para dar una idea de la complejidad del proyecto se da, a continuación, una relación de las herramientas utilizadas: Java SDK, Java Run-Time (JRE), Forte for Java Community Edition, Kdevelop, Sistema operativo Linux, Real-time Linux, Apache, Jakarta-Tomcat, Microsoft Visual C++, GCC.

#### 4 Filosofía del simulador

En el simulador del PLC se ha buscado la simplicidad y la flexibilidad para futuras ampliaciones. Para ello el programa de control se ha asimilado a un vector, donde cada elemento es una instrucción. Cada ciclo de scan consiste en recorrer los elementos del vector ejecutándolos, tal como muestra la figura [3].

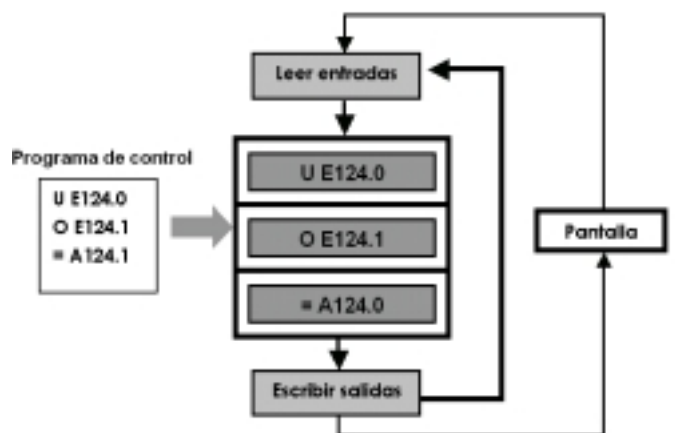


Figura 3 Vector de instrucciones

Esta idea se ha podido implantar fácilmente en JAVA gracias a su capacidad para soportar programación orientadas a objetos:

- Primero: se ha definido la interfaz de los objetos que van a contener las instrucciones (los llamados objeto-instrucción). En esta interfaz está la capacidad de ser ejecutada, de ser leída, de ser escrita, de ser depurada, o de ser traducida a programa en C. Por supuesto, en el PLCFísico corre una versión del vector de instrucciones que ha sido previamente traducida a C por eficiencia.
- Segundo: se definen objetos concretos para instrucciones concretas.
- Tercero: el editor del webPLC construye el vector de objeto-instrucciones una vez que el usuario ha escrito las instrucciones de su programa de control.

Esta filosofía se ha ampliado a instrucciones de salto incorporando a los objeto-instrucciones la capacidad de devolver el índice del siguiente elemento a ser ejecutado en el vector.

Esta organización del simulador, junto con toda la arquitectura de comunicaciones, servidores webs,

zonas de memoria común, etc., hace que werPLC sea un ejemplo muy válido para las clases de Informática Industrial. Su simplicidad conceptual permite al alumno modificarlo para añadirle nuevas opciones.

## 5 Conclusiones

En este artículo se ha presentado la herramienta webPLC que permite a los alumnos probar en su casa sus programas para PLC. Para aquellos alumnos que quieran, pueden construirse una tarjeta muy simple para el puerto paralelo de su ordenador personal, que les permite interactuar como el mundo real desde el simulador de PLC. Además del simulador, la herramienta permite disponer de un PLC real trabajando sobre un ordenador personal y de acceder a dicho PLC desde Internet.

También es útil para las clases de Informática Industrial porque es un compendio de técnicas modernas para la programación de un sistema de control industrial complejo. De hecho, la herramienta se suministra a los alumnos en un cd-rom con todas las aplicaciones que se han utilizado en su desarrollo.

Por último, para completar la herramienta se está trabajando en la versión *webPlanta* que permitirá ver el efecto del programa de control sobre una planta virtual.

## Referencias

- [1] Josep Balcells, José Luis Romeral (1997) Marcombo.
- [2] R. W. Lewis (1998), Programming Industrial Control Systems using IEC 1131-3.
- [3] Robert S. Carrow, Soft Logic (1998) Mc Graw-Hill.

Información sobre Java:

<http://java.sun.com>  
<http://www.javasoft.com>

Información sobre Linux:

<http://www.kernel.org>  
<http://www.linuxdoc.org>

Programación de puertos de E/S en linux:

<http://www.linuxdoc.org/HOWTO/mini/IO-Port-Programming.html>

Programación del puerto paralelo en windows:

<ftp://ftp.rmii.com/pub2/hisys/parport>  
<http://www.doc.ic.ac.uk/~ih/doc/par/index.html>  
[http://www.hut.fi/Misc/Electronics/circuits/parallel\\_output.html](http://www.hut.fi/Misc/Electronics/circuits/parallel_output.html)

<http://www.lvr.com/jansfaq.htm>  
<http://comunidad.ciudad.com.ar/internacional/aruba/megatronica/nuevo2.htm>

Tarjeta de adquisición de datos:

<http://www.ni.com>

Apache:

<http://httpd.apache.org>

Jakarta-Tomcat:

<http://jakarta.apache.org>

Soluciones en tiempo real bajo Linux:

<http://www.rt-linux.org>  
<http://www.rtai.org>  
<http://www.rtlinux.com>

Recursos sobre WAP:

<http://www.wmlclub.com>  
<http://www.phone.com>  
<http://forum.nokia.com>  
<http://www.ericsson.com/developerszone/index.asp>