

Plataforma de Enseñanza de Laboratorio para Educación a Distancia en Ingeniería

V. Antonelli y A. Custodio

Universidad Nacional Experimental Politécnica “Antonio José de Sucre”, Vicerrectorado Puerto Ordaz, Centro de Instrumentación y Control, Puerto Ordaz, Venezuela, Email: cicunexpo@gmail.com

RESUMEN

Se desarrolló un simulador para las prácticas de laboratorio de la asignatura Instrumentación Industrial específicamente para la práctica de circuitos neumáticos. El simulador consta de un Software desarrollado en plataforma Linux con la herramienta de desarrollo de interfaces gráficas GAMBAS 2.5 y un hardware que tiene como cerebro un Microcontrolador PIC 18F4550. Este Hardware está encargado de simular el desplazamiento de un pistón, así como también los fines de carreras y el sentido en que se desplazara. Para la interacción Software/Hardware se configuró un protocolo de conexión USB, por lo que tuvo que programarse un socket para manejar un driver, que hace posible esta conexión. Este sistema posee las características de poder ser operado de forma remota a través de Internet, ya que se creó una página Web que permitirá el acceso por medio de la red al simulador desarrollado.

Palabras claves: Simulador, Laboratorio Virtual, Educación

ABSTRACT

We developed a simulator for the laboratory practices of the Industrial Instrumentation course specifically for the practice of pneumatic circuits. Simulator consists of a software developed on Linux platform development tool for graphical interfaces PRAWNS 2.5 and hardware that has the brain a PIC 18F4550 microcontroller. This hardware is in charge of simulating the movement of a piston, as well as for racing and the sense that were displaced. For the interaction Software / Hardware is configured a USB connection protocol, so it had to be programmed to handle a socket driver that enables this connection. This system has the features to be operated remotely via the Internet because we created a Web page that will allow access through the network simulator developed.

Keywords: Simulator, Virtual Lab, Education

1. INTRODUCCIÓN

Uno de los retos que en el presente siglo se ha planteado en el ámbito educacional es el desarrollo e implementación de estrategias y medios de enseñanza dirigidos a desarrollar habilidades y actitudes perfeccionadas en los estudiantes, a fin de que adquieran competencias que les permitan estar funcionalmente activos en lo profesional, lo que demanda del alumno el ejercicio del principio de compartir los recursos y conocimientos de que dispongan, a través de la práctica de aprendizajes colaborativos. En ese sentido la vinculación de la educación con la tecnología ha ampliado las oportunidades para transformar y mejorar los procesos enseñanza y aprendizaje. En la enseñanza de la ingeniería, especialmente en el área de laboratorios, el problema de la rapidez del cambio tecnológico adquiere especial relevancia ya que no todas las universidades pueden ir a la par con dichos avances. El alto costo de los equipos sigue siendo una limitación, especialmente en los países subdesarrollados. Una solución a este problema es emplear en los laboratorios técnicas de enseñanza y aprendizaje basadas en computadoras personales, en los cuales se reemplacen equipos convencionales por computadoras, instrumentos virtuales y sistemas de adquisición de datos, que permitan a los estudiantes hacer adquisición, procesamiento y control de señales físicas en tiempo real a costos menores.

El uso masivo del Internet por parte de los estudiantes y la necesidad de una herramienta la cual permita a los docentes tener un medio de enseñanza a distancia, permitió idear un entorno el cual permite al estudiante mantenerse al tanto de todo lo ocurrido en clase sin necesidad de estar presente físicamente (el software permite un aprendizaje presencial remoto). Ya Acevedo presentó un laboratorio para prácticas neumáticas controlado a distancia (Acevedo y Custodio, 2006), Moreno un sistema de control hidráulico (Moreno y Custodio, 2006) y Garcés un sistema de control de flujo de fluidos (Garcés y Custodio, 2006). Todos consistían en maquetas didácticas ubicadas en el edificio de ingeniería mecánica, y controladas a distancia desde el edificio de ingeniería electrónica. Bajo el esquema del software libre Sánchez presentó un sistema de control y supervisión vía internet de una maqueta de caudal (Sánchez y Custodio, 2006).

En este proyecto se desarrolla una maqueta didáctica para control de pistones neumáticos, que será controlada desde la intranet para realizar prácticas de laboratorio de instrumentación industrial, y usando software libre. Con esta herramienta de enseñanza a distancia se podrá incluir toda la información acerca de las clases, horarios de exámenes, foro de discusión. Permitiendo al docente tener un control total de las actividades realizadas en este entorno y hacer más flexible el intercambio de conocimientos con el alumno.

Para ello se seguirá como metodología el diseño y construcción de la maqueta. Luego el diseño de la interfaz de adquisición de datos. El desarrollo del software de control y supervisión. La conectividad desde la internet con la maqueta. Y finalmente el gestor del curso.

2. DISEÑO

Para el desarrollo del laboratorio virtual se determinó el diseño de un software, en plataforma Linux, específicamente en la distribución Ubuntu, desarrollado en lenguaje de programación Gambas, que toma los datos, usando USB (Universal Serial Bus), desde un hardware que también fue desarrollado a lo largo de este proyecto. Este hardware simula ser una maqueta con la que se realizan las prácticas de laboratorio, específicamente la experiencia de “Circuitos Neumáticos”.

El diagrama de bloque del proyecto está en la figura 1.

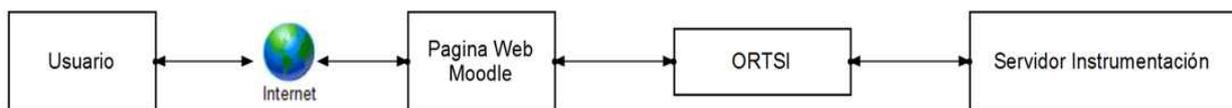


Figura1: Diagrama en Bloques del Proyecto

Consta de una comunicación vía Internet la cual permite al usuario el control y la visualización del simulador de manera remota a través de Moodle, a su vez la página es gestionada por la ORTSI (Unidad regional de tecnología, sistemas e información) que realiza la comunicación entre el simulador y Moodle.

El servidor de Instrumentación está conformado por una interfaz gráfica, la interfaz de comunicación USB (Driver) y un hardware. Como se puede observar en la figura Fig. 2.

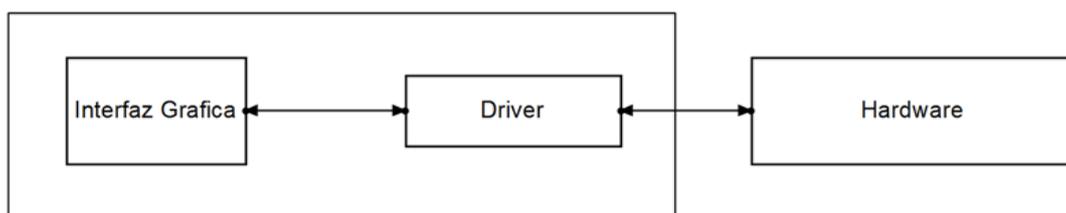


Figura 2 Partes del Servidor de Instrumentación

2.1 SOFTWARE

El software desarrollado lleva por nombre Inst_Virtual y fue confeccionado en lenguaje de programación Gambas. En la figura 3 se muestra el diagrama de flujo de la pantalla del simulador.

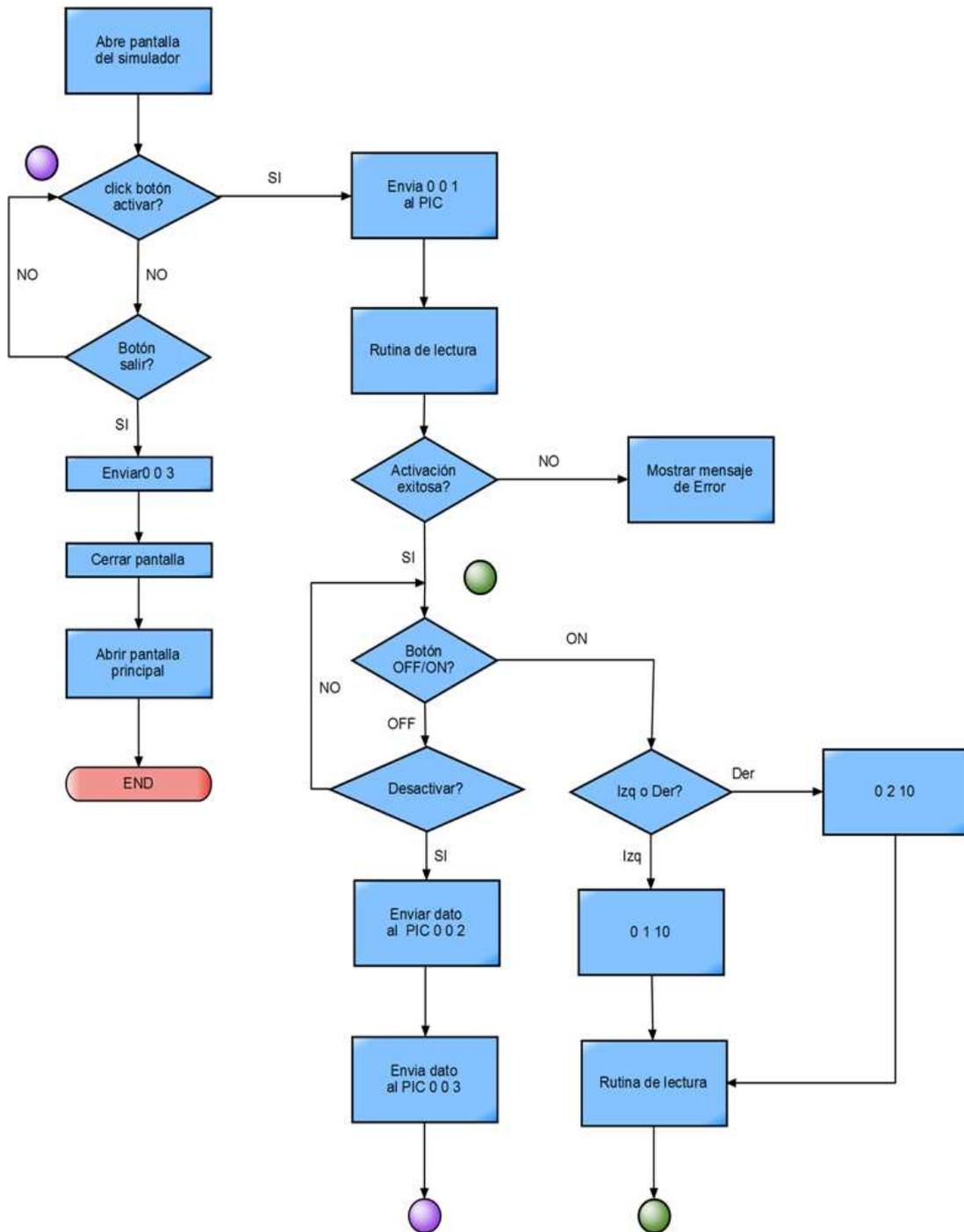


Figura 3: Diagrama de flujo de la pantalla del simulador con gambas

El proyecto está estructurado en dos partes fundamentales: Comunicación con el Hardware, e Interfaz Gráfica (donde el usuario interactúa con el simulador).

Para elaborar el software, en primer lugar se debió establecer la comunicación entre el hardware (basado en un PIC 18f4550) y la interfaz gráfica (software). Para crear la comunicación Hardware-Software, se realizó un programa en lenguaje C, con un editor de texto avanzado, en este caso Kate. Cabe destacar que los editores de textos brindados por Ubuntu tienen la particularidad de poder reconocer varios tipos de lenguajes de programación. En este editor se puede realizar el código requerido y luego compilarlo en consola utilizando las librerías adecuada (gcc, g++...). El programa, elaborado en lenguaje C, permite la conexión USB con el hardware.

La interfaz gráfica de usuario (GUI) del software Inst_Virtual, está conformada por un conjunto de pantallas que fueron diseñadas con la finalidad hacer de las prácticas del Laboratorio de Instrumentación Industrial una experiencia versátil, cómoda y a la vez innovadora. Esta interfaz fue desarrollada con la herramienta de programación gráfica Gambas, bajo plataforma Linux.

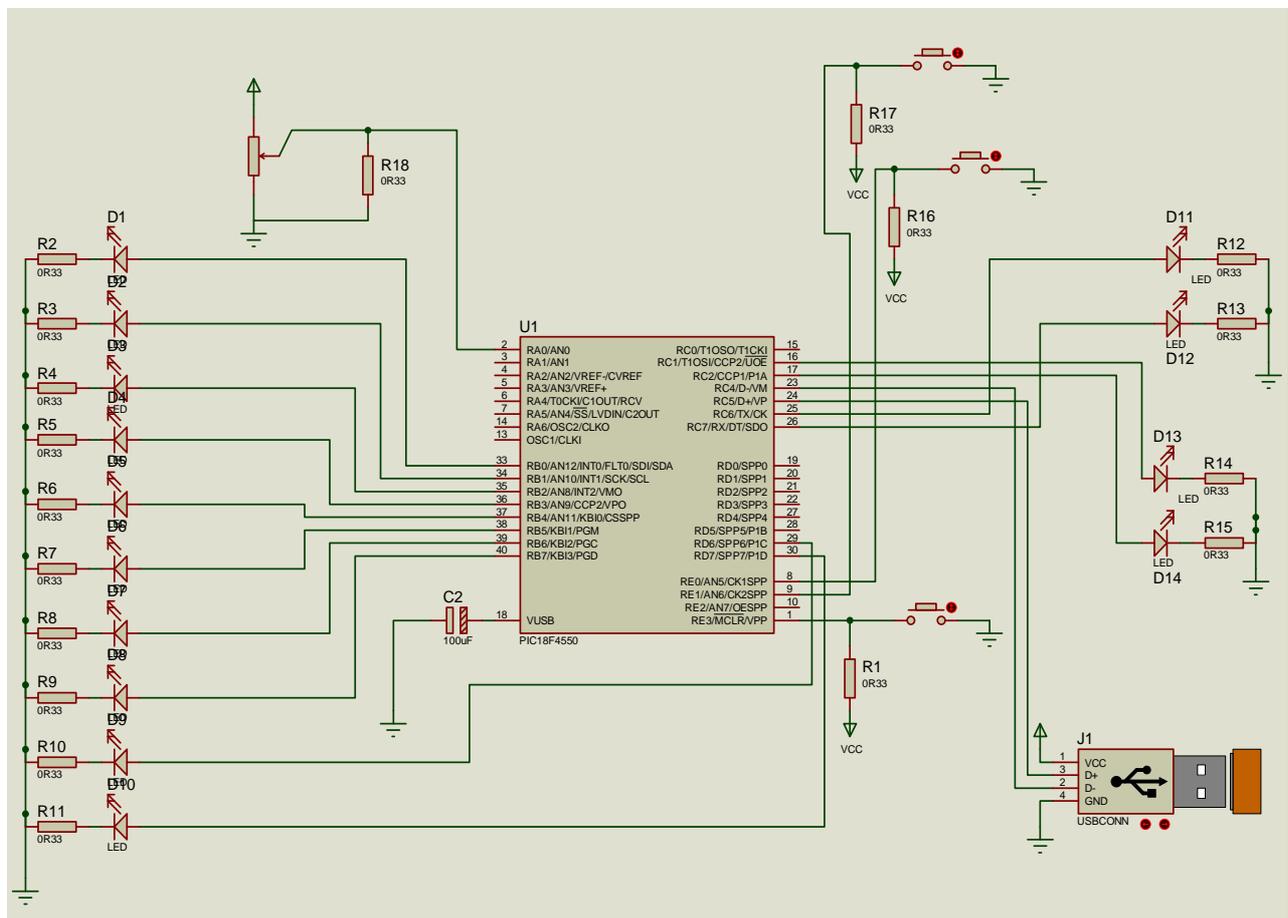


Figura 4: Hardware del simulador

3. RESULTADOS EXPERIMENTALES

En la figura 5 se muestra el hardware instalado.

El hardware posee la capacidad de enviar a la computadora, mediante el protocolo USB, un conjunto determinado de señales para su respectivo procesamiento. Con el uso de la PC se le otorga al usuario un entorno mucho más amigable a la hora de interactuar con el prototipo en cuestión. Una vez enviado los datos a la computadora el usuario podrá visualizar los valores emitidos desde el microcontrolador. La interfaz gráfica presente en la computadora fue configurada en LINUX UBUNTU, con la ayuda del paquete de compilación llamado GAMBAS. En la figura Fig. 6 se muestra el hardware conectado al computador.

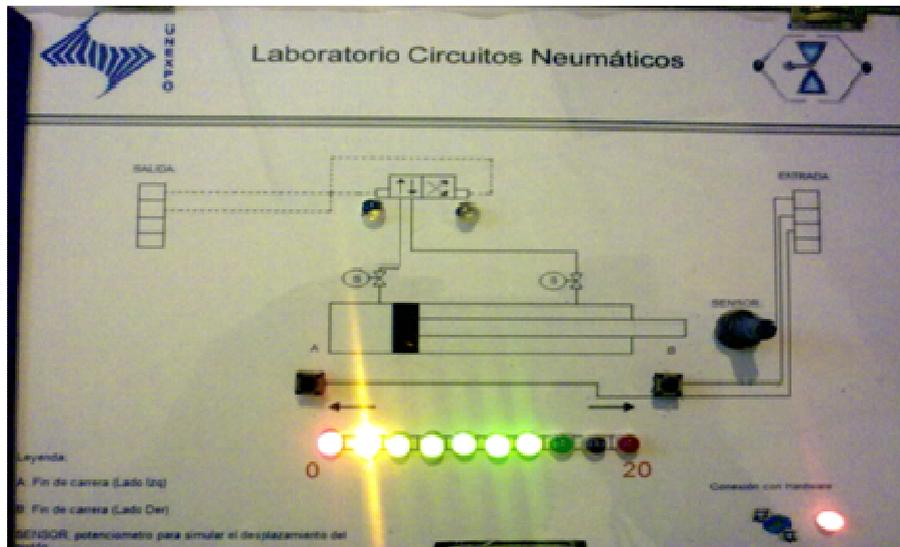


Figura 5: Hardware implementado (parte exterior)

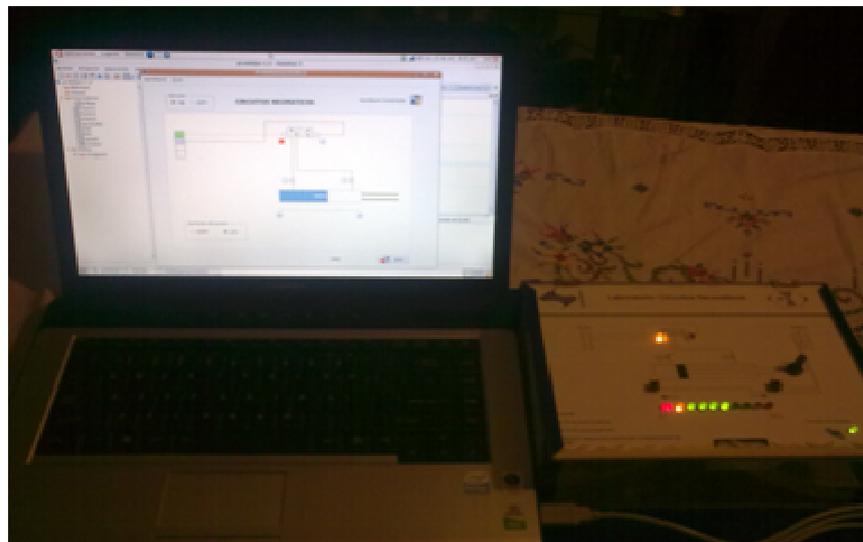


Figura 6: Conexión física hardware/PC

Ahora se describe el proceso de interacción de software y el hardware desarrollado. Cuando se haya accedido a la pantalla del simulador y se quiera establecer la conexión de software y hardware, se debe ir a la sección “Hardware”, lo que despliega un pequeño menú en donde se muestran dos opciones: “Activar” y “Desactivar”. Al

hacer clic en “Activar” comienza el proceso de comunicación entre software y hardware, el sistema indica si la comunicación se logró de distintas maneras: en el software se habilitan los selectores en el recuadro “Simular” y el mensaje de “Hardware Desconectado” cambia a “Hardware Conectado”; mientras que en el hardware se puede visualizar que se estableció la conexión de forma correcta a través de un Led que cambia de luz roja a verde al conectarse.

Habilitados los selectores del recuadro simulador, se elige el selector “ON” que permite el inicio de la simulación, después de esto son habilitados los selectores de desplazamiento del pistón (“Der” y “Izq”). Es importante destacar que el selector de desplazamiento a la derecha “Der” es el que se encontrará por defecto elegido, así como también en el hardware el Led verde que indica el desplazamiento a la derecha.

En el software se encienden unos indicadores del sentido del desplazamiento del pistón. Estos indicadores están ubicados en la parte superior central del esquema de cada lado de las válvulas. Como se muestra en la figura Fig. 7 y 8.

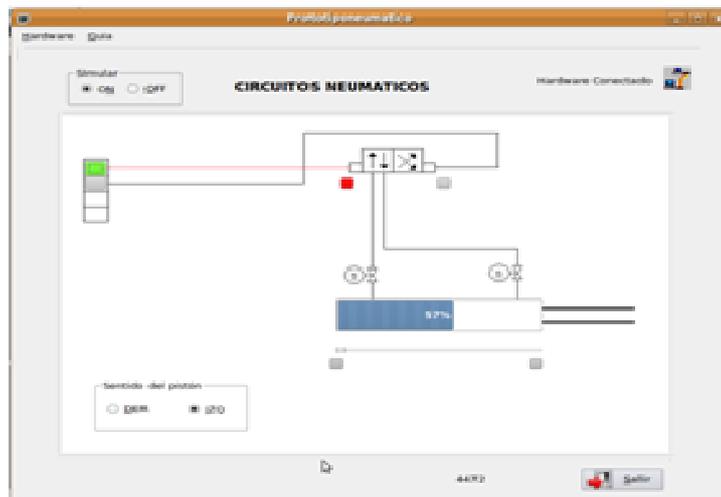


Figura 7: Indicación de dirección izquierda del pistón.

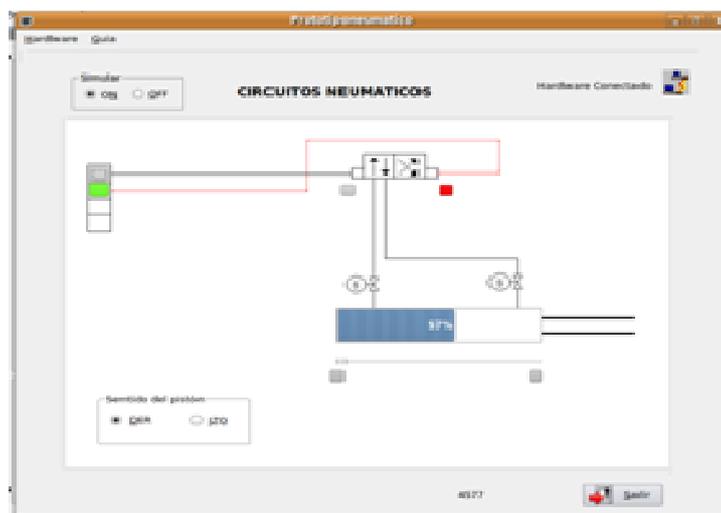


Figura 8: Indicación de dirección derecha del pistón

En el hardware existen dos pulsadores que simulan los fines de carrera del pistón, al presionar cualquier de los dos pulsadores en el hardware (Fig. 7 u 8), en la figura se encuentran resaltados en color rojo), se enciende en el software su indicador correspondiente, estos indicadores se encuentran en el centro de esquema, a los lados de la barra que representa el pistón (Fig. 9).

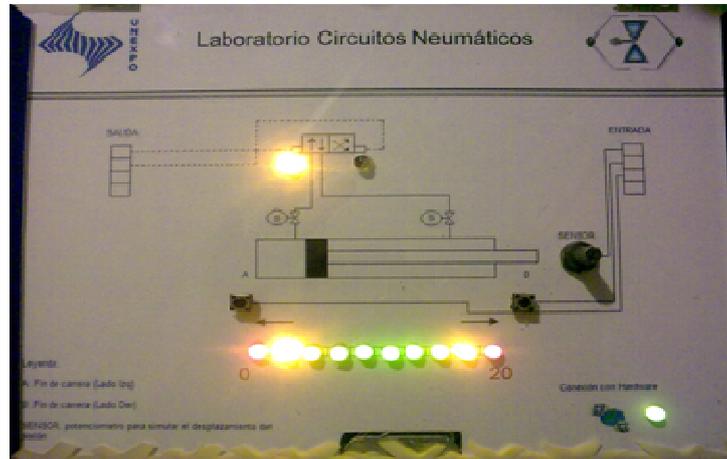


Figura 9: Pulsadores activados (fines de carrera)

Cuando el pistón se desplaza se indica con un tren de leds ubicados en la parte inferior de la maqueta. Esto mismo se muestra en el simulador de forma simultánea (figuras 10 y 11).

Una vez realizada la experiencia de manera satisfactoria y se quiera cerrar la simulación se debe seleccionar “OFF” en el recuadro “Simular”, luego para terminar la conexión del software con el hardware en la sección de “Hardware” de la pantalla del simulador se oprime la opción “Desactivar”. Para abandonar completamente el programa se pulsa “Salir” en la parte inferior derecha de la pantalla.

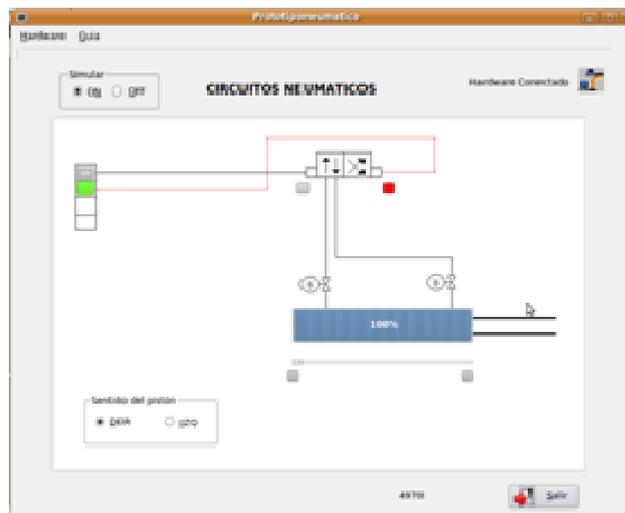


Figura 10: Simulador mostrando el desplazamiento del pistón.

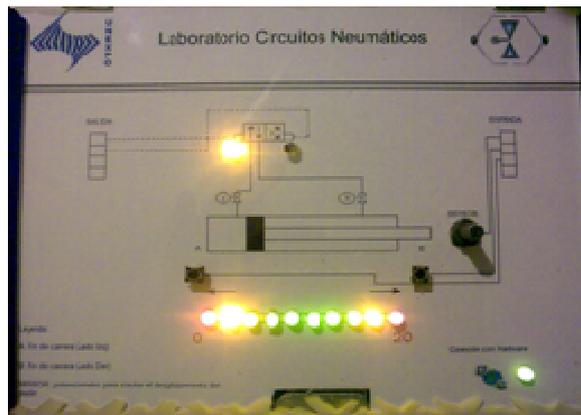


Figura 11: Maqueta mostrando el desplazamiento del pistón de la figura 10

4. CONCLUSIONES

Se diseñó una interfaz gráfica de usuario en plataforma Linux utilizando la herramienta de desarrollo de software Gambas 2.5. Este interfaz resulta ser muy sencilla, didáctica e intuitiva permitiendo facilidad en su utilización por parte de los nuevos usuarios.

Para poder establecer la comunicación entre el software desarrollado y el hardware de adquisición se creó un programa interfaz, compilado en lenguaje C, que permite la conexión USB. Esta comunicación se realiza siguiendo un protocolo ideado también en el transcurso del proyecto.

Con la finalidad de probar la comunicación USB entre el software y el hardware se hicieron distintas pruebas de envío y recepción de datos entre estos dos elementos, al igual que una vez conectado con el servidor a través de Internet, estas pruebas resultaron exitosas.

Una vez culminado el prototipo de software y de hardware se programaron pruebas de aplicación para verificar su buen funcionamiento.

Se realizó exitosamente el control remoto del simulador a través de un software (VNC VIEWER) que permite conectarse del servidor a un navegador Web.

REFERENCIAS

- Acevedo, M. y Custodio, A. (2006). “Desarrollo de un algoritmo de control remoto para un sistema de control neumático”. Trabajo de Grado, UNEXPO Puerto Ordaz, Venezuela.
- Moreno, G. y Custodio, A. (2006). “Desarrollo de un algoritmo de control remoto para un sistema de control hidráulico”. Trabajo de Grado, UNEXPO Puerto Ordaz, Venezuela.
- Garcés, M. y Custodio, A. (2006). “Desarrollo de un algoritmo de control remoto para un sistema de flujo de fluidos”. Trabajo de Grado, UNEXPO Puerto Ordaz, Venezuela.
- Sánchez, G. y Custodio, A. (2008). “Desarrollo de un sistema SCADA para el control de caudal basado en Linux”. Universidad Ciencia y Tecnología, Vol. 11, No. 44, pp 121-128.

Autorización y Renuncia

Los autores autorizan a LACCEI para publicar el escrito en las memorias de la conferencia. LACCEI o los editores no son responsables ni por el contenido ni por las implicaciones de lo que esta expresado en el escrito.