



Ingeniería inversa de pizarra electrónica de béisbol, desarrollo de nuevas propuestas y aplicaciones

Carlos Bazán-Prieto, Adrian Acuña-Pérez, Rigoberto Acosta-González

RESUMEN / ABSTRACT

Las pizarras electrónicas se utilizan en los eventos deportivos como medios de control y visualización de la información. En la mayoría de los casos, estos equipos electrónicos se encuentran regulados por las federaciones deportivas internacionales correspondientes. En Cuba los principales estadios de béisbol, disponen de una pizarra electrónica profesional. Estas pizarras necesitan mantenimiento y reparación, sin embargo no existe la suficiente información del fabricante. Igualmente, el software facilitado por el fabricante para el control de estas pizarras no satisface las necesidades de los operadores. En este trabajo se realiza la ingeniería inversa a los distintos módulos de visualización de la pizarra, así como a los protocolos de comunicaciones. Como resultados, se dispone de los esquemas electrónicos, documentación del funcionamiento, así como de los protocolos de comunicaciones. Con estos resultados, se desarrollaron herramientas para pruebas de los módulos de visualización por separado y un nuevo software de control de la pizarra ajustado a las necesidades de los operadores. Adicionalmente se propone una mejor estructura y protocolo, con su respectiva aplicación de software, que reduce la complejidad de interconexión entre los módulos de visualización de la pizarra.

Palabras claves: Pizarras electrónicas, Protocolos de comunicación, Ingeniería inversa.

Electronic scoreboard is used in sports events as means of control and visualization of information. In most cases, these electronic devices are regulated by the corresponding international sports federations. In Cuba, the main baseball stadiums have a professional electronic scoreboard. These scoreboards need maintenance and repair, but there is not enough information from the manufacturer. Likewise, the software provided by the manufacturer for the control of these electronic scoreboards does not satisfy the needs of the operators. In this work, the reverse engineering of the different modules of the scoreboard is carried out, as well as the communication protocols. As results, there are available the electronic schemes, documentation of the operation, as well as the communication protocols. With these results, tools were developed for testing the visualization modules separately and a new control software for the scoreboard adjusted to the needs of the operators. Additionally, a better structure and protocol is proposed, with its respective software application, which reduces the complexity of interconnection between the visualization modules of the scoreboard.

Key words: Electronic scoreboards, Communication protocols, Reverse engineering.

Reverse engineering of baseball electronic scoreboard, development of new proposals and applications

1. -INTRODUCCIÓN

Los eventos deportivos, necesitan sistemas capaces de controlar y mostrar la información durante el transcurso de la actividad: tiempo, marcador, equipos, etc. Incluso las federaciones deportivas internacionales exigen determinados parámetros para la realización de los eventos oficiales internacionales y nacionales, ya que contribuyen al evento deportivo y a facilitar el trabajo de los árbitros. En la actualidad, las pizarras electrónicas son muy utilizadas en los eventos deportivos ya que facilitan a los deportistas, entrenadores, anotadores, árbitros y espectadores en general, seguir el desarrollo del evento de una manera más clara, mezclando el deporte, la comodidad y la tecnología.

El béisbol es considerado uno de los deportes más populares en el mundo. Los estadios de béisbol utilizan distintos tipos de pizarras electrónicas provenientes de empresas dedicadas a la comercialización, instalación y reparación [1-4]. El béisbol en Cuba es el deporte nacional, con un amplio esquema de eventos hasta la Serie Nacional de Béisbol. Existen estadios de béisbol diseminados por todo el país, entre ellos uno principal en cada ciudad cabecera provincial. En estos últimos se utilizan pizarras electrónicas. La mayoría de estas pizarras electrónicas son de fabricación vietnamita [5]. Estas pizarras están formadas por múltiples módulos electrónicos interconectados en una red con un computador que los controla.

La Empresa Productora y Comercializadora de Artículos Deportivos (EPCAD) es la encargada del mantenimiento y reparación de las pizarras electrónicas de todas las instalaciones deportivas del país, en especial de las referidas al béisbol. Actualmente esta empresa está interesada en perfeccionar su trabajo, por lo que firmó un proyecto de colaboración con la Universidad Central “Marta Abreu” de Las Villas (UCLV), para realizar las investigaciones pertinentes.

En el caso específico de las pizarras electrónicas para el béisbol no se dispone de la información suficiente del hardware, software y protocolos de comunicaciones, que permitan realizar una adecuada reparación y mantenimiento. Por otra parte, los operadores de estas pizarras que utilizan la aplicación de software brindada por el fabricante, señalan algunas limitaciones:

- proceso de instalación engorroso,
- no se puede seleccionar el puerto de comunicación serie a utilizar (solo para el COM1),
- casi todas las funciones están limitadas al uso del mouse,
- mucha demora desde que se modifica un dato hasta que se visualiza en la pizarra,
- código cerrado que impide futuras mejoras.

Los objetivos de este trabajo son, realizar la ingeniería inversa a los módulos electrónicos de la pizarra de béisbol y a los protocolos de comunicaciones. Desarrollar aplicaciones para las pruebas independientes de los módulos y para el control de la pizarra electrónica de béisbol existente. Finalmente, proponer una mejor estructura y protocolo, con sus respectivas aplicaciones de software.

2.- MÓDULOS Y PROTOCOLOS

Para realizar la ingeniería inversa al hardware de la pizarra de béisbol, se parte de la limitada información disponible en EPCAD, varias hojas de datos de componentes y un minucioso trabajo de identificación de las conexiones.

A partir de la inspección visual e identificación de las conexiones de las pizarras electrónicas existentes en los estadios de béisbol cubanos, se pudo identificar la estructura que se muestra en la figura 1. Esta estructura está formada por una PC (microcomputadora) que, con un convertidor USB/RS485 [6], se conecta al dispositivo concentrador y distribuidor (HUB). Este HUB se comunica con cada uno de los 43 módulos de visualización con los que cuenta esta pizarra, numerados del 1 al 43 en la figura 1.

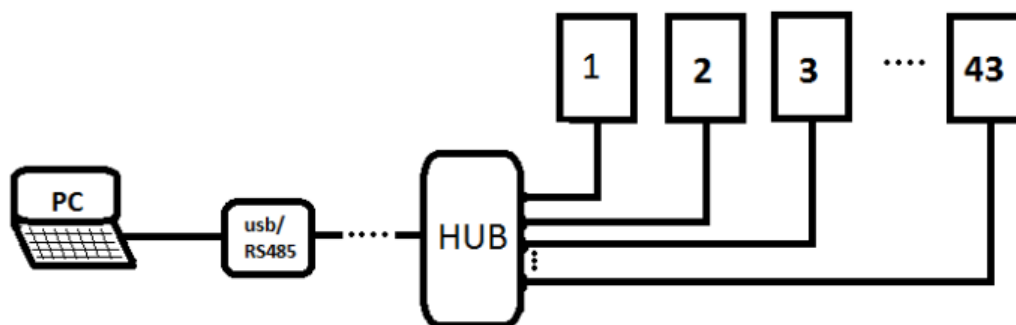


Figura 1

Estructura de las pizarras electrónicas existentes en los estadios de béisbol cubanos.

Los 43 módulos de visualización de la pizarra, están divididos en 5 tipos según sus funciones, que se describen a continuación:

- Módulo tipo A, para visualizar un solo dígito numérico de 7 segmentos (valores del 0 al 9).
- Módulo tipo B, matriz para visualizar letras y números.
- Módulo tipo C, para visualizar un dígito numérico de 7 segmentos más un 1 (valores del 0 al 19).
- Módulo tipo D, para visualizar un solo dígito numérico de 7 segmentos más dos puntos, utilizado como separador de la hora y los minutos.
- Módulo tipo HE, módulo igual al tipo A, pero solo visualiza H (Hit) o E (Error).

Todos los módulos de visualización, incluido el HUB, están basados en el microcontrolador AT89S52 de la familia MCS-51 [7]. El HUB y cada módulo de visualización utilizan como interfaz para las comunicaciones el circuito integrado MAX487 [8]. Para la visualización utilizan LED (*light-emitting diode*) de alta intensidad de brillo. Como resultado de la ingeniería inversa de los módulos de visualización y el HUB, se dispone de los esquemas electrónicos y de conexiones.

En cuanto a los protocolos de comunicaciones, resta identificar el protocolo de comunicaciones que se emplea para la comunicación entre la PC y el HUB (PC-HUB), y el protocolo que se emplea entre el HUB y los módulos de visualización (HUB-módulos).

La capa física, de los protocolos que se emplean en la pizarra electrónica de béisbol, se puede identificar a partir del circuito integrado MAX487 que se utiliza como interfaz para las comunicaciones. Este se corresponde con el estándar de comunicaciones RS485 [9] y por lo tanto con todas las características de este estándar. Todas las conexiones con el HUB, presentadas en la figura 1, se realizan con el estándar de comunicaciones RS485.

Al instalar el convertidor USB/RS485 en la PC (figura 1), esta le asigna un recurso de puerto serie para su funcionamiento, sobre el cual trabaja la aplicación. El método para identificar el protocolo de comunicaciones consistió en utilizar un rastreador y analizador de puerto serie, en este caso SerialMon [10], para identificar los datos transmitidos, y un osciloscopio digital del tipo RIGOL [11] para corroborar y desentrañar el funcionamiento del protocolo de comunicaciones.

Se realizaron varios experimentos para capturar tramas en la comunicación PC-HUB, a partir de datos conocidos que fueron generados a través del programa de control de la pizarra electrónica brindado por el fabricante. Un ejemplo de trama capturada, con el rastreador y analizador de puerto serie y el osciloscopio, se presentan en las figuras 2 y 3 respectivamente.

```
■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ 02 44 76 63 6C 69 6E 6 RTS on .Dvclin
64 3C 44 33 20 31 32 3A 3F 43 20 20 20 20 20 34 d<D3 12:??C 4
37 20 20 20 20 20 20 20 20 20 20 20 30 20 30 30 7 0 00
20 30 30 30 30 30 03 ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ 00000 .Baud rate
■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ ■■ 2400 RTS off DT
```

Figura 2

Captura de trama por puerto serie, con el rastreador y analizador de puerto serie.

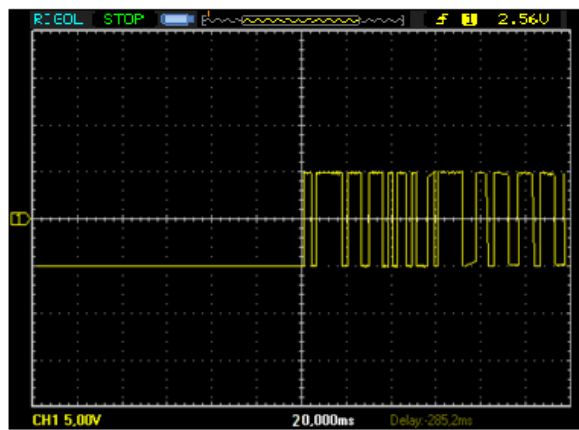


Figura 3

Captura de trama con el osciloscopio.

Como resultado de estos experimentos se observaron regularidades como: bytes de inicio, byte de fin y bytes correspondientes a los datos enviados. Las observaciones permitieron establecer la estructura de la trama, que se presenta en la figura 4 y se analiza a continuación:

- El 1er byte corresponde al byte de inicio de texto según el código ASCII (*American Standard Code for Information Interchange*) (02H).
- El 2do byte corresponde al carácter D en ASCII (44H).
- En los bytes del 3 al 45 se envían los datos con la información de cada uno de los 43 módulos de visualización.
- Para los módulos tipo C, que muestran información del 0 al 19, se envía para representar los números del 0 al 9 los caracteres correspondientes en ASCII, y del 10 al 19 los caracteres reservados (: ; < = > ? @ A B C).
- El último byte corresponde al byte de fin de texto según el código ASCII (03H).

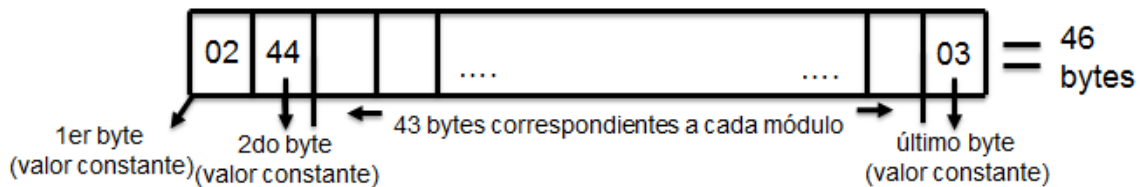


Figura 4
Estructura de la trama PC-HUB.

El programa que controla la pizarra (según la versión original del fabricante), envía consecutivamente dos tramas de 46 bytes con la estructura de la figura 4. La primera trama indica los valores a visualizar en cada uno de los módulos. La segunda trama contiene la información de intensidad de brillo para cada módulo de visualización. Para esto se envía en el espacio reservado en la trama, para cada uno de los módulos de visualización, los valores de 80, 86, 90, 96 o FF en hexadecimal correspondientes a los valores de intensidad de brillo del 1 al 5. Este proceso se repite cada 6 segundos aproximadamente, para actualizar el dato a visualizar y la intensidad de brillo correspondiente a cada uno de los módulos de visualización. El refrescamiento constante provoca una demora adicional que influye en el funcionamiento eficiente de la pizarra. En este sentido, los operadores señalan demoras innecesarias al cambiar el valor a visualizar en los módulos usados con mayor frecuencia, como los correspondientes a mostrar las bolas o strikes.

La estructura de la trama identificada en este trabajo, tiene una estructura parecida al protocolo DMX512 (*Digital Multiplex*) [12]. DMX512 es un protocolo utilizado en el control de la iluminación de espectáculos, permitiendo la comunicación entre los equipos de control de luces y las propias fuentes de luz. Los canales son transmitidos secuencialmente en un formato serie asincrónico, comenzando por el canal 0, que contiene el código de inicio y terminando con el último canal implementado. Ambas tramas son simplemente canales multiplexados en tiempo, con delimitadores o marcas de inicio y fin de trama, que carecen de un mecanismo de detección y corrección de errores.

Igualmente para descubrir la trama de comunicación HUB-módulos, se siguió similar metodología. Con el osciloscopio digital se capturaron los datos enviados y como resultado se obtuvo la trama mostrada en la figura 5.

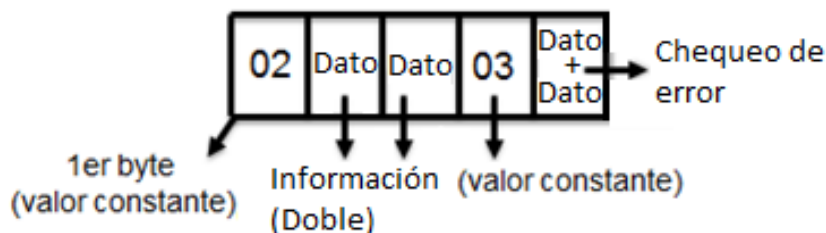


Figura 5
Estructura de la trama HUB-módulos.

El HUB envía a cada módulo de visualización la información que le corresponde de la siguiente manera: comienza con 02H como byte delimitador de inicio de texto según el código ASCII, posteriormente dos byte correspondientes al dato a

transmitir de manera doble, o sea, repite este byte dos veces, luego 03H como byte delimitador de fin de texto según el código ASCII. Finalmente la trama tiene un byte de chequeo de error, el cual contiene la suma del byte de dato con él mismo en hexadecimal. De la misma manera ocurre con la información de intensidad de brillo para cada módulo de visualización.

Esta estructura de trama es la misma para cada módulo de visualización de la pizarra, cambiando solo el dato a transmitir, y el chequeo de error. Si el módulo recibe un error producto al ruido o un cambio de información en el HUB, este se queda mostrando la información previamente visualizada, y envía un byte de respuesta, ERROR (00 en hexadecimal) indicando que se recibió una trama errónea. Si la trama se recibe sin errores envía un byte de respuesta correspondiendo a un ACK (*acknowledgement*, FF en hexadecimal), que indica que la información fue visualizada con éxito. Cuando la información llega de manera incorrecta al módulo de visualización, se procede a la retransmisión de la información.

Como resultado de la ingeniería inversa de los protocolos de comunicaciones, se dispone de la estructura de los protocolos PC-HUB y HUB-módulos. Estos protocolos no se corresponden con otros protocolos bien conocidos [13, 14] sino que poseen una estructura propia para esta aplicación. Los mecanismos de control de error que utilizan son básicos. Con esta información disponible, se pueden programar nuevas aplicaciones. Se pudo observar que el funcionamiento de la pizarra depende en gran medida del HUB, el cual cuenta con una gran complejidad, por lo que si falla se afectaría la pizarra íntegramente.

3.- DESARROLLO DE APLICACIONES PARA ESTRUCTURA Y PROTOCOLOS EXISTENTES

Actualmente para la reparación y mantenimiento de los módulos de visualización de las pizarras electrónicas de béisbol, estos tienen que estar conectados al HUB, y utilizar la aplicación de software brindada por el fabricante. Esto implica trabajar dentro de la pizarra con toda o casi toda la pizarra en funcionamiento. En este trabajo se toman los resultados de la ingeniería inversa al hardware y protocolos, para desarrollar una aplicación con el fin de probar los módulos de visualización de manera independiente al resto de la pizarra.

En la figura 6 se presenta la aplicación realizada en LabVIEW, para probar los módulos de visualización de la pizarra. Esta herramienta utiliza el protocolo HUB-módulos, lo que resulta una potente herramienta de prueba para los técnicos durante el mantenimiento y reparación de los módulos de visualización, ya que no necesitan trabajar directamente con el HUB ni el resto de la pizarra. Basta con conectar un módulo a una PC, utilizando un convertidor USB/RS485. La transmisión de la información se realiza con una razón de 2400 bauds, 8 bits de datos, 1 bit de parada y no paridad, igual a la que utiliza el HUB.



Figura 6
Aplicación para pruebas de módulos.

Las características la aplicación de la figura 6 son las siguientes:

- Ventana Conexión: abre el puerto serie para la posterior transmisión de información, además muestra la hora, minutos y fecha actual de la computadora. Permite seleccionar el puerto serie asignado al convertidor USB/RS485 en la PC.

correspondiente o la letra S, B o O del teclado. Los *Outs* se actualizan de manera automática al presionar 3 veces *Strike*. Los *Strike* y las *Bolas* se ponen en cero al presionar cambio de bateador (Cambio BT).

El refrescamiento constante se ha modificado, de forma que los tiempos de demora en la visualización sean menores. Sin embargo, como se mantiene la estructura original (figura 1), no se logra reducir la complejidad del cableado y se mantiene el HUB. Lo más importante es que ahora se dispone de una aplicación, con algunas mejoras, y del código fuente que permite futuras modificaciones.

4.- DESARROLLO DE NUEVA ESTRUCTURA Y PROTOCOLO

Del análisis del funcionamiento de la pizarra original y sus protocolos, se pudo observar la dependencia en gran medida del HUB. Este HUB es de elevada complejidad y la estructura de la pizarra (figura 1) lleva un complejo cableado. El protocolo original, hecho a la medida para la pizarra electrónica, no tiene un adecuado mecanismo de control de error. Por lo que en este trabajo se propone utilizar un protocolo capaz de simplificar la estructura de la pizarra electrónica y el cableado, a la misma vez que sea capaz de controlar los errores. Los protocolos de bus de campo [13, 14] pueden resolver estos problemas. Modbus [15], aunque no sea uno de los buses de campo más avanzados en la actualidad, posee una dirección única para cada dispositivo de la red. Esto permitiría prescindir del HUB y reducir la complejidad del cableado. También dispone de un mecanismo de control de error para brindar confiabilidad a las comunicaciones y es un bus de campo abierto con una amplia documentación disponible. Además, su comunicación es serial y puede trabajar sobre el estándar RS-485.

Modbus permite el control de una red de dispositivos y comunicar los resultados a un ordenador. Cada dispositivo de la red Modbus posee una dirección única, por lo que en la pizarra de béisbol cada módulo tendrá una dirección diferente. Cualquier dispositivo puede enviar mensajes, aunque lo habitual es permitirlo sólo al dispositivo maestro.

Estos mensajes son conocidos como tramas y están constituidos por un conjunto de caracteres que tienen una longitud en bits que depende del modo de transmisión utilizado. La longitud de las tramas es variable y está acotada a un máximo de 256 caracteres, pero su estructura siempre es la misma y no depende del modo de comunicación a emplear. La figura 8 muestra el formato de las tramas en Modbus.

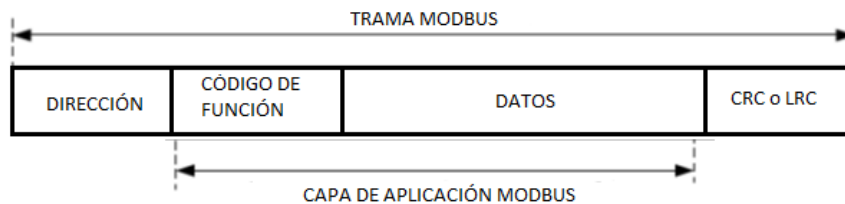


Figura 8
Estructura de la trama en Modbus.

Cada trama Modbus contiene la dirección del dispositivo destinatario. Todos los dispositivos reciben la trama, pero sólo el destinatario la ejecuta. Cada una de las tramas incluye información redundante que asegura su integridad en la recepción. Los comandos básicos Modbus (código de función), de la capa de aplicación, permiten controlar un dispositivo conectado a la red para modificar el valor de alguno de sus registros o bien solicitar el contenido de dichos registros [15].

En Modbus se usan dos tipos de métodos de control de errores: códigos de redundancia cíclica (CRC) o Chequeo de Redundancia Longitudinal (LRC, *longitudinal Redundancy Check*). El campo de chequeo de errores (CRC o LCR) permite a los dispositivos maestros y esclavos detectar errores en la transmisión de mensajes. Debido al ruido eléctrico o a las interferencias de otra naturaleza, se puede producir algún error en las tramas durante la transmisión. El control de errores asegura que los dispositivos receptores no ejecuten alguna acción incorrecta debido a una modificación en el mensaje original. El contenido del campo de chequeo de errores depende del modo de transmisión usado.

En esta propuesta, se implementa el modo de transmisión serial ASCII de Modbus, debido a que este tiene mejor funcionamiento en dispositivos de baja velocidad de procesamiento y una implementación más sencilla. En este caso el modo es ASCII a 7 bits, el inicio de trama es con el carácter ":" y el fin de trama es el LRC. En el modo ASCII, el campo de chequeo de errores contiene dos caracteres ASCII. El valor de este campo es el resultado del cálculo de LRC, aplicado al contenido del mensaje, excluyendo los caracteres delimitadores de trama.

Con la implementación del protocolo Modbus en la pizarra electrónica de béisbol, se logra simplificar la estructura original mostrada en la figura 1. Solo serán necesarios pequeños cambios:

- Agregar un puerto de comunicación RJ45 a la placa de control de cada módulo. La tarjeta que controla actualmente a cada módulo de visualización, cuenta con la conexión no utilizada para agregar este puerto.
- Desarrollar un programa de control de los módulos de visualización, basado en Modbus y con una dirección única para cada uno de los 43 módulos de visualización.
- El HUB será eliminado, por lo que el cableado quedará simplificado a interconectar el conector RJ45 de salida de un módulo de visualización con el conector RJ45 de entrada del siguiente módulo de visualización y finalmente una resistencia terminadora R_t , como se muestra en la figura 9. La estructura propuesta es un bus RS485 con todos los módulos de visualización conectados en paralelo.

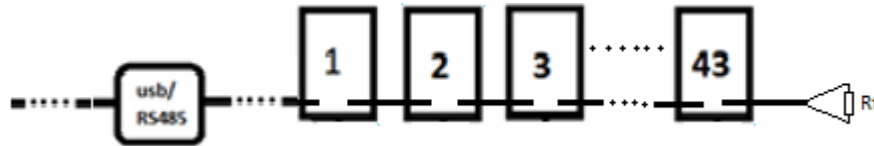


Figura 9

Nueva estructura de conexiones entre módulos.

La propuesta del protocolo de comunicación Modbus en modo ASCII, con el chequeo de errores, permite que la nueva aplicación realice los reenvíos de los datos erróneos, de manera que garantice la seguridad de la información transmitida e indique al operador la existencia del error mientras persista. Como cada módulo contará con una dirección, para mostrar datos en la pizarra no es necesario actualizar todos los módulos sino solamente los modificados. Esto permite mayor rapidez en la actualización de la pizarra.

El protocolo propuesto permite eliminar el HUB, simplificar el cableado y en general la estructura, además garantiza el control de error. Por ello se puede decir que esta propuesta es mejor que la estructura original y sus protocolos. Para comprobar lo anterior, se propone una nueva aplicación diseñada en LabVIEW para el control de la pizarra electrónica de béisbol, utilizando el protocolo Modbus. Con esta aplicación se podrán actualizar los datos inmediatamente después de ser modificados. La transmisión de la información se realizará a través del puerto serie con similar configuración que la aplicación original y el convertidor USB/RS485 conectado a la PC, con la posibilidad de seleccionar el puerto serie.

Con esta nueva estructura de la pizarra electrónica de béisbol (figura 9), no es necesaria una herramienta de prueba de módulos de visualización, ya que cada módulo de visualización es independiente, están conectados en paralelo y cuentan con una dirección propia. Por lo que se puede trabajar desde la aplicación con un solo módulo sin la necesidad de estar conectado al resto de la pizarra, facilitando el mantenimiento y reparación de los módulos de visualización. En la figura 10 se muestra la nueva aplicación que utiliza el protocolo Modbus.



Figura 10

Nueva aplicación con protocolo Modbus.

La distribución de las direcciones para cada módulo de visualización, se realiza según la estructura original del HUB. O sea, la dirección 01 corresponde a la primera letra del equipo visitador y la 43 al de contador de *Outs* (las direcciones 01 a 43 se refiere a los módulos de visualización de la figura 1). Como cada módulo de visualización cuenta con una dirección única, la estructura de la trama es diferente en cada uno de estos. Por ello es necesario realizar el envío mediante un botón, que se ha añadido a cada controlador, como se observa en la figura 10. O sea, en lugar de enviar periódicamente la información de todos los módulos de visualización (como hace la aplicación brindada por el fabricante), se envía solamente la trama correspondiente al módulo de visualización que se quiere modificar. Esto disminuye el tiempo de actualización de la pizarra electrónica. Para el caso de la intensidad de brillo se envía una trama de multidifusión con esta información, la cual se interpreta en todos los módulos. El indicador de error en la transmisión, alerta al operador sobre la existencia de error en la comunicación con los módulos de visualización. Este indicador muestra el número de los módulos de visualización con error en la transmisión (separados por comas) y lo actualizará cuando el mecanismo de control de error logre el reenvío exitoso. El resto de las funciones son similares a las aplicaciones anteriores.

La estructura de la trama está conformada con un byte de encabezado “:” (3A en caracteres ASCII), seguido de dos bytes de dirección, un byte de código de función, y los bytes de datos que pueden variar en función de la trama a enviar, concluyendo con un byte de chequeo de error LRC. Por ejemplo, para enviar el dato 0002 al módulo con dirección 05, se envía la trama de información “:050600000002F3”.

Para probar la funcionalidad de esta propuesta se conectaron varios módulos de visualización de diferentes tipos con la estructura propuesta (figura 9), con resultados satisfactorios. Resta implementarlo en una pizarra electrónica de béisbol, con el apoyo de EPCAD.

Se debe señalar que ahora el punto débil de esta nueva propuesta radica en la posible falla de un cable de comunicaciones, que puede limitar la información de algunos módulos de visualización. Esta se debe evitar con el mantenimiento periódico del cableado.

5.- CONCLUSIONES

Con la realización de este trabajo se facilita la reparación y mantenimiento de las pizarras electrónicas de béisbol de todo el país. Esto disminuye la importación de nuevos equipos para las pizarras.

A través de la ingeniería inversa realizada a la pizarra de béisbol se dispone de toda la documentación actualizada a nivel de hardware, así como de los protocolos de comunicación que intercambia la PC-HUB y el HUB-módulos. Se desarrolló una aplicación para la prueba de los módulos por separado, que utiliza el protocolo del HUB-módulos, para facilitar la reparación y mantenimiento. También se elaboró una nueva aplicación para el control de la pizarra, basado en el protocolo identificado, para satisfacer las limitaciones señaladas por los operadores de las pizarras y disponer de un código propio, que posibilite futuras modificaciones.

Se propone el protocolo de comunicaciones Modbus, con la interfaz de comunicaciones RS-485, como una mejor solución para la pizarra electrónica de béisbol. Esta propuesta, elimina el HUB y reduce la complejidad del cableado, con pequeños cambios. Además, el control de error de Modbus garantiza la seguridad de la información transmitida. Igualmente se desarrolló una nueva aplicación, que utiliza el protocolo de comunicaciones Modbus, para el control de la visualización en la pizarra de béisbol.

REFERENCIAS

1. *Scoreboard Enterprises, Baseball and Softball Scoring and Timing Systems. Mansfield, Massachusetts (USA). 2018.*
2. *All American Scoreboards, Outdoor Scoring Solutions, Pardeeville, Wisconsin (USA). 2018.*
3. *Zeit. Tecnología en Iluminación Publicitaria. Guadalupe, N.L. México. 2016.*
4. *Daktronics. Baseball Scoreboards, Brookings, South Dakota (USA). 2018.*
5. *CÔNG TY CỔ PHẦN ĐẦU TƯ THÁI BÌNH.* Disponible en: <http://thaibinhco.com.vn>
6. *Future Technology Devices International Ltd, USB to RS485 Serial Converter Cable. FTDI Chip. 2010.*
7. *Atmel Corporation, AT89C52 Datasheet, San Jose, CA 95131 USA, 2008.*
8. *Maxim Integrated, MAX487 Datasheet, San Jose, CA 95134 USA, 2018.*
9. *Telecommunications Industry Association, Electrical Characteristics of Generators and Receivers for Use in Balanced Digital Multipoint Systems, TIA/EIA-485-A, March 1998.*
10. *SerialMon. A free RS232 serial port monitor/sniffer software for Windows.* Disponible en: <http://www.serialmon.com/>
11. *RIGOL Technologies Inc, DS1022C Digital Oscilloscopes. Beaverton, Oregon (USA). 2016.*

12. S. Entertainment, Technology, Association, and American National Standards Institute, *American national standard E1.11-2008(R2013): entertainment technology USITT DMX512-A, Asynchronous serial digital data transmission standard for controlling lighting equipment and accessories*. New York: The Entertainment Services and Technology Association, 2013.
13. B. M. Wilamowski and J. D. Irwin, *Industrial communication technology handbook, Second Edition*: CRC Press, 2016.
14. I. E. Commission, "Industrial communication networks Part 1: Overview and guidance for the IEC 61158 and IEC 61784 series," ed, 2014.
15. I. E. Commission, "Industrial communication networks - Fieldbus specifications - Part 6: Application layer protocol specification," ed, 2014.

AUTORES

Carlos Bazán-Prieto. Graduado de Ingeniero Electrónico en 1989, Máster en Electrónica en 1998 por la Universidad Central "Martha Abreu" de Las Villas, Cuba, y de Doctor en Ciencias Técnicas por la Universidad de Alcalá, España en 2013. Actualmente es Profesor Titular en el Departamento de Electrónica y Telecomunicaciones de la Facultad de Ingeniería Eléctrica, Universidad Central "Martha Abreu" de Las Villas, Santa Clara, Cuba.

E-Mail: cabazan@uclv.edu.cu

Adrian Acuña-Pérez. Graduado de Ingeniero en Telecomunicaciones y Electrónica en el año 2016 en la Universidad Central "Martha Abreu" de Las Villas. Actualmente es Instructor en el Departamento de Electrónica y Telecomunicaciones de la Facultad de Ingeniería Eléctrica, Universidad Central "Martha Abreu" de Las Villas, Santa Clara, Cuba.

E-Mail: aacuna@uclv.cu

Rigoberto Acosta-González. Graduado de Ingeniero en Telecomunicaciones y Electrónica en el año 2017 en la Universidad Central "Martha Abreu" de Las Villas. Actualmente se desempeña como Graduado Universitario en Preparación en el Departamento de Electrónica y Telecomunicaciones de la Facultad de Ingeniería Eléctrica, Universidad Central "Martha Abreu" de Las Villas, Santa Clara, Cuba.

E-Mail: riacosta@uclv.cu



Los contenidos de la revista se distribuyen bajo una licencia Creative Commons Attribution-NonCommercial 3.0 Unported License