

## INTERFAZ DE BAJO COSTO USANDO UN PUERTO PARALELO Y VISUAL BASIC

*LOW-COST INTERFACE USING A PARALLEL PORT AND VISUAL BASIC*

ROGELIO ESCOBAR Y KENNON A. LATTAL

WEST VIRGINIA UNIVERSITY

### Resumen

Se describe el diseño de una interfaz de bajo costo que utiliza el puerto paralelo de una PC para controlar experimentos que requieren de 1 a 8 salidas (outputs) en combinación con 1 a 4 entradas (inputs) usando como lenguaje de programación Visual Basic® 2008 Express Edition que se distribuye de manera gratuita. Esta interfaz permite usar tanto consolas de entrada de respuestas como cámaras experimentales para realizar experimentos con humanos y animales no humanos usando únicamente una computadora. Para facilitar la aplicación de la interfaz en experimentos en análisis de la conducta, se describe el uso de micro interruptores y foto receptores para registrar respuestas y el uso de dispositivos por medio de micro relevadores.

**Palabras clave:** Interfaz, puerto paralelo, Visual Basic, instrumentación, relevadores, foto receptor.

---

El presente trabajo fue parte de un proyecto de investigación posdoctoral desarrollado por el primer autor bajo la supervisión del segundo autor en West Virginia University y fue apoyado por el PROFIP de la DGAPA, UNAM. El primer autor agradece a Michael Perone por sus valiosas enseñanzas sobre Visual Basic e instrumentación. Dirigir correspondencia a Rogelio Escobar, Selva 51-403 Col. Insurgentes-Cuicuilco, México, D. F. 04530 (e-mail: rescobar@servidor.unam.mx).

## Abstract

The design of a low-cost interface is described that uses the parallel port of a PC to control experiments that require from 1 to 8 outputs in combination with 1 to 4 inputs using the programming language Visual Basic® 2008 Express Edition that is freely distributed. This interface allows for controlling response-input consoles and experimental chambers that can be used in experiments with humans and nonhuman animals using only a computer. To facilitate the application of the interface in experiments on behavior analysis, the use of micro switches and photobeam sensors for recording responses as well as the use of devices controlled via micro relays is described.

**Keywords:** Interface, parallel port, Visual Basic, instrumentation, relays, photobeam sensors.

Una de las limitaciones a la que se enfrentan los investigadores interesados en el análisis experimental de la conducta, especialmente en Latinoamérica, es el alto costo del equipo que se utiliza para realizar experimentos. Si bien las cámaras de condicionamiento operante son relativamente sencillas, y en algunos laboratorios se fabrican ad hoc para los experimentos (e.g., Roca & Bruner, 2003), el equipo de control comercializado por compañías como Med Associates Inc., Coulbourn Instruments o Lafayette Instrument puede alcanzar un costo de varios miles de dólares.

En algunos experimentos con humanos se ha eliminado la necesidad de usar una interfaz programando el experimento en Visual Basic® (VB) y registrando las respuestas directamente en el teclado (e.g., Stokes, Mechner, & Balsam, 1999). El uso de VB se ha extendido en diferentes laboratorios debido a que permite la presentación de imágenes complejas en un monitor y reduce el equipo experimental con humanos a una computadora (véase Cabello, 2005; Cabello, Barnes-Holmes, O'Hora, & Stewart, 2002). De acuerdo con Cabello et al. VB es probablemente la herramienta más útil que tienen a la mano los investigadores interesados en desarrollar experimentos con una computadora.

Estos procedimientos, sin embargo, tienen el problema de que los sujetos, al tener acceso al teclado, pueden alterar el curso de las sesiones experimentales oprimiendo teclas que no son parte del experimento. En el presente artículo se describe una interfaz que utiliza el puerto paralelo de una computadora y permite controlar una cámara experimental o un dispositivo de entrada como un control de videojuegos para realizar experimentos programados usando Visual Basic® 2008 Express Edition (VBEE) que se distribuye de manera gratuita.

Algunos investigadores han descrito la flexibilidad que ofrece el uso del puerto paralelo para registrar respuestas y controlar dispositivos (e.g., Cushman, 1993; Dalrymple-Alford, 1992; Iversen, 2002; 2008). La interfaz descrita en el presente trabajo está basada en estas descripciones previas pero pretende mostrar de manera más detallada el funcionamiento del puerto paralelo en el control de experimentos

y combinar su funcionamiento con el lenguaje de programación VBEE. Esta interfaz involucra componentes simples que pueden obtenerse fácilmente en tiendas especializadas de electrónica y que en su totalidad, aunque dependiendo de la complejidad del experimento, no deben exceder de unos cuantos cientos de pesos. Cabe señalar que la presente interfaz pretende servir como un dispositivo para fines didácticos para investigadores interesados en el análisis de la conducta y aún se encuentra en su fase de prueba. Por lo tanto, no pretende, al menos en su un diseño actual, funcionar como un sustituto de las interfaces de control disponibles comercialmente que han mostrado su confiabilidad en numerosos experimentos.

### El puerto paralelo

Para entender el funcionamiento de la interfaz, es necesario describir brevemente el funcionamiento del puerto paralelo que está presente en algunas computadoras de hace 5 o más años y que comúnmente sirve para conectar aparatos como impresoras. En la Figura 1 se muestra un esquema de la ubicación de los 25 pines del puerto paralelo de una PC. Los pines se agrupan en tres puertos y se muestran en un color diferente. Los pines 2 a 9 forman parte del puerto de datos (D), los pines 1, 14, 16 y 17 forman parte del puerto de control (C) y los pines 10 a 13 integran el puerto de estatus (S). Los pines 18 a 25 van a tierra. Con fines prácticos en el presente trabajo se utilizarán únicamente los pines D0 a D7 y los pines S3 a S6.

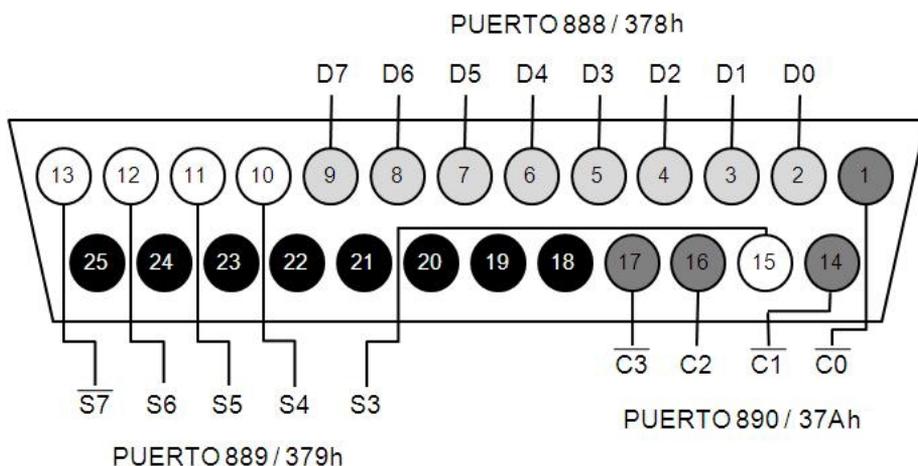
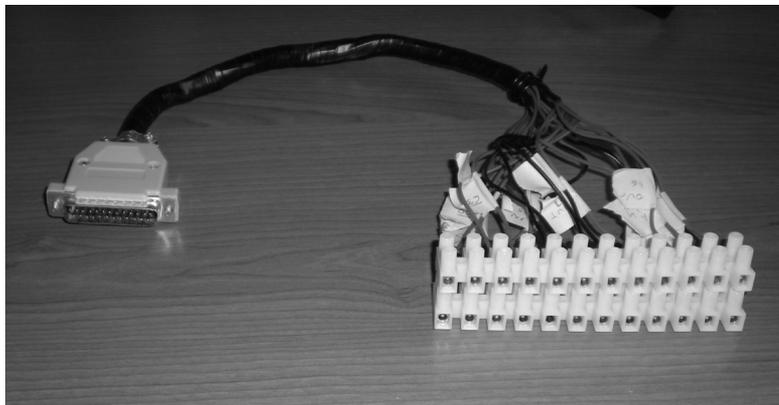


Figura 1. Diagrama de la ubicación de los pines en el puerto paralelo y las direcciones de los puertos.

Cada vez que un pin S hace contacto con un pin a tierra, su valor cambia y podemos detectar el cambio usando VB. Más adelante se describe el uso de VB para leer las entradas y generar salidas. Los pines D0 a D7 generan 5 VDC cada vez que se cambia su estado desde VB. Es importante señalar que el puerto paralelo es delicado y una conexión incorrecta puede dañarlo permanentemente. Se recomienda utilizar una tarjeta de puerto paralelo ISA que puede obtenerse en tiendas especializadas de equipo cómputo para reducir el posible daño que pudiera causarse al puerto paralelo cuando este se encuentra integrado a la tarjeta madre de la computadora.

La configuración más común del puerto paralelo lo ubica en la terminal LPT1 con las direcciones 888 (378h), 889 (379h), 890 (37Ah) como se muestra en la Figura 1. Para facilitar la descripción de la presente interface, en las siguientes secciones se asume que el puerto paralelo está configurado de esta forma. Sin embargo, dicha configuración puede variar y debe verificarse antes de proceder a controlar las entradas y salidas. La configuración del puerto puede verificarse en el Administrador de dispositivos de Windows XP®. Debe abrirse la opción Puertos (COM & LPT) y seleccionarse con el botón derecho del mouse el puerto paralelo que se desea verificar (e.g., LPT1). En el menú emergente debe seleccionarse propiedades. En la pestaña de recursos se muestra el rango de direcciones del puerto paralelo. Si el primer rango mostrado inicia en 0378 entonces las direcciones de los puertos son las que se muestran en la Figura 1. Si el rango es diferente (e.g., 0278 a 027A) pueden usarse los primeros tres valores del rango mostrados (632 [278h], 633 [279h], 634 [27Ah]) o puede incluso cambiarse la configuración desde el BIOS de la computadora. Sin embargo, el cambio de configuración depende de la tarjeta de puerto paralelo que se esté utilizando y debe consultarse el manual apropiado para realizarse.



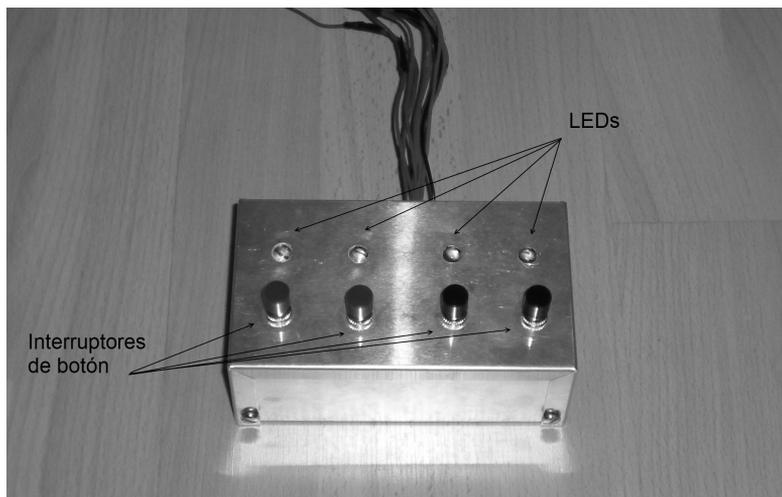
*Figura 2.* Cable de conexión con el puerto paralelo compuesto de un conector macho DB-25 (RadioShack®, Modelo 276-1547) con una cubierta de plástico (RadioShack®, Modelo 276-1549) conectado a dos mini barras de 12 terminales (RadioShack®, Modelo 274-680). Las conexiones se realizaron con cable de calibre 22.

Para facilitar la conexión del resto de los componentes con los pines, puede adquirirse un cable de puerto paralelo y remover uno de los conectores o fabricar un cable usando un conector DB-25 (e.g., Steren®, Modelo 500-025) y soldar cables a cada uno de los pines. En la Figura 2 se muestra un cable de este tipo que permite hacer las conexiones con facilidad.

### Control de experimentos con humanos

En el caso de controles tipo videojuego que se utilizan en experimentos con humanos (e.g., Raia, Shillingford, Miller, & Baier, 2000), podrían requerirse luces para señalar las respuestas o las condiciones. Éstas pueden producirse con diodos emisores de luz (LEDs) sin necesidad de una fuente de poder externa y conectando una resistencia en serie para cada LED. El ánodo del LED se conecta a una resistencia (e.g., 470  $\Omega$ ) que a su vez se conecta a alguno de los pines D0 a D7. El cátodo se conecta a alguno de los pines a tierra.

Para registrar las respuestas pueden usarse interruptores tipo botón. Una de las terminales puede conectarse a cualquiera de los pines S3 a S6 y la otra terminal a alguno de los pines a tierra. Para crear una consola los LEDs y los interruptores pueden montarse en una caja cuyo tamaño dependerá de las necesidades del experimento. Un diodo (e.g., 1N4148) conectado en serie en cada uno de los pines positivos puede ser útil para proteger el puerto paralelo.



*Figura 3.* Mini consola de control para experimentos con humanos. La consola está compuesta por cuatro interruptores de respuesta tipo botón (RadioShack®, Modelo 275-1556) y cuatro LEDs de 10 mm de alto brillo (RadioShack®, Modelo 276-015) para presentar estímulos visuales.

En la Figura 3 se muestra una mini consola de control con cuatro interruptores de respuesta y cuatro luces que sirven como indicadores. El diseño de este dispositivo de registro está basado en los experimentos con humanos de Perone y Kaminski (1992) pero tiene la ventaja de que elimina la necesidad de una interfaz comercial entre la consola de respuestas y la computadora. Un uso de esta mini consola es usar los LEDs para señalar cuales son los botones que se encuentran activos. Este detalle es útil, por ejemplo, en procedimientos en los cuales quiere señalarse que está disponible el reforzador (e.g., puntos en la pantalla) pero para recibirlo es necesario emitir una respuesta diferente a las respuestas que resultan en la disponibilidad del reforzador (véase Perone & Kaminski). Durante este período podrían apagarse los otros LEDs para señalar que sólo un botón se encuentra activo. Una ventaja de la presente interfaz en combinación con esta mini consola de control, sobre sistemas anteriores de registro por medio del teclado (e.g., Stokes et al., 1999), es que los participantes reciben únicamente el dispositivo de entrada de respuestas y el experimentador puede controlar las sesiones experimentales con el teclado.

### Control de cámaras experimentales

#### *Relevadores de estado sólido*

Debido a que los pines de salida del puerto paralelo generan únicamente 5 VDC es necesario añadir una fuente de poder externa si se requiere de encender dispositivos que requieran mayor voltaje. Por ejemplo, en las cámaras modernas de acondicionamiento operante los dispositivos funcionan generalmente con 28 VDC. Una forma simple de controlar dispositivos de este tipo lo describió Iversen (1992) y consiste en usar relevadores de estado sólido (e.g., Panasonic® PhotoMOS™, Modelo AQZ102). De acuerdo con la experiencia de los autores, estos relevadores pueden adquirirse en México en tiendas especializadas de electrónica por internet (e.g., Mouser electronics®: <http://www.mouser.com/>). Este tipo de relevadores tienen cuatro conectores: dos para activarlo (positivo y negativo) y dos para el interruptor (normalmente abierto y común). El relevador se activa directamente con uno de los pines S3 a S6 al conector positivo y uno los pines a tierra del puerto paralelo al conector negativo del relevador. El interruptor permite cargas de hasta 60 VDC que excede notablemente los requerimientos de las cámaras experimentales estándar y el relevador de estado sólido aísla el puerto paralelo de posibles descargas. El precio de cada relevador es cercano a los 10 dólares y el envío a México cuesta alrededor de 30 dólares.

#### *Micro relevadores de bobina*

Otra opción de menor costo y probablemente más accesible en México, consiste en utilizar micro relevadores de 5 VDC (e.g., Steren®, Modelo THD-1201L). Estos relevadores tienen cuatro conectores, dos para la bobina del relevador y tres para activar el interruptor (normalmente abierto, normalmente cerrado y común). El interruptor permite cargas de 30 VDC a 1 Amp o de 24 VDC a 3 Amps.

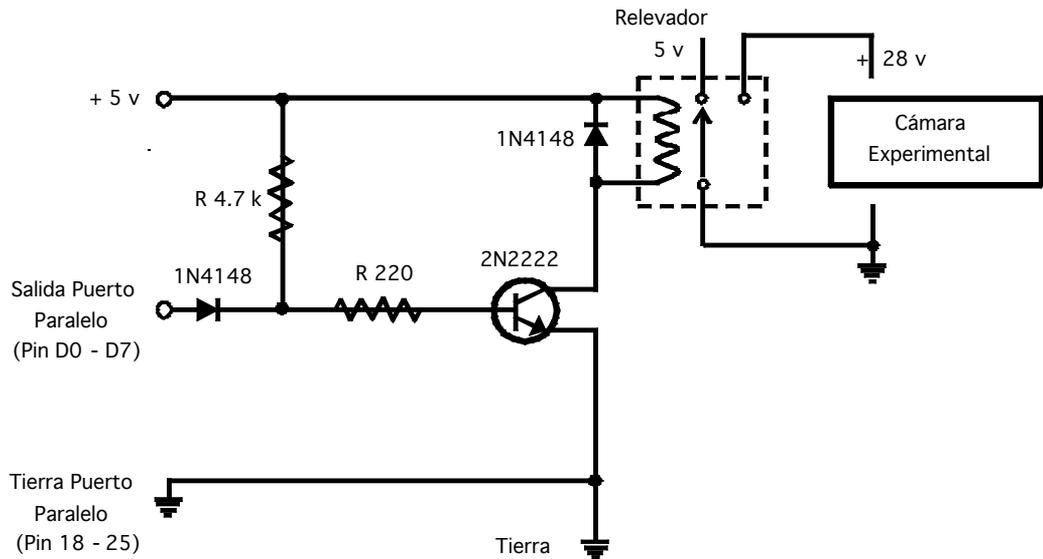


Figura 4. Diagrama del circuito para activar micro relevadores de 5 VDC usando el puerto paralelo.

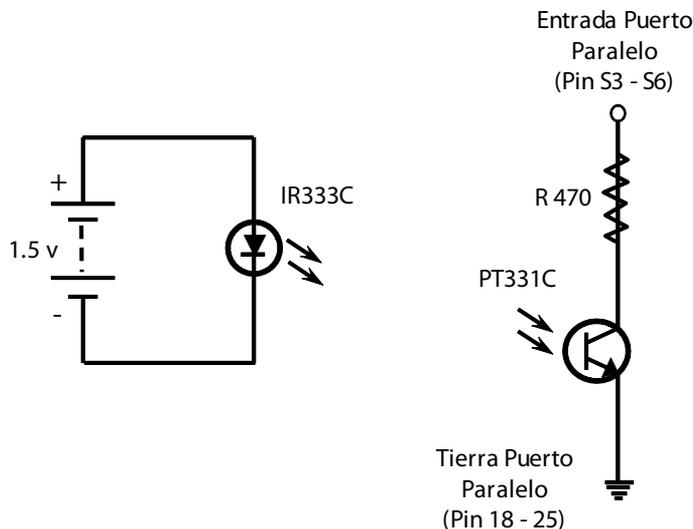
Para activar los micro relevadores de 5 VDC es necesario usar una fuente de poder externa y un transistor de tal forma que la carga no exceda el límite del puerto paralelo. En la Figura 4 se muestra el circuito para activar el micro relevador. En este diagrama y en las pruebas se utilizó una fuente de poder de 5 VDC para activar los relevadores y una fuente de poder de 24 VDC para activar los dispositivos dentro de una cámara experimental. En este diagrama, un transistor (2N2222) conectado a una resistencia de  $220\ \Omega$  (R100 en la figura) sirvió para activar la bobina del relevador y se utilizó un diodo (1N4148) para proteger el puerto paralelo. La resistencia de 4.7 k $\Omega$  (R4.7k en la figura) funciona como una resistencia de polarización (elevación) y puede añadirse para mejorar la estabilidad al sistema. Debido a que cada micro relevador incluye una bobina, es necesario añadir un segundo diodo en paralelo con los conectores de la bobina del relevador para proteger el transistor.

#### *Registro de respuestas con interruptores y fotorreceptores*

El registro de entrada de respuestas usando micro interruptores en teclas o palancas puede hacerse de la misma forma que se describió en la sección anterior sobre el *Control simple para registro con humanos*

Otra aplicación de la presente interfaz es el registro de respuestas que interrumpen un haz de luz. Esta aplicación es importante cuando se necesita registrar el momento en el cual los sujetos se aproximan a la zona donde se encuentra la comida dentro una cámara experimental o cuando las respuestas se especifican con un

criterio de desplazamiento. Una solución de bajo costo consiste en utilizar fototransistores en combinación con LEDs infrarrojos. Un fototransistor (e.g., Steren®, Modelo PT331C) permite la conducción entre el colector y el emisor cuando recibe luz infrarroja.



*Figura 5.* Diagrama del circuito para registrar la interrupción de un haz de luz usando el puerto paralelo. A la izquierda se muestra el diagrama del emisor, un LED infrarrojo, y a la derecha el diagrama del fototransistor que funciona como receptor.

Como se muestra en la Figura 5, los LED infrarrojos (e.g., Steren®, Modelo IR333C) pueden conectarse directamente a una batería de 1.5 v. El colector del fototransistor se conecta a cualquiera de los pines S3 a S6 con una resistencia de  $470 \Omega$  y el emisor se conecta a un pin a tierra. Cuando se interrumpe el paso de luz entre el LED y el fototransistor, se interrumpe también la conducción entre la entrada del puerto paralelo y el pin a tierra. Este cambio puede registrarse al leer el valor de la entrada del puerto paralelo usando VBEE. El procedimiento se describe en la siguiente sección.

Las cámaras experimentales de condicionamiento operante son relativamente simples y en diversas ocasiones se construyen prácticamente en su totalidad para ajustarlas a las necesidades del experimento. En la mayoría de los casos la parte más costosa de las cámaras experimentales es el dispensador de comida. Una solución para reducir gastos y poder operar una cámara experimental con la presente interfaz es reemplazar el dispensador de comida con una válvula de agua de bajo costo. Aunque con un propósito diferente al actual, Roca y Bruner (2003) describieron el uso de una válvula solenoide de 24 VDC para entregar unas gotas de agua a ratas

bajo diferentes valores de un programa de reforzamiento. Esta válvula en combinación con la interfaz descrita reduce considerablemente el costo del equipo necesario para controlar experimentos en el análisis de la conducta.

### Control del puerto paralelo con VBEE

La versión gratuita de VB llamada Express Edition puede descargarse del sitio <http://www.microsoft.com/express/Downloads/#2008-Visual-Basic>. Para el uso de la presente interfaz se utilizó el sistema operativo Windows XP® en una PC Dell™ Optiplex™ con un procesador Pentium 4™ a 2 GHz con 512 MB de memoria RAM con un puerto paralelo. De acuerdo con las especificaciones de VBEE, bajo Windows XP® computadoras con un procesador de al menos 1GHz y 192 MB en memoria RAM pueden correr el programa. Para controlar los relevadores y registrar las respuestas mediante el puerto paralelo es necesario acceder a las funciones incluidas en el archivo inpout32.dll. Este archivo puede descargarse de manera gratuita en diversos sitios de internet y es de distribución libre. El archivo debe copiarse en la carpeta C:\Windows\System32. Es necesario señalar que el archivo inpout32.dll funciona únicamente con Windows 98®, Windows NT®, Windows 2000® y Windows XP®.

Una vez copiado el archivo, debe crearse un módulo en un proyecto en VBEE que servirá para controlar las salidas y registrar las respuestas. En este módulo deben añadirse las siguientes líneas para habilitar las funciones del archivo inpout32.dll:

```
Public Declare Function Inp Lib "inpout32.dll" Alias "Inp32" (ByVal PortAddress As Integer) As Integer
```

```
Public Declare Sub Out Lib "inpout32.dll" Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)
```

Como puede notarse, con las líneas anteriores las funciones *Inp* (dirección, valor) y *Out* dirección, valor usadas en lenguajes anteriores como QuickBASIC o GW-BASIC se habilitan en el proyecto de VBEE.

#### Salidas

Para encender las salidas y generar 5 VDC en los pines D0 a D7 del puerto paralelo, la dirección más común del puerto de datos es 888 ó 378h. El valor de cada uno de los pines de salida es 1, 2, 4, 8, 16, 32, 64, 128 y 256, respectivamente. De esta manera, para generar 5 VDC en el pin 2 o D0 y para posteriormente apagarlo debemos ejecutar:

```
Out (888, 1)
```

```
Out (888, 0)
```

Los valores se suman para encender múltiples salidas de manera simultánea.

#### Entradas

Para registrar las respuestas en los pines S3 a S6 es necesario leer el valor del puerto con dirección 889 ó 379h. Cuando no se ha registrado ninguna respuesta el resulta-

do de leer el valor del puerto es 120 (126 en algunos sistemas). Por ejemplo con la siguiente instrucción, si no se ha introducido una respuesta, el valor de la variable `intRespuesta` es 120.

```
intRespuesta = Inp (889)
```

Si se introduce una respuesta o se conecta el pin S3 a un pin a tierra el valor disminuye 8 y la variable `intRespuesta` es igual a 112. Al conectar los pines S4, S5 y S6 a un pin a tierra, el valor disminuye a 104, 88 y 56, respectivamente. Los valores se acumulan cuando se emiten múltiples respuestas.

En el Apéndice se muestra el código que ejemplifica como pueden leerse entradas y generarse salidas usando un módulo en VBEE. En este ejemplo se utilizaron cuatro salidas y se registraron cuatro respuestas. Las características de VBEE permiten que este módulo sea accesible desde otros módulos en los que puede crearse, por ejemplo, un programa de reforzamiento. En este caso sería necesario ejecutar la función `inp()` repetidamente para leer el estado de las respuestas en “tiempo real” y leer las entradas directamente determinando si el valor de las variables `Resp1` a `Resp4` es igual a 0 (no hay respuesta) o a 1 (ocurrió una respuesta). Como puede notarse en este código, el programa previene que una sola respuesta sea registrada en más de una ocasión. En éste ejemplo, las salidas se generan desde cualquier módulo en el proyecto con el comando `Output (número de salida)`. Por ejemplo, `Output (12)` enciende las salidas 1 y 2 o, en otras palabras, genera 5 VDC en los pines S3 y S4 del puerto paralelo.

### Conclusión

El sistema descrito en el presente trabajo apunta en la dirección de eliminar la necesidad de usar una interfaz comercial en el control de experimentos tanto con humanos como con animales no humanos en el análisis de la conducta. Algunos investigadores han señalado la importancia del uso de VB para diseñar experimentos con sólo una computadora (véase Cabello, 2005; Cabello et al., 2002). La interfaz propuesta combina la flexibilidad de VB que permite la presentación de estímulos complejos tanto visuales como auditivos en un monitor de una computadora y el uso del puerto paralelo que permite activar innumerables dispositivos de control experimental y que también permite el registro de respuestas.

Una limitación de la interfaz con el puerto paralelo es que el sistema permite controlar solamente una cámara experimental o una consola para experimentos con humanos con una computadora. Este sistema puede resultar ineficiente cuando se requiere realizar experimentos con más de un sujeto de manera simultánea. Sin embargo, el costo de varias computadoras equipadas con un puerto paralelo y con la capacidad sugerida para correr el programa VBEE puede ser incluso menor que el costo de las interfaces comerciales.

Otra limitación del presente sistema que también es necesario señalar es que actualmente el puerto paralelo está cayendo en desuso debido al desarrollo de puertos

más rápidos como el Universal Serial Bus (USB). Las computadoras portátiles modernas ya no cuentan con un puerto paralelo y, a pesar de que estos puertos aún se encuentran en una gran cantidad de equipos de escritorio, es previsible que en un futuro próximo dejen de incluirse. Futuros trabajos enfocados a diseñar interfaces de bajo costo podrían aprovechar la creciente popularidad de los puertos USB para incrementar la compatibilidad de los equipos de control experimental.

### Referencias

- Cabello, F. (2005). Reseña de "Visual Basic for behavioral psychologists" de M. Dixon y O. Maclin, *Revista Latinoamericana de Psicología*, 37 (2). Descargado Mayo 1, 2010 de <http://redalyc.uaemex.mx/redalyc/html/805/80537211/80537211.html>.
- Cabello, F., Barnes-Holmes, D., O'Hora, D., & Stewart, I. (2002). Using visual basic in the experimental analysis of human behavior: A brief introduction. *Experimental Analysis of Human Behavior Bulletin*, 20, 18-21.
- Cushman, W. B. (1993). A parallel printer port to matrix driver with high current DAC output. *Behavior Research Methods, Instruments, & Computers*, 25, 48-52.
- Dalrymple-Alford, E. C. (1992). Response-key input via the IBM PC / XT / AT's parallel printer port. *Behavior Research Methods, Instruments, & Computers*, 24, 78-79.
- Iversen, I. H. (2002). Response-initiated imaging of operant behavior using a digital camera. *Journal of the Experimental Analysis of Behavior*, 77, 283-300.
- Iversen, I.H. (2008). An inexpensive and automated method for presenting olfactory or tactile stimuli to rats in a two-choice discrimination task. *Journal of the Experimental Analysis of Behavior*, 90, 113-124.
- Perone, M., & Kaminski, B. J. (1992). Conditioned reinforcement of human observing behavior by descriptive and arbitrary verbal stimuli. *Journal of the Experimental Analysis of Behavior*, 58, 557-575.
- Raia, C. P., Shillingford, S. W., Miller, H. L., Jr., & Baier, P. S. (2000). Interaction of procedural factors in human performance on yoked schedules. *Journal of the Experimental Analysis of Behavior*, 74, 265-281.
- Roca, A., & Bruner, C. A. (2003). Efectos de la frecuencia de reforzamiento sobre el palanqueo por agua en ratas privadas de comida. *Revista Mexicana de Análisis de la Conducta*, 29, 119-130.
- Stokes, P. D., Mechner, F., & Balsam, P. D. (1999). Effects of different acquisition procedures on response variability. *Animal Learning & Behavior*, 27, 28-41.

## Apéndice

Module Module1

```
Public Declare Function Inp Lib "inpout32.dll" Alias "Inp32" (ByVal PortAddress
As Integer) As Integer
```

```
Public Declare Sub Out Lib "inpout32.dll" Alias "Out32" (ByVal PortAddress As
Integer, ByVal Value As Integer)
```

```
Dim m_blnButton1LastTime As Boolean = False
```

```
Dim m_blnButton2LastTime As Boolean = False
```

```
Dim m_blnButton3LastTime As Boolean = False
```

```
Dim m_blnButton4LastTime As Boolean = False
```

```
Dim m_blnResponseActive As Boolean = False
```

```
Dim m_blnButtonPressed As Integer = 0
```

```
Dim m_blnButtonPressed1 As Boolean = False
```

```
Dim m_blnButtonPressed2 As Boolean = False
```

```
Dim m_blnButtonPressed3 As Boolean = False
```

```
Dim m_blnButtonPressed4 As Boolean = False
```

```
Dim m_Button As Boolean = False
```

```
Public Resp1 As Integer = 0
```

```
Public Resp2 As Integer = 0
```

```
Public Resp3 As Integer = 0
```

```
Public Resp4 As Integer = 0
```

```
Public Function Output(ByVal Value As Integer) As Integer
```

```
    Select Case Value
```

```
        Case Is = 0
```

```
            Out(888, 0)
```

```
        Case Is = 1
```

```
            Out(888, 1)
```

```
        Case Is = 2
```

```
            Out(888, 2)
```

```
        Case Is = 3
```

```
            Out(888, 8)
```

```
        Case Is = 4
```

```
            Out(888, 16)
```

```
        Case Is = 12
```

```
            Out(888, 5)
```

```
        Case Is = 13
```

```
            Out(888, 9)
```

```
        Case Is = 14
```

```
        Out(888, 17)
    Case Is = 23
        Out(888, 10)
    Case Is = 24
        Out(888, 20)
    Case Is = 34
        Out(888, 24)
    Case Is = 123
        Out(888, 11)
    Case Is = 134
        Out(888, 25)
    Case Is = 124
        Out(888, 21)
    Case Is = 234
        Out(888, 28)
    Case Is = 1234
        Out(888, 27)
    End Select
End Function

Public Function Input() As Integer
    If Inp(889) = 112 Then
        m_blnButtonPressed1 = True
    ElseIf Inp(889) = 104 Then
        m_blnButtonPressed2 = True
    ElseIf Inp(889) = 88 Then
        m_blnButtonPressed3 = True
    ElseIf Inp(889) = 56 Then
        m_blnButtonPressed4 = True
    Else
        m_blnButtonPressed1 = False
        m_blnButtonPressed2 = False
        m_blnButtonPressed3 = False
        m_blnButtonPressed4 = False
    End If

    Select Case m_blnButtonPressed1
        Case False
            m_blnButton1LastTime = False
        Case True
            If m_blnButton1LastTime = False Then
```

```
        m_blnButton1LastTime = True
        Resp1 = 1
        Return 1
    End If
    If m_blnButton1LastTime = True Then
        Resp1 = 0
        Return 10
    End If
End Select

Select Case m_blnButtonPressed2
    Case False
        m_blnButton2LastTime = False
    Case True
        If m_blnButton2LastTime = False Then
            m_blnButton2LastTime = True
            Resp2 = 1
            Return 2
        End If
        If m_blnButton2LastTime = True Then
            Resp2 = 0
            Return 20
        End If
    End Select

Select Case m_blnButtonPressed3
    Case False
        m_blnButton3LastTime = False
    Case True
        If m_blnButton3LastTime = False Then
            m_blnButton3LastTime = True
            Resp3 = 1
            Return 3
        End If
        If m_blnButton3LastTime = True Then
            Resp3 = 0
            Return 30
        End If
    End Select

Select Case m_blnButtonPressed4
    Case False
```

```
m_blnButton4LastTime = False
Case True
  If m_blnButton4LastTime = False Then
    m_blnButton4LastTime = True
    Resp4 = 1
    Return 4
  End If
  If m_blnButton4LastTime = True Then
    Resp4 = 0
    Return 40
  End If
End Select
Return 0
End Function
End Module
```