

---

# Circuito decodificador de señales IF de control remoto, para el manejo de aplicaciones en la PC

---

Eduardo Romero Aguirre

**Resumen**— Es tal el uso de la PC en aplicaciones tradicionalmente consideradas domésticas, que ha comenzado a reemplazar en los hogares (de la clase media alta) a reproductores de música, películas y sintonizadores de radio o TV.

A pesar de lo anterior, se ha limitado al usuario a tener que manipular cada una de las aplicaciones a través del teclado o del ratón, siendo que en la vida cotidiana, nuestra interacción con dichos aparatos caseros es a través de un dispositivo de mando inalámbrico regularmente infrarrojo.

Por tal motivo, en este artículo presenta a detalle el diseño del *hardware* y *firmware*, necesarios para la implementación de un circuito decodificador de señales infrarrojas provenientes de un mando remoto para su utilización en el manejo de aplicaciones, en especial las multimedia.

Se describe además en forma concisa el protocolo RC-5 y sus características. Se expone el uso del microcontrolador PIC 16F628 como parte fundamental del circuito, debido a que es el encargado de decodificar la señal, identificar el comando de control y enviar dicho comando en forma serial a la PC.

También se verifica en forma exhaustiva la compatibilidad del dispositivo diseñado con el software de aplicación PC Remote Control 4.0©, el cual es el encargado de traducir los datos enviados de manera serial a la PC, a un conjunto de acciones o tareas que debe tomar ejecutar la aplicación que se quiera manejar.

**Palabras clave**— Protocolo RC-5, señal infrarroja, microcontrolador, software de aplicación, codificación Manchester, decodificación.

## I. INTRODUCCIÓN

Actualmente, el control remoto de aparatos electrónicos es algo habitual. Es notoria la cantidad de aparatos electrónicos con mando a distancia que hoy en día podemos encontrar en los hogares.

Resulta evidente que los mandos a distancia han sido una revolución en todos los ámbitos, incluyendo al de la computación. Por otro lado, las computadoras son un recurso muy comúnmente usado en ambientes laborales y educativos,

que poco a poco se está trasladando hacia la mayoría de los hogares, desplazando a la mayoría de los aparatos domésticos tales como: reproductores de música, películas, juegos de video y sintonizadores de radio o TV.

Pero a pesar de lo anterior, la PC tiene la desventaja de no contar con un dispositivo inalámbrico que permita al usuario ejecutar de manera remota, tareas cotidianas como: subir el volumen, comenzar la reproducción de un video, avanzar a la siguiente dispositiva de una presentación, etc. En vez de esto se tiene que acercarse físicamente a donde se encuentra el teclado o el ratón para poder manejar la aplicación en curso.

## II. COMUNICACIÓN INALÁMBRICA POR INFRARROJO

El espectro infrarrojo es una radiación por debajo del rango de frecuencias visibles. Este es empleado más a menudo que la radiofrecuencia en aplicaciones domésticas debido a su inmunidad a la interferencia electromagnética y su bajo costo de implementación.

Sin embargo, otras fuentes pueden contener luz infrarroja e interferir con la comunicación de los dispositivos, tal es el caso de la luz del sol. Cualquier fuente de calor genera rayos infrarrojos, incluso nuestro cuerpo, por tal razón se debe establecer una clave de comunicación para que nuestro receptor discrimine las emisiones de rayos infrarrojos que no sean parte del mensaje que deseamos recibir.

El método preferido para eliminar tales interferencias es modular a una cierta frecuencia el haz infrarrojo. Así, el receptor sólo interpretará esta frecuencia infrarroja, desechando las demás [1].

Actualmente se ha establecido que una frecuencia de 30 a 60KHz de pulsaciones infrarrojas es una forma adecuada de transmitir información a baja velocidad. Por tal motivo, la mayoría de los mandos o controles remotos hacen uso de ella (36-40KHz).

En particular, generar una señal pulsante infrarroja de 36KHz no representa una gran complicación, la problemática, como suele suceder la mayoría de las veces, estriba en el desarrollo de la etapa que se encargará de recibir e identificar esta frecuencia.

---

Manuscrito recibido el 30 de junio del 2005. Este trabajo es respaldado por impulso, revista de, electrónica, eléctrica y sistemas computacionales. Eduardo Romero A. actualmente esta en Dpto. de Ing. Eléctrica del Instituto Tecnológico de Sonora. Ave. Antonio Caso S/N Col. Villa ITSON; Ciudad Obregón, Sonora, México; C.P. 85130; (e-mail eromero@itson.mx).

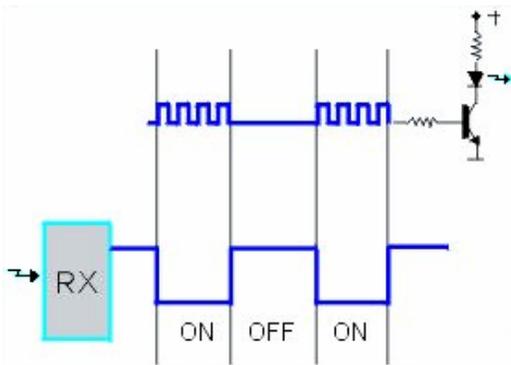


Fig. 1. Circuito típico de transmisión por infrarrojo.

Un circuito típico para un transmisor puede observarse en la figura 1. Un tren de pulsos, que denotará un estado lógico, es inyectado a la base de un transistor, el cual a su vez controlará la activación de un led infrarrojo para generar la señal pulsante [2].

Del lado del receptor, éste tratará de identificar esta frecuencia pulsante y le asignará el estado lógico correspondiente para convertirla en una secuencia de bits.

Protocolo de comunicación por infrarrojo, RC-5.

A continuación se describe éste protocolo que fue inventado por la compañía Phillips© para uso en sus equipos de control remoto [3].

Un mensaje individual es transmitido por un total de 14 *bits* en codificación Manchester. Cada *bit* tiene una duración de 1.776 ms. La codificación Manchester utiliza una transición del nivel bajo al alto para representar un 1 lógico y una transición del nivel alto al bajo para el 0 lógico [2, 4]. En la figura 2 puede visualizarse un ejemplo de mensaje del protocolo RC-5.

Los primeros 2 *bits* de inicio son usados para sincronización y ajuste del receptor, el *bit* de control (*toggle*) cambia cada vez que se presiona una tecla y permanece constante cuando se mantiene presionada una tecla del control. Los *bits* de dirección proporcionan, una característica muy poderosa, ya que cambiándolos podemos controlar más de un dispositivo con el mismo control, por ejemplo controlar dos televisores, uno con la dirección 00000 y otro con la dirección 00001. Por último, los 6 *bits* de comando determinan la función o la orden que se da al dispositivo, como por ejemplo, subir o bajar volumen [2].

### III. DESARROLLO DEL HARDWARE Y EL FIRMWARE

El diagrama a bloques del sistema puede ser visto en la figura 3. En primer término se encuentra el transmisor, el cual es un control remoto universal de la marca Zenith© modelo ZEN100 [5], al cual se le introdujo el código de programación 807, que corresponde al de un televisor Philips® con protocolo infrarrojo RC-5.

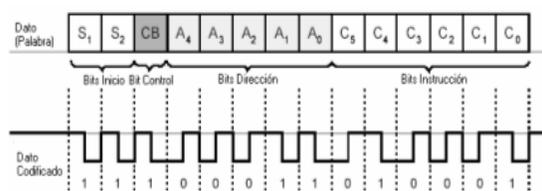


Fig 2: Ejemplo de mensaje del protocolo RC-5.

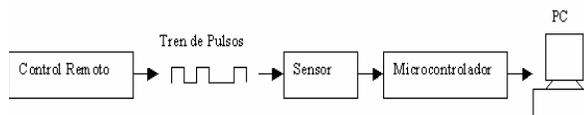


Fig. 3. Diagrama esquemático del sistema [6].

Con respecto a la parte receptora, que es la que se desarrolla en este trabajo, podemos identificar al sensor inalámbrico, el cual capta el haz infrarrojo modulado. Un microcontrolador PIC 16F628 recibe las variaciones de señal provenientes del sensor y las convierte en una serie de *bits*, los cuales son decodificados y traducidos en comandos RC-5 que son enviados por el puerto serie (ya con niveles de voltaje compatibles con RS-232).

Aunque no tan evidente, pero no por eso menos importante, están tanto el programa de control (*firmware*) para el PIC 16F628 como el *software* de interfaz que debe ejecutar la PC.

Para el diseño del *hardware* receptor por principios de cuenta se procedió a caracterizar la señal entregada por el sensor infrarrojo. En la figura 4, se puede ver la pantalla del osciloscopio mostrando distintas formas de la señal a la salida del sensor para diferentes comandos RC-5 (ver tabla 1). Vale la pena mencionar que con esta prueba se pudo constatar que el sensor infrarrojo invierte la señal original transmitida por el control remoto.

El algoritmo del programa de control (*firmware*) para el PIC 16F628 [7, 8, 9], representado en el diagrama de estados de la figura 5, está diseñado para monitorear constantemente el tren de pulsos que genera el sensor infrarrojo. En el instante que se detecta que el primer *bit* es un "1" en la trama, el programa ignora los siete *bits* siguientes, los cuales corresponden a: un segundo *bit* de inicio, un *bit* de control (*toggle*), y los *bits* de dirección (constantes porque el dispositivo a controlar no cambia). Posteriormente, el *firmware* debe extraer del tren de pulsos los seis *bits* que corresponden al comando de control del protocolo RC-5.

TABLA 1. DATOS RC-5 MOSTRADOS EN LA FIGURA 4.

Señal	Descripción	Bits de inicio	Bit de control (Toggle)	Bits de dirección	Bits de comando
A	Chan +	1 1	X	00000	100000
B	Chan -	1 1	X	00000	100001
C	Mute	1 1	X	00000	001101

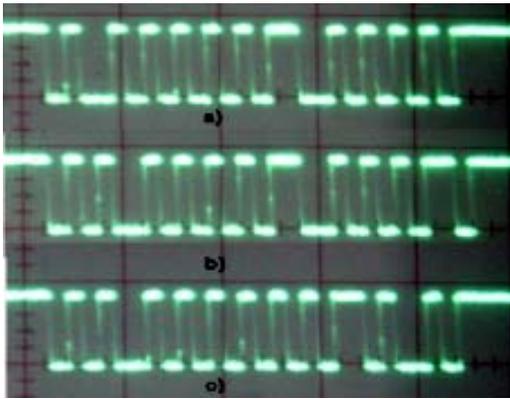


Fig. 4. Formas de la señal a la salida del sensor infrarrojo correspondiente a: a) chan +, b) chan -, c) mute.

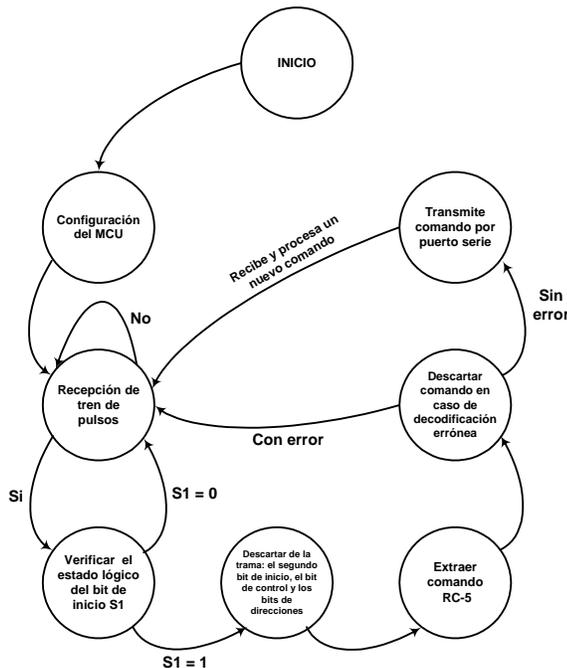


Fig. 5. Diagrama de estados del programa del control (firmware) para el microcontrolador PIC16F628.

En la decodificación de la secuencia de *bits* al comando RC-5 (6 *bits*), el programa del microcontrolador debe agregar dos *bits* en “0” en la parte más significativa para poder formar un dato de un *byte* (ver la tabla 2), mismos que envía al puerto serie de la computadora a través de un convertidor de voltaje TTL/RS-232; MAX232.

En una versión anterior del decodificador [10] se empleó un conector serial tipo DB-25 y el voltaje de alimentación era proporcionado por una batería cuadrada de 9 V.

En esta versión, con la finalidad de reducir el tamaño del circuito, se sustituyó el conector por un DB-9 y para evitar energizar el circuito con baterías, se optó por obtener el voltaje directamente desde el puerto serie de la PC al cual se conecta. Esto requirió de la inclusión de un C.I. regulador, debido a que el voltaje entregado por el puerto serie es diferente en cada PC. Esto es consecuencia de las distintas variantes del estándar RS-232, que sólo definen como un nivel lógico válido cualquier voltaje que sea superior a  $\pm 3$  V.

TABLA 2. LECTURA DEL PUERTO SERIE POR CADA TREN DE PULSO ENVIADO POR EL CONTROL REMOTO.

Botón del control remoto	IV. DATO ENVIADO AL PUERTO	
	Hexadecimal	Decimal
0	00	0
1	01	1
2	02	2
3	03	3
4	04	4
5	05	5
6	06	6
7	07	7
8	08	8
9	09	9
Vol +	10	16
Vol -	11	17
Chan +	20	32
Chan -	21	33
Enter	0A	10
Sleep	26	38
Power	0C	12
Mute	0D	13

La figura 6 muestra el circuito final del decodificador de comandos RC-5. Se puede notar que su tamaño es reducido (más aun que el propio control remoto), lo cual facilita su portabilidad. Lo único que se necesita para su conexión a la PC, es de un cable serial.

#### IV. SOFTWARE DE INTERFAZ

Para validar el funcionamiento del circuito, se desarrolló una programa (figura 7), en Visual Basic 6.0© para manejar las presentaciones en PowerPoint©. La técnica empleada en este punto es la de construcción de prototipos de *software* [11], en la cual se unen el desarrollo del *software* y el diseño de la interfaz en un solo punto.

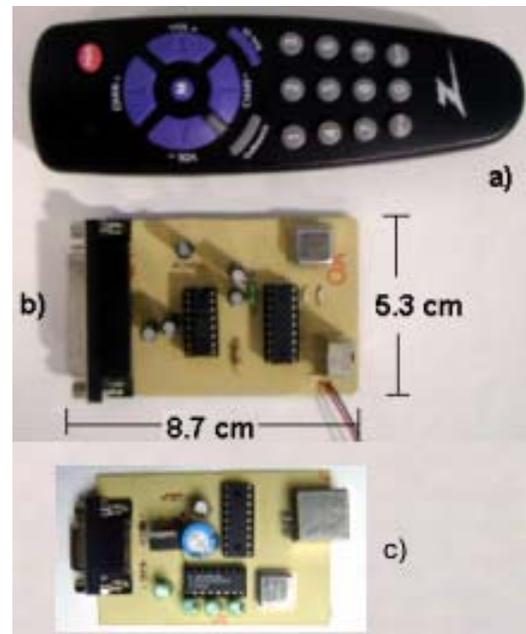


Fig. 6. a) Control universal ZEN100 marca Zenith©; b) Circuito decodificador (baterías); c) Circuito decodificador (energizado desde la PC).



Fig. 7. Panel principal del programa desarrollado.

Uno de los puntos relevantes del programa desarrollado es el referente a su interacción con la API de Windows®, ya que a través de ella, cada dato que se recibe se interpreta como si se hubiera introducido a través del teclado de la PC [12], haciendo esto transparente a PowerPoint®.

#### Funcionamiento con PC Remote Control®

Para investigar en que grado era compatible el decodificador diseñado con los programas comerciales para manejo en forma remota de aplicaciones se optó por PC Remote Control (figura 8) debido a que es el que cuenta con más funciones y comandos para el manejo de aplicaciones.

PC Remote Control es una aplicación de tipo shareware desarrollado por Moisés Cambra [13] y está disponible en idioma español. Además es capaz de procesar información recibida por puerto serial (genérico o propietario), TCP/IP, etc. y asignarle cualquiera de las siguientes acciones:

- ❑ Pulsación de tecla (incluyendo a aquellas que requieran combinarse con Alt, Ctrl y Shift).
- ❑ Movimiento del mouse (incluyendo las pulsaciones de los botones derecho, central e izquierdo).
- ❑ Acceso a la función *PostMessage* de la API de Windows.
- ❑ Control del volumen maestro.
- ❑ Cierre de sesión, apagado o reinicio de la PC.
- ❑ Manipulación de ventanas.

Lo anterior es particularmente útil ya que lo que se requiere manejar con el decodificador son:

- ❑ Presentaciones multimedia.
- ❑ Lectura de archivos extensos o navegación a distancia de páginas.
- ❑ Reproducciones de CD-ROM's, DVD's, VCD's o archivos de audio.
- ❑ Cualquier aplicación que pueda ser manejada con teclas rápidas o el ratón.

#### V. PRUEBAS REALIZADAS

Una vez integrados todos los elementos de *hardware* y *firmware*, se procedió a validar el funcionamiento de todo el sistema.

En las primeras pruebas se utilizó el software de interfaz diseñado para PowerPoint®. El resultado fue satisfactorio con las versiones para Windows 98® y 2000®, no así para la de Windows XP®, en donde a pesar de que los comandos eran recibidos correctamente en la PC, no era posible transferirlos a

la aplicación. Para corregir ese error, se rediseñó de forma total el módulo del programa encargado de tal tarea.

El siguiente conjunto de pruebas fue con PC Remote, para esto el software requirió de un corto periodo de aprendizaje, en el que se compara los datos que provienen del puerto serie con los previamente aprendidos, y si la comparación resulta cierta, se ejecuta la acción especificada para ese comando infrarrojo de forma transparente en la ventana activa.

De esta forma, PC Remote Control se configuró para manejar PowerPoint®, Winamp® y PowerDVD® en forma inalámbrica con el control remoto. Habilitando funciones comunes tales como: avanzar diapositiva, retroceder diapositiva, reproducir siguiente canción, disminuir volumen, detener/reanudar reproducción, etc. El resultado fue satisfactorio, independientemente del sistema operativo.

En lo que respecta al hardware del decodificador, se tuvo el contratiempo de que funcionaba correctamente en PC's de escritorio, no así cuando se conectaba a PC's portátiles. Al realizar pruebas adicionales se llegó a la conclusión de que la falla era producto de que el voltaje que entregaban las PC's portátiles era mucho menor que el de las PC's de escritorio (pero sin salirse del estándar), lo que ocasionaba que el voltaje de alimentación al circuito no fuera el adecuado. Esto se solucionó cambiando el tipo de regulador.

También se determinó que otros parámetros fundamentales a evaluar eran el alcance y la direccionalidad, y con ellos establecer los límites adecuados para la operación del sistema. Debido a que la luz es un factor que influye en la recepción del sensor infrarrojo, fue necesario establecer su magnitud por medio de un medidor de intensidad luminosa.

La figura 9 engloba los resultados de las pruebas de alcance y dirección entre el transmisor (control remoto) y el receptor (circuito decodificador). En línea recta se obtuvo una distancia máxima de operación de 9.68 metros con una iluminación de 1.31 pie-candelas, mientras que a una iluminación mayor, 7.22 pie-candelas, la distancia máxima de operación fue de 8.57 metros. El ángulo de apertura máximo fue de 80° (40° izquierda y 40° derecha) a una iluminación de 1.31 pie-candelas, con una distancia máxima de operación de 6 metros.



Fig. 8. Pantallas de PC Remote Control.

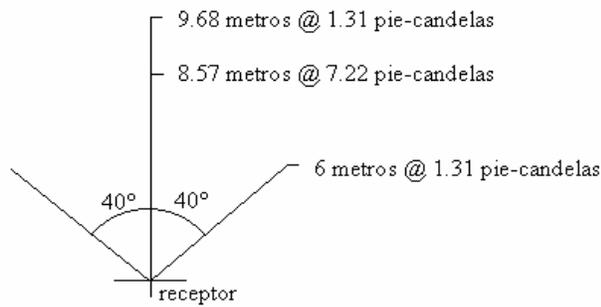


Fig. 9. Alcance y dirección del decodificador.

## VI. CONCLUSIONES.

Se puede observar que el diseño del circuito final es muy simple, sin embargo; el desempeño obtenido es bueno. En la actualidad se planea usarse en las salas audiovisuales y de videoconferencia del ITSON. Sin embargo debido a su flexibilidad, no se descarta la posibilidad de introducirle más mejoras.

También se pudo corroborar que la funcionalidad del circuito se incrementa cuando opera en conjunto con PC Remote Control. Con lo que es posible controlar cualquier aplicación de Windows© que soporte teclas de acceso rápido (*HotKeys* o *KeyShortcuts*) o manejo a través del ratón, con la ventaja de que puede especificarse una ventana de destino preferida para que tenga lugar la acción simulada, de esta forma no es necesario cambiar la aplicación activa.

En lo que respecta a su precio, cabe mencionar que comparado con versiones comerciales, su bajo costo (\$200) resulta atractivo.

## REFERENCIAS

- [1] Vishay (2001), Photo Modules for PCM Remote Control Systems, PA, U.S.A: Vishay Intertechnology, Inc. Disponible en: <http://www.vishay.com/ir-receiver-modules/devices-tsop17x-list/> [2003, 05 de octubre]

- [2] San Bergmans (2001), Philips RC-5 Protocol, Oisterwijk, Holanda: SB-Projects. Disponible en: <http://www.xs4all.nl/~sbp/knowledge/ir/rc5.htm> [2003, 15 de septiembre].
- [3] Infra Red Protocols, Koninklijke Philips Electronics N.V. Disponible en: [www.remotecontrol.philips.com](http://www.remotecontrol.philips.com) [2003, 15 de septiembre].
- [4] Vleeschauwer David, Phillips RC5 infrared remote protocol, Denderleeuw, Belgica: Davshomepage, Disp. en <http://users.pandora.be/davshomepage/rc5.htm> [2003, 15 de septiembre].
- [5] Instructivo control ZEN100., Zenith Electronics Corporation.
- [6] Núñez, Rogelio, Et al. (2002), "Manejo de presentaciones PowerPoint con un control de televisión usando el microcontrolador MC68HC11", Reporte Técnico, Departamento de Ingeniería Eléctrica, ITSON, México.
- [7] Microchip (2003), PIC16F62X Data Sheet, U.S.A., Microchip Technology Incorporated. Disponible en: <http://www.microchip.com/1010/pline/picmicro/category/embctrl/8kbytes/devices/16c628/index.htm> [2003, 05 de octubre].
- [8] Microchip (2003), Complete Reference Manual Mid-Range, U.S.A., Microchip Technology Incorporated. Disponible en: <http://www.microchip.com/1010/pline/picmicro/category/embctrl/8kbytes/devices/16c628/index.htm> [2003, 05 de octubre].
- [9] VALLEJO, Horacio Daniel. "Microcontroladores PIC: funcionamiento, programación, diseños". Ed. Quark. 2002, Argentina. Cap. II Manejo de las instrucciones de un PIC.
- [10] [PRESSMAN, R. S. "Ingeniería del software: un enfoque práctico". Ed. McGraw-Hill. Quinta edición 2002, España. Cap. XII Modelado del análisis. Pág. 205.
- [11] MSDN Library Visual Studio 6.0 (1998). La función SendKeys, [CD-ROM]. Estados Unidos: Microsoft Corporation, [2003, 10 de octubre].
- [12] <http://www.premotecontrol.com> [2004, 30 de Agosto]

**Eduardo Romero Aguirre** obtuvo el grado de Ingeniero en Electrónica opción Instrumentación en el Instituto Tecnológico de Orizaba en 1995 y el grado de Maestro en Ciencias en Ingeniería Electrónica en el área de Sistemas Digitales, en el Centro Nacional de Investigación y Desarrollo Tecnológico - CENIDET de Cuernavaca, Morelos, México en 1999. Ha realizado diversos proyectos relacionados con sistemas de adquisición de datos. Actualmente labora como profesor/investigador de tiempo completo en el Departamento de Ingeniería Eléctrica del Instituto Tecnológico de Sonora y es profesor con perfil PROMEP.