

Propiedades y Eventos básicos del control MSComm

CommPort (Propiedad)

Establece y devuelve el número del puerto de comunicaciones.

Sintaxis

objeto.**CommPort**[= *valor*]

La sintaxis de la propiedad **CommPort** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Una expresión que especifica un objeto en particular.
<i>valor</i>	Un valor entero que especifica el número de puerto.

Comentarios

Puede especificar para *valor* cualquier número entre 1 y 16 (el valor predeterminado es 1) en tiempo de diseño. Sin embargo, el control **MSComm** genera el error 68 (El dispositivo no está disponible) si no existe el puerto cuando intenta abrirlo con la propiedad **PortOpen**.

Advertencia Debe establecer la propiedad **CommPort** antes de abrir el puerto.

Tipo de datos

Integer

Settings (Propiedad)

Establece y devuelve los parámetros de velocidad en baudios, paridad, bits de datos y bits de parada.

Sintaxis

objeto.**Settings** [= *valor*]

La sintaxis de la propiedad **Settings** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Una expresión que especifica un objeto en particular.
<i>valor</i>	Una expresión de cadena que representa el valor del puerto de comunicaciones, como se describe a continuación.

Comentarios

Si *valor* no es válido cuando se abre el puerto, el control **MSComm** genera el error 380 (El valor de la propiedad no es válido).

Valor está compuesto por cuatro valores y tiene el formato siguiente:
"BBBB,P,D,S"

Donde BBBB es la velocidad en baudios, P es la paridad, D es el número de bits de datos y S es el número de bits de parada. El valor predeterminado de *valor* es:
"9600,N,8,1"

La siguiente lista muestra las velocidades en baudios válidas.

Valor
110
300
600
1200
2400
9600 (predeterminado)
14400
19200
28800
38400
56000
128000
256000

La siguiente tabla describe los valores de paridad válidos.

Valor	Descripción
E	Par
M	Marca
N	(Predeterminado) Ninguno
O	Impar
S	Espacio

La siguiente lista muestra los valores válidos de bits de datos.

Valor	
4	
5	
6	
7	
8	(Predeterminado)

La siguiente lista muestra los valores válidos de bits de parada.

Valor	
1	(Predeterminado)
1.5	
2	

Tipo de datos

String

InputLen (Propiedad)

Establece y devuelve el número de caracteres leídos por la propiedad **Input** del búfer de recepción.

Sintaxis

objeto.**InputLen** [= *valor*]

La sintaxis de la propiedad **InputLen** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Una expresión que especifica un objeto en particular.
<i>valor</i>	Una expresión de tipo Integer que especifica el número de caracteres leídos por la propiedad Input del búfer de recepción.

Comentarios

El valor predeterminado de la propiedad **InputLen** es 0. Este valor hace que el control **MSComm** lea todo el contenido del búfer de recepción cuando se usa **Input**.

Si no hay caracteres **InputLen** disponibles en el búfer de recepción, la propiedad **Input** devuelve una cadena de longitud cero (""). Si lo desea, el usuario puede comprobar la propiedad **InBufferCount** para determinar si está presente el número necesario de caracteres antes de usar **Input**.

Esta propiedad es útil para leer datos de las máquinas cuyo resultado tiene un formato de bloques de datos de longitud fija.

Tipo de datos

Integer

PortOpen (Propiedad)

Establece y devuelve el estado del puerto de comunicaciones (abierto o cerrado). Esta propiedad no está disponible en tiempo de diseño.

Sintaxis

objeto.**PortOpen** [= *valor*]

La sintaxis de la propiedad **PortOpen** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Una expresión que especifica un objeto en particular.
<i>valor</i>	Una expresión booleana que especifica el estado del puerto de comunicaciones.

Valores

Los valores admitidos para *valor* son:

Valor	Descripción
True	El puerto está abierto
False	El puerto está cerrado

Comentarios

Cuando la propiedad **PortOpen** es **True** se abre el puerto. Si le asigna el valor **False** se cierra el puerto y se borra el contenido de los búferes de recepción y de transmisión. El control **MSComm** cierra automáticamente el puerto serie cuando finaliza la aplicación.

Asegúrese de que la propiedad **CommPort** contenga un número de puerto válido antes de abrir el puerto. Si la propiedad **CommPort** es un número de puerto no válido, cuando intente abrir el puerto el control **MSComm** generará el error 68 (El dispositivo no está disponible).

Además, el dispositivo del puerto serie tiene que aceptar los valores actuales de la propiedad **Settings**. Si la propiedad **Settings** contiene una configuración de comunicaciones que el hardware no admite, es posible que el hardware no funcione correctamente.

Si cualquiera de las propiedades **DTREnable** o **RTSEnable** tiene el valor **True** antes de que se abra el puerto, tendrán el valor **False** cuando se cierre el puerto. De lo contrario, las líneas DTR y RTS permanecerán en su estado anterior.

Tipo de datos

Boolean

Output (Propiedad)

Escribe una cadena de datos en el búfer de transmisión. Esta propiedad no está disponible en tiempo de diseño y es de sólo lectura en tiempo de ejecución.

Sintaxis

objeto.**Output** [= *valor*]

La sintaxis de la propiedad **Output** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Una expresión que especifica un objeto en particular.
<i>valor</i>	Una cadena de caracteres que se va a escribir en el búfer de transmisión.

Comentarios

La propiedad **Output** puede transmitir datos binarios o de texto. Para enviar texto mediante la propiedad **Output**, debe especificar una variable de tipo Variant que contenga una cadena. Para enviar datos binarios, debe pasar a la propiedad **Output** un Variant que contenga una matriz de bytes.

Generalmente, si envía a una aplicación una cadena ANSI, puede enviarla como datos de texto. Si tiene datos que contengan caracteres de control, nulos, etc., debe pasarlos como datos binarios.

Tipo de datos

Variant

Input (Propiedad)

Devuelve y quita una cadena de datos del búfer de recepción. Esta propiedad no está disponible en tiempo de diseño y es de sólo lectura en tiempo de ejecución.

Sintaxis

objeto.**Input**

Objeto representa una expresión que especifica un objeto en particular.

Comentarios

La propiedad **InputLen** determina el número de caracteres leídos por la propiedad **Input**. Asignar a **InputLen** el valor 0 hace que la propiedad **Input** lea todo el contenido del búfer de recepción.

La propiedad **InputMode** determina el tipo de datos que se recuperan mediante la propiedad **Input**. Si **InputMode** es **comInputModeText**, la propiedad **Input** devuelve datos de texto en una variable de tipo **Variant**. Si **InputMode** es **comInputModeBinary**, la propiedad **Input** devuelve datos binarios en una matriz de bytes de tipo **Variant**.

Tipo de datos

Variant

CommEvent (Propiedad)

Devuelve el evento o el error de comunicación más reciente. Esta propiedad no está disponible en tiempo de diseño y es de sólo lectura en tiempo de ejecución.

Sintaxis

objeto.**CommEvent**

Objeto representa una expresión que especifica un objeto en particular.

Comentarios

Aunque el evento OnComm se genera siempre que se produce un error o un evento de comunicaciones, la propiedad **CommEvent** almacena el código numérico de dicho error o evento. Para determinar el error o el evento real que ha ocasionado el evento OnComm, debe consultar la propiedad **CommEvent**.

La propiedad **CommEvent** devuelve uno de los siguientes valores para eventos o errores de comunicación. Estas constantes también se encuentran en la biblioteca de objetos para este mismo control.

Los errores de comunicación incluyen los siguientes valores:

Constante	Valor	Descripción
comEventBreak	1001	Se ha recibido una señal de interrupción.

comEventFrame	1004	Error de trama. El hardware ha detectado un error de trama.
comEventOverrun	1006	Pérdida de información en puerto. El hardware no ha terminado de leer un carácter antes de llegar el siguiente y lo ha perdido.
comEventRxOver	1008	Desbordamiento del búfer de recepción. No hay espacio para más datos en el búfer de recepción.
comEventRxParity	1009	Error de paridad. El hardware ha detectado un error de paridad.
comEventTxFull	1010	Búfer de transmisión lleno. El búfer de transmisión estaba lleno cuando se ha intentado agregar un carácter a la cola de transmisión.
comEventDCB	1011	Error inesperado al recuperar el Bloque de control de dispositivos (DCB) para el puerto.

Los eventos de comunicación incluyen los valores siguientes:

Constante	Valor	Descripción
comEvSend	1	Hay menos caracteres en el búfer de transmisión de los que indica el número SThreshold.
comEvReceive	2	Número de caracteres RThreshold recibido. Este evento se genera continuamente hasta que se usa la propiedad Input para quitar datos del búfer de recepción.
comEvCTS	3	Cambio en la línea Preparado para enviar.
comEvDSR	4	Cambio en la línea Equipo de datos preparado. Este evento sólo se desencadena cuando la línea DSR cambia de 1 a 0.
comEvCD	5	Cambio en la línea Detección de portadora.
comEvRing	6	Detectada llamada. Algunos UART (transmisores-receptores asíncronos universales) no aceptan este evento.
comEvEOF	7	Carácter de fin de archivo (carácter ASCII 26) recibido.

Tipo de datos

Integer

SThreshold (Propiedad)

Establece y devuelve el número mínimo de caracteres que puede admitir el búfer de transmisión antes de que el control **MSComm** asigne a la propiedad **CommEvent** el valor **comEvSend** y genere el evento OnComm.

Sintaxis

objeto.**SThreshold** [= *valor*]

La sintaxis de la propiedad **SThreshold** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Una expresión que especifica un objeto en particular.
<i>valor</i>	Una expresión de tipo Integer que representa el número mínimo de caracteres del búfer de transmisión antes de que se genere el evento OnComm.

Comentarios

Al asignar a la propiedad **SThreshold** el valor 0 (predeterminado) se desactiva la generación del evento OnComm para los eventos de transmisión de datos. Al asignar a la propiedad **SThreshold** el valor 1 el control **MSComm** genera el evento OnComm cuando el búfer de transmisión está completamente vacío.

Si el número de caracteres que contiene el búfer de transmisión es menor que *valor*, la propiedad **CommEvent** será **comEvSend** y se generará el evento OnComm. El evento comEvSend sólo se desencadena una vez, cuando se llega al número de caracteres establecido en **SThreshold**. Por ejemplo, si **SThreshold** es cinco, el evento comEvSend sólo se produce cuando el número de caracteres baja de cinco a cuatro en la cola de salida. Si en la cola de salida nunca hay más caracteres que los establecidos en **SThreshold**, nunca se desencadenará el evento.

Tipo de datos

Integer

RThreshold (Propiedad)

Establece y devuelve el número de caracteres que se van a recibir antes de que el control **MSComm** asigne a la propiedad **CommEvent** el valor **comEvReceive** y genere el evento OnComm.

Sintaxis

objeto.**RThreshold** [= *valor*]

La sintaxis de la propiedad **RThreshold** consta de las siguientes partes:

Parte	Descripción
<i>objeto</i>	Una expresión que especifica un objeto en particular.
<i>valor</i>	Una expresión de tipo Integer que especifica el número de caracteres que se van a recibir antes de generar el evento OnComm.

Comentarios

Al asignar a la propiedad **RThreshold** el valor 0 (el valor predeterminado) se desactiva la generación del evento OnComm cuando se reciben caracteres.

Al asignar a la propiedad **RThreshold** el valor 1, por ejemplo, hace que el control **MSComm** genere el evento OnComm cada vez que entra un carácter en el búfer de recepción.

Tipo de datos

Integer

OnComm (Evento)

El evento OnComm se genera siempre que cambia el valor de la propiedad **CommEvent** e indica que se ha producido un evento o un error en la comunicación.

Sintaxis

Private Sub *objeto*_OnComm ()

Objeto representa una expresión que especifica un objeto en particular.

Comentarios

La propiedad **CommEvent** contiene el código numérico del error o el evento real que ha generado el evento OnComm. Tenga en cuenta que si asigna a las propiedades **RThreshold** o **SThreshold** el valor 0, se desactiva la interceptación de los eventos **comEvReceive** y **comEvSend**, respectivamente.