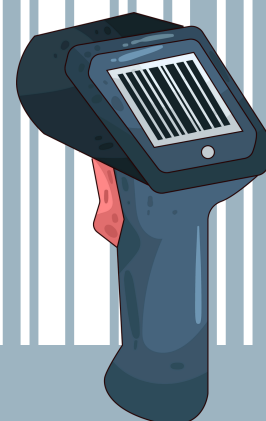


BARRAS 0444 / 0447

Estándar Interleaved 2 of 5



SIRO pone a disposición las especificaciones técnicas del estándar "Interleaved 2 of 5" a fin de facilitar la implementación de las barras con diseño **0444/0447**.

Sinopsis

- **Estándar a utilizar:** Interleaved 2 of 5.
- **Fuente:** IDAutomationHI25S ó IDAutomationHI25L.
- **Tamaño de fuente sugerido:** 10 ó 12.
- **Dimensiones óptimas de las barras impresas:** mínimo 8cm – máximo 10cm.
- **Alcance de este documento:** El presente memo es solo válido para tecnologías de desarrollo basadas en Windows.

Paso 1

Obtención de la Fuente

El estándar a utilizar será el "**Código Entrelazado 2 de 5 (Interleaved 2 of 5)**", cuyas características se indican a continuación:

- **Caracteres permitidos:** numéricos.
- **Densidad máxima:** 7.1 caracteres/cm.
- **Característica adicional:** caracteres inicial y final únicos.
- **Impresión:** no deberá realizarse mediante impresoras del tipo "matriz de punto", a efecto de evitar inconvenientes en la lectura del código de barras. Lo óptimo son las impresoras láser.
- **Decodificación numérica:** el código de barras deberá especificar también los datos en formato numérico, en la parte inferior del mismo, para su ingreso manual en caso de dificultades en la lectura de las barras.

La fuente puede obtenerse desde la **web de SIRO DEVELOPERS** dentro de la sección Código de Barras: **Fuente**.

Paso 2

Tratamiento del string para la obtención de las barras

Una vez desarrollada, acorde a la especificación que Banco Roela le ha suministrado (diseño del código de barra 0444/0447), y obtenida la cadena numérica que representa la identificación del registro de cobranza (string), debe aplicarse una modificación a dicha cadena a fin de que sea compatible con los estándares del código de barras.

Por ejemplo, para la cadena (string):

04440000132671207300093700060098385000000000515000934600

Debe aplicarse una transformación que obtendrá la cadena que cumple con los estándares del código de barra, obteniendo como resultado:

È%M!!.;h5j!*F!'ÉGS!!!!TS!~O!ì

A fin de explicar la transformación, en el **Anexo I**, se comparte un ejemplo desarrollado en VisualBasic 6

A este resultado se le aplica la fuente (recomendamos utilizar: IDAutomationHI25S ó IDAutomationHI25L a un tamaño de 10 o 12) y se obtiene el código de barras:



Consideraciones y sugerencias

Tener en cuenta que la **densidad máxima** debe ser de **7.1 caracteres/cm**.

La densidad del código es la relación entre la cantidad de caracteres codificados y la longitud que ocupa una vez impreso. Dado que el código de barras con diseño **0444/0447** cuenta con **56 caracteres**, para respetar la densidad máxima, una vez impreso en el cupón del cliente, **no debe medir menos de 8 cm**. El tamaño máximo depende del escáner que se utilice para leer el código de barras y de cómo se implemente el mismo (a través de una fuente o programa).

Por pruebas realizadas, se observa que en general si la impresión del código es mayor a 10 cm se puede llegar a tener problemas con los lectores de mano.

Recordar que el código de barras deberá especificar también los **datos en formato numérico**, en la parte inferior del mismo. Por lo anterior sugerimos utilizar las fuentes **IDAutomationHI25S** (versión corta (Short)) o **IDAutomationHI25L** (versión larga (Large)).

Ambos tipos de fuentes mencionados exhiben automáticamente los datos numéricos al pie de las barras. Se sugiere trabajar con un tamaño de **fuente 12**, lo que da como resultado, una medida del código de barra una vez impreso de: 97 mm. Si bien se recomienda que una vez **impreso** el código de barras **mida no menos de 8cm y no más de 10cm**, no quiere decir que otras medidas no funcionen, sino que simplemente se puede tener problemas con los lectores manuales utilizados en las bocas de cobro.

La fuente debe aplicarse en **forma directa** en el comprobante y nunca deben aplicarse imágenes obtenidas en otro proceso, ya que pueden presentarse inconsistencias y pixelado, lo que hace imposible la lectura de las barras.

Verificar que la transformación del string al aplicarle la fuente **no contenga espacios en blanco**, en ningún lugar de la cadena, como tampoco al principio ni al final, de modo de lograr la correcta aplicación de la fuente.

Importante

Se sugiere agregar la siguiente leyenda en el cupón/factura: Abonar en: Rapipago, Pago Fácil, Plus Pagos, Provincia Pagos y Cobro Express. Entidad Recaudadora: BANCO ROELA a través de SIRO.

Es importante realizar la homologación de los códigos de barras en forma previa a imprimirlos en los cupón/factura a emitir. Para su validación, emitir los mismos a: **codigosdebarra@bancoroela.com.ar**.

Ante cualquier consulta técnica sobre el desarrollo, ponemos a disposición los siguientes canales de consulta:

- Por mail a **sirodevelopers@bancoroela.com.ar**
- Por WhatsApp **+54 9 3513 95-2468**

Favor de indicar al momento de realizar la consulta los datos del sistema y del cliente para el cual se está desarrollando el código de barras.

ANEXO I: MACRO VISUAL BASIC 6

Macro para tratar el string numérico y representar la fuente, desarrollada en Visual Basic 6 (<http://www.bancoroela.com.ar/sirodesa/MacroVB6.txt>):

```
Public Function I2of5(DataToEncode As String) As String
    DataToPrint = ""
    DataToEncode = RTrim(LTrim(DataToEncode))
    OnlyCorrectData = ""
    ' Check to make sure data is numeric and remove dashes, etc.
    StringLength = Len(DataToEncode)
    For I = 1 To StringLength
        'Add all numbers to OnlyCorrectData string
        '2006.2 BDA modified the next 3 lines for compatibility with different office
        versions
        'If IsNumeric(Mid(DataToEncode, I, 1)) Then OnlyCorrectData = OnlyCorrectData
        & Mid(DataToEncode, I, 1)
        CurrentCharNum = AscW(Mid(DataToEncode, I, 1))
        If CurrentCharNum > 47 And CurrentCharNum < 58 Then OnlyCorrectData =
        OnlyCorrectData & Mid(DataToEncode, I, 1)
    Next I
    DataToEncode = OnlyCorrectData
    'Check for an even number of digits, add 0 if not even
    If (Len(DataToEncode) Mod 2) = 1 Then DataToEncode = "0" & DataToEncode
    'Assign start and stop codes
    StartCode = ChrW(203)
    StopCode = ChrW(204)
    StringLength = Len(DataToEncode)
    For I = 1 To StringLength Step 2
        'Get the value of each number pair
        CurrentCharNum = Val((Mid(DataToEncode, I, 2)))
        'Get the ASCII value of CurrentChar
        If CurrentCharNum < 94 Then DataToPrint = DataToPrint &
        ChrW(CurrentCharNum + 33)
        If CurrentCharNum > 93 Then DataToPrint = DataToPrint &
        ChrW(CurrentCharNum + 103)
    Next I
    'Get Printable String
    PrintableString = StartCode + DataToPrint + StopCode
    'Return PrintableString
    I2of5 = PrintableString
End Function
```