

Impresora Fiscal



Comunicación y Protocolo Fiscal

Se reservan todos los derechos. Ninguna porción de esta publicación podrá reproducirse, guardarse en un sistema de recuperación de datos o transmitirse en forma o modo alguno, sea electrónico, mecánico, fotocopiado, grabado o de otra forma, sin el permiso previo y por escrito de **ANDRÉS MORETTI E HIJOS S.A.** No se asume responsabilidad de patente con respecto al uso de la información contenida aquí. Si bien se han tomado todas las precauciones en la preparación de este libro, **ANDRÉS MORETTI E HIJOS S.A.** no asume ninguna responsabilidad por errores u omisiones. Tampoco asume responsabilidad alguna que surja por el uso de la información contenida aquí.

Ni **ANDRÉS MORETTI E HIJOS S.A.** ni sus compañías afiliadas serán responsables ante el comprador de este producto o ante terceros por daños, pérdidas, costos, o gastos incurridos por el comprador o por terceros como resultado de: accidente, uso indebido, o abuso de este producto o de modificaciones, reparaciones o alteraciones no autorizadas de este producto, o por no cumplir estrictamente con las instrucciones de operación y mantenimiento emitidas por **ANDRÉS MORETTI E HIJOS S.A.**

ANDRÉS MORETTI E HIJOS S.A. no será responsable de cualquier daño o problema que surja por el uso de cualquier opción o producto de consumo que no esté designado como Producto MORETTI Original o Genuino o bien Aprobado por escrito por **ANDRÉS MORETTI E HIJOS S.A.**

GENESIS es una marca registrada de **ANDRÉS MORETTI E HIJOS S.A.** en Argentina.

AVISO: El contenido de este manual está sujeto a cambios sin previo aviso.

Revisión A, Febrero del 2018

Copyright © 2018 Andrés Moretti e Hijos S.A, Buenos Aires, Argentina.

INDICE

DESCRIPCIÓN GENERAL	4
COMUNICACIÓN POR USB.....	5
INSTALACIÓN DEL DRIVER USB	5
EJEMPLO DE UTILIZACIÓN	6
COMUNICACIÓN POR RED ETHERNET	8
DIRECCIÓN IP y PUERTO	8
EJEMPLO DE UTILIZACIÓN	9
COMUNICACIÓN POR PUERTO SERIAL.....	10
CABLE DE COMUNICACIÓN	10
EJEMPLO DE UTILIZACIÓN	11
PROTOCOLO DE COMUNICACIÓN	12
EJEMPLO:	13
ESTADO del HARDWARE	14
ESTADO FISCAL	14
INFORMACIÓN ADICIONAL	15
CABECERAS, COLAS E INFORMACIÓN DEL ESTABLECIMIENTO	15
FORMATEO DE TEXTO	16
IMPRESIÓN DE CÓDIGOS DE BARRA	17
IMPRESIÓN DE LOGOS GRÁFICOS	18
CÓDIGOS de ERROR	19
DETALLE de los COMANDOS.....	19

DESCRIPCIÓN GENERAL

GRACIAS - y felicitaciones por haber seleccionado la Impresora Fiscal **GENESIS**. La Impresora Fiscal **GENESIS** es una impresora de tecnología de impresión térmica de una sola estación, para imprimir comprobantes fiscales y no fiscales homologados.

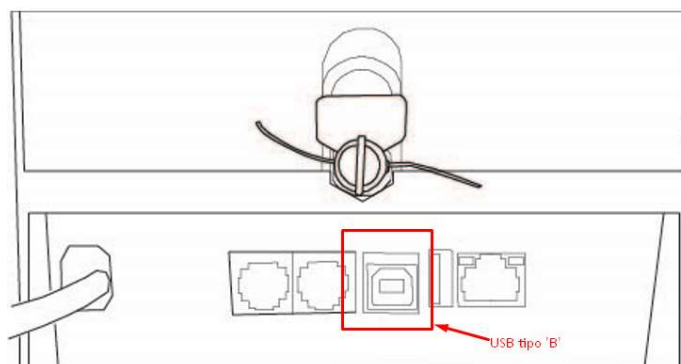
Éste controlador fiscal dispone, en su parte trasera, 3 conectores a través de los cuáles se efectúa la comunicación entre el computador y el controlador fiscal.

Los 3 puertos están disponibles en todo momento, al punto que una operación fiscal (por ejemplo, un ticket), podría abrirse por un port, llenarse por otro y cerrarse por el tercero. La única restricción es que no se puede acceder a más de un puerto por vez.

El flujo de información y la conformación de los paquetes de comunicación es **exactamente la misma** por cualquiera de los 3 puertos, lo que simplifica la programación de la aplicación de punto de venta.

COMUNICACIÓN POR USB

El controlador fiscal Génesis se puede usar a través del puerto USB tipo B. Es el tipo de comunicación preferida y, en la caja, se entrega un cable de USB tipo B listo para ser usado.



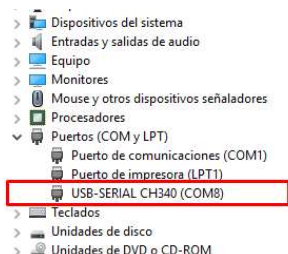
INSTALACIÓN DEL DRIVER USB

Antes de poder usar el puerto USB, Usted deberá instalar el driver del puerto USB, que emula puerto serial COM.

Éste driver debe ser descargado desde http://www.moretti.com.ar/genesis/driver_usb.zip. En éste archivo se incluyen los drivers para Windows de 32 y 64 bits, compatibles desde Windows XP en adelante:

USB TO RS232 driver.rar - archivo RAR, tamaño descomprimido 18.991.523 octetos						
Nombre	Tamaño	Comprimido	Tipo	Modificado	CRC32	
..			Folder			
VCP_V1.3.1_Setup.exe	6.497.077	5.796.523	Application	23/7/2010 10:0...	7841A9DB	
VCP_V1.3.1_Setup_x64.exe	6.497.077	5.796.558	Application	23/7/2010 10:1...	8B244D0C	
VCPDriver_V1.1_Setup.exe	5.997.369	5.333.027	Application	13/6/2008 3:03 ...	7D23DCFB	

Una vez instalado el driver, **conecte el controlador fiscal Génesis y enciéndalo**. Puede consultar por la presencia del driver usando el administrador de dispositivos de Windows:



IMPORTANTE:

- El driver solo aparecerá en el administrador de dispositivos de Windows cuando el controlador fiscal esté conectado y encendido (a diferencia de otros drivers existentes en el mercado, que muestran el driver como “desconectado”)
- Siempre conéctese con éste driver usando 115200:n,8,1 como parámetros de comunicación.

EJEMPLO DE UTILIZACIÓN

A continuación se muestra un ejemplo básico de comunicación con el CF Génesis, usando C# con el .NET Framework versión 2.0 o superior. (Todos estos ejemplos se pueden descargar desde el sitio de Moretti: <http://www.moretti.com.ar/genesis/ejemplos.zip>).

La siguiente clase (*PFTools*) contiene rutinas de utilidad para preparado del buffer, como se verá en las siguientes secciones. La que importa, por ahora, es *'PreparaBuffer'*:

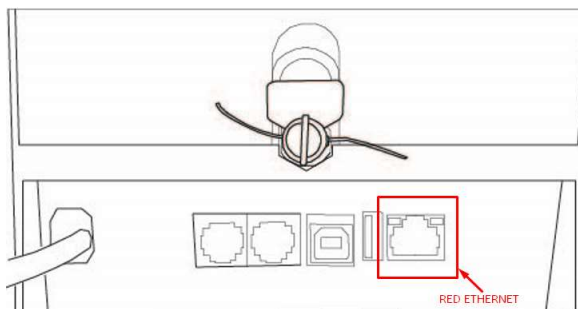
```
1  using System;
2  using System.Collections.Generic;
3
4  namespace TestCommon
5  {
6      public static class PFTools
7      {
8          // Constantes usadas en la conformación del bloque
9          const byte STX = 0x02;
10         const byte ETX = 0x03;
11         const byte SEP = 0x1C;
12
13         /// <summary>
14         /// Prepara un comando traduciendo desde una string a su formato nativo binario.
15         /// </summary>
16         /// <param name="cmd">El comando como una string. Las reglas son:
17         /// - Los bloques se separan con '|', que se traduce como 0x1C en binario.
18         /// - Los primeros 2 bloques (comando y extensión) se traducen como binario.
19         /// - El resto de los argumentos se pega en ascii.
20         /// </param>
21         /// <returns>
22         /// El comando, codificado en binario.
23         /// </returns>
24         static public byte[] PreparaBuffer(string cmd)
25         {
26             List<byte> bin = new List<byte>();
27             var parts = cmd.Split('|');
28
29             bin.Add(STX); // Comienzo del frame
30             bin.Add(0xaa); // Número de secuencia
31
32             int code = Convert.ToInt32(parts[0], 16); // Código de comando
33             bin.Add((byte)(code / 0x100));
34             bin.Add((byte)(code & 0xff));
35
36             bin.Add(SEP);
37             int exten = Convert.ToInt32(parts[1], 16); // Extensión
38             bin.Add((byte)(exten / 0x100));
39             bin.Add((byte)(exten & 0xff));
40
41             for (int i = 2; i < parts.Length; i++) // Otros argumentos del comando
42             {
43                 bin.Add(SEP);
44                 foreach (var ch in parts[i])
45                     bin.Add((byte)ch);
46             }
47             bin.Add(ETX);
48
49             // Agrega el checksum
50             int chksum = 0;
51             bin.ForEach(b => chksum += b);
52             foreach (char ch in chksum.ToString("X").PadLeft(4, '0'))
53                 bin.Add((byte)ch);
54             return bin.ToArray();
55         }
56     }
57 }
```

A continuación, la siguiente (clase principal de una aplicación de consola), emitirá un ticket de prueba (*RippleTest*). Se presupone que, como se describió antes, el port USB se mapeó sobre COM8:

```
1 using System.Collections.Generic;
2 using System.IO.Ports;
3 using System.Threading;
4 using TestCommon;
5
6 namespace TestPuertoUSB
7 {
8     class Program
9     {
10         /// <summary>
11         /// Prueba la comunicación con el Genesis usando el puerto USB.
12         /// </summary>
13         ///
14         /// <remarks>
15         /// Después de instalar el driver, podemos observar que el puerto USB se mapeó
16         /// sobre el COM8.
17         /// </remarks>
18         static void TestUSB()
19         {
20             // Abre el puerto usando 115200:n,8,1
21             using (var port = new SerialPort("com8", 115200, Parity.None, 8, StopBits.One))
22             {
23                 // Completa la inicialización del puerto.
24                 port.ReadTimeout = 2000; // 2 segundos
25                 port.ReceivedBytesThreshold = 1;
26                 port.Open();
27
28                 // Envía el comando para emitir un 'ripple test' (0204|0000|0)
29                 var bin = PFTools.PreparaBuffer("0204|0000|0");
30                 port.Write(bin, 0, bin.Length);
31
32                 // Espera un poco, para dar tiempo a la impresión del ticket.
33                 Thread.Sleep(3000);
34
35                 // Lee la respuesta (descartándola, por ahora)
36                 var respuesta = new List<byte>();
37                 while (port.BytesToRead > 0)
38                     respuesta.Add((byte)port.ReadByte());
39
40                 // No es necesario. Solo se hace para completar el ejemplo
41                 port.Close();
42             }
43
44             /// <summary>
45             /// Lanza la prueba USB.
46             /// </summary>
47             /// <param name="args"></param>
48             static void Main(string[] args)
49             {
50                 TestUSB();
51             }
52         }
53     }
54 }
```


COMUNICACIÓN POR RED ETHERNET

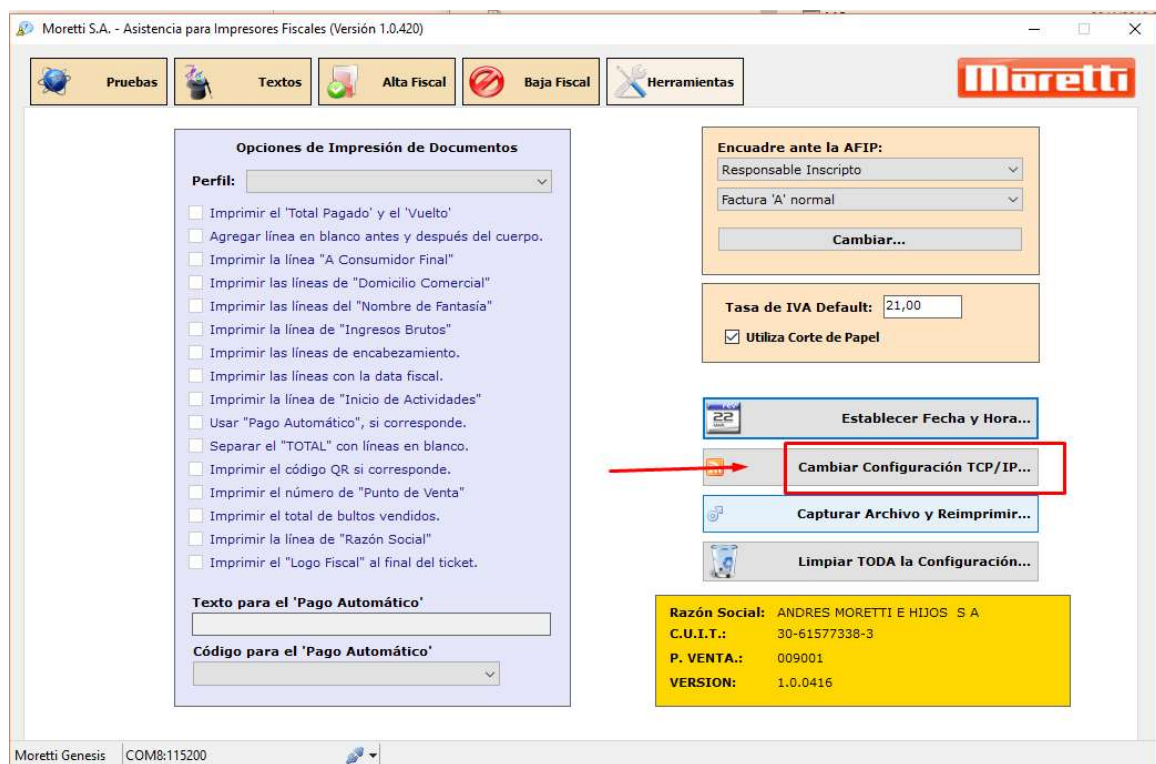
El controlador fiscal Génesis se puede usar a través del puerto de red Ethernet. Para esto, conecte el impresor a un *switch* o *hub* de su red usando un cable de red estándar. Configure la dirección IP (como se indica más abajo) y envíele comandos usando sockets.



DIRECCIÓN IP y PUERTO

El controlador fiscal Génesis se comunica, en red, a través del puerto **5003** (el puerto es fijo, no se puede cambiar).

La dirección IP por defecto, de fábrica, es **192.168.0.90**. Para cambiar la dirección IP, necesitará conectarse al controlador fiscal usando el puerto USB (o el puerto serial). Una vez conectado por USB, puede usar **MAsist** para configurar los parámetros de red o, sino, cambiar la configuración por sus propios medios usando el comando **05C1**.



IMPORTANTE: Personal técnico también puede configurar los parámetros TCP/IP usando el menú LCD, después de romper el precinto fiscal y acceder al menú de servicio técnico.

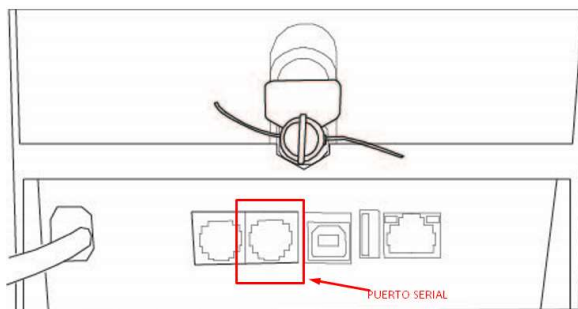
EJEMPLO DE UTILIZACIÓN

Como en el ejemplo anterior, se usa C# con el .NET Framework versión 2.0 o superior. Éste ejemplo también utiliza la clase **PFTools** que se mostró en el ejemplo anterior y aquí se omite:

```
1  using System.Collections.Generic;
2  using System.Net.Sockets;
3  using System.Threading;
4  using TestCommon;
5
6  namespace TestPuertoETHERNET
7  {
8      class Program
9      {
10         /// <summary>
11         /// Prueba la comunicación con el Genesis usando el puerto de red ETHERNET.
12         /// </summary>
13         ///
14         /// <remarks>
15         /// Se utiliza la dirección estándar del impresor (192.168.0.90)
16         /// </remarks>
17         static void TestETHERNET()
18         {
19             // Abre el socket a usar
20             using (var port = new TcpClient())
21             {
22                 // Completa la inicialización del puerto.
23                 port.Connect("192.168.0.90", 5003);
24                 var tcp = port.GetStream();
25
26                 // Envía el comando para emitir un 'ripple test' (0204|0000|0)
27                 var bin = PFTools.PreparaBuffer("0204|0000|0");
28                 tcp.Write(bin, 0, bin.Length);
29
30                 // Espera un poco, para dar tiempo a la impresión del ticket.
31                 Thread.Sleep(3000);
32
33                 // Lee la respuesta (descartándola, por ahora)
34                 var respuesta = new List<byte>();
35                 while (port.Available > 0)
36                     respuesta.Add((byte)tcp.ReadByte());
37
38                 // No es necesario. Solo se hace para completar el ejemplo
39                 port.Close();
40             }
41         }
42     }
43
44     /// <summary>
45     /// Lanza la prueba por puerto ethernet.
46     /// </summary>
47     /// <param name="args"></param>
48     static void Main(string[] args)
49     {
50         TestETHERNET();
51     }
52 }
53 ..
```

COMUNICACIÓN POR PUERTO SERIAL

El controlador fiscal Génesis se puede usar a través del puerto serial. Para esto, conecte el impresor a su computador usando un cable serial (no provisto, ver más abajo).



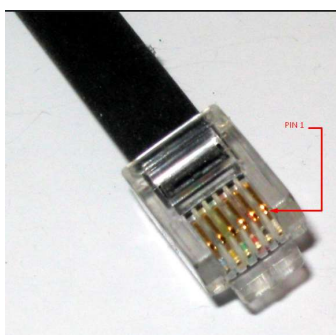
IMPORTANTE: El conector SERIAL y el conector para la gaveta usan el mismo tipo de conector. **Preste extrema atención donde conecta el cable serial.**

CABLE DE COMUNICACIÓN

Puede adquirir el cable serial en un representante de Moretti S.A., o construirlo usando los siguientes lineamientos:

Materiales Necesarios:

- Un conector RJ11 (6 pines)
- Un conector DB9-Hembra
- Cable blindado plano de 3 pares como mínimo – Máximo 4 metros.
- Pinza crimpadora para RJ6.



Cableado entre Conectores	
Pin RJ-11	Pin DB-9
5	2
4	3
3	5

NOTAS:

- Si el cable que usó tiene blindaje, por favor suelde el cable de blindaje al chasis del conector DB9-Hembra.
- Trate de usar un cable plano multifilar, para mejorar el conexionado del lado del conector RJ-11.
- Por favor, no olvide la recomendación efectuada más arriba respecto de prestar atención a no conectar éste cable, por error, en el conector de la gaveta.

EJEMPLO DE UTILIZACIÓN

Puede utilizar exactamente el mismo ejemplo mostrado para el uso del puerto USB, simplemente reemplazando el nombre del puerto USB (**com8**) por el nombre del puerto serial (por ejemplo, **com1**).

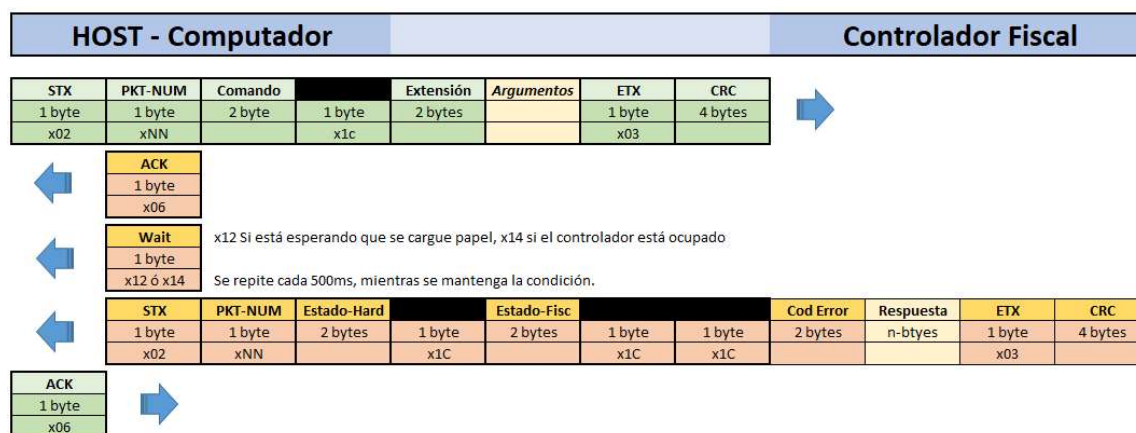
PROTOCOLO DE COMUNICACIÓN

El computador (host) inicia la comunicación con el controlador fiscal, enviándole un paquete de datos en binario. El controlador fiscal acepta el paquete con un carácter ACK e inicia el proceso del comando.

Si durante el proceso del comando el controlador se quedara sin papel, o si el proceso del comando se hace largo, el controlador fiscal enviará un carácter WAIT cada 500ms, para indicar que está procesando.

Completado el proceso requerido, el controlador fiscal enviará un paquete de vuelta con la respuesta al comando enviado. **Como mínimo, siempre se devuelven los campos de Estado-Hard, Estado-Fiscal y Error.**

El siguiente gráfico ilustra el intercambio de datos entre el host y el controlador fiscal:



INDICACIONES:

- **STX (x02), ETX (x03), ACK (x06), NAK (x15), WAIT1 (x12), WAIT2 (x14):** Códigos de control ASCII usados durante el armado de los paquetes.
- **PKT-NUM:** Número correlativo entre x80 y xFF inclusive.
- **Comando, Extensión, Argumentos:** Ver la tabla de comandos más adelante.
- **CRC:** Suma simple de todos los bytes enviados desde el primer x02 (STX), puestos en hexadecimal y completados con '0' a la izquierda hasta completar 4 caracteres. Si la suma superara los 4 caracteres, solo se envían los últimos 4.
- **Estado-Hard:** Estado actual del hardware. Ver más abajo el detalle.
- **Estado-Fisc:** Estado fiscal actual. Ver más abajo el detalle.
- **Cod-Error:** Código de error. Ver más abajo la tabla de errores y su significado. 0 significa que no hubo error.
- **Respuesta:** Campos opcionales dependiendo del comando al que se esté respondiendo. Ve la tabla de comandos más adelante.
- **Escape:** Si entre los caracteres STX y ETX debe enviarse un byte de datos del mismo valor de los indicados en el primer punto de ésta lista (por ejemplo, el x03), entonces previo a ellos debe enviarse el carácter de ESC (x1B). Esto incluye al carácter x1B en sí mismo. El procesador del CF toma como byte de datos cualquier byte posterior a un byte x1B.

EJEMPLO:

A continuación se muestran los bytes intercambiados para consultar al controlador fiscal su fecha y hora actual:



ESTADO del HARDWARE

Como parte de todas sus respuestas, el controlador fiscal incluye 2 bytes donde codifica, en diferentes bits, el estado del equipamiento:

Estado Hard (Equipamiento)	
Bit	Significado
15-11	Reservado (siempre en 0)
12	0 – Gaveta cerrada 1 – Gaveta abierta
11-2	Reservado (siempre en 0)
1-0	00 - Sin problemas 01 - Reservado 10 - Sin papel 11 – Reservado

ESTADO FISCAL

Como parte de todas sus respuestas, el controlador fiscal incluye 2 bytes donde codifica, en diferentes bits, el estado fiscal y operativo del equipo

Estado Fiscal (Operación)	
Bit	Significado
15-14	Modo Operativo: 00 – Reservado 01 – Modo fábrica 10 – Modo de entrenamiento 11 – Modo fiscalizado (Normal)
13	Solo en Modo Entrenamiento: 0 – No hay <i>SBF previo</i> disponible. 1 – Si hay <i>SBF previo</i> disponible.
12	Modo Servicio Técnico: 0: Desactivado 1: Activado
11-10	00 - Sin problemas 01 - Reservado 10 - Sin papel 11 - Reservado
9-8	Estado de la Memoria Fiscal: 00 – Sin problemas 01 – Cerca de llenarse 10 – Llena 11 – Reservado
7	Jornada Fiscal: 0 – Cerrada 1 - Abierta
6-4	Reservados (siempre en 0)
3-0	Documento actualmente abierto: 0000 – Ninguno 0001 – Ticket a consumidor final 0010 – Ticket factura 0011 – Ticket Nota de Crédito 0100 – Ticket Nota de Débito 1001 – Documento no Fiscal Otras combinaciones de bits están reservadas

INFORMACIÓN ADICIONAL

CABECERAS, COLAS E INFORMACIÓN DEL ESTABLECIMIENTO

El controlador fiscal permite que le configuren mensajes para las cabeceras, colas e información del establecimiento. **Estos mensajes no se pueden cambiar en cualquier momento**, más abajo se indica.

Puede usar la aplicación **MA²sist** para efectuarla o, sino, enviando al controlador fiscal los comandos apropiados:

[illegible]

El orden de impresión de estas líneas es el que sigue. **Tenga en cuenta que no se imprimen líneas vacías:**

DESCRIPCIÓN	PERMISOS
Nombre de Fantasía 1	Solo después de una Zeta
Nombre de Fantasía 2	
Nombre de Fantasía 3	
Datos Fiscales del propietario (Razón Social, CUIT, Condición ante el IVA, etc.)	Consulte al servicio técnico antes de cambiar alguno de éstos datos. Hay requisitos fiscales a cumplir.
Domicilio 1	Solo después de una Zeta
Domicilio 2	
Domicilio 3	
Líneas de cabecera (10 líneas, de la 0 a la 9)	Solo con operación cerrada (entre ticket y ticket)
Datos de la Venta	
Líneas de cola (10 líneas, de la 0 a la 9)	Solo con operación cerrada (entre ticket y ticket)

Importante: Es habitual para los software de punto de venta el cambiar “al vuelo” los valores de la línea de cola, antes de cerrar un ticket (por ejemplo, para imprimir el nombre del cajero o para imprimir un código de barras con el total de la operación). Pero, como se indicó más arriba, estos comandos no están disponibles con operación abierta.

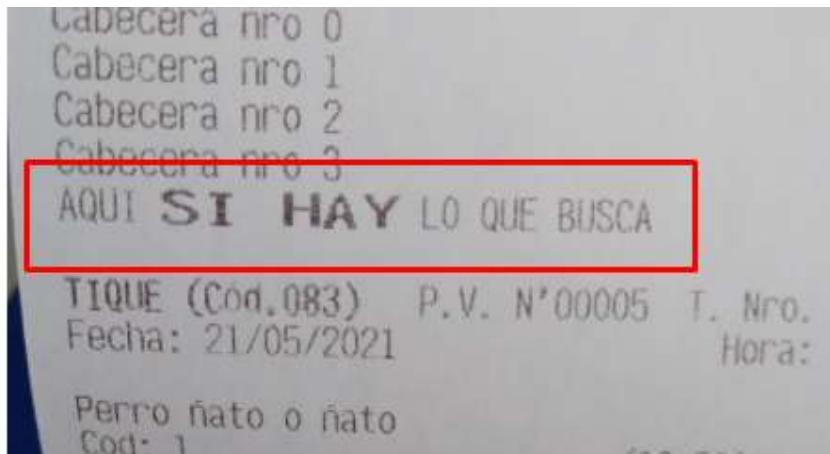
Es por esto que **TODOS** los comandos de cierre de operación disponen de parámetros que permiten reconfigurar momentáneamente hasta 3 líneas de colas. Consulte la documentación de los comandos de cierre de operación.

FORMATEO DE TEXTO

Para darle formato las líneas indicadas antes (cabeceras, colas, etc.), **como así también las descripciones de los artículos**, se usan códigos de formato embebidos dentro del mismo texto. La primera aparición de un código de formato lo dispara y la segunda aparición lo cancela, y así en más.

En el siguiente ejemplo usamos los códigos `\n\ a` para iniciar la impresión en doble ancho y letra negrita, y los códigos `\n\ a` (de nuevo) para desactivar éstos formatos. Recuerde que puede activar/desactivar un formato en el mismo texto la cantidad de veces que lo necesite:

AQUI \n\ aSI HAY\n\ a LO QUE BUSCA



CÓDIGOS DE COMANDO	
COMANDO	DESCRIPCIÓN
<code>\a</code>	Comienza o finaliza la impresión en DOBLE ANCHO.
<code>\n</code>	Comienza o finaliza la impresión en LETRA NEGRITA
<code>\s</code>	Comienza o finaliza la impresión en SUBRAYADO
<code>\x</code>	Vuelve a letra normal
<code>\h</code>	Inicia la impresión en DOBLE ALTO. A diferencia de los códigos anteriores, éste código solo puede usarse al comienzo de una línea y no se puede desactivar.

IMPRESIÓN DE CÓDIGOS DE BARRA

Tanto las líneas de cabecera como de pie pueden imprimir códigos de barra. Como se indicó antes, las líneas de cola se pueden modificar con ticket abierto enviando los datos para la cola en el comando de cierre de ticket. Así, por ejemplo, un software de punto de venta podría desear imprimir un código EAN13 con el total del ticket al pie del mismo.

Para hacerlo, la línea que contendrá la barra debe tener el siguiente formato:

\c	<formato>	<datos>
----	-----------	---------

- \c** Obligatorio al comienzo de la línea. Indica que sigue la definición de código de barra
- <formato>** Indica el tipo de código de barra a emitir, ver tabla siguiente.
- <datos>** Datos a imprimir en el código de barras. **IMPORTANTE:** Los datos del código a imprimir deben contener toda la información necesaria, incluyendo dígitos verificadores calculados por el software de punto de venta. Si el dato no fuera válido, se imprimirá una línea vacía.

FORMATO	TIPO BARRA
A ó a	UPC-A
B ó b	UPC-E
C ó c	EAN 13
D ó d	EAN 8
E ó e	CODE 39
F ó f	ITF
G ó g	CODA BAR
H ó h	CODE 39 SHORT
I ó i	CODE 128

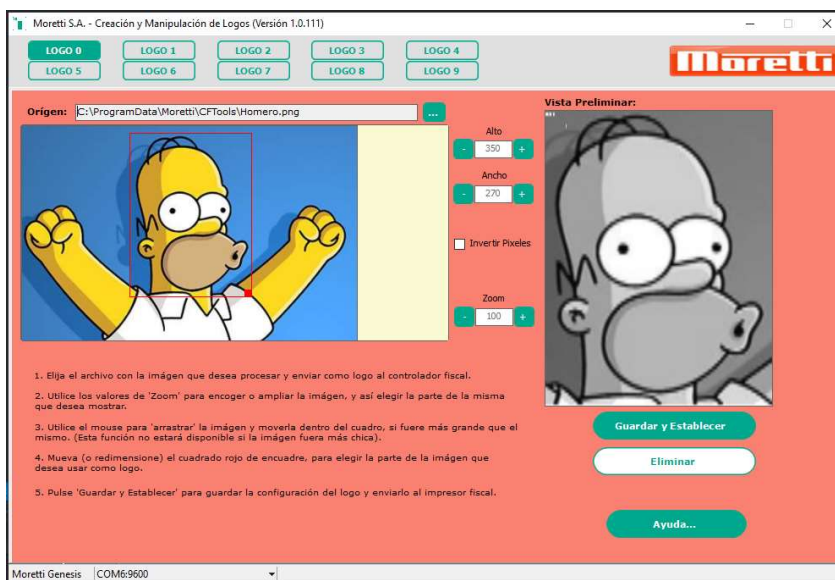
EJEMPLO: Para imprimir un código EAN-13:

\cC070330176591



IMPRESIÓN DE LOGOS GRÁFICOS

El controlador fiscal Génesis puede contener la definición de hasta 10 logos gráficos. Utilice la aplicación MLogos para definirlos y almacenarlos en la memoria de trabajo del controlador fiscal. **Los logos se definen y cargan una sola vez** y, posteriormente, se controla cuando imprimirlos configurando un código especial en las líneas de cabecera o pie.



Para imprimir un logo, debe configurar **\L** seguido del logo a imprimir. Por ejemplo, si en la línea 2 de encabezado se configurara:

\LO



CÓDIGOS de ERROR

Como parte de la respuesta a cada comando, el controlador fiscal devuelve un código de error numérico. Para consultar el significado de cada código, consulte el documento *"Códigos_de_Error.html"* en la carpeta *"Protocolo"*

DETALLE de los COMANDOS

El detalle de conformación de cada comando está disponible como documento separado, en la carpeta *"Protocolo"* (*"Detalle_del_Protocolo.html"*)