

# Lenguaje de programación de Zebra

Grupo ATIA

## 1 Antecedentes

El **lenguaje de programación de Zebra** (ZPL, *zebra programming language*) es un **lenguaje de descripción de página** <sup>1</sup> propietario de la marca Zebra y es utilizado principalmente en aplicaciones de etiquetas. El lenguaje original fue reemplazado por ZPL II, sin embargo, no es completamente compatible con la versión anterior.

Más tarde, el **intérprete de BASIC de Zebra** (ZBI, *Zebra BASIC interpreter*) fue integrado en el software de impresoras y es visto como una mejora de ZPL II por los desarrolladores y está orientado a ANSI BASIC. Su principal ventaja es que evita una reestructuración de código cuando se cambia de impresora en el caso el *software* de la impresora anterior hubiera sido hecho por algún competidor.

En otras palabras, para fines prácticos modernos **se debe de utilizar el formato ZBI en lo mayor posible**.

## 2 Fundamentos de ZBI

ZBI-Developer es una IDE (*Integrated Development Environment*, entorno de desarrollo integrado) diseñado para ayudar en la creación, prueba y distribución de programas escritos en el formato ZBI. El IDE ofrece las siguientes funcionalidades:

- Control de proyectos.
- Ayuda de código por *pop-ups*.
- Impresoras virtuales para pruebas.
- Modos de escritura y *debug*.
- Distribución de programas a impresoras.
- Encriptación de programas para protección de datos propietarios.
- Importación de archivos desde impresoras hacia el IDE de desarrollo.
- Comparación de archivos.

---

<sup>1</sup>Un **lenguaje de descripción de página** es un lenguaje que describe la apariencia de una página impresa en un nivel más elevado que una imagen en mapa de bits. Un par de ejemplos son PCL de HP, PostScript.

## 2.1 Disponibilidad de versiones

El lenguaje **ZBI** consta de la versión 1.0 y la versión 2.0, dependiendo del modelo de la impresora puede que exista solo soporte para la versión 1.0 o para la versión 2.0. Para determinar la versión se puede consultar la siguiente referencia:

### 2.1.1 Para ZBI versión 1.0 a 1.5

La versión 1.x de ZBI está disponible en impresoras que cuentan con un *firmware* con versión X.10 o superior (por ejemplo V48.10.x). Otra forma de determinar si ZBI versión 1.x es soportado es verificando la ausencia de una letra ‘Z’ en el nombre del *firmware*, es decir la versión V.60.13.0.13 soporta ZBI, mientras que V60.13.0.12Z no lo soporta. La siguiente lista muestra los modelos de impresoras que soportan ZBI 1.x.

- LP/TLP 284x-Z and 384x-Z
- S300/S400/S500/S600
- Z4000/Z6000
- Z4M/Z6M
- Z4Mplus/Z6Mplus
- 105SL
- PAX3
- XiII
- XiIII

### 2.1.2 Para ZBI versión 2.0 y superior.

Las impresoras con el firmware con versión X.16 o superior (por ejemplo V60.16.X y V53.16.x) pueden soportar ZBI versión 2.0 y superior. Las siguientes impresoras soportan ZBI 2.0:

- XiIII Plus
- Z4Mplus/Z6Mplus
- 105SL
- S4M
- PAX4
- ZM400/ZM600

El soporte de ZBI puede no estar habilitado por defecto, caso en el cual puede ser habilitado por medio del administrador de licencias. Un archivo **ZBI.nrd** es necesario para que ZBI 2.0 sea habilitado, este archivo está localizado en la memoria ‘E:’ de la impresora y no puede ser borrado o modificado.

### Servidores de impresión soportados:

Los siguientes servidores de impresión soportan ZBI:

- ZebraNet 10/100 Print Server (firmware V1.1.6 required to support on-printer debugging)
- ZebraNet Wireless Print Server (V60.16.x, V53.16.x or later firmware required)
- ZebraNet Wireless Plus Print Server (V60.16.x, V53.16.x or later firmware required)

## 3 Generalidades

- Se debe crear un nuevo proyecto para habilitar la edición del archivo ZBI.
- Haciendo click derecho en la barra izquierda del editor se pueden habilitar los breakpoints.

### 3.1 Impresoras virtuales

Las impresoras virtuales se pueden configurar dando click-derecho en el menu y seleccionando la opción: **Edit Virtual Printer**. De esta manera se puede seleccionar el puerto al que la salida de la impresora debe ser redirigida, si no existen medios físicos para enviar o recibir datos a los puertos las salidas se deben dejar con la opción **Comm Window**.

**Nota:** La principal desventaja de las impresoras virtuales es que no tienen la capacidad de mostrar una imagen a partir de los comandos de impresión.

#### 3.1.1 Estableciendo conexión con impresoras

Una vez que se ha encontrado una impresora física o definido una impresora virtual es necesario establecer una conexión de *debuggeo* utilizando los siguientes pasos:

1. Navegar a la pestaña de vista de impresoras.
2. Click derecho sobre la impresora deseada.
3. Escoger la opción **Create debug connection**.

Una vez que la conexión ha sido establecida aparecerá el ícono de *debuggeo* al lado de la impresora deseada. La conexión se intentará mantener incluso después de reiniciar el *software*.

**Nota:** Solo impresoras con ZBI habilitado y versión de software v60.16.x y v53.16.x o superior soportan *debuggeo* en la impresora física.

#### 3.1.2 Ejecutando un programa

Escribir el siguiente código en el editor:

```
10 close all
20 open #1: name "SER"
30 open #2: name "ZPL"
40 input #1: A$
50 print #2: "~xa~fo50,50~a0n,50,50~fd"&A$&"~fs~xz"
60 goto 40
```

Note que cada línea de código contiene un número, dicho número es necesario para la ejecución del programa. Si se cambia a la vista de *debuggeo* y se da click en el botón de ejecutar, se puede simular la entrada de datos por medio de las ventanas de los puertos correspondientes.

El programa hace lo que sigue:

1. Línea 10: Cierra todos los puertos abiertos.
2. Línea 20: Abre el puerto serial y lo asigna a el puerto '#1'.
3. Línea 30: Abre el puerto ZPL y lo asigna al puerto '#2'.
4. Línea 40: Espera una entrada de datos (comunicación) del puerto '#1', es decir por el puerto serial.
5. Línea 50: Imprime la entrada recibida en conjunto con otra sintáxis de impresión en formato ZPL.
6. Línea 60: Regresa a la línea 40 a esperar un nuevo dato.

El programa se ejecutará de manera permanente hasta que se detenga de manera manual. El comando ZPL ejecutado contiene los siguientes comandos:

- ^XA:** Representa el inicio de un nuevo formato de impresión. Todos los comandos para impresión en ZPL II deben comenzar con este comando.
- ^FO:** *Field origin* Representa el origen con respecto al *HOME* de la etiqueta. En este caso establece el origen de impresión en el punto (50,50).
- ^A:** Establece el tipo de letra a utilizar, en este caso utiliza el tipo de letra **0**, con orientación **normal** y un largo y ancho de letra de  $50 \times 50$ .
- ^FD:** *Field Data* Define la cadena de caracteres a escribir en la impresión hasta un máximo de 3072 bytes. Cualquier caracter puede ser imprimible, excepto los caracteres de escape correspondientes a ZPL, tales como '^' y '~'.
- ^FS:** *Field Separator* Denota el final de un comando ^FD.
- ^XZ:** Denota el final de un formato de impresión. Todos los comandos para ZPL II deben terminar con este comando.