DRIVER ZEBRA ZPL II

***elipse**software*

Filename:	ZebraZPLII.DLL
Vendor:	Zebra Technologies Corporation
Devices:	Zebra printers supporting ZPL II or EPL II language
Protocol:	Zebra Programming Language II (ZPL II) and EPL II
Version:	v2.00
Latest release:	2005.11.01
Platform:	Win32
Dependencies:	IOKit v1.06 or higher

Introduction

This driver was initially developed to allow communication between Elipse systems and Zebra printers via ZPL II language. Since it is a generic driver to be implemented, it was later discovered that it can also be used with printers supporting EPL II language. The driver has not been tested with other languages yet, although it is likely that they can also be supported.

From this driver's version 2.00 on, part of its features are shared with IOKit component, allowing the configuration to be easier and more versatile, including the possibility of communication via Ethernet interface, supported by some new models of printers. To know more about IOKit, a thorough and careful reading of its user's manual is highly recommended.

Equipment configuration

The examples in this section refer only to its use with ZPL II language. However, the use of EPL II language is analogous.

The driver needs a configuration file. This file will contain ZPL II (Zebra Programming Language II) instructions that describe the labels format. To create this file, open the driver Extra Settings window (Other Parameters in E3) and click on tab *"ZebraZPLII"*.

Each label must start with the keyword **[BEGIN]** and end with **[END]**. These keywords define a block that contains the ZPL II program created by the user, describing the label.

It's possible to create variables identified by tokens, which will be automatically replaced by the driver, before being sent to the printer. A token is represented by &n, where n is a positive integer (for example: &0, &1, &2). These variables can be defined inside the Elipse by using PLC tags N1 = 0, as described below and can hold either numeric values or strings.

To insert comments in ZPL code, use // symbol. The driver will ignore the rest of the line.

See the example (from ZPL II Language Programming Guide):

```
// Label 0:
[BEGIN]
^XA
// ^XA – Indicates start of label format
^LH&0,&1
// ^LH – Sets label home position &0 dots to the right and &1 dots down from top edge of label
// (&0 and &1 are variables defined in Elipse in printing-time)
^FO&2,&3^AD^FDZEBRA^FS
// ^FO&2,&3 – Set field origin &2 dots to the right and &3 dots down from the home position
// defined by the ^LH instruction
//^AD – Select font "D"
//^FD – Start of field data
//ZEBRA – Actual field data
//^FS – End of field data
^FO&4,&5^B3^FDAAA001^FS
// ^FO&4,&5 – Set field origin &4 dots to the right and &5 dots down from the home position
// defined by the ^LH instruction
// ^B3 – Select Code 39 bar code
// ^FD – Start of field data for the bar code
// AAA001 – Actual field data
// ^FS – End of field data
^XZ
// ^XZ – Indicates end of label format
[END]
```

In the example above, only the characters without the double slashes (//) on the left will be sent to the printer. All tokens beginning with an ampersand (&) will be replaced with the values of the respective driver variables (0 to 50). The default value of all variables (before any operation) is -1, so it is recommended to initialize every variable used whenever necessary.

Several labels can be defined in the same file. The order each label appears in the configuration file will be used to select them when printing (by using the N2 parameter), beginning with zero. So the first label is zero, the second is 1, and so on.

This driver (when loaded into a new application project) comes pre-configured with the following parameters: Serial port: *Com1*; Baud rate: *9600bps*; Data bits: *7*; Stop bits: *2*; Parity: *Even*; Timeout: *1000ms*. For any change in these configurations, as well as for editing any other, it is necessary to access the driver's extra settings dialog window. This dialog window is explained on IOKit user's manual.

Driver communication [P] parameters

P1 Not used (remains zero).

P2 Not used (remains zero).

P3 Not used (remains zero).

P4 Not used (remains zero).

PLC tags [N] parameters

N1 Function (0 = Internal variables reading/writing; 1 = label printing)

N2 See table 2.

N3 See table 2.

N4 See table 2.

The parameters N2, N3 and N4 will vary as described in the following table:

Table 2 – [N] parameters for printer functions

	Internal variables reading/writing (N1 = 0)	Label printing (N1 = 1)
N2	Variable number (0 to 50)	Number of the label to be printed (0 = 1st; 1 = 2nd, …)
N3	Not used	Number of decimals (0 = none; 1 = 9.9; 2 = 9.99 etc.)
N4	Not used	Not used

Notes

- All values are rounded for printing, as set in N3 parameter.
- The tokens *&n* (&1, &2, &3, &4 ...) are replaced with the values of driver defined variables before labels are sent to printer.
- From this driver's version 1.1 on, not only numbers but also strings can be defined as internal variables values.

Release History

Versão	Data	Autor	Comentários
v2.00	2005.11.01	M. Ludwig	- Driver ported to IOKit (Case 5672).
v1.01	2005.07.12	A. Quites	- Fixed bug that caused the driver to send the [BEGIN] token of the first label to the printer, in case there were empty or commented line before it (case 5855) - Version tested for compatibility with the EPL II language (case 5547) - Driver modified to support strings as internal variables values (case 5547) - Problem keeping "ZPL II Configuration dialog" in Extra Settings from being opened in E3 fixed (Case 5881) - Other updates regarding code from the latest Elipse DDK
v1.00	2003.11.28	A. Quites	- Driver's original version