

DRIVER ZEBRA ZPL II

elipsesoftware

Nome do arquivo:	ZebraZPLII.DLL
Fabricante:	Zebra Technologies Corporation
Equipamentos:	Impressoras Zebra que suportem as linguagens ZPL II e EPLII
Protocolo:	Zebra Programming Language II (ZPL II) e EPL II
Versão:	v2.00
Última atualização:	2005.11.01
Plataforma:	Win32
Dependências:	IOKit v1.06 ou superior

Introdução

Este driver foi desenvolvido para permitir a comunicação dos sistemas Elipse com impressoras Zebra através do protocolo (linguagem) ZPL II. Como se trata de um driver bastante genérico em sua implementação, descobriu-se mais tarde que ele também pode ser usado com a linguagem EPL II. Este driver ainda não foi testado com outras linguagens, embora haja a possibilidade de que ele venha a suportá-las.

A partir da versão 2.00 deste driver, parte de suas funcionalidades são compartilhadas com o componente IOKit (permitindo maior versatilidade e facilidade de configuração), incluindo a possibilidade de comunicação por interface Ethernet, suportada por alguns modelos novos de impressora. Para saber mais sobre o IOKit, recomenda-se a atenta leitura de seu respectivo manual do usuário.

Configuração do Equipamento

Os exemplos desta seção referem-se apenas ao uso da linguagem ZPL II. Entretanto, o uso da linguagem EPL II é análogo.

Para usar este driver é necessário criar o arquivo de configuração no formato ZPL II (*Zebra Programming Language II*) contendo a descrição das etiquetas a serem impressas. Este arquivo deve ser criado e editado na janela de configurações **Extras** do driver (“Outros parâmetros” no E3), clicando-se na aba “*ZebraZPLII*”.

Cada etiqueta deve começar pela palavra-chave **[BEGIN]** e terminar com **[END]**. Dentro de cada um destes blocos, definidos por estas duas palavras-chave, deve existir um programa ZPL II criado pelo usuário, de acordo com a etiqueta que se deseja gerar.

É possível criar variáveis identificadas por “*tokens*”, que serão substituídos no momento da impressão. Um *token* é composto por &*n*, onde *n* é um inteiro positivo (por exemplo: &0, &1, &2). Estas variáveis podem ser definidas dentro do Elipse através de tags PLC N1 = 0, segundo as configurações vistas adiante, e podem conter tanto valores numéricos como strings.

Pode-se inserir comentários dentro do código ZPL usando-se duas barras //, o que faz o driver considerar como comentário todo o restante da linha.

Veja o exemplo a seguir, tirado do Guia de Programação da linguagem ZPL II:

```
// Etiqueta 0:
[BEGIN]
^XA
// ^XA - Indica o início da formatação de etiqueta
^LH&0,&1
// ^LH - Ajusta a posição inicial da etiqueta para &0 pontos para a direita e &1 pontos partindo
// borda superior da etiqueta (&0 e &1 são variáveis definidas no Elipse no momento da impressão)
^FO&2,&3^AD^FDZEBRA^FS
// ^FO&2,&3 - Ajusta a origem do campo para &2 pontos para a direita e &3 pontos abaixo partindo
// da posição inicial (definida pela instrução ^LH) (&2 e &3 são variáveis)
// ^AD - Seleciona a fonte "D"
// ^FD - Início dos dados do campo
// ZEBRA - Os dados propriamente ditos (palavra "ZEBRA")
// ^FS - Fim dos dados do campo
^FO&4,&5^B3^FDAAA001^FS
// ^FO&4,&5 - Ajusta a origem do campo para &4 pontos para a direita e &5 pontos abaixo partindo
// da posição inicial (definida pela instrução ^LH) (&4 e &5 são variáveis)
// ^B3 - Seleciona a fonte de código de barras "Code 39"
// ^FD - Início dos dados do campo para o código de barras
// AAA001 - Dados propriamente ditos ("AAA001")
// ^FS - Fim dos dados do campo
^XZ
// ^XZ - Indica o fim da formatação de etiqueta
[END]
```

No exemplo acima, somente os caracteres sem o início // serão enviados à impressora. Todos os *tokens* iniciando por '&' serão substituídos por uma das variáveis internas do driver (0 a 50), as quais podem ser lidas e escritas com os tags PLC N1 = 0. O valor inicial padrão (antes de qualquer operação) de todas as variáveis é -1. Por isso, deve-se inicializar a variável sempre que necessário.

Várias etiquetas podem ser definidas no mesmo arquivo. A ordem em que cada etiqueta aparece no arquivo de configuração será usada para selecioná-la no momento da impressão, através do parâmetro N2 do tag PLC com N1 = 1, iniciando em zero; ou seja: a primeira é 0, a segunda é 1, e assim por diante.

Este driver (ao ser carregado em um novo projeto) já vem pré-configurado com os seguintes parâmetros: Porta Serial: *Com1*; Velocidade: *9600bps*; Bits de dados: 7; Bits de parada: 2; Paridade: *Par*; Timeout: *1000ms*. Para qualquer mudança nestas configurações, assim como para editar quaisquer outras, é necessário acessar a janela de diálogo de parâmetros extras do driver. Esta janela de diálogo é explicada no manual do usuário do IOKit.

Parâmetros [P] de comunicação do driver

- P1** Não utilizado (manter em zero).
- P2** Não utilizado (manter em zero).
- P3** Não utilizado (manter em zero).
- P4** Não utilizado (manter em zero).

Parâmetros [N] de endereçamento de tags PLC

- N1** Função (0 = leitura/escrita de variáveis internas; 1 = impressão de etiquetas)
- N2** Ver tabela 2.
- N3** Ver tabela 2.
- N4** Ver tabela 2.

Os parâmetros N2, N3 e N4 variam de acordo com a função designada: leitura/escrita de variáveis ou impressão de etiquetas, conforme a tabela abaixo.

Tabela 2 - Parâmetros [N] de acordo com as funções da impressora

	Leitura/Escrita de Variáveis Internas (N1 = 0)	Imprimir Etiquetas (N1 = 1)
N2	Número da variável (0 a 50)	Número da etiqueta a ser impressa (por exemplo: 0 = 1ª. etiqueta; 1 = 2ª. etc.)
N3	Não usado	Número de casas decimais a serem impressas (0 = sem decimais; 1 = 9,9; 2 = 9,99 etc.)
N4	Não usado	Não usado

Notas

- Todos os números são arredondados para impressão, ajustando-se ao indicado no parâmetro (N3).
- Os tokens &n (&1, &2, &3, &4 ...) são substituídos pelos valores das variáveis definidas no Elipse na hora da impressão.
- A partir da versão 1.1 desse driver, além de números, pode-se também definir strings como valores das variáveis internas.

Histórico das revisões do driver

Versão	Data	Autor	Comentários
v2.00	2005.11.01	M. Ludwig	- Driver portado para o IOKit (case 5672).
v1.01	2005.07.12	A. Quites	- Corrigido bug que permitia o envio do token [BEGIN] do primeiro label para a impressora, caso existissem linhas de comentário ou vazias antes do mesmo (case 5855) - Versão testada com linguagem EPL II (case 5547) - Implementação do suporte a strings como valores de variáveis internas (case 5547) - Resolvido bug que impedia a abertura da janela de configuração "ZPL II Configuration" nas configurações extras do driver, quando usando o E3 (Case 5881) - Outras atualizações relativas ao DDK mais recente
v1.00	2003.11.28	A. Quites	- Versão original do driver