

Impressão de Código de Barras

Uma abordagem prática Argox PPLA

por Victory Fernandes e Augusto Frederico

Na Internet estão disponíveis diversos componentes para geração e impressão de Código de Barras utilizando o *Delphi* e impressoras Laser ou Jato de Tinta. Ao trabalharmos com diferentes impressoras, tivemos alguns problemas quanto à resolução, configuração de impressoras, impressão nas margens das etiquetas, impressão de grandes quantidades de etiquetas e etc, o que comprometia a confiabilidade do sistema.

Devido a estes e outros problemas, consideramos que a maneira mais simples, prática e profissional de se imprimir códigos de barras, é utilizando impressoras específicas para impressão de etiquetas e códigos de barras.

Neste artigo fazemos uma abordagem prática para as impressoras Argox, que utilizam linguagem de script PPLA, após este estudo, o leitor estará familiarizado com o conceito geral, e pronto para utilizar qualquer impressora do gênero.

Acessando a impressora

Caso deseje, o usuário da impressora pode utilizar softwares prontos disponíveis para download que acompanham a impressora ou são vendidos separadamente. Este não é o tipo de abordagem que desejamos aos nossos clientes. O ideal é que sejamos capazes de oferecer a funcionalidade de impressão de código de barras por dentro dos nossos programas, utilizando uma interface com a qual o usuário já esteja familiarizado. Para isso é necessário conhecer o formato de comunicação com a impressora.

A impressora em questão não requer nenhum tipo de *driver* específico para funcionar, bastando que lhe seja passada uma seqüência de comandos *ASCII* pela porta a qual esta conectada. Sendo assim o acesso pode ser feito até mesmo pelo *MS-DOS* com um comando do tipo:

```
COPY "MEU_ARQUIVO.TXT" LPT1
```

ou

```
COPY "MEU_ARQUIVO.TXT" COM1
```

Onde "MEU_ARQUIVO.TXT" é um arquivo de texto comum contendo o script em PPLA que se deseja executar e deve ser enviado para a porta paralela (LPT1) ou serial (COM)

Este tipo de abordagem de comunicação facilita, e muito, a implementação, uma vez que a nossa única preocupação será conhecer os comandos, utilizando os métodos padrões do Delphi para o envio das *strings*.

Conhecendo a Linguagem

Os comandos em PPLA obedecem uma sintaxe geral do tipo:

- A Comandos
- p1,p2,p3 Parâmetros
- [p1,p2,p3] Parâmetros opcionais

Apesar da impressora disponibilizar uma infinidade de comandos e opções, tais como contadores, criação e armazenamento de formulários em memória, velocidade de impressão dentre outros. Vamos abordar alguns itens principais:

- Configuração da Etiqueta
- Impressão de Textos
- Impressão de Códigos de Barras

Sempre que desejarmos imprimir uma determinada etiqueta devemos seguir um procedimento geral indicado pela impressora:

- Inicialmente declaramos a unidade de comprimento que iremos utilizar, através do comando *STX m* (milímetro) ou *STX n* (polegadas)
- Configuramos o comprimento da etiqueta, Comando *Stx c*
- A linguagem PPLA necessita de um comando com o intuito de entrar no modo de formatação da etiqueta, utilizamos o comando *STX L*.
- Fixamos a temperatura com o *Comando H*.
- Torna-se necessário fazer a configuração das dimensões da etiqueta com a qual estamos trabalhando – *Comando D*.
- Após esse processo, a impressora estará pronta para receber os comandos relativos a montagem da nova etiqueta - Comandos de Texto e Código de Barras.
- Uma vez concluída a montagem da etiqueta, determina-se quantas vezes a impressora deve repetir a impressão da etiqueta criada - *Comando Q*.
- É definida uma pausa no intervalo de impressão entre as etiquetas - *Comando J*.
- Por fim o *Comando E* para impressão e retorno do modo de sistema.

Veremos agora a análise de cada um dos comandos citados separadamente:

Definição da Unidade de Comprimento

Antes de passarmos qualquer comando de impressão é necessário definirmos a unidade de comprimento que iremos trabalhar a partir do momento. Geralmente, o comando *STX m*, que não aceita nenhum parâmetro e determina a unidade milímetros, porém também é usado o comando *STX n* se o programador utilizar a unidade polegadas. A partir disso todos os valores passados à impressora serão interpretados de acordo com a escolha de um dos comandos acima.

Tamanho da etiqueta

Em seguida definiremos o tamanho da etiqueta com o comando *STX c* que aceita uma variável inteira determinando o comprimento do papel.

Modo de Formato

O comando *STX L*, desprovido de parâmetros, possui a função de habilitar a impressora a receber os comandos para a formatação da etiqueta.

Temperatura

Fixaremos a temperatura padrão da cabeça de impressão através do *Comando H*, que possui uma variável com valor padrão igual a 10 e um máximo recomendado de 16.

A temperatura controla o contraste da impressão. Devem ser considerados fatores como tipo de papel e tipo de *ribbon* para ajuste da qualidade da impressão.

Configuração da Etiqueta

Para configuração da largura e altura da etiqueta, utilizamos o *Comando D*, cuja sintaxe é mostrada abaixo:

Dp1,p2

Onde,

- p1 Largura da etiqueta em *pixels*.
- p2 Altura de da etiqueta em *pixels*.

Variável	Valores
p1	1,2
p2	1,2,3

Obs: os valores em negrito são considerados padrões.

Impressão de Textos

Para impressão de textos formatados na etiqueta, utilizamos os comandos de Texto, cuja sintaxe é mostrada abaixo:

p1,p2, p3,p4, p5,p6, p7, "TEXTO"

Onde,

- p1 Rotação

Valor	Descrição
1	Sem rotação
2	Rotação de 90°
3	Rotação de 180°
4	Rotação de 270°

Tabela 01. Valores válidos de rotação

- p2 Fonte

Caractere	Tipo de subfonte	Tipo de fonte	
0,1,2,3,4,5,6,7,8	000	Fontes 0 a 8	
9	000 até 007	ASD Smooth *	004: 12 pontos
		000: 4 pontos	005: 14 pontos
		001: 6 pontos	006: 18 pontos
		002: 8 pontos	007: 24 pontos
		003: 10 pontos	
9	Xxx	Seleção de fonte PCL. O xxx representa os 3 dígitos decimais de identificação.	
:	000,001,002,003,004,005	Fontes Courier** O 00x representa os tipos abaixo 000 – Roman 8 001 – ECMA-94 002 – PC set 003 – PC set A 004 – PC set B 005 – Legal	

Tabela 02. Valores válidos de Tipos de fontes

- p3 Valor de expansão horizontal do texto (Valores de 0 a 9 e A a O).
- p4 Valor de expansão vertical do texto (Valores de 0 a 9 e A a O).
- p5 Subtipo_Fonte (ver tabela 1)
- p6 Eixo Y em milímetros
- p7 Eixo X em milímetros.

	Mínimo	Máximo
Eixo X	0	101.6
Eixo Y	0	762

Tabela 03. Valores válidos dos Eixos

“TEXTO” Informação a ser impressa.

A *Figura 01* exemplifica o script de geração de um texto:

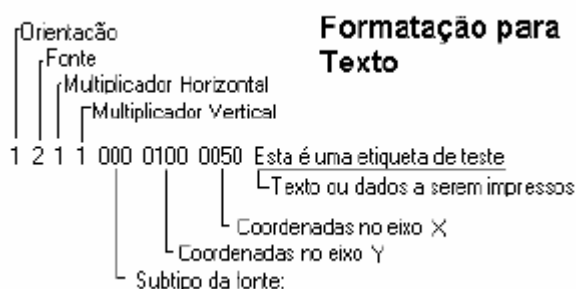


Figura 01. Configuração do Texto

Impressão de Código de Barras

Para impressão de códigos de barras nos mais diversos formatos, utilizamos o comando de Código de Barras, cuja sintaxe é mostrada abaixo:

p1,p2, p3,p4, p5,p6, p7, “VALOR”

Onde,

- p1 Rotação

Valor	Descrição
1	Sem rotação
2	Rotação de 90°
3	Rotação de 180°
4	Rotação de 270°

Tabela 04. Valores válidos de Rotação

- p2 Tipo do Código de Barras
A .. T = Código de barras com caracteres compreensíveis pelos humanos.
a .. t = Código de barras sem caracteres compreensíveis pelos humanos.
- p3 Espessura da barra fina (Valores de 0 a 9 e A a O).
- p4 Espessura da barra larga (Valores de 0 a 9 e A a O).
- p5 Altura do código de barras (valor 000 padrão)
- p6 Eixo_Y em milímetros
- p7 Eixo_X em milímetros

	Mínimo	Máximo
Eixo X	0	101.6
Eixo Y	0	762

Tabela 05. Valores válidos dos Eixos

- “VALOR” Informação a ser impressa em código de barras

A *Figura 02* exemplifica o script de geração de um Código de Barras:

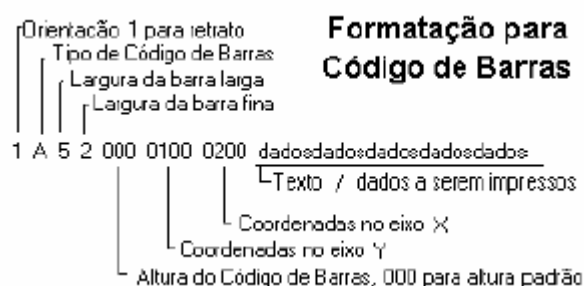


Figura 02. Código de Barras

Comandos Finais da Etiqueta

Após passarmos todos dados para a etiqueta determinamos a quantidade de etiqueta a ser impressa pelo *Comando Q*. É necessário também determinarmos uma pausa entre uma impressão e outra, *Comando J*.

Para finalizarmos, através do *Comando E*, indicamos o fim do modo de formatação de etiquetas, Impressão e retorno para o modo de sistema.

O Programa Exemplo

Veja na *Figura 03* a tela do aplicativo demonstrativo criado:



Figura 03. Tela principal do exemplo de utilização de Impressora de Código de Barras

Este programa exemplifica a impressão de códigos de barras. Nele podemos:

- Configurar todos os parâmetros de formatação dos comandos Texto, Código de Barras e Q.
- Adicionar e excluir produtos para impressão.
- Visualizar a codificação em PPLA gerada para impressão das etiquetas.
- Salvar um arquivo de script PPLA para impressão futura.
- Abrir um arquivo de script PPLA para impressão.
- Imprimir códigos de barras em qualquer impressora paralela compatível com PPLA

Nele foram implementadas as chamadas às funções de impressão de texto e código de barras, descritas anteriormente, como mostrado:

```
Function Comando_STX_m: String;
Function Comando_STX_c(p1: string): String;
Function Comando_STX_L: String;
Function Comando_H(p1:string) String;
Function Comando_D(p1, p2: string): String;
Function Texto_PPLA(p1,p2,p3,p4,p5,p6,p7, 'Texto':string): String;
Function Codigo_Barras_PPLA(p1,p2,p3,p4,p5,p6,p7, 'Valor':string): String;
Function Comando_Q(q1): String;
Function Comando_STX_J(j1): String;
```

```
Function Comando_E: String;
```

O programa faz sucessivas chamadas à estas funções, de acordo com a quantidade de produtos contida no componente *TListView* e a quantidade de etiquetas por produto definida pelo usuário, para gerar o script em PPLA que é enviado para o componente *TMemo*.

Uma vez o *script* em PPLA pronto, utilizamos uma função de escrita na porta da impressora para enviar os comandos contidos no componente *TMemo* para a impressora instalada na porta paralela, como mostrado:

```
procedure DirectPrint(s: String);
var
  PTBlock : TPassThroughData;
begin
  PTBlock.nLen := Length(s);
  StrPCopy(@PTBlock.Data, s);
  Escape(printer.handle, PASSTHROUGH, 0, @PTBlock, nil);
end;

procedure TForm1.Button2Click(Sender: TObject);
begin
  //impressao do codigo de barras
  if memol.Lines.Count > 0 then
  begin
    Printer.BeginDoc;
    DirectPrint(memol.Lines.Text);
    Printer.EndDoc;
  end;
end;
```

A PPLA_Unit

As chamadas dos comandos da impressora demonstradas durante o artigo foram feitas utilizando a *PPLA_Unit.pas*, uma *Unit* desenvolvida em *Delphi* de acordo com o manual de referência da linguagem PPLA para geração de script utilizado nas impressoras de código de barras *Argox*.

- As vantagens de utilizar a *PPLA_Unit* incluem:
- Completa abstração da camada de geração do *script*, sendo necessário apenas fazer chamadas às funções da *Unit* para os comandos desejados.
- Velocidade na implementação da comunicação com a impressora
- Geração de etiquetas com texto e código de barras de forma muito simples
- Facilidade na geração de código PPLA

Maiores informações sobre a *PPLA_Unit* podem ser obtidas no site do produto em http://www.igara.com.br/produto.php?cod_produto=83

Conclusão

Agora que você está familiarizado com as principais opções das impressoras, fica muito mais fácil partir para implementações mais elaboradas, que utilizem outros recursos não abordados neste artigo, como opções de contadores, temporizadores, impressão de imagens e logomarcas, impressão de formulários etc...

Maiores informações sobre impressoras de códigos de barras, bem como download de manuais, citado neste artigo, podem ser encontradas no site do fabricante em <http://www.argox.com/>

Maiores informações sobre a *PPLA_Unit* podem ser obtidas no site do produto em http://www.igara.com.br/produto.php?cod_produto=83

Victory Fernandes é Engenheiro Mestrando em Redes de computadores, e desenvolvedor sócio da TKS Software - Soluções de Automação e Softwares Dedicados. Pode ser contactado em victory@igara.com.br, ou através dos sites www.victory.hpg.com.br - www.enge.cjb.net – www.igara.com.br.

Augusto Frederico é estudante de Engenharia Mecatrônica e desenvolvedor da TKS Software - Soluções de Automação e Softwares Dedicados. Pode ser contactado em fredygbi@yahoo.com.br