

DRIVER VESDD

ACESSO A I/O NO WINDOWS 95

1. INTRODUÇÃO

No Windows 95, o acesso ao I/O do microcomputador pelo programa aplicativo é bloqueado pelo sistema operacional. Desse modo as instruções **INP** e **OUT** existentes no Basic foram eliminadas no Visual Basic para Windows 95/NT. A instrução **Port** existente no Borland Delphi também foi eliminada nas versões posteriores do Delphi.

O driver VESDD é um driver VxD para o Windows 95 que implementa funções para o acesso ao I/O de placas para barramento ISA. Basicamente o driver VESDD e a respectiva biblioteca DLL implementam funções para leitura e escrita de I/O.

O VESDD pode ser utilizado para acessar qualquer endereço de I/O do barramento ISA como por exemplo os I/O das placas ESD3201 e ESD6401. **O driver não faz nenhuma verificação do endereço de I/O a ser acessado pelo programa aplicativo.** Para saber quais endereços de I/O estão sendo utilizados por outros drivers do Windows 95, consulte o programa *Sistema* do *Painel de Controle* do Windows 95. Nesse programa, selecione o ícone *Computador* da pasta *Gerenciador de Dispositivos* e em seguida clique sobre o botão *Propriedades*.

1.1. Escopo

O objetivo deste documento é apresentar a interface com o driver realizada por módulo em Borland Delphi e Visual Basic 5.0.

1.2. Aplicabilidade

O driver VESDD.VXD deve ser utilizado somente no sistema operacional Windows 95 ou compatível e com placas de expansão para barramento ISA.

	Elaboração	Aprovação
Área	SW.P&D	SW.P&D
Nome	Lauro	Lauro
Data	03/jun/1999	03/jun/1999
Visto		

Reprodução Proibida

2. INSTALANDO O DRIVER VESDD

Os arquivos fornecidos para utilização do driver VESDD são:

VESDD.SYS	driver VxD para Windows 95
VESDD.DLL	biblioteca DLL de acesso ao driver
VESDDapi.pas	módulo em pascal para a interface com o driver
TESTE.EXE	programa de teste do driver
TESTE.*	arquivo de projeto do programa exemplo em Delphi 2.0
Main.*	formulário e módulo principal do programa exemplo
TesteDIO.*	formulário e módulo de teste de E/S digital
TESTE.VBP	arquivo de projeto do programa exemplo em Visual Basic 5.0
frmMain.*	formulário e módulo principal do programa exemplo em Visual Basic
VESDDapi.bas	módulo do programa exemplo em Visual Basic com a interface com a DLL

A instalação do driver VESDD.VXD é bastante simples, basta copiar os arquivos VESDD.VXD e VESDD.DLL para o diretório onde se encontra o programa aplicativo.

O driver VESDD.VXD é carregado em tempo de execução do programa aplicativo, não sendo necessário nenhum registro no Windows 95 através do REGEDIT.

Para aplicações em Visual Basic, recomenda-se copiar os arquivos VESDD.VXD e VESDD.DLL para o diretório do Windows 95.

3. INTERFACE DO DRIVER COM O PROGRAMA APLICATIVO

O acesso do programa aplicativo ao driver é disponibilizado através do módulo em Pascal VESDDapi. Esse módulo deve ser incorporado ao projeto do programa aplicativo em Borland Delphi. Se você estiver utilizando outra linguagem de programação, basta criar um módulo equivalente ao VESDDapi.pas com as referências à biblioteca VESDD.DLL. A biblioteca DLL deverá ser copiada para o mesmo diretório do programa aplicativo.

É fornecido também um exemplo de programa aplicativo implementado em Visual Basic 5.0. As declarações necessárias para a utilização das funções da biblioteca VESDD.DLL são fornecidas no arquivo VESDDapi.bas. Esse arquivo deve ser incluído no projeto do programa aplicativo em VB5. Para aplicações em Visual Basic, recomenda-se copiar os arquivos VESDD.DLL e VESDD.VXD no diretório do Windows 95.

4. DESCRIÇÃO DAS PRIMITIVAS DO VESDD

Neste tópico são descritas as primitivas de acesso ao driver VESDD. A tabela abaixo lista as primitivas da DLL:

Primitiva	Descrição
VESDD_Open	Inicia o acesso ao driver
VESDD_Close	Finaliza o acesso ao driver
VESDD_GetVersion	Informação a versão e revisão do driver
VESDD_ReadDI_Byte	Leitura de port de entrada de 8 bits da placa ISA
VESDD_ReadDI	Leitura de port de entrada de 16 bits da placa ISA
VESDD_ReadDI_Byte	Escrita em port de saída de 8 bits da placa ISA
VESDD_WriteDO	Escrita em port de saída de 16 bits da placa ISA

As funções da biblioteca DLL são descritas com a sintaxe do Object Pascal e do Visual Basic.

4.1. Primitiva VESDD_Open

Object Pascal:

```
Function VESDD_Open: boolean;
```

Visual Basic:

```
Declare Function VESDD_Open Lib "VESDD.DLL" () As Byte
```

Esta primitiva abre o acesso ao driver e deve ser a primeira primitiva a ser executada pelo programa aplicativo. A primitiva retorna o valor 1 (true) quando a primitiva foi executada com sucesso. Normalmente o erro ocorre quando o driver não foi encontrado no diretório em que se encontra a biblioteca DLL.

Veja também a primitiva *VESDD_Close*.

4.2. Primitiva VESDD_Close

Object Pascal:

```
Procedure VESDD_Close;
```

Visual Basic:

```
Declare Sub VESDD_Close Lib "VESDD.DLL" ()
```

Esta primitiva finaliza o acesso ao driver e deve ser a última primitiva a ser executada pelo programa aplicativo. Para cada chamada bem sucedida da primitiva *VESDD_Open* deve haver uma correspondente chamada da *VESDD_Close*.

Veja também a primitiva *VESDD_Open*.

4.3. Primitiva VESDD_GetVersion

Object Pascal:

```
Function VESDD_GetVersion: word;
```

Visual Basic:

```
Declare Function VESDD_GetVersion Lib "VESDD.DLL" As Integer
```

Esta primitiva devolve a versão do driver. O byte mais significativo do valor retornado corresponde ao número da versão e o byte menos significativo o número da revisão.

4.4. Primitiva VESDD_ReadDI_Byte

Object Pascal:

```
Function VESDD_ReadDI_Byte (rgPort: word): byte;
```

Visual Basic:

```
Declare Function VESDD_ReadDI_Byte Lib "VESDD.DLL" (ByVal rgPort As Integer) As Byte
```

Esta primitiva permite a leitura de um port de entrada da placa ISA. O port a ser lido deve ser passado no parâmetro *rgPort*.

4.5. Primitiva VESDD_ReadDI

Object Pascal:

```
Function VESDD_ReadDI (rgPort: word): word;
```

Visual Basic:

```
Declare Function VESDD_ReadDI Lib "VESDD.DLL" (ByVal rgPort As Integer) As Integer
```

Esta primitiva permite a leitura de um port de 16 bits da placa ISA. O endereço do port a ser lido deve ser passado no parâmetro *rgPort*.

4.6. Primitiva VESDD_WriteDO_Byte

Object Pascal:

```
Procedure VESDD_WriteDO_Byte (rgPort: word; Value: byte);
```

Visual Basic:

```
Declare Sub VESDD_WriteDO_Byte Lib "VESDD.DLL" (ByVal rgPort As Integer, ByVal Value As Byte)
```

Esta primitiva permite a escrita em port de saída da placa ISA. O dado a ser escrito no port deve ser passado no parâmetro *Value* e o número do port no parâmetro *rgPort*.

4.7. Primitiva VESDD_WriteDO

Object Pascal:

```
Procedure VESDD_WriteDO (rgPort: word; Value: word);
```

Visual Basic:

```
Declare Sub VESDD_WriteDO Lib "VESDD.DLL" (ByVal rgPort As Integer, ByVal Value As Integer)
```

Esta primitiva permite a escrita em um port de saída de 16 bits do barramento ISA. O dado a ser escrito no port deve ser passado no parâmetro *Value*. O endereço do port a ser atualizado deve ser passado no parâmetro *rgPort*.