

Especificação Sk_access.dll V1.91



Foto: SKO-44

Teclado de automação, Tecnologia Óptica
com display LCD e leitor magnético.

Descrição:

Sk_access.dll é uma biblioteca de funções desenvolvida pela Smak Teclados, com o objetivo de facilitar o trabalho daqueles que desenvolvem softwares para os produtos SMAK. Atualmente suporta os teclados da linha SKO-44. Este documento destina-se apenas a desenvolvedores e contém as informações técnicas indispensáveis ao uso da DLL.



Especificação Sk_access.dll

Sumário

Histórico de alterações deste documento.....	3
Descrição SK_access.dll	4
Plug and Play.....	4
Demonstração e testes das funções da DLL Smak	6
Instalando a DLL	6
Utilizando a DLL da SMAK em seu ambiente de desenvolvimento	8
Tabela de funções e procedures disponibilizadas por sk_access.dll	9

Histórico de alterações deste documento:Revisão 1.5 (27-02-2010) :

- Revisão de textos.
- Introdução de novas funcionalidades.

Revisão 1.4 (21-07-2008) :

- Introdução de novas funcionalidades de sk_access.dll versão 1.6.

Revisão 1.3 (11-09-2007) :

- Revisão de textos.
- Introdução de novas funcionalidades de sk_access.dll versão 1.5.

Descrição SK_access.dll :

O teclado de automação SKO-44 disponibiliza em sua documentação técnica, uma série de comandos necessários para que softwares possam ser desenvolvidos de modo a utilizar o teclado de forma apropriada. Adicionalmente a SMAK desenvolveu também uma DLL chamada “*sk_access.dll*” cujo objetivo é disponibilizar ao profissional de programação algumas funções e procedimentos essenciais, o que facilita o seu trabalho de desenvolvimento de software.

Ao instalar a DLL da SMAK, dependendo do sistema operacional em uso, será necessário instalar também “*inpout32.dll*” no caso do Windows XP ou *sdrvw9x.vxd*, se for Windows 9x, conforme exibido na tabela a seguir.

Software requerido para instalação de *sk_access.dll* :

WINDOWS -XP

Arquivo	Descrição
inpout32.dll	DLL requerida para uso no Windows XP

WINDOWS-9x:

Arquivo	Descrição
sdrvw9x.vxd	VXD requerido para uso no Windows 9x

Utilitário de testes (Windows-9x/XP):

Arquivo	Descrição
Exemplo_Delphi.exe	Utilitário de teste da DLL Smak (desenvolvido em Delphi-3)
Exemplo_VB.exe	Utilitário de teste da DLL Smak (desenvolvido em Visual Basic 6)

Plug and Play:

A DLL oferece uma plataforma unificada de desenvolvimento, de forma que um aplicativo feito utilizando suas funções, pode acessar indistintamente os teclados nas interfaces PS2, serial RS-232 e USB, sem modificação do código fonte do aplicativo do usuário. Com uma pequena modificação pode ainda se comunicar com os teclados padrão Ethernet.

Quando qualquer função da DLL é chamada, ele executa a Procedure *Select_Interface*.

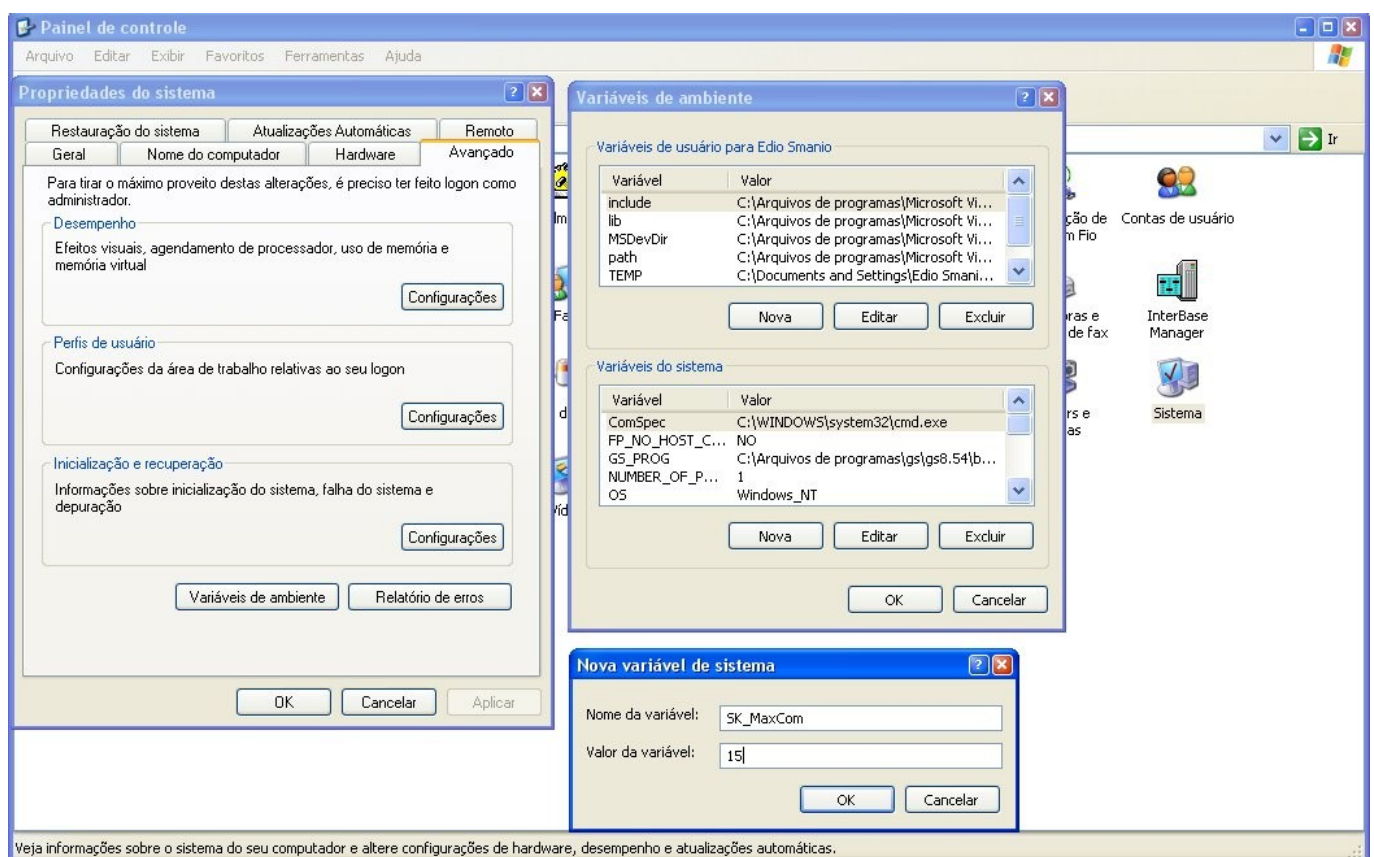
A procedure `Select_Interface` procura por um teclado na interface PS2 e depois nas COM's 1..MAX_COM.

Se o teclado é encontrado em alguma destas interfaces ele é travada e a partir daí todas as operações são formatadas para ela, por exemplo a procedure `DISP(string)` pode mandar uma string para o display de um teclado PS2 ou serial/USB sem que o usuário precise se preocupar com isso.

O valor de MAX_COM por padrão é 8, significa que o teclado pode ser conectado nas COM's 1..8 que a DLL o encontrará, porém em alguns casos podem surgir COM's virtuais com números muito maiores, neste caso é necessário ampliar capacidade de busca da DLL.

A variável de ambiente `SK_MaxCom` redefine a maior porta a ser testada pela DLL, para criar/editar esta variável, no windows, deve-se acessar:

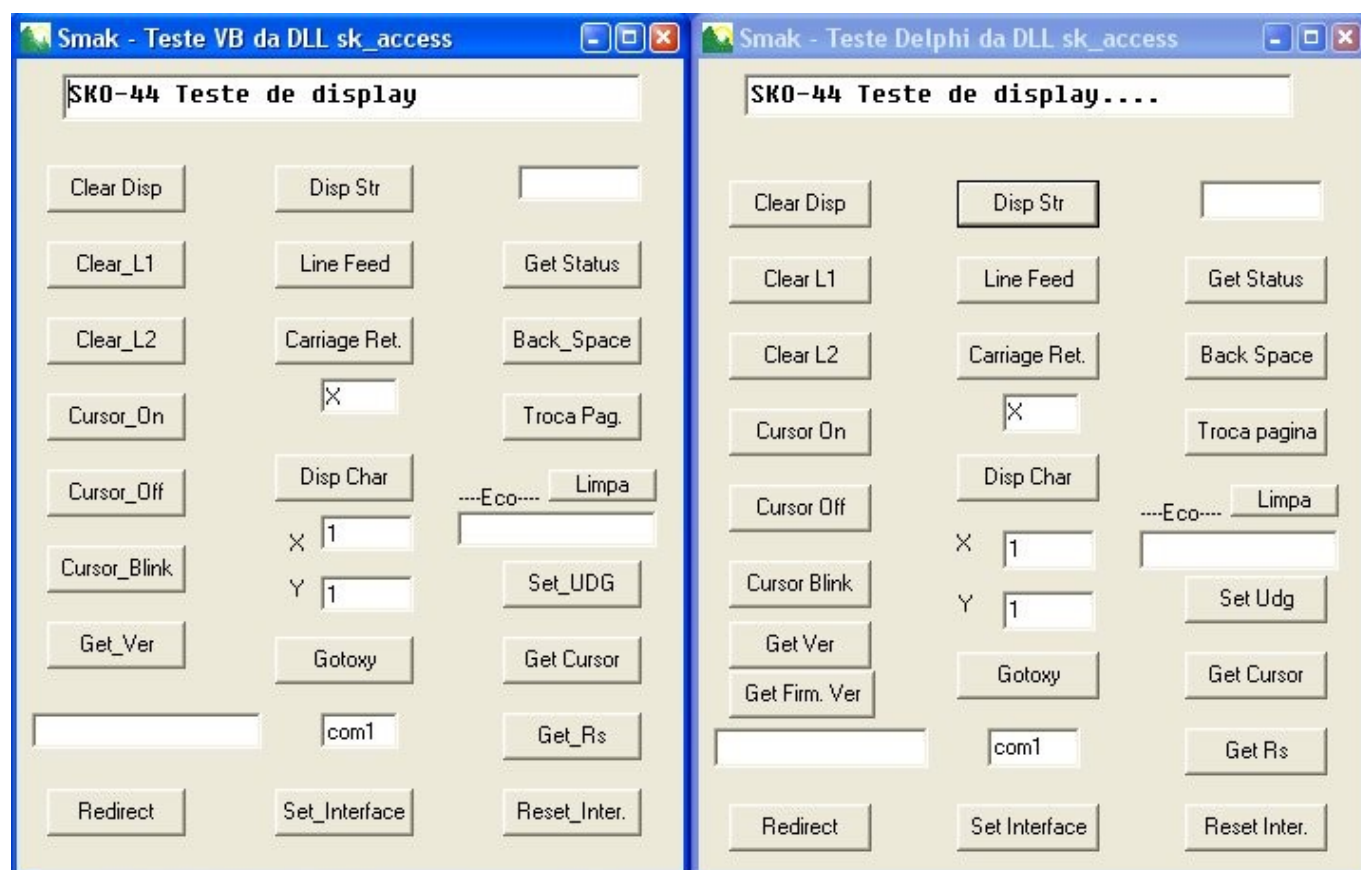
Painel de controle -> Sistema -> Avançado -> Variáveis de ambiente -> Nova/Editar



A Função `Set_Interface` não respeita o limite de `Max_Com`, de forma que `Set_Interface('COM99')`, sempre vai usar COM99 caso esteja disponível.

Demonstração e testes das funções da DLL Smak :

O utilitário de teste disponibilizado em Delphi3 e VB6 (contendo inclusive o código fonte) foi desenvolvido especificamente como um exemplo para verificação da operação das funções exportadas pela DLL. A figura a seguir ilustra sua aparência no Windows. Cada botão tem o nome da função sendo testada. É importante lembrar que Este documento se refere à última versão da DLL sendo que versões anteriores não suportarão todas as funções descritas aqui.

**Instalando a DLL :**

A instalação de consiste em obter o pacote de software, disponível no site www.smak.com.br, descompactar e copiar manualmente os arquivos aos diretórios a que se destinam.

Passos para a instalação Windows 9x ou XP:

1. Descompacte em uma pasta temporária o software que pode ser obtido via download no site www.smak.com.br.

2. Copie as DLLs e o VXD para o diretório "C:\WINDOWS\SYSTEM32".
(sdrv9x.vxd só é necessário ao Windows 9x)

*** **Apenas para Windows-9x:** será preciso adicionar uma configuração na seção [386Enh] do arquivo "C:\WINDOWS\SYSTEM.INI". (se desejar, você pode abrir o system.ini utilizando o sysedit do windows assim: Menu iniciar -> Executar -> sysedit). Editando "SYSTEM.INI" localize a seção [386Enh] e adicione a seguinte linha:

DEVICE=C:\WINDOWS\SYSTEM32\sdrv9x.vxd

3. Crie uma pasta de trabalho por exemplo "C:\SMAK" destinada a receber o utilitário de teste e demais arquivos do SKO44, tais como documentos, exemplos, etc.
4. Testes: Conecte ao computador um teclado SKO-44 em seguida execute o utilitário de testes e em seguida teste alguns comandos.

Exemplos de testes:

- Teste da função DISP: Digite um texto na caixa de texto superior e aperte o botão "Disp Str". O texto digitado deverá aparecer no display do SKO-44.
- Teste da função CLEAR_L1 e CLEAR_L2: Aperte os botões "Clear_L1" e "Clear_L2" para limpar as linhas do display.

Para saber sobre todos os testes consulte a informação na tabela no final deste documento, a qual contém a descrição das funções exportadas pela DLL.

Utilizando a DLL da SMAK em seu ambiente de desenvolvimento :

Uma vez que você conseguiu verificar o correto funcionamento do utilitário de testes, isto significa que *“sk_access.dll”* está corretamente disponível para uso no sistema. É interessante lembrar que em caso de dúvidas sobre o uso das funções da DLL, o profissional desenvolvedor poderá consultar o código fonte do aplicativo de testes disponibilizado em VB6 e Delphi3.

Veja à seguir a tabela com as funções e procedimentos disponíveis na DLL Smak.

Tabela de funções e procedures disponibilizadas por sk_access.dll :

Index	PROTÓTIPO	DESCRIÇÃO
1	Procedure Send_Data(data:byte); stdcall;	Envia um byte para o teclado
2	Procedure Send_Disp_Ctrl(data:byte); stdcall;	Envia comando direto para o display
3	Procedure Clear_Disp; stdcall;	Apagar o display
4	Procedure Clear_L1; stdcall;	Apagar a linha 1 do display
5	Procedure Clear_L2; stdcall;	Apagar a linha 2 do display
6	Procedure Cursor_Off; stdcall;	Desligar o cursor no display
7	Procedure Cursor_On; stdcall;	Ligar o cursor
8	Procedure Cursor_Blink; stdcall;	Piscar o cursor
9	Procedure Back_Space; stdcall;	Back space do cursor
10	Procedure Line_Feed; stdcall;	Line-feed no display
11	Procedure Carriage_Return; stdcall;	Carriage return
12	Procedure Gotoxy(x:byte; y:byte); stdcall;	Posiciona cursor na Linha Y, Coluna X. (sendo: Y: 1ou 2 e X:de 1 a 40)
13	Procedure Disp_Char(data:char); stdcall;	Exibe um caracter no display
14	Procedure Disp(data:PChar); stdcall;	Exibe uma string de até 80 caracteres.
15	Function Get_Cursor(x_pos:pointer; y_pos:pointer):byte; stdcall;	Lê em x_pos e y_pos as coordenadas atuais do cursor (home = 1,1), Adicionalmente retorna a posição absoluta do cursor (0 a 79). Caso o teclado não responda retorna 255.
16	Function Get_Status:byte; stdcall;	Lê status da última operação, Retorna: 00h se não houve erros; 01h se houve erro e FFh se saiu por time-out.
17	Procedure Set_Page1; stdcall	Ativa pagina 1 de scancode.
18	Procedure Set_Page0; stdcall	Ativa pagina 0 de scancode.
19	Procedure Free_Sk_Access; stdcall;	Remove hooks e libera memória
20	Function GetOperatingSystem: Integer; stdcall;	Identifica sistema operacional retorna um valor numérico de -1 a 6 sendo: -1=Desconhecido; 0=Win95; 1=Win98; 2=Win98se; 3=WinME; 4=WinNT; 5=Win2000 e 6=WinXP

Index	PROTÓTIPO	DESCRIÇÃO
21	Procedure Set_Eco_On; stdcall;	Liga eco
22	Procedure Set_Eco_Off; stdcall;	Desliga eco
23	Procedure Set_Udg(caracter,def0,def1,def2,def3, def4,def5,def6,def7:byte); stdcall	<p>UDG (User Defined Graphics) Permite o desenho de até oito caracteres especiais que ficam armazenados nas posições 0 a 7 da tabela interna do display. Na procedure, o parâmetro "caracter" representa o código a ser atribuído ao desenho. "def0" a "def7" definem os elementos ativos da matriz 5x7 do caracter no display.</p> <p>Veja abaixo um exemplo dos valores de "caracter" e de "def0 a 7" para formar a letra 'R' na posição 01h da tabela interna</p> <p style="text-align: center;">caracter=01h</p> <p>def1 = x x x 1 1 1 1 0 = 1Eh def2 = x x x 1 0 0 0 1 = 11h def3 = x x x 1 0 0 0 1 = 11h def4 = x x x 1 1 1 1 0 = 1Eh def5 = x x x 1 0 1 0 0 = 14h def6 = x x x 1 0 0 1 0 = 12h def7 = x x x 1 0 0 0 1 = 11h</p>
25	Function Set_Interface(interf:Pchar):integer; stdcall;	→Para parâmetro "interf"='PS2', 'COM1' a 'COMn', Fixa comunicação nestas interfaces.
		→Para parâmetro "interf"= 'COM', Procura teclado nas interfaces 'COM1' a 'COMn'.
		Retorna: -1 Se interface indisponível 0 Se interface disponível / PS2 1..99 Se teclado ligado à COM(1..99)
26	Procedure Get_Dll_Version(ver:PChar); stdcall;	Devolve na variável especificada, a versão da DLL smak. "ver:PChar" deve ser capaz de acomodar uma string de 20 caracteres.
27	Function Select_Interface:integer; stdcall;	Procura por um teclado SK044 conectado, retorna:-1 se não encontrar; 0 se Encontrar na PS2; 1 .. 99 se encontrar na COM(1..99)
28	Procedure Reset_Interface; stdcall;	Esquece as informações de Interface.
29	Procedure Redirect; stdcall;	Captura a porta serial e redireciona para o buffer do teclado.
30	Function Get_Data_Rs (buffer:pointer ; size:integer): integer; stdcall;	Transfere para "buffer" um máximo de "size" caracteres lidos da porta serial. Retorna o número de caracteres efetivamente lidos.
31	Procedure Keyb_Reset; stdcall	Reseta o teclado.

Index	PROTÓTIPO	DESCRIÇÃO
32	Procedure Get_Firmware_Version(ver:PChar); stdcall;	Retorna string no format (PS2)=v.vvv ou (RS)=v.vvv
33	Function Get_Com_Type:boolean; stdcall;	Retorna tipo da porta COM: TRUE = Virtual, FALSE = Real