

INTERFACE PARA ECF SWEDA

1. SERSWEDA.SYS E S2070.SYS

O device driver SERSWEDA.SYS e S2070.sys foram desenvolvidos para ser a interface de acesso a impressora fiscal SWEDA para aplicativos desenvolvidos em COBOL, CLIPPER,C, DELPHI, VISUAL BASIC etc.

1.1 INSTALAÇÃO DO SERSWEDA.SYS:

- ✓ O device deve ser instalado no arquivo CONFIG.SYS do seguinte modo:

DEVICE = C:\SERSWEDA.SYS /COMX /T04B0

onde : COMX -> indica a porta serial na qual esta conectada a impressora fiscal SWEDA.

T04B0 -> constante de tempo (hexadecimal) múltiplo de 55 ms que e responsável pela atuação dos time-outs de transmissão e recepção.

Constante	Valor em s
T04B0	Aproximadamente 60s
T00C8	Aproximadamente 11s
T0064	Aproximadamente 5s

OBS: DEVERÁ SER MANTIDA A CONSTANTE /T???? NA LINHA DE INSTALAÇÃO.

1.2 INSTALAÇÃO DO S2070.SYS:

- ✓ O device deve ser instalado no arquivo CONFIG.SYS do seguinte modo:

DEVICE = C:\S2070.SYS /T04B0

onde : T04B0 -> constante de tempo (hexadecimal) múltiplo de 55 ms que é responsável pela atuação dos time-outs de transmissão e recepção.

INTERFACE PARA ECF SWEDA

OBS: DEVERA SER MANTIDA A CONSTANTE /T04B0 NA LINHA DE INSTALAÇÃO.

Esta seqüência deverá ser seguida para enviar qualquer comando ao ECF utilizando qualquer um dos dois devices:

Envio do comando ao ECF:

- 1) Abrir o arquivo IFSWEDA ;
- 2) Escrever os comandos desejados no arquivo IFSWEDA;
- 3) Fechar o arquivo IFSWEDA ;

Leitura o status do ultimo comando enviado:

- 1) Abrir o arquivo IFSWEDA ;
- 2) Ler o arquivo IFSWEDA ;
- 3) Fechar o arquivo IFSWEDA.

Caso ocorram time-outs , serão retornados os seguintes status :

TIME-OUT DE TRANSMISSAO : Status = .-P002}

TIME-OUT DE RECEPCAO : Status = .-P006}

Esta interface é compatível para os ambientes DOS e WINDOWS 95.

2. SERSWEDA.DLL

DLL para Comunicação com o ECF SWEDA - Versão 1.00

INTERFACE PARA ECF SWEDA

A SERSWEDA.DLL foi desenvolvida para ser a interface de acesso a impressora fiscal SWEDA para aplicativos desenvolvidos em DELPHI.

2.1 INSTALAÇÃO DO SERSWEDA.SYS:

- ✓ Copie o arquivo IFSWEDA.DLL para um diretório que esteja no Path do seu computador, como por exemplo:
C:\WINDOWS\SYSTEM.

Funções da DLL

1. Função ABREPORTA

São passadas para a função as seguintes variáveis:

- Porta Serial do tipo numérico.
- TimeOut do tipo numérico, valor em segundos que a dll deverá aguardar a resposta do ECF.

A função retorna True ou False, se a porta foi aberta com sucesso ou não.

Seu índice na DLL é 1.

2. Função ENVIACOMANDO

É passada apenas o texto a ser enviado do tipo alfanumérico, respeitando os formatos aceitos pelo ECF SWEDA.

A função retorna a resposta vinda do ECF, conforme formato descrito no manual de programação.

Caso ocorram time-outs , retornará um dos seguintes status :

TIME-OUT DE TRANSMISSAO : Status = .-P002}

TIME-OUT DE RECEPCAO : Status = .-P006}

O índice na DLL é 2.

INTERFACE PARA ECF SWEDA

3. Função FECHAPORTA

É passada apenas a porta serial do tipo numérico.

A função retorna True ou False, se a porta foi aberta com sucesso ou não.

Seu índice na DLL é 3.

Versão 1.00 - 15 de Setembro de 1998.

Esta interface é compatível para os ambientes WINDOWS 95 e WINDOWS 98.

3. SWECEF.DLL

Desenvolvida usando o Microsoft Visual C/C++ versão 5.0, obedecendo a arquitetura de 32 bits a partir do WINDOWS 95. Utiliza apenas as funções da API do Windows com a finalidade de prover acesso a impressora fiscal SWEDA para os aplicativos desenvolvidos em linguagens C e DELPHI. Na linguagem DELPHI, o compilador deverá ser orientado através da diretiva "stdcall" que a passagem dos argumentos da função é no formato Microsoft em vez de Pascal.

Outras linguagens também poderão ter acesso a este programa, desde que tenham uma estrutura definida para acessá-la. Caso não funcione, consultar o manual ou suporte da linguagem.

3.1 INSTALAÇÃO DO SERSWEDA.SYS:

Copiar para o diretório onde ser executada a aplicação.

Caso a linguagem usada pela aplicação tenha sido escrita em "C", efetuar o "LINK" com a biblioteca "SWECEF.LIB".

INTERFACE PARA ECF SWEDA

Funções da DLL

1. int ECFOpen(Porta,Timeout,log,ShowErro)

```
unsigned int Porta    // numero da porta serial 1 = COM1
                                     2 = COM2
                        // não aceita valor zero.
                        // Aceita até COM15(=15).    (32 bits)

unsigned int Timeout  // Tempo em segundos          (32 bits)

unsigned int Log       // 0 = Nao tem log            (32 bits)
                        // 1 = Log semanal(renovado semanalmente)
                        // >= 2 = tamanho máximo em Kbytes do arquivo

unsigned int ShowErro
                        // FALSE= Não mostra caixa de dialogo se faltar
                        //      sinais CTS ou DSR do ECF, ou for
                        //      recebido texto com formato incorreto.
                        // TRUE= Mostra uma caixa de dialogo quando
                        //      ocorrer erro.    (32 bits)
```

A função "ECFOpen()" programa a porta serial selecionada para operar a velocidade de 9600 bits/s, sem paridade, 8 bits e 1 stop bit. O timeout é expresso em segundos. Se houver LOG, ser criado ou aberto um arquivo em modo texto no diretório "\temp". O nome deste arquivo terá o seguinte formato: ECFxxxxxxx.CMy onde <xxxxxxx> é o nome do dia da semana e <y> o numero da porta serial.

O argumento "ShowErro" = TRUE, mostrará caixa de diálogo em todas as condições de erro. Igual a FALSE, deixa de mostrar se houver falta do sinal CTS (ECF desconectado) ou falta de sinal DSR (ECF desligado). Em ambas as situações, não mostrará nenhuma mensagem de erro na tela, caso não tenha recebido o status do ECF. O retorno desta função igual a zero indica que a função conseguiu iniciar a porta serial e todas as suas variáveis.

2. int ECFWrite(comando)

INTERFACE PARA ECF SWEDA

LPSTR comando = endereço de uma "string ASCIIZ"(terminada em caracter null).

Na linguagem DELPHI é identificada com o tipo Pchar. Os dois primeiros caracteres devem ser escape (0x1b) e '.'.

O caracter '}' deve encerrar o comando. O retorno desta função igual a zero indica que a função conseguiu enviar o comando e receber o status do ECF. Esta função pode gerar um dos seguintes textos de erro (ignorar as haspas):

".-P002}" = indica que venceu o tempo de espera de CTS/DSR.

".-P006}" = indica que nao recebeu o status do ECF.

O texto gerado estará disponível para a aplicação, por intermédio da funcao "ECFRead()".

3. int ECFRead(Status, ext)

LPSTR Status = endereço de uma "string ASCIIZ"(terminada em caracter null). Na linguagem DELPHI é identificada com o tipo Pchar.

ext extensão = A aplicação deve informar o tamanho real da área "Status" informada no primeiro argumento. A extensão da área não deve ser inferior a 132 bytes para não ocorrer overflow e erro interno na aplicação.

O retorno desta função diferente de zero, indica que a DLL não foi iniciada com a função ECFOpen(). Receberá uma dos seguintes textos de erro (ignorar as haspas):

".-P002}" = indica que venceu o tempo de espera de CTS/DSR.

".-P006}" = indica que não recebeu o status do ECF.

Quando a aplicação receber um dos dois textos, ela deve enviar o comando <ESC>.23(vide manual de programação) para comparar o numero consecutivo.

INTERFACE PARA ECF SWEDA

Se o consecutivo recebido do ECF for inferior, então o comando não foi processado. Se o número recebido for maior, enviar novamente caso seja início de cupom, comprovante ou relatório. Igual indica que o ECF processou o comando.

Confirmar através do comando ESC.28 se operação foi aberta.

4. int ECFClose()

O retorno desta função diferente de zero, indica que a DLL não foi iniciada com a função ECFOpen(). Esta função libera o uso da serial para outra aplicação.

5. int ECFPathLog(Path)

LPSTR Path = endereço de uma "string ASCIIZ"(terminada em caracter null). Na linguagem DELPHI é identificada com o tipo Pchar.

Informa o "Path" onde se encontra o log usado pela DLL(se habilitado). O retorno desta função diferente de zero, indica que a DLL não foi iniciada com a função ECFOpen().