

BemaFI32.dll

**Documentação da nova DLL para impressora fiscal MP-20 FI II e
MP40 FI II ,versão 3.10 ou superior.**

Revisão 1.9

Índice

1.0 – Arquivo de configuração bemafi32.ini	1
2.0 - Funções de inicialização	3
Bematech_FI_AlteraSimboloMoeda.....	3
Bematech_FI_ProgramaAliquota	3
Bematech_FI_ProgramaHorarioVerao.....	4
Bematech_FI_NomeiaTotalizadorNaoSujeitoIcms.....	4
Bematech_FI_ProgramaArredondamento	5
Bematech_FI_ProgramaTruncamento.....	6
Bematech_FI_NomeiaDepartamento.....	6
Bematech_FI_LinhasEntreCupons	7
Bematech_FI_EspacoEntreLinhas.....	7
Bematech_FI_ForcaImpactoAgulhas	8
Bematech_FI_ResetaImpressora.....	9
3.0 - Funções do cupom fiscal.....	10
Bematech_FI_AbreCupom	10
Bematech_FI_VendeItem	10
Bematech_FI_VendeItemDepartamento.....	12
Bematech_FI_CancelaItemAnterior	13
Bematech_FI_CancelaItemGenerico	13
Bematech_FI_IniciaFechamentoCupom	14
Bematech_FI_EfetuaFormaPagamento.....	15
Bematech_FI_EfetuaFormaPagamentoDescricaoForma.....	16
Bematech_FI_TerminaFechamentoCupom	16
Bematech_FI_FechaCupom	17
Bematech_FI_FechaCupomResumido	18
Bematech_FI_CancelaCupom	19
Bematech_FI_AumentaDescricaoItem.....	20
Bematech_FI_UsaUnidadeMedida	20
Bematech_FI_EstornoFormasPagamento	21
4.0 - Funções de relatórios fiscais.....	22
Bematech_FI_ReduçãoZ	22
Bematech_FI_LeituraX	22
Bematech_FI_LeituraXSerial	23
Bematech_FI_LeituraMemoriaFiscalData	23
Bematech_FI_LeituraMemoriaFiscalReducao	24
Bematech_FI_LeituraMemoriaFiscalSerialData	25
Bematech_FI_LeituraMemoriaFiscalSerialReducao	25
5.0 - Funções das operações não fiscais.....	27
Bematech_FI_RelatórioGerencial	27
Bematech_FI_FechaRelatórioGerencial	27

Bematech_FI_RecebimentoNaoFiscal	28
Bematech_FI_AbreComprovanteNaoFiscalVinculado	29
Bematech_FI_UsaComprovanteNaoFiscalVinculado	29
Bematech_FI_FechaComprovanteNaoFiscalVinculado.....	30
Bematech_FI_Sangria	31
Bematech_FI_Suprimento	31
6.0 – Funções de autenticação	33
Bematech_FI_Autenticacao	33
Bematech_FI_ProgramaCaracterAutenticacao.....	33
7.0 - Funções da gaveta de dinheiro	35
Bematech_FI_AcionaGaveta	35
Bematech_FI_VerificaEstadoGaveta	35
8.0 – Funções de informações da impressora	37
Bematech_FI_VerificaEstadoImpressora.....	37
Bematech_FI_RetornoAliquotas	37
Bematech_FI_VerificaTotalizadoresParciais.....	38
Bematech_FI_SubTotal	39
Bematech_FI_MonitoramentoPapel	39
Bematech_FI_DadosUltimaReducao	40
Bematech_FI_NumeroSerie.....	41
Bematech_FI_VersaoFirmware.....	41
Bematech_FI_CGC_IE	42
Bematech_FI_GrandeTotal	42
Bematech_FI_Descontos.....	43
Bematech_FI_Cancelamentos	43
Bematech_FI_NumeroCupom	44
Bematech_FI_NumeroOperacoesNaoFiscais.....	44
Bematech_FI_NumeroCuponsCancelados	45
Bematech_FI_NumeroReducoes	45
Bematech_FI_NumeroIntervencoes.....	46
Bematech_FI_NumeroSubstituicoesProprietario	46
Bematech_FI_UltimoItemVendido.....	47
Bematech_FI_ClicheProprietario	47
Bematech_FI_NumeroCaixa	48
Bematech_FI_NumeroLoja	49
Bematech_FI_SimboloMoeda	49
Bematech_FI_FlagsFiscais	50
Bematech_FI_MinutosLigada.....	51
Bematech_FI_MinutosImprimindo.....	51
Bematech_FI_VerificaModoOperacao	52
Bematech_FI_VerificaEpromConectada	52
Bematech_FI_ValorPagoUltimoCupom	53
Bematech_FI_DataHoraImpressora.....	53
Bematech_FI_ContadoresTotalizadoresNaoFiscais	54

Bematech_FI_VerificaTotalizadoresNaoFiscais	55
Bematech_FI_DataHoraReducao	56
Bematech_FI_DataMovimento.....	56
Bematech_FI_VerificaTruncamento.....	57
Bematech_FI_VerificaAliquotasIss.....	57
Bematech_FI_Acrescimos	58
Bematech_FI_ContadorBilhetePassagem	58
Bematech_FI_VerificaFormasPagamento	59
Bematech_FI_VerificaRecebimentoNaoFiscal	60
Bematech_FI_VerificaDepartamentos	61
Bematech_FI_VerificaTipoImpressora.....	62
Bematech_FI_VerificaIndiceAliquotasIss	63
Bematech_FI_ValorFormaPagamento.....	64
Bematech_FI_ValorTotalizadorNaoFiscal	64
9.0 – Funções para impressão de cheques	66
Bematech_FI_ProgramaMoedaSingular	66
Bematech_FI_ProgramaMoedaPlural	66
Bematech_FI_VerificaStatusCheque	67
Bematech_FI_CancelaImpressaoCheque	68
Bematech_FI_ImprimeCheque	68
Bematech_FI_ImprimeCopiaCheque	69
Bematech_FI_IncluiCidadeFavorecido	70
10.0 – Funções para a impressora restaurante.....	71
Bematech_FIR_RegistraVenda.....	71
Bematech_FIR_CancelaVenda.....	72
Bematech_FIR_AbreConferenciaMesa.....	73
Bematech_FIR_FechaConferenciaMesa	73
Bematech_FIR_ConferenciaMesa	74
Bematech_FIR_TransferenciaMesa.....	75
Bematech_FIR_Transferencialtem	76
Bematech_FIR_AbreCupomRestaurante.....	77
Bematech_FIR_FechaCupomRestaurante	77
Bematech_FIR_FechaCupomResumidoRestaurante	79
Bematech_FIR_ContaDividida	79
Bematech_FIR_FechaCupomContaDividida	81
Bematech_FIR_RelatorioMesasAbertas	83
Bematech_FIR_RelatorioMesasAbertasSerial	83
Bematech_FIR_ImprimeCardapio	84
Bematech_FIR_CardapioPelaSerial	85
Bematech_FIR_RegistroVendaSerial	85
Bematech_FIR_VerificaMemoriaLivre.....	86
11.0 – Outras funções da dll	87
Bematech_FI_AbrePortaSerial.....	87
Bematech_FI_FechaPortaSerial	87

Bematech_FI_VerificaImpressoraLigada	88
Bematech_FI_RetornoImpressora	88
Bematech_FI_AbreBilhetePassagem	89
Bematech_FI_MapaResumo.....	90
Bematech_FI_AberturaDoDia	92
Bematech_FI_FechamentoDoDia	92
Bematech_FI_RelatorioTipo60Analitico	93
Bematech_FI_RelatorioTipo60Mestre.....	94
Bematech_FI_ImprimeDepartamentos	95
Bematech_FI_ImprimeConfiguracoesImpressora.....	95
12.0 – Funções para transferência eletrônica de fundos (TEF)	97
Bematech_FITEF_VendaCartao.....	97
Bematech_FITEF_Status.....	98
Bematech_FITEF_ConfirmaVenda.....	98
Bematech_FITEF_NaoConfirmaVendaImpressao	99
Bematech_FITEF_CancelaVendaCartao.....	100
Bematech_FITEF_ADM	101
Bematech_FITEF_ImprimeTEF	102
Bematech_FITEF_ImprimeRelatorio	103
Bematech_FITEF_VendaCompleta	103
Bematech_FITEF_ConfiguraDiretorioTef	104
Bematech_FITEF_VendaCheque	105
Bematech_FITEF_ApagaResiduos.....	106
14.0 – Principais diferenças entre a dll mp20fi32 e a dll bemafi32.....	107
14.0 – Fluxogramas TEF	109

1.0 – Arquivo de configuração bemafi32.ini

A nova dll BemaFI32 possui um arquivo de configuração chamado BemaFi32.ini onde algumas opções podem ser configuradas. Esse arquivo de configuração possui seções com alguns parâmetros de configuração, a saber:

Seção Sistema: Essa seção possui as seguintes opções de configuração:

Porta: Indica a porta serial de comunicação entre a impressora e o micro. Se o valor for igual a "Default", a DLL buscará a porta onde estiver a impressora, automaticamente.

Status: Se essa opção estiver setada com o valor 1 as funções da dll irão gravar o status da impressora (ACK, ST1, ST2) no arquivo chamado "STATUS.TXT". O valor default dessa opção é 0 com o status da impressora sendo retornado através da função Bematech_FI_RetornoImpressora.

Retorno: Se essa opção estiver setada com o valor 1 as funções de informações* da dll irão gravar o retorno no arquivo chamado "RETORNO.TXT". O valor default dessa opção é 0, com as informações sendo retornadas através das variáveis passadas como parâmetro nas funções.

Log: Se essa opção estiver setada com o valor 1 a dll irá criar um arquivo de log chamado Bemafi32.log com todas as funções executadas, seus parâmetros e valores de retorno.

Path: Configura o diretório onde será criado o arquivo STATUS.TXT, RETORNO.TXT e Bemafi.log. O path default é o diretório raiz na unidade C:\.

ControlePorta: Se estiver setada com o valor 0 a abertura e o fechamento da porta serão controlados pelo programador. Se estiver setada com o valor 1, serão controlados pela dll. A dll irá abrir e fechar a porta automaticamente em cada função. É recomendável que o controle da porta seja realizado pelo programador e que a mesma seja aberta no início da execução do programa e fechada somente no término da execução do mesmo.

StatusFuncao: Esta chave configura a dll para retornar o valor -27 nas funções quando o status da impressora (ACK, ST1, ST2) for diferente de 6,0,0. Com essa configuração a função Bematech_FI_RetornoImpressora deverá ser executada somente quando o retorno for -27. O valor um "1" habilita e o valor zero "0" desabilita essa configuração. O valor default é zero "0".

* Funções de informações são todas as funções que retornam informações da impressora, por exemplo: número do cupom, retorno de alíquotas etc.

Exemplo da seção Sistema:

[Sistema]
Porta=COM2
Path=C:\

Status=0
Retorno=0
Log=0
ControlePorta=0

Seção Opcional: Essa seção grava as informações para Cidade e Favorecido. Essas opções são configuradas pela função Bematech_FI_IncluiCidadeFavorecido.

Seção Formato: Essa seção contém os bancos e suas coordenadas para impressão de cheques na impressora MP40.

Seção Copia: Contém os dados do último cheque emitido (impressora MP40).

Seção RelatorioTipo60: Contém informações usadas na emissão do relatório “tipo 60”.

2.0 - Funções de inicialização

Bematech_FI_AlteraSimboloMoeda

Descricao:

Altera o símbolo da moeda programada na impressora

Protótipo:

```
int Bematech_FI_AlteraSimboloMoeda ( char * Moeda )
```

Parâmetros:

char *	Moeda	string contendo o símbolo da moeda. O \$ é inserido automaticamente.
--------	-------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.
- Será executada somente após uma redução Z.

Bematech_FI_ProgramaAliquota

Descricao:

Programa alíquota tributária na impressora fiscal.

Protótipo:

```
int Bematech_FI_ProgramaAliquota ( char * Aliquota, int Vinculo )
```

Parâmetros:

char *	Aliquota	String com o valor da alíquota a ser programada.
int	Vinculo	Inteiro com o valor "0" para vincular a alíquota ao ICMS e "1" para vincular ao ISS.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

Bematech_FI_ProgramaHorarioVerao

Descricao:

Programa e desprograma o horário de verão. Se a impressora já estiver no horário de verão o mesmo será desprogramado atrasando o relógio em uma (1) hora, caso contrário será adiantado uma (1) hora.

Protótipo:

```
int Bematech_FI_ProgramaHorarioVerao()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação

Comentários:

- A programação do horário de verão será realizada somente após uma redução Z. Para desprogramar, somente 1 hora após a redução Z e não pode ter havido movimento na impressora nesse período.

Bematech_FI_NomeiaTotalizadorNaoSujeitoIcms

Descricao:

Programa totalizador não sujeito ao ICMS.

Protótipo:

```
int Bematech_FI_NomeiaTotalizadorNaoSujeitoIcms(int Indice, char * Totalizador)
```

Parâmetros:

int	Indice	Inteiro com a posição em que o totalizador será programado.
char *	Totalizador	String até 19 caracteres com o nome do totalizador

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado
- Não pode haver outro totalizador cadastrado com o mesmo índice
- Será executada somente após uma redução z.

Bematech_FI_ProgramaArredondamento

Descricao:

Programa o modo arredondamento na impressora. Este arredondamento se refere à venda de item com quantidade fracionária.

Protótipo:

```
int Bematech_FI_ProgramaArredondamento()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta

- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.
- Será executada somente após uma redução Z.

Bematech_FI_ProgramaTruncamento

Descricao:

Programa o modo truncamento na impressora.

Protótipo:

int Bematech_FI_ProgramaTruncamento ()

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado
- Será executada somente após uma redução Z.

Bematech_FI_NomeiaDepartamento

Descricao:

Programa departamento na impressora.

Protótipo:

int Bematech_FI_NomeiaDepartamento (int Indice, char * Departamento)

Parâmetros:

int	Índice	Inteiro com a posição em que o Departamento será cadastrado.
char *	Departamento	String com até 10 caracteres com o nome do departamento.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado
- Será executada somente após uma redução Z.

Bematech_FI_LinhasEntreCupons**Descricao:**

Programa o espaçamento de linhas entre os cupons.

Protótipo:

int Bematech_FI_LinhasEntreCupons (int * Linhas)

Parâmetros:

int *	Linhas	Inteiro entre 0 e 255 indicando o número de linhas. O valor default da impressora é 8 linhas.
-------	--------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_EspacoEntreLinhas**Descricao:**

Programa o espaçamento entre as linhas impressas no cupom.

Protótipo:

int Bematech_FI_EspacoEntreLinhas (int Dots)

Parâmetros:

int	Dots	Inteiro entre 0 e 255 indicando o espaço (dots) entre as linhas. O valor default da impressora é 0.
-----	------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_ForcaImpactoAguilhas

Descricao:

Permite tornar a impressão mais forte nos equipamentos baseados na MP-20 FI II

Protótipo:

int Bematech_FI_ForcaImpactoAguilhas (short int ForcaImpacto)

Parâmetros:

short int	ForcaImpacto	Valor da força de impacto das agulhas que pode ser: 1 – Impacto fraco (default) 2 – Impacto médio 3 – Impacto forte
-----------	--------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- A seleção de uma força de impacto diferente da default implica na diminuição da vida útil do cabeçote de impressor.

Bematech_FI_ResetaImpressora

Descricao:

Reseta a impressora em caso de erro.

Protótipo:

```
int Bematech_FI_ResetaImpressora()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Será executada somente se a impressora estiver em erro.

3.0 - Funções do cupom fiscal

Bematech_FI_AbreCupom

Descricao:

Abre o cupom fiscal na impressora.

Protótipo:

```
int Bematech_FI_AbreCupom ( char * CGC)
```

Parâmetros:

char *	CGC	string até 29 caracteres com o CGC ou CPF do cliente.
--------	-----	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado

Bematech_FI_VendItem

Descricao:

Vende item após a abertura do cupom fiscal. Essa função permite também a venda de itens com 3 casas no valor unitário.

Protótipo:

```
int Bematech_FI_VendItem (char *Codigo, char *Descricao, char *Aliquota,
                           char *TipoQuantidade, char *Quantidade,
                           int CasasDecimais, char *ValorUnitario,
                           char *TipoDesconto, char *ValorDesconto )
```

Parâmetros:

char *	Codigo	String até 13 caracteres com o código
--------	--------	---------------------------------------

char *	Descricao	do produto. String até 29 caracteres com a descrição do produto.
char *	Aliquota	String com o valor ou o índice da alíquota tributária. Se for o valor deve ser informado com o tamanho de 4 caracteres ou 5 com a vírgula. Se for o índice da alíquota deve ser 2 caracteres. Ex. (18,00 para o valor ou 05 para o índice)
char *	TipoQuantidade	Um caracter indicando o tipo de quantidade. I - Inteira e F - Fracionária
char *	Quantidade	String com até 4 dígitos para quantidade inteira e 7 dígitos para quantidade fracionária. Na quantidade fracionária são 3 casas decimais.
int	CasasDecimais	Inteiro indicando o número de casas decimais para o valor unitário (2 ou 3).
char *	ValorUnitario	String até 8 dígitos para valor unitário.
char *	TipoDesconto	Um caracter indicando a forma do desconto. '\$' desconto por valor e '%' desconto percentual.
char *	ValorDesconto	String com até 8 dígitos para desconto por valor (2 casas decimais) e 4 dígitos para desconto percentual.

Valores de Retorno:

- 0 – Erro de comunicação
- 1 – Ok
- 2 – Parâmetro inválido
- 3 – Alíquota não programada
- 4 – Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 – Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Imagine que há duas alíquotas com o valor 12% cadastradas na impressora. A primeira cadastrada na posição 01 como ICMS e a outra na posição 05 como ISS. Se você informar o valor 1200 ou 12,00 no parâmetro “alíquota” a função irá imprimir o item usando a alíquota 01 de ICMS. A função lê as alíquotas da impressora e usa o índice da primeira ocorrência. Para usar a alíquota de ISS você deverá passar o índice 05 e não o valor 1200 ou 12,00. Se você não tiver duas alíquotas com o mesmo valor cadastradas na impressora use sempre o valor no parâmetro alíquota.
- O cupom fiscal deve estar aberto.

Bematech_FI_VendeItemDepartamento

Descrição:

Essa função permite a venda de itens com entrada de departamento, desconto e unidade de medida.

Protótipo:

```
int Bematech_FI_VendeItemDepartamento (char * Codigo, char * Descricao,
                                         char * Aliquota, char * ValorUnitario,
                                         char * Quantidade, char * Acrescimo,
                                         char * Desconto, char * Indice
                                         Departamento, char * UnidadeMedida)
```

Parâmetros:

char *	Codigo	String até 49 caracteres com o código do produto.
char *	Descricao	String até 201 caracteres com a descrição do produto.
char *	Aliquota	String com o valor ou o índice da alíquota tributária. Se for o valor deve ser informado com o tamanho de 4 caracteres ou 5 com a vírgula. Se for o índice da alíquota deve ser 2 caracteres. Ex. (18,00 para o valor ou 05 para o índice)
char *	ValorUnitario	String com até 9 dígitos para o valor (tres casas decimais).
char *	Quantidade	String com até 7 dígitos para a quantidade. Na venda com departamento a quantidade é fracionária e são 3 casas decimais.
char *	Acrescimo	String com o acréscimo por valor com até 10 dígitos (2 casas decimais).
char *	Desconto	String com o desconto por valor com até 10 dígitos (2 casas decimais).
char *	IndiceDepartamento	Índice do departamento com 2 dígitos.
char *	UnidadeMedida	String com no máximo 2 caracteres para a unidade de medida. Caso não seja passado nenhum caracter a unidade não é impressa.

Valores de Retorno:

0 - Erro de comunicação

- 1 - Ok
- 2 - Parâmetro inválido
- 3 - Alíquota não programada
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Obedece a mesma situação descrita no item “comentários” da função acima Bematech_FI_VendeItem().

Bematech_FI_CancelarItemAnterior

Descricao:

Cancela o último item vendido.

Protótipo:

int Bematech_FI_CancelarItemAnterior ()

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar aberto
- Ao menos um item deve ter sido vendido
- Não deve ter sido cancelado nenhum item imediatamente anterior.

Bematech_FI_CancelarItemGenerico

Descricao:

Cancela qualquer um dentre os cem (100) últimos itens vendidos.

Protótipo:

int Bematech_FI_CancelarItemGenerico (char * NumeroItem)

Parâmetros:

char *	NumeroItem	String com o número do item a ser cancelado com no máximo 3 dígitos.
--------	------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar aberto
- Ao menos um item deve ter sido vendido
- O item não pode ter sido cancelado anteriormente e nem fora da faixa dos últimos itens vendidos.

Bematech_FI_IniciaFechamentoCupom

Descricao:

Inicia o fechamento do cupom com o uso das formas de pagamento.

Protótipo:

int Bematech_FI_IniciaFechamentoCupom (char * acrescimoDesconto,
char * TipoAcrescimoDesconto,
char * ValorAcrescimoDesconto)

Parâmetros:

char *	AcrescimoDesconto	Indica se haverá acréscimo ou desconto no cupom. 'A' para acréscimo e 'D' para desconto.
char *	TipoAcrescimoDesconto	Indica se o acréscimo ou desconto é por valor ou por percentual. '\$' para desconto por valor e '%' para percentual.
char *	ValorAcrescimoDesconto	String com no máximo 14 dígitos para

acréscimo ou desconto por valor e 4 dígitos para acréscimo ou desconto por percentual.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom deve estar aberto
- Pelo menos um item deve ter sido vendido
- O valor do acréscimo ou descontado deve ser menor que o subtotal do cupom.

Bematech_FI_EfetuaFormaPagamento**Descricao:**

Imprime a(s) forma(s) de pagamento e o(s) valor(es) pago(s) nessa forma.

Protótipo:

```
int Bematech_FI_EfetuaFormaPagamento ( char * FormaPagamento,  
                                         char * ValorFormaPagamento )
```

Parâmetros:

char *	FormaPagamento	String com a forma de pagamento com no máximo 16 caracteres.
char *	ValorFormaPagamento	Valor da forma de pagamento com até 14 dígitos.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 24 - Forma de pagto não programada
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O fechamento do cupom com formas de pagamento deve ter sido iniciado.

Bematech_FI_EfetuaFormaPagamentoDescricaoForma

Descricao:

Imprime a(s) forma(s) de pagamento e o(s) valor(es) pago(s). Permite a impressão de comentários na(s) forma(s) de pagamento.

Protótipo:

```
int Bematech_FI_EfetuaFormaPagamentoDescricaoForma (
                                                    char * FormaPagamento,
                                                    char * ValorFormaPagamento,
                                                    char * DescricaoFormaPagto )
```

Parâmetros:

char *	FormaPagamento	String com a forma de pagamento com no máximo 16 caracteres.
char *	ValorFormaPagamento	Valor da forma de pagamento com até 14 dígitos.
char *	DescricaoFormaPagto	String com a descrição da forma de pagamento com no máximo 80 caracteres.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 24 - Forma de pagto não programada
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- A descrição será impressa uma linha após a forma de pagamento.
- O fechamento do cupom com formas de pagamento deve ter sido iniciado.

Bematech_FI_TerminaFechamentoCupom

Descricao:

Termina o fechamento do cupom com mensagem promocional.

Protótipo:

```
int Bematech_FI_TerminaFechamentoCupom ( char * Mensagem )
```

Parâmetros:

char *	Mensagem	Mensagem promocional com até 490 caracteres.
--------	----------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- A forma de pagamento deve ter sido efetuada.

Bematech_FI_FechaCupom

Descricao:

Fecha o cupom fiscal com a impressão da mensagem promocional.

Protótipo:

```
int Bematech_FI_FechaCupom ( char * FormaPagamento,
                             char * AcrescimoDesconto,
                             char * TipoAcrescimoDesconto,
                             char * ValorAcrescimoDesconto,
                             char * ValorPago, char * Mensagem )
```

Parâmetros:

char *	FormaPagamento	String com o nome da forma de pagamento com no máximo 16 caracteres.
char *	AcrescimoDesconto	Indica se haverá acréscimo ou desconto no cupom. 'A' para acréscimo e 'D' para desconto.
char *	TipoAcrescimoDesconto	Indica se o acréscimo ou desconto é

char *	ValorAcrescimoDesconto	por valor ou por percentual. '\$' para desconto por valor e '%' para percentual. String com no máximo 14 dígitos para acréscimo ou desconto por valor e 4 dígitos para acréscimo ou desconto por percentual.
char *	ValorPago	Valor pago com no máximo 14 dígitos.
char *	Mensagem	Mensagem promocional com até 490 caracteres.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom deve estar aberto
- Pelo menos 1 (um) item deve ter sido vendido e não pode ter sido cancelado
- A utilização dessa função elimina a obrigatoriedade de uso das funções Bematech_FI_IniciaFechamentoCupom, Bematech_FI_EfetuaFormaPagamento e Bematech_FI_TerminaFechamentoCupom que estão implementadas internamente na função.

Bematech_FI_FechaCupomResumido

Descricao:

Permite fechar o cupom de forma resumida, ou seja, sem acréscimo ou desconto no cupom e com apenas uma forma de pagamento. Essa função lê o subtotal do cupom para fecha-lo.

Protótipo:

```
int Bematech_FI_FechaCupomResumido ( char * FormaPagamento,
                                     char * Mensagem )
```

Parâmetros:

char *	FormaPagamento	Forma de pagamento com no máximo 16 caracteres.
char *	Mensagem	Mensagem promocional com até 490 caracteres.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom deve estar aberto
- Pelo menos 1 (um) item deve ter sido vendido e não pode ter sido cancelado
- A utilização dessa função elimina a obrigatoriedade de uso das funções Bematech_FI_IniciaFechamentoCupom, Bematech_FI_EfetuaFormaPagamento e Bematech_FI_TerminaFechamentoCupom que estão implementadas internamente na função.

Bematech_FI_CancelaCupom

Descricao:

Cancela o último cupom emitido.

Protótipo:

```
int Bematech_FI_CancelaCupom()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- No caso de cancelamento com o cupom ainda aberto, pelo menos um item deve ter sido vendido.
- Não poderá ter ocorrido redução Z e nem leitura X após a emissão do cupom.

Bematech_FI_AumentaDescricaoItem

Descricao:

Esta função permite aumentar a descrição do item até 200 caracteres.

Protótipo:

```
int Bematech_FI_AumentaDescicaoItem ( char * Descricao )
```

Parâmetros:

char *	Descricao	Descrição do item com até 200 caracteres.
--------	-----------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Esta função tem validade somente para a impressão de um item, voltando ao default que é a impressão com 29 caracteres na descricao do item.

Bematech_FI_UsaUnidadeMedida

Descricao:

Imprime a unidade de medida após a quantidade do produto na venda de item.

Protótipo:

```
int Bematech_FI_UsaUnidadeMedida ( char * UnidadeMedida )
```

Parâmetros:

char *	UnidadeMedida	Unidade de medida até 2 caracteres.
--------	---------------	-------------------------------------

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok

- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Esta função tem validade somente para a impressão de um item, voltando ao default que é a impressão de dois (2) espaços após a quantidade do item vendido.

Bematech_FI_EstornoFormasPagamento**Descricao:**

Permite estornar valores de uma forma de pagamento e inserir em outra.

Protótipo:

```
int Bematech_FI_EstornoFormasPagamento ( char * FormaOrigem,  
                                           char * FormaDestino, char * Valor )
```

Parâmetros:

char *	FormaOrigem	Forma de pagamento de onde o valor será estornado com até 16 caracteres.
char *	FormaDestino	Forma de pagamento onde o valor será inserido.
char *	Valor	Valor a ser estornado com até 14 dígitos. Não pode ser maior que o total da forma de pagamento de origem.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

4.0 - Funções de relatórios fiscais

Bematech_FI_ReduçãoZ

Descricao:

Emite a redução Z na impressora. Permite ajustar o relógio interno da impressora em até 5 minutos.

Protótipo:

```
int Bematech_FI_ReducacaoZ ( char * Data, char * Hora )
```

Parâmetros:

char *	Data	Data atual da impressora no formato ddmmaa ou dd/mm/aa
char *	Hora	dd/mm/aaaa ou dd/mm/aa. Hora a ser alterada no formato hhmmss ou hh:mm:ss.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Somente será aceito um ajuste de +/- 5 minutos. Se os valores estiverem fora dessa faixa serão limitados a 5 minutos.
- O cupom fiscal deve estar fechado.

Bematech_FI_LeituraX

Descricao:

Emite a leitura X na impressora.

Protótipo:

```
int Bematech_FI_LeituraX()
```

Parâmetros:

Não há.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

Bematech_FI_LeituraXSerial

Descricao:

Recebe os dados da leitura X pela serial e grava em arquivo texto.

Protótipo:

```
int Bematech_FI_LeituraXSerial()
```

Parâmetros:

Não há.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

Bematech_FI_LeituraMemoriaFiscalData

Descricao:

Emite a leitura da memória fiscal da impressora por intervalo de datas.

Protótipo:

```
int Bematech_FI_LeituraMemoriaFiscalData ( char * DataInicial, char * DataFinal )
```

Parâmetros:

char *	DataInicial	Data inicial no formato ddmmaa, dd/mm/aa, ddmmaaaa ou dd/mm/aaaa
char *	DataFinal	Data final no formato ddmmaa, dd/mm/aa, ddmmaaaa ou dd/mm/aaaa

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

Bematech_FI_LeituraMemoriaFiscalReducao

Descricao:

Emite a leitura da memória fiscal da impressora por intervalo de reduções.

Protótipo:

```
int Bematech_FI_LeituraMemoriaFiscalReducao ( char * ReducaoInicial,
                                              char * ReducaoFinal )
```

Parâmetros:

char *	ReducaoInicial	Número da reducao inicial com até 4 dígitos
char *	ReducaoFinal	Número da reducao final com até 4 dígitos.

Valores de Retorno:

- 0 - Erro de comunicação

- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

Bematech_FI_LeituraMemoriaFiscalSerialData

Descricao:

Recebe os dados da memória fiscal por intervalo de datas pela serial e grava em arquivo texto.

Protótipo:

```
int Bematech_FI_LeituraMemoriaFiscalSerialData ( char * DataInicial,
                                                char * DataFinal )
```

Parâmetros:

char *	DataInicial	Data inicial no formato ddmmaa, dd/mm/aa, ddmmaaaa ou dd/mm/aaaa
char *	DataFinal	Data final no formato ddmmaa, dd/mm/aa, ddmmaaaa ou dd/mm/aaaa

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

Bematech_FI_LeituraMemoriaFiscalSerialReducao

Descricao:

Recebe os dados da leitura da memória fiscal por intervalo de reduções pela serial e grava em arquivo texto.

Protótipo:

```
int Bematech_FI_LeituraMemoriaFiscalSerialReducao ( char * ReducaoInicial,  
                                                    char * ReducaoFinal )
```

Parâmetros:

char *	ReducaoInicial	Número da reducao inicial com até 4 dígitos
char *	ReducaoFinal	Número da reducao final com até 4 dígitos.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

5.0 - Funções das operações não fiscais

Bematech_FI_RelatórioGerencial

Descricao:

Imprime o relatório gerencial.

Protótipo:

```
int Bematech_FI_RelatorioGerencial ( char * Texto )
```

Parâmetros:

char *	Texto	Texto a ser impresso no relatório com até 618 caracteres.
--------	-------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.
- Se qualquer função diferente da Bematech_FI_RelatorioGerencial for chamada com o relatorio aberto, efetuará seu fechamento automaticamente.

Bematech_FI_FechaRelatórioGerencial

Descricao:

Encerra o relatório gerencial.

Protótipo:

```
int Bematech_FI_FechaRelatorioGerencial()
```

Parâmetros:

Não há.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O relatório gerencial deve estar aberto.

Bematech_FI_RecebimentoNaoFiscal

Descricao:

Imprime o comprovante não fiscal não vinculado.

Protótipo:

```
int Bematech_FI_RecebimentoNaoFiscal ( char * IndiceTotalizador,
                                         char * ValorRecebimento,
                                         char * FormaPagamento )
```

Parâmetros:

char *	IndiceTotalizador	Indice do totalizador para recebimento parcial com até 2 dígitos.
char *	ValorRecebimento	Valor do recebimento com até 14 dígitos (duas casas decimais)
char *	FormaPagamento	Forma de pagamento com até 16 caracteres.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado
- O totalizador deve estar cadastrado.

Bematech_FI_AbreComprovanteNaoFiscalVinculado

Descrição:

Abre o comprovante não fiscal vinculado.

Protótipo:

```
int Bematech_FI_AbreComprovanteNaoFiscalVinculado(char * FormaPagamento,
                                                    char * Valor,
                                                    char * NumeroCupom )
```

Parâmetros:

char *	FormaPagamento	Forma de pagamento com até 16 caracteres
char *	Valor	Valor pago na forma de pagamento com até 14 dígitos (2 casas decimais)
char *	NumeroCupom	Número do cupom a que se refere o comprovante com até 6 dígitos.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- A forma de pagamento utilizada não pode ser dinheiro
- A forma de pagamento deve ter sido utilizada no cupom
- Só pode ser emitido um comprovante não fiscal por forma de pagamento
- Os parâmetros "Valor" e "Numero do Cupom" tornam-se obrigatórios se o comprovante emitido não for referente ao último cupom fiscal emitido.

Bematech_FI_UsaComprovanteNaoFiscalVinculado

Descrição:

Imprime o comprovante não fiscal vinculado.

Protótipo:

```
int Bematech_FI_UsaComprovanteNaoFiscalVinculado (char * Texto)
```

Parâmetros:

char *	Texto	Texto a ser impresso no comprovante não fiscal vinculado com até 618 caracteres.
--------	-------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Só pode ser usado durante 2 (dois) minutos após a abertura do comprovante não fiscal vinculado. Se esse tempo for ultrapassado o comprovante é fechado automaticamente.

Bematech_FI_FechaComprovanteNaoFiscalVinculado

Descricao:

Encerrar o comprovante não fiscal vinculado.

Protótipo:

```
int Bematech_FI_FechaComprovanteNaoFiscalVinculado()
```

Parâmetros:

Não há.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O comprovante não fiscal vinculado deve ter sido aberto.

Bematech_FI_Sangria

Descricao:

Faz uma sangria na impressora (retirada de dinheiro).

Protótipo:

int Bematech_FI_Sangria (char * Valor)

Parâmetros:

char *	Valor	Valor da sangria com até 14 dígitos (2 casas decimais).
--------	-------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom fiscal deve estar fechado.

Bematech_FI_Suprimento

Descricao:

Faz um suprimento na impressora.

Protótipo:

int Bematech_FI_Suprimento (char * Valor, char * FormaPagamento)

Parâmetros:

char *	Valor	Valor do suprimento com até 14 dígitos (2 casas decimais).
char *	FormaPagamento	Forma de pagamento com até 16 caracteres. Se não for informada, o suprimento será feito em dinheiro.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

6.0 – Funções de autenticação

Bematech_FI_Autenticacao

Descricao:

Permite a autenticação de documentos.

Protótipo:

```
int Bematech_FI_Autenticacao()
```

Parâmetros:

Não há.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Deverá ser executada imediatamente após um recebimento não fiscal ou o término de um cupom fiscal.
- Poderá ser repetido até 5 vezes para cada recebimento, após isso o comando será ignorado
- A impressora irá aguardar 5 segundos para o posicionamento do documento e autenticará no documento a data, hora, número da loja, número do caixa e o número do último cupom fiscal
- Se decorrido os 5 segundos sem o posicionamento do documento, a impressora retornará ao seu estado normal, retornando o status de “comando não executado”.

Bematech_FI_ProgramaCaracterAutenticacao

Descricao:

Programa o caracter gráfico para autenticação.

Protótipo:

```
int Bematech_FI_ProgramaCaracterAutenticacao( char * Coordenadas )
```

Parâmetros:

char * Coordenadas String com os 18 valores para programação do carácter gráfico, separados por vírgula. Veja exemplo abaixo em "Comentários".

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Exemplo de programação do carácter gráfico:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Bit 0 (1)	■														■	■	■	■
Bit 1 (2)		■												■				
Bit 2 (4)			■										■					
Bit 3 (8)				■								■						
Bit 4 (16)					■						■							
Bit 5 (32)						■				■								
Bit 6 (64)							■		■									
Bit 7 (128)								■							■	■	■	■

Para programar o carácter acima deve-se passar a seguinte string de parâmetros: "001,002,004,008,016,032,064,128,064,032,016,008,004,002,129,129,129,129"

Uma vez programado, este carácter será mantido na memória da impressora mesmo que a mesma seja desligada.

7.0 - Funções da gaveta de dinheiro

Bematech_FI_AcionaGaveta

Descricao:

Abre a gaveta de dinheiro.

Protótipo:

```
int Bematech_FI_AcionaGaveta()
```

Parâmetros:

Não há.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_VerificaEstadoGaveta

Descricao:

Retorna se a gaveta está fechada ou aberta.

Protótipo:

```
int Bematech_FI_VerificaEstadoGaveta ( short int * Estado )
```

Parâmetros:

short int *	Estado	Variável para receber o estado da gaveta. 1 sensor em nível 1 (aberta), 0 sensor em nível 0 (fechada)
-------------	--------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1- Erro de execução da função.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação

- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

8.0 – Funções de informações da impressora

Bematech_FI_VerificaEstadoImpressora

Descricao:

Retorna o estado da impressora.

Protótipo:

```
int Bematech_FI_VerificaEstadoImpressora ( short int * ACK, short int * ST1,  
                                           short int * ST2 )
```

Parâmetros:

short int *	ACK	Variáveis para receber os 3 bytes de estado da impressora.
short int *	ST1	
short int *	ST2	

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_RetornoAliquotas

Descricao:

Retorna as alíquotas cadastradas na impressora.

Protótipo:

```
int Bematech_FI_RetornoAliquota ( char * Aliquotas )
```

Parâmetros:

char *	Aliquotas	Variável com o tamanho de 79 posições para receber as alíquotas.
--------	-----------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta

- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- As alíquotas retornadas estarão separadas por vírgula. Ex.: 1200, 1700

Bematech_FI_VerificaTotalizadoresParciais

Descricao:

Retorna os totalizadores parciais da impressora

Protótipo:

```
int Bematech_FI_VerificaTotalizadoresParciais( char * Totalizadores )
```

Parâmetros:

char *	Totalizadores	Variável string com o tamanho de 445 posições para receber os totalizadores parciais cadastrados.
--------	---------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- São retornadas as seguintes informações separadas por vírgulas:

Totalizadores parciais tributados:	224 bytes
Isenção:	14 bytes
Não insidência:	14 bytes
Substituição:	14 bytes
Totalizadores parciais não sujeitos ao ICMS:	126 bytes
Sangria:	14 bytes
Suprimento:	14 bytes
Grande Total:	18 bytes

Bematech_FI_SubTotal

Descricao:

Retorna o valor do subtotal do cupom.

Protótipo:

int Bematech_FI_SubTotal (char * SubTotal)

Parâmetros:

char *	SubTotal	Variável string com o tamanho de 14 posições para receber o subtotal do cupom.
--------	----------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_MonitoramentoPapel

Descricao:

Retorna o número de linhas impressas após o status de pouco papel.

Protótipo:

int Bematech_FI_MonitoramentoPapel (int * LinhasImpressas)

Parâmetros:

int *	LinhasImpressas	Variável inteira para receber a quantidade de linhas impressas.
-------	-----------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_DadosUltimaReducao

Descricao:

Retorna os dados da impressora no momento da última redução Z.

Protótipo:

```
int Bematech_FI_DadosUltimaReducao ( char * DadosReducao )
```

Parâmetros:

char *	DadosReducao	Variável string com o tamanho de 631 posições para receber os dados da última redução.
--------	--------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- São retornados os valores das seguintes informações separados por vírgulas:

Modo de redução Z:	2 bytes
se 00 redução por comando	
se 01 redução automática	
Grande Total:	18 bytes
Cancelamentos:	14 bytes
Descontos:	14 bytes
Tributos:	64 bytes
Totalizadores Parciais Tributados:	266 bytes
Sangria:	14 bytes
Suprimentos:	14 bytes
Totalizadores não sujeitos ao ICMS:	126 bytes
Contadores dos TP's não sujeitos ao ICMSI:	36 bytes
Contador de ordem de operação:	6 bytes
Contador de operações não sujeitas ao ICMS	6 bytes
Número de Alíquotas cadastradas	2 bytes
Data do movimento:	6 bytes
Acréscimos:	14 bytes

Acréscimo financeiro:

14 bytes

Bematech_FI_NumeroSerie

Descricao:

Retorna o número de série da impressora.

Protótipo:

```
int Bematech_FI_NumeroSerie ( char * NumeroSerie )
```

Parâmetros:

char *	NumeroSerie	Variável string com o tamanho de 15 posições para receber o número de série.
--------	-------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_VersaoFirmware

Descricao:

Retorna a versão do firmware da impressora.

Protótipo:

```
int Bematech_FI_VersaoFirmware ( char * VersaoFirmware )
```

Parâmetros:

char *	VersaoFirmware	Variável string com o tamanho de 4 posições para receber a versão do firmware.
--------	----------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok

- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_CGC_IE

Descricao:

Retorna o CGC e a Inscrição Estadual do cliente cadastrado na impressora.

Protótipo:

```
int Bematech_FI_CGC_IE ( char * CGC, char * IE )
```

Parâmetros:

char *	CGC	Variável string com 18 posições para receber o CGC
char *	IE	Variável string com 15 posições para receber a Inscrição Estadual.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_GrandeTotal

Descricao:

Retorna o valor do Grande Total da impressora.

Protótipo:

```
int Bematech_FI_GrandeTotal ( char * GrandeTotal )
```

Parâmetros:

char *	GrandeTotal	Variável string com 18 posições para receber o valor do grande total com 2 casas decimais.
--------	-------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_Descontos

Descricao:

Retorna a valor acumulado dos descontos.

Protótipo:

int Bematech_FI_Descontos (char * Descontos)

Parâmetros:

char *	Descontos	Variável string com 14 posições para receber o valor dos descontos com 2 casas decimais.
--------	-----------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_Cancelamentos

Descricao:

Retorna o valor acumulado dos itens e dos cupons cancelados

Protótipo:

int Bematech_FI_Cancelamentos (char * Cancelamentos)

Parâmetros:

char *	Cancelamentos	Variável string com 14 posições para receber o valor dos cancelamentos com 2 casas decimais.
--------	---------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroCupom**Descricao:**

Retorna o número do cupom.

Protótipo:

int Bematech_FI_NumeroCupom (char * NumeroCupom)

Parâmetros:

char *	NumeroCupom	Variável string com 6 posições para receber o número do último cupom.
--------	-------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroOperacoesNaoFiscais**Descricao:**

Retorna o número de operações não fiscais executadas na impressora.

Protótipo:

int Bematech_FI_NumeroOperacoesNaoFiscais (char * Operacoes)

Parâmetros:

char *	Operacoes	Variável string com 6 posições para receber o número de operações.
--------	-----------	--

Valores de Retorno:

0 - Erro de comunicação
1 - Ok
-4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
-5 - Erro ao abrir a porta de comunicação
-8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroCuponsCancelados

Descricao:

Retorna o número de cupons cancelados.

Protótipo:

int Bematech_FI_NumeroCuponsCancelados (char * CuponsCancelados)

Parâmetros:

char *	CuponsCancelados	Variável string com 4 posições para receber o número de cupons cancelados.
--------	------------------	--

Valores de Retorno:

0 - Erro de comunicação
1 - Ok
-4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
-5 - Erro ao abrir a porta de comunicação
-8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroReducoes

Descricao:

Retorna o número de reduções Z realizadas na impressora.

Protótipo:

int Bematech_FI_NumeroReducoes (char * Reducoes)

Parâmetros:

char *	Reducoes	Variável string com 4 posições para receber o número de reduções z.
--------	----------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroIntervencoes

Descricao:

Retorna o número de intervenções técnicas realizadas na impressora.

Protótipo:

int Bematech_FI_NumeroIntervencoes (char * Intervencoes)

Parâmetros:

char *	Intervencoes	Variável string com 4 posições para receber o número de intervenções.
--------	--------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroSubstituicoesProprietario

Descricao:

Retorna o número de substituições de proprietário.

Protótipo:

int Bematech_FI_NumeroSubstituicoesProprietario (char * Substituicoes)

Parâmetros:

char *	Substituicoes	Variável string com 4 posições para receber o número de substituições.
--------	---------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_UltimoltemVendido

Descricao:

Retorna o número do último item vendido.

Protótipo:

int Bematech_FI_UltimoltemVendido (char * Ultimoltem)

Parâmetros:

char *	Ultimoltem	Variável string com 4 posições para receber o número do último item vendido.
--------	------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_ClicheProprietario

Descricao:

Retorna o clichê do proprietário cadastrado na impressora.

Protótipo:

int Bematech_FI_ClicheProprietario(char * ClicheProprietario)

Parâmetros:

char *	ClicheProprietario	Variável string com 186 posições para receber clichê cadastrado.
--------	--------------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroCaixa

Descricao:

Retorna o número do caixa cadastrado na impressora.

Protótipo:

int Bematech_FI_NumeroCaixa (char * NumeroCaixa)

Parâmetros:

char *	NumeroCaixa	Variável string com 4 posições para receber o número do caixa cadastrado na impressora.
--------	-------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_NumeroLoja

Descricao:

Retorna o número da loja cadastrado na impressora.

Protótipo:

int Bematech_FI_NumeroLoja (char * NumeroLoja)

Parâmetros:

char *	NumeroLoja	Variável string com 4 posições para receber o número da loja cadastrado na impressora.
--------	------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_SimboloMoeda

Descricao:

Retorna o símbolo da moeda cadastrado na impressora.

Protótipo:

int Bematech_FI_SimboloMoeda (char * SimboloMoeda)

Parâmetros:

char *	SimboloMoeda	Variável string com 2 posições para receber o símbolo da moeda.
--------	--------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_FlagsFiscais**Descricao:**

Retorna um número referente ao flag fiscal da impressora. Veja discriminação abaixo.

Protótipo:

int Bematech_FI_FlagsFiscais (short int * FlagFiscal)

Parâmetros:

short int *	FlagFiscal	Variável inteira para receber um número representando o flag fiscal da impressora. Veja discriminação abaixo.
-------------	------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Discriminação dos valores para o flag fiscal:

Descrição	Valor
Cupom fiscal aberto	1
Fechamento de formas de pagamento iniciado	2
Horário de verão selecionado	4
Já houve redução Z no dia	8
Permite cancelar cupom fiscal	32
Memória fiscal sem espaço	128

Os valores podem vir somados o que indica mais de um estado. Ex.:

FlagFiscal = 37 - Cupom fiscal aberto, Horário de verão selecionado, Permite cancelar o cupom fiscal.

Bematech_FI_MinutosLigada

Descricao:

Retorna o tempo em minutos que a impressora está ligada.

Protótipo:

int Bematech_FI_MinutosLigada (char * MinutosLigada)

Parâmetros:

char *	MinutosLigada	Variável string com 4 posições para receber os minutos em que a impressora está ligada.
--------	---------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_MinutosImprimindo

Descricao:

Retorna o tempo em minutos que a impressora está ou esteve imprimindo.

Protótipo:

int Bematech_FI_MinutosImprimindo (char * MinutosImprimindo)

Parâmetros:

char *	MinutosImprimindo	Variável string com 4 posições para receber os minutos em impressão.
--------	-------------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_VerificaModoOperacao

Descricao:

Verifica se a impressora está em modo normal ou em intervenção técnica.

Protótipo:

```
int Bematech_FI_VerificaModoOperacao ( char * ModoOperacao )
```

Parâmetros:

char *	ModoOperacao	Variável string com 1 posição para receber o modo de operação da impressora. 1 - Modo normal 0 - Intervenção técnica.
--------	--------------	---

Valores de Retorno:

0 - Erro de comunicação
1 - Ok
-1 - Erro de execução.
-4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
-5 - Erro ao abrir a porta de comunicação
-8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_VerificaEpromConectada

Descricao:

Verifica se a Eprom está conectada.

Protótipo:

```
int Bematech_FI_VerificaEpromConectada ( char * FlagEprom )
```

Parâmetros:

char *	FlagEprom	Variável string com 1 posição para receber o flag de Eprom conectada. 1 - Eprom conectada 0 - Eprom desconectada.
--------	-----------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_ValorPagoUltimoCupom

Descricao:

Retorna o valor pago no último cupom.

Protótipo:

int Bematech_FI_ValorPagoUltimoCupom (char * Valor)

Parâmetros:

char *	Valor	Variável string com 14 posições para receber o valor pago no último cupom.
--------	-------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Valor pago no último cupom é diferente do valor total do cupom.
Ex.: Imaginemos que o valor do cupom seja R\$15,00 e foi pago pelo cliente R\$ 20,00. Teve um troco de R\$ 5,00. O valor retornado por essa função será R\$ 20,00 e não R\$ 15,00

Bematech_FI_DataHoraImpressora

Descricao:

Retorna a data e a hora atual da impressora.

Protótipo:

int Bematech_FI_DataHoraImpressora (char * Data, char * Hora)

Parâmetros:

char *	Data	Variável string com 6 posições para receber a data atual da impressora no formato ddmmaa.
char *	Hora	Variável string com 6 posições para receber a hora atual da impressora no formato hhmmss.

Valores de Retorno:

0 - Erro de comunicação
1 - Ok
-1 - Erro de execução.
-4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
-5 - Erro ao abrir a porta de comunicação
-8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_ContadoresTotalizadoresNaoFiscais

Descricao:

Retorna o número de vezes em que os totalizadores não sujeitos ao ICMS foram usados.

Protótipo:

int Bematech_FI_ContadoresTotalizadoresNaoFiscais (char * Contadores)

Parâmetros:

char *	Contadores	Variável string com 44 posições para receber os contadores dos totalizadores.
--------	------------	---

Valores de Retorno:

0 - Erro de comunicação
1 - Ok
-1 - Erro de execução.
-4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
-5 - Erro ao abrir a porta de comunicação
-8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.

-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- A impressora permite a programação de até 50 totalizadores não fiscais, porém essa função retorna somente os contadores dos 9 primeiros totalizadores cadastrados.
- O conteúdo da variável retornada será 36 dígitos separados de 4 em 4 por vírgula que representam os 9 primeiros totalizadores. Ex.:
"0001,0003,0001,0005,0004,0002,0003,0004,0007"
O primeiro valor corresponde ao número de vezes que o totalizador 01 foi usado, o segundo corresponde ao totalizador 02 e assim sucessivamente.

Bematech_FI_VerificaTotalizadoresNaoFiscais

Descricao:

Retorna a descrição dos totalizadores não fiscais programados na impressora.

Protótipo:

```
int Bematech_FI_VerificaTotalizadoresNaoFiscais ( char * Totalizadores )
```

Parâmetros:

char *	Totalizadores	Variável string com 179 posições para receber a descrição dos totalizadores não fiscais programados.
--------	---------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Os totalizadores virão separados por vírgula.
- A impressora permite a programação de até 50 totalizadores não fiscais, porém essa função retorna somente os 9 primeiros cadastrados.

Bematech_FI_DataHoraReducao

Descricao:

Retorna a data da última Redução Z.

Protótipo:

int Bematech_FI_DataHoraReducao (char * DataReducao, char * HoraReducao)

Parâmetros:

char *	DataReducao	Variável string com 6 posições para receber a data da última redução no formato ddmmaa.
char *	HoraReducao	Variável string com 6 posições para receber a hora da última redução no formato hhmmss.

Valores de Retorno:

0 - Erro de comunicação
1 - Ok
-1 - Erro de execução.
-4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
-5 - Erro ao abrir a porta de comunicação
-8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_DataMovimento

Descricao:

Retorna a data do último movimento.

Protótipo:

int Bematech_FI_DataMovimento (char * DataMovimento)

Parâmetros:

char *	DataMovimento	Variável string com 6 posições para receber a data do movimento no formato ddmmaa.
--------	---------------	--

Valores de Retorno:

0 - Erro de comunicação
1 - Ok

- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_VerificaTruncamento

Descricao:

Retorna 1 se a impressora estiver no modo truncamento e 0 se estiver no modo arredondamento.

Protótipo:

```
int Bematech_FI_VerificaTruncamento ( char * FlagTruncamento )
```

Parâmetros:

char *	FlagTruncamento	Variável string com 1 posição para receber o flag de truncamento.
--------	-----------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_VerificaAliquotasIss

Descricao:

Retorna as alíquotas de vinculação ao ISS.

Protótipo:

```
int Bematech_FI_VerificaAliquotasIss ( char * AliquotasIss )
```

Parâmetros:

char *	AliquotasIss	Variável string com 79 posições para receber as alíquotas vinculadas ao Iss.
--------	--------------	--

Comentários:

- As alíquotas retornadas estarão separadas por vírgula. Ex.: 0300, 0500

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_Acrescimos

Descricao:

Retorna a valor acumulado dos acréscimos efetuados nos cupons.

Protótipo:

int Bematech_FI_Acrescimos (char * ValorAcrescimo)

Parâmetros:

char *	ValorAcrescimo	Variável string com 14 posições para receber o valor dos acréscimos.
--------	----------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_ContadorBilhetePassagem

Descricao:

Retorna o número de bilhetes de passagem emitidos.

Protótipo:

int Bematech_FI_ContadorBilhetePassagem (char * NumeroBilhetes)

Parâmetros:

char *	NumeroBilhetes	Variável string com 6 posições para receber o número de passagens emitidas.
--------	----------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_VerificaFormasPagamento

Descricao:

Retorna as formas de pagamento e seus valores acumulados.

Protótipo:

int Bematech_FI_VerificaFormasPagamento (char * FormasPagamento)

Parâmetros:

char *	FormasPagamento	Variável string com 3016 posições para receber as formas programadas.
--------	-----------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Essa função retorna as informações das formas de pagamento na seguinte ordem:

Descricao da forma de pagamento :	16 bytes
-----------------------------------	----------

Valor acumulado (2 casas decimais):	20 bytes
Valor recebido no último cupom (2 casas decimais):	20 bytes
Valor indicando se a forma foi usada para a emissão do cupom não fiscal vinculado (1 usado, 0 não usado):	1 byte

são retornadas 50 formas de pagamento mais a forma “Valor Recebido” e “Troco” obedecendo a ordem descrita anteriormente. As formas de pagamento vêm separadas por vírgula. Veja exemplo a seguir:

Dinheiro	00000000000000004155800000000000000015580,
Cartao Credito	0000000000000000394700000000000000000000,
Cheque	0000000000000000289400000000000000000000,
Ticket	0000000000000000900000000000000000000000,
	00,
	00,
	.
	.
	.
Valor Recebido	000000000000000011896800000000000000000015580,
Troco	00

Obs.: A impressora permite programar até 50 formas de pagamento, as formas que não estiverem programadas estarão com os valores zerados e a descrição em branco (veja exemplo acima).

Bematech_FI_VerificaRecebimentoNaoFiscal

Descrição:

Retorna os recebimentos não fiscais não vinculados programados na impressora.

Protótipo:

```
int Bematech_FI_VerificaRecebimentoNaoFiscal ( char * Recebimentos )
```

Parâmetros:

char *	Recebimentos	Variável string com 2200 posições para receber as informações.
--------	--------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.

-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Essa função retorna as informações para os 50 totalizadores não fiscais não vinculados na seguinte ordem:

Valor indicando quantas vezes cada recebimento foi utilizado :	4 bytes
Valor acumulado para cada recebimento (2 casas decimais) :	20 bytes
Descrição do recebimento:	19 bytes
Exemplo:	

```
0002000000000000000000005460Conta de água ,
001500000000000000000000145850Conta de Luz ,
00000000000000000000000000000000Carnê ,
00000000000000000000000000000000 ,
00000000000000000000000000000000 ,
00000000000000000000000000000000 ,
00000000000000000000000000000000Iptu ,
000500000000000000000000078437Conta de Telefone ,
```

.
.
.

```
00000000000000000000000000000000Prestacao de contas
```

Obs.: A impressora permite programar até 50 totalizadores não fiscais não vinculados, os totalizadores que não estiverem programados estarão com os valores zerados e a descrição em branco (veja exemplo acima).

Bematech_FI_VerificaDepartamentos

Descricao:

Retorna os departamentos e seus valores acumulados.

Protótipo:

```
int Bematech_FI_VerificaDepartamentos ( char * Departamentos )
```

Parâmetros:

char *	Departamentos	Variável string com 1019 posições para receber as informações dos departamentos.
--------	---------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação

- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Essa função retorna as informações dos departamentos na seguinte ordem:

Valor acumulado para cada departamento:	20 bytes
Valor com o número de vezes em que o departamento foi utilizado:	20 bytes
Descrição do departamento:	10 bytes

Exemplo:

```
00000000000000000000194480000000000000000000018548GERAL      ,
00000000000000000000164480000000000000000000015548Vestuario    ,
000000000000000000001581970000000000000000000158197Calcados    ,
00000000000000000000164480000000000000000000015548Equipament ,
00000000000000000000101642700000000000000001016427Alimentos    ,
000000000000000000000000000000000000000000000000000000000000 ,
.
.
.
000000000000000000000000000000000000000000000000000000000000 ,
000000000000000000000000000000000000000000000000000000000000
```

Obs.: A impressora permite programar até 20 departamentos, os departamentos que não estiverem programados estarão com os valores zerados e a descrição em branco (veja exemplo acima).

Bematech_FI_VerificaTipoImpressora**Descricao:**

Retorna o tipo de impressora.

Protótipo:

```
int Bematech_FI_VerificaTipoImpressora ( short int * TipoImpressora )
```

Parâmetros:

short int *	TipoImpressora	Variável inteira para receber o tipo da impressora (veja abaixo em “Comentários” os valores retornados).
-------------	----------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O valor do tipo de impressora pode ser:
 - 1 - Impressora fiscal, gaveta, autenticação
 - 2 - Impressora fiscal, gaveta, cutter
 - 3 - Impressora fiscal, presenter, autenticação
 - 4 - Impressora fiscal, presenter, cutter
 - 5 - Impressora bilhete de passagem, gaveta, autenticação
 - 6 - Impressora bilhete de passagem, gaveta, cutter
 - 7 - Impressora bilhete de passagem, presenter, autenticação
 - 8 - Impressora bilhete de passagem, presenter, cutter

Bematech_FI_VerificaIndiceAliquotasIss

Descricao:

Retorna os índices das alíquotas de ISS.

Protótipo:

int Bematech_FI_VerificaIndiceAliquotasIss (char * IndiceAliquotas)

Parâmetros:

int * IndiceAliquotas	Variável string com o tamanho de 48 posições para receber os índices das alíquotas de ISS.
-------------------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Os índices retornados estarão separados por vírgula. Ex.: 03, 05

Bematech_FI_ValorFormaPagamento

Descricao:

Retorna o valor acumulado em uma determinada forma de pagamento.

Protótipo:

int Bematech_FI_ValorFormaPagamento (char * FormaPagamento, char * Valor)

Parâmetros:

int * FormaPagamento	Variável string com até 16 posições com a forma de pagamento.
int * Valor	Variável string com 20 posições para receber a informação.

Valores de Retorno:

1 - Ok
-1 - Erro de execução
-2 - Parâmetro inválido
-24 - Forma de pagamento não programada
-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_ValorTotalizadorNaoFiscal

Descricao:

Retorna o valor acumulado em um determinado totalizador não fiscal.

Protótipo:

int Bematech_FI_ValorTotalizadorNaoFiscal (char * Totalizador, char * Valor)

Parâmetros:

char * Totalizador	Variável string com até 19 posições com o totalizador
char * Valor	Variável string com 20 posições para receber a informação.

Valores de Retorno:

- 1 - Ok
- 1 - Erro de execução
- 2 - Parâmetro inválido
- 25 - Totalizador não programado
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

9.0 – Funções para impressão de cheques

Bematech_FI_ProgramaMoedaSingular

Descricao:

Programa o nome da moeda no singular para a impressão de cheques. Ex. (Real)

Protótipo:

```
int Bematech_FI_ProgramaMoedaSingular ( char * MoedaSingular )
```

Parâmetros:

char *	MoedaSingular	Nome da Moeda no singular com até 19 caracteres.
--------	---------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Função usada somente na impressora MP40.

Bematech_FI_ProgramaMoedaPlural

Descricao:

Programa o nome da moeda no plural para a impressão de cheques. Ex. (Reais)

Protótipo:

```
int Bematech_FI_ProgramaMoedaPlural ( char * MoedaPlural )
```

Parâmetros:

char *	MoedaPlural	Nome da moeda no plural com até 22 caracteres.
--------	-------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Função usada somente na impressora MP40.

Bematech_FI_VerificaStatusCheque

Descricao:

Verifica o status do cheque.

Protótipo:

int Bematech_FI_VerificaStatusCheque (short int* StatusCheque)

Parâmetros:

short int *	StatusCheque	Variável inteira para receber o status do cheque (veja abaixo em "Comentários" os valores de status).
-------------	--------------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 1 - Erro de execução
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT.
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Os valores de status são:
 - 1 – Impressora ok
 - 2 – Cheque em impressão
 - 3 – Cheque posicionado
 - 4 – Aguardando o posicionamento do cheque
- Função usada somente na impressora MP40.

Bematech_FI_CancelaImpressaoCheque

Descrição:

Cancela a impressão do cheque que está sendo aguardado pela impressora. O cheque que está em impressão não pode ser cancelado.

Protótipo:

```
int Bematech_FI_CancelaImpressaoCheque()
```

Parâmetros:

Não há.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Função usada somente na impressora MP40.

Bematech_FI_ImprimeCheque

Descrição:

Imprime cheque na impressora MP40.

Protótipo:

```
int Bematech_FI_ImprimeCheque ( char * NumeroBanco, char * Valor,  
                                char * Favorecido, char * Cidade,  
                                char * Data, char * Mensagem )
```

Parâmetros:

char * NumeroBanco	Numero do banco com 3 dígitos
char * Valor	Valor do cheque com até 14 dígitos
char * Favorecido	Favorecido com até 45 caracteres
char * Cidade	Cidade com até 27 caracteres
char * Data	Data no formato ddmmaa, dd/mm/aa, ddmmaaaa ou dd/mm/aaaa
char * Mensagem	Comentários até 120 caracteres. A

mensagem será impressa 1 (uma) linha após a cidade.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 7 - Banco não localizado no arquivo de configuração BemaFi32.ini
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Função usada somente na impressora MP40.

Bematech_FI_ImprimeCopiaCheque

Descricao:

Imprime cópia do último cheque impresso.

Protótipo:

```
int Bematech_FI_ImprimeCopiaCheque ()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 7 - Banco não localizado no arquivo de configuração BemaFi32.ini
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Função usada somente na impressora MP40.

Bematech_FI_IncluiCidadeFavorecido

Descrição:

Inclui o nome da cidade e do favorecido no arquivo de configuração BemaFi32.ini.

Protótipo:

```
int Bematech_FI_IncluiCidadeFavorecido ( char * Cidade, char * Favorecido )
```

Parâmetros:

char *	Cidade	Nome da cidade com até 27 caracteres.
char *	Favorecido	Nome do favorecido com até 45 caracteres.

Valores de Retorno:

- 1 - Ok
- 1 - Erro de execução da função
- 2 - Parâmetro inválido
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Após gravar o nome da cidade e do favorecido no arquivo de configuração não é mais necessário passar essas informações como parâmetros para a função Bematech_FI_ImprimeCheque(), a não ser que seja necessário imprimir uma cidade ou um favorecido diferente aos que estão programados.
- Função usada somente na impressora MP40.

10.0 – Funções para a impressora restaurante

Bematech_FIR_RegistraVenda

Descrição:

Faz um registro de venda na mesa informada e cadastra o item no cardápio com o código informado se ele ainda não existir.

Protótipo:

```
int Bematech_FIR_RegistraVenda ( char * Mesa, char * Codigo, char * Descricao,
                                char * Aliquota, char * Quantidade, char *
                                ValorUnitario, char * FlagAcrescimoDesconto,
                                char * ValorAcrescimoDesconto )
```

Parâmetros:

char *	Mesa	String com até 4 dígitos
char *	Codigo	String com até 14 dígitos
char *	Descricao	String com até 17 caracteres
char *	Aliquota	String com o valor ou o índice da alíquota tributária. Se for o valor deve ser informado com o tamanho de 4 caracteres ou 5 com a vírgula. Se for o índice da alíquota deve ser 2 caracteres. Ex. (18,00 para o valor ou 05 para o índice)
char *	Quantidade	String com até 6 dígitos (são três casas decimais)
char *	ValorUnitario	String com até 8 dígitos (são duas casas decimais)
char *	FlagAcrescimoDesconto	"A" para acréscimo e "D" para desconto
char *	ValorAcrescimoDesconto	String com até 8 dígitos (são duas casas decimais). Se não tiver acréscimo nem desconto use "0" no valor.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Esta função já cadastra o item no cardápio ao ser enviado para a impressora, não existe mais o comando para enviar cardápio para a impressora. A impressora verifica se o item já está cadastrado se não estiver o item será cadastrado no cardápio.

Bematech_FIR_CancelaVenda

Descricao:

Essa função cancela um registro de venda da mesa informada.

Protótipo:

```
int Bematech_FIR_CancelaVenda ( char * Mesa, char * Codigo, char * Descricao,  
                                char * Aliquota, char * Quantidade, char *  
                                ValorUnitario, char * FlagAcrescimoDesconto,  
                                char * ValorAcrescimoDesconto )
```

Parâmetros:

char *	Mesa	String com até 4 dígitos
char *	Codigo	String com até 14 dígitos
char *	Descricao	String com até 17 caracteres
char *	Aliquota	String com o valor ou o índice da alíquota tributária. Se for o valor deve ser informado com o tamanho de 4 caracteres ou 5 com a vírgula. Se for o índice da alíquota deve ser 2 caracteres. Ex. (18,00 para o valor ou 05 para o índice).
char *	Quantidade	String com até 6 dígitos (são três casas decimais)
char *	ValorUnitario	String com até 8 dígitos (são duas casas decimais)
char *	FlagAcrescimoDesconto	"A" para acréscimo e "D" para desconto
char *	ValorAcrescimoDesconto	String com até 8 dígitos (são duas casas decimais). Se não tiver acréscimo nem desconto use "0" no valor.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

O cancelamento será executado somente se houver um item registrado nessa mesa e se a quantidade cancelada não exceder a quantidade registrada.

Bematech_FIR_AbreConferenciaMesa

Descricao:

Abre o cupom de conferência de mesa e imprime os itens registrados nessa mesa. Essa função mantém o cupom de conferência aberto permitindo registrar outros itens na mesa. Só são permitidos registros com o mesmo número da mesa a qual foi aberta o cupom de conferência.

Protótipo:

```
int Bematech_FIR_AbreConferenciaMesa ( char * Mesa )
```

Parâmetros:

char *	Mesa	String com até 4 dígitos
--------	------	--------------------------

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Use a função Bematech_FIR_FechaConferenciaMesa() para fechar o cupom de conferência de mesa.

Bematech_FIR_FechaConferenciaMesa

Descricao:

Fecha o cupom de conferência de mesa. Essa função permite incluir um acréscimo ou desconto sobre o valor total vendido na mesa.

Protótipo:

```
int Bematech_FIR_FechaConferenciaMesa ( char * FlagAcrescimoDesconto,  
                                         char * TipoAcrescimoDesconto,  
                                         char * ValorAcrescimoDesconto )
```

Parâmetros:

char *	FlagAcrescimoDesconto	"A" para acréscimo e "D" para desconto
char *	TipoAcrescimoDesconto	"\$" para acréscimo ou desconto por valor e "%" para percentual.
char *	ValorAcrescimoDesconto	String com no máximo 14 dígitos para acréscimo ou desconto por valor e 4 dígitos para acréscimo ou desconto por percentual (são duas casas decimais).

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

A função somente será executada se o cupom de conferência estiver aberto.

Bematech_FIR_ConferenciaMesa

Descricao:

Emite um cupom de conferência de mesa. Essa função reúne as funções Bematech_FIR_AbreConferenciaMesa e Bematech_FIR_FechaConferenciaMesa. Ela abre e fecha o cupom de conferência não permitindo registrar produtos nesse cupom de conferência.

Protótipo:

```
int Bematech_FIR_ConferenciaMesa ( char * Mesa,
                                   char * FlagAcrescimoDesconto,
                                   char * TipoAcrescimoDesconto,
                                   char * ValorAcrescimoDesconto )
```

Parâmetros:

char *	Mesa	String com até 4 dígitos
char *	FlagAcrescimoDesconto	"A" para acréscimo e "D" para desconto
char *	TipoAcrescimoDesconto	"\$" para acréscimo ou desconto por valor e "%" para percentual.
char *	ValorAcrescimoDesconto	String com no máximo 14 dígitos para

acrécimo ou desconto por valor e 4 dígitos para acréscimo ou desconto por percentual (são duas casas decimais).

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FIR_TransferenciaMesa**Descricao:**

Faz a transferência dos registros de venda da mesa de origem para a mesa de destino, se a mesa de destino já tiver itens registrados os registros serão acrescentados.

Protótipo:

```
int Bematech_FIR_TransferenciaMesa( char * MesaOrigem, char * MesaDestino )
```

Parâmetros:

char *	MesaOrigem	String com até 4 dígitos
char *	MesaDestino	String com até 4 dígitos

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Se a mesa de origem não existir o comando não será executado.

Bematech_FIR_Transferencialtem

Descrição:

Permite a transferência parcial ou total dos itens registrados em uma mesa para outra.

Protótipo:

```
int Bematech_FIR_Transferencialtem ( char * MesaOrigem, char * Codigo,
                                     char * Descricao, char * Aliquota,
                                     char * Quantidade, char * ValorUnitario,
                                     char * FlagAcrescimoDesconto,
                                     char * ValorAcrescimoDesconto,
                                     char * MesaDestino )
```

Parâmetros:

char *	MesaOrigem	String com até 4 dígitos
char *	Codigo	String com até 14 dígitos
char *	Descricao	String com até 17 caracteres
char *	Aliquota	String com o valor ou o índice da alíquota tributária. Se for o valor deve ser informado com o tamanho de 4 caracteres ou 5 com a vírgula. Se for o índice da alíquota deve ser 2 caracteres. Ex. (18,00 para o valor ou 05 para o índice).
char *	Quantidade	String com até 6 dígitos (são três casas decimais)
char *	ValorUnitario	String com até 8 dígitos (são duas casas decimais)
char *	FlagAcrescimoDesconto	"A" para acréscimo e "D" para desconto
char *	ValorAcrescimoDesconto	String com até 8 dígitos (são duas casas decimais). Se não tiver acréscimo nem desconto use "0" no valor.
char *	MesaDestino	String com até 4 dígitos

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Esta função já cadastra o item no cardápio ao ser enviado para a impressora, não existe mais o comando para enviar cardápio para a impressora. A impressora verifica se o item já está cadastrado se não estiver o item será cadastrado no cardápio.

Bematech_FIR_AbreCupomRestaurante

Descricao:

Abre o cupom fiscal na impressora restaurante e imprime os itens registrados na mesa. Se a mesa for "0000", abre o cupom e aguarda a venda dos itens.

Protótipo:

int Bematech_FIR_AbreCupomRestaurante (char * Mesa, char * CPF)

Parâmetros:

char *	Mesa	String com até 4 dígitos
char *	CPF	String com até 29 caracteres

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

A mesa "0000" pode ser usada para efetuar venda no balcão, de maneira que ao abrir uma mesa com esse número a impressora não busca registros vendidos nessa mesa e sim abre o cabeçalho do cupom fiscal e espera o registro de uma venda no balcão.

Bematech_FIR_FechaCupomRestaurante

Descricao:

Fecha o cupom fiscal na impressora restaurante com acréscimo ou desconto, usando apenas uma forma de pagamento.

Protótipo:

int Bematech_FIR_FechaCupomRestaurante (char * FormaPagamento,
char * AcrescimoDesconto,

char * TipoAcrescimoDesconto,
char * ValorAcrescimoDesconto,
char * ValorPago,
char * Mensagem)

Parâmetros:

char *	FormaPagamento	String com o nome da forma de pagamento com no máximo 16 caracteres.
char *	AcrescimoDesconto	Indica se haverá acréscimo ou desconto no cupom. 'A' para acréscimo e 'D' para desconto.
char *	TipoAcrescimoDesconto	Indica se o acréscimo ou desconto é por valor ou por percentual. '\$' para desconto por valor e '%' para percentual.
char *	ValorAcrescimoDesconto	String com no máximo 14 dígitos para acréscimo ou desconto por valor e 4 dígitos para acréscimo ou desconto por percentual.
char *	ValorPago	Valor pago com no máximo 14 dígitos.
char *	Mensagem	Mensagem promocional com até 490 caracteres.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom deve estar aberto
- Pelo menos 1 (um) item deve ter sido vendido e não pode ter sido cancelado
- A utilização dessa função elimina a obrigatoriedade de uso das funções Bematech_FI_IniciaFechamentoCupom, Bematech_FI_EfetuaFormaPagamento e Bematech_FI_TerminaFechamentoCupom que estão implementadas internamente na função.
- Para fechar o cupom usando mais de uma forma de pagamento você deve usar as mesmas funções usadas na impressora fiscal normal: Bematech_FI_IniciaFechamentoCupom para iniciar o fechamento do cupom, a Bematech_FI_EfetuaFormaPagamento várias vezes até usar todas as formas e a Bematech_FI_TerminaFechamentoCupom para fechar o cupom fiscal.

Bematech_FIR_FechaCupomResumidoRestaurante

Descrição:

Permite fechar o cupom de forma resumida, ou seja, sem acréscimo ou desconto no cupom e com apenas uma forma de pagamento. Essa função lê o subtotal do cupom para fecha-lo.

Protótipo:

```
int Bematech_FIR_FechaCupomResumidoRestaurante( char * FormaPagamento,  
                                                char * Mensagem )
```

Parâmetros:

char *	FormaPagamento	Forma de pagamento com no máximo 16 caracteres.
char *	Mensagem	Mensagem promocional com até 490 caracteres.

Valores de Retorno:

0 - Erro de comunicação
1 - Ok
-2 - Parâmetro inválido.
-4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
-5 - Erro ao abrir a porta de comunicação
-27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O cupom deve estar aberto
- Pelo menos 1 (um) item deve ter sido vendido e não pode ter sido cancelado
- A utilização dessa função elimina a obrigatoriedade de uso das funções Bematech_FI_IniciaFechamentoCupom, Bematech_FI_EfetuaFormaPagamento e Bematech_FI_TerminaFechamentoCupom que estão implementadas internamente na função.

Bematech_FIR_ContaDividida

Descrição:

Permite que a conta seja dividida por todos os clientes. Essa função termina o fechamento do cupom fiscal e imprime um cupom para cada cliente.

Protótipo:

```
int Bematech_FIR_ContaDividida ( char * NumeroCupons,
```

char * ValorPagoCliente, char * CPF)

Parâmetros:

char *	NumeroCupons	String com até 2 dígitos com o número de cupons em que a conta será dividida. O número mínimo de cupons é 2 e o máximo é 20.
char *	ValorPagoCliente	String com os valores pagos por cada cliente. Os valores devem ter no máximo 14 dígitos e serem separados por ponto e vírgula “;”. Ex.: (10,00; 5,00)
char *	CPF	String com o CPF dos clientes. Os CPF's devem ter no máximo 29 caracteres e serem separados por ponto e vírgula “;”.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- Para executar essa função você deve iniciar o fechamento do cupom fiscal usando a função Bematech_FI_IniciaFechamentoCupom() e efetuar as formas de pagamento usando a função Bematech_FI_EfetuaFormaPagamento.
- O fechamento do cupom só será executado caso a soma dos valores passados no parâmetro “ValorPagoCliente” seja exatamente igual ao total do Cupom Fiscal.
- As formas de pagamento aparecerão somente no primeiro cupom fiscal. Ao término do primeiro cupom será feita a emissão dos demais automaticamente.
- Todos os documentos fracionados terão o mesmo COO, o que os diferencia é a informação do número da via do cupom ao lado da inscrição “CUPOM FISCAL”. Exemplo: “CUPOM FISCAL 2ª VIA”
- O parâmetro CPF dessa função é opcional e é utilizado a partir do segundo cliente. Para utiliza-lo você, obrigatoriamente, deve informar o CPF para o primeiro cliente, caso contrário a dll irá ignorar o(s) CPF(s) informado(s) e não irá imprimi-lo(s). Se você quiser usar o CPF por exemplo, para o cliente dois e não quiser usar no cliente 1 você deve passar espaço em branco no parâmetro CPF da função Bematech_FIR_AbreCupomRestaurante.

Bematech_FIR_FechaCupomContaDividida

Descrição:

Essa função permite fechar o cupom fiscal com formas de pagamento e permite dividir a conta por todos os clientes.

Protótipo:

```
int Bematech_FIR_FechaCupomContaDividida ( char * NumeroCupons,
                                           char * AcrescimoDesconto,
                                           char * TipoAcrescimoDesconto,
                                           char * ValorAcrescimoDesconto,
                                           char * FormaPagamento,
                                           char * ValorFormaPagamento,
                                           char * ValorPagoCliente,
                                           char * CPF )
```

Parâmetros:

char *	NumeroCupons	String com até 2 dígitos com o número de cupons em que a conta será dividida. O número mínimo de cupons é 2 e o máximo é 20.
char *	AcrescimoDesconto	Indica se haverá acréscimo ou desconto no cupom. 'A' para acréscimo e 'D' para desconto.
char *	TipoAcrescimoDesconto	Indica se o acréscimo ou desconto é por valor ou por percentual. '\$' para desconto por valor e '%' para percentual.
char *	ValorAcrescimoDesconto	String com no máximo 14 dígitos para acréscimo ou desconto por valor e 4 dígitos para acréscimo ou desconto percentual.
char *	FormaPagamento	String com as formas de pagamento. As formas devem ser separadas por ponto e vírgula (",") se for utilizada mais de uma, e deve ter no máximo 16 caracteres cada. Ex: (Dinheiro;Cartão). É permitido a utilização de até 20 formas.
char *	ValorFormaPagamento	String com os valores das formas de pagamento. Os valores devem ter no máximo 14 dígitos e serem separados por ponto e vírgula ",".
char *	ValorPagoCliente	String com os valores pagos por cada cliente. Obedecem à mesma situação acima.

char *	CPF	String com o CPF dos clientes. Os CPF's devem ter no máximo 29 caracteres e serem separados por ponto e vírgula “;”.
--------	-----	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O fechamento do cupom somente será executado caso a soma dos valores passados no parâmetro “ValorPagoCliente” seja exatamente igual ao total do Cupom Fiscal.
- O número de formas de pagamento e a quantidade de valores devem ser idênticos, caso contrário a dll retornará “Parâmetro Inválido”.
- Se a conta deu R\$ 10,00 reais e for dividida em dois clientes cada um pagando com uma forma de pagamento diferente, onde o primeiro pagará R\$ 4,00 e o segundo R\$ 6,00, deverá ser informado no parâmetro “FormaPagamento” as duas formas, os valores deverão ser informados no parâmetro “ValorFormaPagamento”.
- Poderá ser informado uma única forma de pagamento com o valor total do Cupom Fiscal e dividi-la entre os clientes que irão pagar. Por exemplo: se a conta deu R\$ 50,00 reais e todos irão pagar em Dinheiro, informe somente a forma de pagamento “Dinheiro” com o valor e em seguida faça a divisão da conta entre os clientes.
- As formas de pagamento aparecerão somente no primeiro cupom fiscal. Ao término do primeiro cupom será feita a emissão dos seguintes automaticamente.
- O cancelamento do cupom em emissão bloqueará a emissão dos demais.
- Todos os documentos fracionados terão o mesmo COO sendo que, o que os diferencia é a informação do número da via do cupom ao lado da inscrição “CUPOM FISCAL”. Exemplo: “CUPOM FISCAL 2ª VIA”
- O parâmetro CPF dessa função é opcional e é utilizado a partir do segundo cliente. Para utilizá-lo você, obrigatoriamente, deve informar o CPF para o primeiro cliente, caso contrário a dll irá ignorar o(s) CPF(s) informado(s) e não

irá imprimi-lo(s). Se você quiser usar o CPF por exemplo para o cliente dois e não quiser usar no cliente 1 você deve passar espaço em branco no parâmetro CPF da função Bematech_FIR_AbreCupomRestaurante.

Bematech_FIR_RelatorioMesasAbertas

Descricao:

Imprime um relatório das mesas que estão abertas.

Protótipo:

int Bematech_FIR_RelatorioMesasAbertas (short int TipoRelatorio)

Parâmetros:

short int	TipoRelatorio	0 para relatório parcial e, 1 para relatório completo
-----------	---------------	--

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro Inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- No relatório parcial, para cada registro de venda será impresso apenas uma linha com as seguintes informações:

MESA	CÓDIGO	QTDE	REG/CAN	ACR/DES.
------	--------	------	---------	----------

- No relatório completo, além das informações anteriores, será impresso para cada registro mais uma linha com as seguintes informações:

DESCRIÇÃO	VLR UNIT	VLR ACR/DES
-----------	----------	-------------

Bematech_FIR_RelatorioMesasAbertasSerial

Descricao:

Retorna, pela porta serial da impressora, o relatório das mesas que estão abertas.

Protótipo:

int Bematech_FIR_RelatorioMesasAbertasSerial ()

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Essa função retorna o relatório completo no arquivo "RETORNO.TXT" no diretório especificado no arquivo ini (veja o tópico "*Arquivo de Configuração BemaFi32.ini para maiores informações*").

Bematech_FIR_ImprimeCardapio

Descricao:

Imprime os itens do cardápio com as seguintes informações: Código, Descrição, Alíquota e Quantidade.

Protótipo:

int Bematech_FIR_ImprimeCardapio ()

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FIR_CardapioPelaSerial

Descricao:

Retorna os itens do cardápio pela serial com as seguintes informações: Código, Descrição, Alíquota e Quantidade.

Protótipo:

```
int Bematech_FIR_CardapioPelaSerial ()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FIR_RegistroVendaSerial

Descricao:

Retorna os registros de venda da mesa pela porta serial.

Protótipo:

```
int Bematech_FIR_RegistroVendaSerial ( char * Mesa )
```

Parâmetros:

char *	Mesa	String com até 4 dígitos
--------	------	--------------------------

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Essa função retorna as informações dos registros de venda no seguinte formato:

Mesa:	4 bytes
Valor unitário:	8 bytes
Flag de acréscimo ou desconto ("A" - acréscimo "D" - desconto):	1 byte
Valor do acréscimo/desconto:	8 bytes
Quantidade:	6 bytes
Código:	14 bytes
Descrição:	17 bytes
Alíquota:	2 bytes
Flag de registro/cancelamento ("R" – registro "C" - cancelamento):	1 byte

Bematech_FIR_VerificaMemoriaLivre**Descricao:**

Retorna a quantidade de bytes livres na memória da impressora para registros de venda ou itens de cardápio.

Protótipo:

```
int Bematech_FIR_VerificaMemoriaLivre( char * Bytes )
```

Parâmetros:

char *	Bytes	Variável string com o tamanho de 6 posições para o valor correspondente ao bytes de memória livres.
--------	-------	---

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

11.0 – Outras funções da dll

Bematech_FI_AbrePortaSerial

Descricao:

Abre a porta serial para comunicação entre a impressora e o micro.

Protótipo:

```
int Bematech_FI_AbrePortaSerial()
```

Parâmetros:

Não há.

Valores de Retorno:

- 1 - Ok
- 4 - Arquivo BemaFi32.ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta

Comentários:

A função lê o nome da porta a ser aberta no arquivo BemaFi32.ini. Se o parâmetro “porta” estiver configurado com a palavra “Default” a função localiza onde a impressora está conectada e configura o arquivo ini.

Bematech_FI_FechaPortaSerial

Descricao:

Fecha a porta serial.

Protótipo:

```
int Bematech_FI_FechaPortaSerial()
```

Parâmetros:

Não há.

Valores de Retorno:

- 1 – Ok
- 9 – Erro ao fechar a porta

Bematech_FI_VerificaImpressoraLigada

Descricao:

Verifica se a impressora está ligada ou conectada no computador.

Protótipo:

```
int Bematech_FI_VerificaImpressoraLigada()
```

Parâmetros:

Não há.

Valores de Retorno:

- 1 - Impressora ligada.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 6 - Impressora desligada ou desconectada.

Bematech_FI_RetornoImpressora

Descricao:

Lê o retorno da impressora referente ao último comando enviado.

Protótipo:

```
int Bematech_FI_RetornoImpressora ( short int * ACK, short int * ST1,  
                                     short int * ST2 )
```

Parâmetros:

short int *	ACK	Variáveis para receber os 3 bytes de status da impressora.
short int *	ST1	
short int *	ST2	

Valores de Retorno:

- 1 – Ok
- 8 – Erro ao gravar o arquivo STATUS.TXT

Comentários:

- Essa função deve ser usada após a execução de qualquer outra função da dll para ler o retorno da impressora referente a função executada. Essa função

devolve o status da impressora (pouco papel, comando não executado, tipo de parâmetro de CMD inválido etc.).

Bematech_FI_AbreBilhetePassagem

Descrição:

Abre o cupom na impressora bilhete de passagem.

Protótipo:

```
int Bematech_FI_AbreBilhetePassagem( char * ImprimeValorFinal,
                                     char * ImprimeEnfatizado,
                                     char * Embarque, char * Destino,
                                     char * Linha, char * Prefixo,
                                     char * Agente, char * Agencia,
                                     char * Data, char * Hora,
                                     char * Poltrona, char * Plataforma )
```

Parâmetros:

char *	ImprimeValorFinal	“1” – Imprime o valor pago no final do cupom “0” – Não Imprime o valor pago no final do cupom
char *	ImprimeEnfatizado	“1” – Imprime as informações “EMBARQUE, POLTRONA e PLATAFORMA” enfatizadas. “0” – Não Imprime as informações enfatizadas (negrito).
char *	Embarque	String com até 40 caracteres com o local de embarque.
char *	Destino	String com até 40 caracteres com o local de destino.
char *	Linha	String com até 40 caracteres com o nome da linha (Ex. Curitiba x São Paulo – Executivo)
char *	Prefixo	String com até 40 caracteres
char *	Agente	String com até 40 caracteres com o nome do agente.
char *	Agencia	String com até 40 caracteres com o nome da agência.
char *	Data	String com a data de embarque no formato ddmmaa, dd/mm/aa, ddmmaaaa ou dd/mm/aaaa
char *	Hora	String com a hora do embarque no formato hhmmss ou hh:mm:ss local de embarque.

char *	Poltrona	String com até 2 caracteres com o número da poltrona
char *	Plataforma	String com até 3 caracteres com o número da poltrona.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Para o registro de itens, utilize a função Bematech_FI_VendeItem(), porém valem algumas observações:

- Os campos de código e descrição devem ser informados, porém não serão impressos;
- É obrigatório pelo convênio SINIEF 06/89 no mínimo o registro da tarifa (num totalizador tributado) e do seguro (no totalizador não incidência).

O canhoto do motorista somente será impresso se a emissão do cupom adicional for ativada durante a lacração da impressora.

Bematech_FI_MapaResumo

Descricao:

Gera o relatório "Mapa Resumo" referente ao movimento do dia. As informações serão geradas no arquivo RETORNO.TXT no diretório configurado no parâmetro "path" do arquivo ini. O diretório default configurado é o diretório raiz (C:\).

Protótipo:

```
int Bematech_FI_MapaResumo()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok

- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- As informações contidas no mapa resumo podem variar de Estado para Estado. Essa função gera o relatório com as informações padrões que são usadas na maioria dos Estados. O relatório gerado terá o seguinte layout:

Contador de Redução Z...:	0339
COO	000050
Venda Bruta	143,88
Venda Líquida.....	122,95
Cancelamentos.....	7,68
Acréscimos.....	2,72
Descontos	0,62
ISS.....	12,63
Isonção	9,45
Não Incidência.....	3,15
Substituição Tributária.....	12,61
1700.....	1,25
0100.....	57,12
0200.....	10,25
0400.....	10,56
0600.....	4,23
0700.....	4,59
0900.....	0,10
1000.....	0,20
1100.....	4,35
1500.....	5,09

- Após a linha “Substituição Tributária” serão gravadas as informações de vendas referente as alíquotas de ICMS. Portanto, o número de linhas irá variar de acordo com o número de alíquotas de ICMS cadastradas na impressora.
- As informações contidas no mapa resumo serão referentes aos dados da última redução Z, portanto, para gerar o mapa resumo referente ao movimento do dia você deve executar essa função após a redução Z, caso contrário o mapa resumo será gerado com as informações referente ao movimento do dia anterior.

Bematech_FI_AberturaDoDia

Descricao:

Faz a abertura do caixa emitindo um suprimento e uma leitura X. Essa função grava o COO inicial e o Grande Total inicial que serão usados na função Bematech_FI_RelatorioTipo60Mestre. Portanto, se você for emitir o relatório "tipo 60 mestre" é obrigatório o uso dessa função.

Protótipo:

```
int Bematech_FI_AberturaDoDia( char * Valor, char * FormaPagamento )
```

Parâmetros:

char *	Valor	Valor do suprimento com até 14 dígitos (2 casas decimais). Se o valor for "0" ou não for informado não será feito o suprimento.
char *	FormaPagamento	Forma de pagamento com até 16 caracteres. Se não for informado, o suprimento será feito em dinheiro.

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_FechamentoDoDia

Descricao:

Faz o fechamento do dia emitindo uma redução Z. Essa função grava o COO final e o Grande Total final que serão usados na função Bematech_FI_RelatorioTipo60Mestre.

Protótipo:

```
int Bematech_FI_FechamentoDoDia()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_RelatorioTipo60Analitico

Descricao:

Gera o relatório "Tipo 60 analítico" exigido pelo convênio de ICMS 50/2001. As informações serão geradas no arquivo RETORNO.TXT no diretório configurado no parâmetro "path" do arquivo ini. O diretório default é o diretório raiz (C:\).

Protótipo:

```
int Bematech_FI_RelatorioTipo60Analitico( )
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O relatório gerado terá o seguinte layout:

Tipo do relatório	60
Modelo do Relatório	A
Data de emissão	26/06/2001
Número PDV ou ECF	0001
Isenção	7,68
Não Incidência.....	2,72
Substituição Tributária.....	0,62
1700.....	0,00
1800.....	0,00
1500.....	0,00
0500.....	0,00

0800.....:	0,00
1000.....:	0,00
2000.....:	0,00
0300.....:	0,00

- Após a linha "Substituição Tributária" serão gravadas as informações de vendas referentes às alíquotas tributárias. Portanto, o número de linhas irá variar de acordo com o número de alíquotas cadastradas na impressora.

As informações contidas no relatório tipo 60 analítico são referentes aos dados da última redução Z. Portanto, para gerar o relatório referente ao movimento do dia você deve executar essa função após a redução Z, caso contrário as informações geradas serão referentes ao movimento do dia anterior.

Bematech_FI_RelatorioTipo60Mestre

Descricao:

Gera o relatório "Tipo 60 mestre" exigido pelo convênio de ICMS 50/2001. As informações serão geradas no arquivo RETORNO.TXT no diretório configurado no parâmetro "path" do arquivo ini. O diretório default é o diretório raiz (C:\).

Protótipo:

```
int Bematech_FI_RelatorioTipo60Mestre()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

- O relatório gerado terá o seguinte layout:

Tipo do relatório	60
Modelo do relatório	M
Data de emissão	26/06/2001
Número PDV ou ECF	0001
Número de série	4708991023070

Modelo do documento fiscal.....:	CF
COO inicial.....:	000887
COO final.....:	001427
Contador de reduções.....:	0106
Grande total inicial.....:	3.845,16
Grande total final.....:	5.482,51

As informações contidas no relatório tipo 60 mestre são alimentadas pela função Bematech_FI_AberturaDoDia e Bematech_FI_FechamentoDoDia (veja essas funções para maiores informações). A função Bematech_FI_RelatorioTipo60Mestre deve ser executada após a função Bematech_FI_FechamentoDoDia() ou após a redução Z (caso a redução tenha sido executada automaticamente pela impressora).

Bematech_FI_ImprimeDepartamentos

Descricao:

Imprime os departamentos e seus valores acumulados em um relatório gerencial. Será emitida uma leitura X antes. Essas informações eram impressas na leitura X até a versão 3.0 e foram retiradas por solicitação do fisco.

Protótipo:

```
int Bematech_FI_ImprimeDepartamentos()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Bematech_FI_ImprimeConfiguracoesImpressora

Descricao:

Imprime configurações da impressora fiscal em um relatório gerencial. Será emitida uma leitura X antes. Veja abaixo em “comentários” as informações que serão impressas.

Protótipo:

```
int Bematech_FI_ImprimeConfiguracoesImpressora()
```

Parâmetros:

Não há

Valores de Retorno:

- 0 - Erro de comunicação
- 1 - Ok
- 2 - Parâmetro inválido.
- 4 - Arquivo ini não encontrado ou parâmetro inválido para o nome da porta
- 5 - Erro ao abrir a porta de comunicação
- 8 - Erro ao criar ou gravar no arquivo STATUS.TXT ou RETORNO.TXT
- 27 - Status da impressora diferente de 6,0,0 (Ack, St1 e St2)

Comentários:

Serão impressas as seguintes configurações:

Data da impressora	27/06/2001
Hora da impressora	11:38:37
Versão do firmware	0310
Número de série	4708991023070
Número da loja	0001
Número do caixa	0001
Símbolo da moeda	R\$
Modo de operação	Arredondamento
Horário de verão	Não
Já houve redução Z	Não
Memória fiscal	Com espaço
Versão da dll	1.9
log da dll	Desabilitado
Lin. Imp. após pouco papel	0
Tipo Impressora	fiscal + gaveta+autenticação

12.0 – Funções para transferência eletrônica de fundos (TEF)

Bematech_FITEF_VendaCartao

Descrição:

Pedido de autorização para transação por meio de cartão.

Protótipo:

Int Bematech_FITEF_VendaCartao (char * Identificacao,
char * ValorCompra)

Parâmetros:

char *	Identificacao	String numérica até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada nova solicitação. Este mesmo conteúdo é devolvido no arquivo de resposta.
char *	ValorCompra	String numérica com até 14 dígitos. Valor total da forma de pagamento.

Valores de Retorno:

- 0 - Gerenciador inativo;
- 1 - Ok
- 2 - Parâmetro inválido.
- 6 - Impressora desligada ou desconectada.
- 18 - Não foi possível abrir arquivo .001 ou .sts .
- 19 - Número de identificação diferente do arquivo intpos.001 (*)
- 20 - Operação cancelada pelo operador.
- 21 - Transação não aprovada
- 23 - Não foi possível terminar a operação.
- 26 - Transação já realizada.
- 28 - Não há informações para serem impressas.

Comentários:

(*) Esse retorno ocorrerá quando, no arquivo de resposta, o parâmetro **Identificacao** não for o mesmo que o passado pela automação. Isso indica que o arquivo **IntPos.001** de alguma operação passada não foi excluído ou a operação não foi confirmada.

Bematech_FITEF_Status

Descrição:

Verifica se o Gerenciador padrão está ativo.

Protótipo:

```
int Bematech_FITEF_Status (char *cIdentificacao)
```

Parâmetros:

char *	Identificacao	String numérica com até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada nova solicitação. Este mesmo conteúdo é devolvido no arquivo de resposta.
--------	---------------	---

Valores de Retorno:

- 1 - Gerenciador ativo.
- 0 - Gerenciador inativo.
- 2 - Parâmetro inválido
- 18 - Não foi possível criar arquivo .001 ou .STS.

Bematech_FITEF_ConfirmaVenda

Descrição:

Confirma a venda e impressão do cupom.

Protótipo:

```
int Bematech_FITEF_ConfirmaVenda (char * Identificacao, char * ValorCompra,  
char * Header)
```

Parâmetros:

char *	Identificacao	String numérica com até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada
--------	---------------	---

		nova solicitação. Este mesmo conteúdo é devolvido no arquivo de resposta.
char *	ValorCompra	String numérica com 14 dígitos (duas casas decimais)
char *	Header	String alfanumérica de 3 bytes. Identifica o tipo de operação (*).

Valores de Retorno:

- 1 - Ok.
- 2 - Parâmetro inválido.
- 18 - Não foi possível abrir arquivo .001 ou .sts.
- 19 - Número de identificação diferente do arquivo intpos.001
- 20 - Operação cancelada pelo operador.
- 21 - Transação não aprovada
- 23 - Não foi possível terminar a operação.
- 26 - Transação já realizada.
- 28 - Não há informações para serem impressas.

Comentários:

(*) Os tipos de operações aceitas são: "CNF","CRT","CHQ" → Para confirmar operação.

Bematech_FITEF_NaoConfirmaVendaImpressao**Descrição :**

Não confirma a venda e/ou a impressão do cupom.

Protótipo :

```
int Bematech_FITEF_NaoConfirmaVendaImpressao ( char * Identificacao,  
                                                char * ValorCompra )
```

Parâmetros:

char *	Identificacao	String numérica com até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada nova solicitação. Este mesmo conteúdo é devolvido no arquivo de resposta.
char *	ValorCompra	String numérica com 14 dígitos com o

valor da forma de pagamento (duas casas decimais).

Valores de Retorno:

- 1 - Ok.
- 2 - Parâmetro inválido.
- 18 - Não foi possível abrir arquivo .001 ou .sts.
- 19 - Número de identificação dirente do arquivo intpos.001
- 20 - Operação cancelada pelo operador.
- 21 - Transação não aprovada
- 23 - Não foi possível terminar a operação.
- 26 - Transação já realizada.
- 28 - Não há informações para serem impressas.

Bematech_FITEF_CancelaVendaCartao**Descrição:**

Cancelamento de uma venda feita com qualquer tipo de cartão.

Protótipo:

```
int Bematech_FITEF_CancelaVendaCartao (char * Identificacao, char *  
ValorCompra, char * Nsu, char *  
Data, char * Hora, char * Rede)
```

Parâmetros:

char *	Identificacao	String numérica de até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada nova solicitação. Este mesmo conteúdo é devolvido no arquivo de resposta.
char *	ValorCompra	String numérica com 14 dígitos com o valor da forma de pagamento (duas casas decimais)
char *	Nsu	String numérica de até 12 bytes. Indica o número de sequência (NSU - Número Sequencial Único) da transação atribuído pelo Host (Sistema da Rede de Cartão que recebe e trata as solicitações das transações TEF).

		Quando este campo é enviado do gerenciador padrão para o aplicativo de automação comercial, ele representa o NSU da transação a ser tratada.
char *	NumeroCupom	Número do documento fiscal vinculado à forma de pagamento. String numérica de até 12 bytes.
char *	Data	Data da transação no formato DDMMAAAA
char *	Hora	Hora da transação no formato HHMMSS
char *	Rede	Nome da rede (operadora) que tratou a transação. String alfanumérica com até 8 bytes.

Valores de Retorno:

- 1 - Ok;
- 2 - Parâmetro inválido;
- 18 - Não foi possível abrir arquivo .001 ou .sts;
- 19 - Número de identificação diferente do arquivo intpos.001
- 20 - Operação cancelada pelo operador.
- 21 - Transação não aprovada
- 23 - Não foi possível terminar a operação.
- 26 - Transação já realizada.
- 28 - Não há informações para serem impressas.

Bematech_FITEF_ADM**Descrição:**

Permite o acionamento da Solução TEF Discado para execução das funções administrativas.

Protótipo :

int Bematech_FITEF_ADM (char * Identificacao)

Parâmetros:

char *	clidentificacao	String numérica com até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada nova solicitação. Este mesmo
--------	-----------------	---

conteúdo é devolvido no arquivo de resposta.

Valores de Retorno:

- 0 - Gerenciador Inativo.
- 1 - Ok.
- 2 - Parâmetro Inválido
- 18 - Não foi possível abrir/criar arquivo .001 ou .sts.
- 19 - Número de identificação diferente do arquivo intpos.001
- 20 - Operação cancelada pelo operador
- 23 - Não foi possível terminar a operação.

Bematech_FITEF_ImprimeTEF**Descrição:**

Emita o comprovante não fiscal vinculado referente à operação TEF.

Protótipo:

```
int Bematech_FITEF_ImprimeTEF (char * Identificacao, char *  
                                FormaPagamento, char * ValorCompra)
```

Parâmetros:

char *	cIdentificacao	String numérica com até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada nova solicitação. Este mesmo conteúdo é devolvido no arquivo de resposta.
char *	FormaPagamento	String com a descrição da forma de pagamento com até 16 bytes.
char *	ValorCompra	String numérica com 14 posições com o valor do pagamento (duas casas decimais).

Valores de Retorno:

- 1 - Ok .
- 2 - Parâmetro inválido.
- 18 - Não foi possível abrir/criar arquivo .001 ou .sts.
- 19 - Número de identificação diferente do arquivo intpos.001
- 20 - Operação cancelada pelo operador.

-29 - Erro no envio da "NCN".

Bematech_FITEF_ImprimeRelatorio

Descrição:

Imprime todo tipo de relatório gerado por uma operação Administrativa

Protótipo:

```
int Bematech_FITEF_ImprimeRelatorio( void )
```

Parâmetros:

Não há.

Valores de Retorno:

- 1 - Ok.
- 2 - Parâmetro inválido.
- 18 - Não foi possível abrir/criar arquivo .001 ou .sts .
- 19 - Número de identificação diferente do arquivo intpos.001
- 20 - Operação cancelada pelo operador
- 21 - Transação não aprovada
- 23 - Não foi possível terminar a operação.
- 26 - Transação já realizada.
- 28 - Não há informações para serem impressas.
- 29 - Erro no envio da "NCN".

Comentário:

Após ser feita a impressão de algum relatório ou impressão de cancelamento, via função Administrativa, é preciso mandar uma Confirmação de que a operação foi feita com sucesso.

Bematech_FITEF_VendaCompleta

Descrição:

Faz uma venda com, executa o fechamento do cupom fiscal após a venda e imprime o cupom TEF.

Protótipo:

```
int Bematech_FITEF_VendaCompleta (char * Identificacao,  
char * ValorCompra,  
char * FormaPagamento, char * Texto)
```

Parâmetros:

char *	Identificação	String numérica com até 10 bytes com o número do documento (NSU).
char *	ValorCompra	String numérica com até 14 dígitos com o valor da compra (duas casas decimais).
char *	FormaPagamento	String com até 16 bytes com a descrição da forma de pagamento.
char *	Texto	Mensagem promocional.

Retornos:

- 1 - Ok.
- 0 - Gerenciador inativo
- 1 - Erro de execução (erro na leitura do número do cupom)
- 2 - Parâmetro Inválido.
- 6 - Impressora desligada ou desconectada
- 18 - Não foi possível abrir/criar arquivo .001 ou .sts .
- 19 - Número de identificação diferente do arquivo intpos.001
- 20 - Operação cancelada pelo operador
- 21 - Transação não aprovada
- 23 - Não foi possível concluir a operação
- 26 - Transação já realizada
- 28 - Não há informações para serem impressas
- 29 - Erro no envio da "NCN".

Observações:

- No arquivo BemaFI32.ini existe uma chave chamada VIAS. Nela contém o número de vias a serem impressas. No caso de homologação a chave vias deve ser igual a 2(dois). Isso serve para impressoras que usam papel que não têm via carbonada.

Bematech_FITEF_ConfiguraDiretorioTef**Descrição:**

Configura no arquivo bemafi32.ini o diretório onde será gerado os arquivos de envio e recepção das operações tef.

Protótipo:

int Bematech_FITEF_ConfiguraDiretorioTef (char * PathReq,
char * PathResp)

Parâmetros:

char *	PathReq	String com o nome do diretório onde serão gerados os arquivos de requisição ou solicitação. Exemplo: C:\TEF_DIAL\REQ
char *	PathResp	String com o nome do diretório onde serão gerados os arquivos de resposta da operadora de cartão. Exemplo: C:\TEF_DIAL\RESP

Retornos:

- 1 - Ok.
- 1 - Erro de execução (erro ao gravar no arquivo ini)
- 2 - Parâmetro Inválido.

Bematech_FITEF_VendaCheque

Descrição:

Pedido de autorização para transação com cheque.

Protótipo:

```
Int Bematech_FITEF_VendaCheque(char * Identificacao,  
                                char * ValorCompra)
```

Parâmetros:

char *	Identificacao	String numérica até 10 bytes. Indica o número de controle da solicitação que está sendo feita. Este número é gerado pelo aplicativo de automação comercial, o qual deverá colocar um conteúdo diferente a cada nova solicitação. Este mesmo conteúdo é devolvido no arquivo de resposta.
char *	ValorCompra	String numérica com até 14 dígitos. Valor total da forma de pagamento.

Valores de Retorno:

- 0 - Gerenciador inativo;
- 1 - Ok
- 1 - Erro de execução (não conseguiu ler o numero do cupom)

- 2 - Parâmetro inválido.
- 18 - Não foi possível abrir arquivo .001 ou .sts .
- 19 - Número de identificação diferente do arquivo intpos.001 (*)
- 20 - Operação cancelada pelo operador.
- 21 - Transação não aprovada
- 23 - Não foi possível concluir a operação.
- 26 - Transação já realizada.
- 28 - Não há informações para serem impressas.

Bematech_FITEF_ApagaResiduos

Descrição:

Apaga os arquivos gerados pela operação TEF. Essa função deve ser executada após a função Bematech_FITEF_VendaCheque.

Protótipo:

void Bematech_FITEF_ApagaResiduos(void)

Parâmetros:

Não há

Valores de Retorno:

Não há

14.0 – Principais diferenças entre a dll mp20fi32 e a dll bemafi32

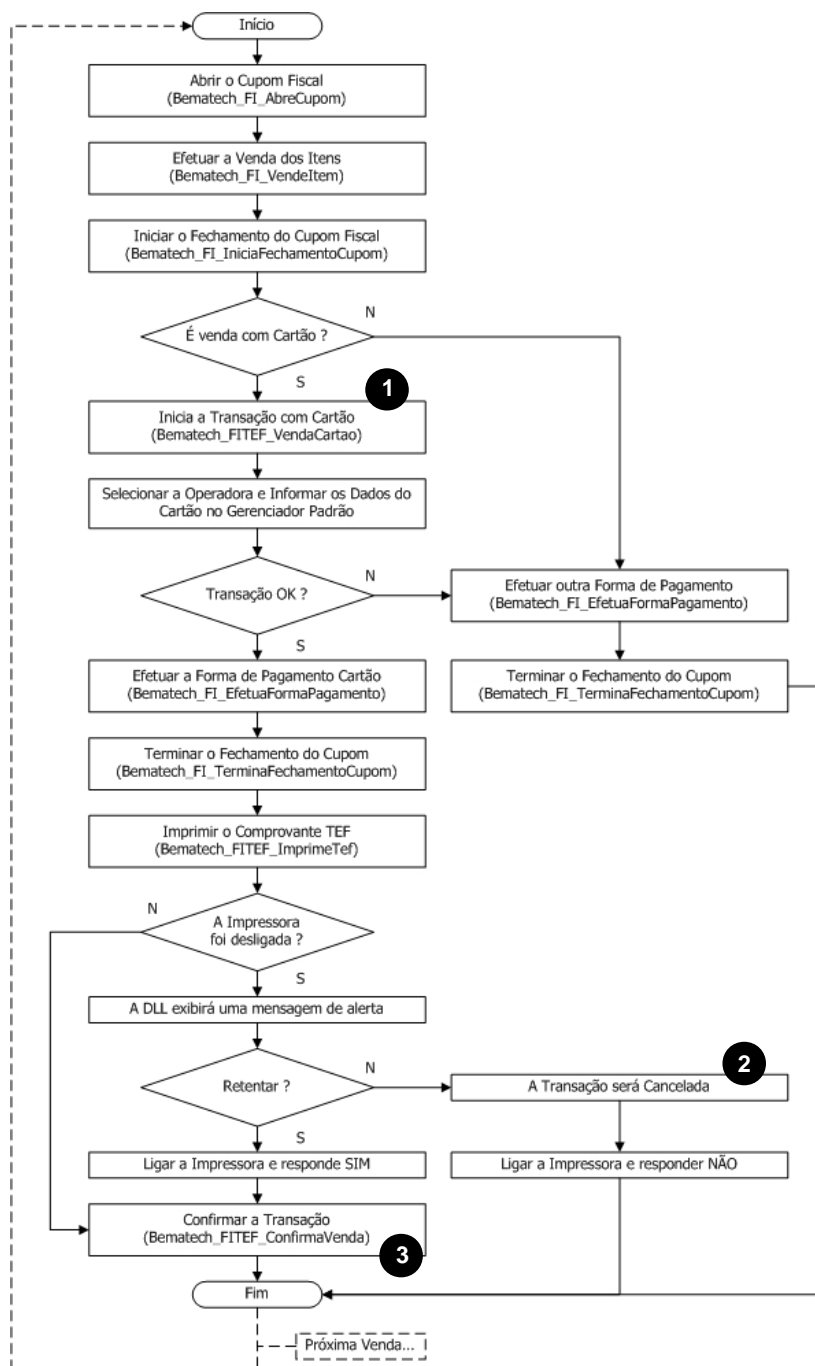
Comando	MP20FI32	BemaFI32
Venda de Item	No parâmetro “Alíquota” deve ser informado o índice da alíquota.	No parâmetro “Alíquota” deve ser informado o valor da alíquota.
Formas de Pagamento	Devem ser programadas no início do dia ou podem ser programadas no fechamento do cupom.	Não existe função para programar as formas de pago. A dll programa automaticamente ao serem usadas.
Leitura das Alíquotas	Retorna no primeiro byte o número de alíquotas cadastradas e na sequência as alíquotas.	Retorna apenas as alíquotas cadastradas separadas por vírgula.
Flag de Vinculação	Retorna 2 bytes indicando quais alíquotas são vinculadas ao ISS.	Retorna os valores das alíquotas vinculadas ao ISS.
Flag de EPROM Conectada	Retorna o caracter e o valor decimal do caracter: U 85 – Eprom conectada ~ 170– Eprom desconectada	Retorna 1 byte com os seguintes valores: 1 – Eprom conectada 0 – Eprom desconectada
Flag de Intervenção Técnica	Retorna o caracter e o valor decimal do caracter: U 85 – modo normal ~ 170– intervenção técnica	Retorna 1 byte com os seguintes valores: 1 – modo normal 0 – intervenção técnica
Flag de Truncamento	Retorna o valor decimal do flag: 255 – Arredondamento 000 – Truncamento	Retorna 1 byte com os seguintes valores: 1 – Truncamento 0 – Arredondamento
Flags Fiscais	Retorna o caracter e o valor decimal do caracter: \$ 36 – Horário de verão selecionado; – Permite cancelar o cupom fiscal	Retorna o valor decimal do flag: Ex.: 36

Principais diferenças entre a dll mp20fi32 e a dll bemafi32

Leitura do Estado da Gaveta de Dinheiro	Retorna o valor do estado: 255 – Sensor em nível um 000 – Sensor em nível zero	Retorna 1 byte com os seguintes valores: 1 – Sensor em nível um 0 – Sensor em nível zero
Status do Cheque	Retorna um caracter ASCII com o valor do status.	Retorna 1 byte com os seguintes valores: 1 – Impressora ok 2 – Cheque em impressão 3 – Cheque posicionado 4 – Aguardando o posicionamento do cheque

14.0 – Fluxogramas TEF

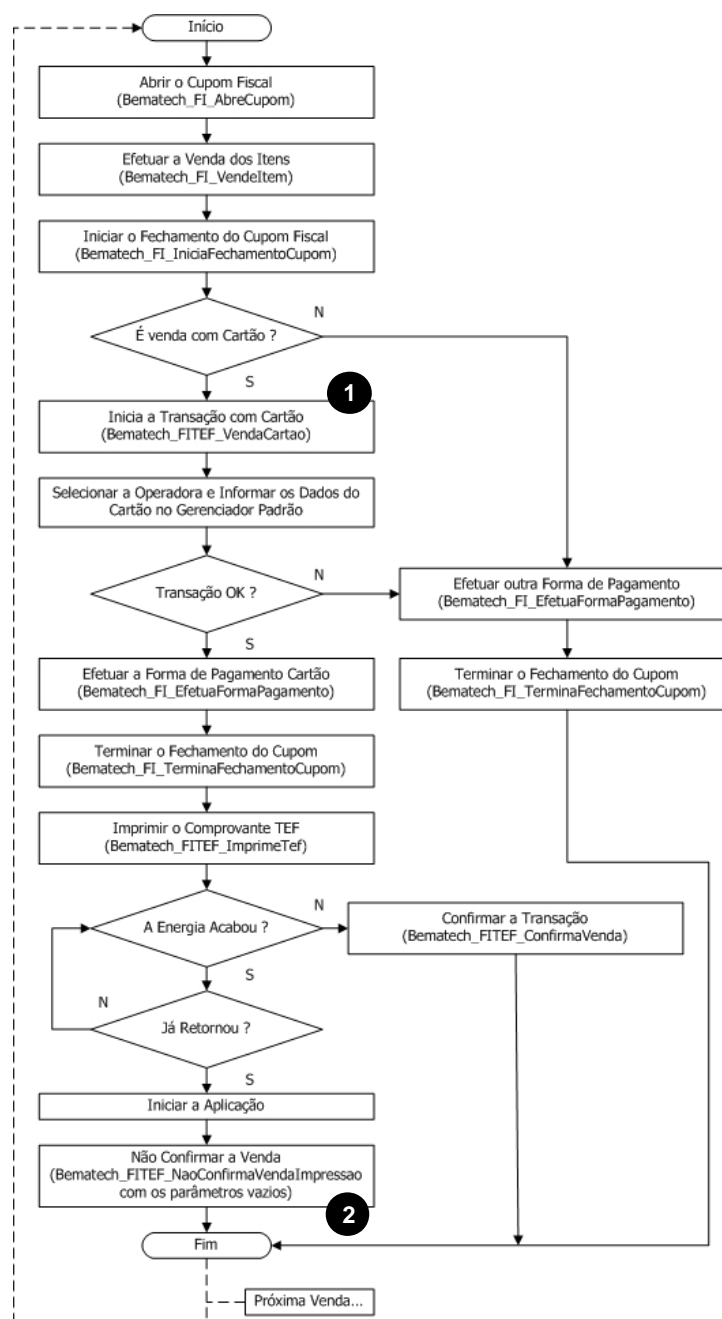
- Simulação de “Venda com Cartão” desligando a impressora durante a transação:



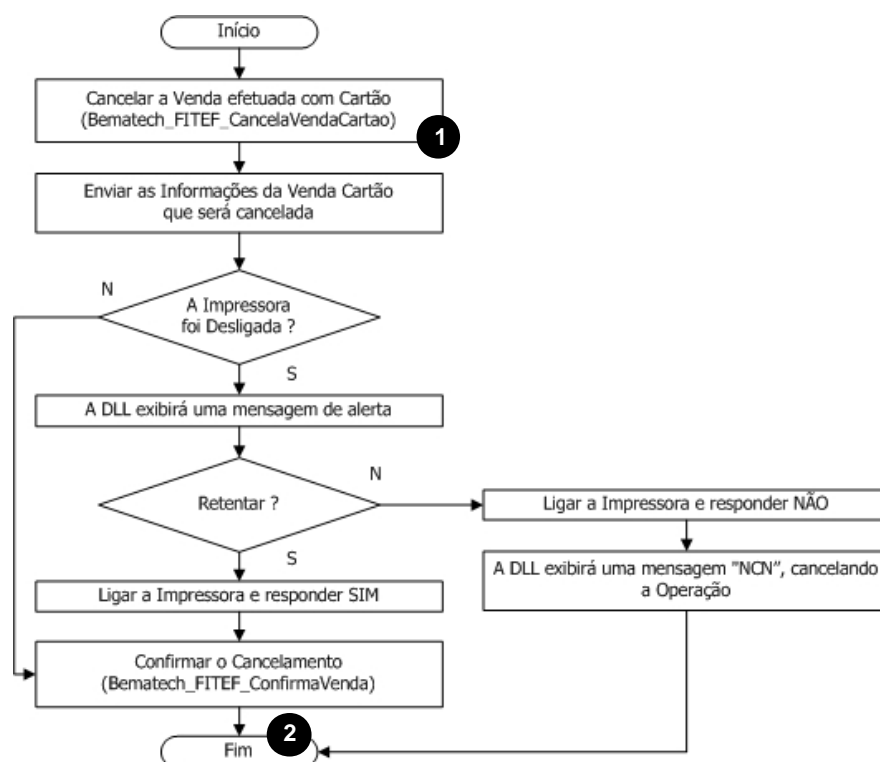
- 1** Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.

- 2 O retorno da função **Bematech_FITEF_ImprimeTEF** será -20 (menos vinte) “Cancelado pelo Operador”.
- 3 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Venda com Cartão” com queda de energia durante a transação:

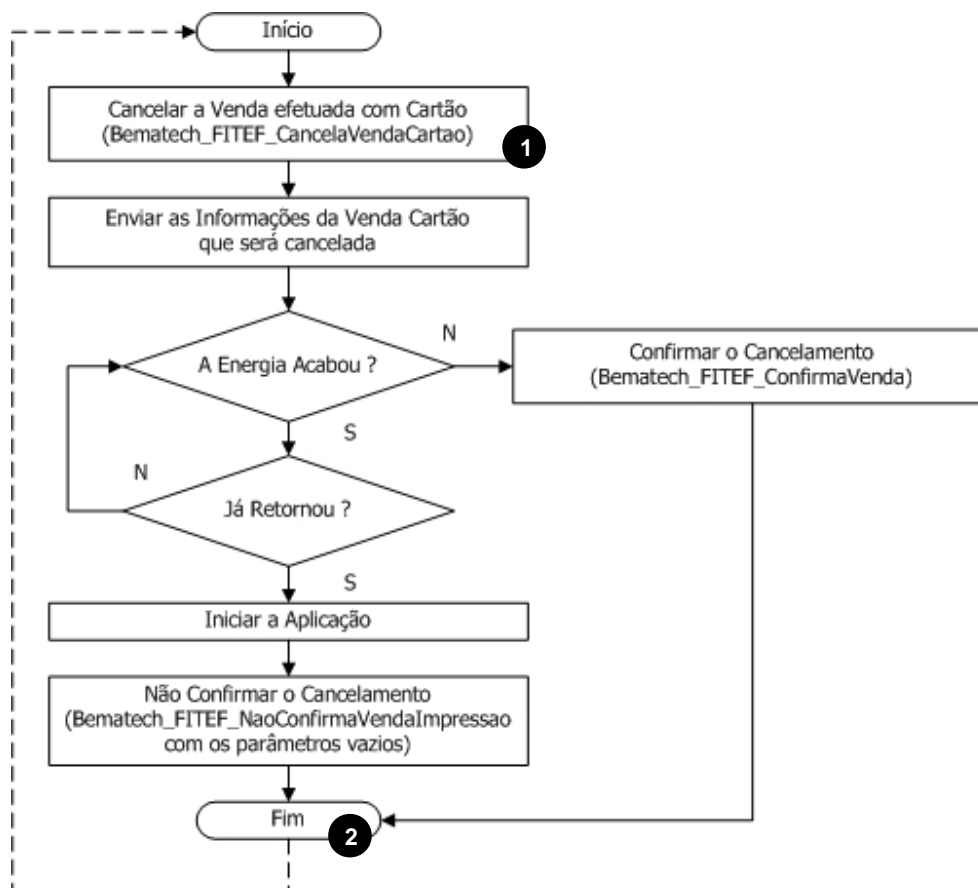


- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
 - 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.
- **Simulação de “Cancelamento de Venda” com cartão desligando a impressora durante a transação:**



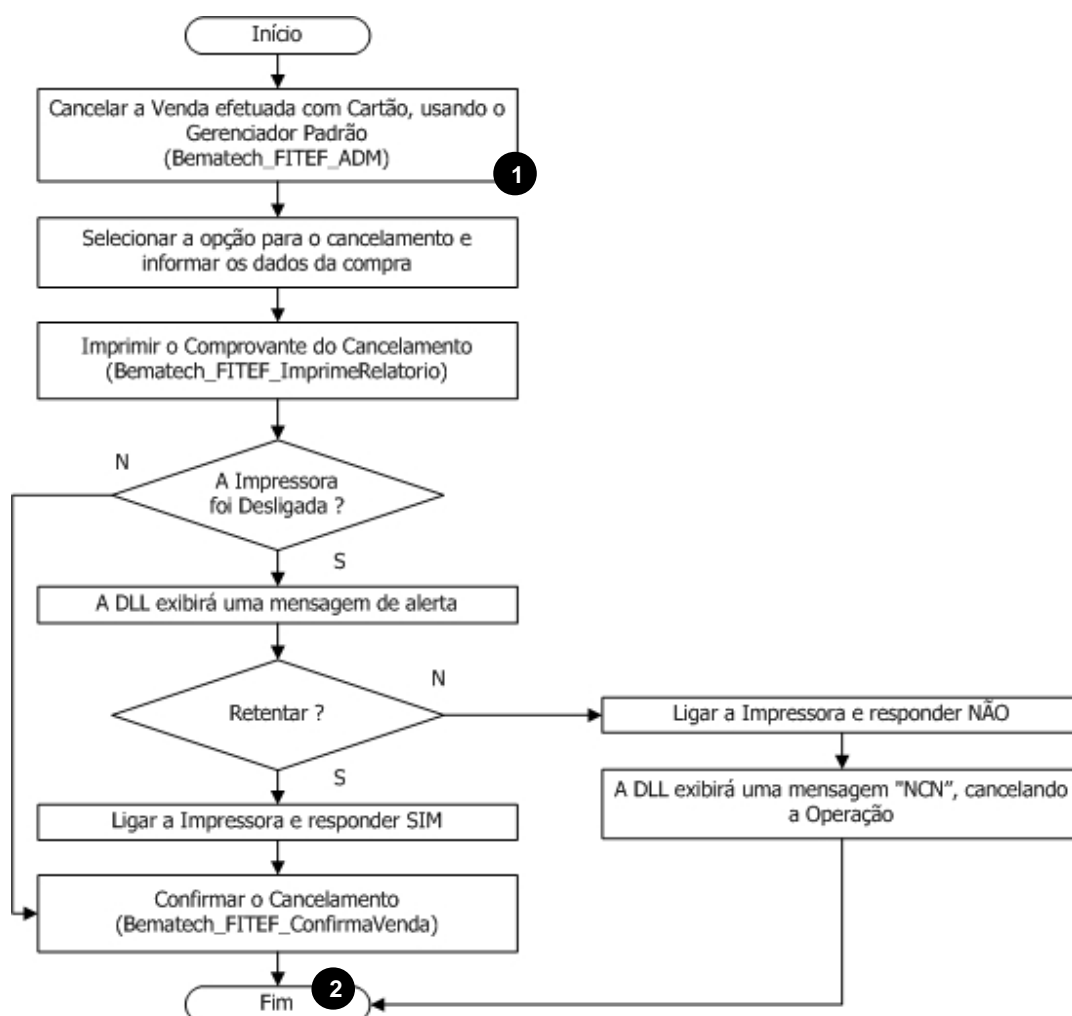
- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Cancelamento de Venda” com cartão com queda de energia durante a transação:



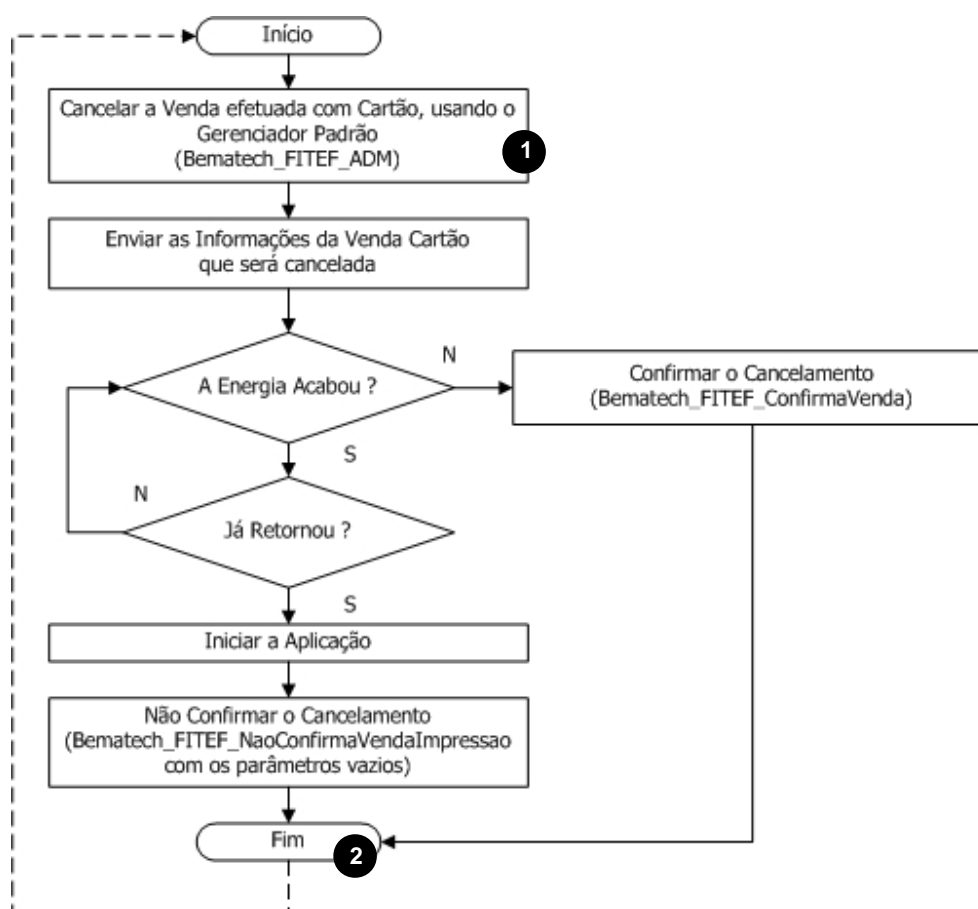
- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao ser inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Cancelamento de Venda” com cartão desligando a impressora durante a transação, usando o ADM:



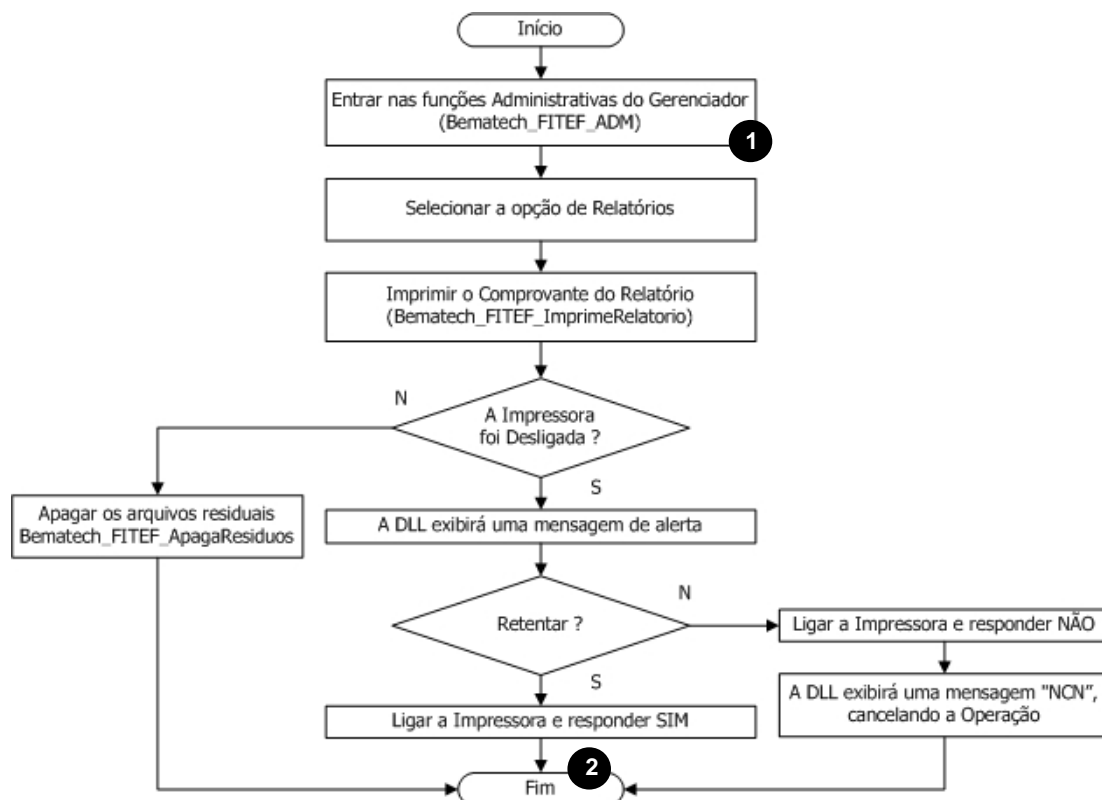
- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Cancelamento de Venda” com cartão com queda de energia durante a transação, usando o ADM:



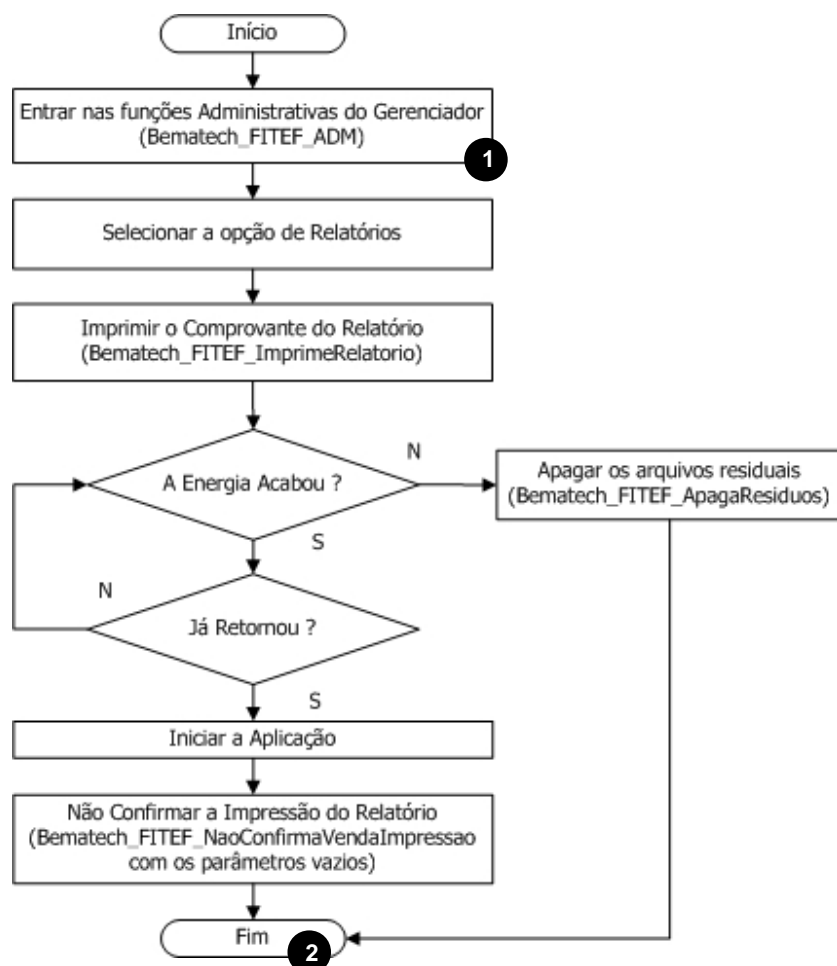
- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Impressão de Relatório” desligando a impressora durante a transação:



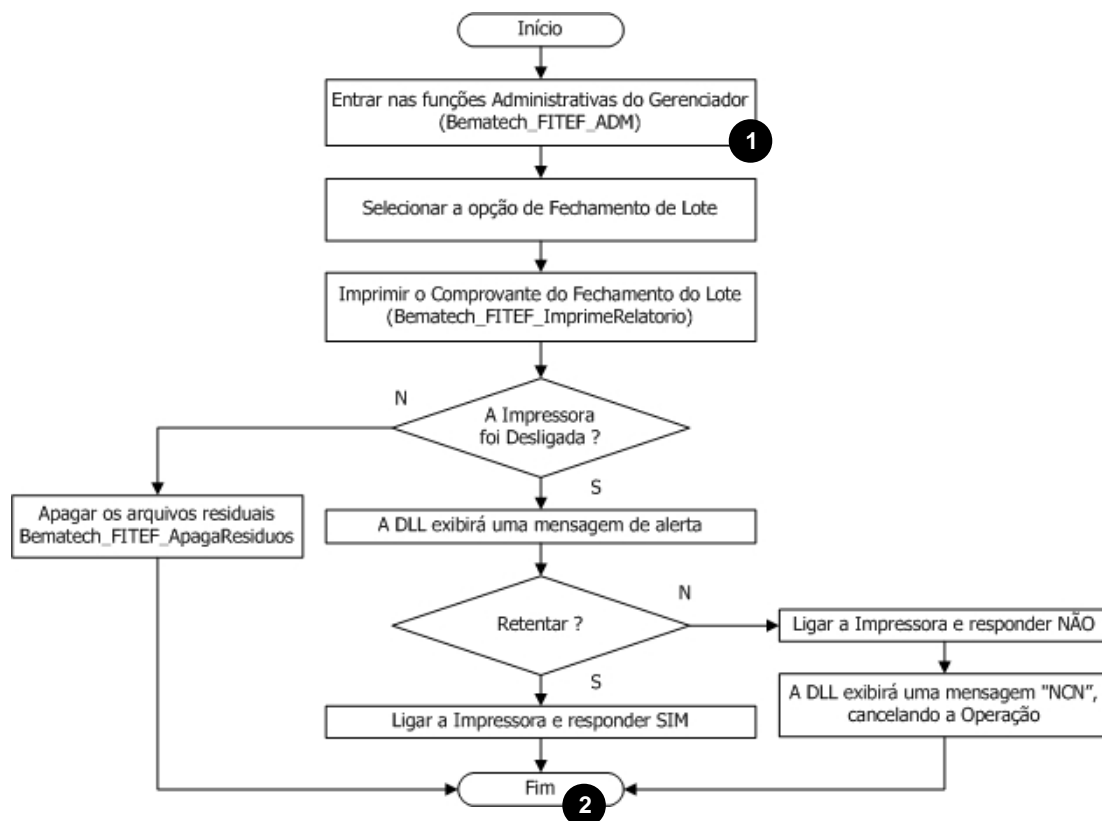
- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Impressão de Relatório” com queda de energia durante a transação:



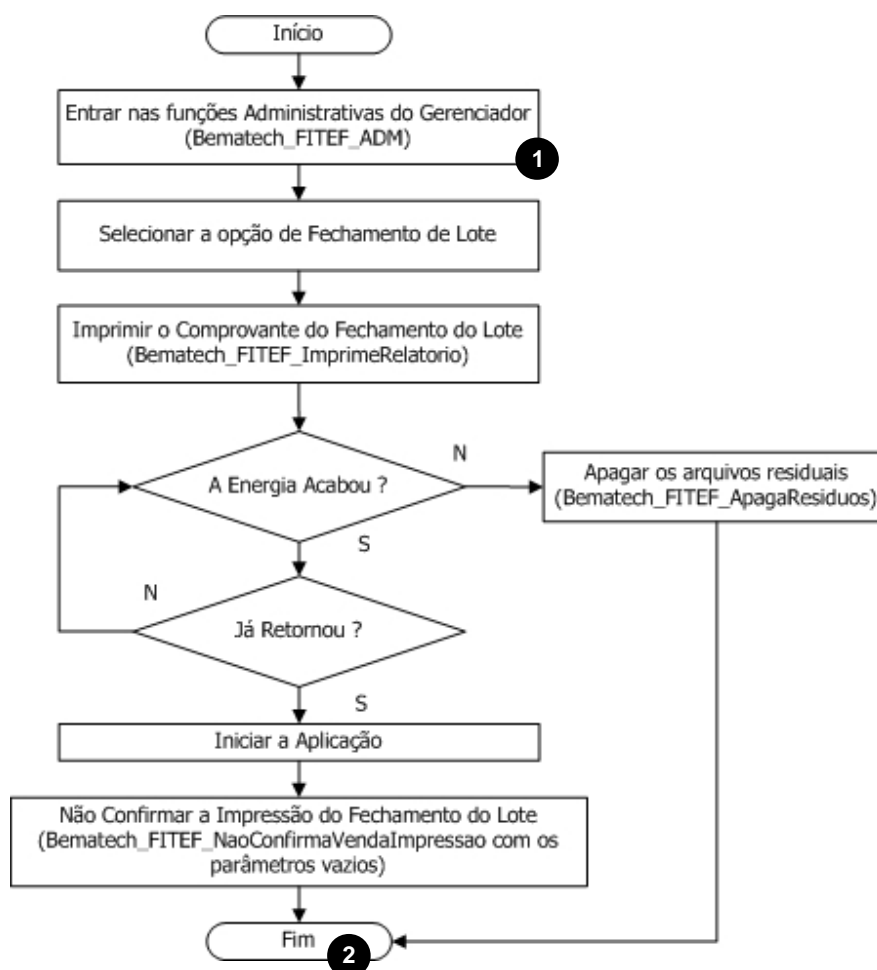
- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Fechamento de Lote” desligando a impressora durante a transação:



- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.

- Simulação de “Fechamento de Lote” com queda de energia durante a transação:



- 1 Gravar um arquivo texto para identificar que foi inicializada uma operação TEF. Caso ocorra uma queda de energia o software poderá identificar que há uma transação pendente, ao seu inicializado.
- 2 Apagar o arquivo texto, criado no início da transação, que indicava que havia uma transação pendente.