



Pinpad Abecs

Protocolo de Comunicação e Funcionamento

Versão: 2.20 (28-fev-25)

Copyright 2013-2025 © Abecs

Este documento possui informações de propriedade intelectual exclusiva da Abecs, não podendo ser reproduzido, utilizado ou divulgado por qualquer modo ou meio, total ou parcialmente, para qualquer fim, sem a devida autorização prévia.

Histórico

| Versão | Data | Autor | Comentários |
|--------|-----------|-------------|---|
| 2.20 | 28/fev/25 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ Quantidade mínima de chaves aumentada para 64 ^{[01][04]}. ▪ Corrigido tamanho de CHP_CMD no comando "CHP" ^[02]. ▪ Revisada observação dos campos GOC_CONNECT ^[03] e SPE_GOXOPT ^[04]. |
| 2.20 | 20/dez/24 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ Incorporação das Atualizações de Especificação SU001 ^{[01][02][03][04][05][06][07][08][09][10]}, SU002 ^[11], SU004 ^{[12][13]}, SU005 ^{[14][15][16]} e SU006 ^{[17][18]}. ▪ Incluída referência ao kernel EMVCo C-8 (uso futuro) ^[19]. ▪ Comando "CHP" retorna ↵ST_INVPARM caso não suporte o SAM referido ^{[21][22][23]}. ▪ Revisados tempos de espera no protocolo de comunicação serial ^{[25][26]}. ▪ O parâmetro T1_CTLSAPPVER da Tabela de AID não tem mais uso e foi obsoleto ^[28]. ▪ Melhorado detalhamento do TTQ Visa ^[29] e Discover ^[30]. ▪ Incluídas em PP_CAPAB (na resposta de "GIX") informações de suporte a cartão magnético, ICC, SAM e ausência de display ^[31]. ▪ Incluído suporte a cartões Quick Pass (Union Pay sem contato) ^{[32][33][34][35][36]}. ▪ Incluído PP_CTLSAPAB, PP_BATTINFO e PP_COMMINFO na resposta do comando "GIX" ^{[37][45][46][47]}. ▪ Revisados retornos de erro de CTLS em "GCR" e "GCX" ^{[48][50][51][52][53]}. ▪ Objeto <i>Track 1 Data</i> (tag 56h) é dado protegido ^[54]. ▪ O comando "GCR" também mostra o valor da transação ao solicitar o cartão ^[57]. ▪ Melhoria na descrição de quais campos são retornados na resposta ao "GCR" para cada tipo de cartão ^[56]. ▪ Incluído parâmetro no comando "DEX" para com opções para o aprimoramento da apresentação das mensagens ^{[62][63]}. ▪ O parâmetro T1_MOBCVM da Tabela de AID passa a ser ignorado pelo pinpad, ficando mandatório o suporte à verificação de portador pelo dispositivo móvel ^{[64][65][66][67][68][69]}. ▪ Incluído suporte a APDU estendido no comando "CHP" ^[70]. ▪ Excluídos todos os comandos genéricos "GEN" ^[71]. ▪ Incluída entrada alfanumérica no comando "GCD" ^{[72][73][74]}. ▪ Outros ajustes e explicações sem mudança funcional ^{[20][24][27][38][39][40][41][42][43][44][49][55][58][59][60][61][75]}. |
| 2.12 | 11/abr/19 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ Comando "LMF" sempre retorna os nomes em maiúsculas ^{[01][02]}. |

| Versão | Data | Autor | Comentários |
|--------|-----------|-------------|---|
| | | | <ul style="list-style-type: none"> ▪ Corrigida informação de mínimo de AIDs candidatos nos comandos “GCR” e “GCX” ^{[03][04]}. ▪ Excluído parâmetro PP_ENCKEY ^[05]. ▪ Corrigido “timeout” de PIN no comando “CHP” ^[06]. ▪ Corrigido valor do <i>Contactless Application/Kernel Capabilities</i> do Pure Contactless ^[07]. ▪ Visa payWave: TTQ byte 3 bit 7 deve estar sempre ativo ^[08]. ▪ Incluída seção explicativa 6.9.5.3 ^[09]. ▪ Reforçado uso de “<i>partial match</i>” no processo de seleção EMV ^{[10][11]}. ▪ Reforçada prioridade dos limites CTLS caso passados em SPE_EMVDATA no comando “GCX” ^[12]. ▪ Incluído parâmetro SPE_TRNCURR em “GOX” ^[13]. ▪ Melhor detalhamento do <i>Enhanced Contactless Reader Capabilities</i> do CTLS Amex ^[14]. ▪ Incluída seção explicativa referente ao gerenciamento do display do pinpad ^[15]. ▪ O parâmetro SPE_TRNCURR somente é usado para ICC, sendo ignorado para CTLS ^{[16][17][18]}. |
| 2.11 | 14/nov/18 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ PP_SERNUM e PP_PARTNBR aumentados para 32 bytes ^[01]. ▪ Criado retorno ↵ ST_CARDBLOCKED ^{[02][03][04]}. ▪ Timeout do comando “CHP” reduzido para 1 minuto ^[05]. ▪ Incluído processamento de cartão Pure Contactless ^{[06][07]}. ▪ Incluído parâmetro SPE_TRNTYPE em “GOX” ^{[08][09]}. ▪ Excluídos métodos para carga remota de chaves e os comandos “IKS” e “IKE”. ▪ TAB_RECIDX passa de “N2” para “A2” ^{[10][11][12][13][14]}. ▪ Excluída configuração CTLS “somente magstripe” em T1_CTLSMODE ^{[15][16]}. ▪ Alterado valor de SPE_PANMASK ^[17]. |
| 2.10 | 04/abr/18 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ Incorporação das Atualizações de Especificação SU001 ^[01], SU002 ^{[02][03]}, SU003 ^{[04][05][06][07]}, SU004 ^{[08][09]} e SU005 ^[10]. ▪ Inclusão das mensagens de 000Dh a 0035h no comando “GCD” ^[11]. ▪ Incluído suporte de GIF para comandos multimídia ^{[12][13]}. ▪ Excluído o modo de criptografia DUKPT:TDES:DAT#1 ^{[14][15]} e, conseqüentemente, o comando “GEN/03/03” ^[16]. ▪ Retirado <u>completamente</u> o suporte às chaves do tipo “DES” ^{[17][18][19][20][21][22][23][24][25][26][27][28]}. ▪ Quantidade mínima posições para chaves de criptografia aumentada de 20 para 32 ^[29]. ▪ Incluídos parâmetros para “código da moeda” e “tipo de |

| Versão | Data | Autor | Comentários |
|--------|-----------|-------------|--|
| | | | <p>transação” em “GCX” [30][31].</p> <ul style="list-style-type: none"> ▪ Tratamento de T1_CTLSTRNLIM passa a ser “igual ou superior” [34][35][36][38]. ▪ Incluídos novos códigos de retorno ↪ST_CTLSEXTCVM e ↪ST_CTLSIFCHG [40][41][42][43][44][45][46]. ▪ Evoluídas especificações de CTLS para as mais recentes das bandeiras, incluindo Discover D-PAS [47]. ▪ Revisão completa do processamento de CTLS nos comandos “GCR” e “GCX” para adequação às novas especificações, bem como inclusão do Discover D-PAS [48][49][50]. ▪ Comando “GCX” apresenta o valor na tela de espera do cartão [51][52]. ▪ Comando “GED” pode ser usado para cartões CTLS emulando tarja [53][54]. ▪ O <i>Terminal Risk Management</i> deixa de ser opcional no processamento de ICC [55][56][57][58]. ▪ Por restrição do PCI, todas as telas de captura de PIN passam a ter o tempo de espera “default” de 1 minuto [59][60][61][62]. ▪ Revisado processamento de <i>Issuer Scripts</i> para ICC cuja transação foi finalizada <i>offline</i> em “GOC” ou “GOX” [63][65]. ▪ Revisadas regras de resolução de conflitos de AIDs [66]. ▪ Incluído suporte à verificação de portador em dispositivo móvel CTLS [40][67]. ▪ Permite o processamento de <i>Issuer Scripts</i> para CTLS no comando “FCX” [70][71][72][73]. ▪ TAB_RECIDX passa de “N2” para “B2”, de forma a comportar mais de 99 registros por rede. ▪ Incluído método para carga remota de chaves e os comandos “IKS” e “IKE”. ▪ Kernel EMV deve suportar até 128 aplicações candidatas [81]. ▪ Incluída possibilidade de mascaramento do PAN nas respostas dos comandos “CEX” e “GCX” [82][83][84]. ▪ Incluída possibilidade de criptografia “End-to-End” entre pinpad e SPE usando chave de sessão [85][86]. ▪ Incluído comando “GEN/03/02” [87]. ▪ Criadas <i>tags</i> proprietárias para objetos que não possuíam referência nas Tabelas de AID [88]. ▪ Incluídos PP_DPCTLSVER e PP_TLRMEM em “GIX” [89]. ▪ Explicações e melhorias gerais ao longo do documento (sem mudança funcional). |
| 2.03 | 23/nov/15 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ Alteração no formato do retorno PP_TRACK1 do comando “GTK” (referente à quantidade de dígitos em aberto). ▪ Inclusão de esclarecimentos quanto ao “<i>Offline Data Authentication</i>” no processamento de CTLS (seções 6.8.1.3 e |

| Versão | Data | Autor | Comentários |
|---------------|-------------|--------------|---|
| | | | <p>6.9.3.2).</p> <ul style="list-style-type: none"> ▪ Definidas telas de entrada de PIN para quando o valor é zero (seção 6.8.6.1). ▪ Alteração nas regras de aglutinação na seção 6.9.5.2. ▪ O comando “OPN” (clássico) passou a ser opcional para o funcionamento do pinpad. |
| 2.02 | 01/jul/15 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ Recolocadas as referências às chaves tipo “DES”. ▪ Pequenas correções nas seções 6.3.4.1 e 6.5.11. |
| 2.01 | 05/mai/15 | WFM (SETIS) | <ul style="list-style-type: none"> ▪ Retiradas todas as referências às chaves tipo “DES”, consideradas obsoletas. ▪ Seção 5.4.1 (“Trilhas incompletas”) reescrita para melhor compreensão. ▪ Incluído comando “GEN/03/03”. ▪ Diversas melhorias e correções pontuais ao longo do documento. |
| 2.00 | 25/jul/14 | WFM (SETIS) | Primeira versão “oficial”. |

Índice

| | |
|---|-----------|
| 1. Introdução | 15 |
| 1.1. Público Alvo | 16 |
| 1.2. Versionamento | 16 |
| 1.3. Formatos usados no documento | 16 |
| 2. Protocolo de Comunicação | 18 |
| 2.1. Nível Físico | 18 |
| 2.2. Nível de Enlace | 19 |
| 2.2.1. Formato do pacote | 19 |
| 2.2.2. Fluxo de comunicação | 20 |
| 2.2.2.1. Envio de comando pelo SPE | 21 |
| 2.2.2.2. Devolução de resposta pelo pinpad | 22 |
| 2.2.2.3. Cancelamento de comando "blocante" | 23 |
| 2.2.3. Fluxos de Processamento no SPE | 24 |
| 2.3. Nível de Aplicação | 28 |
| 2.3.1. Formato do Comando | 28 |
| 2.3.2. Formato da Resposta | 28 |
| 2.3.3. Mensagens de notificação | 29 |
| 2.3.4. Situações de exceção | 29 |
| 3. Comandos | 31 |
| 3.1. Informações Preliminares | 31 |
| 3.1.1. Códigos de retorno | 31 |
| 3.1.2. Comandos obsoletos | 33 |
| 3.1.3. Comandos Abecs | 35 |
| 3.1.3.1. Formato dos comandos | 35 |
| 3.1.3.2. Formato das respostas | 40 |
| 3.2. Comandos de controle | 48 |
| 3.2.1. Comando "OPN" (clássico) | 49 |
| 3.2.2. Comando "OPN" (seguro) | 50 |
| 3.2.3. Comando "GIN" | 54 |
| 3.2.4. Comando "GIX" | 57 |
| 3.2.5. Comando "DWK" | 61 |
| 3.2.6. Comando "CLO" | 64 |
| 3.2.7. Comando "CLX" | 65 |
| 3.3. Comandos básicos | 66 |
| 3.3.1. Comando "CEX" | 67 |
| 3.3.2. Comando "CHP" | 69 |
| 3.3.3. Comando "CKE" | 72 |
| 3.3.4. Comando "DEX" | 75 |
| 3.3.5. Comando "DSP" | 77 |
| 3.3.6. Comando "EBX" | 78 |
| 3.3.7. Comando "ENB" | 80 |
| 3.3.8. Comando "GCD" | 82 |
| 3.3.9. Comando "GDU" | 85 |
| 3.3.10. Comando "GKY" | 86 |
| 3.3.11. Comando "GPN" | 87 |

| | | |
|-------------------|--|----------------|
| 3.3.12. | Comando "GTK" | 89 |
| 3.3.13. | Comando "MNU" | 94 |
| 3.3.14. | Comando "RMC" | 96 |
| 3.4. | Comandos multimídia..... | 97 |
| 3.4.1. | Comando "MLI" | 98 |
| 3.4.2. | Comando "MLR" | 99 |
| 3.4.3. | Comando "MLE" | 104 |
| 3.4.4. | Comando "LMF" | 105 |
| 3.4.5. | Comando "DMF" | 106 |
| 3.4.6. | Comando "DSI" | 107 |
| 3.5. | Comandos para manutenção de Tabelas EMV..... | 108 |
| 3.5.1. | Comando "GTS" | 109 |
| 3.5.2. | Comando "TLI" | 111 |
| 3.5.3. | Comando "TLR" | 112 |
| 3.5.4. | Comando "TLE" | 115 |
| 3.6. | Comandos de processamento de cartão (obsoletos)..... | 116 |
| 3.6.1. | Comando "GCR" | 117 |
| 3.6.2. | Comando "CNG" | 124 |
| 3.6.3. | Comando "GOC" | 126 |
| 3.6.4. | Comando "FNC" | 130 |
| 3.6.5. | Fluxo de operação | 132 |
| 3.7. | Comandos Abecs de processamento de cartão | 133 |
| 3.7.1. | Comando "GCX" | 134 |
| 3.7.2. | Comando "GED" | 141 |
| 3.7.3. | Comando "GOX" | 142 |
| 3.7.4. | Comando "FCX" | 146 |
| 3.7.5. | Fluxo de operação | 149 |
| 3.8. | Comandos genéricos | 150 |
| 3.8.1. | Comando "GEN/02/K3" | 151 |
| 3.8.2. | Comando "GEN/04/01" | 153 |
| 3.8.3. | Comando "GEN/04/02" | 155 |
| 3.8.4. | Comando "GEN/04/03" | 157 |
| 3.8.5. | Comando "GEN/04/04" | 159 |
| 3.8.6. | Comando "GEN/03/02" | 160 |
| 4. | Gerenciamento de Tabelas EMV | 161 |
| 4.1. | Tipos de Tabela..... | 162 |
| 4.1.1. | Tabelas de AID | 162 |
| 4.1.2. | Tabelas de CAPK | 166 |
| 4.1.3. | Tabelas de Certificados Revogados | 168 |
| 4.2. | Versão de Tabelas..... | 169 |
| 4.2.1. | Gerenciamento unificado | 169 |
| 4.2.2. | Gerenciamento apartado | 170 |
| 5. | Segurança | 171 |
| 5.1. | Mapeamento de chaves | 172 |
| 5.1.1. | Criptografia DUKPT:TDES..... | 173 |
| 5.2. | Comunicação Segura | 174 |
| 5.2.1. | Estabelecimento | 174 |
| 5.2.2. | Troca de pacotes | 175 |
| 5.2.2.1. | Envio de pacotes criptografados | 175 |
| 5.2.2.2. | Recepção de pacotes criptografados..... | 176 |

| | | |
|-----------|---|------------|
| 5.2.2.3. | Finalização | 178 |
| 5.3. | PAN Criptografado | 179 |
| 5.3.1. | Codificação do PAN | 180 |
| 5.3.2. | Decodificação das trilhas pelo SPE | 183 |
| 5.3.3. | Criptograma RSA..... | 184 |
| 5.4. | Criptografia “End-to-End” | 185 |
| 5.4.1. | Trilhas incompletas e mascaramento..... | 185 |
| 5.4.2. | Criptografia de trilhas | 186 |
| 5.4.2.1. | Trilha 1..... | 187 |
| 5.4.2.2. | PAN e trilhas 2/3..... | 187 |
| 6. | Funcionamento Interno do Pinpad..... | 189 |
| 6.1. | Arquitetura de <i>software</i> | 190 |
| 6.1.1. | Estrutura de aplicações | 190 |
| 6.1.2. | Capacidades mínimas requeridas | 190 |
| 6.1.3. | Protocolos adicionais..... | 191 |
| 6.2. | Protocolo de comunicação | 192 |
| 6.2.1. | Nível de Enlace | 192 |
| 6.2.2. | Nível de Aplicação | 193 |
| 6.3. | Segurança | 194 |
| 6.3.1. | Mapeamento de chaves | 194 |
| 6.3.2. | Comunicação Segura | 194 |
| 6.3.3. | PAN Criptografado..... | 195 |
| 6.3.4. | Criptografia “End-to-End” | 195 |
| 6.3.4.1. | Trilhas Incompletas | 195 |
| 6.3.4.2. | Criptografia DUKPT..... | 195 |
| 6.3.5. | Proteção contra ataques de <i>replay</i> em ICC | 196 |
| 6.4. | Processamento dos comandos de controle | 198 |
| 6.4.1. | Comando “OPN” | 198 |
| 6.4.2. | Comando “GIN” | 199 |
| 6.4.3. | Comando “GIX” | 199 |
| 6.4.4. | Comando “DWK” | 199 |
| 6.4.5. | Comandos “CLO” e “CLX” | 200 |
| 6.5. | Processamento dos comandos básicos | 201 |
| 6.5.1. | Comando “CEX” | 201 |
| 6.5.2. | Comando “CHP” | 202 |
| 6.5.2.1. | Desligar o cartão (CHP_OPER = “0”) | 202 |
| 6.5.2.2. | Ligar o cartão (CHP_OPER = “1”) | 202 |
| 6.5.2.3. | Troca de comando (CHP_OPER = “2”) | 203 |
| 6.5.2.4. | Verificação de PIN (CHP_OPER = “3”)..... | 204 |
| 6.5.3. | Comando “CKE” | 204 |
| 6.5.4. | Comando “DEX” | 205 |
| 6.5.5. | Comando “DSP” | 206 |
| 6.5.6. | Comando “EBX” | 206 |
| 6.5.7. | Comando “ENB” | 207 |
| 6.5.8. | Comando “GCD” | 207 |
| 6.5.9. | Comando “GDU” | 208 |
| 6.5.10. | Comando “GKY” | 208 |
| 6.5.11. | Comando “GPN” | 208 |
| 6.5.12. | Comando “GTK” | 209 |
| 6.5.13. | Comando “MNU” | 210 |
| 6.5.14. | Comando “RMC” | 211 |

| | | |
|----------|---|-----|
| 6.6. | Processamento dos comandos multimídia | 212 |
| 6.6.1. | Comando "MLI" | 212 |
| 6.6.2. | Comando "MLR" | 212 |
| 6.6.3. | Comando "MLE" | 213 |
| 6.6.4. | Comando "LMF" | 213 |
| 6.6.5. | Comando "DMF" | 213 |
| 6.6.6. | Comando "DSI" | 214 |
| 6.7. | Processamento dos comandos para manutenção de tabelas..... | 215 |
| 6.7.1. | Comando "GTS" | 215 |
| 6.7.2. | Comando "TLI" | 215 |
| 6.7.3. | Comando "TLR" | 216 |
| 6.7.4. | Comando "TLE" | 216 |
| 6.8. | Processamento dos comandos de cartão (obsoletos)..... | 217 |
| 6.8.1. | Comando "GCR" | 218 |
| 6.8.1.1. | Cartão magnético | 219 |
| 6.8.1.2. | Cartão com <i>chip</i> de contato (ICC EMV) | 221 |
| 6.8.1.3. | Cartão com <i>chip</i> sem contato (CTLS)..... | 224 |
| 6.8.2. | Comando "GCR" (vazio)..... | 235 |
| 6.8.3. | Comando "CNG" | 235 |
| 6.8.4. | Comando "GOC" | 236 |
| 6.8.4.1. | Cartão com <i>chip</i> de contato (ICC EMV) | 236 |
| 6.8.4.2. | Cartão com <i>chip</i> sem contato (CTLS EMV)..... | 240 |
| 6.8.5. | Comando "FNC" | 242 |
| 6.8.5.1. | ICC EMV - encerrada <i>offline</i> | 242 |
| 6.8.5.2. | ICC EMV - impossibilidade de conexão <i>online</i> | 242 |
| 6.8.5.3. | ICC EMV - autorização <i>online</i> bem-sucedida | 244 |
| 6.8.5.4. | Cartão com <i>chip</i> sem contato (CTLS EMV)..... | 245 |
| 6.8.6. | Regras gerais..... | 247 |
| 6.8.6.1. | Telas de captura de PIN | 247 |
| 6.8.6.2. | Valores da Transação | 249 |
| 6.8.6.3. | Dados protegidos | 249 |
| 6.8.6.4. | Objetos do cartão..... | 249 |
| 6.9. | Processamento dos comandos Abecs de cartão | 251 |
| 6.9.1. | Comando "GCX" | 252 |
| 6.9.1.1. | Cartão magnético | 254 |
| 6.9.1.2. | Cartão com <i>chip</i> de contato (ICC EMV) | 255 |
| 6.9.1.3. | Cartão com <i>chip</i> sem contato (CTLS)..... | 259 |
| 6.9.2. | Comando "GED" | 265 |
| 6.9.3. | Comando "GOX" | 266 |
| 6.9.3.1. | Cartão com <i>chip</i> de contato (ICC EMV) | 266 |
| 6.9.3.2. | Cartão com <i>chip</i> sem contato (CTLS EMV)..... | 271 |
| 6.9.4. | Comando "FCX" | 273 |
| 6.9.4.1. | ICC EMV - encerrada <i>offline</i> | 273 |
| 6.9.4.2. | ICC EMV - impossibilidade de conexão <i>online</i> | 274 |
| 6.9.4.3. | ICC EMV - autorização <i>online</i> bem-sucedida | 275 |
| 6.9.4.4. | Cartão com <i>chip</i> sem contato (CTLS EMV)..... | 277 |
| 6.9.5. | Regras gerais..... | 279 |
| 6.9.5.1. | Telas de captura de PIN | 279 |
| 6.9.5.2. | Tabelas de AID (resolução de conflitos) | 279 |
| 6.9.5.3. | Valor da transação, dados protegidos, objetos do cartão | 281 |
| 6.10. | Processamento dos comandos genéricos | 282 |
| 6.10.1. | Comando "GEN/02/K3" | 282 |
| 6.10.2. | Comando "GEN/03/02" | 282 |

| | | |
|-----------|--|------------|
| 6.10.3. | Comando "GEN/04/01" | 282 |
| 6.10.4. | Comando "GEN/04/02" | 283 |
| 6.10.5. | Comando "GEN/04/03" | 283 |
| 6.10.6. | Comando "GEN/04/04" | 283 |
| 6.11. | Teclas especiais..... | 284 |
| 7. | Informações Complementares | 285 |
| 7.1. | Codificação TLV..... | 286 |
| 7.1.1. | Codificação do campo "tag" (T)..... | 286 |
| 7.1.2. | Codificação do campo "length" (L)..... | 286 |
| 7.2. | Cálculo de CRC..... | 287 |
| 7.3. | Display do pinpad | 288 |
| 7.3.1. | Uso pelos comandos..... | 288 |
| 7.3.2. | Tabela de Caracteres | 289 |
| 7.4. | Diferenças em relação à Biblioteca Compartilhada | 291 |

Referências

-  **BibComp** Biblioteca Compartilhada para Pinpad - Especificação Detalhada - Versão 1.08a (15/Abr/2013).
-  **EMV#1** EMV - *Integrated Circuit Card Specifications for Payment Systems - Book 1 - Application Independent ICC to Terminal Interface Requirements - [Version 4.4 - October 2022](#)*.
-  **EMV#2** EMV - *Integrated Circuit Card Specifications for Payment Systems - Book 2 - Security and Key Management - [Version 4.4 - October 2022](#)*.
-  **EMV#3** EMV - *Integrated Circuit Card Specifications for Payment Systems - Book 3 - Application Specification - [Version 4.4 - October 2022](#)*.
-  **EMV#4** EMV - *Integrated Circuit Card Specifications for Payment Systems - Book 4 - Cardholder, Attendant, and Acquirer Interface Requirements - [Version 4.4 - October 2022](#)*.
-  **EMV#CtlsA** EMV - *Contactless Specifications for Payment Systems - Book A - Architecture and General Requirements - [Version 2.11 - June 2023](#)*.
-  **EMV#CtlsB** EMV - *Contactless Specifications for Payment Systems - Book B - Entry Point Specification - [Version 2.11 - June 2023](#)*.
-  **PPMChip** ~~Master Card PayPass - M/Chip Reader Card Application Interface Specification V3.0.2 - May 2013; e~~
EMV - *Contactless Specifications for Payment Systems - Book C-2 - Kernel 2 Specification - [Version 2.11 - June 2023](#)*.
-  **VCPS** ~~VCPS - Visa Contactless Payment Specification - Version 2.2 - January 2016; e~~
EMV - *Contactless Specifications for Payment Systems - Book C-3 - Kernel 3 Specification - [Version 2.11 - June 2023](#)*.
-  **ExpPay** ~~Expresspay - American Express - Terminal Specification - Version 3.1 - April 2015; e~~
EMV - *Contactless Specifications for Payment Systems - Book C-4 - Kernel 4 Specification - [Version 2.11 - June 2023](#)*.
-  **D-PAS** ~~D-PAS - Discover Contactless - Terminal Application Specification - Version 1.1 - March 2015; e~~
EMV - *Contactless Specifications for Payment Systems - Book C-6 - Kernel 6 Specification - [Version 2.11 - June 2023](#)*.
-  ~~**Pure** Gemalto PURE - Contactless reader Specifications for PURE Dual Interface cards and Mobile PURE - Version 2.1.8 - August 2016.~~
-  **QckPass** EMV - *Contactless Specifications for Payment Systems - Book C-7 - Kernel 7 Specification - [Version 2.11 - June 2023](#)*.

Glossário

- Abecs** Ou “Associação Brasileira das Empresas de Cartões de Crédito e Serviços”.
- AES** Ou “*Advanced Encryption Standard*”, também conhecido como “Rijndael”, trata-se de um algoritmo de criptografia simétrica definido pela norma FIPS 197 ou ISO/IEC 18033-3. Este algoritmo prevê chaves de diferentes tamanhos, porém esta especificação utiliza especificamente o AES-128 (chave de 16 bytes).

Sendo um algoritmo simétrico, o AES possui uma função reversa, nesta especificação denotada como **AES⁻¹**.

| | |
|----------------------|---|
| AID | Ou “ <i>Application Identifier</i> ”, trata-se de um dado de 5 a 16 bytes que identifica uma aplicação de pagamento em um cartão EMV (Ex: Visa Crédito = A0000000031010h). |
| APDU | Ou “ <i>Application Protocol Data Unit</i> ”, trata-se do formato padrão de uma mensagem de comando enviada pelo pinpad ao ICC (C-APDU) ou de uma mensagem de resposta devolvida ao pinpad pelo ICC (R-APDU). |
| Bandeira | Instituição que define regras e proporciona a interoperabilidade de emissão e aceitação de cartões de pagamento (ex: VISA, Mastercard, etc.). |
| Bypass | Situação na qual o portador do cartão se recusa a digitar o PIN, pressionando a tecla [OK/ENTRA] do pinpad com o campo de entrada vazio. |
| CBC | Ou “ <i>Cipher-block Chaining</i> ”, método de criptografia de bloco de dados. |
| Comando | Instrução enviada do SPE ao pinpad para que este o execute, devolvendo uma resposta. |
| CRC | Ou “ <i>Cyclic Redundancy Check</i> ”, código de validação de informações e detecção de erro (ver seção 7.2). |
| Credenciadora | Empresa que efetua a captura e processamento de transações com cartão de pagamento. |
| Criptograma | Bloco de dados criptografados usando-se uma chave simétrica (DES, TDES, AES) ou uma chave pública assimétrica (RSA). |
| CTLS | Não se trata de uma sigla. Esta definição foi criada nesta especificação para referenciar cartão com <i>chip sem contato</i> (“ <i>contactless</i> ”), de forma a diferenciá-lo do ICC . |
| DES | Ou “ <i>Data Encryption Standard</i> ”, algoritmo de criptografia de chave simétrica definido pela norma FIPS-46-3. Sendo um algoritmo simétrico, o DES possui uma função reversa, nesta especificação denotada como DES⁻¹ . |
| Display | Dispositivo para apresentação de texto e imagens no pinpad, normalmente uma tela de cristal líquido. |
| DUKPT | Ou “ <i>Derived Unique Key Per Transaction</i> ”, método de criptografia de PIN (ou dados quaisquer) definido pela norma ANSI X9.24:2009. |
| ECB | Ou “ <i>Electronic Codebook</i> ”, método de criptografia de bloco de dados. |
| EMV | Norma para processamento de cartões de pagamento ICC , definida em EMV#1 , EMV#2 , EMV#3 e EMV#4 . |
| Em claro | Uma informação ou dado não criptografado é dito nesta especificação como “em claro”. |
| Emissor | Entidade, normalmente um banco, que emite cartões para uso nos pinpads, sejam magnéticos, ICC ou CTLS. |
| Fallback | Processo de contingência através do qual um ICC é aceito pelo SPE através de sua tarja magnética, normalmente devido a um problema técnico com o <i>chip</i> . |
| ICC | Ou “ <i>Integrated Circuit Card</i> ”, para esta especificação refere-se exclusivamente a cartão com <i>chip de contato</i> , de acordo com a norma ISO-7816. |
| Kernel EMV | Núcleo de <i>software</i> certificado “ <i>EMV Type Approval Level 2</i> ” que é responsável pelo processamento de cartões EMV (ICC ou CTLS) no pinpad. |

| | |
|--|---|
| K_{MOD}/K_{PUB}/K_{PRV} | Chave RSA gerenciada pelo SPE, utilizada nos modos de “Comunicação Segura” (seção 5.2) e “PAN Criptografado” (seção 5.3), composta por um “módulo” (K_{MOD}), um “expoente público” (K_{PUB}) e um “expoente privado” (K_{PRV}). |
| K_{SEC} | Chave AES criada pelo pinpad no modo de “Comunicação Segura” (seção 5.2). |
| K_{RAND} | Chave TDES aleatória usada para codificar as trilhas na “Criptografia End-to-End” (seção 5.4). |
| KSN | Ou “ <i>Key Serial Number</i> ”, trata-se do número de série de uma chave usada na criptografia DUKPT . |
| MK | Ou “ <i>Master Key</i> ”, chave de criptografia TDES inserida no pinpad (nesta especificação referenciada como MK:TDES). |
| MK/WK | Método de criptografia de PIN (ou dados quaisquer) definido pela norma ANSI X9.8, que utiliza uma MK e uma “ <i>Working Key</i> ” fornecida externamente. |
| Nibble | Equivalente a meio byte, ou seja, um conjunto de 4 bits (representa valores 0h a Fh). |
| PAN | Ou “ <i>Primary Account Number</i> ”, ou seja, o número de um cartão de pagamento. |
| PCI | Ou “ <i>Payment Card Industry Security Standards Council</i> ”, ou seja, conselho normativo que define regras de segurança para sistemas de pagamento com cartão. |
| PIN | Ou “ <i>Personal Identification Number</i> ”, ou seja, a senha do portador de um cartão. |
| Pinpad | Formalmente “ <i>PIN-pad</i> ”, trata-se de um equipamento seguro (“ <i>tamper proof</i> ”) que preserva chaves de criptografia (MK/WK ou DUKPT) e contempla interfaces de teclado, <i>display</i> , cartão magnético, ICC, SAM, CTLS e comunicação serial (RS232, USB, Bluetooth, etc.). |
| Portador | Também mencionado como “Portador de Cartão” (“ <i>cardholder</i> ” em inglês), refere-se à pessoa que utiliza um cartão para efetuar uma transação de pagamento. |
| Protocolo | Também mencionado como “Protocolo de Comunicação”, trata-se do mecanismo bidirecional de transferência de dados entre o SPE e o pinpad, de forma que o SPE possa enviar os comandos. |
| RSA | Ou “ <i>Rivest, Shamir & Adleman</i> ”, algoritmo de criptografia assimétrica definido pelo padrão PKCS #1 (RFC 3447). Uma chave de criptografia RSA é composta de “módulo”, “expoente público” e “expoente privado”. |
| RUF | Ou “Reservado para Uso Futuro”. |
| SAM | Ou “ <i>Secure Application Module</i> ”, refere-se a um cartão com <i>chip</i> (formato “2FF”) embutido no pinpad. |
| SPE | Ou “Sistema de Pagamento Eletrônico”, ou seja, o sistema que utiliza o pinpad, podendo ser, por exemplo, um “ <i>checkout</i> ” de pagamento ou uma máquina de autoatendimento. |
| Tag | Ver “ TLV ”. |
| TDES | Ou “ <i>Triple-DES</i> ”, algoritmo de criptografia de chave simétrica definido pela norma NIST SP 800-57 e SP 800-78-3 (2TDEA - <i>keying option 2</i>). Sendo um algoritmo simétrico, o TDES possui uma função reversa, nesta especificação denotada como TDES⁻¹ . |
| TLV | Ou “ <i>Tag, Length and Value</i> ”, trata-se de um método de codificação de dados usado pela norma EMV (ver seção 7.1). |
| Trilha | Um dos três blocos de dados gravados em um cartão magnético, referidos como Trilha 1, Trilha 2 e Trilha 3. Esses blocos de dados geralmente contêm o PAN, a data de |

validade e outras informações relevantes. O ICC e o CLTS podem conter os mesmos blocos de dados em sua memória.

WK_{PAN} Chave de criptografia ~~DES~~ **TDES** usada para codificação de informações sensíveis nas mensagens de comunicação (principalmente o **PAN**) no método denominado nesta especificação como “PAN Criptografado” (ver **seção 5.3**).

XOR Ou “*Exclusive OR*”, trata-se de uma operação lógica binária também representada pelo símbolo “ \oplus ”.

Observação: Os termos extraídos da norma **EMV** permanecem em inglês neste documento (em *destaque*) de maneira a se evitar a perda de referência e, assim, facilitar sua compreensão.

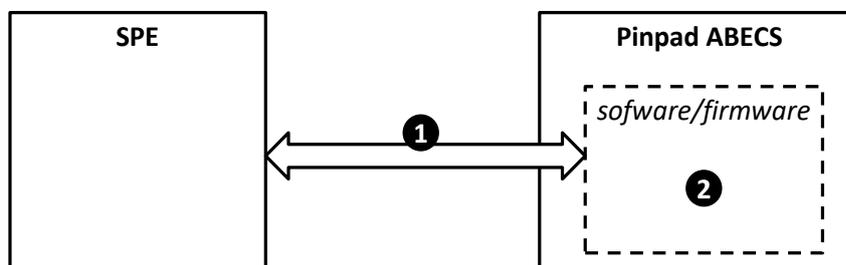
1. Introdução

Este documento destina-se a especificar detalhadamente o “**Pinpad Abecs**”, tendo como objetivo definir um padrão de interoperabilidade para a utilização de dispositivos do tipo pinpad no mercado brasileiro, englobando principalmente as seguintes funcionalidades:

- Captura segura de PIN;
- Leitura de cartão magnético;
- Processamento de cartões com *chip* EMV (com e sem contato);
- Operações básicas de “interface homem-máquina” com o portador do cartão; e
- Identificação e gestão logística do equipamento.

Por “Pinpad Abecs”, entende-se um dispositivo do tipo pinpad, cujo *software/firmware* respeita esta especificação técnica funcional, que não entra no mérito do *hardware* do equipamento.

Esta especificação tem enfoque em dois pontos técnicos principais para garantir a interoperabilidade de um Pinpad Abecs nos diversos SPE:



- ❶ Protocolo de comunicação entre o SPE e o Pinpad Abecs; e
- ❷ Funcionamento interno do pinpad, ou seja, especificação de seu *software/firmware*.

1.1. Público Alvo

Esta especificação destina-se aos seguintes públicos:

- Credenciadoras;
- Desenvolvedores de SPE; e
- Fornecedores de pinpad e seus desenvolvedores de *software/firmware*.

1.2. Versionamento

Esta especificação adota a convenção de versão “**A.BC**”, numérica, sendo:

“**C**” = Dígitos incrementados quando a especificação é alterada apenas para melhorias estruturais ou explicativas, não incorrendo em alterações funcionais.

“**B**” = Dígitos incrementados quando a especificação sofre alterações funcionais no pinpad, mantendo total compatibilidade com os SPE.

“**A**” = Dígitos incrementados quando a especificação sofre alterações funcionais que influenciam ambos os lados: SPE e pinpad.

1.3. Formatos usados no documento

Este documento menciona diversos dados em comandos e tabelas, sendo que estes dados, por suas características, devem respeitar diferentes regras de codificação.

A representação de um formato segue a seguinte regra: “[**Caractere de Formato**][.][**Tamanho**]”

[**Caractere de Formato**] = Letra maiúscula que define o formato.

[.] = Opcional, indica que o dado é de tamanho variável, podendo ter de zero a [**Tamanho**] bytes.

[**Tamanho**] = De um a três dígitos numéricos representando a quantidade de bytes utilizada pela informação.

Exemplos:

- O código “W256” indica uma informação de 256 bytes codificada de acordo com o formato “W”.
- O código “K..99” indica uma informação de tamanho variável (de 0 a 99 bytes) codificada de acordo com o formato “K”.

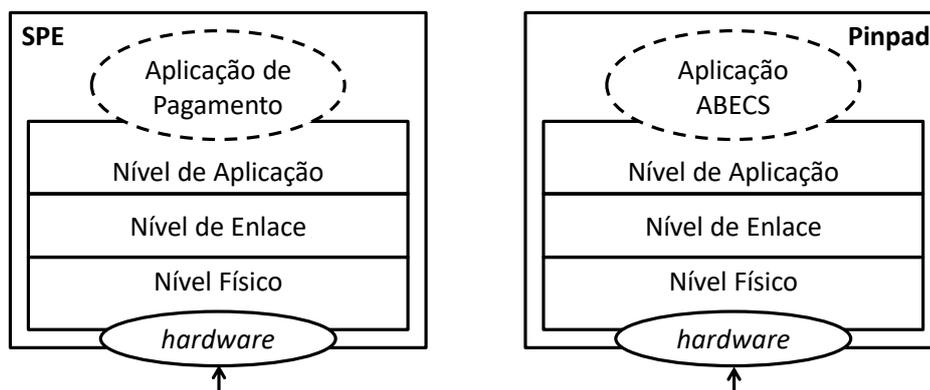
A tabela seguir detalha os formatos adotados neste documento:

| Formato | Descrição |
|----------|--|
| A | Alfanumérico codificado segundo tabela ASCII, podendo conter bytes de 20h (espaço) a 7Eh (~). Quando a informação for menor do que o campo definido, ela deverá ser alinhada à esquerda com espaços (20h) à direita. <u>Exemplo:</u> Se um campo de formato "A6" contém a informação "TEXTO", ele é codificado como: 54h 45h 58h 54h 4Fh 20h. |
| S | Alfanumérico codificado segundo a tabela de caracteres definida na seção 7.3.2 , podendo conter bytes de 20h (espaço) a FFh. Quando a informação for menor do que o campo definido, ela deverá ser alinhada à esquerda com espaços (20h) à direita. <u>Exemplo:</u> Se um campo de formato "S8" contém a informação "Ação", ele é codificado como: 41h E7h E3h 6Fh 20h 20h 20h 20h. |
| N | Numérico decimal codificado segundo a tabela ASCII, podendo conter somente bytes de 30h ("0") a 39h ("9"). Quando a informação for menor do que o campo definido, ela deverá ser alinhada à direita com zeros (30h) à esquerda. <u>Exemplo:</u> Se um campo de formato "N8" contém a informação numérica 1234, ele é codificado como: 30h 30h 30h 30h 31h 32h 33h 34h. |
| H | Numérico <u>hexadecimal</u> codificado segundo a tabela ASCII, podendo conter somente bytes de 30h ("0") a 39h ("9"), 41h ("A") a 46h ("F") e 61h ("a") a 66h ("f"). Quando a informação for menor do que o campo definido, ela deverá ser alinhada à direita com zeros (30h) à esquerda. Cada dois caracteres em hexadecimal representam um byte (valor de 00h a FFh), por isso o [Tamanho] deve ser sempre um número <u>par</u> . <u>Exemplo:</u> Se um campo de formato "H4" contém a informação numérica 3F6Ch, ele é codificado como: 33h 46h 36h 43h. |
| X | Numérico em representação binária, precedida pelo byte <u>mais significativo</u> . Quando a informação for menor do que o campo definido, ela deverá ser alinhada à direita com zeros à esquerda. <u>Exemplo:</u> Se um campo de formato "X3" contém a informação numérica 3000 (0BB8h), ele é codificado como: 00h 0Bh B8h. |
| B | Informação genérica que permite qualquer byte de 00h a FFh. |

▲ IMPORTANTE: Dados de tipo "H.???" são sempre precedidos por um campo numérico contendo a informação do seu tamanho. Entretanto, por razões históricas, este valor é sempre dividido por dois (÷ 2), de forma a representar a quantidade de bytes "originais" que geraram a codificação em hexadecimal.

2. Protocolo de Comunicação

Este capítulo descreve o protocolo de comunicação entre o SPE e o pinpad, considerando-se três níveis:



2.1. Nível Físico

O “nível físico” é a camada inferior do protocolo que garante a transmissão e recepção de bytes de dados entre o SPE e o pinpad.

O Pinpad Abecs considera essencialmente um “nível físico” de comunicação serial, independentemente da tecnologia (RS-232, USB, Bluetooth, etc.), com as seguintes configurações quando relevantes para o meio utilizado:

- Velocidade: 19.200 bps (bits por segundo);
- 8 bits por byte;
- Sem bit de paridade (“*parity*”); e
- 1 bit de parada (“*stop bit*”).

2.2. Nível de Enlace

O Nível de Enlace destina-se a definir o fluxo de comunicação de dados entre o SPE e o pinpad, bem como garantir a integridade das informações trocadas (aqui denominadas “pacotes”).

Para a implementação do Nível de Enlace, os seguintes bytes especiais (caracteres de controle) são utilizados:

| Nome | Valor | Descrição |
|-------|-------|--|
| «EOT» | 04h | Resposta do pinpad ao receber um «CAN». |
| «ACK» | 06h | Enviado do pinpad para o SPE ao receber um pacote válido. |
| «DC3» | 13h | Byte de substituição, para impedir que bytes especiais trafeguem no corpo do pacote. |
| «NAK» | 15h | É devolvido ao lado que enviou um pacote inválido, solicitando sua retransmissão. |
| «SYN» | 16h | Indica o início de um pacote. |
| «ETB» | 17h | Indica o final de um pacote. |
| «CAN» | 18h | Enviado do SPE para o pinpad para cancelar a execução de um comando. |

2.2.1. Formato do pacote

Os pacotes de dados trocados entre as partes, independentemente do sentido (SPE ↔ pinpad) possuem sempre o seguinte formato:

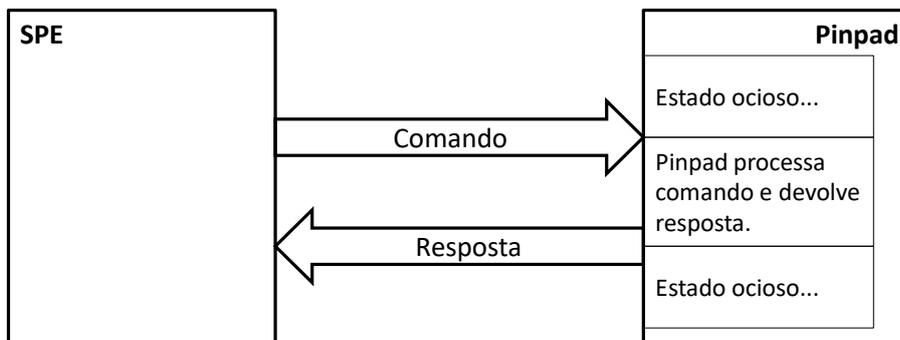
| Nome | Formato | Descrição |
|----------|---------|---|
| PKTSTART | B1 | Byte 16h («SYN») para identificação do início do pacote. |
| PKTDATA | ??? | Conteúdo do pacote, considerando-se a seguinte regra de substituição: <ul style="list-style-type: none"> ▪ O byte 13h («DC3») é substituído pelos bytes 13h («DC3») e 33h; ▪ O byte 16h («SYN») é substituído pelos bytes 13h («DC3») e 36h; e ▪ O byte 17h («ETB») é substituído pelos bytes 13H («DC3») e 37h. O pacote “original” (desconsiderando-se eventuais substituições) pode ter no máximo 2049 bytes . |
| PKTSTOP | B1 | Byte 17h («ETB») para identificação do final do pacote. |
| PKTCRC | X2 | CRC-16 dos dados de PKTDATA e PKTSTOP , calculado sobre os dados “originais”, <u>desconsiderando-se</u> eventuais substituições feitas usando-se o byte «DC3» (ver algoritmo na seção 7.2). |

⚠ Por questão de compatibilidade com a base legada, o SPE só poderá enviar um pacote ao pinpad com **PKTDATA** maior do que **1024 bytes** no caso de um “Comando Abecs” (ver **seção 3.1.3**).

2.2.2. Fluxo de comunicação

O fluxo de comunicação sempre se inicia no SPE, sendo que o pinpad é uma entidade “passiva”, ou seja, nunca envia dados ao SPE a menos que isso seja requisitado.

- Um pacote de dados enviado pelo SPE ao pinpad é chamado “**comando**”; e
- O pacote de dados devolvido pelo pinpad ao SPE é chamado de “**resposta**”.

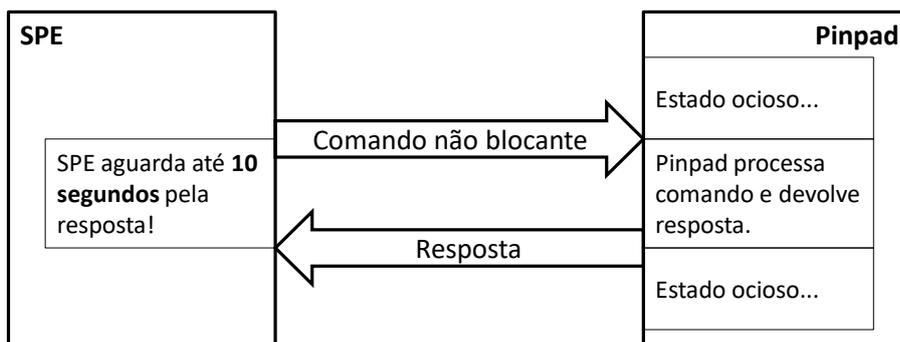


Esta especificação prevê dois tipos de comandos, chamados “**blocantes**” e “**não blocantes**”, conforme detalhamento a seguir. Para saber o tipo de um determinado comando, deve-se consultar sua definição no **Capítulo 3**.

➔ Comandos “não blocantes”

Comandos que não exigem a interação com o portador do cartão são chamados de “**não blocantes**”.

Neste caso, o SPE deve aguardar até 10 segundos pela resposta, retornando erro de “tempo esgotado” caso esta não seja devolvida no tempo esperado.

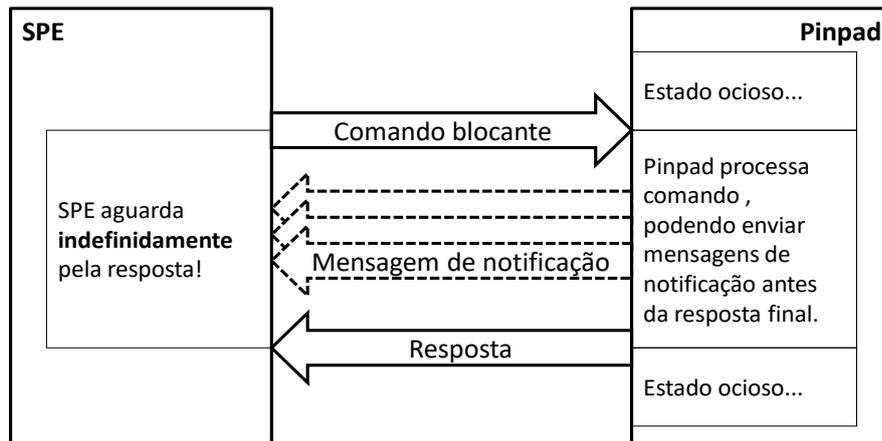


➔ Comandos “blocantes”

Comandos que exigem a interação com o portador do cartão (por exemplo, captura de PIN), fazem com que o pinpad segure o processamento indefinidamente, sendo chamados de “**blocantes**”.

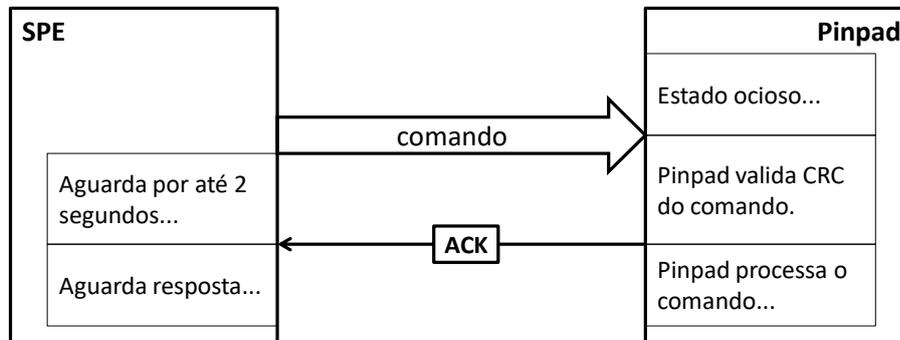
Neste caso, o SPE deve esperar indefinidamente pela resposta, nunca retornando erro de “tempo esgotado”.

Este tipo de comando também permite que o pinpad devolva ao SPE respostas intermediárias denominadas “mensagens de notificação” (ver **seção 2.3.3**).

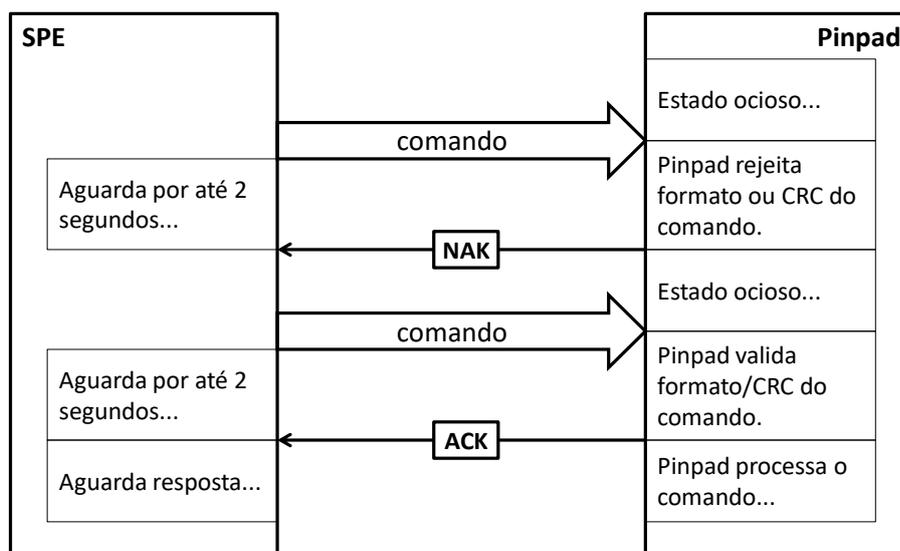


2.2.2.1. Envio de comando pelo SPE

O SPE envia um pacote de comando ao pinpad conforme formato descrito na seção 2.2.1.



Ao receber o comando, o pinpad confere o CRC e envia um «ACK» (06h) se os dados estiverem corretos. Caso os valores não coincidam, ou o formato do pacote não estiver íntegro, o pinpad envia um «NAK» (15h) e descarta o pacote.



O SPE deve aguardar um «ACK» ou um «NAK» durante 2 segundos após o envio do comando. O não recebimento de algum desses bytes aborta a comunicação.

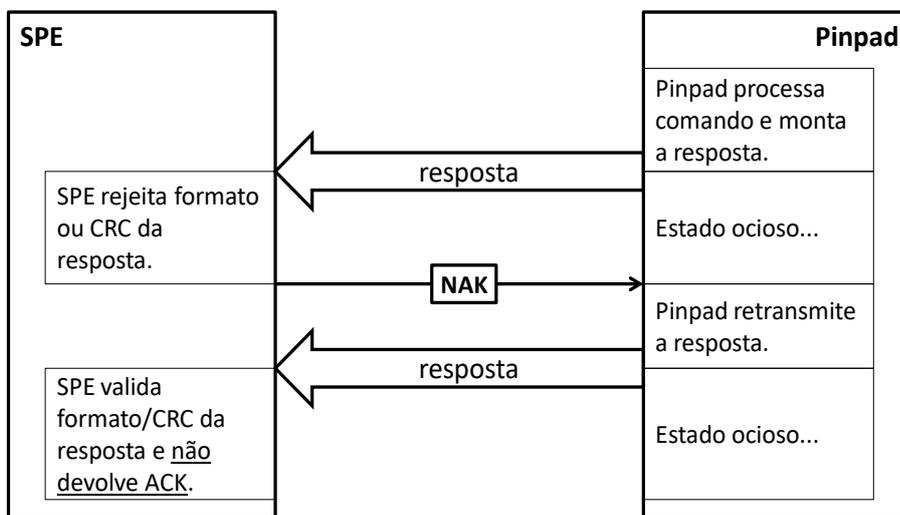
Ao receber um «NAK», o SPE deve retransmitir o comando. O SPE deve tentar o envio do comando até 3 vezes, abortando após o 3º «NAK» recebido.

➔ Exemplos:

| | | |
|--|----------------------|---------|
| SPE envia comando ao pinpad, porém este não recebe o CRC corretamente. | | |
| SPE ⇒ | 16 4F 50 4E 17 00 00 | •OPN••• |
| O pinpad não reconhece o comando como válido e retorna «NAK». | | |
| ⇐ PP | 15 | • |
| SPE reenvia o comando, que agora é recebido com CRC correto. | | |
| SPE ⇒ | 16 4F 50 4E 17 A8 A9 | •OPN••• |
| O pinpad devolve «ACK» acata o comando. | | |
| ⇐ PP | 06 | • |

2.2.2.2. Devolução de resposta pelo pinpad

Ao processar um comando, o pinpad devolve ao SPE um ou mais pacotes de resposta (no caso de mensagens de notificação), conforme formato descrito na seção 2.2.1.



Ao receber uma resposta do pinpad, o SPE deve verificar o CRC do pacote recebido e enviar um «NAK» caso haja erro, voltando a aguardar a resposta. Este processo deve ser repetido até 3 vezes.

Caso o pacote recebido esteja íntegro, nada deverá ser enviado.

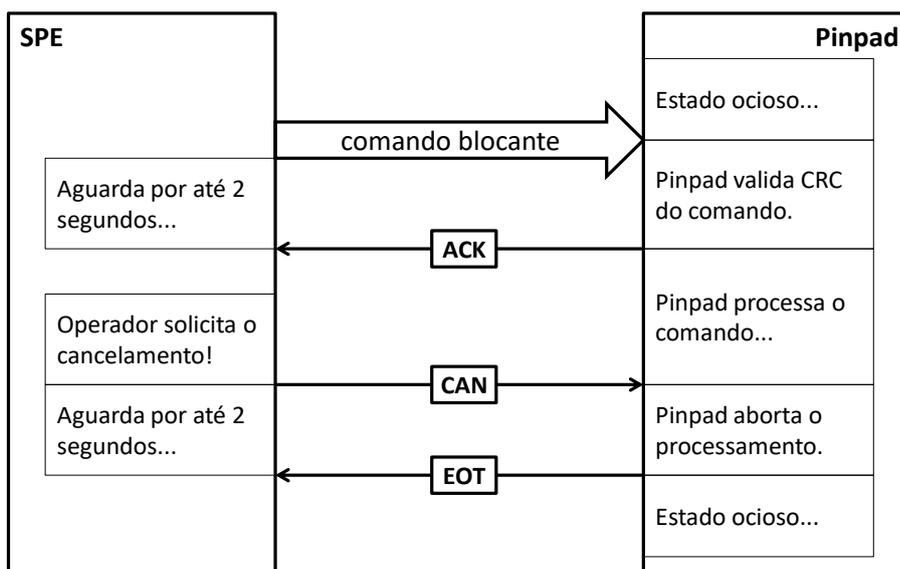
➤ Exemplos:

| | | |
|---|---|--|
| O SPE envia um comando ao pinpad. | | |
| SPE ⇒ | 16 44 53 50 30 33 32 20 20 20 20 4F 50 45 52 41 C7 C3 4F 20 20 20 20 20 20 20 46 49 4E 41 4C 49 5A 41 44 41 20 20 20 17 52 13 | •DSP032•••••OPERA ÇÃO•••••FINALI ZADA•••••R• |
| O pinpad devolve «ACK» e acata o comando. | | |
| ⇐ PP | 06 | • |
| O pinpad devolve a resposta, porém o SPE não recebe um CRC válido. | | |
| ⇐ PP | 16 44 53 50 30 30 30 17 FF FF | •DSP000•ÿÿ |
| O SPE não reconhece a resposta e envia um «NAK», solicitando sua retransmissão. | | |
| SPE ⇒ | 15 | • |
| O pinpad devolve a resposta que agora é recebida com CRC válido. | | |
| ⇐ PP | 16 44 53 50 30 30 30 17 39 63 | •DSP000•9c |
| O SPE acata a resposta. | | |

2.2.2.3. Cancelamento de comando “bloqueante”

No caso dos comandos “bloqueantes”, o SPE deverá esperar pela resposta indefinidamente. Entretanto, este tipo de comando pode ser abortado a qualquer momento pelo SPE através do envio de um byte «CAN».

Ao receber o byte «CAN», o pinpad aborta a operação em curso, devolve um byte «EOT» e volta ao estado ocioso, de forma a aguardar um novo comando. Na verdade, o pinpad sempre responde «EOT» a um «CAN», independentemente do seu estado.



O SPE deve aguardar o «EOT» durante 2 segundos, de modo a obter confirmação do cancelamento. Caso este byte não seja recebido, o SPE deve tentar envio do «CAN» até 3 vezes.

Durante essa espera, o SPE deve ignorar outros bytes que venha a receber, pois, coincidentemente, pode haver uma resposta do pinpad ou uma mensagem de notificação sendo devolvida no momento do cancelamento.

⚠ É importante que o SPE sempre inicie o fluxo de comunicação com o pinpad enviando um «CAN», de forma a abortar qualquer comando bloqueante que eventualmente esteja em processamento.

➔ Exemplos:

O SPE envia um comando bloqueante ao pinpad.

| | | |
|-------|--|------------------------------------|
| SPE ⇒ | 16 47 43 44 30 31 36 00 0C 00 01 3C 00 0E 00 01 0A 00 0B 00 02 00 09 17 C1 42 | •GCD016•••••<••••• ••••••••••ÁB |
|-------|--|------------------------------------|

O pinpad devolve «ACK» e acata o comando.

| | | |
|------|----|---|
| ⇐ PP | 06 | • |
|------|----|---|

Após um tempo de espera, o SPE decide abortar o comando enviando um «CAN».

| | | |
|-------|----|---|
| SPE ⇒ | 18 | • |
|-------|----|---|

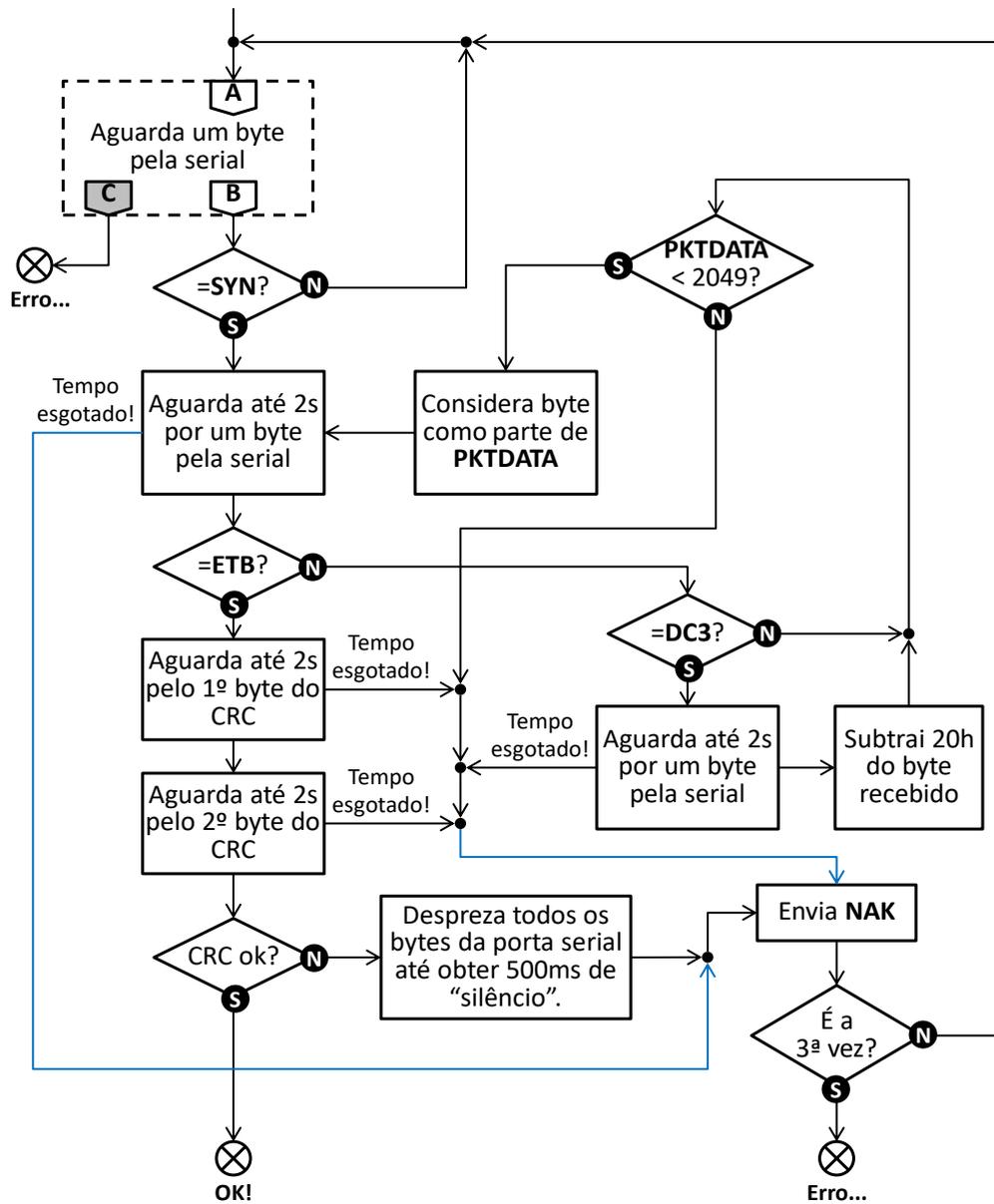
O pinpad aborta a execução imediatamente e devolve «EOT».

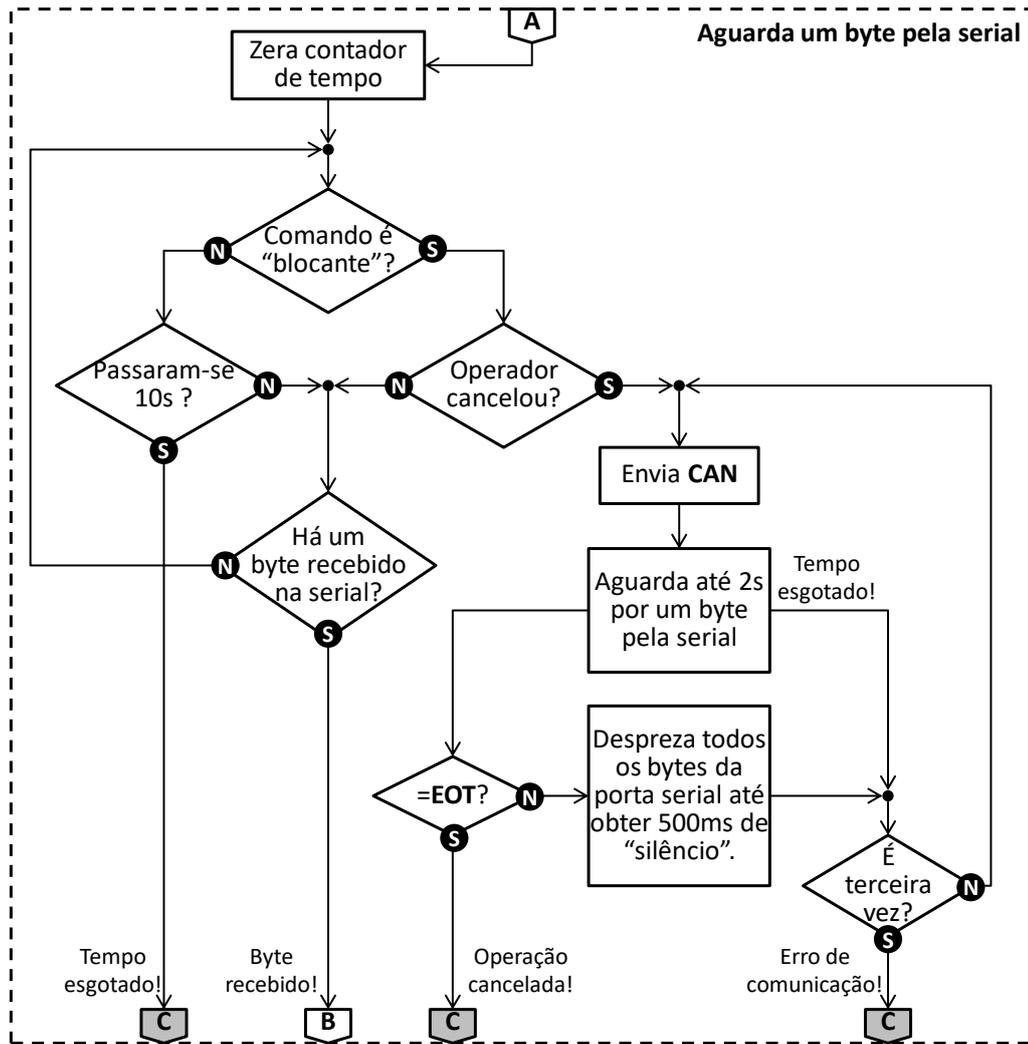
| | | |
|------|----|---|
| ⇐ PP | 04 | • |
|------|----|---|

2.2.3. Fluxos de Processamento no SPE

Esta seção descreve os fluxos internos de processamento no SPE para o correto tratamento do Nível de Enlace.

➔ Recepção de resposta





2.3. Nível de Aplicação

O “Nível de Aplicação” define o formato dos dados que circulam em **PKTDATA** através do Nível de Enlace, sendo que ele depende do sentido do pacote (SPE ↔ pinpad).

⚠ Se **PKTDATA** for iniciado pelo byte «**DC2**» (**12h**), ele está criptografado segundo método de “Comunicação Segura” descrito na **seção 5.2**. Neste caso, os formatos descritos nesta seção se referem ao campo **CLRDATA**.

2.3.1. Formato do Comando

Todos os comandos enviados pelo SPE ao pinpad devem respeitar o formato descrito a seguir.

Um comando pode conter ou não blocos de dados (parâmetros) de até 999 bytes cada, sempre precedidos pela informação de tamanho.

| Id. do Campo | Formato | Descrição |
|-----------------|---------|--|
| CMD_ID | A3 | Código do comando (descritos no Capítulo 3). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (de “000” a “999” bytes) |
| CMD_BLK1 | B..999 | Primeiro bloco de parâmetros do comando |
| CMD_LEN2 | N3 | Tamanho dos dados a seguir (de “000” a “999” bytes) |
| CMD_BLK2 | B..999 | Segundo bloco de parâmetros do comando |
| ... | ... | ... |
| CMD_LENn | N3 | Tamanho dos dados a seguir (de “000” a “999” bytes) |
| CMD_BLKn | B..999 | Último bloco de parâmetros do comando |

2.3.2. Formato da Resposta

As respostas devolvidas pelo pinpad ao SPE devem respeitar os formatos descritos a seguir.

➡ Execução bem-sucedida

Uma resposta à execução bem-sucedida de um comando pode conter (ou não) blocos de dados de até 999 bytes cada, sempre precedidos pela informação de tamanho.

| Id. do Campo | Formato | Descrição |
|-----------------|---------|---|
| RSP_ID | A3 | Código da resposta (igual a CMD_ID) |
| RSP_STAT | N3 | Valor “000”, indicando o sucesso do processamento. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (de “000” a “999” bytes) |

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_BLK1 | B..999 | Primeiro bloco de dados da resposta |
| RSP_LEN2 | N3 | Tamanho dos dados a seguir (de "000" a "999" bytes) |
| RSP_BLK2 | B..999 | Segundo bloco de dados da resposta |
| ... | ... | ... |
| RSP_LENn | N3 | Tamanho dos dados a seguir (de "000" a "999" bytes) |
| RSP_BLKn | B..999 | Último bloco de dados da resposta |

↻ Erro de execução

Em caso de erro na execução de um comando reconhecido, o pinpad sempre devolve a resposta a seguir, de 6 bytes.

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (igual a CMD_ID) |
| RSP_STAT | N3 | Resultado do processamento do comando (\neq "000"), conforme definido na seção 3.1 . |

⚠ Se **RSP_STAT** \neq "000", a resposta nunca deve conter dados!

2.3.3. Mensagens de notificação

Durante a execução de comandos "bloqueantes", o pinpad pode enviar mensagens de notificação ao SPE, para que este as apresente ao operador.

As mensagens de notificação possuem o seguinte formato:

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | "NTM" |
| RSP_STAT | N3 | "000" (sempre sucesso) |
| RSP_LEN1 | N3 | "000" a "032" |
| NTM_MSG | S32 | Mensagem a ser apresentada ao operador do SPE, formatada para poder ser visualizada corretamente em 2 linhas de 16 caracteres. |

2.3.4. Situações de exceção

Caso um comando não seja reconhecido pelo pinpad como válido, ele não pode devolver uma resposta coerente (**RSP_ID** com o mesmo valor de **CMD_ID**). Neste caso, utiliza-se a seguinte resposta:

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | “ERR” |
| RSP_STAT | N3 | ↳ ST_NOSEC = “Comunicação Segura” não estabelecida (ver seção 5.2); ↳ ST_ERRPKTSEC = Erro de decodificação de PKTDATA no caso de pacote criptografado (ver seção 5.2); ou ↳ ST_INVCALL = CMD_ID não é conhecido pelo pinpad. |

3. Comandos

Este capítulo detalha os comandos processados pelo pinpad no **Nível de Aplicação**, conforme formato apresentado na **seção 2.3**.

▲ Todos os formatos e exemplos descritos neste capítulo desconsideram o Nível de Enlace, bem como o modo de “Comunicação Segura”, dado que os comandos e respostas operam acima destas “camadas”.

3.1. Informações Preliminares

3.1.1. Códigos de retorno

Conforme apresentado na **seção 2.3**, os pacotes de resposta do pinpad devem conter uma informação de “resultado do processamento” (**RSP_STAT**) indicando sucesso ou, em caso de falha, qual o motivo. Os valores aceitos por esta especificação estão descritos na tabela a seguir:

| Nome | Valor | Descrição |
|------------|-------|--|
| ↵ST_OK | 000 | Comando executado com sucesso. |
| ↵ST_NOSEC | 003 | Tentativa de uso de “Comunicação Segura” quando esta não foi estabelecida. |
| ↵ST_F1 | 004 | Pressionada tecla de função #1. |
| ↵ST_F2 | 005 | Pressionada tecla de função #2. |
| ↵ST_F3 | 006 | Pressionada tecla de função #3. |
| ↵ST_F4 | 007 | Pressionada tecla de função #4. |
| ↵ST_BACKSP | 008 | Pressionada tecla de limpar (“backspace”). |

| Nome | Valor | Descrição |
|-------------------|-------|--|
| ↳ST_ERRPKTSEC | 009 | Erro na decodificação dos dados recebidos via “Comunicação Segura”; ou Comando recebido “em claro” com a “Comunicação Segura” estabelecida. |
| ↳ST_INVCALL | 010 | Chamada inválida à função (operações prévias são necessárias) ou comando desconhecido (em caso de resposta “ERR”). |
| ↳ST_INVPARAM | 011 | Um parâmetro inválido foi passado à função. |
| ↳ST_TIMEOUT | 012 | Esgotado o tempo máximo estipulado para a operação. |
| ↳ST_CANCEL | 013 | Operação cancelada pelo portador do cartão. |
| ↳ST_MANDAT | 019 | Um parâmetro mandatário não foi enviado pelo SPE. |
| ↳ST_TABVERDIF | 020 | Versão das Tabelas EMV difere da esperada. |
| ↳ST_TABERR | 021 | Erro ao tentar gravar tabelas (falta de espaço, por exemplo). |
| ↳ST_INTERR | 040 | Erro interno do pinpad (situação inesperada que não possui correspondência com os outros códigos de erro aqui descritos). |
| ↳ST_MCDATAERR | 041 | Erro de leitura do cartão magnético. |
| ↳ST_ERRKEY | 042 | MK / DUKPT referenciado não está presente no pinpad. |
| ↳ST_NOCARD | 043 | Não há ICC presente no acoplador ou CTLS detectado na antena. |
| ↳ST_PINBUSY | 044 | Pinpad não pode processar a captura de PIN temporariamente devido a questões de segurança (como quando é atingido o limite de capturas dentro de um intervalo de tempo). |
| ↳ST_RSPOVRFL | 045 | Montagem dos dados de resposta supera o tamanho máximo permitido. |
| ↳ST_ERRCRYPT | 046 | Erro genérico de validação criptográfica. |
| ↳ST_SECURITY | 047 | Operação finalizada por motivo de segurança. |
| ↳ST_DUMBCARD | 060 | ICC inserido, mas não responde (“mudo”). |
| ↳ST_ERRCARD | 061 | Erro de comunicação entre o pinpad e o ICC ou CTLS . |
| ↳ST_CARDINVALIDAT | 067 | Cartão foi invalidado. |
| ↳ST_CARDPROBLEMS | 068 | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↳ST_CARDINVDATA | 069 | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ↳ST_CARDAPPNAV | 070 | ICC EMV sem nenhuma aplicação disponível para as condições pedidas. |
| ↳ST_CARDAPPNAUT | 071 | A aplicação selecionada no ICC EMV não pode ser utilizada nesta situação. |

| Nome | Valor | Descrição |
|------------------------------|-------|--|
| ↳ST_ERRFALLBACK | 076 | Erro de alto nível no ICC EMV que é passível de “fallback” para tarja magnética. |
| ↳ST_INVAMOUNT | 077 | Valor inválido para a transação. |
| ↳ST_ERRMAXAID | 078 | Número de AIDs candidatos supera a capacidade de tratamento do Kernel EMV. |
| ↳ST_CARDBLOCKED | 079 | Cartão ICC ou CTLS está bloqueado. |
| ↳ST_CTLSMULTIPLE | 080 | Mais de um CTLS foi apresentado ao leitor simultaneamente. |
| ↳ST_CTLSCOMMERR | 081 | Erro de comunicação entre o pinpad (antena) e o CTLS. |
| ↳ST_CTLSINVALIDAT | 082 | CTLS foi invalidado (reservado). |
| ↳ST_CTLSPROBLEMS | 083 | CTLS com problemas. Esse status é válido para muitas ocorrências no processamento onde o CTLS não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↳ST_CTLSAPPNAV | 084 | CTLS sem nenhuma aplicação disponível para as condições pedidas (reservado). |
| ↳ST_CTLSAPPNAUT | 085 | A aplicação selecionada no CTLS não pode ser utilizada nesta situação (reservado). |
| ↳ST_CTLSEXTCVM | 086 | Portador deve efetuar operação no seu dispositivo (telefone celular, por exemplo) para depois reapresentá-lo ao terminal. |
| ↳ST_CTLSIFCHG | 087 | Processamento de CTLS resultou em “mudança de interface” (solicitar ICC ou cartão magnético). |
| ↳ST_MFNFOUND | 100 | Arquivo multimídia inexistente. |
| ↳ST_MFERRFMT | 101 | Erro no formato do arquivo multimídia. |
| ↳ST_MFERR | 102 | Erro na carga do arquivo multimídia. |

⚠ Nas seções de detalhamento dos comandos nesta especificação, procura-se listar somente os retornos relevantes para o comando sendo descrito. A maioria dos comandos admite os retornos ↳ST_OK, ↳ST_INVPARM, ↳ST_MANDAT e ↳ST_INTERR e, portanto, estes são omitidos para simplificar o documento.

3.1.2. Comandos obsoletos

Alguns comandos aqui descritos são considerados **obsoletos**, ou seja, eles serão retirados em versões futuras desta especificação.

⚠ O SPE **não deve utilizar um comando obsoleto** para um pinpad que reconhecidamente segue esta especificação. Para reconhecer um pinpad Abecs, deve-se usar o comando descrito na **seção 3.2.2**.

⚠ O pinpad **deve implementar um comando obsoleto** enquanto este estiver descrito nesta especificação, de forma a manter a compatibilidade com sistemas legados.

Os comandos definidos como obsoletos são identificados individualmente ao longo deste capítulo.

3.1.3. Comandos Abecs

Todos os comandos novos desta especificação (não contemplados em  **BibComp**) são denominados “**Comandos Abecs**” e seguem um formato flexível, em que os parâmetros e dados de resposta são codificados de forma padronizada, sempre precedidos por uma identificação e um tamanho, similar à codificação TLV descrita na **seção 7.1**, porém de forma proprietária e simplificada. Isso permite total flexibilidade em eventuais evoluções futuras dos comandos.

Para os “Comandos Abecs”, os pacotes de dados trafegados entre o SPE o pinpad podem ter até 2044 bytes. Para os demais comandos desta especificação o limite é de 1024 bytes.

3.1.3.1. Formato dos comandos

Os comandos enviados do SPE ao pinpad seguem o seguinte formato:

| Id. do Campo | | Formato | Descrição |
|--------------|------------|---------|--|
| CMD_ID | | A3 | Código do comando. |
| CMD_LEN1 | | N3 | Tamanho dos dados a seguir. |
| CMD_BLK1 | CMD_PARID | X2 | Identificador do parâmetro (SPE_xxxx). |
| | CMD_PARLEN | X2 | Tamanho do parâmetro, até 995 (03E3h). |
| | CMD_PAR | ??? | Dados do parâmetro. |
| | ... | ... | ... |
| | CMD_PARID | X2 | Identificador do parâmetro (SPE_xxxx). |
| | CMD_PARLEN | X2 | Tamanho do parâmetro, até 995 (03E3h). |
| | CMD_PAR | ??? | Dados do parâmetro. |
| CMD_LEN2 | | N3 | Tamanho dos dados a seguir. |
| CMD_BLK2 | CMD_PARID | X2 | Identificador do parâmetro (SPE_xxxx). |
| | CMD_PARLEN | X2 | Tamanho do parâmetro, até 995 (03E3h). |
| | CMD_PAR | ??? | Dados do parâmetro. |
| | ... | ... | ... |

➡ Regras de composição

- O SPE poderá enviar os parâmetros em qualquer ordem, não necessariamente a mesma apresentada na descrição dos comandos neste **Capítulo**.
- O SPE poderá dividir os parâmetros em um ou mais blocos (CMD_BLKn), dado que o discriminador de tamanho CMD_LENn permite um máximo de somente 999 bytes.
- Os parâmetros enviados ao pinpad podem ser mandatários ou opcionais, de acordo com a necessidade do comando. O pinpad desprezará eventuais parâmetros desconhecidos ou desnecessários ao processamento do comando.

➤ Representação na especificação

Para simplificar a especificação dos Comandos Abecs neste **Capítulo**, adota-se a seguinte convenção:

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|------------------------------|
| CMD_ID | M | Código do comando (= "XXX"). |
| SPE_xxx | (*) | Parâmetro de entrada. |
| ... | ... | ... |
| SPE_xxx | (*) | Parâmetro de entrada. |

(*) Definição de presença:

M = Parâmetro é mandatório para o comando. Se não for enviado pelo SPE, o pinpad retornará ↵ST_MANDAT.

MD = Parâmetro é mandatório dependendo da situação (pode ser um complemento de outro parâmetro, por exemplo). Se a situação exigir o parâmetro e o SPE não o enviar, o pinpad retornará ↵ST_MANDAT.

O = Parâmetro é opcional e o SPE somente o enviará caso desejado para o processamento do comando. Caso o pinpad necessite desta informação, ele deve usar um valor "default" predefinido no **Capítulo 6**.

➤ Lista de parâmetros previstos

| CMD_PARID | Valor | Formato | Descrição |
|-------------|-------|------------------------|--|
| SPE_IDLIST | 0001h | B..128 (n×X2, n≤64) | Lista de até 64 identificadores (concatenados) de dados de retorno do pinpad. |
| SPE_MTHDPIN | 0002h | N1 | Método a ser usado na criptografia de PIN: "1" = MK/WK:TDES:PIN; e "3" = DUKPT:TDES:PIN (ver seção 5.1.1). |
| SPE_MTHDDAT | 0003h | N2 | Método a ser usado na criptografia de dados: "10" = MK/WK:TDES:DAT (criptografia de bloco ECB); "11" = MK/WK:TDES:DAT (criptografia de bloco CBC); "50" = DUKPT:TDES:DAT#3 (criptografia de bloco ECB , ver seção 5.1.1); e "51" = DUKPT:TDES:DAT#3 (criptografia de bloco CBC , ver seção 5.1.1). |
| SPE_TAGLIST | 0004h | B..128 | Lista de <i>tags</i> referentes aos objetos EMV requeridos pelo SPE. |
| SPE_EMVDATA | 0005h | B..512 | Lista de objetos EMV enviados ao pinpad (no formato TLV - ver seção 7.1). |

| CMD_PARID | Valor | Formato | Descrição |
|-------------|-------|---------|---|
| SPE_CEXOPT | 0006h | A6 | Opções do comando " CEX ". "0xxxxx" = Ignora teclas; "1xxxxx" = Verifica pressionamento de tecla. "x0xxxx" = Ignora cartão magnético; "x1xxxx" = Verifica passagem de cartão magnético. "xx0xxx" = Ignora ICC; "xx1xxx" = Verifica inserção de ICC; "xx2xxx" = Verifica remoção de ICC. "xxx0xx" = Ignora CTLS (não ativa a antena); "xxx1xx" = Ativa a antena e verifica a presença de um CTLS. "xxxx00" = RUF. |
| SPE_TRACKS | 0007h | N4 | Identificação dos dados de trilha a serem devolvidos pelo pinpad no comando " GTK ". |
| SPE_OPNDIG | 0008h | N1 | Quantidade de dígitos numéricos (número par) a serem preservados "em claro" no início das trilhas criptografadas (valores aceitos: "0", "2", "4", "6", "8"). |
| SPE_KEYIDX | 0009h | N2 | Índice de uma chave de criptografia MK ou DUKPT ("00" a "99") |
| SPE_WKENC | 000Ah | B16 | <i>Working Key</i> criptografada pela MK:TDES. |
| SPE_MSGIDX | 000Bh | X2 | Índice da mensagem a ser apresentada. |
| SPE_TIMEOUT | 000Ch | X1 | Tempo de espera por uma ação do portador do cartão em segundos (0 a 255), sendo que 0 indica que o comando retorna imediatamente. IMPORTANTE: Este dado reflete o tempo de <u>inatividade</u> do portador e não o tempo máximo de execução do comando. |
| SPE_MINDIG | 000Dh | X1 | Quantidade mínima de dígitos a ser capturada no pinpad (de 0 a 32). |
| SPE_MAXDIG | 000Eh | X1 | Quantidade máxima de dígitos a ser capturada no pinpad (de 1 a 32). |
| SPE_DATAIN | 000Fh | B..995 | Dados genéricos a serem enviados ao pinpad. |
| SPE_ACQREF | 0010h | N2 | Identificador da Rede Credenciadora para pesquisa nas Tabelas de AID (de "01" a "99"). |
| SPE_APPTYPE | 0011h | N..20 | Identificadores do tipo de aplicação, para pesquisa nas Tabelas de AID (de "01" a "98"). Este campo suporta de 1 a 10 identificadores diferentes. |

| CMD_PARID | Valor | Formato | Descrição |
|--------------|-------|---------|---|
| SPE_AIDLIST | 0012h | A..512 | Lista específica de registros das Tabelas de AID para uso no processamento transação, podendo contemplar até 128 entradas no formato "AARR", sendo: "AA" = Identificador da Rede Credenciadora responsável pela tabela (de "01" a "99"); e "RR" = Índice do registro na tabela (de "01" a "ZZ"). |
| SPE_AMOUNT | 0013h | N12 | Valor da transação em centavos (<i>Amount, authorized</i>). |
| SPE_CASHBACK | 0014h | N12 | Valor da transação referente a saque ou troco - <i>cashback</i> (<i>Amount, other</i>) em centavos. |
| SPE_TRNDATE | 0015h | N6 | Data da transação ("AAMMDD") |
| SPE_TRNTIME | 0016h | N6 | Hora da transação ("HHMMSS") |
| SPE_GCXOPT | 0017h | N5 | Opções do comando " G <u>C</u> X ": "0xxxx" = Aguarda cartão magnético ou ICC; ou "1xxxx" = Aguarda cartão magnético, ICC ou CTLS. "x0xxx" = Mostra o valor da transação na tela de espera pelo cartão, se este for diferente de zero. "x1xxx" = Não mostra o valor da transação. "xx000" = RUF. |
| SPE_GOXP | 0018h | N5 | Opções do comando " G <u>O</u> X ": "1xxxx" = PAN consta na Lista de Exceção (só usado se ICC EMV). "x1xxx" = Transação não pode ser aprovada <i>offline</i> (só usado se ICC EMV). "xx1xx" = Não permite <i>bypass</i> de PIN. "xxx00" = RUF. |
| SPE_FCXP | 0019h | N4 | Opções do comando " F <u>C</u> X ": "0xxx" = Transação <u>aprovada</u> pela Rede Credenciadora. "1xxx" = Transação <u>negada</u> pela Rede Credenciadora. "2xxx" = A comunicação foi malsucedida (ou não foi possível receber uma resposta válida da Rede Credenciadora). "x000" = RUF. |
| SPE_TRMPAR | 001Ah | B10 | Parâmetros para o processamento do <i>Terminal Risk Management</i> no comando " G <u>O</u> X ": <ul style="list-style-type: none"> ▪ <i>Terminal Floor Limit</i> (formato "X4", em centavos); ▪ <i>Target Percentage to be used for Biased Random Selection</i> (formato "X1"); ▪ <i>Threshold Value for Biased Random Selection</i> (formato "X4", em centavos); e ▪ <i>Maximum Target Percentage to be used for Biased Random Selection</i> (formato "X1"). |

| CMD_PARID | Valor | Formato | Descrição |
|-------------|-------|---------|--|
| SPE_DSPMSG | 001Bh | S..128 | Mensagem a ser apresentada no <i>display</i> do pinpad, em formato livre, com caracteres de quebras de linha (0Dh). Ao formatar esta mensagem, o SPE deve se atentar às capacidades do <i>display</i> do pinpad (ver PP_DSPTXTSZ). |
| SPE_ARC | 001Ch | A2 | <i>Authorization Response Code</i> (código de aprovação/negação devolvido pela Rede Credenciadora). |
| SPE_IVCBC | 001Dh | B8 | "IV" (<i>Initialization Vector</i>) a ser usado na criptografia de bloco CBC . |
| SPE_MFNAME | 001Eh | A8 | Nome do arquivo multimídia (somente caracteres numéricos e letras, sem espaços ou símbolos). O nome de arquivo <u>não é "case sensitive"</u> , ou seja, os nomes "ImgAlt01" e "IMGALT01" representam o mesmo arquivo. |
| SPE_MFINFO | 001Fh | B10 | Informações sobre o arquivo multimídia: X4 = Tamanho (de 0 a 4294967295 bytes). B2 = CRC do arquivo. B1 = Tipo (01h = PNG , 02h = JPG , 03h = GIF , outros valores = RUF); e B3 = RUF (000000h). |
| SPE_MNUOPT | 0020h | S..24 | Texto com uma opção de menu. |
| SPE_TRNTYPE | 0021h | B1 | Tipo de transação a ser efetuada: 00h = Compra; 01h = Saque; 09h = Compra com saque/troco (<i>cashback</i>); 20h = Cancelamento (<i>refund</i>); 30h = Consulta de saldo; ou Outros valores de acordo com ISO 8583:1987. |
| SPE_TRNCURR | 0022h | N3 | Código da moeda a ser usada na transação se ICC (ex.: Real = "986", Dólar = "840"). |
| SPE_PANMASK | 0023h | N4 | Definição para mascaramento do PAN, no formato "eedd", sendo: "ee" = Quantidade de dígitos em aberto à esquerda; e "dd" = Quantidade de dígitos em aberto à direita. |
| SPE_PBKMOD | 0024h | B256 | Módulo de uma chave pública RSA de 2048 bits. |
| SPE_PBKEXP | 0025h | B..3 | Expoente de uma chave pública RSA. |
| SPE_GCDOPT | 0026h | N4 | Opções do comando " GCD ": "0xxx" = Entrada numérica; "1xxx" = Entrada alfanumérica; "x000" = RUF. |

3.1.3.2. Formato das respostas

As respostas devolvidas ao SPE pelo pinpad seguem o seguinte formato:

| Id. do Campo | | Formato | Descrição |
|--------------|------------|---------|--|
| RSP_ID | | A3 | Código do comando. |
| RSP_STAT | | N3 | Resultado do processamento do comando, conforme definido na seção 3.1 . |
| RSP_LEN1 | | N3 | Tamanho dos dados a seguir (RSP_BLK1). |
| RSP_BLK1 | RSP_DATID | X2 | Identificador do dado de resposta (PP_xxxx). |
| | RSP_DATLEN | X2 | Tamanho do dado de resposta, até 995 (03E3h). |
| | RSP_DAT | ??? | Dado de resposta. |
| | ... | ... | ... |
| | RSP_DATID | X2 | Identificador do dado de resposta (PP_xxxx). |
| | RSP_DATLEN | X2 | Tamanho do dado de resposta, até 995 (03E3h). |
| | RSP_DAT | ??? | Dado de resposta. |
| RSP_LEN2 | | N3 | Tamanho dos dados a seguir (RSP_BLK2). |
| RSP_BLK2 | RSP_DATID | X2 | Identificador do dado de resposta (PP_xxxx). |
| | RSP_DATLEN | X2 | Tamanho do dado de resposta, até 995 (03E3h). |
| | RSP_DAT | ??? | Dado de resposta. |
| | ... | ... | ... |

➤ Regras de composição

- O pinpad poderá devolver os parâmetros em qualquer ordem, não necessariamente a mesma apresentada na descrição dos comandos neste **Capítulo**.
- O pinpad poderá dividir os dados de resposta em um ou mais blocos (**RSP_BLK_n**), dado que o discriminador de tamanho **RSP_LEN_n** permite um máximo de somente 999 bytes.
- Os dados de resposta devolvidos pelo pinpad podem ser mandatórios ou opcionais, de acordo com a necessidade do comando. O SPE desprezará eventuais dados de resposta desconhecidos ou desnecessários ao processamento do comando.

➤ Representação na especificação

Para simplificar a especificação dos Comandos Abecs neste **Capítulo**, adota-se a seguinte convenção:

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|--|
| CMD_ID | M | Código da resposta (= " XXX "). |
| RSP_STAT | M | Somente os retornos de erro <u>relevantes</u> são apresentados, de forma a complementar a seção 3.1 . |

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|------------------------|
| PP_xxx | (*) | Dado de resposta. |
| ... | ... | ... |
| PP_xxx | (*) | Dado de resposta. |

(*) Definição de presença:

M = Dado de retorno é mandatório para o comando. Se não for devolvido pelo pinpad, o SPE deverá finalizar a operação com erro fatal.

MD Dado é mandatório dependendo da situação (pode ser um complemento de outro dado, por exemplo). Se a situação exigir um dado não devolvido pelo pinpad, o SPE deverá finalizar a operação com erro fatal.

MR Dado é mandatório se requerido pelo SPE no comando recebido.

O = Dado é opcional e depende do processamento. Sua eventual ausência não deve ser criticada pelo SPE.

➤ Lista de dados de retorno previstos

| RSP_DATID | Valor | Formato | Descrição |
|---------------------------|-------|---------|---|
| PP_SERNUM ^(†) | 8001h | A..32 | Número de série do pinpad (formato livre, depende do fabricante). |
| PP_PARTNBR ^(†) | 8002h | A..32 | "Part Number" do pinpad (formato livre, depende do fabricante). |
| PP_MODEL ^(†) | 8003h | A..20 | Modelo / versão do <i>hardware</i> , no formato " xx...xx;m...m ", onde: <ul style="list-style-type: none"> ▪ "xx...xx" é o nome do equipamento; e ▪ "m...m" a capacidade de memória ("512KB", "1MB", "2MB", ...). |
| PP_MNNAME ^(†) | 8004h | A..20 | Nome do fabricante do pinpad (formato livre). |

| RSP_DATID | Valor | Formato | Descrição |
|-----------------------------|-------|---------|--|
| PP_CAPAB ^(†) | 8005h | A10 | <p>Capacidades do pinpad:</p> <p>“0xxxxxxxx” = Não suporta CTLS;</p> <p>“1xxxxxxxx” = Suporta CTLS.</p> <p>“x0xxxxxxxx” = Não Possui <i>display</i> não gráfico;</p> <p>“x1xxxxxxxx” = Possui <i>display</i> gráfico monocromático;</p> <p>“x2xxxxxxxx” = Possui <i>display</i> gráfico colorido.</p> <p>“x9xxxxxxxx” = Não possui <i>display</i>.</p> <p>“xx0xxxxxxxx” = O suporte a ICC não é informado (especificação ≤ V2.12);</p> <p>“xx1xxxxxxxx” = Suporta ICC;</p> <p>“xx9xxxxxxxx” = Não suporta ICC.</p> <p>“xxx0xxxxxx” = O suporte a cartão magnético não é informado (especificação ≤ V2.12).</p> <p>“xxx1xxxxxx” = Suporta cartão magnético trilhas 1, 2 e 3.</p> <p>“xxx2xxxxxx” = Suporta cartão magnético trilhas 1 e 2.</p> <p>“xxx3xxxxxx” = Suporta cartão magnético trilha 1.</p> <p>“xxx4xxxxxx” = Suporta cartão magnético trilha 2.</p> <p>“xxx9xxxxxx” = Não suporta cartão magnético.</p> <p>“xxxx0xxxxx” = O suporte a SAM não é informado (especificação ≤ V2.12).</p> <p>“xxxxNxxxxx” = Possui slots de SAM de 1 a “N”.</p> <p>“xxxx9xxxxx” = Não possui slot para SAM.</p> <p>“xxxxx00000” = RUF.</p> |
| PP_SOVER ^(†) | 8006h | A..20 | Versão do <i>software</i> básico ou sistema operacional (formato livre). |
| PP_SPECVER ^(†) | 8007h | A4 | Versão da especificação, no formato “V.VV” (neste caso, fixo “2.20”) |
| PP_MANVERS ^(†) | 8008h | A16 | Versão da aplicação “Gerenciadora”, no formato “VVV.VV AAMMDD”. |
| PP_APPVERS ^(†) | 8009h | A16 | Versão da aplicação “Abecs”, no formato “VVV.VV AAMMDD”. |
| PP_GENVERS ^(†) | 800Ah | A16 | Versão da aplicação “Extensão”, no formato “VVV.VV AAMMDD”. |
| PP_CTLSCAPAB ^(†) | 800Bh | A..16 | <p>Lista contendo os padrões de CTLS suportados, com codificação similar à usada em T1_CTLSMODE. Para esta especificação, o valor a ser retornado é “2469B”, sendo:</p> <p>“2” = VISA payWave;</p> <p>“4” = Mastercard PayPass;</p> <p>“6” = Amex Expresspay;</p> <p>“9” = Discover D-PAS; e</p> <p>“B” = Union Pay Quick Pass.</p> |
| PP_KRNLVER ^(†) | 8010h | A..20 | Versão do Kernel EMV para ICC. |
| PP_CTLsver ^(†) | 8011h | A..20 | Versão principal (<i>gerenciador ou Entry Point</i>) dos Kernels EMV para CTLS. |

| RSP_DATID | Valor | Formato | Descrição |
|--|-------|-------------------------|--|
| PP_MCTLSVER ^(†) | 8012h | A..20 | Versão do Kernel EMV CTLS para cartões Mastercard PayPass. |
| PP_VCTLSVER ^(†) | 8013h | A..20 | Versão do Kernel EMV CTLS para cartões VISA payWave. |
| PP_AECTLSVER ^(†) | 8014h | A..20 | Versão do Kernel EMV CTLS para cartões Amex ExpressPay. |
| PP_DPCTLSVER ^(†) | 8015h | A..20 | Versão do Kernel EMV CTLS para cartões Discover D-PAS. |
| PP_PUREVER^(‡) ----- | 8016h | A..20 --- | Versão do Kernel EMV CTLS para cartões Pure- Reservado. |
| ----- | 8017h | --- | Reservado para JCB. |
| PP_QPCTLSVER ^(†) | 8018h | A..20 | Versão do Kernel EMV CTLS para cartões Union Pay Quick Pass. |
| PP_DSPTXTSZ ^(†) | 8020h | N4 | Número máximo de linhas e colunas do <i>display</i> do pinpad para apresentação de mensagens em modo texto (formato "LLCC"). |
| PP_DSPGRSZ ^(†) | 8021h | N8 | Número máximo de linhas e colunas do <i>display</i> gráfico do pinpad para apresentação de imagens (formato "LLLLCCCC", em pixels). |
| PP_MFSUP ^(†) | 8022h | A..20 | Tipos de arquivo multimídia suportados: "1xx..." = Suporta o formato PNG; "x1x..." = Suporta o formato JPG. "xx1x..." = Suporta o formato GIF. |
| ----- | 8030h | --- | Reservado. |
| ----- | 8031h | --- | Reservado. |
| PP_MKTDESP ^(†) | 8032h | A100 | 100 caracteres contendo o mapa de chaves MK:TDES:PIN contidas no pinpad, sendo que cada caractere corresponde a uma posição (de "00" a "99"), indicando: "0" = Chave ausente (não carregada); "1" = Chave presente (carregada); e "2" = Posição não suportada pelo pinpad. |
| PP_MKTDESD ^(†) | 8033h | A100 | Idem para chaves MK:TDES:DAT . |
| ----- | 8034h | --- | Reservado. |
| PP_DKPTTDESP ^(†) | 8035h | A100 | Idem para chaves DUKPT:TDES:PIN . |
| PP_DKPTTDESD ^(†) | 8036h | A100 | Idem para chaves DUKPT:TDES:DAT . |

| RSP_DATID | Valor | Formato | Descrição |
|------------|-------|---------|--|
| PP_EVENT | 8040h | A2 | Evento detectado pelo pinpad no comando “ CEX ”: “00” = Tecla [OK/ENTRA] foi pressionada; “02” = Tecla [↑] foi pressionada; “03” = Tecla [↓] foi pressionada; “04” = Tecla [F1] foi pressionada; “05” = Tecla [F2] foi pressionada; “06” = Tecla [F3] foi pressionada; “07” = Tecla [F4] foi pressionada; “08” = Tecla [LIMPA] foi pressionada; “13” = Tecla [CANCELA] foi pressionada; “90” = Um cartão magnético foi passado no leitor; “91” = ICC foi removido (ou já não estava presente); “92” = ICC foi inserido (ou já estava presente); “93” = CTLS não foi detectado em 2 (dois) minutos; e “94” = CTLS foi detectado. |
| PP_TRK1INC | 8041h | A..60 | Trilha 1 do cartão, <u>incompleta</u> (ver seção 5.4.1) |
| PP_TRK2INC | 8042h | A..30 | Trilha 2 do cartão, <u>incompleta</u> (ver seção 5.4.1) |
| PP_TRK3INC | 8043h | A..30 | Trilha 3 do cartão, <u>incompleta</u> (ver seção 5.4.1) |
| PP_TRACK1 | 8044h | B..88 | Trilha 1 completa do cartão, aberta ou criptografada (ver seção 5.4.2.1). OBS: Apesar da trilha 1 ser representada em codificação ASCII, este campo segue o formato “B” para o caso de seus dados estarem criptografados. |
| PP_TRACK2 | 8045h | B..28 | Trilha 2 completa do cartão, aberta ou criptografada (ver seção 5.4.2.2). Cada símbolo da trilha 2 ocupa um <i>nibble</i> , de acordo com a seguinte codificação: 0h (0000) → “0” Ah (1010) → “.” Dh (1101) → “=” ... Bh (1011) → “;” Eh (1110) → “>” 9h (1001) → “9” Ch (1100) → “<” Fh (1111) → “?” Os dados são alinhados à <u>esquerda</u> , com preenchimento Fh (“?”) à direita, se necessário. |
| PP_TRACK3 | 8046h | B..60 | Trilha 3 completa do cartão, aberta ou criptografada (mesmo formato de PP_TRACK2). |
| PP_TRK1KSN | 8047h | B10 | KSN da chave DUKPT usada na criptografia da Trilha 1. |
| PP_TRK2KSN | 8048h | B10 | KSN da chave DUKPT usada na criptografia da Trilha 2. |
| PP_TRK3KSN | 8049h | B10 | KSN da chave DUKPT usada na criptografia da Trilha 3. |
| PP_ENCPAN | 804Ah | B..16 | PAN do cartão, aberto ou criptografado (ver seção 5.4.2.2). Cada dígito do PAN ocupa um <i>nibble</i> , com alinhamento à esquerda e preenchimento Fh à direita, caso necessário. Exemplo: O PAN “9781234789432” é codificado como: 97h 81h 23h 47h 89h 43h 2Fh. |

| RSP_DATID | Valor | Formato | Descrição |
|---------------|-------|---------|--|
| PP_ENCPANKSN | 804Bh | B10 | KSN da chave DUKPT usada na criptografia do PAN. |
| PP_KSN | 804Ch | B10 | KSN da chave DUKPT usada na criptografia (PIN ou dados). |
| PP_VALUE | 804Dh | A..32 | Valor capturado pelo pinpad. |
| PP_DATAOUT | 804Eh | B..256 | Dados genéricos devolvidos pelo pinpad. |
| PP_CARDTYPE | 804Fh | N2 | Resposta de “ G CX” : Tipo de cartão detectado. “00” = Magnético; “03” = ICC EMV; “05” = CTLS simulando tarja; e “06” = CTLS EMV. |
| PP_ICCSTAT | 8050h | N1 | Resposta de “ G CX” : Status da última leitura de ICC. |
| PP_AIDTABINFO | 8051h | A..120 | Resposta de “ G CX” : Informações da Tabela AID, podendo conter até 20 registros do tipo “A6” concatenados. |
| PP_PAN | 8052h | N..19 | PAN do cartão lido. |
| PP_PANSEQNO | 8053h | N2 | <i>Application PAN Sequence Number</i> do cartão lido. |
| PP_EMVDATA | 8054h | B..512 | Lista de objetos EMV devolvidos pelo pinpad (no formato TLV - ver seção 7.1). |
| PP_CHNAME | 8055h | A..26 | Nome do portador do cartão lido. |
| PP_GOXRES | 8056h | N6 | Resultado do processamento EMV na resposta de “ G OX” : “0xxxxx” = Transação aprovada <i>offline</i> ; “1xxxxx” = Transação negada; ou “2xxxxx” = Transação requer autorização <i>online</i> . “x1xxxx” = Deve-se coletar assinatura em papel. “xx1xxx” = PIN foi verificado com sucesso <i>offline</i> . “xx2xxx” = PIN capturado para verificação <i>online</i> . “xxx1xx” = Verificação de portador efetuada no dispositivo móvel (telefone celular, por exemplo). “xxxx00” = RUF. |
| PP_PINBLK | 8057h | B8 | PIN criptografado. |
| PP_FCXRES | 8058h | N3 | Resultado do processamento EMV na resposta de “ F CX” : “0xx” = Transação aprovada; ou “1xx” = Transação negada. “x00” = RUF. |
| PP_ISRESULTS | 8059h | B..50 | <i>Issuer Script Results</i> (múltiplo de 5 - até 10 resultados). |
| PP_BIGRAND | 805Ah | B900 | 900 bytes aleatórios gerados pelo pinpad (usado apenas para testes). |
| PP_LABEL | 805Bh | S..16 | Etiqueta da aplicação sendo processada (ICC EMV ou CTLS). |

| RSP_DATID | Valor | Formato | Descrição |
|--------------------------|---------------------|---------|---|
| PP_ISSCNTRY | 805Ch | N3 | Código do país do cartão (<i>Issuer Country Code</i>). |
| PP_CARDEXP | 805Dh | N6 | Data de expiração do cartão (<i>Application Expiration Date</i>), no formato "AAMMDD". |
| PP_MFNAME | 805Eh | A8 | Nome de um arquivo multimídia carregado no pinpad, sempre em letras maiúsculas. |
| PP_DEVTYPE | 8060h | N2 | Tipo de dispositivo usado na transação: "00" = Cartão; "01", "07", "08", "09", "14", "16", "21" = Telefone móvel; "02" = Chaveiro; "03", "18", "19", "23" = Relógio; "04", "27" = Etiqueta móvel (" <i>mobile tag</i> "); "05" = Pulseira / bracelete; "06" = Capa de telefone móvel (" <i>case/sleeve</i> "); "10", "11", "12", "13", "15", "17", "22" = Tablet ou e-Reader; Outros valores = Uso futuro. |
| ----- | 8061h | --- | Reservado |
| PP_TLRMEM ^(*) | 8062h | X4 | Quantidade de memória disponível (em bytes) para carga dos registros das Tabelas EMV usando o comando " TLR ". |
| PP_ENCKRAND | 8063h | B256 | Chave aleatória K_{RAND} criptografada por uma chave pública RSA no formato PKCS #1. |
| PP_BATTINFO | 8064h | B2 | Informações sobre bateria somente para <u>pinpads móveis</u> , no formato XXh YYh, sendo: XXh (X1) = Nível da bateria de 0% (00h) a 100% (64h); e YYh (B1) = Máscara de bits, sendo: 'xxxxxx0' = se desconectado da fonte de carga; 'xxxxxx1' = se conectado à fonte de carga; '000000x' = RUF |
| PP_COMMINFO | 8065h | N4 | Informações sobre o modo de comunicação sendo usado pelo pinpad no momento: "0xxx" = Serial RS-232 ou USB; "1xxx" = Bluetooth; "2xxx" = Wi-Fi; "7xxx" = Outro/proprietário (com fio); "8xxx" = Outro/proprietário (sem fio); "9xxx" = Biblioteca / software. "x000" = RUF |
| ----- | 9000h a 9063h | --- | Faixa reservada. |

| RSP_DATID | Valor | Formato | Descrição |
|---------------|---------------------|---------|---|
| PP_KSNTDESPnn | 9100h a 9163h | B10 | KSN da chave DUKPT:TDES:PIN, índice #nn (de 00 a 99). IMPORTANTE: Deve-se tomar cuidado com o valor hexadecimal apresentado (PP_KSNTDESP14 = 910Eh)!! |
| PP_KSNTDESDnn | 9200h a 9263h | B10 | KSN da chave DUKPT:TDES:DAT, índice #nn (de 00 a 99) IMPORTANTE: Deve-se tomar cuidado com o valor hexadecimal apresentado (PP_KSNTDESD79 = 924Fh)!! |
| PP_TABVERnn | 9300h a 9363h | A10 | Versão das Tabelas EMV da rede credenciadora de índice #nn (00 a 99). O índice #00 corresponde à versão “geral” todas as redes. |

(*) Ver comando “**GIX**” (seção 3.2.4).

3.2. Comandos de controle

Esta seção detalha os seguintes comandos necessários para o controle do pinpad:

| CMD_ID | Significado | Obsoleto | Blocante | Abecs |
|--------|-----------------------------------|-------------------------------------|--------------------------|-------------------------------------|
| "OPN" | <i>Open Pinpad</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "OPN" | <i>Open Pinpad (Secure)</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "GIN" | <i>Get Information</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "GIX" | <i>Get Information - Extended</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| "DWK" | <i>Define WK_{PAN}</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "CLO" | <i>Close Pinpad</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "CLX" | <i>Close Pinpad - Extended</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

3.2.1. Comando “OPN” (clássico)

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

Este comando aloca os recursos de *hardware* e *software* necessários ao funcionamento do pinpad.

A chamada bem-sucedida deste comando é pré-requisito para todos os outros descritos adiante.

|| ⚠ Este formato de comando é **obsoleto**. O SPE deve usar o formato descrito na **seção 3.2.2**.

➡ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|------------------------------|
| CMD_ID | A3 | Código do comando (= “OPN”). |

➡ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|-------------------------------|
| RSP_ID | A3 | Código da resposta (= “OPN”). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |

➡ Exemplos

SPE solicita “abertura” do pinpad.

| | | |
|-------|----------|-----|
| SPE ⇒ | 4F 50 4E | OPN |
|-------|----------|-----|

A operação é bem-sucedida.

| | | |
|------|-------------------|--------|
| ⇐ PP | 4F 50 4E 30 30 30 | OPN000 |
|------|-------------------|--------|

3.2.2. Comando “OPN” (seguro)

Obsoleto
 Blocante
 ABECS

Este comando realiza as mesmas funções do “OPN” (clássico), porém estabelece a chave de “Comunicação Segura” entre o SPE e o pinpad (ver [seção 5.2](#)).

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= “OPN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| OPN_OPMODE | N1 | Modo de operação (por enquanto, fixo em “0”). |
| OPN_MODLEN | N3 | Quantidade de bytes representados em OPN_MOD (tamanho ÷ 2), fixo “256” (ver explicação na seção 5.2). |
| OPN_MOD | H512 | Módulo da chave RSA criada pelo SPE (K_{MOD}). |
| OPN_EXPLEN | N1 | Quantidade de bytes representados em OPN_EXP (tamanho ÷ 2). |
| OPN_EXP | H..6 | Expoente da chave RSA criada pelo SPE (K_{PUB}). |

➔ Resposta (pinpad Abecs)

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “OPN”). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| OPN_CRKLEN | N3 | Quantidade de bytes representados em OPN_CRKSEC (tamanho ÷ 2), fixo “256”. |
| OPN_CRKSEC | H512 | Criptograma (CRKSEC) gerado pela chave pública fornecida, contendo a chave K_{SEC} (ver formato na seção 5.2.1). |

➔ Resposta (pinpad obsoleto)

| Id. do Campo | Formato | Descrição |
|--------------|---------|-----------------------------------|
| RSP_ID | A3 | Código da resposta (= “OPN”). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |

⚠ Se o pinpad devolver este formato de resposta, significa que ele ainda não segue esta especificação. Neste caso específico, não há “Comunicação Segura” e, por questão de compatibilidade, o SPE não poderá usar Comandos Abecs.

➤ Exemplos

SPE solicita chave K_{SEC} , fornecendo uma chave RSA de módulo de 256 bytes e expoente público de valor 13 (0Dh).

| | | |
|-------|---|--|
| SPE ⇒ | <pre> 4F 50 4E 35 31 39 30 32 35 36 41 38 32 41 36 36 30 42 33 43 34 39 32 32 36 45 46 43 44 41 42 41 37 46 43 36 38 30 36 36 42 38 33 44 32 33 44 30 35 36 30 45 44 41 33 41 31 32 42 36 33 45 39 31 33 32 46 32 39 39 46 42 46 33 34 30 41 35 41 45 42 43 34 43 44 35 44 43 31 46 31 34 38 37 33 46 38 33 41 38 30 42 41 39 41 38 38 44 33 46 45 41 42 42 41 42 34 31 44 46 46 43 31 39 34 34 42 42 42 41 41 38 39 46 32 36 41 46 39 43 43 32 38 46 46 33 31 43 34 39 37 45 42 39 31 44 38 32 46 38 36 31 33 45 37 34 36 33 43 34 37 35 32 39 46 42 44 31 39 32 35 46 44 33 33 32 36 41 38 44 43 30 32 37 37 30 34 44 41 36 38 38 36 30 45 36 38 42 44 30 41 31 43 45 41 38 44 45 36 45 43 37 35 36 30 34 43 44 33 44 39 41 36 41 46 33 38 38 32 32 44 45 34 35 41 41 41 30 43 39 46 42 46 32 42 44 34 37 38 33 42 30 46 39 41 38 31 46 36 33 35 30 43 30 31 38 38 31 35 36 46 39 30 38 46 41 42 31 46 35 35 39 43 46 43 45 31 46 39 31 41 33 39 33 34 33 31 45 38 42 46 32 43 44 37 38 43 30 34 42 44 35 33 30 44 42 34 34 31 30 39 31 43 44 46 46 42 34 30 30 44 41 43 30 38 42 31 34 35 30 44 42 36 35 43 30 30 45 32 44 34 41 46 34 45 39 41 38 35 41 31 41 31 39 42 36 31 46 35 35 30 46 30 43 32 38 39 42 31 34 42 44 36 33 44 46 38 41 31 35 33 39 41 38 43 46 36 32 39 46 39 38 46 38 38 45 41 39 34 34 44 39 30 35 36 36 37 35 30 30 46 39 35 42 46 44 30 46 45 46 43 35 36 46 39 44 39 44 36 36 45 32 37 30 31 42 44 42 44 37 31 39 33 33 31 39 31 41 45 39 39 32 38 46 35 44 36 32 33 46 45 38 42 39 39 45 43 43 37 37 37 34 34 46 46 41 41 38 33 44 45 34 35 36 46 35 43 38 44 33 43 38 33 45 43 35 31 31 41 46 31 30 44 </pre> | <pre> OPN5190256A82A66 0B3C49226EFCDA8A 7FC68066B83D23D0 560EDA3A12B63E91 32F299FBF340A5AE BC4CD5DC1F14873F 83A80BA9A88D3FEA BBAB41DFFC1944BB BAA89F26AF9CC28F F31C497EB91D82F8 613E7463C47529FB D1925FD3326A8DC0 27704DA68860E68B D0A1CEA8DE6EC756 04CD3D9A6AF38822 DE45AAA0C9FBF2BD 4783B0F9A81F6350 C0188156F908FAB1 F559CFCE1F91A393 431E8BF2CD78C04B D530DB441091CDFF B400DAC08B1450DB 65C00E2D4AF4E9A8 5A1A19B61F550F0C 289B14BD63DF8A15 39A8CF629F98F88E A944D9056675000F 95BFD0FEFC56F9D9 D66E2701BDBD7193 3191AE9928F5D623 FE8B99ECC777444F FAA83DE456F5C8D3 C83EC511AF10D </pre> |
|-------|---|--|

Pinpad gera K_{SEC} aleatória (DB3B4D015432AB3223555A1F81759A94) e devolve o criptograma gerado pela chave pública.

| | | |
|------|---|--|
| ← PP | 4F 50 4E 30 30 30 35 31 35 32 35 36 34 45 35 38 30 35 45 35 41 43 46 33 42 45 34 41 33 46 44 32 37 33 30 30 45 36 38 32 44 44 42 30 32 38 44 43 34 33 32 32 33 44 36 44 32 45 35 39 44 42 31 32 42 43 42 35 32 44 32 33 38 44 31 38 37 35 43 46 31 39 41 36 39 46 45 34 30 35 32 42 37 46 45 44 30 31 36 30 41 44 46 33 30 30 36 44 38 44 36 36 31 35 36 41 41 31 41 30 41 35 35 45 32 46 31 41 30 34 35 33 32 32 46 45 44 35 39 34 35 42 32 46 34 41 37 41 36 45 36 36 43 38 44 32 46 41 39 37 34 37 39 44 33 31 42 31 30 36 46 45 43 31 41 35 39 33 37 30 31 38 34 41 43 36 33 37 33 42 31 30 35 33 44 41 39 42 45 37 44 43 30 31 42 32 41 41 31 38 43 32 30 38 45 31 43 30 37 37 39 43 30 43 43 44 37 44 34 34 39 36 45 35 33 32 36 45 39 38 41 45 37 34 34 43 43 43 35 38 43 41 37 42 34 36 33 30 44 39 36 44 44 33 37 46 42 37 42 37 39 44 36 46 42 41 37 39 33 30 31 38 43 39 32 43 36 31 35 31 36 39 33 39 43 43 41 31 32 44 31 39 32 34 31 34 36 31 36 30 35 44 35 38 39 30 38 32 42 42 35 45 44 37 39 45 35 41 45 37 32 30 43 39 44 43 43 30 37 32 35 30 46 45 45 35 32 37 44 31 38 41 44 38 41 42 33 37 34 39 45 32 45 45 30 44 34 38 44 39 42 43 32 45 30 41 45 44 37 35 41 44 37 34 39 45 31 31 41 33 37 39 43 33 37 42 36 38 34 30 31 30 34 38 41 44 37 39 44 45 32 35 34 45 30 42 34 35 45 31 34 33 45 42 44 30 37 39 37 43 38 33 46 37 41 44 38 38 44 32 35 35 46 31 39 31 35 33 43 38 30 42 31 35 39 42 45 41 34 46 35 45 36 30 34 45 46 41 39 38 44 30 39 31 39 33 46 42 39 42 45 34 46 45 39 32 32 42 43 31 44 31 42 46 44 39 37 39 31 45 37 37 36 34 43 36 32 35 41 45 33 45 38 35 42 44 43 43 38 39 30 33 42 44 | OPN0005152564E58 05E5ACF3BE4A3FD2 7300E682DDB028DC 43223D6D2E59DB12 BCB52D238D1875CF 19A69FE4052B7FED 0160ADF3006D8D66 156AA1A0A55E2F1A 045322FED5945B2F 4A7A6E66C8D2FA97 479D31B106FEC1A5 9370184AC6373B10 53DA9BE7DC01B2AA 18C208E1C0779C0C CD7D4496E5326E98 AE744CCC58CA7B46 30D96DD37FB7B79D 6FBA793018C92C61 516939CCA12D1924 1461605D589082BB 5ED79E5AE720C9DC C07250FEE527D18A D8AB3749E2EE0D48 D9BC2E0AED75AD74 9E11A379C37B6840 1048AD79DE254E0B 45E143EBD0797C83 F7AD88D255F19153 C80B159BEA4F5E60 4EFA98D09193FB9B E4FE922BC1D1BFD9 791E7764C625AE3E 85BDCC8903BD |
|------|---|--|

Para efeito de validação, este exemplo considera o seguinte valor para o expoente privado:

| | |
|-------------|--|
| $K_{PRV} =$ | 40 AD D8 7A 79 A5 F9 8D 26 2C BD E2 60 0A 00 1F 79 FA 15 0D 68 2C 8C 7D 59 C9 4B 89 BF C5 12 22 7B 53 6A 97 31 3E 8F BD 2F 47 B5 F7 8F 66 F2 7B E7 8E BC BE 55 8F 7D 88 58 7C E5 BD F2 15 D3 CD 63 AD 4B 0E BC 1C 44 6E 95 32 5F 87 DC F1 B0 37 DE 4B 39 77 FD 38 8C 4E 77 C0 5D 99 03 CF 18 AA 9B 6C 5D 28 DB C5 A3 69 3E 4C AA EE 27 8D D8 EE 0E E5 97 41 CC 06 8C 9C 74 98 70 2F 32 A6 87 67 6B A0 D1 02 AD F1 70 45 5D E2 6B 71 6E 0A C1 CA 13 93 71 D0 B5 27 5F 0B 93 F7 07 9F 2F 9C F0 1D 21 D6 C0 D4 1E 21 2E 20 FE 40 C1 E3 AF AF 73 47 3F 5B 7C 16 79 01 A9 5B 49 44 80 4E DC D6 8D 4C A4 E2 C5 D3 3C BF 88 AC 42 71 2C ED 32 47 9A 03 6B 48 9F 38 23 D8 B8 63 FA 9C EB 9E 5A 4C ED AB AD 25 19 11 D4 F9 20 D1 5D 72 B5 47 A0 AD 21 27 6E 9C FD 79 F8 7B 83 0C 32 B7 65 05 68 D8 EB D5 |
|-------------|--|

Utilizando-se a chave RSA com o **K_{PRV}** definido acima, obtém-se o seguinte bloco de dados ao se “abrir” o **CRKSEC**:

| | | | |
|---------------|--|---|--|
| CRKSEC | <u>00 02</u> | FA 6D BD 58 30 43 21 4C A1 BA EA EA 54 F2 | |
| aberto = | DB 72 2E 7F 96 41 89 7D C7 57 DB 31 6C 79 88 07 | | |
| | C1 27 AA 16 88 6D 4E 31 0A CC 97 1B 0B 2D 1F 22 | | |
| | 60 DD B1 E7 15 17 AC 33 5F FB CD B3 16 C7 98 80 | | |
| | 7B 78 BE 8B 96 BE 37 97 A0 3C BD 23 C8 7A 92 CD | | |
| | 26 BD C7 37 E3 8C 39 4C 96 D9 70 96 75 B1 FA 7C | | |
| | 49 2E E2 23 B7 1D BD 63 6E 87 FE A8 C0 46 F4 9C | | |
| | F9 B4 45 FA 57 FA 6D BD 58 30 43 21 4C A1 BA EA | | |
| | EA 54 F2 DB 72 2E 7F 96 41 89 7D C7 57 DB 31 6C | | |
| | 79 88 07 C1 27 AA 16 88 6D 4E 31 0A CC 97 1B 0B | | |
| | 2D 1F 22 60 DD B1 E7 15 17 AC 33 5F FB CB 78 BE | | |
| | 8B 96 BE 37 97 A0 3C BD 23 C8 7A 92 CD 26 BD C7 | K_{SEC} = | |
| | 37 E3 8C 39 4C 96 D9 70 96 75 B1 FA 7C 49 2E E2 | DB 3B 4D 01 | |
| | 23 B7 1D BD 63 6E 87 FE A8 C0 46 F4 9C F9 B4 45 | 54 32 AB 32 | |
| | FA 57 6E 87 FE A8 C0 46 F4 9C F9 B4 45 FA 57 00 | 23 55 5A 1F | |
| | <u>DB 3B 4D 01 54 32 AB 32 23 55 5A 1F 81 75 9A 94</u> | 81 75 9A 94 | |

3.2.3. Comando “GIN”

Obsoleto
 Blocante
 ABECS

Este comando obtém informações gerais sobre o pinpad e seu *software/firmware*. Caso uma informação não exista ou não se aplique para o modelo de pinpad, ela é fornecida em branco (espaços).

⚠ Este comando é **obsoleto**. O SPE deve usar o comando “GIX” para esta funcionalidade.

➤ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= “GIN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “002”). |
| GIN_ACQIDX | N2 | Índice da Rede Credenciadora. Se diferente de “00”, são requeridas informações sobre o <i>software/firmware</i> responsável pelo processamento da Rede Credenciadora de índice GIN_ACQIDX . Se “00”, são requeridas informações <u>gerais</u> sobre o pinpad. |

➤ Resposta (para GIN_ACQIDX = “00”)

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “GIN”). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “100”). |
| GIN_MNAME | A20 | Nome do fabricante do pinpad. |
| GIN_MODEL | A19 | Modelo / versão do <i>hardware</i> , no formato “ xx...xx;m...m ”, onde: <ul style="list-style-type: none"> ▪ “xx...xx” é o nome do equipamento; e ▪ “m...m” a capacidade de memória (“512KB”, “1MB”, “2MB”, ...). |
| GIN_CTLSSUP | A1 | Se o pinpad suporta CTLS, este campo deve conter a letra “C”, caso contrário um espaço em branco. |
| GIN_SOVER | A20 | Versão do <i>software</i> básico ou sistema operacional (formato livre). |
| GIN_SPECVER | A4 | Versão da especificação, no formato “V.VV” (neste caso, fixo “2.20”) |
| GIN_MANVER | A16 | Versão da aplicação “Gerenciadora”, no formato “VVV.VV AAMMDD”. |
| GIN_SERNUM | A20 | Número de série do pinpad (formato livre). |

➤ Resposta (para GIN_ACQIDX = "02")

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= "GIN"). |
| RSP_STAT | N3 | Ver seção 3.1.1. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo "042"). |
| GIN_ACQNAM | A8 | Nome da Rede Credenciadora (fixo "Abecs") |
| GIN_KRNLVER | A12 | Versão do Kernel EMV para ICC. |
| GIN_APPVERS | A13 | Versão da aplicação "Abecs", no formato "VVV.VV AAMMDD". |
| GIN_SPECVER | A4 | Versão da especificação, no formato "V.VV" (neste caso, fixo "2.20") |
| GIN_RUF1 | A3 | RUF (preenchido com espaços) |
| GIN_RUF2 | N2 | RUF (fixo "00") |

➤ Resposta (para GIN_ACQIDX = "03")

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= "GIN"). |
| RSP_STAT | N3 | Ver seção 3.1.1. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo "042"). |
| GIN_ACQNAM | A6 | Nome da Rede Credenciadora (fixo "Abecs") |
| GIN_KRNLVER | A4 | Versão do Kernel EMV para ICC. |
| GIN_CTLsver | A4 | Versão principal do Kernel EMV para CTLS. |
| GIN_MCTLSVER | A3 | Versão do Kernel EMV CTLS para cartões Mastercard PayPass. |
| GIN_VCTLSVER | A3 | Versão do Kernel EMV CTLS para cartões VISA PayWaye. |
| GIN_APPVERS | A13 | Versão da aplicação "Abecs", no formato "VVV.VV AAMMDD". |
| GIN_SPECVER | A4 | Versão da especificação, no formato "V.VV" (neste caso, fixo "2.20") |
| GIN_RUF3 | A2 | RUF (preenchido com espaços) |
| GIN_DUKPT | A1 | Identificador da presença de chave DUKPT:TDES:PIN na posição "01": "T" = Chave presente; ou " " (espaço em branco) = Chave ausente. |
| GIN_RUF2 | N2 | RUF (fixo "00") |

3.2.4. Comando “GIX”

Obsoleto
 Blocante
 ABECS

Este comando obtém informações gerais sobre o pinpad e seu *software/firmware*, bem como as chaves nele carregadas. Caso uma informação não exista ou não se aplique para o modelo de pinpad, ela é não é devolvida.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|---|
| CMD_ID | M | Código do comando (= “GIX”). |
| SPE_IDLIST | O | Lista dos identificadores dos dados a serem retornados pelo pinpad, podendo incluir qualquer um dos identificadores listados na resposta. Caso este campo não seja fornecido, o pinpad considerará <u>todos</u> os objetos identificados com “(+)” na tabela da seção 3.1.3.2 . |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------------------------|----------|--|
| RSP_ID | M | Código da resposta (= “GIX”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↳ST_RSPOVRF..... Tamanho da resposta ultrapassa máximo permitido pelo protocolo. |
| PP_SERNUM⁽⁺⁾ | MR | |
| PP_PARTNBR⁽⁺⁾ | O | Se informação suportada pelo pinpad. |
| PP_MODEL⁽⁺⁾ | MR | |
| PP_MNNAME⁽⁺⁾ | MR | |
| PP_CAPAB⁽⁺⁾ | MR | |
| PP_SOVER⁽⁺⁾ | MR | |
| PP_SPECVER⁽⁺⁾ | MR | |
| PP_MANVERS⁽⁺⁾ | MR | |
| PP_APPVERS⁽⁺⁾ | MR | |
| PP_GENVERS⁽⁺⁾ | MR | |
| PP_CTLSCAPAB⁽⁺⁾ | MR | Somente se o pinpad suportar cartões sem contato. |
| PP_KRNLVER⁽⁺⁾ | MR | |
| PP_CTLSVER⁽⁺⁾ | MD | Somente se o pinpad suportar cartões sem contato. |
| PP_MCTLSVER⁽⁺⁾ | MD | Somente se o pinpad suportar cartões sem contato. |

| Id. do Campo | Presença | Descrição / Observação |
|--|---------------|---|
| <u>PP_VCTLSVER</u> ^(†) | MD | Somente se o pinpad suportar cartões sem contato. |
| <u>PP_AECTLSVER</u> ^(†) | MD | Somente se o pinpad suportar cartões sem contato. |
| <u>PP_DPCTLSVER</u> ^(†) | MD | Somente se o pinpad suportar cartões sem contato. |
| <u>PP_PUREVER</u>^(‡) | MD | Somente se o pinpad suportar cartões sem contato. |
| <u>PP_QPCTLSVER</u> ^(†) | MD | Somente se o pinpad suportar cartões sem contato. |
| <u>PP_DSPTXTSZ</u> ^(†) | MR | |
| <u>PP_DSPGRSZ</u> ^(†) | O | Somente se o pinpad possui <i>display</i> gráfico. |
| <u>PP_MFSUP</u> ^(†) | O | Somente se o pinpad suportar o comando “ <u>DSI</u> ”. |
| <u>PP_MKTDESP</u> ^(†) | MR | |
| <u>PP_MKTDESD</u> ^(†) | MR | |
| <u>PP_DKPTTDESP</u> ^(†) | MR | |
| <u>PP_DKPTTDESD</u> ^(†) | MR | |
| <u>PP_TLRMEM</u> ^(†) | MR | |
| <u>PP_BATTINFO</u> | MD | Somente para pinpads móveis com bateria. |
| <u>PP_COMMINFO</u> | MR | |
| <u>PP_KSNTDESPnn</u> | O | Somente se o pinpad possuir carregada a chave DUKPT:TDES:PIN, índice #nn. |
| <u>PP_KSNTDESDnn</u> | O | Somente se o pinpad possuir carregada a chave DUKPT:TDES:DAT, índice #nn. |
| <u>PP_TABVERnn</u> | O | Valor de acordo com regras definidas para o comando “ <u>GTS</u> ” (ver seção 3.5.1). |
| <u>PP_BIGRAND</u> | MR | É usado somente para testes de protocolo. |

➤ Exemplos

SPE solicita as informações PP_SERNUM, PP_MNNAME, PP_DKPTTDESP, PP_KSNTDESP01 e PP_KSNTDESP14.

| | | |
|-------|--|-----------------------------|
| SPE ⇒ | 47 49 58 30 31 34 00 01 00 0A 80 01 80 04 80 34 91 01 91 0E | GIX014....€.€.€4 , , , , |
|-------|--|-----------------------------|

Pinpad devolve as informações, porém não devolve o KSN do DUKPT:TDES:PIN #14, dado que esta chave não está carregada.

| | | |
|-------------|---|------------------|
| ← PP | 47 49 58 30 30 30 31 35 31 80 01 00 0C 39 39 31 | GIX000151€...991 |
| | 32 37 34 33 36 36 31 35 35 80 04 00 0D 48 45 4D | 274366155€...HEM |
| | 49 53 50 48 45 52 45 53 20 20 80 34 00 64 30 31 | ISPHERES••€4.d01 |
| | 31 31 30 30 31 31 30 30 30 30 30 30 30 30 30 30 | 1100110000000000 |
| | 30 30 30 30 30 30 30 30 32 32 32 32 32 32 32 32 | 0000000222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 2222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 2222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 2222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 2222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 2222222222222222 |
| | 32 32 91 01 00 0A FF FF F9 13 25 00 43 20 04 43 | 22'...ÿÿù.%.C .C |

SPE envia o comando sem parâmetros.

| | | |
|--------------|----------|-----|
| SPE → | 47 49 58 | GIX |
|--------------|----------|-----|

3.2.5. Comando “DWK”

Obsoleto
 Blocante
 ABECS

Este comando permite que o SPE habilite o modo “PAN Criptografado”, evitando que o número do cartão circule em claro pelo protocolo serial do pinpad, conforme processo detalhado na **seção 5.3**.

Este comando estabelece a chave (**WK_{PAN}**) a ser utilizada no processo e pode ser chamado a qualquer momento depois de execução bem-sucedida de “**OPN**”. A partir desse instante o pinpad passa a trabalhar no modo “PAN Criptografado”, até que seja chamado o comando “**CLO/CLX**”.

- ⚠ O modo “PAN Criptografado” é **obsoleto** e foi substituído pelo método “Comunicação Segura”, descrito na **seção 5.2**. Ele só deve ser utilizado pelo SPE caso este identifique que pinpad ainda não respeite esta especificação.
- ⚠ O modo “PAN Criptografado” não é aceito pelo pinpad se o SPE estiver utilizando o método “Comunicação Segura” descrito na **seção 5.2**.

➡ Comando (Modalidade 1)

| Id. do Campo | Formato | Descrição |
|-------------------|---------|--|
| CMD_ID | A3 | Código do comando (= “ DWK ”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “036”). |
| DWK_MODE | N1 | Modalidade: “1” = WK_{PAN} externa criptografada por MK. |
| DWK_METHOD | N1 | Método de criptografia: “1” = MK/WK:TDES:DAT |
| DWK_MKIDX | N2 | Índice da MK a ser utilizada. |
| DWK_WKPAN | H32 | WK_{PAN} criptografada pela MK. |

➡ Resposta (Modalidade 1)

| Id. do Campo | Formato | Descrição |
|-----------------|---------|--|
| RSP_ID | A3 | Código da resposta (= “ DWK ”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_ERRKEY MK não está presente no pinpad. ↳ ST_INVPARM Índice fornecido (DWK_MKIDX) está fora da faixa usada pelo pinpad. ↳ ST_INVCALL Pinpad está em modo de “Comunicação Segura”. |

➔ Comando (Modalidade 2)

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= "DWK"). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo "263"). |
| DWK_MODE | N1 | Modalidade: "2" = WK _{PAN} TDES gerada aleatoriamente pelo pinpad. |
| DWK_RSAMOD | H256 | Módulo da chave pública RSA criada pelo SPE (K _{MOD} - fixo 128 bytes / 1024 bits). IMPORTANTE: O primeiro byte do módulo da chave deve ser maior do que 54h, devido ao formato do bloco de dados (ver seção 5.3.3). |
| DWK_RSAEXP | H6 | Expoente da chave pública RSA criada pelo SPE (K _{PUB} - tipicamente "000003" ou "010001"). |

➔ Resposta (Modalidade 2)

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= "DWK"). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_INVCALL Pinpad está em modo de "Comunicação Segura". |
| RSP_LEN | N3 | Tamanho dos dados a seguir (fixo "256"). |
| DWK_CRYPT | H256 | Criptograma RSA contendo uma chave WK _{PAN} aleatória, conforme definido na seção 5.3. |

➔ Exemplos

O SPE inicia o modo "PAN Criptografado" na modalidade 2, fornecendo uma chave pública RSA.

| | | |
|-------|--|--|
| SPE ➔ | <pre> 44 57 4B 32 36 33 32 43 30 45 34 45 36 41 41 44 39 44 43 38 31 45 32 45 42 46 38 41 43 31 32 36 45 37 45 45 45 36 35 36 38 30 38 39 38 42 42 41 43 33 30 30 36 33 44 43 44 35 34 33 44 37 30 35 30 34 30 45 39 31 36 44 39 33 45 45 33 31 36 42 39 45 43 34 39 32 42 37 39 36 46 31 37 32 31 34 32 35 46 30 46 30 32 38 38 33 34 32 35 31 41 41 44 35 31 43 45 42 31 37 38 33 33 30 38 45 43 37 44 35 30 37 32 44 38 34 38 31 33 42 44 41 35 39 42 33 31 36 31 43 42 34 38 37 39 34 36 34 45 42 35 41 46 37 31 39 36 39 38 36 35 46 44 33 34 37 34 35 41 37 31 31 44 31 44 41 33 44 44 42 34 44 32 39 44 32 39 44 30 34 32 32 43 36 45 31 37 43 32 35 46 31 37 43 30 42 35 42 33 39 45 36 38 38 43 34 44 30 36 31 32 33 44 44 42 35 46 35 35 38 45 46 30 33 31 36 42 33 46 37 34 34 43 37 30 37 31 46 32 39 37 39 31 30 31 30 30 31 </pre> | <pre> DWK2632C0E4E6AAD 9DC81E2EBF8AC126 E7EEE65680898BBA C30063DCD543D705 040E916D93EE316B 9EC492B796F17214 25F0F028834251AA D51CEB1783308EC7 D5072D84813BDA59 B3161CB4879464EB 5AF71969865FD347 45A711D1DA3DDB4D 29D29D0422C6E17C 25F17C0B5B39E688 C4D06123DDB5F558 EF0316B3F744C707 1F29791010001 </pre> |
|-------|--|--|

O pinpad gera uma chave **WK_{PAN}** aleatória (2A525553482A43524F4E49434C45532A) e a devolve criptografada pela chave pública fornecida.

| | | |
|-------------|---|------------------|
| ← PP | 44 57 4B 30 30 30 32 35 36 42 37 45 30 42 37 38 | DWK000256B7E0B78 |
| | 41 39 34 42 30 32 42 34 38 30 32 32 38 43 39 33 | A94B02B480228C93 |
| | 44 35 42 39 31 31 42 41 33 38 33 37 35 33 38 41 | D5B911BA3837538A |
| | 45 38 41 42 45 46 44 46 38 38 41 41 46 30 42 46 | E8ABEFDF88AAF0BF |
| | 36 46 33 34 38 35 34 39 31 30 30 41 38 34 30 45 | 6F348549100A840E |
| | 35 38 30 41 41 46 36 35 31 46 33 35 34 44 33 39 | 580AAF651F354D39 |
| | 31 39 32 43 30 30 38 33 36 44 39 30 35 32 32 46 | 192C00836D90522F |
| | 44 34 35 32 38 39 46 32 35 42 43 43 33 41 31 30 | D45289F25BCC3A10 |
| | 45 41 43 35 35 35 32 31 46 35 35 30 37 34 41 37 | EAC55521F55074A7 |
| | 38 37 34 34 39 42 38 34 42 43 36 44 42 32 31 39 | 87449B84BC6DB219 |
| | 32 39 44 37 34 33 32 45 38 33 36 45 44 41 30 39 | 29D7432E836EDA09 |
| | 46 46 41 41 32 30 42 33 39 43 45 44 36 38 37 42 | FFAA20B39CED687B |
| | 37 35 37 39 45 36 31 46 30 44 30 35 39 45 35 32 | 7579E61F0D059E52 |
| | 33 42 38 41 35 42 41 43 36 31 45 46 39 41 30 46 | 3B8A5BAC61EF9A0F |
| | 41 32 39 37 32 38 30 41 32 31 41 41 38 44 34 34 | A297280A21AA8D44 |
| | 35 42 32 42 45 35 42 45 34 34 35 44 41 38 39 30 | 5B2BE5BE445DA890 |
| | 41 43 36 42 41 37 39 30 30 | AC6BA7900 |

Para efeito de validação, este exemplo considera o seguinte valor para o expoente privado:

| | |
|--------------------------|---|
| K_{PRV} = | 65 3C BD C3 95 AC 21 8F 53 81 A3 ED D8 88 4D DE |
| | 73 07 70 01 AF 91 54 F5 42 BA 9F B4 3E AA 92 AB |
| | 27 41 D6 35 AB 46 D3 F0 39 3F 90 C8 27 E9 74 1B |
| | 44 18 FA 10 52 3E C9 58 63 59 85 A9 78 EB AC 19 |
| | E4 25 CE 7F 6B 78 66 7E 9C C1 85 C8 1A 0B F2 FF |
| | A7 4A CC 33 FF A3 6F DB 95 66 80 12 FF 32 4E BD |
| | 58 04 60 C3 2D 76 61 8B E8 16 98 61 F5 33 2B 83 |
| | 5C FC 31 1F 7C C5 41 65 87 0D 78 9D 6B 72 68 F1 |

3.2.6. Comando “CLO”

Obsoleto
 Blocante
 ABECS

Este comando libera os recursos de *hardware* e *software* alocados pelo pinpad, além de finalizar os processos de “Comunicação Segura” ou “PAN Criptografado”.

É recomendável que o SPE use este comando ao final do processamento de uma transação.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| CMD_ID | A3 | Código do comando (= “CLO”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “032”). |
| CLO_MSG | S32 | Mensagem de 32 caracteres a ser apresentada no <i>display</i> do pinpad após a execução do comando, já formatada corretamente para 2 linhas e 16 colunas. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|-----------------------------------|
| RSP_ID | A3 | Código da resposta (= “CLO”). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |

➔ Exemplos

SPE solicita “fechamento” do pinpad, deixando a mensagem “POSTO FORÇA 10”/“OBRIGADO!!” no *display*.

| | | |
|-------|---|--|
| SPE ⇒ | 43 4C 4F 30 33 32 50 4F 53 54 4F 20 46 4F 52 C7 41 20 31 30 20 20 4F 42 52 49 47 41 44 4F 21 21 21 20 20 20 20 20 | CLO032POSTO•FORÇ A•10••OBRIGADO!! !••••• |
|-------|---|--|

A operação é bem-sucedida.

| | | |
|------|-------------------|--------|
| ⇐ PP | 43 4C 4F 30 30 30 | CLO000 |
|------|-------------------|--------|

3.2.7. Comando “CLX”

Obsoleto
 Blocante
 ABECS

Este comando é equivalente ao comando “**CLO**”, porém utiliza uma mensagem de formato livre (permitindo o uso de todos os recursos do *display* do equipamento) ou permite a apresentação de um arquivo multimídia (se suportado).

Este comando sempre retorna imediatamente (é não blocante), mesmo se o arquivo multimídia informado contiver animação (ou vídeo), que será apresentada enquanto o pinpad não recebe um novo comando.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|--|
| CMD_ID | M | Código do comando (= “ CLX ”). |
| SPE_DSPMSG | O | Mensagem a ser deixada no <i>display</i> do pinpad após a execução do comando. |
| SPE_MFNAME | O | Nome do arquivo multimídia a ser deixado no <i>display</i> do pinpad após a execução do comando. |

OBSERVAÇÕES:

- ⇒ Se nenhum parâmetro for fornecido, o conteúdo do *display* é simplesmente apagado.
- ⇒ **SPE_MFNAME** tem prioridade sobre **SPE_DSPMSG**, ou seja, se **SPE_MFNAME** for fornecido e o arquivo multimídia informado existir, **SPE_DSPMSG** é desprezado.

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------|----------|--|
| RSP_ID | M | Código da resposta (= “ CLX ”). |
| RSP_STAT | M | Ver seção 3.1.1 . |

➔ Exemplos

SPE solicita “fechamento” do pinpad, deixando uma mensagem de três linhas no *display*.

| | | |
|----------------------------|---|---|
| SPE ⇒ | 43 4C 58 30 34 30 00 1B 00 24 50 52 45 53 54 4F 20 53 48 4F 50 0D 4F 42 52 49 47 41 44 4F 20 45 0D 56 4F 4C 54 45 20 53 45 4D 50 52 45 21 | CLX040... (PRESTO •SHOP.OBRIGADO•E .VOLTE•SEMPRE! |
| A operação é bem-sucedida. | | |
| ← PP | 43 4C 58 30 30 30 | CLX000 |

3.3. Comandos básicos

Chamamos aqui de “comandos básicos” aqueles destinados ao acesso simples aos periféricos e recursos do pinpad.

Os seguintes comandos estão contemplados nesta seção:

| CMD_ID | Significado | Obsoleto | Blocante | Abecs |
|---------------|-----------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| “CEX” | <i>Check Event - Extended</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| “CHP” | <i>Chip Direct Access</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| “CKE” | <i>Check Event</i> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| “DEX” | <i>Display Message - Extended</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| “DSP” | <i>Display Message</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| “EBX” | <i>Encrypt Buffer - Extended</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| “ENB” | <i>Encrypt Buffer</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| “GCD” | <i>Get Clear Data</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| “GDU” | <i>Get DUKPT Serial Number</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| “GKY” | <i>Get Key</i> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| “GPN” | <i>Get Encrypted PIN</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| “GTK” | <i>Get Tracks</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| “MNU” | <i>Prompt Menu</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| “RMC” | <i>Remove Card</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |

3.3.1. Comando “CEX”

| |
|--|
| <input type="checkbox"/> Obsoleto |
| <input checked="" type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando aguarda a ocorrência de um determinado evento no pinpad. Os seguintes eventos podem ser verificados:

- Pressionamento de uma tecla (não numérica);
- Passagem de um cartão magnético;
- Inserção/remoção de um ICC; e
- Aproximação de um CTLS.

No caso da passagem de um cartão magnético, as trilhas são devolvidas incompletas, conforme processo de segurança descrito na **seção 5.4**. Para obter as trilhas completas (abertas ou criptografadas), deve-se usar o comando “**GTK**”.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| CMD_ID | M | Código do comando (= “CEX”). |
| SPE_CEXOPT | M | Evento a ser verificado pelo pinpad: “0xxxxx” = Ignora teclas; “1xxxxx” = Verifica pressionamento de tecla. “x0xxxx” = Ignora cartão magnético; “x1xxxx” = Verifica passagem de cartão magnético. “xx0xxx” = Ignora ICC; “xx1xxx” = Verifica inserção de ICC; “xx2xxx” = Verifica remoção de ICC. “xxx0xx” = Ignora CTLS (não ativa a antena); “xxx1xx” = Ativa a antena e verifica a presença de um CTLS. “xxxx00” = RUF. |
| SPE_TIMEOUT | O | Tempo máximo para aguardar um evento. |
| SPE_PANMASK | O | Definições para mascaramento do PAN nos campos de resposta PP_TRK1INC , PP_TRK2INC e PP_TRK3INC . Se ausente, não há mascaramento. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------|----------|--|
| RSP_ID | M | Código da resposta (= “CEX”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_TIMEOUT Esgotado tempo definido por SPE_TIMEOUT . |

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|--|
| <u>PP_EVENT</u> | M | Identificação do evento ocorrido: “00” = Tecla [OK/ENTRA] foi pressionada; “02” = Tecla [↑] foi pressionada; “03” = Tecla [↓] foi pressionada; “04” = Tecla [F1] foi pressionada; “05” = Tecla [F2] foi pressionada; “06” = Tecla [F3] foi pressionada; “07” = Tecla [F4] foi pressionada; “08” = Tecla [LIMPA] foi pressionada; “13” = Tecla [CANCELA] foi pressionada; “90” = Um cartão magnético foi passado no leitor; “91” = ICC foi removido (ou já não estava presente); “92” = ICC foi inserido (ou já estava presente); “93” = CTLS não foi detectado em 2 (dois) minutos; e “94” = CTLS foi detectado. |
| <u>PP_TRK1INC</u> | O | Trilha 1 <u>incompleta</u> , se lida do cartão magnético, podendo ter o PAN mascarado de acordo com <u>SPE_PANMASK</u> . |
| <u>PP_TRK2INC</u> | O | Trilha 2 <u>incompleta</u> , se lida do cartão magnético, podendo ter o PAN mascarado de acordo com <u>SPE_PANMASK</u> . |
| <u>PP_TRK3INC</u> | O | Trilha 3 <u>incompleta</u> , se lida do cartão magnético, podendo ter o PAN mascarado de acordo com <u>SPE_PANMASK</u> . |

⚠ Caso um cartão magnético tenha sido passado (PP_EVENT = “90”), mas nenhuma trilha pôde ser lida, RSP_STAT = ST_OK e os campos PP_TRK1INC, PP_TRK2INC e PP_TRK3INC não serão devolvidos. Isso difere do comportamento do comando “CKE”, que retorna RSP_STAT = ST_MCDATAERR neste caso.

➡ Exemplos

SPE solicita somente o evento de passagem de cartão magnético.

| | | |
|--------------|---|------------------|
| <u>SPE</u> ⇒ | 43 45 58 30 31 30 00 06 00 06 30 31 30 30 30 30 | CEX010....010000 |
|--------------|---|------------------|

Pinpad acusa passagem de cartão, mas somente a trilha 2 é lida.

| | | |
|-------------|--|---|
| ⇐ <u>PP</u> | 43 45 58 30 30 30 30 33 34 80 40 00 02 39 30 80 42 00 18 34 33 31 33 30 33 32 39 32 39 38 33 30 30 31 31 3D 31 35 30 38 36 30 31 | CEX000034€@..90€ B..4313032929830 011=1508601 |
|-------------|--|---|

3.3.2. Comando “CHP”

Obsoleto
 Blocante
 ABECS

Este comando possibilita o acesso direto a um ICC (principal ou SAM) bem como a um CTLS.

Adicionalmente, este comando possibilita a captura de um PIN para verificação direta no cartão, seja qual for a tecnologia (ICC, SAM ou CTLS).

➔ Comando

| Id. do Campo | Formato | Descrição |
|-------------------|---------|---|
| CMD_ID | A3 | Código do comando (= “CHP”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| CHP_SLOT | N1 | Identificação do cartão a ser usado: “0” = ICC no acoplador principal; “1” = SAM na posição #1; ... “8” = SAM na posição #8; e “9” = CTLS. |
| CHP_OPER | N1 | Operação a ser executada: “0” = Desligar o cartão (para CTLS, desativa-se a antena); “1” = Ligar o cartão (para CTLS, ativa-se primeiro a antena e, depois, o cartão); “2” = Trocar comando com o cartão; e “3” = Verificar PIN diretamente no cartão. |
| CHP_CMDLEN | N3 | Quantidade de bytes representados em CHP_CMD (tamanho ÷ 2). Para CHP_OPER = “0” ou “1”, CHP_CMDLEN é “000”. |
| CHP_CMD | H..497 | Comando a ser enviado ao cartão. Se CHP_OPER = “2”, aceitam-se seguintes formatos de C-APDU: CLA INS P1 P2 CLA INS P1 P2 Le CLA INS P1 P2 Lc XX XX ... XX CLA INS P1 P2 Lc XX XX ... XX Le Para CTLS, ainda se pode usar “C-APDU estendido”, nos seguintes formatos: CLA INS P1 P2 00 Lc+ Lc- XX XX ... XX CLA INS P1 P2 00 Lc+ Lc- XX XX ... XX Le Se CHP_OPER = “3”, deve-se fornecer somente os quatro primeiros bytes do comando a ser enviado ao cartão (CLA INS P1 P2), pois o resto é montado automaticamente conforme formato do “pinblock” (CHP_PINFMT). |

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| CHP_PINFMT | N1 | Formato do “pinblock” (somente se CHP_OPER = “3”): “0” = 0Th PPh PPh ... FFh (8 bytes, PIN 4 a 12 dígitos); “1” = 2Th PPh PPh ... FFh (8 bytes, PIN 4 a 12 dígitos); “2” = PPh PPh PPh ... FFh (8 bytes, PIN 4 a 12 dígitos); e “9” = Sequência de dígitos ASCII (tamanho variável). |
| CHP_PINMSG | S32 | Mensagem de 2 linhas por 16 colunas para apresentação no momento do pedido do PIN (somente se CHP_OPER = “3”). |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= “CHP”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_CANCEL Portador pressionou a tecla [CANCELA]. ↪ ST_NOCARD Não há cartão presente no acoplador ou antena. ↪ ST_DUMBCARD ICC inserido, mas não responde (“mudo”) (não se aplica a CTLS). ↪ ST_ERRCARD Erro de comunicação entre o pinpad e o ICC. ↪ ST_CTLSMULTIPLE Mais de um CTLS foi apresentado ao leitor simultaneamente. ↪ ST_CTLSCOMMERR Erro de comunicação entre o pinpad (antena) e o CTLS. ↪ ST_TIMEOUT Tempo esgotado para captura de PIN (CHP_OPER = “3”). |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| CHP_RSPLEN | N3 | Quantidade de bytes representados em CHP_RSP (tamanho ÷ 2). |
| CHP_RSP | H..514 | Resposta do cartão: Se CHP_OPER = “0”, não há (CHP_RSPLEN é sempre “000”). Se CHP_OPER = “1”, é o ATR completo do cartão. Se CHP_OPER = “2” ou “3”, é a resposta ao comando enviado, seguida obrigatoriamente pelos bytes SW1 e SW2. |

➔ Observações

- O pinpad **não** tratará internamente os status de retorno 61xxh e 6Cxxh dos cartões T=0. Dessa forma, o SPE terá que estar preparado para tratar externamente esses dois casos.
- O SPE deve sempre desativar a antena ao finalizar o processamento de um CTLS.

➤ Exemplos

O SPE solicita a ativação do ICC no acoplador principal.

| | | |
|--------------|----------------------------------|-------------|
| SPE ⇒ | 43 48 50 30 30 35 30 31 30 30 30 | CHP00501000 |
|--------------|----------------------------------|-------------|

A operação é bem-sucedida e o pinpad devolve o ATR do cartão (3B29008072A4456400FF0010).

| | | |
|-------------|---|--|
| ⇐ PP | 43 48 50 30 30 30 30 32 37 30 31 32 33 42 32 39 30 30 38 30 37 32 41 34 34 35 36 34 30 30 46 46 30 30 31 30 | CHP0000270123B29 008072A4456400FF 0010 |
|-------------|---|--|

O SPE envia o comando de seleção (SELECT) para o AID Mastercard.

| | | |
|--------------|---|---|
| SPE ⇒ | 43 48 50 30 32 39 30 32 30 31 32 30 30 41 34 30 34 30 30 30 37 41 30 30 30 30 30 30 30 34 31 30 31 30 | CHP0290201200A40 40007A0000000041 010 |
|--------------|---|---|

A operação é bem-sucedida, sendo que o cartão devolve os bytes de status 6132h.

| | | |
|-------------|---|------------------|
| ⇐ PP | 43 48 50 30 30 30 30 30 37 30 30 32 36 31 33 32 | CHP0000070026132 |
|-------------|---|------------------|

Dado que cartão devolveu 61xxh (protocolo T=0), o SPE envia o comando GET RESPONSE.

| | | |
|--------------|---|---------------------------|
| SPE ⇒ | 43 48 50 30 31 35 30 32 30 30 35 30 30 43 30 30 30 30 30 33 32 | CHP0150200500C00 00032 |
|--------------|---|---------------------------|

A operação é bem-sucedida, sendo que o cartão devolve a resposta ao comando SELECT.

| | | |
|-------------|--|--|
| ⇐ PP | 43 48 50 30 30 30 31 30 37 30 35 32 36 46 33 30 38 34 30 37 41 30 30 30 30 30 30 30 30 34 31 30 31 30 41 35 32 35 35 30 30 41 34 44 36 31 37 33 37 34 36 35 37 32 34 33 36 31 37 32 36 34 38 37 30 31 30 31 35 46 32 44 30 36 37 30 37 34 36 35 36 45 36 35 37 33 39 46 31 31 30 31 30 31 39 46 31 32 30 36 34 33 37 32 36 35 36 34 36 39 37 34 39 30 30 30 | CHP0001070526F30 8407A00000000410 10A525500A4D6173 7465724361726487 01015F2D06707465 6E65739F1101019F 1206437265646974 9000 |
|-------------|--|--|

O SPE solicita a verificação do PIN diretamente no cartão (formato "1").

| | | |
|--------------|--|--|
| SPE ⇒ | 43 48 50 30 34 36 30 33 30 30 34 30 30 32 30 30 30 30 30 31 44 49 47 49 54 45 20 41 20 53 45 4E 48 41 20 20 44 4F 20 43 41 52 54 C3 4F 20 49 43 43 2E 2E 2E | CHP0460300400200 0001DIGITE•A•SEN HA••DO•CARTÃO•IC C... |
|--------------|--|--|

O cartão retorna erro 6A86h.

| | | |
|-------------|---|------------------|
| ⇐ PP | 43 48 50 30 30 30 30 30 37 30 30 32 36 41 38 36 | CHP0000070026A86 |
|-------------|---|------------------|

3.3.3. Comando “CKE”

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input checked="" type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

Este comando aguarda a ocorrência de um determinado evento no pinpad. Os seguintes eventos podem ser verificados:

- Pressionamento de uma tecla (não numérica);
- Passagem de um cartão magnético;
- Inserção/remoção de um ICC; e
- Aproximação de um CTLS.

|| ⚠ Este comando é **obsoleto**. O SPE deve usar o comando “CEX” para esta funcionalidade.

➡ Comando

| Id. do Campo | Formato | Descrição |
|-------------------------|---------|---|
| CMD_ID | A3 | Código do comando (= “CKE”) |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (“003” ou “004”) |
| CKE_KEY | N1 | Controla evento de pressionamento de tecla. “0” = Ignora teclas. “1” = Verifica pressionamento de tecla. |
| CKE_MAG | N1 | Controla evento de passagem de cartão magnético. “0” = Ignora cartão magnético. “1” = Verifica passagem de cartão magnético. |
| CKE_ICC | N1 | Controla evento de inserção/remoção de ICC. “0” = Ignora ICC. “1” = Verifica inserção de ICC. “2” = Verifica remoção de ICC. |
| CKE_CTLS (opcional!) | N1 | Controla evento de aproximação de CTLS. “0” = Não ativa a antena. “1” = Ativa a antena e verifica a presença de um CTLS. |

➡ Resposta (se tecla pressionada)

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “CKE”) |
| RSP_STAT | N3 | Ver seção 3.1.1. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “003”) |
| CKE_EVENT | N1 | Identificação do evento ocorrido: “0” |

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| CKE_KEYCODE | N2 | Código da tecla pressionada: “00” = [OK/ENTRA] “04” = [F1] “05” = [F2] “06” = [F3] “07” = [F4] “08” = [LIMPA] “13” = [CANCELA] |

➔ Resposta (se cartão magnético)

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= “CKE”) |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_MCDATAERR Detectado evento de cartão magnético, porém houve erro de leitura (nenhuma trilha pôde ser lida). |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “225”). |
| CKE_EVENT | N1 | Identificação do evento ocorrido: “1” |
| CKE_TRK1LEN | N2 | Tamanho da trilha 1. |
| CKE_TRK1 | A76 | Trilha 1 (sem as sentinelas e com o <i>byte</i> de formato - primeiro caractere alfanumérico), alinhada à esquerda com espaços à direita. |
| CKE_TRK2LEN | N2 | Tamanho da trilha 2. |
| CKE_TRK2 | A37 | Trilha 2 (sem as sentinelas), alinhada à esquerda com espaços à direita. |
| CKE_TRK3LEN | N3 | Tamanho da trilha 3. |
| CKE_TRK3 | A104 | Trilha 3 (sem as sentinelas), alinhada à esquerda com espaços à direita. |

⚠ Se o pinpad estiver em modo “PAN Criptografado” (ver **seção 5.3**), os PAN das trilhas vêm codificados pela chave **WK_{PAN}**.

⚠ Se o pinpad estiver em modo “PAN Criptografado”, **CKE_TRK3LEN** não é preenchido, pois a Trilha 2 pode atingir 40 caracteres (ver explicação na **seção 5.3**)!!

➔ Resposta (se ICC)

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “CKE”) |
| RSP_STAT | N3 | Ver seção 3.1.1 . |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “002”) |
| CKE_EVENT | N1 | Identificação do evento ocorrido: “2” |

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CKE_ICCSTAT | N1 | "0" = ICC ausente; ou "1" = ICC presente. |

➔ Resposta (se CTLS)

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= "CKE") |
| RSP_STAT | N3 | Ver seção 3.1.1. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo "002") |
| CKE_EVENT | N1 | Identificação do evento ocorrido: "3" |
| CKE_CTLSSTAT | N1 | "0" = CTLS não foi detectado em 2 (dois) minutos. "1" = CTLS foi detectado. |

➔ Exemplos

O SPE solicita ao pinpad que aguarde qualquer um dos quatro eventos possíveis.

| | | |
|-------|-------------------------------|------------|
| SPE ⇒ | 43 4B 45 30 30 34 31 31 31 31 | CKE0041111 |
|-------|-------------------------------|------------|

Um cartão magnético é passado no pinpad, que devolve suas trilhas 1 e 2.

| | | |
|------|---|---|
| ⇐ PP | 43 4B 45 30 30 30 32 32 35 31 37 34 42 35 31 34 38 36 38 32 32 32 32 32 32 32 32 37 37 5E 41 4C 45 58 20 4C 49 46 45 53 4F 4E 20 20 20 20 20 20 20 20 20 20 20 20 5E 32 31 31 32 32 30 31 39 38 37 36 30 30 30 30 30 30 30 30 30 30 30 34 34 39 37 30 30 30 30 30 20 20 33 37 35 31 34 38 36 38 32 32 32 32 32 32 32 32 37 37 3D 31 35 30 36 32 30 31 30 30 30 30 39 38 37 36 34 34 39 37 30 30 30 30 | CKE000225174B514 8682222222277^AL EX•LIFESON••••• •••••^211220198 760000000000449 700000••37514868 2222222277=15062 0100009876449700 00 |
|------|---|---|

O SPE solicita ao pinpad que aguarde somente o evento de tecla (não enviando o campo opcional CKE_CTLS).

| | | |
|-------|----------------------------|-----------|
| SPE ⇒ | 43 4B 45 30 30 33 31 30 30 | CKE003100 |
|-------|----------------------------|-----------|

A tecla F1 é pressionada no pinpad.

| | | |
|------|-------------------------------------|--------------|
| ⇐ PP | 43 4B 45 30 30 30 30 30 33 30 30 34 | CKE000003004 |
|------|-------------------------------------|--------------|

3.3.4. Comando “DEX”

Obsoleto
 Blocante
 ABECS

Este comando envia uma mensagem ao *display* do pinpad, de formato livre, permitindo o uso de todos os recursos do *display* do equipamento.

O *display* do pinpad é previamente apagado e mensagens anteriores não são mantidas.

O campo DEX_OPTIONS é opcional e, caso presente, o pinpad usa as informações para gerar uma apresentação mais direcionada usando ícones, cores e som, de forma a melhorar a acessibilidade. A maneira como o pinpad monta a tela fica a cargo do fabricante, não sendo objeto desta especificação.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= “DEX”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| DEX_MSGLEN | N3 | Tamanho de <u>DEX_MSG</u> . |
| DEX_MSG | S..160 | Mensagem a ser apresentada, podendo <u>excepcionalmente</u> conter o caractere de controle CR (ODh) para quebra de linha. |
| DEX_OPTIONS | N6 | <p>Parâmetros para apresentação da mensagem:</p> <p><u>Alinhamento horizontal:</u> “0xxxx” = À esquerda (default); “1xxxx” = À direita; “2xxxx” = Centralizada.</p> <p><u>Alinhamento vertical:</u> “x0xxx” = Superior (default); “x1xxx” = Inferior; “x2xxx” = Centralizada.</p> <p><u>Tipo de mensagem:</u> “xx0xx” = Genérica (default); “xx1xx” = Sucesso; “xx2xx” = Erro; “xx3xx” = Alerta; “xx4xx” = Informativa. “xxx000” = RUF.</p> <p>Este parâmetro é <u>opcional</u>. Se não fornecido, o pinpad assume os valores marcados como “default”.</p> |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|-----------------------------------|
| RSP_ID | A3 | Código da resposta (= "DEX"). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |

➔ Exemplos

SPE envia mensagem de quatro linhas para apresentação no *display*.

| | | |
|-------|---|--|
| SPE ⇒ | 44 45 58 30 33 38 30 33 35 46 65 6C 69 7A 20 4E 61 74 61 6C 0D 65 20 75 6D 0D 50 72 F3 73 70 65 72 6F 0D 41 6E 6F 20 4E 6F 76 6F 21 | DEX038035Feliz•N atal.e•um.Próspe ro.Ano•Novo! |
|-------|---|--|

A operação é bem-sucedida.

| | | |
|------|-------------------|--------|
| ⇐ PP | 44 45 58 30 30 30 | DEX000 |
|------|-------------------|--------|

SPE envia mensagem de erro centralizada na parte superior do *display*.

| | | |
|-------|---|--|
| SPE ⇒ | 44 45 58 30 34 30 30 33 31 4E 41 4F 20 41 55 54 4F 52 49 5A 41 44 41 0D 54 45 4E 54 45 20 4E 4F 56 41 4D 45 4E 54 45 21 32 30 32 30 30 30 | DEX040031NAO•AUT ORIZADA.TENTE•NO VAMENTE!202000 |
|-------|---|--|

A operação é bem-sucedida.

| | | |
|------|-------------------|--------|
| ⇐ PP | 44 45 58 30 30 30 | DEX000 |
|------|-------------------|--------|

3.3.5. Comando “DSP”

Obsoleto
 Blocante
 ABECS

Este comando envia uma mensagem ao *display* do pinpad em um formato fixo de 2 linhas por 16 colunas.

O *display* do pinpad é previamente apagado e mensagens anteriores não são mantidas.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= “DSP”) |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “032”) |
| DSP_MSG | S32 | Mensagem de 32 caracteres a ser apresentada no <i>display</i> do pinpad, já formatada corretamente para 2 linhas e 16 colunas. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|----------------------------------|
| RSP_ID | A3 | Código da resposta (= “DSP”) |
| RSP_STAT | N3 | Ver seção 3.1.1. |

➔ Exemplos

SPE envia a mensagem “ERRO DE OPERAÇÃO”/“CÓDIGO: 2112/76”, corretamente formatada para apresentação no *display* em 2 linhas e 16 colunas.

| | | |
|-------|---|---|
| SPE ⇒ | 44 53 50 30 33 32 45 52 52 4F 20 44 45 20 4F 50 45 52 41 C7 C3 4F 43 D3 44 49 47 4F 3A 20 20 32 31 31 32 2F 37 36 | DSP032ERRODE•OPE RAÇÃOCÓDIGO:••21 12/76 |
|-------|---|---|

A operação é bem-sucedida.

| | | |
|------|-------------------|--------|
| ← PP | 44 53 50 30 30 30 | DSP000 |
|------|-------------------|--------|

3.3.6. Comando “EBX”

Obsoleto
 Blocante
 ABECS

Este comando criptografa um bloco de dados qualquer (de até 256 bytes) utilizando-se uma chave de dados (MK/WK ou DUKPT), em modo ECB ou CBC.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| CMD_ID | M | Código do comando (= “EBX”). |
| SPE_DATAIN | M | Bloco de dados a ser criptografado, obrigatoriamente com tamanho múltiplo de 8 (oito), máximo de 256 bytes. |
| SPE_MTHDDAT | M | Identificação do modo de criptografia a ser utilizado: “10” = MK/WK:TDES:DAT (criptografia de bloco ECB); “11” = MK/WK:TDES:DAT (criptografia de bloco CBC); “50” = DUKPT:TDES:DAT#3 (criptografia de bloco ECB , ver seção 5.1.1); e “51” = DUKPT:TDES:DAT#3 (criptografia de bloco CBC , ver seção 5.1.1). |
| SPE_KEYIDX | M | Índice da chave a ser utilizada (MK:DAT ou DUKPT:DAT). |
| SPE_WKENC | MD | <i>Working Key</i> (criptografada pela MK) a ser usada na criptografia. Este campo é mandatário somente se SPE_MTHDDAT = “0x” ou “1x”. |
| SPE_IVCBC | O | “IV” (<i>Initialization Vector</i>) a ser usado na criptografia, se SPE_MTHDDAT = “x1” (criptografia de bloco CBC). Se ausente, o pinpad considerará o “IV” zerado. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|---|
| RSP_ID | M | Código da resposta (= “EBX”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_ERRKEY Chave não está presente no pinpad. ↪ ST_INVPARM..... Índice fornecido (SPE_KEYIDX) está fora da faixa usada pelo pinpad. ↪ ST_INVPARM..... Tamanho de SPE_DATAIN não é múltiplo de 8 ou é maior do que 256. |
| PP_DATAOUT | M | Dados criptografados (mesmo tamanho de SPE_DATAIN). |
| PP_KSN | MD | KSN da chave usada na criptografia (no caso de método DUKPT). |

➡ Exemplos

SPE solicita a criptografia de um bloco de 24 bytes, contendo a mensagem em ASCII “DADO A SER CRIPTOGRAFADO”, usando a chave DUKPT:TDES de índice “07”, com **variante #3** e modo CBC.

| | | |
|--------------|--|--|
| SPE ⇒ | 45 42 58 30 34 30 00 0F 00 18 44 41 44 4F 20 41 20 53 45 52 20 43 52 49 50 54 4F 47 52 41 46 41 44 4F 00 03 00 02 35 31 00 09 00 02 30 37 | EBX040...DADO•A •SER•CRIPTOGRAFA DO.... 5107 |
|--------------|--|--|

Pinpad devolve dado criptografado, acompanhado do KSN.

| | | |
|-------------|---|---|
| ⇐ PP | 45 42 58 30 30 30 30 34 32 80 4E 00 18 0F 77 0C 3A 6C AF CA 69 5D 00 50 14 41 82 7B A5 2C 21 81 48 C3 5C 94 D1 80 4C 00 0A FF FF F1 23 45 00 88 80 06 C3 | EBX000042€N...w. : ~Êi].P.A,{¥,!. HÃ\”Ñ€L..ÿÿñ#E.^ €.Ă |
|-------------|---|---|

3.3.7. Comando “ENB”

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

Este comando criptografa um bloco de dados qualquer (de 8 bytes) através do método **MK/WK**, utilizando-se obrigatoriamente uma MK de dados.

⚠ Este comando é **obsoleto**. O SPE deve usar o comando “**EBX**” para esta funcionalidade.

➔ Comando

| Id. do Campo | Formato | Descrição |
|-------------------|---------|---|
| CMD_ID | A3 | Código do comando (= “ENB”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “051”). |
| ENB_METHOD | N1 | Método de criptografia: “1” = MK/WK:TDES:DAT |
| ENB_MKIDX | N2 | Índice da MK a ser utilizada. |
| ENB_WKENC | H32 | <i>Working Key</i> criptografada pela MK. |
| ENB_INPUT | H16 | Dados a serem criptografados. No modo “PAN Criptografado”, estes dados <u>sempre</u> vêm codificados usando-se TDES reverso com a chave WK_{PAN} (ver seção 5.3), independentemente do seu conteúdo. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|-------------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “ENB”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_ERRKEY.....MK não está presente no pinpad. ↳ ST_INVPARAMÍndice fornecido (ENB_MKIDX) está fora da faixa usada pelo pinpad. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “016”). |
| ENB_OUTPUT | H16 | Dados criptografados. |

➔ Exemplos

SPE solicita a criptografia do dado 4C45455045415254 usando a MK:TDES:DAT de índice “14”.

| | | |
|-------|---|--|
| SPE ➔ | 45 4E 42 30 35 31 31 31 34 46 45 34 42 31 33 36 34 34 36 33 32 39 46 45 36 30 30 30 30 30 30 30 30 30 30 30 30 30 30 34 43 34 35 34 35 35 30 34 35 34 31 35 32 35 34 | ENB051114FE4B136 446329FE60000000 000000004C45455 045415254 |
|-------|---|--|

A operação é bem-sucedida.

| | | |
|-------------|---|-------------------------------|
| ← PP | 45 4E 42 30 30 30 30 31 36 46 43 31 43 37 41 41 43 38 35 32 45 35 44 39 46 | ENB000016FC1C7AA C852E5D9F |
|-------------|---|-------------------------------|

3.3.8. Comando “GCD”

Obsoleto
 Blocante
 ABECS

Este comando permite que o SPE capture dados “em claro” através do teclado do pinpad. Para cumprir com as exigências de segurança PCI, a mensagem apresentada no *display* pinpad para a solicitação dos dados deve ser escolhida entre uma das disponíveis em uma tabela fixa definida por esta especificação.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| CMD_ID | M | Código do comando (= “GCD”). |
| SPE_MSGIDX | M | Índice da mensagem a ser apresentada, conforme tabela fixa a seguir. |
| SPE_GCDOPT | O | Opções do comando “GCD”: “0xxx” = Entrada numérica; “1xxx” = Entrada alfanumérica; “x000” = RUF. |
| SPE_MINDIG | O | Quantidade mínima de dígitos a ser capturada. Se ausente, assume-se o valor 0 (zero), ou seja, uma entrada vazia. |
| SPE_MAXDIG | O | Quantidade máxima de dígitos a ser capturada. Se ausente, assume-se o valor 32. Se presente, deve ser <u>maior ou igual a SPE_MINDIG</u> . |
| SPE_TIMEOUT | O | Tempo máximo de espera por uma ação do portador do cartão, em segundos. Se ausente, este comando nunca finaliza por erro ↪ST_TIMEOUT. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------|----------|--|
| RSP_ID | M | Código da resposta (= “GCD”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ST_CANCEL..... Portador pressionou a tecla [CANCELA]. ↪ST_TIMEOUT Esgotado tempo definido por SPE_TIMEOUT . |
| PP_VALUE | M | Valor digitado pelo portador do cartão. |

➔ Tabela fixa de mensagens

| SPE_MSGIDX | Mensagem | SPE_MSGIDX | Mensagem |
|-------------------|--------------|-------------------|--------------------------|
| 0001h | DIGITE O DDD | 001Ch | ANO DO NASCIMENTO (AAAA) |

| SPE_MSGIDX | Mensagem | SPE_MSGIDX | Mensagem |
|-------------------|-------------------------------|-------------------|---------------------------------|
| 0002h | REDIGITE O DDD | 001Dh | DIGITE IDENTIFICAÇÃO |
| 0003h | DIGITE O TELEFONE | 001Eh | CÓDIGO DE FIDELIDADE |
| 0004h | REDIGITE O TELEFONE | 001Fh | NÚMERO DA MESA |
| 0005h | DIGITE DDD+TELEFONE | 0020h | QUANTIDADE DE PESSOAS |
| 0006h | REDIGITE DDD+TELEFONE | 0021h | DIGITE QUANTIDADE |
| 0007h | DIGITE O CPF | 0022h | NÚMERO DA BOMBA |
| 0008h | REDIGITE O CPF | 0023h | NÚMERO DA VAGA |
| 0009h | DIGITE O RG | 0024h | NÚMERO DO GUICHÊ/CAIXA |
| 000Ah | REDIGITE O RG | 0025h | CÓDIGO DO VENDEDOR |
| 000Bh | DIGITE OS 4 ÚLTIMOS DÍGITOS | 0026h | CÓDIGO DO GARÇOM |
| 000Ch | DIGITE CÓDIGO DE SEGURANÇA | 0027h | NOTA DO ATENDIMENTO |
| 000Dh | DIGITE O CNPJ | 0028h | NÚMERO DA NOTA FISCAL |
| 000Eh | REDIGITE O CNPJ | 0029h | NÚMERO DA COMANDA |
| 000Fh | DIGITE A DATA (DDMMAAAA) | 002Ah | PLACA DO VEÍCULO |
| 0010h | DIGITE A DATA (DDMMAA) | 002Bh | DIGITE QUILOMETRAGEM |
| 0011h | DIGITE A DATA (DDMM) | 002Ch | QUILOMETRAGEM INICIAL |
| 0012h | DIGITE O DIA (DD) | 002Dh | QUILOMETRAGEM FINAL |
| 0013h | DIGITE O MÊS (MM) | 002Eh | DIGITE PORCENTAGEM |
| 0014h | DIGITE O ANO (AA) | 002Fh | PESQUISA DE SATISFAÇÃO (0 a 10) |
| 0015h | DIGITE O ANO (AAAA) | 0030h | AVALIE ATENDIMENTO (0 a 10) |
| 0016h | DATA DE NASCIMENTO (DDMMAAAA) | 0031h | DIGITE O TOKEN |
| 0017h | DATA DE NASCIMENTO (DDMMAA) | 0032h | DIGITE NÚMERO DO CARTÃO |
| 0018h | DATA DE NASCIMENTO (DDMM) | 0033h | NÚMERO DE PARCELAS |
| 0019h | DIA DO NASCIMENTO (DD) | 0034h | CÓDIGO DO PLANO |
| 001Ah | MÊS DO NASCIMENTO (MM) | 0035h | CÓDIGO DO PRODUTO |
| 001Bh | ANO DO NASCIMENTO (AA) | | |

➡ Exemplos

SPE solicita o RG do portador do cartão, com no máximo 10 dígitos, com tempo máximo de inatividade de 1 minuto (60 seg).

| | | |
|--------------|--|----------------------------|
| SPE ⇒ | 47 43 44 30 31 36 00 0C 00 01 3C 00 0E 00 01 0A 00 0B 00 02 00 09 | GCD016.....<..... |
|--------------|--|----------------------------|

Pinpad devolve com sucesso o dado digitado, de 9 dígitos.

| | | |
|-------------|--|----------------------------|
| ⇐ PP | 47 43 44 30 30 30 30 31 33 80 4D 00 09 31 36 39 39 33 37 38 32 33 | GCD000013€M..169 937823 |
|-------------|--|----------------------------|

3.3.9. Comando “GDU”

Obsoleto
 Blocante
 ABECS

Este comando obtém o KSN (*Key Serial Number*) corrente de um registro de tratamento DUKPT:TDES:PIN.

⚠ Este comando é **obsoleto**. O SPE deve usar o comando “**GIX**” para esta funcionalidade.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= “GDU”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “003”). |
| GDU_METHOD | N1 | Método de criptografia: “3” = DUKPT:TDES |
| GDU_IDX | N2 | Índice do registro de tratamento DUKPT:TDES:PIN. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= “GDU”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_ERRKEY Registro DUKPT não está presente no pinpad. ↳ ST_INVPARM Índice fornecido está fora da faixa usada pelo pinpad. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “020”). |
| GDU_KSN | H20 | KSN obtido. |

➔ Exemplos

SPE solicita o KSN da chave DUKPT:TDES:PIN de índice “12”

| | | |
|-------|----------------------------|-----------|
| SPE ⇒ | 47 44 55 30 30 33 33 31 32 | GDU003312 |
|-------|----------------------------|-----------|

A operação é bem-sucedida (KSN = FFFF102910025800001).

| | | |
|------|---|-----------------------------------|
| ⇐ PP | 47 44 55 30 30 30 30 32 30 46 46 46 46 46 31 30 32 39 31 30 30 32 35 38 30 30 30 30 31 | GDU000020FFFFF10 2910025800001 |
|------|---|-----------------------------------|

3.3.10. Comando “GKY”

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input checked="" type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

Este comando é utilizado para aguardar o pressionamento de uma tecla no pinpad, retornando seu código. Por questão de segurança, este comando não retorna teclas numéricas, sendo que o pressionamento destas teclas é simplesmente ignorado pelo pinpad durante a execução do comando.

⚠ Este comando é **obsoleto**. Para esta funcionalidade, o SPE deve usar o comando “**CEX**” com **SPE_CEXOPT = “100000”**.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|------------------------------|
| CMD_ID | A3 | Código do comando (= “GKY”). |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “GKY”). |
| RSP_STAT | N3 | Retornos relevantes (ver seção 3.1.1): ↪ ST_OK Pressionada tecla de confirmação (OK ou ENTER) ↪ ST_CANCEL Pressionada a tecla [CANCELA]. ↪ ST_BACKSP..... Pressionada a tecla [LIMPA]. ↪ ST_F1 a ↪ ST_F4 Pressionada tecla de função. |

➔ Exemplos

SPE solicita o pressionamento de uma tecla no pinpad.

| | | |
|-------|----------|-----|
| SPE ➔ | 47 4B 59 | GKY |
|-------|----------|-----|

A operação é bem-sucedida (é pressionada a tecla [CANCELA]).

| | | |
|------|-------------------|--------|
| ← PP | 47 4B 59 30 31 33 | GKY013 |
|------|-------------------|--------|

3.3.11. Comando “GPN”

Obsoleto
 Blocante
 ABECS

Este comando captura a senha do portador do cartão (PIN) e retorna um bloco de dados criptografados segundo os métodos MK/WK:TDES ou DUKPT:TDES.

O pinpad sempre apaga o conteúdo do *display* ao final do processamento, seja ele bem ou malsucedido.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| CMD_ID | A3 | Código do comando (= “GPN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| GPN_METHOD | N1 | Método de criptografia: “1” = MK/WK:TDES:PIN “3” = DUKPT:TDES:PIN |
| GPN_KEYIDX | N2 | Índice da MK ou do registro de tratamento DUKPT. |
| GPN_WKENC | H32 | <i>Working Key</i> (criptografada pela MK indicada em GPN_KEYIDX). Se GPN_METHOD = “3”, este campo é ignorado pelo pinpad. |
| GPN_PANLEN | N2 | Tamanho do PAN (de “02” a “19”). Caso este comando esteja sendo usado depois de “GCX” ou “CEX”, Se a criptografia “End-to-End” estiver sendo utilizada (ver seção 5.4) e o comando “GTK” ainda não foi executado, deve-se poder-se fornecer um PAN “vazio” (tamanho “00”) para que o pinpad considere o PAN já armazenado em sua memória, preservando-se assim a segurança da criptografia “End-to-End” (ver seção 5.4). |
| GPN_PAN | A19 | PAN, alinhado à esquerda (espaços à direita). Se o pinpad estiver em modo “PAN Criptografado”, este campo deve ser codificado usando-se TDES ⁻¹ (<u>reverso</u>) com a chave WK _{PAN} (ver seção 5.3). |
| GPN_ENTRIES | N1 | Quantidade de dados a serem capturados (fixo em “1”). |
| GPN_MIN1 | N2 | Tamanho mínimo do dado #1 (≥ “04”). |
| GPN_MAX1 | N2 | Tamanho máximo do dado #1 (≥ GPN_MIN1). |
| GPN_MSG1 | S32 | Mensagem de 2 linhas por 16 colunas para apresentação no momento do pedido do dado #1. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|-------------------------------|
| RSP_ID | A3 | Código da resposta (= “GPN”). |

| Id. do Campo | Formato | Descrição |
|-------------------|---------|--|
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_CANCELPortador pressionou a tecla [CANCELA]. ↪ ST_TIMEOUT.....Esgotado tempo máximo de ociosidade. ↪ ST_ERRKEY.....MK ou DUKPT não está presente no pinpad. ↪ ST_INVPARMÍndice fornecido (GPN_KEYIDX) está fora da faixa usada pelo pinpad. ↪ ST_INVPARM GPN_MIN1 é inferior a “04”. ↪ ST_INVPARM GPN_ENTRIES é diferente de “1”. ↪ ST_INVCALLO PAN não é conhecido pelo pinpad. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “036”). |
| GPN_PINBLK | H16 | Resultado criptografado. |
| GPN_KSN | H20 | Número de série da chave (<i>Key Serial Number</i>), somente no caso de DUKPT (GPN_METHOD = “3”). Para MK/WK, este campo é devolvido zerado. |

➤ Exemplos

SPE solicita a captura de PIN usando o método MK/WK:TDES, com chave de índice “08”.

| | | |
|--------------|---|---|
| SPE ⇒ | 47 50 4E 30 39 33 31 30 38 34 31 33 35 45 41 35 38 42 41 31 33 45 32 36 32 46 34 34 43 35 39 45 44 37 38 39 39 41 41 33 43 31 36 34 34 34 34 33 33 33 33 32 32 32 32 31 31 31 31 20 20 20 31 30 34 31 32 52 24 20 20 20 20 20 20 20 20 33 34 2C 35 36 44 49 47 49 54 45 20 53 55 41 20 53 45 4E 48 41 | GPN0931084135EA5 8BA13E262F44C59E D7899AA3C1644443 33322221111•••10 412R\$••••••••34 ,56DIGITE•SUA•SE NHA |
|--------------|---|---|

A operação é bem-sucedida.

| | | |
|-------------|---|--|
| ⇐ PP | 47 50 4E 30 30 30 30 33 36 42 42 36 42 45 32 38 46 44 46 33 35 32 32 45 39 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 | GPN000036BB6BE28 FDF3522E90000000 00000000000000 |
|-------------|---|--|

3.3.12. Comando “GTK”

Obsoleto
 Blocante
 ABECS

Este comando permite que o SPE obtenha as trilhas completas do cartão lido através dos comandos “**CEX**” ou “**G CX**” (caso em que pode também retornar o PAN). Os dados podem ser devolvidos “em claro” ou criptografados conforme definido na **seção 5.4**.

- ▲ O comando “**GTK**” somente pode ser usado uma única vez depois de “**CEX**” ou “**G CX**”.
- ▲ Para trilhas criptografadas, devem-se utilizar os parâmetros definidos nas especificações da Rede Credenciadora que irá processar a transação.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| CMD_ID | M | Código do comando (= “ GTK ”). |
| SPE_TRACKS | O | Identificação de quais informações de trilha devem ser devolvidas: “ 1xxx ” = PAN é requerido, <u>se disponível</u> ¹ ; “ 0xxx ” = PAN não é requerido. “ x1xx ” = Trilha 1 é requerida, <u>se disponível</u> ; “ x0xx ” = Trilha 1 não é requerida. “ xx1x ” = Trilha 2 é requerida, <u>se disponível</u> ; “ xx0x ” = Trilha 2 não é requerida. “ xxx1 ” = Trilha 3 é requerida, <u>se disponível</u> ; “ xxx0 ” = Trilha 3 não é requerida. Se este campo estiver ausente, todas as informações conhecidas pelo pinpad serão devolvidas. |
| SPE_MTHDDAT | O | Identificação do modo de criptografia a ser utilizado: “ 10 ” = MK/WK:TDES:DAT (criptografia de bloco ECB); “ 11 ” = MK/WK:TDES:DAT (criptografia de bloco CBC); “ 50 ” = DUKPT:TDES:DAT#3 (criptografia de bloco ECB , ver seção 5.1.1); e “ 51 ” = DUKPT:TDES:DAT#3 (criptografia de bloco CBC , ver seção 5.1.1). “ 90 ” = TDES com chave aleatória (criptografia de bloco ECB). “ 91 ” = TDES com chave aleatória (criptografia de bloco CBC). Se este campo estiver ausente, as trilhas são devolvidas “em claro”. |

¹ Entende-se que o dado está “disponível” quando este é lido com sucesso do cartão magnético, ou, no caso de cartão com *chip*, quando os objetos TLV equivalentes estão presentes.

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|--|
| <u>SPE_IVCBC</u> | O | “IV” (<i>Initialization Vector</i>) a ser usado na criptografia, se <u>SPE_MTHDDAT</u> = “x1” (criptografia de bloco CBC). Se ausente, o pinpad considerará o “IV” zerado. |
| <u>SPE_OPNDIG</u> | O | Quantidade de dígitos numéricos (número par) a serem preservados “em claro” no início das trilhas. Se não fornecido, a trilha inteira é criptografada. |
| <u>SPE_KEYIDX</u> | MD | Índice da chave a ser utilizada (MK:DAT ou DUKPT:DAT) na criptografia de trilhas. Este campo é mandatório se <u>SPE_MTHDDAT</u> estiver presente e diferente de “9x”. |
| <u>SPE_WKENC</u> | MD | <i>Working Key</i> (criptografada pela MK) a ser usada na criptografia das trilhas. Este campo é mandatório somente se <u>SPE_MTHDDAT</u> = “1x”. |
| <u>SPE_PBKMOD</u> | MD | Módulo de uma chave pública RSA. Este campo é mandatório se <u>SPE_MTHDDAT</u> = “9x”. |
| <u>SPE_PBKEXP</u> | MD | Expoente de uma chave pública RSA. Este campo é mandatório se <u>SPE_MTHDDAT</u> = “9x”. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|---|
| <u>RSP_ID</u> | M | Código da resposta (= “GTK”). |
| <u>RSP_STAT</u> | M | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_INVCALL..... Comandos “CEX” ou “GCX” não foram executados previamente com sucesso. ↳ ST_INVCALL..... O comando “GTK” já foi utilizado. ↳ ST_ERRKEY MK ou DUKPT não está presente no pinpad. ↳ ST_INVPARAM..... Índice fornecido (<u>SPE_KEYIDX</u>) está fora da faixa usada pelo pinpad. |
| <u>PP_ENCPAN</u> | O | PAN do cartão, “em claro” ou criptografado, se disponível no pinpad e requerido em <u>SPE_TRACKS</u> (somente para cartão com <i>chip</i> , depois de “GCX”). |
| <u>PP_TRACK1</u> | O | Trilha 1 do cartão, “em claro” ou criptografada, se disponível no pinpad e requerida em <u>SPE_TRACKS</u> . |
| <u>PP_TRACK2</u> | O | Trilha 2 do cartão, “em claro” ou criptografada, se disponível no pinpad e requerida em <u>SPE_TRACKS</u> . |
| <u>PP_TRACK3</u> | O | Trilha 3 do cartão, “em claro” ou criptografada, se disponível no pinpad e requerida em <u>SPE_TRACKS</u> e conhecida pelo pinpad. |
| <u>PP_TRK1KSN</u> | MD | KSN gerado na criptografia da trilha 1. Este campo é mandatório se <u>PP_TRACK1</u> presente e foi solicitada criptografia DUKPT. |

| Id. do Campo | Presença | Descrição / Observação |
|---------------------|----------|---|
| <u>PP_TRK2KSN</u> | MD | KSN gerado na criptografia da trilha 2. Este campo é mandatório se <u>PP_TRACK2</u> presente e foi solicitada criptografia DUKPT. |
| <u>PP_TRK3KSN</u> | MD | KSN gerado na criptografia da trilha 3. Este campo é mandatório se <u>PP_TRACK3</u> presente e foi solicitada criptografia DUKPT. |
| <u>PP_ENCPANKSN</u> | MD | KSN gerado na criptografia do PAN. Este campo é mandatório se <u>PP_ENCPAN</u> presente e foi solicitada criptografia DUKPT. |
| <u>PP_ENCKRAND</u> | MD | Chave K_{RAND} criptografado pela chave pública fornecida, se <u>SPE_MTHDDAT</u> = "9x". |

▲ Caso um cartão magnético tenha sido passado em "CEX" ou "GCX" mas nenhuma trilha pôde ser lida (erro de leitura), "GTK" deve retornar ↵ST_OK sem dados de cartão.

➡ Exemplos

SPE solicita as três trilhas com 6 posições "em claro" no início, utilizando criptografia DUKPT:TDES (variante #2) em modo ECB, com a chave de índice "12".

| | | |
|--------------|---|-------------------------------------|
| SPE ⇒ | 47 54 4B 30 32 35 00 03 00 02 34 30 00 07 00 04 30 31 31 31 00 08 00 01 36 00 09 00 02 31 32 | GTK025....40.... 0111....6....12 |
|--------------|---|-------------------------------------|

Pinpad retorna as trilhas 1 e 2 e os respectivos KSN gerados, mas não devolve a trilha 3 por desconhecê-la.

| | | |
|-------------|---|---|
| ← PP | 47 54 4B 30 30 30 31 33 33 80 44 00 4E 35 34 37 38 32 33 7A E2 FA 69 BA 8C 62 93 9E C2 38 2C 33 D5 A1 6C 06 A2 D4 F6 EA 24 1E DC 93 73 21 92 FD D5 32 74 95 66 7C 8F D2 DF E6 A0 1C B7 94 BE C5 8C 57 65 D9 4C E1 8A CD CC CB 57 68 51 64 DD 65 56 C7 35 BE 35 7E 39 45 6A 68 DB 80 47 00 0A FF FF F8 19 46 00 18 70 00 1F 80 45 00 13 54 78 23 EA 2F B6 CD 92 89 F9 70 1C B0 88 3F D6 CC 6F 79 80 48 00 0A FF FF F8 19 46 00 18 70 00 1F | GTK000133€D.N547 823zâúí°œb“ŽÅ8,3 Öj l.çÖöê\$.Ü“s!‘ý Ö2t•f •Ößæ . .”%A œweÜLáŠÍÏËwhQdÝe Vç5¼5~9EjhÜ€G..ÿ ÿø.F...p...€E..Tx# ê/¶Í’%ùp. °?Öïoy €H..ÿÿø.F..p.. |
|-------------|---|---|

SPE solicita o PAN e a Trilha 2 com 4 posições "em claro" no início, utilizando criptografia MK/WK:TDES:DAT em modo CBC com "IV" (Initialization Vector), usando a chave de índice "07".

| | | |
|--------------|---|---|
| SPE ⇒ | 47 54 4B 30 35 37 00 07 00 04 31 30 31 30 00 03 00 02 31 31 00 1D 00 08 7F 7C 1A FA C0 A8 4F B7 00 08 00 01 34 00 09 00 02 30 37 00 0A 00 10 C2 BC A2 4F 3E F8 F2 EF 1C 0F 07 A9 7D 38 C3 38 | GTK057....1010.. ..11.... .úÁ`O·4....07....Ã ¼çO>øòï...@}8Å8 |
|--------------|---|---|

Pinpad retorna corretamente o PAN e a Trilhas 2 criptografados.

| | | |
|-------------|---|---|
| ← PP | 47 54 4B 30 30 30 30 34 34 80 4A 00 0A 41 23 FC 45 2F 36 15 44 A7 32 80 45 00 1A 41 23 BB 80 F6 58 D4 4F BC 29 4B 8A 63 99 01 26 95 48 B8 8A C9 52 01 E8 4F BF | GTK000044€J..A#ü E/6.D\$2€E..A#»€ö xÔ0¼)KŠc™.&•H, ŠÉ R.èöç |
|-------------|---|---|

SPE solicita o PAN, a Trilha 1 e a Trilha 2 totalmente criptografados utilizando uma chave aleatória TDES em modo CBC sem "IV" (Initialization Vector).

| | | |
|-------|---|--|
| SPE ⇒ | 47 54 4B 32 38 31 00 07 00 04 31 31 31 30 00 03 00 02 39 31 00 24 00 80 80 45 05 9A 9D C7 D2 77 09 06 DC FD 01 04 E3 1E 23 CE 30 85 71 61 5D 1D BA 6E C2 29 91 13 76 26 3B 6B 64 A3 CE 89 21 A7 9C 94 80 E5 32 1E 52 66 28 7D 43 48 60 B7 5A 92 FD B0 4B A8 8A 59 95 C2 4B FC 02 EC 2D CB 5C 8F AA C0 62 D7 60 D3 5E 79 98 9D 8E D9 8A D0 E3 56 53 F4 B4 84 68 39 55 17 C3 17 12 AD E5 62 3C F5 29 4C BC CF EA CE 1A DA 9B 89 E2 21 22 D7 5C 39 31 BC 14 E6 C1 BD 39 1B BF BF D9 E8 E8 A4 E5 4D F8 7B 05 AC 4E 43 E1 3F AA 93 EB A6 7D 95 D4 D3 B6 C3 D2 47 D3 C2 55 A7 F8 65 B3 96 82 2E 19 85 08 04 95 8E C9 1B 31 A2 3D 68 6F FE 4A 76 E6 4C 31 B8 EA 51 BC 03 41 B5 79 7D AB 18 F6 F9 97 03 35 6A B1 8D 9B FD 62 33 CD BC 31 DC 2C 46 F1 76 1A F5 AF 5C EF C8 2A 29 32 99 0A 4D 04 67 D9 15 79 CF E1 26 83 48 DA 19 FF 3F C7 EA 96 9E B3 47 37 7A EA EA 64 21 AA 55 00 25 00 03 01 00 01 | GTK281...1110.. ..91.\$..€E.š•Çõw ..Üý...ã.#î0...qã]. °nÃ)' .v&;kdfî%!§ æ"ëã2.Rf{)CH` .Z' ý°K`ŠY•ÁKü.ì-È`• °Àbx`óÁy~•ZÛŠDãV Sô` ,h9U.Ã. .-ãb<õ)L¼îêî.ú>%â!"\x)9 1¼.æÁ½9.¿¿Üèèæãm ø{.-Ncá?ª"è!}•õõ ¶ÃÖGÓÂUšøe³- ,..... ..•ZÉ.1¢=hopjvæL 1`êQ¼.Auy}«.òù-. 5j±>ýb3í¼1Ü, Fñv .õ` \îÊ*)2™.M.gÜ. yİá&fHÚ.ý?Çê-Z³G 7zêêd!ªU.%...... |
|-------|---|--|

Pinpad retorna corretamente o PAN e a trilhas 1 criptografados, mas não devolve a trilha 2 por desconhecê-la.

| | | |
|------|---|--|
| ⇐ PP | 47 54 4B 33 35 37 30 30 30 80 4A 00 08 F1 58 F8 C2 2E 09 59 1E 80 44 00 51 42 FB A4 60 A1 A9 17 B1 72 5C E1 E7 32 35 33 D0 7C 9F 0B 9A 6B E5 AB AD 0D DB A1 D6 7F F0 F7 DE A3 7F 5A 4F 5A 17 DA 95 17 E7 3F 77 70 D7 7B 64 38 C7 FA 04 0B C4 BD 71 8F 80 56 86 7B 6F F9 51 76 A0 63 7B 67 91 F4 04 8D C3 38 5C 45 58 8D 82 07 80 63 01 00 13 F7 3B C3 B1 9D 6A 2D 25 0D 96 80 6D 1A 98 5F DF D1 96 35 02 A2 5A B1 07 E1 28 87 CC D1 C0 5E 5E 9B EE C6 CA 3D 81 AA 34 36 57 66 9B D1 76 0C 9B 5B FD 48 CD 77 93 F5 15 4E 6B 15 49 F3 99 33 B1 22 1A 15 8E 7B F7 E8 C0 6B 7B FE 5F 47 38 13 E7 FE 6A 93 47 84 36 10 5F 7E 85 40 00 15 3E BC 95 38 56 12 FF 90 5D D3 8B 3F 6D 86 1F EA B9 E4 1A 7F EA 6D 61 0A 71 0A 4A E4 F2 2B C6 35 A7 18 0C 2D 6C A4 A6 FA A3 F8 FD 51 E8 CA 0C 9E D1 DA 70 E1 FC 1D BF C6 DB CB 29 BF 90 4F 07 40 BC C1 7D FB 82 16 D5 81 46 F6 4B 46 23 8B 85 5D 86 C6 CF 8F 4E 8B 0B 0E DF EE 90 3C 82 01 F7 8E C7 8C 88 31 12 0E C4 D2 F6 CA E2 A2 39 ED FF A9 94 50 EE 4D 5C 95 B8 8B A4 A9 7A C3 2D 3A FD 62 69 88 B1 BE EE D3 A4 CB 16 E1 87 0D 88 74 F6 E0 F8 B7 B6 7C D7 35 B0 F7 96 1E 5A 22 18 1D D2 A6 2D 77 | GTK357000€J..ñXø Â. .Y.€D.QBûª`j@. ±r\ác253Ð Ý.šká« -.Üj;öð÷þf□ZOZ.Ú .ç?wp×{d8Çú..Ã½ q•€V†{ouQv c{g'ó .•Ã8\EX.,.€c...÷ ;Ã±•j-%.-€m.~_ßÑ -5.ÇZ±.á(†İÑÅ^^> îÆÊ=ª46wf>ñv.>[ýHÍw"ö.Nk.Ió™3±" . .Z{÷èAk{p_g8.çp j"G,,6._~...@..>¼•8 v.ÿ•]ó<?m†.ê'ã.□ êma.q.Jãð+Æ5§..-]ª!úføyQèÊ.žNúPá ü.¿ÆÜÈ)¿•O.@¼A}û ,.õ•FöKF#<...]†Æİ• N<..ßî•<.,÷žÇÆ~1 ..ÄööÊãç9íÿ@"PîM \.,<ª@zÃ-:ýbi`±¾ îOªÊ.á†.ˆtòàø.¶ ×5°÷-.Z"...ò!-w |
|------|---|--|

Para efeito de validação, este exemplo considera o seguinte valor para a chave aleatória K_{RAND} e para o expoente privado K_{PRV}:

K_{RAND} = FF 47 55 39 9A E4 28 93 44 D4 BB C0 7D 96 8B 5F

K_{PRV} =

| | | | | | | | | | | | | | | | |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 24 | 2B | D2 | 9D | BC | 5A | AA | 16 | 19 | 3C | 8F | 3A | E5 | 7B | AC | 54 |
| 46 | 82 | 91 | 9A | 3F | D3 | D5 | FF | 59 | 20 | 7C | AE | 5E | 13 | DF | E0 |
| 7E | 27 | 15 | B5 | 3F | BB | D9 | FA | BB | 24 | 01 | 89 | 20 | 6D | FE | 8C |
| 82 | 64 | 78 | 81 | C3 | 8C | 51 | 05 | 5C | 76 | C7 | 8F | 1A | 9C | 92 | A7 |
| BC | E7 | AF | 27 | 4C | EE | A9 | 06 | 76 | 7F | 54 | 20 | 2A | 54 | D0 | B2 |
| 77 | 80 | 0E | D5 | 77 | D8 | DA | 12 | F1 | 0F | F3 | 8B | D7 | 1C | 3B | CB |
| BC | 9F | 18 | 0C | 63 | C0 | 25 | 32 | 79 | 58 | 03 | 72 | 9A | 63 | 4E | 9D |
| 50 | F9 | 3C | 04 | 5E | 1F | DF | 08 | DD | E6 | 8C | FA | 59 | AD | F3 | 99 |
| 62 | 5F | 01 | 5E | 0E | 32 | 70 | BB | 2B | 7F | 27 | D2 | 16 | E8 | AE | 43 |
| 28 | 1C | 2E | 43 | E4 | A2 | 4E | 77 | 34 | 05 | 86 | 94 | C5 | 93 | 45 | 35 |
| C2 | 4E | FD | 21 | B2 | CC | 47 | AE | 93 | 82 | 7F | C9 | 38 | 1B | 6D | 59 |
| F3 | 50 | B2 | F3 | 53 | 43 | 71 | AF | A3 | E4 | 0D | 5C | A3 | 1A | C7 | 74 |
| 45 | 83 | A3 | 86 | 1E | 08 | E4 | 42 | 36 | 34 | B2 | 9D | B2 | C3 | BA | 14 |
| D2 | F3 | 7E | 70 | 4F | 1A | AB | E6 | 51 | F2 | 5C | 43 | E0 | DE | 57 | 7F |
| B5 | 30 | EF | 17 | AC | B8 | F1 | 5A | A5 | A9 | 0D | 20 | D8 | 35 | DA | 78 |
| 2C | 5D | 69 | 6A | 44 | DB | F8 | EB | 21 | 3E | B3 | E3 | 46 | 3E | 53 | 01 |

3.3.13. Comando “MNU”

Obsoleto
 Blocante
 ABECS

Este comando faz com que o pinpad apresente em seu *display* um menu de até 20 opções para escolha do portador do cartão, utilizando da melhor maneira os seus recursos de *hardware*.

- Cada opção pode ter no máximo 24 caracteres.
- O pinpad apresentará o menu de opções sempre respeitando a ordem em que foram fornecidas.
- Caso a opção seja iniciada por um caractere numérico (“0” a “9”), o pinpad poderá permitir a seleção através do teclado (“*hot key*”), mediante o pressionamento da tecla correspondente ao caractere. Caso o SPE opte por usar este recurso, cabe a ele garantir a integridade das opções para que não haja repetição.
- O SPE deve fornecer ao menos uma opção.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|---|
| CMD_ID | M | Código do comando (= “MNU”). |
| SPE_TIMEOUT | O | Tempo máximo de espera por uma ação do portador do cartão, em segundos. Se ausente, este comando nunca finaliza por erro ↪ST_TIMEOUT. |
| SPE_DSPMSG | O | Título do menu. Caso não seja fornecido, o menu é apresentado sem título. |
| SPE_MNUOPT | M | Texto para a primeira opção do menu (índice “01”). |
| SPE_MNUOPT | O | Texto para a segunda opção do menu (índice “02”). |
| ... | ... | ... |
| SPE_MNUOPT | O | Texto para a última opção do menu (índice “xx”, onde xx é o número total de opções fornecidas). |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------|----------|---|
| RSP_ID | M | Código da resposta (= “MNU”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ST_CANCEL..... Portador pressionou a tecla [CANCELA]. ↪ST_TIMEOUT..... Esgotado tempo definido por SPE_TIMEOUT . |
| PP_VALUE | M | Índice de dois dígitos numéricos referente à opção de menu selecionada, considerando-se ordem em que foram fornecidas pelo SPE (a partir de “01”). |

➔ Exemplos

O SPE solicita ao pinpad que apresente um menu com o título “Selecione, por favor:” e as opções:

⇒ “5.Chamado Técnico”

⇒ “1.Consultas”

⇒ “3.Ajuda”

⇒ “Voltar!!”

O tempo máximo de inatividade é de 30 segundos.

| | | |
|--------------|---|--|
| SPE ⇒ | 4D 4E 55 30 38 39 00 0C 00 01 1E 00 20 00 11 35 2E 43 68 61 6D 61 64 6F 20 54 E9 63 6E 69 63 6F 00 20 00 0B 31 2E 43 6F 6E 73 75 6C 74 61 73 00 20 00 07 33 2E 41 6A 75 64 61 00 20 00 08 56 6F 6C 74 61 72 21 21 00 1B 00 15 53 65 6C 65 63 69 6F 6E 65 2C 20 70 6F 72 20 66 61 76 6F 72 3A | MNU089.....•..5 .Chamado•Técnico ...1.Consultas. ...3.Ajuda...Vo ltar!!....Selec ione,•por•favor: |
|--------------|---|--|

Pinpad devolve com sucesso o valor “02”, indicando que a opção “1.Consultas” foi selecionada.

| | | |
|-------------|--|-----------------|
| ⇐ PP | 4D 4E 55 30 30 30 30 30 36 80 4D 00 02 30 32 | MNU000006€M..02 |
|-------------|--|-----------------|

3.3.14. Comando “RMC”

| |
|--|
| <input type="checkbox"/> Obsoleto |
| <input checked="" type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

Este comando aguarda a remoção do ICC. Ele possui dois comportamentos diferentes, de acordo com a presença ou não de um cartão no acoplador.

Cartão presente: Apresenta a mensagem definida por **RMC_MSG**, alternada com uma mensagem de “RETIRE O CARTÃO”, permanecendo neste estado até a retirada do cartão.

Cartão ausente: Apresenta a mensagem definida por **RMC_MSG** e retorna imediatamente.

Em ambos os casos, a mensagem definida por **RMC_MSG** é deixada no *display* ao final.

➔ Comando

| Id. do Campo | Formato | Descrição |
|-----------------|---------|--|
| CMD_ID | A3 | Código do comando (= “RMC”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “032”). |
| RMC_MSG | S32 | Mensagem de 32 caracteres a ser apresentada no <i>display</i> do pinpad, já formatada corretamente para 2 linhas e 16 colunas. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|-----------------|---------|-------------------------------|
| RSP_ID | A3 | Código da resposta (= “RMC”). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |

➔ Exemplos

SPE solicita remoção do cartão, apresentando a mensagem “OPERAÇÃO FINALIZADA”.

| | | |
|----------------------------|---|--|
| SPE ⇒ | 52 4D 43 30 33 32 20 20 20 20 4F 50 45 52 41 C7 C3 4F 20 20 20 20 20 20 20 46 49 4E 41 4C 49 5A 41 44 41 20 20 20 | RMC032.....OPERAÇ ÃO.....FINALIZ ADA.... |
| A operação é bem-sucedida. | | |
| ⇐ PP | 52 4D 43 30 30 30 | RMC000 |

3.4. Comandos multimídia

Esta especificação prevê uma série de comandos para uso em pinpads com recurso de multimídia (display gráfico colorido e/ou áudio). O suporte a estes comandos é opcional e depende dos recursos do equipamento.

Esta especificação prevê os seguintes formatos de arquivo, que podem ou não ser suportados pelo pinpad, definição informada no comando "**GIX**" (**PP_MFSUP**).

- Imagem PNG (*Portable Network Graphics*) de acordo com ISO/IEC 15948;
- Imagem JPG (ou JPEG) de acordo com ISO/IEC 10918; e
- Imagem ou animação GIF (*Graphics Interchange Format - CompuServe*).

⚠ Quando um comando desta seção não é suportado pelo pinpad, ele simplesmente devolve a resposta de erro definida na **seção 2.3.4** (com **RSP_STAT** = "010"), como faz para qualquer outro comando desconhecido.

Os seguintes comandos estão contemplados nesta seção:

| CMD_ID | Significado | Obsoleto | Blocante | Abecs |
|----------------|--|--------------------------|--------------------------|-------------------------------------|
| " MLI " | <i>Multimedia File Load - Initialization</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| " MLR " | <i>Multimedia File Load - Record</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| " MLE " | <i>Multimedia File Load - End</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| " LMF " | <i>List Multimedia File</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| " DMF " | <i>Delete Multimedia Files</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| " DSI " | <i>Display Image</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

3.4.1. Comando “MLI”

| |
|---|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando inicia o processo de carga (ou substituição) de um arquivo multimídia no pinpad. Este arquivo é armazenado de forma “não volátil” e é preservado mesmo depois do pinpad ser desligado.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|---|
| CMD_ID | M | Código do comando (= “MLI”). |
| SPE_MFNAME | M | Nome do arquivo multimídia a ser carregado. |
| SPE_MFINFO | M | Informações sobre o arquivo multimídia a ser carregado: X4 = Tamanho (de 0 a 4294967295 bytes). B2 = CRC do arquivo. B1 = Tipo (01h = PNG , 02h = JPG , 03h = GIF , outros valores = RUF); e B3 = RUF (000000h). |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------|----------|-------------------------------|
| RSP_ID | M | Código da resposta (= “MLI”). |
| RSP_STAT | M | Ver seção 3.1.1. |

➔ Exemplos

SPE solicita o início da carga de um arquivo PNG de 3.334 bytes de nome “QRCODE01”.

| | | |
|--------------|--|-----------------------------------|
| SPE ➔ | 4D 4C 49 30 32 36 00 1E 00 08 51 52 43 4F 44 45 30 31 00 1F 00 0A 00 00 0D 06 F2 11 01 00 00 00 | MLI026....QRCODE 01.....ò..... |
|--------------|--|-----------------------------------|

A operação é bem-sucedida.

| | | |
|-------------|-------------------|--------|
| ⬅ PP | 4D 4C 49 30 30 30 | MLI000 |
|-------------|-------------------|--------|

3.4.2. Comando “MLR”

Obsoleto
 Blocante
 ABECS

Através de uma ou mais chamadas a este comando, o SPE envia os dados do arquivo multimídia cuja carga foi iniciada pelo comando “**MLI**”.

Os dados podem ser divididos em vários blocos para se respeitar a estrutura padrão dos pacotes de protocolo, conforme descrito na **seção 3.1.3.1**.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|---------------------------------------|
| CMD_ID | M | Código do comando (= “ MLR ”). |
| SPE_DATAIN | M | Bloco de dados do arquivo. |
| SPE_DATAIN | O | Bloco de dados do arquivo. |
| ... | .. | ... |
| SPE_DATAIN | O | Bloco de dados do arquivo. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------|----------|--|
| RSP_ID | M | Código da resposta (= “ MLR ”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_INVCALL..... Comando “ MLI ” não foi executado previamente. ↪ ST_INTERR..... Falta de memória para gerenciamento dos dados recebidos. |

➡ Exemplos

SPE inicia a carga dos dados do arquivo PNG do exemplo do comando “MLI” (seção 3.4.1). Pode-se notar que o comando é dividido em dois blocos (CMD_BLK1 e CMD_BLK2), ambos de 436 bytes.

| | | |
|----------------------------|--|---|
| SPE ⇒ | <pre> 4D 4C 52 34 33 36 00 0F 01 B0 89 50 4E 47 0D 0A 1A 0A 00 00 00 0D 49 48 44 52 00 00 00 7D 00 00 00 7D 08 02 00 00 00 00 00 E2 FB 72 00 00 0A 37 69 43 43 50 73 52 47 42 20 49 45 43 36 31 39 36 36 2D 32 2E 31 00 00 78 9C 9D 96 77 54 53 D9 16 87 CF BD 37 BD 50 92 10 8A 94 D0 6B 68 52 02 48 0D BD 48 91 2E 2A 31 09 10 4A C0 90 00 22 36 44 54 70 44 51 91 A6 08 32 28 E0 80 A3 43 91 B1 22 8A 85 01 51 B1 EB 04 19 44 D4 71 70 14 1B 96 49 64 AD 19 DF BC 79 EF CD 9B DF 1F F7 7E 6B 9F BD CF DD 67 EF 7D D6 BA 00 90 FC 83 05 C2 4C 58 09 80 0C A1 58 14 E1 E7 C5 88 8D 8B 67 60 07 01 0C F0 00 03 6C 00 E0 70 B3 B3 42 16 F8 46 02 99 02 7C D8 8C 6C 99 13 F8 17 BD BA 0E 20 F9 FB 2A D3 3F 8C C1 00 FF 9F 94 B9 59 22 31 00 50 98 8C E7 F2 F8 D9 5C 19 17 C9 38 3D 57 9C 25 B7 4F C9 98 B6 34 4D CE 30 4A CE 22 59 82 32 56 93 73 F2 2C 5B 7C F6 99 65 0F 39 F3 32 84 3C 19 CB 73 CE E2 65 F0 E4 DC 27 E3 8D 39 12 BE 8C 91 60 19 17 E7 08 F8 B9 32 BE 26 63 83 74 49 86 40 C6 6F E4 B1 19 7C 4E 36 00 28 92 DC 2E E6 73 53 64 6C 2D 63 92 28 32 82 2D E3 79 00 E0 48 C9 5F F0 D2 2F 58 CC CF 13 CB 0F C5 CE CC 5A 2E 12 24 A7 88 19 26 5C 53 86 8D 93 13 8B E1 CF CF 4D E7 8B C5 CC 30 0E 37 8D 23 E2 31 D8 99 19 59 1C E1 72 00 66 CF FC 59 14 79 6D 19 B2 22 3B D8 38 39 38 30 6D 2D 6D BE 28 D4 7F 5D FC 9B 92 F7 76 96 5E 84 7F EE 19 44 1F F8 C3 F6 57 7E 99 0D 00 34 33 36 00 0F 01 B0 B0 A6 65 B5 D9 FA 87 6D 69 15 00 5D EB 01 50 BB FD 87 CD 60 2F 00 8A B2 BE 75 0E 7D 71 1E BA 7C 5E 52 C4 E2 2C 67 2B AB DC DC 5C 4B 01 9F 6B 29 2F E8 EF FA 9F 0E 7F 43 5F 7C CF 52 BE DD EF E5 61 78 F3 93 38 92 74 31 43 5E 37 6E 66 7A A6 44 C4 C8 CE E2 70 F9 0C E6 9F 87 F8 1F 07 FE 75 1E 16 11 FC 24 BE 88 2F 94 45 44 CB A6 4C 20 4C 96 B5 5B C8 13 88 05 99 42 86 40 F8 9F 9A F8 0F C3 FE A4 D9 B9 96 89 DA F8 11 D0 96 58 02 A5 21 1A 40 7E 1E 00 28 2A 11 20 09 7B 64 2B D0 EF 7D 0B C6 47 03 F9 CD 8B D1 99 98 9D FB CF 82 FE 7D 57 B8 4C FE C8 16 24 7F 8E 63 47 44 32 B8 12 51 CE EC 9A FC 5A 02 34 20 00 45 40 03 EA 40 1B E8 03 13 C0 04 B6 C0 11 B8 00 0F E0 03 02 41 28 88 04 71 60 31 E0 82 14 90 01 44 20 17 14 80 B5 A0 18 94 82 AD 60 27 A8 06 75 A0 11 34 83 36 70 18 74 81 63 E0 34 38 07 2E 81 CB 60 04 DC 01 52 30 0E 9E 80 29 F0 0A CC 40 10 84 85 C8 10 15 52 87 74 20 43 C8 1C B2 85 58 90 1B E4 03 05 43 11 50 1C 94 08 25 43 42 48 02 15 40 EB A0 52 A8 1C AA 86 EA A1 66 E8 5B E8 28 74 1A BA 00 0D 43 B7 A0 51 68 12 FA 15 7A 07 23 30 09 AB C1 5A B0 11 6C 05 B3 60 4F 38 08 8E 84 17 C1 C9 F0 32 38 1F 2E 82 B7 C0 95 70 03 7C 10 EE 84 4F C3 97 E0 11 58 0A 3F 81 A7 11 80 10 11 3A A2 8B 30 11 16 C2 46 42 91 78 24 09 11 21 AB 90 12 A4 02 69 40 DA 90 1E A4 1F B9 8A 48 91 A7 C8 5B 14 06 45 45 31 50 4C </pre> | <pre> MLR436... °%PNG..IHDR...}.. .}......âûr...7i CCPSRGB•IEC61966 •2.1..xœ•wTSÙ.‡ İ½7½P' .Š"ĐkH.R.H. ½H' .*1..JÀ•."6DT pDQ' .2(à€fC"±"Š ...Q±è..DŌqp...•Id -.ß¼yİİ>ß.÷~kY½İ Ýgİ}Ō°.•üf.ĀLX.€ .ıX.áčĀ^<g...đ ..l.àp³³B.øF.™. øĒl™.ø.½°.•ûŋ*Ō? ĒĀ.ÿY"¹Y"¹.P.æSdl.c' øÜ\..É8=wœ%.ŌÉ'¶ 4MİŌJİ"Y,2V"sò,[ŏ™e.9ó2,,<.Ēšİæ đäü'ã.9.¾Ē'`..ç. ø¹2¾&cftİ†@Ēoã±. N6.(Ū.æsSdl.c' (2,•ăy.àHÉ_đŌ/Xİ İ.Ē.Āİİz..\$Œ.&\ S†•".<ăİİMç<ĀİŌ. 7•#â1ø™.Y.âr.fİü Y.ym.²";ø8980m•m ¾(Ō[]ü>'÷v•Λ,,ŋİ. D.øĀöw~™.436... °.•leµŪú†mi...].Ē.P »ý†İ'/.Š²¾u.}q.° ΛRĀâ,g+«ŪŪ\K.Yk)/èıüŸ.□C_ İR¾Yİ ăaxó'8't1CĀ7nfz DĀĒİâpù.æY†ø..þu ...ü\$¾/"EDĒ L.L •µ[Ē.™B†øYŠø. ĀþŪ¹•%Ūø.Đ•X.Ş! .>@~..(*.•.{d+Đİ} .ÆG.Ūİ<Ŋ™~.Ūİ,þ} w LpĒ.Š□ŽcGD2 .Q İİšŪZ.4•E@.è@.è ..À.¶Ā..à..A(ˆ .g`lâ, .•D...€µ .,- .u .4f6p. t•cà48...Ē.Ū.RŌ .ž€)đ.İ@.,...Ē.R† t•ĒĒ.²...X.ă..C.P ."%CBH..@ē R' .ă †êİfè[è(t.°..C. Qh.ú.z.#0.ĀZ°.1 .³`08.ž,,.ĀĒđ28.. , .Ā.p. .î,,ŌĀ.à.X .?.Ş.€..:ç<Ō..ĀF B'x\$.!«•.x.İ@Ū• .x.¹ŠH'ŞĒ[...ĒE1P L </pre> |
| A operação é bem-sucedida. | | |
| ⇐ PP | <pre> 4D 4C 52 30 30 30 </pre> | MLR000 |

SPE continua a carga dos dados, novamente dividindo o comando em dois blocos (CMD_BLK1 e CMD_BLK2), ambos de 436 bytes.

| | | |
|-------|--|---|
| SPE ⇒ | 4D 4C 52 34 33 36 00 0F 01 B0 94 0B CA 1F 15 85 E2 A2 96 A1 56 A1 36 A3 AA 51 07 50 9D A8 3E D4 55 D4 28 6A 0A F5 11 4D 46 6B A2 CD D1 CE E8 00 74 2C 3A 19 9D 8B 2E 46 57 A0 9B D0 1D E8 B3 E8 11 F4 38 FA 15 06 83 A1 63 8C 31 8E 18 7F 4C 1C 26 15 B3 02 B3 19 B3 1B D3 8E 39 85 19 C6 8C 61 A6 B1 58 AC 3A D6 1C EB 8A 0D C5 72 B0 62 6C 31 B6 0A 7B 10 7B 12 7B 05 3B 8E 7D 83 23 E2 74 70 B6 38 5F 5C 3C 4E 88 2B C4 55 E0 5A 70 27 70 57 70 13 B8 19 BC 12 DE 10 EF 8C 0F C5 F3 F0 CB F1 65 F8 46 7C 0F 7E 08 3F 8E 9F 21 28 13 8C 09 AE 84 48 42 2A 61 2D A1 92 D0 46 38 4B B8 4B 78 41 24 12 F5 88 4E C4 70 A2 80 B8 86 58 49 3C 44 3C 4F 1C 25 BE 25 51 48 66 24 36 29 81 24 21 6D 21 ED 27 9D 22 DD 22 BD 20 93 C9 46 64 0F 72 3C 59 4C DE 42 6E 26 9F 21 DF 27 BF 51 A0 2A 58 2A 04 28 F0 14 56 2B D4 28 74 2A 5C 51 78 A6 88 57 34 54 F4 54 5C AC 98 AF 58 A1 78 44 71 48 F1 A9 12 5E C9 48 89 AD C4 51 5A A5 54 A3 74 54 E9 86 D2 B4 32 55 D9 46 39 54 39 43 79 B3 72 8B F2 05 E5 47 14 2C C5 88 E2 43 E1 51 8A 28 FB 28 67 28 63 54 84 AA 4F 65 53 B9 D4 75 D4 46 EA 59 EA 38 0D 43 33 A6 05 D0 52 69 A5 B4 6F 68 83 B4 29 15 8A 8A 9D 4A B4 4A 9E 4A 8D CA 71 15 29 1D A1 1B D1 03 E8 E9 F4 32 FA 61 FA 75 FA 3B 55 2D 55 4F 55 BE EA 26 D5 36 D5 2B AA AF D5 E6 A8 79 A8 F1 D5 4A D4 DA D5 46 D4 DE A9 33 D4 7D D4 D3 D4 B7 A9 77 A9 DF D3 40 69 98 69 84 6B 34 33 36 00 0F 01 B0 E4 6A EC D1 38 AB F1 74 0E 6D 8E CB 1C EE 9C 92 39 87 E7 DC D6 84 35 CD 34 23 34 57 68 EE D3 1C D0 9C D6 D2 D6 F2 D3 CA D2 AA D2 3A A3 F5 54 9B AE ED A1 9D AA BD 43 FB 84 F6 A4 0E 55 C7 4D 47 A0 B3 43 E7 A4 CE 63 86 0A C3 93 91 CE A8 64 F4 31 A6 74 35 75 FD 75 25 BA F5 BA 83 BA 33 7A C6 7A 51 7A 85 7A ED 7A F7 F4 09 FA 2C FD 24 FD 1D FA BD FA 53 06 3A 06 21 06 05 06 AD 06 B7 0D F1 86 2C C3 14 C3 5D 86 FD 86 AF 8D 8C 8D 62 8C 36 18 75 19 3D 32 56 33 0E 30 CE 37 6E 35 BE 6B 42 36 71 37 59 66 D2 60 72 CD 14 63 CA 32 4D 33 DD 6D 7A D9 0C 36 B3 37 4B 31 AB 31 1B 32 87 CD 1D CC 05 E6 BB CD 87 2D D0 16 4E 16 42 8B 06 8B 1B 4C 12 D3 93 99 C3 6C 65 8E 5A D2 2D 83 2D 0B 2D BB 2C 9F 59 19 58 C5 5B 6D B3 EA B7 FA 68 6D 6F 9D 6E DD 68 7D C7 86 62 13 68 53 68 D3 63 F3 AB AD 99 2D D7 B6 C6 F6 DA 5C F2 5C DF B9 AB E7 76 CF 7D 6E 67 6E C7 B7 DB 63 77 D3 9E 6A 1F 62 BF C1 BE D7 FE 83 83 A3 83 C8 A1 CD 61 D2 D1 C0 31 D1 B1 D6 F1 06 8B C6 0A 63 6D 66 9D 77 42 3B 79 39 AD 76 3A E6 F4 D6 D9 C1 59 EC 7C D8 F9 17 17 A6 4B 9A 4B 8B CB A3 79 C6 F3 F8 F3 1A E7 8D B9 EA B9 72 5C EB 5D A5 6E 0C B7 44 B7 BD 6E 52 77 5D 77 8E 7B 83 FB 03 0F 7D 0F 9E 47 93 C7 84 A7 A9 67 AA E7 41 CF 67 5E D6 5E 22 AF 0E AF D7 6C 67 F6 4A F6 29 6F C4 DB CF BB C4 7B D0 87 E2 13 E5 53 ED 73 DF 57 CF D9 B7 D5 77 CA CF DE 6F | MLR436... °.Ê.... âç•jVj6fªQ.P.°>Ô UÔ(j.ô.MFkçIÑîè. t,:.•<.FW >Đ.è³è .ô8ú..fjçE1Ž.□L. &.³.³.³.ÓŽ9...ÆEa ±X-:ô.èŠ.År°bll ¶.{.{.{.;Ž}f#âtp ¶8_<N +AUâzp'pw p..¼.p.iE.ÁôðËñ eøF .~.?Žÿ!(.E.® „HB*a.j'ĐF8K,KxA \$.ô^NÄpçE.†XI<D< O.%%QHF\$6)•\$!m! í'•"Ý"½.“ÉFd.r<Y LpBn&ÿ!ß'¿Q *X*. (ð.v+ô(t*Qx ^w4 TôT\~ XjxDqHñ@. ΛÉH%-AQZÿTftTéTô `2UÛF9T9Cy³r<ò.ã G.,A^âcáQŠ(û(g(c T,,ªOeS'ÔuOFÊYè8. C3 .ĐRiÿ'ohf').Š Š.J JŽJ.Êg.).j.i.Ň .ééô2úaúúú;U•UOU %é&ô6ô+ª~ôæ y ñô JÔÚÔFÔp@3ô}ôôô.© w@ßô@i~i,,k436... °äjîÑ8«ñt.mžĚ.îæ '9†cÜö,,5I4#4whîó .Đæôôôôôôôªô: fôT >®íj.ª½Cû,,öª.UçM G ³CçªIç†.Ā“‘Ī d ô1!t5uýu%ªôªfª3z ÄzQz...ziz÷ô.ú,yšý .ú½ús...!...-... ñ†,Ā.Ā]†††~•E•BĀ 6.u.=2V3.0Ī7n5¾k B6q7Yfô`rÍ.cĒ2M3 YmzÛ.6³7K1«1.2†Ī .Ī.æ»Ī†•Đ.N.B<.< .L.ô™Āležžô.f•. •»,ÿY.XĀ[mªè.úhm o•nYh}ç†b.hshócé «-™.x¶Āôú\ò\ß'«ç vĪ}ngnç.Úcwôžj.b žĀ¾xpffjffĒjĪaôÑĀ lÑ±ôñ.<Ā.cmf•wB; y9-v:æôôUĀYi ôÛ. .!KŠK<ĒfyĀóóó.ç• iê'r\è]ŷn..D.½nR w wž{fũ..}žG“ç,, Š@gªçĀĪgĀôĀ™".x ĪgôJô)ôĀŪĪ»Ā{Đ†â .āsísßwĪ7Ū•ôwĪP o |
|-------|--|---|

A operação é bem-sucedida.

| | | |
|------|-------------------|--------|
| ⇐ PP | 4D 4C 52 30 30 30 | MLR000 |
|------|-------------------|--------|

SPE continua a carga dos dados, novamente dividindo o comando em dois blocos (CMD_BLK1 e CMD_BLK2), ambos de 436 bytes.

| | | |
|-----------------------------------|---|--|
| <p>SPE ⇒</p> | <pre> 4D 4C 52 34 33 36 00 0F 01 B0 85 DF 29 7F B4 7F 90 FF 36 FF 1B 01 5A 01 DC 80 E6 80 A9 40 C7 C0 95 81 7D 41 A4 A0 05 41 D5 41 0F 82 CD 82 45 C1 3D 21 70 48 60 C8 F6 90 BB F3 0D E7 0B E7 77 85 82 D0 80 D0 ED A1 F7 C2 8C C3 96 85 7D 1F 8E 09 0F 0B AF 09 7F 18 61 13 51 10 D1 BF 80 BA 60 C9 82 96 05 AF 22 BD 22 CB 22 EF 44 99 44 49 A2 7A A3 15 A3 13 A2 9B A3 5F C7 78 C7 94 C7 48 63 AD 62 57 C6 5E 8A D3 88 13 C4 75 C7 63 E3 A3 E3 9B E2 A7 17 FA 2C DC B9 70 3C C1 3E A1 38 E1 FA 22 E3 45 79 8B 2E 2C D6 58 9C BE F8 F8 12 C5 25 9C E3 45 79 8B 2E 2C D6 58 9C BE F8 F8 12 C5 25 9C 25 47 12 D1 89 31 89 2D 89 EF 39 A1 9C 06 CE F4 D2 80 A5 B5 4B A7 B8 6C EE 2E EE 13 9E 07 6F 07 6F 92 EF CA 2F E7 4F 24 B9 26 95 27 3D 4A 76 4D DE 9E 3C 99 E2 9E 52 91 F2 54 C0 16 54 0B 9E A7 FA A7 D6 A5 BE 4E 0B 4D DB 9F F6 29 3D 26 BD 3D 03 97 91 98 71 54 48 11 A6 09 FB 32 B5 33 F3 32 87 B3 CC B3 8A B3 A4 CB 9C 97 ED 5C 36 25 0A 12 35 65 43 D9 8B B2 BB C5 34 D9 CF D4 80 C4 44 B2 5E 32 9A E3 96 53 93 F3 26 37 3A F7 48 9E 72 9E 30 6F 60 B9 D9 F2 4D CB 27 F2 7D F3 BF 5E 81 5A C1 5D D1 5B A0 5B B0 B6 60 74 A5 E7 CA FA 55 D0 AA A5 AB 7A 57 EB AF 2E 5A 3D BE C6 6F CD 81 B5 84 B5 69 6B 7F 28 B4 2E 2C 2F 7C B9 2E 66 5D 4F 91 56 D1 9A A2 B1 F5 7E EB 5B 8B 15 8A 45 C5 37 36 B8 6C A8 DB 88 DA 28 D8 38 B8 69 EE A6 AA 4D 1F 4B 78 25 17 4B AD 4B 2B 4A DF 6F E6 6E BE F8 95 CD 57 95 5F 7D DA 92 B4 65 34 33 36 00 0F 01 B0 B0 CC A1 6C CF 56 CC 56 E1 D6 EB DB DC B7 1D 28 57 2E CF 2F 1F DB 1E B2 BD 73 07 63 47 C9 8E 97 3B 97 EC BC 50 61 57 51 B7 8B B0 4B B2 4B 5A 19 5C D9 5D 65 50 B5 B5 EA 7D 75 4A F5 48 8D 57 4D 7B AD 66 ED A6 DA D7 BB 79 BB AF EC F1 D8 D3 56 A7 55 57 5A F7 6E AF 60 EF CD 7A BF FA CE 06 A3 86 8A 7D 98 7D 39 FB 1E 36 46 37 F6 7F CD FA BA B9 49 A3 A9 B4 E9 C3 7E E1 7E E9 81 88 03 7D CD 8E CD CD 2D 9A 2D 65 AD 70 AB A4 75 F2 60 C2 C1 CB DF 78 7F D3 DD C6 6C AB 6F A7 B7 97 1E 02 87 24 87 1E 7F 9B F8 ED F5 C3 41 87 7B 8F B0 8E B4 7D 67 F8 5D 6D 07 B5 A3 A4 13 EA 5C DE 39 D5 95 D2 25 ED 8E EB 1E 3E 1A 78 B4 B7 C7 A5 A7 E3 7B CB EF F7 1F D3 3D 56 73 5C E5 78 D9 09 C2 89 A2 13 9F 4E E6 9F 9C 3E 95 75 EA E9 E9 E4 D3 63 BD 4B 7A EF 9C 89 3D 73 AD 2F BC 6F F0 6C D0 D9 F3 E7 7C CF 9D E9 F7 EC 3F 79 DE F5 FC B1 0B CE 17 8E 5E 64 5D EC BA E4 70 A9 73 C0 7E A0 E3 07 FB 1F 3A 06 1D 06 3B 87 1C 87 BA 2F 3B 5D EE 19 9E 37 7C E2 8A FB 95 D3 57 BD AF 9E BB 16 70 ED D2 C8 FC 91 E1 EB 51 D7 6F DE 48 B8 21 BD C9 BB F9 E8 56 FA AD E7 B7 73 6E CF DC 59 73 17 7D B7 E4 9E D2 BD 8A FB 9A F7 1B 7E 34 FD B1 5D EA 20 3D 3E EA 3D 3A F0 60 C1 83 3B 63 DC B1 27 3F 65 FF F4 7E BC E8 21 F9 61 C5 84 CE 44 F3 23 DB 47 C7 26 7D 27 2F 3F 5E F8 78 FC 49 D6 93 99 A7 C5 3F 2B FF 5C FB CC E4 D9 77 BF 78 FC 32 30 15 3B 35 </pre> | <pre> MLR436...°...ß)□□ •ÿÿ..Z.Ü€€@QÇÀ ••}Aα .AÖA.,Í,ÉÁ =!pH`Èö•»ó.Ç.ÇW... ,ÐÉÍ¡¡÷ÁÆÁ•...}.Ž. ..-□.a.Q.Ñ¿€°`É ,.-"½"E"iD™DIÇz f.f.Ç>f_CxÇ"CHC- bwΛΛSÓ^`AuÇcãfã> âŞ.ú,U'p<Á>¡ðáú" ãEy<.,ÖXœ¼øø.Å%œ %G.N%1%•%i9¡e.ÎÖ Ö€¥µKŞ,lí.î.ž.o. o'îË/CO\$'Λ&.'=JvM pž<™ázR'òTA.T.žŞ úŞÖ¥¼N.MÜYö)=&½= .·'~gTH.!.ú2µ3ó2 ‡³I³S³αÈœ.í\6%. 5eCÜ<²>»Á4ÜÎÖ€AD² Λ2šã.s"ó&7:÷Hžrž 0o'¹UòMÉ'ò}ó¿Λ.Z Á]Ñ[[°¶] t¥ÇÉúUÐ ª¥«zwē-.Z=¼ÆoÍ•µ „mik□(´.,/ ¹.f)O 'VNŞÇ±ö~è[<.ŞEA7 6,1'Ö'Ú(Ø8,ii!αM .Kx%.K-K+Jßoæ¾ø •Íw_}Ú' e436%. °°i¡IiviváöeÜü.. (w.í/.Ü.²½s.cgÉŽ •;•¼PawQ.◊°K²KZ .\Ü]ePµµê}uJÖH•W M{-fí!Úx»y}~iñÖÓ VŞUWZ÷n`íiz¿úÍ. f†Ş}~}9û.6F7ó□Íú °¹If@`éÁ~á~é•} ÍŽÍÍ•š•e-p«αuò`Á ÁÈßx□ÓYÆ]«oŞ••• ‡\$‡.□>øíöÁA‡f•°ž ~}gø]m.µfα.è\p9Ö •0%ížē.>.x'·Ç¥Şã {Ëi÷.Ö=Vs\áxÚ.Á% Ç.YNæYœ>•uêééäÖc ½Kziœ%=s-/¼oðlÐÜ óç i.é÷i?ypöü±.î .ŽΛd]i°áp@sA~ã. û.:.:.;‡.‡°/;]î. ž7 âšû•öw½~ž».pí ÒËÛ'áéQxopH,!½É» ùèVú-ç.snIÛYs.}• ãž0½Súš÷.~4ý±]ê =>é=:ð`Áf;cÜ±'?e ÿô~¼è!úaA,,ÍDó#ÜG Ç&}'/?ΛøxúIÖ"™ŞÁ ?+ÿ\úIäUw¿xú20.; 5 </pre> |
| <p>A operação é bem-sucedida.</p> | | |
| <p>← PP</p> | <pre>4D 4C 52 30 30 30</pre> | <p>MLR000</p> |

SPE finaliza a carga dos dados, desta vez dividindo o comando em dois blocos (CMD_BLK1 e CMD_BLK2) de 436 e 314 bytes.

| | | |
|----------------------------|--|--|
| SPE ⇒ | <pre> 4D 4C 52 34 33 36 00 0F 01 B0 FE 5C F4 FC D3 AF 9B 5F A8 BF D8 FF D2 EE 65 EF 74 D8 F4 FD 57 19 AF 66 5E 97 BC 51 7F 73 E0 2D EB 6D FF BB 98 77 13 33 B9 EF B1 EF 2B 3F 98 7E E8 F9 18 F4 F1 EE A7 8C 4F 9F 7E 03 F7 84 F3 FB 8F 70 66 2A 00 00 00 09 70 48 59 73 00 00 0B 12 00 00 0B 12 01 D2 DD 7E FC 00 00 02 75 49 44 41 54 78 9C ED 9D 41 8E 83 30 0C 00 B7 12 FF FF 72 F7 EE 43 90 6B 27 63 D0 CC 35 10 CA C8 92 95 E0 B8 D7 DF 29 BE DF 6F D7 54 9F CF E7 E7 07 85 7B C3 C5 EB 99 1B B9 CE 3C 46 02 7A 67 D0 3B 83 DE 19 A2 F7 63 D9 AF C2 3A 19 36 3E 77 9F 0D E3 9D 41 EF 0C 7A 67 D0 3B C3 8D F7 54 8E 4A 65 A1 D4 BA B1 71 B4 42 A3 0D E3 9D 41 EF 0C 7A 67 D0 3B C3 50 EF A9 ED D9 D4 68 63 9A AD 30 D4 FB EB D1 3B 83 DE 19 F4 CE 30 D4 7B 2A 19 EE BB 78 1F 43 BD BF 1E BD 33 E8 9D 41 EF 0C 37 DE 87 AC EE 02 95 DA A3 CA 1B 35 DA 30 DE 19 F4 CE A0 77 06 BD 33 44 EF C7 0A 62 8F 51 F9 FA BA CF 86 F1 CE A0 77 06 BD 33 E8 9D E1 1A B2 22 A5 4A 91 02 C7 6C 18 EF 0C 7A 67 D0 3B 83 DE 19 AE 63 E7 39 C3 CC D4 27 D3 63 0B 54 EB 81 27 A2 77 06 BD 33 E8 9D E1 66 1F 78 DF 01 97 F5 C5 C7 DA 35 EC 5B 27 AF EF 35 DE 19 F4 CE A0 77 06 BD 33 74 9E 5F 6D BC 77 4D A5 2F 44 65 BD DA 38 B3 F1 CE A0 77 06 BD 33 E8 9D 21 57 0F 5C 69 8C D0 98 B2 F6 ED 5D 37 AE 57 D7 53 19 EF 0C 7A 67 D0 3B 83 DE 19 62 DD 52 2A 29 55 52 65 A0 92 B2 1A A7 AA B4 98 48 BD 33 31 34 00 0F 01 36 AF F1 CE A0 77 06 BD 33 E8 9D 21 57 B7 44 75 0C 6C 3C A1 5A F9 AE DB 38 6A BC 33 E8 9D 41 EF 0C 7A 67 E8 F4 DE B8 40 4D 8D 56 FA EE 1F 3B CE EA 3E F0 08 F4 CE A0 77 06 BD 33 E4 BC 0F 29 09 DA D7 51 69 4D E5 67 98 57 47 A0 77 06 BD 33 E8 9D 21 E7 7D C8 E1 CF F5 C5 33 0F E5 B8 0F 3C 02 BD 33 E8 9D 41 EF 0C 37 FD 96 1A CB 89 02 8D 55 BB A9 D1 C6 23 A9 95 57 30 DE 19 F4 CE A0 77 06 BD 33 C4 BA A5 7D 54 FA 2D A5 66 AE D0 58 89 B5 7E 05 E3 9D 41 EF 0C 7A 67 D0 3B 43 F4 7E EC 5F C0 1B 8F 86 1E 9B B9 72 AF DF 57 47 A0 77 06 BD 33 E8 9D A1 B3 DF 52 63 4E 1E D2 8F 22 85 E7 57 1F 80 DE 19 F4 CE A0 77 86 29 DE 1B EB 81 D7 F7 A6 9E 9B 9A D9 BC FA 00 F4 CE A0 77 06 BD 33 3C D2 FB B1 3F DD B1 8F E1 DB D0 3B 83 DE 19 F4 CE 90 EB 0F DC C8 BE 92 A0 CA D7 D7 7D 7D F7 03 C6 3B 83 DE 19 F4 CE A0 77 86 9B FF 89 DB 47 63 8E AA 9C 23 4D 4D B5 C6 7D E0 07 A0 77 06 BD 33 E8 9D E1 1F AC 1F 66 FE AE F3 F7 6D 00 00 00 49 45 4E 44 AE 42 60 82 </pre> | <pre> MLR436... °p\ôüó~ >_ ``zÿöïeítôöýw. ~fΛ.¼Q□sà•ëmÿ»~w .3'î±î+?~èù.ôñî §œOÿ~.÷,,ôû•pf*. ..pHYS.....ô ÿ~ü...uIDATxéí•A Žf0...ÿÿr÷îC•k' cDİ5.ÊÊ'•à xß)¼ß oxTYİçç....{ÅÄë™.¹ Î<F.zgĐ;fp.ç÷cÜ~ Â:.6>wÿ.ã•Aï.zgĐ ;Ã•÷TZJe;ô°±q Bf .ã•Aï.zgĐ;ÃPi@íÜ ôhcš-0ôüëN;fp.ôî 00{*•î>x.C½j.½3è •Aï.7p†-î.•ÚfÊ.5 Ú0p.ôî w.½3Dİç.b •Quú°î†ñî w.½3è• á.²"¥J'.çl.î.zgĐ ;fp.°cc9ÁİÖ'óc.T è.'çw.½3è•áf.xß. •ôÅÇU5î['~î5p.ô î w.½3tž_m¼wM¥/D e½U8³ñî w.½3è!W .\iœ~²öí]7°wxs. î.zgĐ;fp.bYR*)UR e'².šã~H½314.. .6~ñî w.½3è!W•D u.l<j;Zù°U8j¼3è•A î.zgeôp,@M•Vúî.; îê>ð.ôî w.½3ã¼.) .ÚxQiMäg~WG w.½3 è!ç}ÉáİôÁ3.ã.< .½3è•Aï.7ý.•Ê%• U»@ÑÅ#@•w0p.ôî w .½3Å°¥}Tú•¥f°DX% µ~.ã•Aï.zgĐ;Cô~î _À.†.›¹r_ßWG w. ½3è•j³BRCN.ô•"...ç w.€p.ôî w†)p.ë•x ÷!ž>šÜ¼ú.ôî w.½3 <ôú±?Y±•áÜĐ;fp.ô î•ë.ÜÊ¾' Êxx}}÷. Æ;fp.ôî w†>ÿ%ÜGc Zªœ#MMµÆ}ã. w.½3 è.á.-.fp°ó÷m.... IEND®B` , </pre> |
| A operação é bem-sucedida. | | |
| ⇐ PP | <pre> 4D 4C 52 30 30 30 </pre> | MLR000 |

3.4.3. Comando “MLE”

| |
|---|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando finaliza o processo de carga de um arquivo multimídia iniciado pelo comando “MLI”. Ao recebê-lo, o pinpad confere os dados recebidos através dos comandos “MLR”, acatando ou não o arquivo.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|------------------------------|
| CMD_ID | M | Código do comando (= “MLE”). |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|--|
| RSP_ID | M | Código da resposta (= “MLE”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_INVCALL..... Comando “MLI” não foi executado previamente. ↪ ST_MFERR..... Tamanho do arquivo recebido ou CRC não correspondem às informações fornecidas no comando “MLI” (SPE_MFINFO). ↪ ST_INTERR..... Falta de memória para gerenciamento ou armazenamento do arquivo recebido. |

➔ Exemplos

| | | |
|--|-------------------|--------|
| SPE indica a finalização da carga do arquivo multimídia. | | |
| SPE ⇒ | 4D 4C 45 | MLE |
| A operação é bem-sucedida. | | |
| ← PP | 4D 4C 45 30 30 30 | MLE000 |

3.4.4. Comando “LMF”

- | |
|---|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando retorna uma lista com os nomes dos arquivos multimídia carregados no pinpad.

Se não houver arquivos carregados, o comando é bem-sucedido e a lista retornada é vazia. Não há ordem específica para a montagem da lista, dependendo exclusivamente das características de implementação pinpad.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|------------------------------|
| CMD_ID | M | Código do comando (= “LMF”). |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|-------------------------------|
| RSP_ID | M | Código da resposta (= “LMF”). |
| RSP_STAT | M | Ver seção 3.1.1. |
| PP_MFNAME | O | Nome do arquivo carregado. |
| PP_MFNAME | O | Nome do arquivo carregado. |
| ... | .. | ... |
| PP_MFNAME | O | Nome do arquivo carregado. |

➔ Exemplos

SPE solicita a lista de arquivos carregados no pinpad.

| | | |
|-------|----------|-----|
| SPE ⇒ | 4C 4D 46 | LMF |
|-------|----------|-----|

A operação é bem-sucedida, retornando os nomes de 5 arquivos multimídia.

| | | |
|------|--|---|
| ← PP | 4C 4D 46 30 30 30 30 36 30 80 5E 00 08 53 49 47 4E 41 4C 53 20 80 5E 00 08 50 52 45 53 54 4F 20 20 80 5E 00 08 51 52 43 4F 44 45 30 31 80 5E 00 08 46 45 45 44 42 41 43 4B 80 5E 00 08 4D 4F 56 4E 50 49 43 54 | LMF000060€^..SIG NALS•€^..PRESTO• •€^..QRCODE01€^. .FEEDBACK€^..MOV NPICT |
|------|--|---|

3.4.5. Comando “DMF”

| |
|---|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando exclui um ou mais arquivos multimídia armazenados no pinpad.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|---------------------------------|
| CMD_ID | M | Código do comando (= “DMF”). |
| SPE_MFNAME | M | Nome do arquivo a ser excluído. |
| SPE_MFNAME | O | Nome do arquivo a ser excluído. |
| ... | ... | ... |
| SPE_MFNAME | O | Nome do arquivo a ser excluído. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|--|
| RSP_ID | M | Código da resposta (= “DMF”). |
| RSP_STAT | M | Ver seção 3.1.1. Este comando <u>não retorna erro</u> caso um ou mais arquivos já estejam ausentes no pinpad. |

➔ Exemplos

SPE solicita a exclusão de dois arquivos multimídia no pinpad.

| | | |
|-------|--|------------------------------------|
| SPE ➔ | 44 4D 46 30 32 34 00 1E 00 08 54 45 53 54 45 43 48 4F 00 1E 00 08 4D 4F 56 4E 50 49 43 54 | DMF024....TESTEC HO....MOVNPICT |
|-------|--|------------------------------------|

A operação é bem-sucedida.

| | | |
|------|-------------------|--------|
| ← PP | 44 4D 46 30 30 30 | DMF000 |
|------|-------------------|--------|

3.4.6. Comando “DSI”

| |
|---|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECs |

Este comando apresenta no *display* um arquivo multimídia previamente carregado no pinpad. O conteúdo será centralizado no *display* caso suas dimensões sejam inferiores à capacidade do equipamento.

O *display* do pinpad é previamente apagado, sendo que mensagens ou imagens anteriores não são mantidas.

Este comando sempre retorna imediatamente (é não blocante), mesmo se o arquivo multimídia contiver animação (ou vídeo), que será apresentada enquanto o pinpad não recebe um novo comando.

⚠ Os pinpads não são obrigados a suportar todos os formatos de arquivo multimídia previstos por esta especificação. O SPE deve obter a informação dos formatos suportados através do comando “GIX” (parâmetro PP_MFSUP).

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|------------------------------------|
| <u>CMD_ID</u> | M | Código do comando (= “DSI”). |
| <u>SPE_MFNAME</u> | M | Nome do arquivo a ser apresentado. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|-----------------|----------|---|
| <u>RSP_ID</u> | M | Código da resposta (= “DSI”). |
| <u>RSP_STAT</u> | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ST_MFNFOUND..... Arquivo multimídia não está presente no pinpad. ↪ST_MFERRFMT Formato do arquivo não aceito pelo pinpad, ou suas dimensões superam a capacidade do <i>display</i> . |

➔ Exemplos

| | | |
|--|--|-----------------------|
| SPE solicita a apresentação do arquivo de nome “QRCODE01”. | | |
| SPE ⇒ | 44 53 49 30 31 32 00 1E 00 08 51 52 43 4F 44 45 30 31 | DSI012...QRCODE 01 |
| A operação é bem-sucedida. | | |
| ← PP | 44 53 49 30 30 30 | DSI000 |

3.5. Comandos para manutenção de Tabelas EMV

Conforme detalhado no **Capítulo 4**, o pinpad deve armazenar diversas tabelas que são utilizadas no processamento de cartões EMV (ICC ou CTLS).

Esta seção descreve os comandos utilizados para gerenciamento e carga destas tabelas no pinpad:

| CMD_ID | Significado | Obsoleto | Blocante | Abecs |
|--------|------------------------------------|-------------------------------------|--------------------------|--------------------------|
| "GTS" | <i>Get Table Version</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "TLI" | <i>Table Load - Initialization</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "TLR" | <i>Table Load - Record</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "TLE" | <i>Table Load - End</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

3.5.1. Comando “GTS”

Obsoleto
 Blocante
 ABECS

Este comando obtém a versão das Tabelas EMV carregadas no pinpad. Para mais informações, ver a **seção 4.2**.

 Este comando é **obsoleto**. Para esta funcionalidade, o SPE deve usar o comando “**GIX**” com **PP_TABVERnn**.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| CMD_ID | A3 | Código do comando (= “GTS”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “002”). |
| GTS_ACQIDX | N2 | Identificador da Rede Credenciadora referente às Tabelas EMV cuja versão está sendo requisitada. Deve-se usar o valor “00” quando se utiliza uma versão de tabelas única para todas as redes (isso só faz sentido se as tabelas foram carregadas usando-se também “00” no comando “ TLI ”). |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “GTS”). |
| RSP_STAT | N3 | Ver seção 3.1.1 . |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “010”). |
| GTS_TABVER | A10 | Versão atual das Tabelas EMV referentes à Rede Credenciadora (ou do conjunto total de tabelas se GTS_ACQIDX = “00”). Caso não exista tabela carregada para a rede informada, este campo retorna <u>zeros</u> (“0000000000”). Caso as tabelas tenham sido carregadas de forma isolada para as diferentes Redes Credenciadoras (com diferentes versões) e GTS_ACQIDX = “00”, este campo retorna <u>zeros</u> (“0000000000”), uma vez que não existe uma versão “geral” para as tabelas. |

➔ Exemplos

O SPE solicita a versão das Tabelas EMV da Rede Credenciadora de índice “02”.

| | | |
|-------|-------------------------|----------|
| SPE ➔ | 47 54 53 30 30 32 30 32 | GTS00202 |
|-------|-------------------------|----------|

O pinpad retorna a versão "XEMVST0003".

| | | |
|------|---|-------------------------|
| ← PP | 47 54 53 30 30 30 30 31 30 58 45 4D 56 53 54 30 30 30 33 | GTS000010XEMVST0 003 |
|------|---|-------------------------|

3.5.2. Comando “TLI”

| |
|-----------------------------------|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

Este comando inicia o processo de carga (ou atualização) de tabelas. Caso ele retorne ↩ST_OK ou ↩ST_TABVERDIF, o processo pode continuar através dos comandos “TLR” e “TLE”.

➡ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| CMD_ID | A3 | Código do comando (= “TLI”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “012”). |
| TLI_ACQIDX | N2 | Identificador da Rede Credenciadora cujas Tabelas EMV serão atualizadas Para abranger <u>todas as redes</u> , deve-se usar o valor “00”. |
| TLI_TABVER | A10 | Nova versão das Tabelas EMV que serão carregadas (formato <u>totalmente livre</u> criado pelo SPE). |

➡ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= “TLI”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↩ST_OKProcesso de carga iniciado, porém TLI_TABVER coincide com a versão atual das Tabelas EMV já carregadas. ↩ST_TABVERDIFProcesso de carga iniciado, porém TLI_TABVER difere da versão atual das Tabelas EMV já carregadas. |

➡ Exemplos

O SPE solicita a carga completa de tabelas (todas as Redes Credenciadoras), informando a nova versão de Tabelas EMV (“TABVER0008”).

| | | |
|--|--|--------------------|
| SPE ⇒ | 54 4C 49 30 31 32 30 30 54 41 42 56 45 52 30 30 30 38 | TLI01200TABVER0008 |
| O pinpad inicia o processo com sucesso, informando que a versão fornecida difere da atual. | | |
| ← PP | 54 4C 49 30 32 30 | TLI020 |

3.5.3. Comando “TLR”

Obsoleto
 Blocante
 ABECS

Este comando carrega um ou mais registros das Tabelas EMV. O pinpad armazena os registros de forma temporária para preservar as tabelas atuais em caso de erro na operação de atualização, que é finalizada pelo comando “TLI”.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= “TLR”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| TLR_NREC | N2 | Quantidade de <u>registros</u> a seguir. |
| ---- | ??? | Um ou mais registros concatenados, cada um iniciado pela informação de tamanho, conforme formato descrito na seção 4.1 . Ao concatenar os registros, deve-se atentar ao tamanho máximo permitido por CMD_LEN1 (“999”). |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|-----------------|---------|--|
| RSP_ID | A3 | Código da resposta (= “TLR”). |
| <u>RSP_STAT</u> | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_INVCALLComando “ <u>TLI</u> ” não foi chamado previamente. ↪ ST_TABERRErro ao tentar armazenar os registros (falta de memória, por exemplo). |

➤ Exemplos

O SPE envia os registros "01" e "02" da Tabela de AID da rede "03".

| | | |
|-------|---|--|
| SPE ⇒ | 54 4C 52 36 33 30 30 32 33 31 34 31 30 33 30 31 30 37 41 30 30 30 30 30 30 30 30 34 31 30 31 30 31 43 54 4C 45 53 53 2D 2D 43 52 45 44 49 54 4F 20 30 33 30 30 30 31 30 30 30 32 30 30 30 31 30 37 36 38 34 30 32 30 32 30 35 30 33 30 30 30 30 30 30 30 30 34 30 30 30 30 30 30 30 30 30 30 30 30 45 30 46 38 45 38 37 30 30 30 46 30 46 30 30 31 32 32 32 30 35 30 30 34 41 30 30 30 44 38 30 30 45 38 30 30 30 30 32 30 35 30 30 34 46 38 30 30 30 30 30 30 30 30 30 30 52 30 34 30 30 30 30 31 33 38 37 30 30 30 30 30 35 44 42 30 30 30 30 30 39 43 33 31 32 33 34 30 39 46 30 32 30 36 35 46 32 41 30 32 39 41 30 33 39 43 30 31 39 35 30 35 39 46 33 37 30 34 30 30 30 30 30 30 30 30 30 30 39 46 33 37 30 34 30 59 31 5A 31 59 33 5A 33 46 30 30 30 30 34 38 30 30 30 30 30 30 30 30 30 30 30 30 30 46 30 30 30 30 34 38 30 30 30 33 31 34 31 30 33 30 32 30 37 41 30 30 30 30 30 30 30 30 34 33 30 36 30 32 43 54 4C 45 53 53 2D 2D 44 45 42 49 54 4F 20 20 30 33 30 30 30 31 30 30 30 32 30 30 30 31 30 37 36 38 34 30 32 30 32 30 35 30 33 30 30 30 30 30 30 30 30 34 30 30 30 30 30 30 30 30 30 30 30 30 45 30 46 38 45 38 37 30 30 30 46 30 46 30 30 31 32 32 32 30 35 30 30 34 41 30 30 30 44 38 30 30 45 38 30 30 30 30 32 30 35 30 30 34 46 38 30 30 30 30 30 30 30 30 30 30 52 30 34 30 30 30 30 31 33 38 37 30 30 30 30 30 35 44 42 30 30 30 30 30 30 39 43 33 31 32 33 34 30 39 46 30 32 30 36 35 46 32 41 30 32 39 41 30 33 39 43 30 31 39 35 30 35 39 46 33 37 30 34 30 30 30 30 30 30 30 30 30 30 39 46 33 37 30 34 30 59 31 5A 31 59 33 5A 33 46 30 30 30 30 34 38 30 30 30 30 30 30 30 30 30 30 30 30 30 46 30 30 30 34 38 30 30 30 | TLR6300231410301 07A000000041010 0000000000000000 0001CTLESS--CRED ITO•030001000200 0107684020205030 0000000400000000 0000E0F8E87000F0 F00122205004A000 D800E80000205004 F80000000000R040 0001387000005DB0 00009C3123409F02 065F2A029A039C01 95059F3704000000 00009F3704000000 0000000000000000 00000000000000Y1Z1 Y3Z3F00004800000 0000000F0000480 003141030207A000 0000043060000000 00000000000002CT LESS--DEBITO••03 0001000200010768 4020205030000000 04000000000000E0 F8E87000F0F00122 205004A000D800E8 0000205004F80000 000000R040000138 7000005DB000009C 3123409F02065F2A 029A039C0195059F 370400000000009F 3704000000000000 0000000000000000 000000Y1Z1Y3Z3F0 0004800000000000 00F000048000 |
|-------|---|--|

O pinpad recebe os registros com sucesso.

| | | |
|------|-------------------|--------|
| ⇐ PP | 54 4C 52 30 30 30 | TLR000 |
|------|-------------------|--------|

O SPE envia o registro "13" da Tabela de CAPK da rede "02", seguido dos registros "01", "02" e "03" da Tabela de Certificados Revogados da rede "01".

| | | |
|-------|--|------------------|
| SPE → | 54 4C 52 36 39 31 30 34 36 31 31 32 30 32 31 33 | TLR6910461120313 |
| | 41 30 30 30 30 30 30 30 30 30 34 45 46 30 30 31 30 | A000000004EF0010 |
| | 33 30 30 30 30 32 34 38 41 31 39 31 43 42 38 37 | 30000248A191CB87 |
| | 34 37 33 46 32 39 33 34 39 42 35 44 36 30 41 38 | 473F29349B5D60A8 |
| | 38 42 33 45 41 45 45 30 39 37 33 41 41 36 46 31 | 8B3EAE0973AA6F1 |
| | 41 30 38 32 46 33 35 38 44 38 34 39 46 44 44 46 | A082F358D849FDDF |
| | 46 39 43 30 39 31 46 38 39 39 45 44 41 39 37 39 | F9C091F899EDA979 |
| | 32 43 41 46 30 39 45 46 32 38 46 35 44 32 32 34 | 2CAF09EF28F5D224 |
| | 30 34 42 38 38 41 32 32 39 33 45 45 42 42 43 31 | 04B88A2293EEBBC1 |
| | 39 34 39 43 34 33 42 45 41 34 44 36 30 43 46 44 | 949C43BEA4D60CFD |
| | 38 37 39 41 31 35 33 39 35 34 34 45 30 39 45 30 | 879A1539544E09E0 |
| | 46 30 39 46 36 30 46 30 36 35 42 32 42 46 32 41 | F09F60F065B2BF2A |
| | 31 33 45 43 43 37 30 35 46 33 44 34 36 38 42 39 | 13ECC705F3D468B9 |
| | 44 33 33 41 45 37 37 41 44 39 44 33 46 31 39 43 | D33AE77AD9D3F19C |
| | 41 34 30 46 32 33 44 43 46 35 45 42 37 43 30 34 | A40F23DCF5EB7C04 |
| | 44 43 38 46 36 39 45 42 41 35 36 35 42 31 45 42 | DC8F69EBA565B1EB |
| | 43 42 34 36 38 36 43 44 32 37 34 37 38 35 35 33 | CB4686CD27478553 |
| | 30 46 46 36 46 36 45 39 45 45 34 33 41 41 34 33 | 0FF6F6E9EE43AA43 |
| | 46 44 42 30 32 43 45 30 30 44 41 45 43 31 35 43 | FDB02CE00DAEC15C |
| | 37 42 38 46 44 36 41 39 42 33 39 34 42 41 42 41 | 7B8FD6A9B394BABA |
| | 34 31 39 44 33 46 36 44 43 38 35 45 31 36 35 36 | 419D3F6DC85E1656 |
| | 39 42 45 38 45 37 36 39 38 39 36 38 38 45 46 45 | 9BE8E76989688EFE |
| | 41 32 44 46 32 32 46 46 37 44 33 35 43 30 34 33 | A2DF22FF7D35C043 |
| | 33 33 38 44 45 41 41 39 38 32 41 30 32 42 38 36 | 338DEAA982A02B86 |
| | 36 44 45 35 33 32 38 35 31 39 45 42 42 43 44 36 | 6DE5328519EBBCD6 |
| | 46 30 33 43 44 44 36 38 36 36 37 33 38 34 37 46 | F03CDD686673847F |
| | 38 34 44 42 36 35 31 41 42 38 36 43 32 38 43 46 | 84DB651AB86C28CF |
| | 31 34 36 32 35 36 32 43 35 37 37 42 38 35 33 35 | 1462562C577B8535 |
| | 36 34 41 32 39 30 43 38 35 35 36 44 38 31 38 35 | 64A290C8556D8185 |
| | 33 31 32 36 38 44 32 35 43 43 39 38 41 34 43 43 | 31268D25CC98A4CC |
| | 36 41 30 42 44 46 46 46 44 41 32 44 43 43 41 33 | 6A0BDFFFDA2DCCA3 |
| | 41 39 34 43 39 39 38 35 35 39 45 33 30 37 46 44 | A94C998559E307FD |
| | 44 46 39 31 35 30 30 36 44 39 41 39 38 37 42 30 | DF915006D9A987B0 |
| | 37 44 44 41 45 42 33 42 31 32 31 37 36 36 45 42 | 7DDAEB3B121766EB |
| | 42 30 45 45 31 32 32 41 46 42 36 35 44 37 38 34 | B0EE122AFB65D784 |
| | 35 42 37 33 44 42 34 36 42 41 42 36 35 34 32 37 | 5B73DB46BAB65427 |
| | 41 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 | A000000000000000 |
| | 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 | 0000000000000000 |
| | 30 30 30 30 30 30 30 30 30 30 30 30 32 36 33 30 | 0000000000002630 |
| | 31 30 31 41 30 30 30 30 30 30 30 30 33 30 31 34 | 301A000000003014 |
| | 34 34 34 34 34 30 32 36 33 30 31 30 32 41 30 30 | 4444402630302A00 |
| | 30 30 30 30 30 30 33 39 37 35 35 35 35 35 30 | 0000003975555550 |
| | 32 36 33 30 31 30 33 41 30 30 30 30 30 30 30 | 2630303A00000000 |
| | 33 39 34 36 36 36 36 36 36 | 3946666666 |

O pinpad recebe os registros com sucesso.

| | | |
|------|-------------------|--------|
| ← PP | 54 4C 52 30 30 30 | TLR000 |
|------|-------------------|--------|

3.5.4. Comando “TLE”

| |
|-----------------------------------|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

Este comando finaliza o processo de carga (ou atualização) de tabelas, fazendo com que os registros fornecidos através de “TLR” sejam armazenados de forma definitiva, substituindo as Tabelas EMV anteriores (se existentes). Nesse momento, TLI_TABVER passa a vigorar para as novas tabelas.

Caso o comando “TLR” não seja chamado entre “TLI” e “TLE”, as tabelas da referida rede devem simplesmente ser apagadas.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|------------------------------|
| CMD_ID | A3 | Código do comando (= “TLE”). |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| RSP_ID | A3 | Código da resposta (= “TLE”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_INVCALL Comando “TLI” não foi chamado previamente. ↪ ST_TABERR Erro ao tentar armazenar os registros (falta de memória, por exemplo). |

➔ Exemplos

| | | |
|---|-------------------|--------|
| O SPE solicita a finalização da carga de tabelas. | | |
| SPE ⇒ | 54 4C 45 | TLE |
| O pinpad acata o comando com sucesso, atualizando as tabelas. | | |
| ⇐ PP | 54 4C 45 30 30 30 | TLE000 |

3.6. Comandos de processamento de cartão (obsoletos)

Esta seção detalha comandos de alto-nível responsáveis pelo processamento completo de um cartão durante uma operação de pagamento, seja magnético, ICC ou CTLS.

⚠ Todos os comandos descritos nesta seção são **obsoletos**. Para estas funcionalidades, o SPE deve utilizar os comandos descritos na **seção 3.7**.

Os seguintes comandos estão contemplados nesta seção:

| CMD_ID | Significado | Obsoleto | Blocante | Abecs |
|--------|-------------------------------|-------------------------------------|-------------------------------------|--------------------------|
| "GCR" | <i>Get Card</i> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| "CNG" | <i>Change EMV Parameter</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "GOC" | <i>Go On Chip Processing</i> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| "FNC" | <i>Finish Chip Processing</i> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

3.6.1. Comando “GCR”

Obsoleto
 Blocante
 ABECS

Este comando inicia um processo de transação com cartão de pagamento (seja ele magnético, ICC ou CTLS), conforme apresentado na **seção 3.6.5**.

Ao ser acionado, o pinpad mostra uma mensagem no *display* solicitando a apresentação de um cartão. Caso seja utilizado um cartão com *chip* (ICC ou CTLS), o processamento EMV é iniciado automaticamente. Para isso, o pinpad necessita que as Tabelas EMV estejam carregadas em sua memória (ver **Capítulo 4**).

➔ Comando

| Id. do Campo | Formato | Descrição |
|---------------|---------|---|
| CMD_ID | A3 | Código do comando (= “GCR”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| GCR_ACQIDXREQ | N2 | Identificador da Rede Credenciadora cujas Tabelas EMV serão utilizadas caso seja apresentado um ICC ou CTLS. Para abranger as tabelas de <u>todas as redes</u> , deve-se utilizar o valor GCR_ACQIDXREQ = “00” (ver Observação #1). |
| GCR_APPTYPREQ | N2 | Tipo de aplicação desejada, de forma a considerar somente as Tabelas de AID em que T1_APPTYPE = GCR_APPTYPREQ (ver seção 4.1.1). <ul style="list-style-type: none"> ▪ Para ignorar T1_APPTYPE, deve-se usar o GCR_APPTYPREQ = “99”. ▪ Para utilizar uma lista específica de registros das Tabelas de AID, deve-se usar o GCR_APPTYPREQ = “00” (a lista segue ao final do comando). |
| GCR_AMOUNT | N12 | Valor inicial da transação em centavos (<i>Amount, authorized</i>), devendo ser zero (0) caso este dado não esteja disponível no início da transação. |
| GCR_DATE | N6 | Data da transação (“AAMMDD”) |
| GCR_TIME | N6 | Hora da transação (“HHMMSS”) |
| GCR_TABVER | A10 | Versão <u>esperada</u> das Tabelas EMV referentes à Rede Credenciadora definida em GCR_ACQIDXREQ (ou a versão “geral” das tabelas se GCR_ACQIDXREQ = “00”). |
| GCR_QTDAPP | N2 | Quantidade de entradas na lista a seguir (somente se GCR_APPTYPREQ = “00”). IMPORTANTE: Este campo <u>não é opcional</u> , devendo receber o valor “00” caso não exista a lista a seguir. |
| GCR_IDAPP1 | A4 | Referência direta a um registro das Tabelas de AID, composta da concatenação de TAB_ACQ e TAB_RECIDX . |
| ... | ... | |

| Id. do Campo | Formato | Descrição |
|---------------------------|---------|---|
| GCR_IDAPPn | A4 | Referência direta a um registro das Tabelas de AID, composta da concatenação de TAB_ACQ e TAB_RECIDX . |
| GCR_CTLSON (opcional!) | N1 | Habilita interface de cartão sem contato (ver Observação #2): "1" = Sim (<i>default</i>); "0" = Não. IMPORTANTE: Para manter compatibilidade com sistemas anteriores a esta especificação, este campo é opcional . Um pinpad que suporta CTLS considerará o valor "1" (sim) caso este parâmetro esteja ausente ao final do comando. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= "GCR"). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_MCDATAERR Um cartão magnético foi passado, porém houve erro de leitura (nenhuma trilha pôde ser lida). ↳ ST_TABVERDIF GCR_TABVER difere da versão atual das Tabelas EMV já carregadas. Ver procedimento a seguir em "Comando (depois de ↳ ST_TABVERDIF)" . ↳ ST_CARDINVALIDAT ... Aplicação ICC está invalidada. ↳ ST_CARDBLOCKED ICC ou CTLS está bloqueado. ↳ ST_CARDPROBLEMS ... ICC inválido ou com problemas. ↳ ST_CARDINVDATA ICC com dados inválidos ou faltando. ↳ ST_CARDAPPNAV Modo inválido para o ICC. ↳ ST_CARDAPPNAUT ICC não aceito. ↳ ST_ERRFALLBACK Erro de ICC sujeito a "fallback" para tarja. ↳ ST_CTLSSINVALIDAT CTLS está inválido/bloqueado. ↳ ST_CTLSPROBLEMS CTLS inválido ou com problemas. ↳ ST_CTLSSAPPNAV Modo inválido para o CTLS. ↳ ST_CTLSSAPPNAUT CTLS não aceito. ↳ ST_CTLSEXTCVM Solicitar verificação no dispositivo do portador. ↳ ST_CTLSSIFCHG Mudar interface (usar ICC ou tarja). |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| GCR_CARDTYPE | N2 | Tipo de cartão lido: "00" = Magnético; "03" = ICC EMV; "05" = CTLS simulando tarja; e "06" = CTLS EMV. |

| Id. do Campo | Formato | Descrição |
|--------------|-----------------|---|
| GCR_STATCHIP | N1 | Status da última leitura de ICC. Este dado é usado pelo SPE quando GCR_CARDTYPE for "00" (magnético) e este tem indicação da presença de <i>chip</i> , de modo a recusá-lo ou não. "0" = Bem-sucedida (ou outro status que não implica em <i>fallback</i>); "1" = Erro passível de <i>fallback</i> ; ou "2" = Aplicação requerida não suportada (<i>fallback</i> depende das definições da Rede Credenciadora). |
| GCR_APPTYPE | N2 | Retorna o valor de T1_APPTYPE do registro da Tabela de AID usada no processamento do cartão com <i>chip</i> . Este campo volta zerado para GCR_CARDTYPE = "00" (magnético). |
| GCR_ACQIDX | N2 | Retorna o valor de TAB_ACQ do registro da Tabela de AID usada no processamento do cartão com <i>chip</i> . Este campo volta zerado para GCR_CARDTYPE = "00" (magnético). |
| GCR_RECIDX | A2 | Retorna o valor de TAB_RECIDX do registro da Tabela de AID usada no processamento do cartão com <i>chip</i> . Este campo volta zerado para GCR_CARDTYPE = "00" (magnético). |
| GCR_TRK1LEN | N2 | Tamanho da trilha 1. |
| GCR_TRK1 | A76 | Trilha 1 (sem as sentinelas e com o <i>byte</i> de formato - primeiro caractere alfanumérico), alinhada à esquerda com espaços à direita. |
| GCR_TRK2LEN | N2 | Tamanho da trilha 2. |
| GCR_TRK2 | A37 | Trilha 2 (sem as sentinelas), alinhada à esquerda com espaços à direita. |
| GCR_TRK3LEN | N3 (ou A3**) | Tamanho da trilha 3. |
| GCR_TRK3 | A104 | Trilha 3 (sem as sentinelas), alinhada à esquerda com espaços à direita. |
| GCR_PANLEN | N2 | Tamanho do PAN. |
| GCR_PAN | A19 | PAN, alinhado à esquerda com espaços à direita. Este campo não é devolvido para GCR_CARDTYPE = "00" (magnético) ou "05" (CTLS simulando tarja). |
| GCR_PANSEQNO | N2 | <i>Application PAN Sequence Number</i> . Este campo volta zerado para GCR_CARDTYPE = "00" (magnético) ou "05" (CTLS simulando tarja). |
| GCR_APPLABEL | A16 | Etiqueta da aplicação sendo processada, com espaços à direita. Este campo não é devolvido para GCR_CARDTYPE = "00" (magnético). |
| GCR_SRVCODE | N3 | <i>Service Code</i> . Este campo volta zerado para GCR_CARDTYPE = "00" (magnético) ou "05" (CTLS simulando tarja). |

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| GCR_CHNAME | A26 | Cardholder Name, com espaços à direita. Este campo somente retorna se GCR_CARDTYPE = "03" (ICC EMV). |
| GCR_CARDEXP | N6 | Data de expiração do cartão (<i>Application Expiration Date</i>), no formato "AAMMDD". Este campo volta zerado para GCR_CARDTYPE = "00" (magnético) ou "05" (CTLS simulando tarja). |
| GCR_RUF1 | N29 | RUF (deve ser ignorado pelo SPE). |
| GCR_ISSCNTRY | N3 | Código do país do cartão (<i>Issuer Country Code</i>). Este campo volta zerado para GCR_CARDTYPE = "00" (magnético) ou "05" (CTLS simulando tarja), podendo voltar zerado também para "06" (CTLS EMV), dependendo da situação. |
| GCR_ACQRDLEN | N3 | Tamanho de GCR_ACQRD , em caracteres. <ul style="list-style-type: none"> ▪ Se GCR_ACQIDX = "01", GCR_ACQRDLEN é "066"; ▪ Se GCR_ACQIDX = "02", GCR_ACQRDLEN é "010"; e ▪ Para outros valores de GCR_ACQIDX, o campo GCR_ACQRD não existe (GCR_ACQRDLEN é "000"). |
| GCR_ACQRD | A..66 | Dados de retorno específicos da Rede Credenciadora selecionada (ver tabelas a seguir). |

⚠ Se o pinpad estiver em modo "PAN Criptografado" (ver **seção 5.3**), **GCR_PAN** e os PAN das trilhas vêm codificados pela chave **WK_{PAN}**.

⚠ Se o pinpad estiver em modo "PAN Criptografado", **GCR_TRK3LEN** não é preenchido, pois a trilha 2 pode atingir 40 caracteres (ver explicação na **seção 5.3**).

** Neste caso ele deixa de ter formato "N3" e passa a ter formato "A3"!!

Se **GCR_ACQIDX** = "01":

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| GCR_ACQRD | N2 | Quantidade de bytes representativos no <i>Application Identifier</i> (tamanho ÷ 2). |
| | H32 | <i>DF Name</i> (tag 84h), com FFh à direita. |
| | A16 | <i>Application Label</i> (tag 50h), com espaços à direita. |
| | A16 | <i>Application Preferred Name</i> (tag 9F12h), com espaços à direita. Caso o <i>Issuer Code Table Index</i> não for compatível com o <i>Additional Terminal Capabilities</i> , este campo deverá estar preenchido com espaços. |

Se **GCR_ACQIDX** = "02":

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| GCR_ACQRD | H10 | Application Usage Control (tag 9F07h), no seguinte formato: "9F0702xxxx" |

➔ Comando (depois de ↪ST_TABVERDIF)

Caso o a resposta a "GCR" retorne ↪ST_TABVERDIF, isso indica que o comando não foi processado pelo fato das Tabelas EMV carregadas não estarem com a versão esperada pelo SPE.

Neste caso, o SPE poderá ou não proceder com a atualização das tabelas (através dos comandos descritos na **seção 3.4.4**) e, em seguida, submeter novamente o comando "GCR" sem parâmetros, conforme formato a seguir:

| Id. do Campo | Formato | Descrição |
|--------------|---------|------------------------------|
| CMD_ID | A3 | Código do comando (= "GCR"). |

➔ Observação #1

O processamento de cartões EMV requer o conhecimento dos AIDs suportados, que são fornecidos através das Tabelas de AID (ver **seção 4.1.1**), sendo que diferentes Redes Credenciadoras podem suportar o processamento dos mesmos AIDs. Desta forma, ao utilizar opção **GCR_ACQIDXREQ** = "00", o SPE deve se certificar que o conjunto total de Tabelas de AID carregadas não possua registros com AIDs conflitantes. O pinpad não faz nenhum tratamento para solução deste tipo de conflito e, caso essa restrição não seja observada pelo SPE, o comportamento do pinpad será imprevisível.

➔ Observação #2

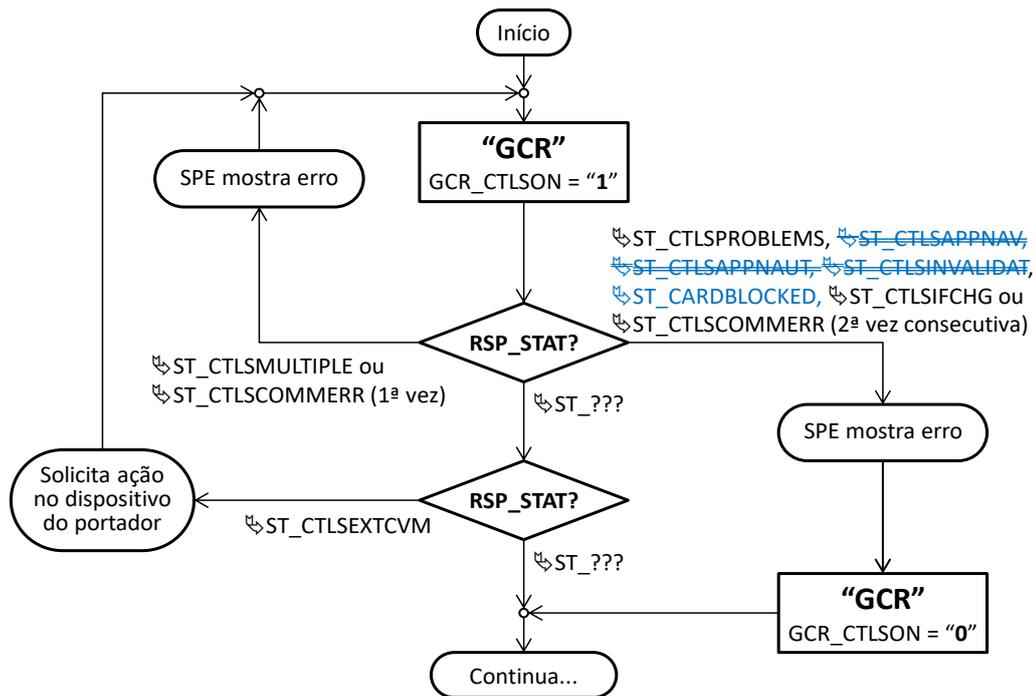
Um SPE que suporta CTLS deve acionar o comando "GCR" inicialmente permitindo essa interface usando **GCR_CTLSON** = "1" (ou omitindo este parâmetro). Entretanto, o SPE deverá desabilitar essa interface através de **GCR_CTLSON** = "0" e submeter novamente o comando nos seguintes casos:

- Quando o comando retornar os erros ↪ST_CTLSPROBLEMS, ↪ST_CARDBLOCKED, ↪ST_CTLSPROBLEMS, ↪ST_CTLSPROBLEMS, ↪ST_CTLSPROBLEMS, ↪ST_CTLSPROBLEMS ou ↪ST_CTLSPROBLEMS;
- Quando o comando retornar o erro ↪ST_CTLSCOMMERR pela segunda vez consecutiva.

➔ Observação #3

Se o comando "GCR" retornar ↪ST_CTLSEXTCVM, o SPE deve apresentar uma mensagem ao portador solicitando uma ação no seu dispositivo (ex: "SIGA INSTRUÇÕES NO TELEFONE") e acionar novamente o comando.

O diagrama a seguir ilustra este processo:



Exemplos

O SPE solicita um cartão informando uma lista de três índices da rede "02" (a interface CTLS é ativada, pois **GCR_CTLSON** não foi fornecido).

| | | |
|--------------|--|---|
| SPE ⇒ | 47 43 52 30 35 32 30 30 30 30 30 30 30 30 30 30 30 30 31 30 30 30 31 33 31 32 30 37 31 30 32 33 35 35 38 37 36 35 32 33 34 35 36 38 30 33 30 32 31 34 30 32 32 32 30 32 31 37 | GCR052000000000 0010001312071023 5587652345680302 1402220217 |
|--------------|--|---|

O pinpad notifica o SPE da aplicação selecionada no cartão.

| | | |
|-------------|---|--|
| ⇐ PP | 4E 54 4D 30 30 30 30 33 32 53 45 4C 45 43 49 4F 4E 41 44 4F 3A 20 20 20 20 43 52 45 44 49 54 4F 20 20 20 20 20 20 20 20 | NTM000032SELECIO NADO:.....CREDITO |
|-------------|---|--|

Um ICC EMV foi inserido e processado com sucesso.

| | | |
|------|--|------------------|
| ← PP | 47 43 52 30 30 30 33 35 32 30 33 30 30 31 30 32 | GCR0003520300102 |
| | 32 32 30 30 20 20 20 20 20 20 20 20 20 20 20 20 | 2200..... |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 33 37 35 34 36 34 39 39 39 39 31 30 37 36 39 39 | 3754649999107699 |
| | 39 30 3D 31 39 31 32 32 30 36 32 38 31 32 33 36 | 90=1912206281236 |
| | 36 39 30 37 37 30 30 30 30 30 20 20 20 20 20 20 | 6907700000..... |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 31 36 35 34 36 34 39 39 39 39 31 30 37 36 | ..16546499991076 |
| | 39 39 39 30 20 20 20 30 30 43 52 45 44 49 54 4F | 9990...00CREDITO |
| | 20 20 20 20 20 20 20 20 20 20 30 30 30 4C 45 45 20 |000LEE• |
| | 57 45 49 4E 52 49 42 20 2F 20 47 45 44 44 59 20 | WEINRIB•/•GEDDY• |
| | 20 20 20 20 20 20 31 39 31 32 33 31 30 30 30 30 |1912310000 |
| | 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 | 0000000000000000 |
| | 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 | 0000000000000000 |
| | 30 30 30 30 37 36 30 30 30 | 000076000 |

O SPE solicita um cartão considerando todas as redes e registros, valor \$28,37, habilitando CTLS.

| | | |
|-------|---|------------------|
| SPE ⇒ | 47 43 52 30 34 31 30 30 39 39 30 30 30 30 30 30 | GCR0410099000000 |
| | 30 30 32 38 33 37 31 34 30 37 32 34 31 31 33 39 | 0028371407241139 |
| | 32 32 54 41 42 56 45 52 30 39 31 38 30 30 30 | 22TABVER0918000 |

Um cartão magnético foi passado e seus dados devolvidos com sucesso, havendo também indicação de erro passível de "fallback" no processamento anterior de cartão com chip.

| | | |
|------|--|-------------------|
| ← PP | 47 43 52 30 30 30 33 35 32 30 30 31 30 30 30 30 | GCR0003520010000 |
| | 30 30 37 36 42 34 34 34 34 33 33 33 33 32 32 32 | 0076B44443333222 |
| | 32 31 31 31 31 5E 54 4F 4D 20 53 41 57 59 45 52 | 211111^TOM•SAWYER |
| | 5E 31 36 30 38 31 30 31 38 31 32 37 33 36 35 34 | ^160810181273654 |
| | 37 36 31 35 32 33 36 34 35 31 37 38 36 32 33 35 | 7615236451786235 |
| | 34 38 37 36 31 32 33 37 36 34 35 37 36 31 32 33 | 4876123764576123 |
| | 33 37 34 34 34 34 33 33 33 33 32 32 32 32 31 31 | 3744443333222211 |
| | 31 31 3D 31 36 30 38 31 30 31 38 31 32 37 33 36 | 11=1608101812736 |
| | 35 34 37 36 31 35 34 30 39 30 34 34 34 34 33 33 | 5476154090444433 |
| | 33 33 32 32 32 32 31 31 31 31 31 3D 3D 31 36 30 38 | 3322221111==1608 |
| | 31 30 31 38 31 32 37 33 36 35 34 37 36 31 35 32 | 1018127365476152 |
| | 33 36 34 35 31 37 38 36 32 33 35 34 38 37 36 31 | 3645178623548761 |
| | 32 33 37 36 34 35 37 36 31 32 33 3D 38 33 37 34 | 23764576123=8374 |
| | 38 32 37 34 37 38 37 32 33 36 38 34 30 30 30 30 | 8274787236840000 |
| | 31 39 39 31 20 20 20 20 20 20 20 20 20 20 20 20 | 1991..... |
| | 20 20 30 30 20 20 20 20 20 20 20 20 20 20 20 20 | ..00..... |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 |00..... |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 |000..... |
| | 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | |
| | 20 20 20 20 20 20 30 30 30 30 30 30 30 30 30 30 |0000000000 |
| | 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 | 0000000000000000 |
| | 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 30 | 0000000000000000 |
| | 30 30 30 30 30 30 30 30 30 30 | 000000000 |

3.6.2. Comando “CNG”

Obsoleto
 Blocante
 ABECS

Este comando permite que o SPE abasteça o pinpad com parâmetros EMV adicionais (inclusive proprietários) para serem usados no processamento dos comandos “**GOC**” e “**FNC**”. Estes parâmetros podem coincidir com os existentes no registro da Tabela de AID (ver **seção 4.1.1**) correspondente à aplicação selecionada no cartão com *chip* EMV. Neste caso, os valores não são alterados nas tabelas, sendo somente relevantes para o cartão em processamento.

Ele é extremamente útil para resolver situações específicas não previstas pelas tabelas, como, por exemplo, o caso de estabelecimentos que utilizam mais de um tipo de moeda, ou no caso de cartões que necessitam de parâmetros proprietários não previstos na norma EMV.

⚠ Este comando somente pode ser utilizado após a execução bem-sucedida do comando “**GCR**”, no caso específico de **GCR_CARDTYPE** = “03” (ICC EMV).

➔ Comando

| Id. do Campo | Formato | Descrição |
|---------------------|---------|---|
| CMD_ID | A3 | Código do comando (= “ CNG ”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| CNG_EMVDTLEN | N2 | Quantidade de bytes representados em CNG_EMV DAT (tamanho ÷ 2). |
| CNG_EMV DAT | H..198 | Sequência de parâmetros específicos a serem usados pelo processamento EMV nos comandos “ GOC ” e/ou “ FNC ”, no <u>formato TLV</u> (ver seção 7.1). |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|-----------------|---------|--|
| RSP_ID | A3 | Código da resposta (= “ CNG ”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_INVCALLA chamada anterior de “ GCR ” não processou com sucesso um cartão ICC EMV. ↳ ST_INVPARMEstrutura TLV em CNG_EMV DAT não está íntegra. |

➡ Exemplos

O SPE informa os valores dos seguintes parâmetros EMV para uso no processamento:

- *Terminal Capabilities (tag 9F33h)* = E0D0C8h
- *Transaction Currency Code (tag 5F2Ah)* = 0840h
- Dado proprietário de uso do emissor (*tag DF04h*) = 169937823Fh

| | | |
|--------------|---|--|
| SPE ⇒ | 43 4E 47 30 34 30 31 39 39 46 33 33 30 33 45 30 44 30 43 38 35 46 32 41 30 32 30 38 34 30 44 46 30 34 30 35 31 36 39 39 33 37 38 32 33 46 | CNG040199F3303E0 D0C85F2A020840DF 0405169937823F |
|--------------|---|--|

O pinpad recebe os dados com sucesso.

| | | |
|-------------|-------------------|--------|
| ⇐ PP | 43 4E 47 30 30 30 | CNG000 |
|-------------|-------------------|--------|

3.6.3. Comando “GOC”

Obsoleto
 Blocante
 ABECS

Este comando continua o processo de tratamento de cartões com *chip*, conforme apresentado na **seção 3.6.5**.

Caso “**GCR**” tenha reportado a passagem de um cartão magnético (ou CTLS simulando tarja), este comando não deverá ser utilizado.

➔ Comando

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| CMD_ID | A3 | Código do comando (= “ GOC ”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (de GOC_AMOUNT a GOC_ACQPR). |
| GOC_AMOUNT | N12 | Novo valor da transação (<i>Amount, authorized</i>) em centavos, podendo incluir novos valores apresentados ao SPE após “ GCR ” (como por exemplo, taxa do serviço, saque ou troco). Caso não existam acréscimos ao valor, ele deverá ser aqui mantido idêntico ao passado em “ GCR ”. |
| GOC_CASHBACK | N12 | Parcela do valor da transação referente a saque ou troco - <i>cashback (Amount, other)</i> em centavos. Caso esse valor não exista, este campo deve ser preenchido com zeros. |
| GOC_EXCLIST | N1 | Resultado da consulta à Lista de Exceção (só para ICC EMV): “0” = PAN não consta na Lista de Exceção. “1” = PAN consta na Lista de Exceção. |
| GOC_CONNECT | N1 | Obrigatoriedade de conexão (só para ICC EMV): “0” = Transação pode ser aprovada <i>offline</i> . “1” = Transação não pode ser aprovada <i>offline</i> (somente pode ser efetuada <i>online</i> ou negada <i>offline</i>). |
| GOC_RUF1 | N1 | RUF (fixo “0”). |
| GOC_METHOD | N1 | Método de criptografia de PIN <i>online</i> , a ser usado caso requerido pelo processamento EMV: “1” = MK/WK:TDES:PIN “3” = DUKPT:TDES:PIN |
| GOC_KEYIDX | N2 | Índice da MK ou do registro de tratamento DUKPT. |
| GOC_WKENC | H32 | <i>Working Key</i> (criptografada pela MK indicada em GOC_KEYIDX). Se GOC_METHOD = “3”, este campo é ignorado pelo pinpad. |
| GOC_RISKMAN | N1 | Gerenciamento de risco ICC EMV (<i>Terminal Risk Management</i>), usando os parâmetros GOC_FLRLIMIT , GOC_TPBRs , GOC_TVBRs e GOC_MTPBRs : “1” = Faz o gerenciamento de risco (<u>sempre</u>). |
| GOC_FLRLIMIT | H8 | <i>Terminal Floor Limit</i> (em centavos) |

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| GOC_TPBR | N2 | Target Percentage to be used for Biased Random Selection |
| GOC_TVBR | H8 | Threshold Value for Biased Random Selection (em centavos) |
| GOC_MTPBR | N2 | Maximum Target Percentage to be used for Biased Random Selection |
| GOC_ACQPRLEN | N3 | Tamanho de <u>GOC_ACQPR</u> , em caracteres. <ul style="list-style-type: none"> ▪ Se <u>GCR_ACQIDX</u> = "01", <u>GOC_ACQPRLEN</u> é "003"; ▪ Se <u>GCR_ACQIDX</u> = "02", <u>GOC_ACQPRLEN</u> é "032"; e ▪ Para outros valores de <u>GCR_ACQIDX</u>, o campo <u>GOC_ACQPR</u> não existe (<u>GOC_ACQPRLEN</u> é "000"). |
| GOC_ACQPR | A..32 | Parâmetros de entrada específicos da Rede Credenciadora selecionada (ver tabelas a seguir). |
| CMD_LEN2 | N3 | Tamanho dos dados a seguir (<u>GOC_TAGS1LEN</u> e <u>GOC_TAGS1</u>). |
| GOC_TAGS1LEN | N3 | Quantidade de bytes representados em <u>GOC_TAGS1</u> (tamanho ÷ 2). |
| GOC_TAGS1 | H..256 | Primeira lista de <i>tags</i> identificando os objetos de dados EMV a serem devolvidos em <u>GOC_EMVDAT</u> . As <i>tags</i> devem ser <u>simplesmente concatenadas</u> , respeitando-se sua regra de formação (ver seção 7.1). |
| CMD_LEN3 | N3 | Tamanho dos dados a seguir. |
| GOC_TAGS2LEN | N3 | Quantidade de bytes representados em <u>GOC_TAGS2</u> (tamanho ÷ 2). |
| GOC_TAGS2 | H..256 | Segunda lista de <i>tags</i> , adicional à <u>GOC_TAGS1</u> . Este campo existe simplesmente por razões históricas. |

Se GCR_ACQIDX = "01":

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| GOC_ACQPR | N2 | Transaction Type (tag 9Ch) |
| | N1 | "0" – Não permite o <i>bypass</i> de PIN. "1" – Permite o <i>bypass</i> de PIN. |

Se GCR_ACQIDX = "02":

| Id. do Campo | Formato | Descrição |
|--------------|---------|---|
| GOC_ACQPR | S32 | Mensagem a ser apresentada na captura de PIN, seja <i>online</i> ou <i>offline</i> , já formatada para 2 linhas e 16 colunas. |

➔ Resposta

| Id. do Campo | Formato | Descrição |
|---------------|---------|--|
| RSP_ID | A3 | Código da resposta (= "GOC"). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_INVCALL Comando "GCR" não foi executado previamente com sucesso para ICC/CTLS EMV. ↳ ST_ERRKEY Problema na chave de criptografia de PIN. ↳ ST_TIMEOUT Tempo esgotado na tela de captura de PIN. ↳ ST_CARDPROBLEMS ... Cartão inválido ou com problemas. ↳ ST_CARDINVDATA ICC com dados inválidos ou faltando. ↳ ST_ERRFALLBACK Erro sujeito a "fallback" para tarja. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| GOC_DECISION | N1 | Decisão tomada: "0" = Transação aprovada <i>offline</i> . "1" = Transação negada. "2" = Transação requer autorização <i>online</i> . |
| GOC_SIGNAT | N1 | Assinatura em papel deve ser obtida ("0"-não / "1"-sim). |
| GOC_PINOFF | N1 | PIN foi verificado <i>offline</i> ("0"-não / "1"-sim). |
| GOC_ERRPINOFF | N1 | Número de apresentações inválidas de PIN <i>offline</i> <u>nesta transação</u> . |
| GOC_PBLOCKED | N1 | PIN <i>offline</i> foi bloqueado na última apresentação inválida <u>nesta transação</u> ("0"-não / "1"-sim). |
| GOC_PINONL | N1 | PIN capturado para verificação <i>online</i> ("0"-não / "1"-sim). Se este campo estiver com "0", GOC_PINBLK e GOC_KSN não devem ser considerados. |
| GOC_PINBLK | H16 | PIN criptografado. |
| GOC_KSN | H20 | Número de série da chave (<i>Key Serial Number</i>), somente no caso de DUKPT (GOC_METHOD = "3"). Para MK/WK, este campo é devolvido zerado. |
| GOC_EMVDTLEN | N3 | Quantidade de bytes representados em GOC_EMVDAT (tamanho ÷ 2). |
| GOC_EMVDAT | H.512 | Dados da transação EMV para envio à Rede Credenciadora, no formato TLV (ver seção 7.1). O pinpad concatena os dados pedidos por GOC_TAGS1 e GOC_TAGS2, <u>se encontrados</u> , respeitando a ordem em que foram solicitados. Objetos EMV que contenham informações de trilha de cartão (ou PAN) não serão devolvidos pelo pinpad! |
| GOC_ACQRDLEN | N3 | Tamanho dos dados de retorno específicos da Rede Credenciadora (<u>não usado</u> - fixo "000"). |

➡ Exemplos

O SPE solicita a continuidade da transação, alterando o valor para \$12,00, fornecendo os parâmetros de gerenciamento de risco EMV e os parâmetros para eventual captura de PIN *online*.

| | | |
|--------------|--|--|
| SPE ⇒ | 47 4F 43 30 38 36 30 30 30 30 30 30 30 30 31 32 30 30 30 30 30 30 30 30 30 30 30 30 32 30 30 30 30 31 33 30 31 30 31 30 30 30 30 31 33 38 38 32 30 30 30 30 30 30 33 45 38 38 30 30 30 30 30 32 35 30 31 31 38 32 39 46 32 37 39 46 32 36 39 46 33 36 39 35 38 46 39 46 33 37 30 30 33 30 30 30 | GOC0860000000012 0000000000020000 1301000000000000 0000000000000000 0000100001388200 00003E8800000250 11829F279F269F36 958F9F37003000 |
|--------------|--|--|

O pinpad notifica o SPE da necessidade de captura do PIN.

| | | |
|-------------|---|---|
| ⇐ PP | 4E 54 4D 30 30 30 30 33 32 53 4F 4C 49 43 49 54 45 20 41 20 53 45 4E 48 41 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 | NTM000032SOLICIT E•A•SENHA•••••••• •••••••• |
|-------------|---|---|

A operação é bem-sucedida, com captura de PIN *offline*, sendo que cartão pede autorização *online*.

| | | |
|-------------|--|---|
| ⇐ PP | 47 4F 43 30 30 30 31 33 30 32 30 31 30 34 31 38 32 30 32 35 38 30 30 39 46 32 37 30 31 38 30 39 46 32 36 30 38 37 36 35 44 43 31 33 38 30 37 44 31 45 34 43 38 39 46 33 36 30 32 30 30 30 36 39 35 30 35 30 30 31 30 30 30 30 30 30 30 38 46 30 31 30 35 39 46 33 37 30 34 35 41 37 37 41 43 46 30 30 30 30 | GOC0001302010000 0000000000000000 0000000000000000 000041820258009F 2701809F2608765D C13807D1E4C89F36 0200069505001000 00008F01059F3704 5A77ACF0000 |
|-------------|--|---|

3.6.4. Comando “FNC”

Obsoleto
 Blocante
 ABECS

Este comando finaliza o processamento de cartão com *chip* e deve ser chamado caso “GOC” tenha requerido aprovação *online* (GOC_DECISION = “2”), conforme apresentado na **seção 3.6.5**.

No caso de aprovação ou negação *offline* (GOC_DECISION = “0” ou “1”), este comando pode ser chamado apenas para se manter o mesmo fluxo operacional de uma transação *online*.

➔ Comando

| Id. do Campo | Formato | Descrição |
|---------------|---------|---|
| CMD_ID | A3 | Código do comando (= “FNC”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (de <u>FNC_COMMST</u> a <u>FNC_ACQPRLEN</u>). |
| FNC_COMMST | N1 | Status da comunicação com a Rede Credenciadora: “0” = Comunicação bem-sucedida, sendo que uma resposta válida foi recebida na transação <i>online</i> (ou transação foi finalizada <i>offline</i> em “GOC”). “1” = Não foi possível comunicar com a Rede Credenciadora. Nesse caso, os demais campos deste comando devem ser passados zerados. “9” = Comunicação bem-sucedida, transação <u>aprovada</u> , porém o <i>Authorization Response Code</i> é diferente de “00”. |
| FNC_ISSMODE | N1 | Tipo de Emissor: fixo “0” (EMV “full grade”) |
| FNC_ARC | A2 | <i>Authorization Response Code</i> (código de aprovação/negação devolvido pela Rede Credenciadora). |
| FNC_ISSDATLEN | N3 | Quantidade de bytes representados em <u>FNC_ISSDAT</u> (tamanho ÷ 2). |
| FNC_ISSDAT | H..512 | Dados da transação EMV recebidos da Rede Credenciadora, no formato TLV (ver seção 7.1). |
| FNC_ACQPRLEN | N3 | Tamanho dos parâmetros de entrada específicos da Rede Credenciadora (<u>não usado</u> - fixo “000”). |
| CMD_LEN2 | N3 | Tamanho dos dados a seguir. |
| FNC_TAGSLLEN | N3 | Quantidade de bytes representados em <u>FNC_TAGS</u> (tamanho ÷ 2). |
| FNC_TAGS | H..256 | Lista de <i>tags</i> identificando os objetos de dados EMV a serem devolvidos em <u>FNC_EMVDAT</u> . As <i>tags</i> devem ser <u>simplesmente concatenadas</u> , respeitando-se sua regra de formação (ver seção 7.1). |

➤ Resposta

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| RSP_ID | A3 | Código da resposta (= "FNC"). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_INVCALL Comando "GOC" não foi executado previamente com sucesso. ↳ ST_CARDPROBLEMS ... Cartão inválido ou com problemas. ↳ ST_CARDINVDATA ICC com dados inválidos ou faltando. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| FNC_DECISION | N1 | Decisão tomada: "0" = Transação aprovada. "1" = Transação negada pelo cartão. "2" = Transação negada pela Rede Credenciadora. |
| FNC_EMVDTLEN | N3 | Quantidade de bytes representados em FNC_EMV DAT (tamanho ÷ 2). |
| FNC_EMV DAT | H..512 | Dados da transação EMV para envio à Rede Credenciadora, no formato TLV (ver seção 7.1). O pinpad concatena os dados pedidos por FNC_TAGS , se encontrados, respeitando a ordem em que foram solicitados. Objetos EMV que contenham informações de trilha de cartão (ou PAN) não serão devolvidos pelo pinpad! |
| FNC_ISRLEN | N2 | Quantidade de bytes representados em FNC_ISR (tamanho ÷ 2). |
| FNC_ISR | H..100 | <i>Issuer Script Results</i> |
| FNC_ACQRDLEN | N3 | Tamanho dos dados de retorno específicos da Rede Credenciadora (não usado - fixo "000"). |

➤ Exemplos

O SPE solicita a finalização da transação EMV. A Rede Credenciadora aprova a transação, devolvendo também o *Issuer Authentication Data* (tag 91h).

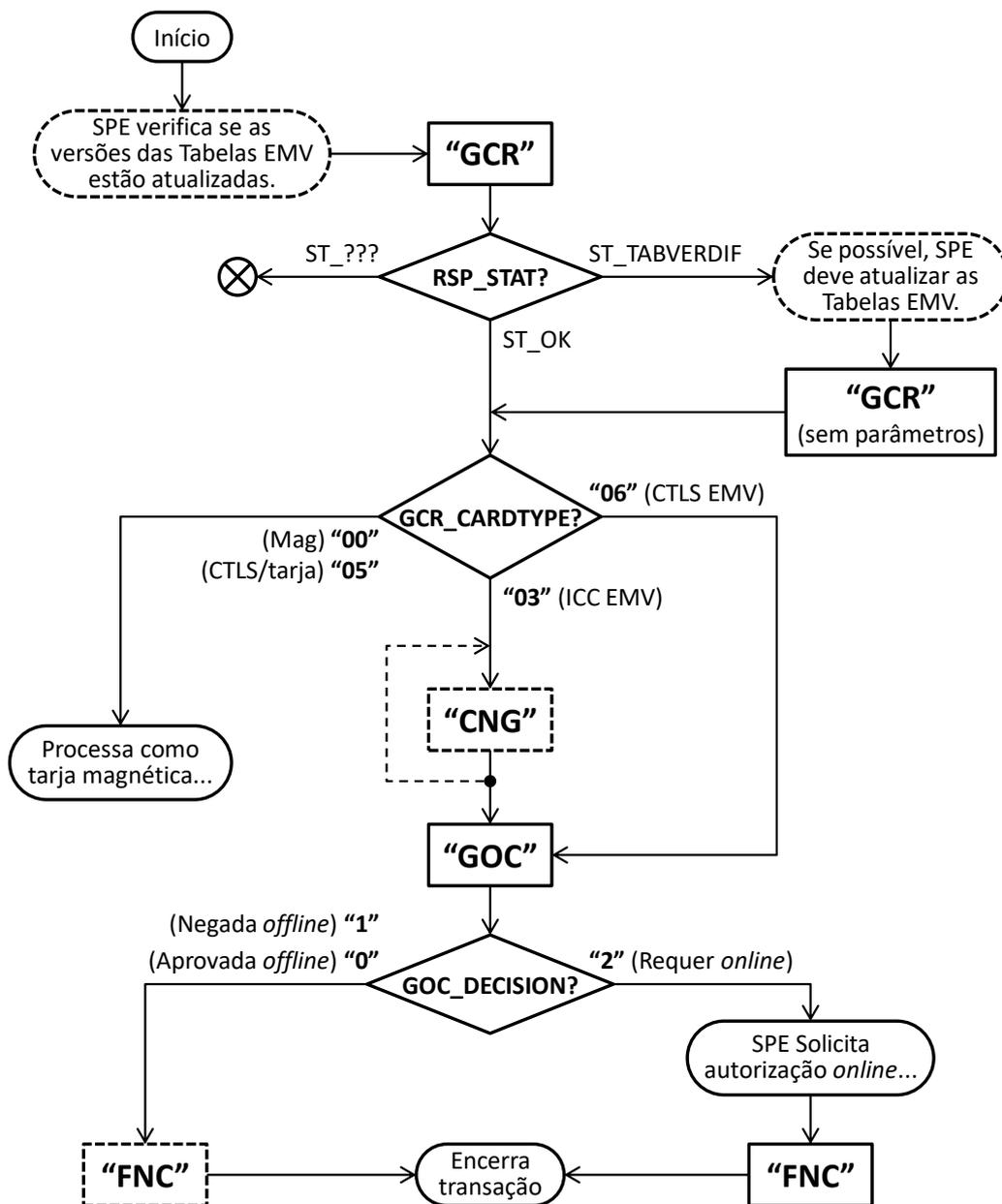
| | | |
|-------|--|--|
| SPE ⇒ | 46 4E 43 30 33 30 30 30 30 30 30 31 30 39 31 30 38 45 36 34 41 32 46 45 32 31 46 44 38 38 36 37 32 30 30 30 30 32 35 30 31 31 38 32 39 46 32 37 39 46 32 36 39 46 33 36 39 35 38 46 39 46 33 37 | FNC0300000010910 8E64A2FE21FD8867 2000025011829F27 9F269F36958F9F37 |
|-------|--|--|

A operação é bem-sucedida, mas o cartão nega a transação ao final (o SPE deverá desfazer a transação com a Rede Credenciadora).

| | | |
|------|---|--|
| ⇐ PP | 46 4E 43 30 30 30 30 39 31 31 30 34 31 38 32 30 32 35 38 30 30 39 46 32 37 30 31 30 30 39 46 32 36 30 38 36 39 45 42 41 33 42 45 31 43 43 38 42 33 38 44 39 46 33 36 30 32 30 30 30 36 39 35 30 35 30 30 31 30 30 30 30 30 30 30 38 46 30 31 30 35 39 46 33 37 30 34 35 41 37 37 41 43 46 30 30 30 30 30 30 | FNC0000911041820 258009F2701009F2 60869EBA3BE1CC8B 38D9F36020006950 500100000008F010 59F37045A77ACF00 0000 |
|------|---|--|

3.6.5. Fluxo de operação

O fluxo a seguir ilustra a sequência de chamada dos comandos obsoletos de processamento de cartão. Os blocos pontilhados referem-se a processamentos opcionais que dependem da especificação da Rede Credenciadora.



3.7. Comandos Abecs de processamento de cartão

Esta seção detalha comandos de alto-nível responsáveis pelo processamento completo de um cartão durante uma operação de pagamento, seja magnético, ICC ou CTLS.

Os seguintes comandos estão contemplados nesta seção:

| CMD_ID | Significado | Obsoleto | Blocante | Abecs |
|--------|--|--------------------------|-------------------------------------|-------------------------------------|
| "GCX" | <i>Get Card - Extended</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| "GED" | <i>Get EMV Data</i> | <input type="checkbox"/> | <input type="checkbox"/> | <input checked="" type="checkbox"/> |
| "GOX" | <i>Go On Chip Processing - Extended</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| "FCX" | <i>Finish Chip Processing - Extended</i> | <input type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |

 Os comandos apresentados nesta seção são bastante flexíveis e sua forma de uso depende profundamente das especificações dos sistemas de pagamento da Rede Credenciadora.

3.7.1. Comando “GCX”

Obsoleto
 Blocante
 ABECS

Este comando inicia um processo de transação com cartão de pagamento (seja ele magnético, ICC ou CTLS), conforme apresentado na **seção 3.7.5**.

Ele é equivalente ao comando “**GCR**”, porém com as seguintes diferenças:

- Utiliza o formato Abecs para permitir flexibilidade e facilitar evoluções futuras.
- Resolve automaticamente eventuais conflitos de AID ao se considerar as tabelas de todas as Redes Credenciadoras no processamento.
- Não efetua controle de versão das Tabelas EMV. O SPE deve efetuar esse controle de forma independente através dos comandos da **seção 3.5**, devendo verificar a versão e, se necessário, atualizar as tabelas necessárias antes da execução deste comando.
- Permite ao SPE enviar ao pinpad uma lista de parâmetros EMV a serem usados no processamento.
- Permite ao SPE obter uma lista qualquer de dados EMV do cartão.
- Retorna os dados de trilha incompletos, conforme processo de segurança descrito na **seção 5.4**. Para obter as trilhas completas (abertas ou criptografadas), deve-se usar o comando “**GTK**”.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| CMD_ID | M | Código do comando (= “ GCX ”). |
| SPE_TRNTYPE | O | Tipo de transação a ser efetuada: 00h = Compra; 01h = Saque; 09h = Compra com saque/troco (<i>cashback</i>); 20h = Cancelamento (<i>refund</i>); 30h = Consulta de saldo; ou Outros valores de acordo com ISO 8583:1987. Se este parâmetro não for fornecido, o pinpad considerará a transação como sendo de “compra” (se SPE_CASHBACK ausente) ou “ <i>cashback</i> ” (se SPE_CASHBACK presente e diferente de zero). |
| SPE_ACQREF | O | Valor de TAB_ACQ da Tabela de AID a ser utilizado no processamento (se este parâmetro não for fornecido, o pinpad considerará as tabelas de todas as Redes Credenciadoras). |
| SPE_APPTYPE | O | Valor(es) de T1_APPTYPE dos registros das Tabelas de AID a serem utilizados no processamento (se este parâmetro não for fornecido, o pinpad considerará qualquer valor). |
| SPE_AIDLIST | O | Lista específica de registros das Tabelas de AID a serem usados no processamento, sendo cada entrada composta pela concatenação de TAB_ACQ e TAB_RECIDX . IMPORTANTE: Se este parâmetro estiver presente, SPE_ACQREF e SPE_APPTYPE serão simplesmente desprezados pelo pinpad se existentes no comando. |

| Id. do Campo | Presença | Descrição / Observação |
|---------------------|----------|--|
| <u>SPE_AMOUNT</u> | O | Valor da transação em centavos (<i>Amount, authorized</i>). Se este parâmetro estiver ausente, o pinpad considerará este dado como zerado. |
| <u>SPE_CASHBACK</u> | O | Valor da transação referente a saque ou troco - <i>cashback (Amount, other)</i> em centavos. Se este parâmetro estiver ausente, o pinpad considerará este dado como zerado. |
| <u>SPE_TRNCURR</u> | O | Código da moeda a ser usada na transação (<i>Transaction Currency Code</i>), <u>somente para ICC</u> . Se este parâmetro estiver ausente, o pinpad usa o valor definido em <u>T1_TRNCURR</u> . |
| <u>SPE_TRNDATE</u> | M | Data da transação. |
| <u>SPE_TRNTIME</u> | M | Hora da transação. |
| <u>SPE_GCXOPT</u> | O | Opções do comando: “0xxx” = Aguarda cartão magnético ou ICC; ou “1xxx” = Aguarda cartão magnético, ICC ou CTLS. “x0xxx” = Mostra o valor da transação na tela de espera pelo cartão, se este for diferente de zero. “x1xxx” = Não mostra o valor da transação. “xx000” = RUF. Se este parâmetro estiver ausente, o pinpad considerará este dado como zerado. |
| <u>SPE_PANMASK</u> | O | Definições para mascaramento do PAN nos campos de resposta <u>PP_PAN</u> , <u>PP_TRK1INC</u> , <u>PP_TRK2INC</u> e <u>PP_TRK3INC</u> . Se ausente, não há mascaramento. |
| <u>SPE_EMVDATA</u> | O | Lista opcional de parâmetros EMV (no formato TLV). Os dados aqui fornecidos têm prioridade em relação aos objetos das Tabelas de AID, caso sejam coincidentes. |
| <u>SPE_TAGLIST</u> | O | Lista de <i>tags</i> dos objetos EMV a serem retornados na resposta ao comando. |
| <u>SPE_TIMEOUT</u> | O | Tempo máximo para se aguardar a apresentação de um cartão ou outra ação do operador. |
| <u>SPE_DSPMSG</u> | O | Mensagem a ser apresentada no <i>display</i> do pinpad para a solicitação do cartão. Se este parâmetro não for fornecido, o pinpad usa uma mensagem padrão. |

➤ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|---------------|----------|-------------------------------|
| <u>RSP_ID</u> | M | Código da resposta (= “GCX”). |

| Id. do Campo | Presença | Descrição / Observação |
|----------------------|----------|--|
| <u>RSP_STAT</u> | M | <p>Retornos de erro relevantes (ver seção 3.1.1):</p> <ul style="list-style-type: none"> ↳ <u>ST_RSPOVRFL</u>..... Tamanho dos dados EMV ultrapassa máximo permitido para PP_EMVDATA. ↳ <u>ST_CARDINVALIDAT</u>... Aplicação ICC está invalidada. ↳ <u>ST_CARDBLOCKED</u> ICC ou CTLS está bloqueado. ↳ <u>ST_CARDPROBLEMS</u>... ICC inválido ou com problemas. ↳ <u>ST_CARDINVDATA</u> ICC com dados inválidos ou faltando. ↳ <u>ST_CARDAPPNAV</u>..... Modo inválido para o ICC. ↳ <u>ST_CARDAPPNAUT</u>..... ICC não aceito. ↳ <u>ST_ERRFALLBACK</u> Erro de ICC sujeito a “<i>fallback</i>” para tarja. ↳ <u>ST_CTLSINVALIDAT</u> CTLS está invalidado/bloqueado. ↳ <u>ST_CTLSPROBLEMS</u> CTLS inválido ou com problemas. ↳ <u>ST_CTLSAPPNAV</u> Modo inválido para o CTLS. ↳ <u>ST_CTLSAPPNAUT</u> CTLS não aceito. ↳ <u>ST_CTLSEXTCVM</u> Solicitar verificação no dispositivo do portador. ↳ <u>ST_CTLSIFCHG</u> Mudar interface (usar ICC ou tarja). ↳ <u>ST_TIMEOUT</u> Esgotado tempo definido por SPE_TIMEOUT. |
| <u>PP_CARDTYPE</u> | M | <p>Tipo de cartão lido:</p> <p>“00” = Magnético; “03” = ICC EMV; “05” = CTLS simulando tarja; e “06” = CTLS EMV.</p> |
| <u>PP_ICCSTAT</u> | MD | <p>Este campo só é retornado se PP_CARDTYPE = “00” (magnético), sendo mandatório neste caso.</p> <p>Status da última leitura de ICC, usado pelo SPE se o cartão passado tiver indicação de presença de <i>chip</i>, de modo a recusá-lo ou não.</p> <p>“0” = Bem-sucedida (ou outro status que não implica em <i>fallback</i>); “1” = Erro passível de <i>fallback</i>; ou “2” = Aplicação requerida não suportada (<i>fallback</i> depende das definições da Rede Credenciadora).</p> |
| <u>PP_AIDTABINFO</u> | MD | <p>Este campo só é retornado se PP_CARDTYPE ≠ “00” (ICC ou CTLS), sendo mandatório neste caso.</p> <p>Contém informações do(s) registro(s) da(s) Tabela(s) de AID usado(s) no processamento, sendo a concatenação de TAB_ACQ, TAB_RECIDX e T1_APPTYPE.</p> <p>IMPORTANTE: Caso mais de uma Rede Credenciadora seja a apta a processar o cartão, este campo pode conter uma lista com múltiplas entradas.</p> |

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| <u>PP_PAN</u> | MD | Número do cartão lido (PAN), podendo ser mascarado de acordo com <u>SPE_PANMASK</u> Este campo só é retornado se <u>PP_CARDTYPE</u> = "03" (ICC EMV) ou "06" (CTLS EMV), sendo mandatório nestes casos. |
| <u>PP_PANSEQNO</u> | MD | <i>Application PAN Sequence Number.</i> Este campo só é retornado se <u>PP_CARDTYPE</u> = "03" (ICC EMV) ou "06" (CTLS EMV), sendo mandatório nestes casos. |
| <u>PP_TRK1INC</u> | O | Trilha 1 <i>incompleta</i> , se lida do cartão, podendo ter o PAN mascarado de acordo com <u>SPE_PANMASK</u> . |
| <u>PP_TRK2INC</u> | O | Trilha 2 <i>incompleta</i> , se lida do cartão, podendo ter o PAN mascarado de acordo com <u>SPE_PANMASK</u> |
| <u>PP_TRK3INC</u> | O | Trilha 3 <i>incompleta</i> , se lida do cartão, podendo ter o PAN mascarado de acordo com <u>SPE_PANMASK</u> |
| <u>PP_CHNAME</u> | O | <i>Cardholder Name</i> , se existente no cartão lido (ICC ou CTLS). Este campo não é devolvido pelo pinpad se <u>PP_CARDTYPE</u> = "00" (magnético) ou "05" (CTLS simulando tarja). Este campo só é retornado se <u>PP_CARDTYPE</u> = "03" (ICC EMV). |
| <u>PP_LABEL</u> | MD | Etiqueta da aplicação sendo processada. Este campo só é retornado se <u>PP_CARDTYPE</u> ≠ "00" (ICC ou CTLS), sendo mandatório neste caso. |
| <u>PP_ISSCNTRY</u> | O | Código do país do cartão (<i>Issuer Country Code</i>), se existente no cartão lido (ICC ou CTLS). Este campo <u>não é devolvido</u> pelo pinpad se <u>PP_CARDTYPE</u> = "00" (magnético) ou "05" (CTLS simulando tarja). Dependendo da situação, também pode não ser devolvido se <u>PP_CARDTYPE</u> = "06" (CTLS EMV). |
| <u>PP_CARDEXP</u> | O | Data de expiração do cartão (<i>Application Expiration Date</i>), se existente no cartão lido (ICC ou CTLS). Este campo <u>não é devolvido</u> pelo pinpad se <u>PP_CARDTYPE</u> = "00" (magnético) ou "05" (CTLS simulando tarja). |
| <u>PP_EMVDATA</u> | MR | Lista de objetos EMV definida por <u>SPE_TAGLIST</u> . Objetos não encontrados simplesmente não são devolvidos pelo pinpad, <u>assim como objetos que contenham informações de trilha de cartão (ou PAN)</u> . Este campo é mandatório sempre que <u>SPE_TAGLIST</u> existir no comando, <u>mesmo que nenhum objeto seja encontrado</u> (caso em que é retornado com tamanho zerado). |

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|--|
| <u>PP_DEVTYPE</u> | MD | <p>Tipo de dispositivo CTLS usado na transação (se <u>PP_CARDTYPE</u> = "05" ou "06"):</p> <p>"00" = Cartão;</p> <p>"01", "07", "08", "09", "14", "16", "21" = Telefone móvel;</p> <p>"02" = Chaveiro;</p> <p>"03", "18", "19", "23" = Relógio;</p> <p>"04", "27" = Etiqueta móvel ("<i>mobile tag</i>");</p> <p>"05" = Pulseira / bracelete;</p> <p>"06" = Capa de telefone móvel ("<i>case/sleeve</i>");</p> <p>"10", "11", "12", "13", "15", "17", "22" = Tablet ou e-Reader;</p> <p>Outros valores = Uso futuro.</p> <p>Na ausência deste campo, assume-se o dispositivo "cartão".</p> |

⚠ Caso um cartão magnético tenha sido passado (PP_CARDTYPE = "00") mas nenhuma trilha pôde ser lida, RSP_STAT = ↵ST_OK e os campos PP_TRK1INC, PP_TRK2INC e PP_TRK3INC não serão devolvidos.

➡ Observação #1

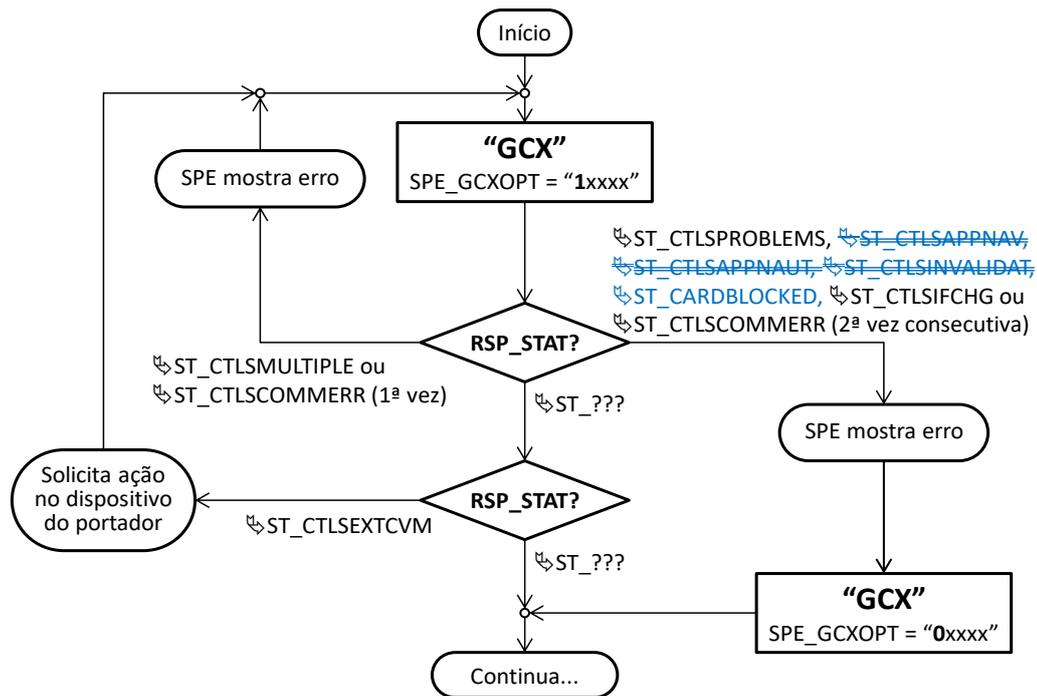
Um SPE que suporta CTLS deve acionar o comando "**GCX**" inicialmente permitindo essa interface usando SPE_GCXOPT = "1xxxx". Entretanto, o SPE deverá desabilitar essa interface através de SPE_GCXOPT = "0xxxx" (ou omitindo esse parâmetro) e submeter novamente o comando nos seguintes casos:

- Quando o comando retornar os erros ↵ST_CTLSPROBLEMS, ↵ST_CARDBLOCKED, ↵ST_CTLSPROBLEMS, ↵ST_CTLSPROBLEMS, ↵ST_CTLSPROBLEMS ou ↵ST_CTLSPROBLEMS; ou
- Quando o comando retornar o erro ↵ST_CTLSCOMMERR pela segunda vez consecutiva.

➡ Observação #2

Se o comando "**GCX**" retornar ↵ST_CTLSEXTCVM, o SPE deve apresentar uma mensagem ao portador solicitando uma ação no seu dispositivo (ex: "SIGA INSTRUÇÕES NO TELEFONE") e acionar novamente o comando.

O diagrama a seguir ilustra este processo:



Exemplos

O SPE inicia o processamento de uma transação com as seguintes características:

- Valor R\$ 483,00, sem *cashback*;
- Somente cartão magnético ou ICC (sem CTLS);
- Usa os parâmetros de todos os registros da Tabela de AID da Rede Credenciadora "08";
- Força o valor EOF8C8h para o *Terminal Capabilities* (tag 9F33h);
- Solicita os seguintes objetos EMV se existentes no cartão: *Issuer Country Code* (tag 5F28h) e *Application Expiration Date* (tag 5F24h).

| | | |
|-------|---|--|
| SPE ⇒ | 47 43 58 30 36 39 00 17 00 05 30 30 30 30 30 00 10 00 02 30 38 00 13 00 0C 30 30 30 30 30 30 34 38 33 30 30 00 15 00 06 31 33 30 39 30 31 00 16 00 06 32 30 31 38 34 37 00 05 00 06 9F 33 03 E0 F8 C8 00 04 00 04 5F 28 5F 24 | GCX069....00000. ...08....0000000 48300....130901. ...201847....Y3. àøÈ.....(_\$ |
|-------|---|--|

Pinpad processa com sucesso um cartão ICC EMV.

| | | |
|------|--|---|
| ⇐ PP | 47 43 58 30 30 30 31 32 35 80 55 00 08 4A 4F 48 4E 20 44 4F 45 80 52 00 10 34 34 34 34 33 33 33 33 32 32 32 32 31 31 31 31 80 42 00 18 34 34 34 34 33 33 33 33 32 32 32 31 31 31 31 3D 31 36 30 38 32 30 31 80 4F 00 02 30 33 80 51 00 06 30 38 30 33 30 31 80 53 00 02 30 31 80 54 00 0B 5F 28 02 00 76 5F 24 03 16 08 31 80 5B 00 06 52 C9 44 49 54 4F 80 5C 00 04 30 30 37 36 80 5D 00 06 31 36 30 38 33 31 | GCX000125€U..JOH N•DOE€R..4444333 322221111€B..444 4333322221111=16 08201€O..03€Q..0 80301€S..01€T... (.v_\$...1€[.RÉ DITO€\..0076€].. 160831 |
|------|--|---|

O SPE inicia o processamento de uma transação com as seguintes características:

- Valor R\$ 1128,00, com *cashback* de R\$ 128,00;
- Todos os tipos de cartão (magnético, ICC e CTLS);
- Usa uma lista de registros específica das Tabelas de AID;
- Define um tempo de espera de 42 segundos;
- Define a mensagem a ser usada na solicitação do cartão como “POR FAVOR AMIGO, USE SEU CARTÃO COMO QUISER!”.

| | | |
|-------|---|---|
| SPE ⇒ | 47 43 58 31 33 34 00 0C 00 01 2A 00 1B 00 2C 50 4F 52 20 46 41 56 4F 52 20 41 4D 49 47 4F 2C 20 55 53 45 20 53 45 55 20 43 41 52 54 C3 4F 20 43 4F 4D 4F 20 51 55 49 53 45 52 21 00 12 00 10 30 31 30 31 30 32 30 35 30 33 30 38 32 35 30 34 00 13 00 0C 30 30 30 30 30 30 31 31 32 38 30 30 00 14 00 0C 30 30 30 30 30 30 30 31 32 38 30 30 00 15 00 06 31 34 30 37 32 35 00 16 00 06 30 38 32 35 35 39 00 17 00 05 31 30 30 30 30 | GCX134....*....,P OR•FAVOR•AMIGO,• USE•SEU•CARTÃO•C OMO•QUISER!....0 101020503082504. ...000000112800. ...000000012800. ...140725....082 559....10000 |
|-------|---|---|

Pinpad processa com sucesso um cartão ~~ICC~~ CTLS, entretanto informa que a transação pode ser processada por duas redes distintas no SPE.

| | | |
|------|--|--|
| ⇐ PP | 47 43 58 30 30 30 30 39 36 80 42 00 19 35 30 30 39 38 32 33 37 32 33 34 32 33 38 30 30 32 3D 31 37 30 31 36 30 30 80 4F 00 02 30 36 80 51 00 0C 30 32 30 35 30 33 32 35 30 34 30 33 80 52 00 11 35 30 30 39 38 32 33 37 32 33 34 32 33 38 30 30 32 80 53 00 02 30 30 80 5B 00 07 50 41 59 50 41 53 53 80 5C 00 03 38 34 30 | GCX000096€B..500 98237234238002=1 701600€o..06€Q.. 020503250403€R.. 5009823723423800 2€S..00€[. .PAYPA SS€\..840 |
|------|--|--|

3.7.2. Comando “GED”

| |
|---|
| <input type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando permite ao SPE obter dados do processamento EMV, desde que o comando “GCX” tenha sido executado previamente com sucesso para um ICC EMV (PP_CARDTYPE = “03”), CTLS simulando tarja (PP_CARDTYPE = “05”) ou CTLS EMV (PP_CARDTYPE = “06”).

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|--|
| CMD_ID | M | Código do comando (= “GED”). |
| SPE_TAGLIST | M | Lista de tags dos objetos EMV a serem retornados na resposta ao comando. |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|---|
| RSP_ID | M | Código da resposta (= “GED”). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_INVCALL..... Comando “GCX” não foi executado previamente com sucesso para ICC/CTLS. ↪ ST_RSPOVRFL..... Tamanho dos dados EMV ultrapassa máximo permitido para PP_EMVDATA. |
| PP_EMVDATA | M | Lista de objetos EMV definida por SPE_TAGLIST. Objetos não encontrados simplesmente não são devolvidos pelo pinpad, <u>assim como objetos que contenham informações de trilha de cartão (ou PAN).</u> |

➔ Exemplos

SPE solicita os seguintes objetos EMV se existentes no cartão: *Application Usage Control* (tag 9F07h), *Application Version Number* (tag 9F08h), *ADF Name* (4Fh) e um objeto proprietário de tag DF55h.

| | | |
|-------|--|-----------------------|
| SPE ⇒ | 47 45 44 30 31 31 00 04 00 07 9F 07 9F 08 4F DF 55 | GED011....ÿ.ÿ.oß U |
|-------|--|-----------------------|

Pinpad retorna objetos solicitados com exceção do *Application Version Number* (tag 9F08h), por não ser conhecido.

| | | |
|------|---|---|
| ← PP | 47 45 44 30 30 30 30 32 39 80 54 00 19 9F 07 02 FF 00 4F 07 A0 00 00 00 03 10 10 DF 55 08 11 22 33 44 55 66 77 88 | GED000029€T..ÿ.. ÿ.o.ßU.. 3DUfw^ |
|------|---|---|

3.7.3. Comando “GOX”

| |
|--|
| <input type="checkbox"/> Obsoleto |
| <input checked="" type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando continua o processo de tratamento de cartões com *chip* caso o comando “GCX” tenha sido executado previamente com sucesso para um ICC EMV (PP_CARDTYPE = “03”) ou CTLS EMV (PP_CARDTYPE = “06”), conforme apresentado no fluxo da seção 3.7.5.

Este comando é equivalente ao comando “GOC”, porém com as seguintes diferenças:

- Utiliza o formato Abecs para permitir flexibilidade e facilitar evoluções futuras.
- Permite ao SPE enviar ao pinpad uma lista de parâmetros EMV a serem usados no processamento (útil somente no caso de ICC EMV!).
- Permite ao SPE definir a mensagem de apresentação no *display* caso seja requerida a entrada de PIN.

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|---------------------|----------|---|
| <u>CMD_ID</u> | M | Código do comando (= “GOX”). |
| <u>SPE_ACQREF</u> | M | Identificador da Rede Credenciadora (<u>TAB_ACQ</u>) cujas Tabelas EMV serão usadas no processamento. Deve-se selecionar, obrigatoriamente, uma das Redes Credenciadoras devolvidas em <u>PP_AIDTABINFO</u> na resposta de “GCX”. |
| <u>SPE_TRNTYPE</u> | O | Tipo de transação a ser efetuada: 00h = Compra; 01h = Saque; 09h = Compra com saque/troco (<i>cashback</i>); 20h = Cancelamento (<i>refund</i>); 30h = Consulta de saldo; ou Outros valores de acordo com ISO 8583:1987. Se este parâmetro não for fornecido, o pinpad considerará a transação como sendo de “compra” (se <u>SPE_CASHBACK</u> ausente) ou “ <i>cashback</i> ” (se <u>SPE_CASHBACK</u> presente e diferente de zero). |
| <u>SPE_AMOUNT</u> | O | Valor da transação em centavos (<i>Amount, authorized</i>). Se este parâmetro estiver ausente, o pinpad considerará este dado como zerado. |
| <u>SPE_CASHBACK</u> | O | Valor da transação referente a saque ou troco - <i>cashback</i> (<i>Amount, other</i>) em centavos. Se este parâmetro estiver ausente, o pinpad considerará este dado como zerado. |

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| <u>SPE_TRNCURR</u> | O | Código da moeda a ser usada na transação (<i>Transaction Currency Code</i>), <u>somente para ICC</u> . Se este parâmetro estiver ausente, o pinpad usa: <ul style="list-style-type: none"> ▪ O valor passado em <u>SPE_TRNCURR</u> no comando "<u>GCTX</u>", se existente; ou ▪ O valor definido em <u>T1_TRNCURR</u>. |
| <u>SPE_GOLOPT</u> | O | Opções do comando: " <u>1xxxx</u> " = PAN consta na Lista de Exceção (só usado se ICC EMV). " <u>x1xxx</u> " = Transação não pode ser aprovada <i>offline</i> (só usado se ICC EMV). " <u>xx1xx</u> " = Não permite <i>bypass</i> de PIN. " <u>xxx00</u> " = RUF. Se este parâmetro estiver ausente, o pinpad considerará este dado como zerado. |
| <u>SPE_MTHDPIN</u> | M | Método de criptografia de PIN <i>online</i> , a ser usado caso requerido pelo processamento EMV. " <u>1</u> " = MK/WK:TDES:PIN; e " <u>3</u> " = DUKPT:TDES:PIN (ver seção 5.1.1). |
| <u>SPE_KEYIDX</u> | M | Índice da MK ou do registro de tratamento DUKPT a ser usado na criptografia de PIN <i>online</i> . |
| <u>SPE_WKENC</u> | MD | <i>Working Key</i> (criptografada pela MK) a ser usada na criptografia de PIN <i>online</i> . Este campo é mandatório somente se <u>SPE_MTHDPIN</u> = " <u>0</u> " ou " <u>1</u> ". |
| <u>SPE_DSPMSG</u> | O | Mensagem a ser apresentada no <i>display</i> do pinpad no caso de uma captura de PIN. Se este parâmetro não for fornecido, o pinpad usa uma mensagem padrão. |
| <u>SPE_TRMPAR</u> | O | Parâmetros para o processamento do <i>Terminal Risk Management</i> , sendo a concatenação dos seguintes dados: <ul style="list-style-type: none"> ▪ <i>Terminal Floor Limit</i> (formato "<u>X4</u>", em centavos); ▪ <i>Target Percentage to be used for Biased Random Selection</i> (formato "<u>X1</u>"); ▪ <i>Threshold Value for Biased Random Selection</i> (formato "<u>X4</u>", em centavos); e ▪ <i>Maximum Target Percentage to be used for Biased Random Selection</i> (formato "<u>X1</u>"). Se este campo estiver ausente, o pinpad efetua o <i>Terminal Risk Management</i> com valores zerados. |
| <u>SPE_EMVDATA</u> | O | Lista opcional de parâmetros (no formato TLV), para uso no processamento de ICC EMV . Os dados aqui fornecidos têm prioridade em relação aos objetos das Tabelas de AID, caso sejam coincidentes. |
| <u>SPE_TAGLIST</u> | O | Lista de <i>tags</i> dos objetos EMV a serem retornados na resposta ao comando. |

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|---|
| <u>SPE_TIMEOUT</u> | O | Tempo máximo de inatividade na tela de captura de PIN. Se este campo estiver ausente, o pinpad considerará <u>1 minuto</u> (60 segundos). |

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|------------------|----------|---|
| <u>RSP_ID</u> | M | Código da resposta (= "GOX"). |
| <u>RSP_STAT</u> | M | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_INVCALL..... Comando "GCX" não foi executado previamente com sucesso para ICC CTLS/EMV. ↳ ST_RSPOVRFL..... Tamanho dos dados EMV ultrapassa máximo permitido para <u>PP_EMVDATA</u> . ↳ ST_ERRKEY Problema na chave de criptografia de PIN. ↳ ST_TIMEOUT Tempo esgotado na tela de captura de PIN. ↳ ST_CARDPROBLEMS... Cartão inválido ou com problemas. ↳ ST_CARDINVDATA..... ICC com dados inválidos ou faltando. ↳ ST_ERRFALLBACK Erro sujeito a "fallback" para tarja. ↳ ST_TIMEOUT Esgotado tempo definido em <u>SPE_TIMEOUT</u> para captura de PIN. |
| <u>PP_GOXRES</u> | M | Resultados do processamento EMV: "0xxxxx" = Transação aprovada <i>offline</i> ; "1xxxxx" = Transação negada; ou "2xxxxx" = Transação requer autorização <i>online</i> . "x1xxxx" = Deve-se coletar assinatura em papel. "xx1xxx" = PIN foi verificado com sucesso <i>offline</i> . "xx2xxx" = PIN capturado para verificação <i>online</i> . "xxx1xx" = Verificação de portador efetuada no dispositivo móvel (telefone celular, por exemplo). "xxxx00" = RUF. |
| <u>PP_PINBLK</u> | MD | PIN criptografado para verificação <i>online</i> . Este campo é mandatório se <u>PP_GOXRES</u> = "xx2xxx". |
| <u>PP_KSN</u> | MD | Número de série da chave DUKPT usada na criptografia do PIN. Este campo é mandatório se <u>PP_GOXRES</u> = "xx2xxx" e <u>SPE_MTHDPIN</u> = "3" (DUKPT:TDES:PIN). |

| Id. do Campo | Presença | Descrição / Observação |
|-------------------|----------|---|
| <u>PP_EMVDATA</u> | MR | <p>Lista de objetos EMV definida por <u>SPE_TAGLIST</u>. Objetos não encontrados simplesmente não são devolvidos pelo pinpad, <u>assim como objetos que contenham informações de trilha de cartão (ou PAN)</u>.</p> <p>Este campo é mandatório sempre que <u>SPE_TAGLIST</u> existir no comando, <u>mesmo que nenhum objeto seja encontrado</u> (caso em que é retornado com tamanho zerado).</p> |

➔ Exemplos

SPE solicita a continuação do processamento de um ICC EMV com as seguintes características:

- Processar usando tabelas da Rede Credenciadora “08”;
- Valor da transação R\$ 234,50, com *cashback* de R\$ 100,00;
- Se houver necessidade PIN *online*, usar DUKPT:TDES de posição “07”;
- Efetuar o *Terminal Risk Management* com: *Floor Limit* = R\$ 100,00; *Target Percentage to be used for Biased Random Selection* = 20%; *Threshold Value for Biased Random Selection* = R\$ 25,00; *Maximum Target Percentage to be used for Biased Random Selection* = 80%;
- Sem parâmetros ~~EMV~~ opcionais de EMV; e
- Solicita os objetos TVR (*tag* 95h), *Application Cryptogram* (*tag* 9F26h), *Cryptogram Information Data* (*tag* 9F27h), *Issuer Application Data* (*tag* 9F10h), *CVM Results* (*tag* 9F34h) e ATC (*tag* 9F36h).

| | | |
|-------|--|---|
| SPE ⇒ | <pre> 47 4F 58 31 31 36 00 13 00 0C 30 30 30 30 30 30 30 32 33 34 35 30 00 14 00 0C 30 30 30 30 30 30 30 31 30 30 30 30 00 02 00 01 33 00 09 00 02 30 37 00 1B 00 22 43 52 C9 44 49 54 4F 0D 52 24 20 32 33 34 2C 35 30 0D 44 49 47 49 54 45 20 53 55 41 20 53 45 4E 48 41 00 1A 00 0A 00 00 27 10 14 00 00 00 19 50 00 04 00 0B 95 9F 26 9F 27 9F 10 9F 34 9F 36 00 10 00 02 30 38 </pre> | <pre> GOX116....000000 023450....000000 010000....3....0 7..."CRÉDITO.R\$. 234,50.DIGITE.SU A.SENHA.....'P....Y&Y'Y. Y4Y6....08 </pre> |
|-------|--|---|

Pinpad finaliza a operação com sucesso (sendo que o cartão requisita autorização *online*), devolvendo os dados EMV requeridos.

| | | |
|------|---|--|
| ⇐ PP | <pre> 47 4F 58 30 30 30 30 38 38 80 56 00 06 32 30 32 30 30 30 80 54 00 30 95 05 00 80 00 00 00 9F 26 08 E0 DB 51 A3 74 2F EA 83 9F 27 01 80 9F 10 0C 2C 51 4D 27 0F C3 CD 87 6C A4 00 00 9F 34 03 42 03 02 9F 36 02 00 4C 80 57 00 08 B9 DF 0A 99 6E A6 CC B7 80 4C 00 0A FF FF F7 98 41 00 34 40 00 08 </pre> | <pre> GOX000088€v..202 000€T.0...€...Y& .àÛqft/éfY'.€Y.. ,QM'.ÃÍ†lα..Y4.B ..Y6..L€W..¹ß.™n !Ì.€L..ÿÿ÷~A.4@. . </pre> |
|------|---|--|

3.7.4. Comando “FCX”

| |
|--|
| <input type="checkbox"/> Obsoleto |
| <input checked="" type="checkbox"/> Blocante |
| <input checked="" type="checkbox"/> ABECS |

Este comando é equivalente ao comando “**FNC**”, porém utilizando o formato Abecs. Ele finaliza o processamento de cartão com *chip* e deve ser chamado caso “**GCX**” tenha requerido aprovação *online* (**PP_GOXRES** = “2xxxx”), conforme apresentado na **seção 3.7.5**.

No caso de aprovação ou negação *offline* (**PP_GOXRES** = “0xxxx” ou “1xxxx”), este comando pode ser chamado, de acordo com a especificação da Rede Credenciadora (para o caso de processamento de *Issuer Scripts* de manutenção em ICC, por exemplo).

No caso de CTLS, este comando pode solicitar a reapresentação do mesmo cartão processado em “**GCX**” para execução de *Issuer Scripts* de manutenção, situação na qual o comando assume um comportamento bloqueante².

➔ Comando

| Id. do Campo | Presença | Descrição / Observação |
|--------------------|----------|--|
| CMD_ID | M | Código do comando (= “ FCX ”). |
| SPE_FCXOPT | M | Resultado da comunicação com a Rede Credenciadora: “0xxx” = Transação <u>aprovada</u> pela Rede Credenciadora. “1xxx” = Transação <u>negada</u> pela Rede Credenciadora. “2xxx” = A comunicação foi malsucedida (ou não foi possível receber uma resposta válida da Rede Credenciadora). “x000” = RUF. |
| SPE_ARC | MD | <i>Authorization Response Code</i> (código de aprovação/negação devolvido pela Rede Credenciadora). Este parâmetro é mandatório se SPE_FCXOPT = “0xxx” ou “1xxx”. |
| SPE_EMVDATA | O | Objetos TLV opcionalmente recebidos da Rede Credenciadora, podendo conter o <i>Issuer Authentication Data</i> (<i>tag</i> 91h) e <i>Issuer Scripts</i> (<i>tags</i> 71h e 72h). |
| SPE_TAGLIST | O | Lista de <i>tags</i> dos objetos EMV a serem retornados na resposta ao comando. |
| SPE_TIMEOUT | O | Tempo de espera para reapresentação do CTLS caso isso seja requerido. |

² Neste caso, o SPE pode cancelar a operação mediante envio de um «CAN», porém deve considerar o resultado da transação (aprovação/negação) informado pela resposta da Rede Credenciadora, dado que o pinpad somente devolverá um «EOT» e, portanto, o campo **PP_FCXRES** não estará presente.

➔ Resposta

| Id. do Campo | Presença | Descrição / Observação |
|--------------|----------|---|
| RSP_ID | M | Código da resposta (= "FCX"). |
| RSP_STAT | M | Retornos de erro relevantes (ver seção 3.1.1): ↪ ST_INVCALL..... Comando "GOX" não foi executado previamente com sucesso. ↪ ST_RSPOVRFL..... Tamanho dos dados EMV ultrapassa máximo permitido para PP_EMVDATA. ↪ ST_CARDPROBLEMS... Cartão inválido ou com problemas. ↪ ST_CARDINVDATA..... ICC com dados inválidos ou faltando. |
| PP_FCXRES | M | Resultado do processamento EMV: "0xx" = Transação aprovada; ou "1xx" = Transação negada. "x00" = RUF. |
| PP_EMVDATA | MR | Lista de objetos EMV definida por SPE_TAGLIST. Objetos não encontrados simplesmente não são devolvidos pelo pinpad, <u>assim como objetos que contenham informações de trilha de cartão (ou PAN)</u> . Este campo é mandatório sempre que SPE_TAGLIST existir no comando, <u>mesmo que nenhum objeto seja encontrado</u> (caso em que é retornado com tamanho zerado). |
| PP_ISRESULTS | O | Resultado do processamento de scripts do emissor (<i>Issuer Script Results</i>). Este campo somente estará presente se o comando receber <i>Issuer Scripts</i> em SPE_EMVDATA. |

➔ Exemplos

SPE solicita a finalização do processamento de um ICC EMV com as seguintes características:

- A Rede Credenciadora aprova a transação *online*, porém com código de resposta "Y3";
- A Rede Credenciadora devolve o objeto *Issuer Authentication Data* (tag 91h) e um *Issuer Script* (tag 72h); e
- Solicita os objetos TVR (tag 95h), *Application Cryptogram* (tag 9F26h), *Cryptogram Information Data* (tag 9F27h) e *Issuer Application Data* (tag 9F10h).

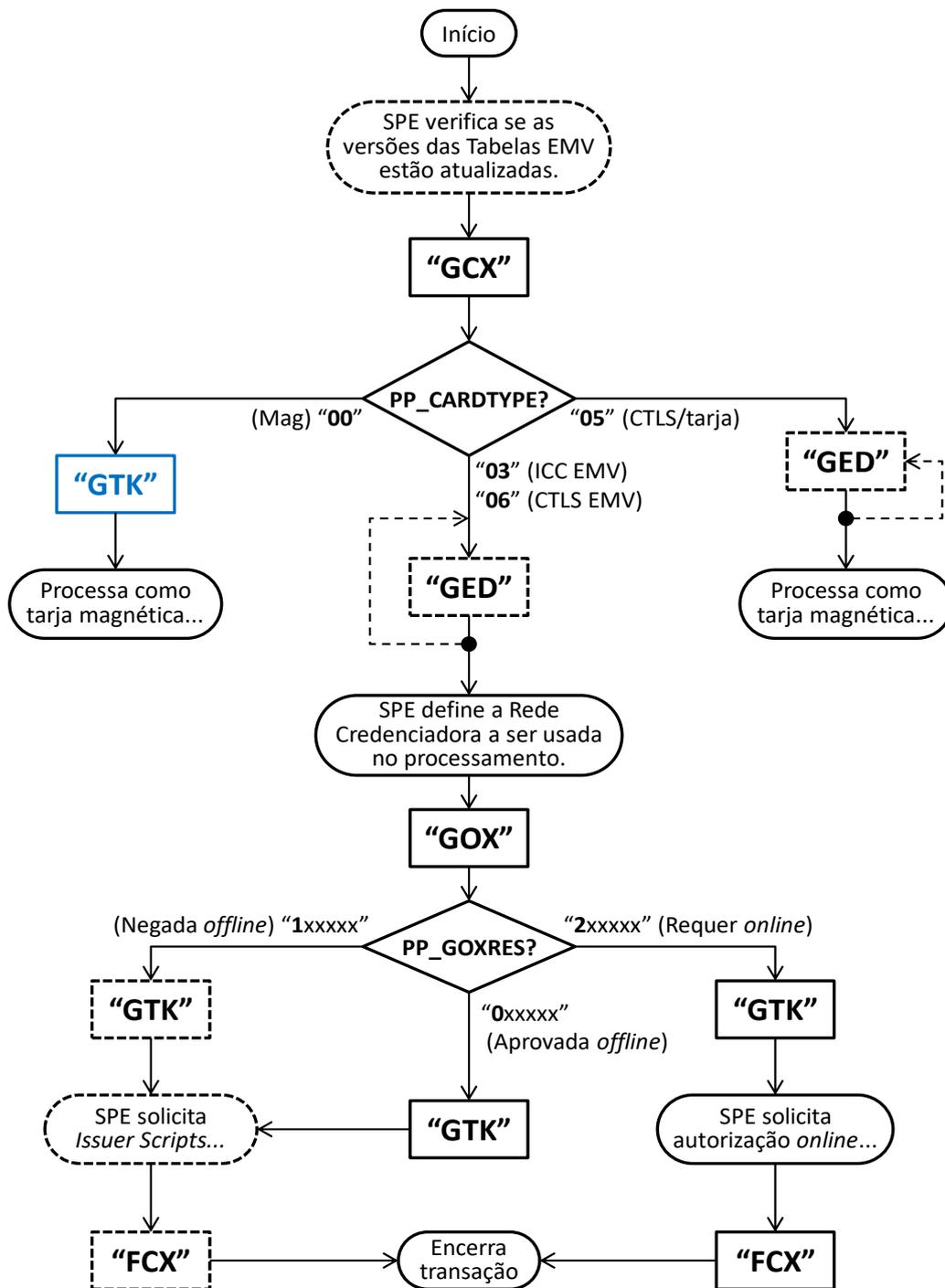
| | | |
|-------|--|---|
| SPE ⇒ | 46 43 58 30 35 39 00 05 00 1E 91 08 A1 02 DB 6D 41 C6 79 63 72 12 9F 18 00 86 0D 84 24 00 00 08 A0 71 54 4A 23 76 1A A1 00 04 00 07 95 9F 26 9F 27 9F 10 00 1C 00 02 59 33 00 19 00 04 30 30 30 30 | FCX059....' .: .Ùm AÆycr.ÿ..†.,\$. qTJ#v.i....•ÿ&ÿ 'ÿ.....Y3....000 0 |
|-------|--|---|

Pinpad finaliza a operação com sucesso (aprovação) e devolve o *Issuer Script Results*, bem como os objetos EMV solicitados.

| | | |
|------|--|---|
| ⇐ PP | 46 43 58 30 30 30 30 35 35 80 56 00 03 30 30 30 80 59 00 05 20 00 00 00 00 80 54 00 23 95 05 00 80 00 00 00 9F 26 08 95 24 B3 FC 02 5E 51 72 9F 27 01 40 9F 10 0A 7D 89 5F FF F0 15 D7 72 FB C9 | FCX000055€v..000 €Y.....€T.#... €...ÿ&..\$³ü.ΛQrÿ '.@ÿ..}‰_ÿð.xrûÉ |
|------|--|---|

3.7.5. Fluxo de operação

O fluxo a seguir ilustra a sequência de chamada dos comandos Abecs de processamento de cartão. Os blocos pontilhados referem-se a processamentos opcionais que dependem da especificação da Rede Credenciadora.



~~3.8. Comandos genéricos~~

~~Esta seção inclui os comandos proprietários gerados no passado pelas Redes Credenciadoras e que constam nesta especificação para preservar a compatibilidade com SPE legados.~~

~~Os seguintes comandos estão contemplados nesta seção:~~

| EMD_ID | GEN_ACQ | GEN_CMD | Significado | Obsoleto | Blocante | Abecs |
|-------------------|--------------------|--------------------|---|--|--|-------------------------------------|
| "GEN" | "02" | "K3" | Pesquisa chaves de criptografia armazenadas no pinpad. | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "GEN" | "03" | "02" | Obtém dados de processamento de cartões EMV. | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "GEN" | "04" | "01" | Controla e consulta o status de leitor de cartão com chip. | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "GEN" | "04" | "02" | Troca comandos com o cartão com chip. | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| "GEN" | "04" | "03" | Captura PIN e envia a o cartão com chip. | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| "GEN" | "04" | "04" | Obtém dados de processamento de cartões EMV. | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

~~3.8.1. Comando “GEN/02/K3”~~

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

~~Este comando pesquisa chaves de criptografia armazenadas no pinpad.~~

~~⚠ Este comando é **obsoleto**. O SPE deve usar o comando “GIX” para esta funcionalidade.~~

~~Comando~~

| Id do Campo | Formate | Descrição |
|--------------------------|--------------------|--|
| CMD_ID | A3 | Código do comando (= “GEN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “013”). |
| GEN_ACQ | N2 | Identificador da Rede Credenciadora (fixo “02”). |
| GEN_INLEN | N3 | Tamanho dos dados a seguir (fixo “008”). |
| GEN_CMD | A2 | Código do comando (“K3”). |
| G02K3_KEYTYPE | N1 | Tipo de chave sendo pesquisada: “1” = MK:TDES; “3” = DUKPT:TDES. |
| G02K3_KEYUSE | N1 | Uso da chave: “0” = Criptografia de PIN; “1” = Criptografia de dados. |
| G02K3_IDXINI | N2 | Índice “aa” do início da pesquisa (de “00” a “99”). |
| G02K3_IDXEND | N2 | Índice “bb” do final da pesquisa (de “aa” a “99”). |

~~Resposta~~

| Id do Campo | Formate | Descrição |
|------------------------|--------------------|---|
| RSP_ID | A3 | Código da resposta (= “GEN”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_INVPARM Parâmetro inválido (exemplo: se “aa” → “bb”). |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| GEN_OUTLEN | N3 | Tamanho dos dados a seguir. |
| G02K3_KEYST | N1 | Estado da chave “aa”: “0” = Ausente (não carregada); “1” = Presente (carregada); ou “2” = Chave não suportada pelo pinpad. |

| Id do Campo | Formato | Descrição |
|------------------------|--------------------|-----------------------------------|
| G02K3=KEYST | N1 | Estado da chave "aa+1" |
| G02K3=KEYST | N1 | Estado da chave "aa+2" |
| ≡ | ≡ | ≡ |
| G02K3=KEYST | N1 | Estado da chave "bb" |

Exemplos

~~O SPE solicita ao pinpad informações sobre as chaves Master Key Triple-DES (para PIN) dos índices de "05" a "15".~~

| | | |
|------------------|---|-------------------------------------|
| SPE → | 47 45 4E 30 31 33 30 32 30 30 38 4B 33 31 30 30 35 31 35 | GEN01302008K3100 515 |
|------------------|---|-------------------------------------|

~~O pinpad devolve 11 caracteres (representando as chaves de "05" a "15"), indicando que somente a chave de índice "08" está presente nesta faixa.~~

| | | |
|-----------------|---|---|
| ← PP | 47 45 4E 30 30 30 30 31 34 30 31 31 30 30 30 31 30 30 30 30 30 30 30 | GEN0000140110001 0000000 |
|-----------------|---|---|

~~O SPE solicita ao pinpad informações sobre as chaves DUKPT Triple-DES (para dados) dos índices de "00" a "40".~~

| | | |
|------------------|---|-------------------------------------|
| SPE → | 47 45 4E 30 31 33 30 32 30 30 38 4B 33 33 31 30 30 34 30 | GEN01302008K3310 040 |
|------------------|---|-------------------------------------|

~~O pinpad devolve 41 caracteres (representando as chaves de "00" a "40"), indicando que as chaves de índices "05", "15" e "19" estão presentes nesta faixa, e que as chaves a partir do índice "25" não são suportadas.~~

| | | |
|-----------------|--|---|
| ← PP | 47 45 4E 30 30 30 30 34 34 30 34 31 30 30 30 30 30 31 30 30 30 30 30 30 30 30 30 31 30 30 30 31 30 30 30 30 30 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | GEN0000440410000 0100000000010001 0000022222222222 22222 |
|-----------------|--|---|

~~3.8.2. Comando “GEN/04/01”~~

Obsoleto
 Blocante
 ABECS

~~Este comando é usado para controlar e consultar o status do leitor de ICC.~~

~~O comando pode ser utilizado para executar um “power on”, “reset” ou “power off” no ICC, ou apenas para consultar o status atual do leitor e do cartão.~~

~~▲ Este comando é **obsoleto**. O SPE deve usar o comando “**CHP**” para esta funcionalidade.~~

Comando

| Id. do Campo | Formate | Descrição |
|-------------------------|--------------------|---|
| CMD_ID | A3 | Código do comando (= “GEN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “009”). |
| GEN_ACQ | N2 | Identificador da Rede Credenciadora (fixo “04”). |
| GEN_INLEN | N3 | Tamanho dos dados a seguir (fixo “004”). |
| GEN_CMD | A2 | Código do comando (“01”). |
| G0401_OPER | N1 | Código da operação: “1” = Executar um “power on” no ICC; “2” = Executar um “reset” no ICC; “3” = Executar um “power off” no ICC; “4” = Retornar o status atual do ICC. |
| G0401_SLOT | N1 | Identificação do cartão a ser usado: “0” = ICC no acoplador principal; “1” = SAM na posição #1; ... “8” = SAM na posição #8. |

Resposta

| Id. do Campo | Formate | Descrição |
|-------------------------|--------------------|--|
| RSP_ID | A3 | Código da resposta (= “GEN”). |
| RSP_STAT | N3 | Ver seção 3.1.1. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo “079”). |
| GEN_OUTLEN | N3 | Tamanho dos dados a seguir (fixo “076”). |
| G0401_SLOT | N1 | Identificação do cartão usado (igual a G0401_SLOT do comando). |

| Id. do Campo | Formato | Descrição |
|-------------------------|--------------------|--|
| G0401_STAT | N1 | Status de presença e ativação do ICC: "0" = Cartão ausente ("absent"); "1" = Cartão presente, mas inativo ("present"); "2" = Cartão presente, sem comunicação ("mute"); "3" = Cartão presente, alimentado e protocolo estabelecido ("active"); e "4" = Leitor inexistente ou com defeito. |
| G0401_ATRLEN | N2 | Quantidade de bytes significativos representados em G0401_ATR (tamanho : 2). |
| G0401_ATR | H72 | ATR do cartão, caso G0401_STAT = "3" ("active"). |

~~3.8.3. Comando “GEN/04/02”~~

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

~~Este comando é usado para troca comandos APDU com o ICC, que deve ter sido previamente ativado através dos comandos “GEN/04/01” ou “GCR”.~~

~~▲ Este comando é **obsoleto**. O SPE deve usar o comando “CHP” para esta funcionalidade.~~

~~Comando~~

| Id. do Campo | Formato | Descrição |
|-------------------------|--------------------|---|
| CMD_ID | A3 | Código do comando (= “GEN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| GEN_ACQ | N2 | Identificador da Rede Credenciadora (fixo “04”). |
| GEN_INLEN | N3 | Tamanho dos dados a seguir. |
| GEN_CMD | A2 | Código do comando (“02”). |
| G0402_SLOT | N1 | Identificação do cartão a ser usado: “0” = ICC no acoplador principal; “1” = SAM na posição #1; ... “8” = SAM na posição #8. |
| G0402_APDU | H8 | Cabeçalho do comando APDU (CLA, INS, P1, P2). |
| G0402_LC | N3 | Tamanho (Lc) em bytes dos dados a serem enviados para o ICC (“000” a “255”). |
| G0402_CMDDAT | H..510 | Dados do comando, conforme tamanho no campo anterior (Lc). |
| G0402_LE | N3 | Tamanho (Le) em bytes dos dados de resposta esperados do ICC (“000” a “256”). |

~~Resposta~~

| Id. do Campo | Formato | Descrição |
|-------------------------|--------------------|---|
| RSP_ID | A3 | Código da resposta (= “GEN”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ST_ERRCARD Erro de comunicação entre o pinpad e o ICC ↳ST_NOCARD ICC ausente ou removido. ↳ST_DUMBCARD ICC inserido, mas não responde. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| GEN_OUTLEN | N3 | Tamanho dos dados a seguir. |
| G0402_SLOT | N1 | Identificação do cartão usado (igual a G0402_SLOT do comando). |

| Id do Campo | Formato | Descrição |
|-------------------------|--------------------|---|
| G0402_SW | H4 | Status Word (SW1, SW2) recebido do ICC. |
| G0402_LA | N3 | Tamanho (La) em bytes dos dados recebidos do ICC em resposta a comando APDU enviado ("000" a "256"). |
| G0402_RSPDAT | H..512 | Dados da resposta, conforme tamanho no campo anterior (La). |

~~3.8.4. Comando “GEN/04/03”~~

- Obsoleto
- Blocante
- ABECS

~~Este comando é usado para se efetuar uma captura de PIN no pinpad e já apresentá-lo ao ICC para validação *offline*.~~

~~Este comando é **obsoleto**. O SPE deve usar o comando “**CHP**” para esta funcionalidade.~~

~~Comando~~

| Id. do Campo | Formato | Descrição |
|-------------------------|--------------------|--|
| CMD_ID | A3 | Código do comando (= “GEN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir (fixo “053”). |
| GEN_ACQ | N2 | Identificador da Rede Credenciadora (fixo “04”). |
| GEN_INLEN | N3 | Tamanho dos dados a seguir (fixo “048”). |
| GEN_CMD | A2 | Código do comando (“03”). |
| G0403_SLOT | N1 | Identificação do cartão a ser usado: “0” = ICC no acoplador principal; “1” = SAM na posição #1; ... “8” = SAM na posição #8. |
| G0403_APDU | H8 | Cabeçalho do comando APDU (CLA, INS, P1, P2) que o pinpad utilizará para submeter o PIN ao ICC. |
| G0403_FORMAT | H1 | Formato do bloco do PIN a ser submetido ao cartão: “0” = Formato ISO-0; “2” = Formato ISO-2; ou “A” = Sequência de dígitos ASCII. |
| G0403_MIN | N2 | Quantidade mínima de dígitos aceita para o PIN. |
| G0403_MAX | N2 | Quantidade máxima de dígitos aceita para o PIN. |
| G0403_PINMSG | S32 | Mensagem de 2 linhas por 16 colunas para apresentação no momento do pedido do PIN. |

~~Resposta~~

| Id. do Campo | Formato | Descrição |
|-------------------------|--------------------|--|
| RSP_ID | A3 | Código da resposta (= “GEN”). |

| Id. do Campe | Formate | Descrição |
|-------------------------|--------------------|--|
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ST_CANCEL Portador pressionou a tecla [CANCELA]. ↳ST_ERRCARD Erro de comunicação entre o pinpad e o ICC. ↳ST_NOCARD ICC ausente ou removido. ↳ST_DUMBCARD ICC inserido, mas não responde. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir (fixo "008"). |
| GEN_OUTLEN | N3 | Tamanho dos dados a seguir (fixo "005"). |
| G0403_SLOT | N1 | Identificação do cartão usado (igual a G0403_SLOT do comando). |
| G0403_SW | H4 | Status Word (SW1, SW2) recebido do ICC. |

~~3.8.5. Comando “GEN/04/04”~~

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

~~Este comando é usado para recuperar um ou mais elementos de dados adicionais (campos em formato TLV) durante a execução de uma transação EMV, através do fornecimento de uma lista de tags.~~

~~Este comando somente pode ser utilizado após a execução bem sucedida do comando “GCR”, “GOC” ou “FNC”, podendo ser chamado mais de uma vez de acordo com a necessidade do SPE.~~

~~**⚠** Este comando é **obsoleto**. O SPE deve usar o comando “GED” para esta funcionalidade.~~

~~Comando~~

| Id. do Campo | Formato | Descrição |
|-------------------------|--------------------|---|
| CMD_ID | A3 | Código do comando (= “GEN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| GEN_ACQ | N2 | Identificador da Rede Credenciadora (fixo “04”). |
| GEN_INLEN | N3 | Tamanho dos dados a seguir. |
| GEN_CMD | A2 | Código do comando (“04”). |
| G0404_TAGSLN | N3 | Quantidade de bytes representados em G0404_TAGS (tamanho ÷ 2). |
| G0404_TAGS | H..256 | Lista de tags identificando os objetos de dados EMV a serem devolvidos em G0404_EMVDAT. As tags devem ser simplesmente concatenadas, respeitando-se sua regra de formação (ver seção 7.1). |

~~Resposta~~

| Id. do Campo | Formato | Descrição |
|-------------------------|--------------------|---|
| RSP_ID | A3 | Código da resposta (= “GEN”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ ST_INVCALL Comando “GCR” não foi executado previamente com sucesso para ICC-EMV. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| GEN_OUTLEN | N3 | Tamanho dos dados a seguir. |
| G0404_EMVDTL | N3 | Quantidade de bytes representados em G0404_EMVDAT (tamanho ÷ 2). |
| G0404_EMVDAT | H..512 | Dados EMV, no formato TLV (ver seção 7.1). O pinpad concatena os dados pedidos por G0404_TAGS, se encontrados, respeitando a ordem em que foram solicitados. |

~~3.8.6. Comando “GEN/03/02”~~

| |
|--|
| <input checked="" type="checkbox"/> Obsoleto |
| <input type="checkbox"/> Blocante |
| <input type="checkbox"/> ABECS |

~~Este comando é usado para recuperar um ou mais elementos de dados adicionais (campos em formato TLV) durante a execução de uma transação EMV, através do fornecimento de uma lista de tags.~~

~~Este comando somente pode ser utilizado após a execução bem sucedida do comando “GCR”, “GOC” ou “FNC”, podendo ser chamado mais de uma vez de acordo com a necessidade do SPE.~~

~~⚠ Este comando é **obsoleto**. O SPE deve usar o comando “GED” para esta funcionalidade.~~

~~Comando~~

| Id do Campo | Formato | Descrição |
|------------------------|--------------------|--|
| CMD_ID | A3 | Código do comando (“GEN”). |
| CMD_LEN1 | N3 | Tamanho dos dados a seguir. |
| GEN_ACQ | N2 | Identificador da Rede Credenciadora (fixo “03”). |
| GEN_INLEN | N3 | Tamanho dos dados a seguir. |
| GEN_CMD | A2 | Código do comando (“02”). |
| G0302_TAGS | H..256 | Lista de tags identificando os objetos de dados EMV a serem devolvidos em G0302_EMV DAT. As tags devem ser simplesmente concatenadas, respeitando-se sua regra de formação (ver seção 7.1). |

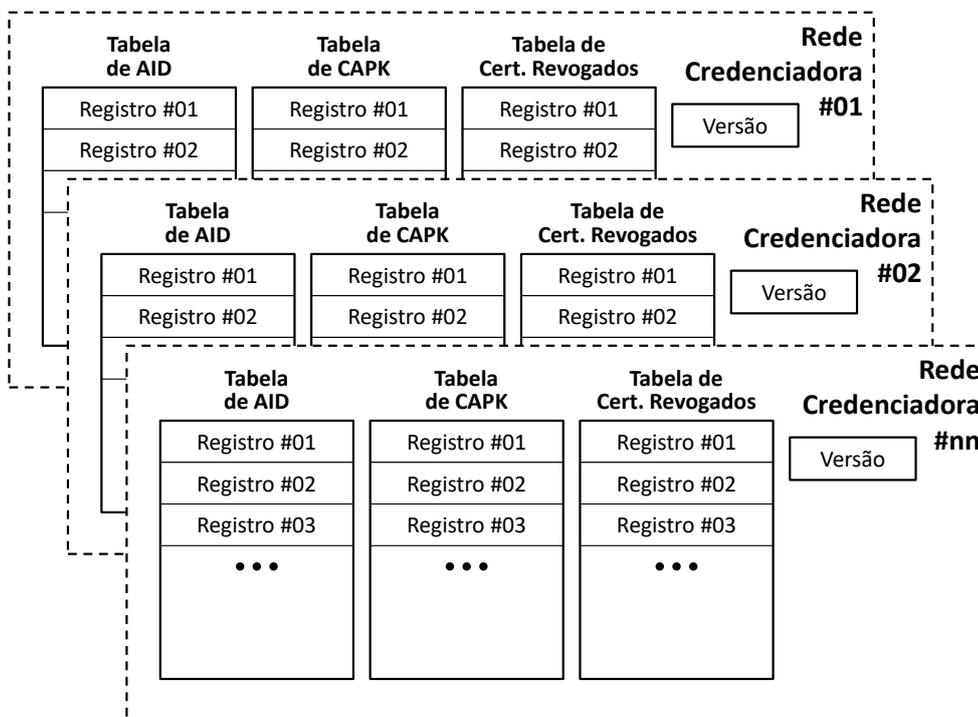
~~Resposta~~

| Id do Campo | Formato | Descrição |
|--------------------------|--------------------|---|
| RSP_ID | A3 | Código da resposta (“GEN”). |
| RSP_STAT | N3 | Retornos de erro relevantes (ver seção 3.1.1): ↳ST_INVCALL Comando “GCR” não foi executado previamente com sucesso para ICC EMV. |
| RSP_LEN1 | N3 | Tamanho dos dados a seguir. |
| GEN_OUTLEN | N3 | Tamanho dos dados a seguir. |
| G0302_EMV DAT | H..512 | Dados EMV, no formato TLV (ver seção 7.1). O pinpad concatena os dados pedidos por G0302_TAGS, se encontrados, respeitando a ordem em que foram solicitados. |

4. Gerenciamento de Tabelas EMV

Para otimizar o processamento de cartões EMV (ICC ou CTLS) nos comandos das **seções 3.6 e 3.7**, o pinpad precisa ser pré-carregado com um conjunto de tabelas de parâmetros, que são armazenadas de forma “não volátil” (são preservadas mesmo quando o pinpad é desligado).

Estas tabelas são separadas por Rede Credenciadora conforme diagrama:



Estas tabelas são geradas pelo SPE (a partir das informações recebidas das Redes Credenciadoras) e são transferidas para o pinpad utilizando-se os comandos descritos na **seção 3.5**.

- ⚠ A consistência dos registros enviados é de total responsabilidade do SPE, sendo que o pinpad não faz críticas complexas, como, por exemplo, identificar colisões de índices e registros. O pinpad simplesmente despreza registros cujo conteúdo é notavelmente inválido.

4.1. Tipos de Tabela

Os registros das tabelas, independentemente do seu tipo, possuem o seguinte formato padronizado:

| Id. do Campo | Formato | Descrição |
|--------------|---------|--|
| TAB_LEN | N3 | Tamanho <u>total</u> do registro, incluindo este campo. |
| TAB_ID | N1 | Identificação da tabela: "1" = Tabela de AID; "2" = Tabela de CAPK; e "3" = Tabela de Certificados Revogados. |
| TAB_ACQ | N2 | Identificador da Rede Credenciadora responsável pela tabela (de "01" a "99"). |
| TAB_RECIDX | A2 | Índice do registro na tabela (de "01" a "ZZ"). |
| ... | .. | ... |

Observações:

- Cada registro deve ter um valor diferente de **TAB_RECIDX** (não necessariamente sequencial) para uma determinada Rede Credenciadora.
- O conjunto **TAB_ID**, **TAB_ACQ** e **TAB_RECIDX** identificam univocamente um registro de uma tabela.

4.1.1. Tabelas de AID

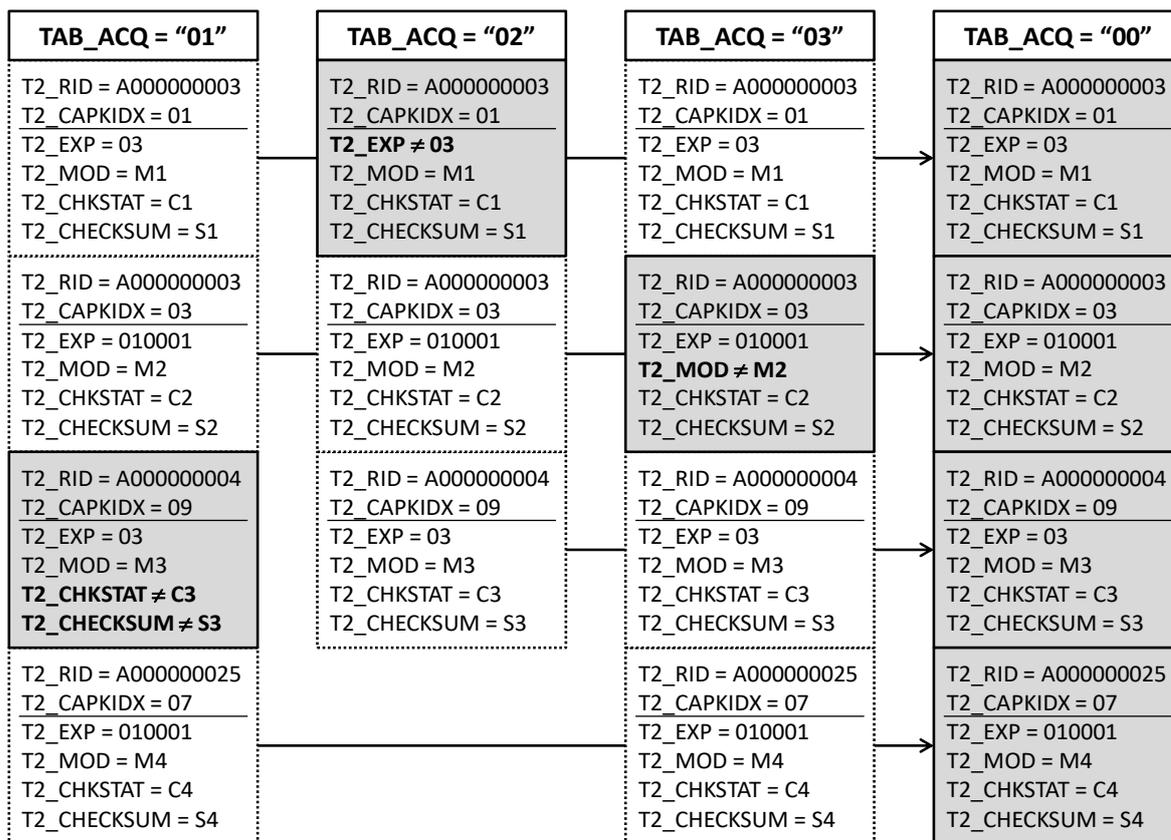
Estas tabelas contêm os identificadores das aplicações EMV suportadas (AIDs) e diversos parâmetros a serem utilizados no processamento, seja para ICC ou CLTS. Os parâmetros que possuem correspondência direta com as normas EMV estão identificados por suas "*tags*".

Cada tabela é composta por um ou mais registros com o *layout* a seguir, tendo como "chave" o AID (*Application Identifier*):

| Id. do Campo | Formato | Tag | Descrição |
|--------------|---------|---------|---|
| TAB_LEN | N3 | | Tamanho do registro, incluindo este campo. O pinpad deverá ser capaz de aceitar registros de: <ul style="list-style-type: none"> ▪ 284 bytes: correspondente à especificação BibComp (campos posteriores a T1_ARCOFFLN não são fornecidos). ▪ 314 bytes: correspondente à especificação v2.0x (campos posteriores a T1_CTLSTACONL não são fornecidos). ▪ 340 bytes: correspondente a esta especificação. ▪ >340 bytes: previsão para especificações futuras (desprezar eventuais dados extras recebidos). |
| TAB_ID | N1 | | Identificação da Tabela de AID (fixo "1"). |
| TAB_ACQ | N2 | | Identificador da Rede Credenciadora responsável pela tabela (de "01" a "99"). |
| TAB_RECIDX | A2 | | Índice do registro na tabela (de "01" a "ZZ"). |
| T1_AIDLEN | N2 | | Tamanho do AID, <u>em bytes</u> (de "05" a "16"). |
| T1_AID | H32 | | AID - <i>Application Identifier</i> (alinhado à esquerda). |
| T1_APPTYPE | N2 | | Tipo de aplicação, para uso no comando "GCR" ou "GCX" (valores de "01" a "98"). |
| T1_DEFLABEL | S16 | | Etiqueta <i>default</i> da aplicação (obsoleto - não usado a partir da norma EMV 4.3). |
| T1_ICCSTD | N2 | | Padrão da aplicação: fixo "03" = EMV. |
| T1_APPVER1 | H4 | 9F09h | <i>Application Version Number (Terminal)</i> - opção #1 |
| T1_APPVER2 | H4 | 9F09h | <i>Application Version Number (Terminal)</i> - opção #2 |
| T1_APPVER3 | H4 | 9F09h | <i>Application Version Number (Terminal)</i> - opção #3 |
| T1_TRMCNTRY | N3 | 9F1Ah | <i>Terminal Country Code</i> |
| T1_TRNCURR | N3 | 5F2Ah | <i>Transaction Currency Code</i> |
| T1_TRNCRREXP | N1 | 5F36h | <i>Transaction Currency Exponent</i> |
| T1_MERCHID | A15 | 9F16h | <i>Merchant Identifier</i> |
| T1_MCC | N4 | 9F15h | <i>Merchant Category Code</i> |
| T1_TRMID | A8 | 9F1Ch | <i>Terminal Identification</i> |
| T1_TRMCAPAB | H6 | 9F33h | <i>Terminal Capabilities</i> |
| T1_ADDTRMCP | H10 | 9F40h | <i>Additional Terminal Capabilities</i> |
| T1_TRMTYP | N2 | 9F35h | <i>Terminal Type</i> |
| T1_TACDEF | H10 | DF9F0Dh | <i>Terminal Action Code – Default</i> |
| T1_TACDEN | H10 | DF9F0Eh | <i>Terminal Action Code – Denial</i> |

| Id. do Campo | Formato | Tag | Descrição |
|--------------------------|------------------|------------------|---|
| T1_TACONL | H10 | DF9F0Fh | <i>Terminal Action Code – Online</i> |
| T1_FLRLIMIT | H8 | 9F1Bh | <i>Terminal Floor Limit</i> (valor “default” para uso em “GCR” ou “GCX”), em centavos, expresso na moeda definida em T1_TRNCURR. |
| T1_TCC | A1 | 9F53h | <i>Transaction Category Code</i> |
| T1_CTLSZEROAM | A1 | | Indica a ação para CTLS se o valor da transação estiver zerado: “1” = Suporta, porém somente <i>online</i> ; “0” ou outro valor = Não suporta. |
| T1_CTLSMODE | A1 | | Capacidade de tratamento do terminal para o referido AID, caso este seja localizado em um CTLS: “1” ou “2” = Suporta VISA qVSDC; “3” ou “4” = Suporta Mastercard PayPass M/Chip; “5” ou “6” = Suporta Amex Expresspay EMV Mode; “7” = Não suporta (reservado); Suporta Pure Contactless; “8” ou “9” = Suporta Discover D-PAS EMV Mode; “A” = Suporta JCB Contactless (uso futuro) “B” = Suporta Union Pay Quick Pass (uso futuro); “C” = Suporta Interac Flash (uso futuro); e “D” = Suporta kernel EMVCo C-8 (uso futuro). “0” ou outro valor = Não suporta. |
| T1_CTLSTRNLIM | H8 | DF8124h | <i>Terminal/Reader Contactless Transaction Limit</i> , em centavos, expresso na moeda definida em T1_TRNCURR. |
| T1_CTLSFLRLIM | H8 | DF8123h | <i>Terminal/Reader Contactless Floor Limit</i> , em centavos, expresso na moeda definida em T1_TRNCURR. |
| T1_CTLSCVMLIM | H8 | DF8126h | <i>Terminal/Reader CVM Required Limit</i> , em centavos, expresso na moeda definida em T1_TRNCURR. |
| T1_CTLSAPPVER | H4 | 9F6Dh | PayPass Mag Stripe Application Version Number (Terminal). |
| T1_RUF1 | N1 N5 | | RUF (fixo “0000”). |
| T1_TDOLDEF | H40 | | <i>Default Transaction Certificate Data Object List (TDOL)</i> (completado com bytes “00” à direita) |
| T1_DDOLDEF | H40 | | <i>Default Dynamic Data Authentication Data Object List (DDOL)</i> (completado com bytes “00” à direita) |
| T1_ARCOFFLN | A8 | | <i>Authorization Response Codes</i> para transações <i>offline</i> . Este campo <u>é desprezado pelo pinpad</u> , dado que estes códigos foram fixados a partir da norma EMV 4.0, deixando de ser parâmetros. Apenas por convenção, manter “Y1Z1Y3Z3”. |

O diagrama a seguir ilustra esse processo:



- ⚠ As tabelas aglutinadas somente são utilizadas pelos comandos Abecs de processamento de cartão (descritos na **seção 3.7**), não sendo reconhecidas pelos comandos obsoletos.
- ⚠ Este processo de aglutinação somente faz sentido quando o SPE utiliza um gerenciamento “unificado” de tabelas (ver **seção 4.2.1**), dado que registros com TAB_ACQ = “00” não podem ser carregados no pinpad quando o gerenciamento é “apartado” (ver **seção 4.2.2**).

4.1.3. Tabelas de Certificados Revogados

Estas tabelas contêm os números de série dos certificados revogados de chave pública de emissor. Cada tabela é composta por um ou mais registros com o *layout* a seguir, tendo como “chave” o RID, o CAPK Index e o Certificate Serial Number.

| Id. do Campo | Formato | Tag | Descrição |
|--------------|---------|-----|---|
| TAB_LEN | N3 | | Tamanho do registro, incluindo este campo (fixo “026”). |
| TAB_ID | N1 | | Identificação da Tabela de Certificados Revogados (fixo “3”). |
| TAB_ACQ | N2 | | Identificador da Rede Credenciadora responsável pela tabela (de “01” a “99”). |
| TAB_RECIDX | A2 | | Índice do registro na tabela (de “01” a “ZZ”). |

| Id. do Campo | Formato | Tag | Descrição |
|--------------|---------|-------|---|
| T3_RID | H10 | | RID - <i>Registered Application Provider Identifier</i> |
| T3_CAPKIDX | H2 | 9F22h | <i>Certification Authority Public Key Index</i> |
| T3_CERTSN | H6 | | <i>Certificate Serial Number</i> |

➤ Exemplos

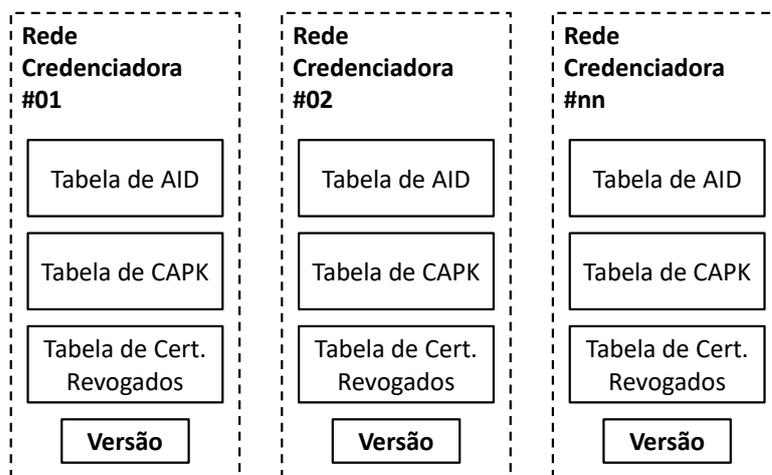
TAB_ACQ = "02", TAB_RECIDX = 3233h ("23"): Certificado Mastercard de número de série 333333h.

02620223A000000004FE333333

4.2. Versão de Tabelas

As Tabelas EMV possuem informação de versão para que o SPE possa controlar a necessidade (ou não) de atualizá-las no pinpad.

Cada conjunto de tabelas de uma Rede Credenciadora possui uma versão distinta, conforme apresentado no diagrama:



Esta informação de versão consiste em um campo de 10 caracteres que pode ser obtido utilizando-se os comandos "**GTS**" ou "**GIX**" (com **PP_TABVERnn**).

Dependendo da filosofia do SPE, este pode operar de duas formas:

- Gerenciar as tabelas de todas as Redes Credenciadoras de forma unificada; ou
- Gerenciar as tabelas das Redes Credenciadoras de forma independente.

4.2.1. Gerenciamento unificado

Quando o SPE não pré-seleciona a Rede Credenciadora antes de efetuar uma transação, recomenda-se um gerenciamento unificado das tabelas, através das seguintes regras:

- As tabelas de todas as Redes Credenciadoras são carregadas em um único momento, sendo utilizado TLI_ACQIDX = "00" no comando "TLI".
- A versão TLI_TABVER informada no comando "TLI" passa a valer para as tabelas de todas as Redes Credenciadoras.
- O comando "GCR" deve ser acionado com GCR_ACQIDXREQ = "00", sendo que GCR_TABVER se refere à versão comum de todas as tabelas.
- O comando "GCX" deve ser acionado sem o parâmetro SPE_ACQREF.

4.2.2. Gerenciamento apartado

Quando o SPE pré-seleciona a Rede Credenciadora antes de efetuar uma transação, recomenda-se um gerenciamento apartado das tabelas, através das seguintes regras:

- As tabelas de cada Rede Credenciadora podem ser carregadas em momentos distintos, sendo utilizado TLI_ACQIDX ≠ "00" no comando "TLI". Neste caso, somente as tabelas da rede em questão são alteradas, sendo as outras preservadas.
- A versão TLI_TABVER informada no comando "TLI" passa a valer somente para as tabelas da Rede Credenciadora em questão. A partir deste momento, o comando "GTS" retornará a versão "0000000000" se acionado com GTS_ACQIDX = "00".
- O comando "GCR" deve ser acionado com GCR_ACQIDXREQ ≠ "00", sendo que GCR_TABVER se refere somente à versão das tabelas da Rede Credenciadora desejada.
- O comando "GCX" deve ser acionado com o parâmetro SPE_ACQREF ≠ "00".

5. Segurança

Este capítulo detalha os mecanismos de segurança criptográfica utilizados por esta especificação, fornecendo explicações quanto às chaves injetadas pelo fabricante do pinpad, bem como os processos destinados a assegurar o sigilo das informações trafegadas na comunicação com o SPE.

5.1. Mapeamento de chaves

Os pinpads possuem em sua memória, em uma área protegida, diversas chaves de criptografia “injetadas” pelo fabricante, considerando-se dois algoritmos diferentes:

- **MK/WK TDES;** e
- **DUKPT TDES.**

Estas chaves são utilizadas pelos comandos desta especificação para criptografia do PIN digitado pelo portador e para outros dados (“DAT”), sendo referenciadas por um índice de dois dígitos numéricos.

Desta forma, esta especificação considera o seguinte mapeamento de chaves, diferenciando quatro tipos para cada índice numérico existente:

| Índice ↓ | MK:TDES | | DUKPT:TDES | |
|-------------|---------|-----|------------|-----|
| | PIN | DAT | PIN | DAT |
| “00” | | | | |
| “01” | | | | |
| “02” | | | | |
| ... | | | | |
| “62” | | | | |
| “63” | | | | |

256 chaves

➤ Considerações importantes:

- Do ponto de vista de injeção em fábrica, as chaves de PIN e dados (“DAT”) não possuem nenhum tratamento especial. Trata-se apenas de uma separação lógica para acatar as restrições do PCI (uma chave usada para criptografia de PIN não pode ser usada para outros propósitos).
- Os seguintes comandos usam única e exclusivamente chaves de PIN: **“GDU”, “GPN”, “GOC” e “GOX”**.
- Os seguintes comandos usam única e exclusivamente chaves de dados (“DAT”): **“DWK”, “EBX”, “ENB” e “GTK”**.
- O índice “00” é válido e, considerando-se que o máximo índice permitido é “99”, pode-se ter até 100 chaves de cada tipo. Entretanto, a quantidade de chaves possíveis para cada tipo depende do modelo de pinpad, respeitando-se o mínimo definido na seção 6.1.2 (por exemplo, um determinado modelo de pinpad pode permitir até 77 chaves DUKPT:TDES, de índices “00” a “76”).
- As chaves DUKPT:TDES de dados (“DAT”) permitem diferentes variantes no momento do uso (ver **seção 5.1.1**), entretanto a existência destas variantes não requer nenhum tratamento especial no processo de injeção em fábrica.

5.1.1. Criptografia DUKPT:TDES

A criptografia **DUKPT:TDES** é definida pela norma **ANSI X9.24:2009**, que contempla cinco variantes para modificação da chave utilizada. Esta especificação considera somente algumas destas variantes, conforme tabela a seguir:

| Descrição na norma ANSI X9.24:2009 | Constante utilizada para modificação da chave | Referência nesta especificação |
|---|--|--------------------------------|
| <i>PIN Encryption</i> | 00 00 00 00 00 00 00 FF 00 00 00 00 00 00 00 FF | DUKPT:TDES:PIN |
| <i>Message Authentication, request or both ways</i> | 00 00 00 00 00 00 FF 00 00 00 00 00 00 00 FF 00 | Não usada. |
| <i>Data Encryption, request or both ways (*)</i> | 00 00 00 00 00 FF 00 00 00 00 00 00 00 FF 00 00 | DUKPT:TDES:DAT#3 |
| <i>Message Authentication, response</i> | 00 00 00 00 FF 00 00 00 00 00 00 00 FF 00 00 00 | Não usada. |
| <i>Data Encryption, response (*)</i> | 00 00 00 FF 00 00 00 00 00 00 00 FF 00 00 00 00 | Não usada. |

(*) Além da constante para modificação, estas duas variantes acrescentam uma diversificação adicional da chave utilizando **TDES**, conforme descrito na **seção A.4.1** da norma **ANSI X9.24:2009**.

⚠ Sempre que esta especificação considerar criptografia de bloco de dados usando DUKPT, independentemente da modalidade (ECB ou CBC) ou da variante utilizada, o pinpad deverá utilizar a mesma "Current Transaction Key" (um único KSN) para todos os pedaços de 8 bytes do bloco, não importando a quantidade de iterações necessárias para o processo.

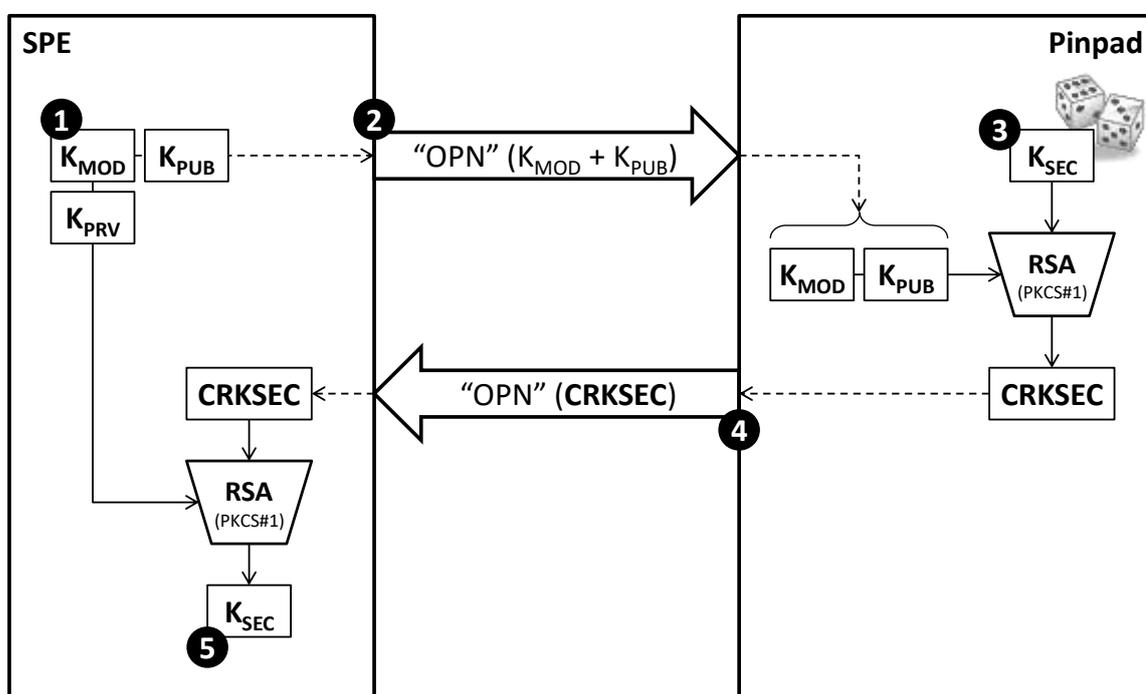
5.2. Comunicação Segura

Esta especificação prevê uma modalidade denominada “**Comunicação Segura**” em que dados trafegados pela interface serial entre o SPE e o pinpad são criptografados utilizando-se algoritmo **AES** através de uma chave “**K_{SEC}**”.

Este método destina-se a dificultar a monitoração da interface serial, minimizando o risco de fraudes.

5.2.1. Estabelecimento

O fluxo a seguir ilustra o processo de estabelecimento da “Comunicação Segura”:



- ❶ O SPE cria uma chave RSA (ou utiliza uma chave fixa "hardcoded"). Esta especificação considera que o módulo deve ter 256 bytes (podendo ser aumentado no futuro).
- ❷ O SPE envia K_{MOD} e K_{PUB} para o pinpad através do comando "OPN".
- ❸ O pinpad gera aleatoriamente uma chave K_{SEC} (16 bytes) e a criptografa utilizando algoritmo RSA e chave K_{MOD}/K_{PUB} . Para isso, utiliza-se como entrada do algoritmo o formato de bloco recomendado pela norma PKCS #1 (tabela a seguir), que deve possuir o mesmo tamanho de K_{MOD} .
- ❹ O pinpad devolve o criptograma (CRKSEC) gerado na resposta do comando "OPN".
- ❺ O SPE decodifica o criptograma (CRKSEC) recebido utilizando algoritmo RSA e chave K_{MOD}/K_{PRV} , obtendo assim a chave K_{SEC} aleatória gerada pelo pinpad.

Formato do bloco PKCS #1:

| Formato | Descrição |
|---------|---|
| B2 | Cabeçalho (fixo: 00h 02h). |
| Bxxx | Bytes aleatórios <u>diferentes de 00h</u> . O tamanho “xxx” deve ser calculado de forma que o tamanho total desta estrutura seja o mesmo de K_{MOD} . |
| B1 | Separador (fixo: 00h). |
| B16 | Chave aleatória gerada pelo pinpad (K_{SEC}). |

➡ Exemplo:

Um exemplo detalhado do processo de estabelecimento da “Comunicação Segura” encontra-se na seção 3.2.2.

5.2.2. Troca de pacotes

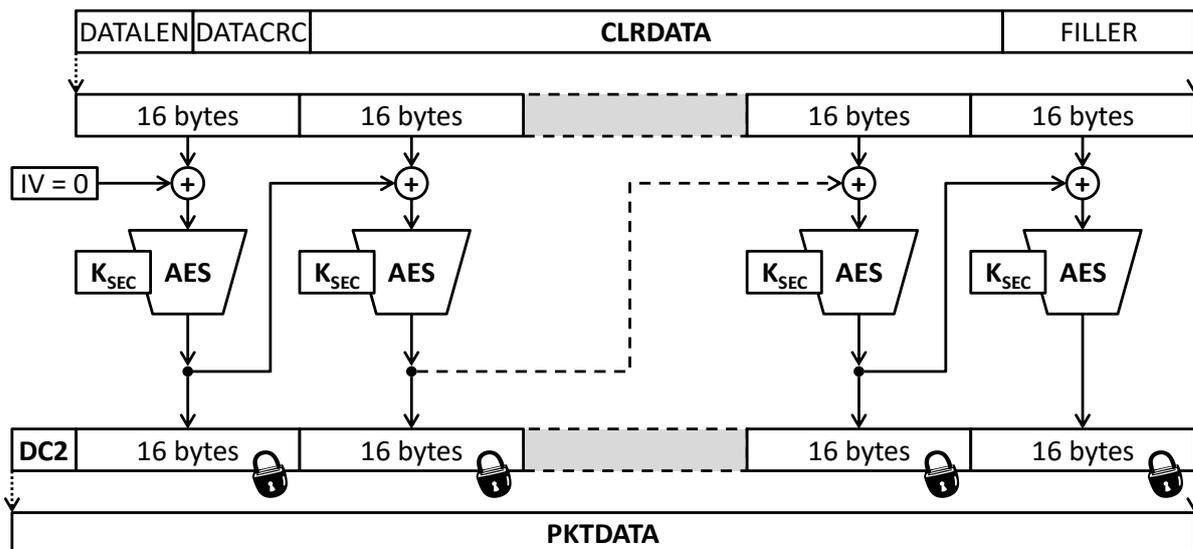
Estabelecida a “Comunicação Segura”, o SPE e o pinpad passam a ter a capacidade de trocar pacotes criptografados através da interface serial. Para isso, os dados dos comandos e respostas devem ser encapsulados no seguinte formato:

| Nome | Formato | Descrição |
|-----------------|---------|---|
| DATALEN | X2 | Tamanho do campo CLRDATA (até 2044 bytes). |
| DATA CRC | X2 | CRC-16 dos dados contidos no campo CLRDATA . |
| CLRDATA | ??? | Dados do comando ou resposta. |
| FILLER | B..15 | Preenchimento com bytes 00h de forma que o tamanho total desta estrutura seja <u>múltiplo de 16</u> . |

- ⚠ Depois de estabelecida a “Comunicação segura”, o SPE somente deverá enviar comandos criptografados (excetuando-se o “**OPN**”). Caso o pinpad receba um comando “em claro” nesta situação, ele retornará ↵ST_ERRPKTSEC para o comando em questão. A resposta de erro será devolvida “em claro”, porém a “Comunicação segura” é mantida ativa.
- ⚠ Depois de estabelecida a “Comunicação Segura”, o pinpad sempre devolverá respostas criptografadas, incluindo as mensagens de notificação (“**NTM**”), excetuando-se a resposta para os comandos “**CLQ**” e “**CLX**”, que sempre é devolvida “em claro”.
- ⚠ Independentemente do estado da “Comunicação Segura”, o comando “**OPN**” (**seguro ou clássico**), somente pode ser enviado “em claro”.

5.2.2.1. Envio de pacotes criptografados

Independentemente do sentido (SPE ↔ pinpad), os dados de comando/resposta (**CLRDATA**) devem ser embutidos no *layout* descrito anteriormente e criptografados usando-se o algoritmo **AES** em modo CBC através da chave K_{SEC} , conforme diagrama:



Conforme descrito no Nível de Enlace (seção 2.2.1), se **PKTDATA** está criptografado, ele deve ser iniciado pelo byte «DC2».

➡ Exemplo:

Considerando-se $K_{SEC} = DB3B4D015432AB3223555A1F81759A94$, o SPE deseja enviar o comando “GIX” abaixo em “Comunicação Segura”:

| | | |
|----------------|--|-----------------------------|
| CLRDATA | 47 49 58 30 31 34 00 01 00 0A 80 01 80 04 80 34 91 01 91 0E | GIX014....€.€.€4 , , , , |
|----------------|--|-----------------------------|

Incluindo-se os campos de controle (**DATALEN**, **DATACRC** e **FILLER**), o bloco a ser criptografado fica:

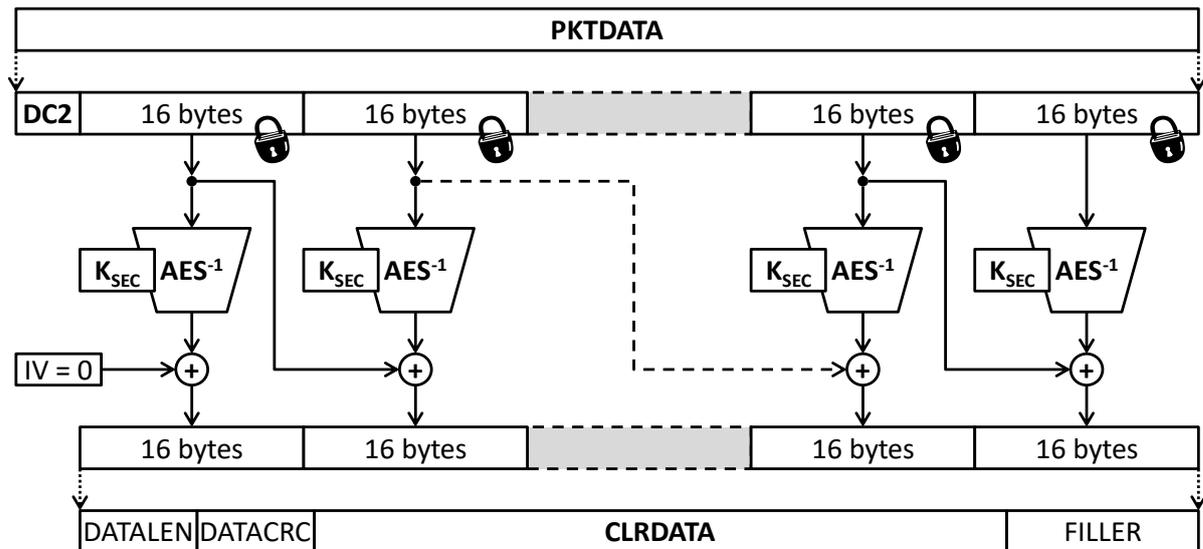
| | | |
|----------------|---|------------------|
| DATALEN | 00 14 8D F2 47 49 58 30 31 34 00 01 00 0A 80 01 | ...òGIX014....€. |
| DATACRC | 80 04 80 34 91 01 91 0E 00 00 00 00 00 00 00 00 | €.€4‘.’..... |
| CLRDATA | | |
| FILLER | | |

Aplicando-se o AES (CBC) com a chave K_{SEC} , obtém-se o seguinte resultado (precedido pelo byte «DC2»):

| | | |
|----------------|--|--|
| PKTDATA | 12 EA 22 9E DD 36 F8 4C 2A A7 E0 02 75 10 5C 3A 8A 78 7F C9 B2 88 35 40 AE E8 27 BA 1C 5A 03 94 96 | .ê"žY6øL*şà.u.\: Šx.É²~5@è'o.Z." - |
|----------------|--|--|

5.2.2.2. Recepção de pacotes criptografados

Ao receber um pacote criptografado (detectado pela presença do byte «DC2» no início de **PKTDATA**), o SPE ou o pinpad deve descriptografá-lo usando-se o algoritmo **AES**⁻¹ em modo CBC através da chave K_{SEC} , conforme diagrama:



Ao receber um pacote criptografado, as seguintes conferências devem ser efetuadas:

- O tamanho de **PKTDATA** (excluindo-se o byte «**DC2**») deve ser múltiplo de 16;
- O valor de **DATALEN** deve ser coerente (menor do que o tamanho de **PKTDATA**, excluindo-se os 5 bytes de **DATALEN**, **DATACRC** e «**DC2**»); e
- O CRC-16 calculado sobre **CLRDATA** deve ser igual ao valor informado em **DATACRC**.

⚠ Se o SPE detectar alguma destas inconsistências em uma resposta criptografada, ele deve finalizar a operação com erro fatal.

➡ Exemplo:

O SPE recebe a seguinte resposta a um comando iniciada pelo byte «**DC2**», indicando “Comunicação Segura”.

| | | |
|----------------|---|---|
| PKTDATA | <pre> 12 BA 90 C3 82 65 12 69 B2 2D 0E FC 90 B9 2B C3 08 83 71 38 6A 69 B9 A7 A8 5B C6 AC 76 E4 84 37 BC 73 A2 02 86 EC B6 73 A4 93 4C 85 35 4E 47 16 0F 27 2E 1A 2B 53 BA C1 B7 95 85 9E 4C 62 2F C8 66 1A 4B AE 1F EE 45 09 75 B7 CA 04 20 C6 18 A1 FC 74 47 65 C3 E7 08 AF 56 02 25 6B 75 A9 07 C3 F9 A2 56 89 CB 11 23 9C 01 E3 6F C6 18 B4 17 A0 2A 21 77 E3 C3 C8 73 B1 F0 6E 3B D6 20 8F F2 B4 96 A2 B0 BD F8 12 32 FD A0 97 30 0C 7D 19 B0 07 DD C1 7E 6D EF 8B E7 BB 0E 82 58 8C 07 11 C0 1B 39 B1 21 BB 8C 66 E3 E0 31 3C 82 69 27 FB 7F 13 36 </pre> | <pre> .º.Ã,e.i²..ü•¹+Ã .fq8ji¹§´[Æ-vä,,7 ¼s¢.†i¶ s×“L...5NG. .'...+SºÁ•...ŽLb/Ē f.K®.îE.u.Ê.•Æ.j ütGeÃç.~V.%ku©.Ã ù¢V%È.#æ.ãøÆ. . *!wãÃËs±ðñ;Ö•ò´ •¢°½ø.2ý•0.}.º. YÁ~mi<ç>.,XÆ..À. 9±!»æfãà1<,i'û•. 6 </pre> |
|----------------|---|---|

O SPE decriptografa a mensagem (sem o «**DC2**») usado AES (CBC) com a chave **K_{SEC}** = DB3B4D015432AB3223555A1F81759A94, obtendo:

| | | |
|----------------|---|-------------------|
| | 00 A0 66 EB 47 49 58 30 30 30 31 35 31 80 01 00 | . fëGIX000151€.. |
| | 0C 39 39 31 32 37 34 33 36 36 31 35 35 80 04 00 | .991274366155€.. |
| | 0D 48 45 4D 49 53 50 48 45 52 45 53 20 20 80 34 | .HEMISPHERES••€4 |
| DATALEN | 00 64 30 31 31 31 30 30 31 31 30 30 30 30 30 | .d01110011000000 |
| DATACRC | 30 30 30 30 30 30 30 30 30 30 30 32 32 32 32 | 000000000022222 |
| CLRDATA | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| FILLER | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 32 32 32 32 91 01 00 0A FF FF F9 13 25 00 | 222222'...ÿÿù.%.C |
| | 43 20 04 43 00 00 00 00 00 00 00 00 00 00 00 | .C..... |

O SPE identifica o tamanho **DATALEN** = 00A0h (160 bytes) e valida o **DATACRC** = 66EBh, extraindo o bloco **CLRDATA** (resposta ao comando “**GIX**”).

| | | |
|----------------|---|------------------|
| | 47 49 58 30 30 30 31 35 31 80 01 00 0C 39 39 31 | GIX000151€...991 |
| | 32 37 34 33 36 36 31 35 35 80 04 00 0D 48 45 4D | 274366155€...HEM |
| | 49 53 50 48 45 52 45 53 20 20 80 34 00 64 30 31 | ISPHERES••€4.d01 |
| | 31 31 30 30 31 31 30 30 30 30 30 30 30 30 30 | 1100110000000000 |
| CLRDATA | 30 30 30 30 30 30 30 30 32 32 32 32 32 32 32 | 000000022222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 32 32 32 32 32 32 32 32 32 32 32 32 32 | 222222222222222 |
| | 32 32 91 01 00 0A FF FF F9 13 25 00 43 20 04 43 | 22'...ÿÿù.%.C .C |

5.2.2.3. Finalização

O processo de “Comunicação Segura” é finalizado e a chave **K_{SEC}** limpa da memória nos seguintes casos:

- Um comando “**CLO**”/**CLX**” é recebido.
- O pinpad detecta alguma inconsistência no comando criptografado, retornando “**ERR009**” (↳ST_ERRPKTSEC, conforme descrito na **seção 2.3.4**).
- O pinpad recebe um comando “**OPN**” criptografado.

5.3. PAN Criptografado

Para evitar que dados sensíveis (como o número do cartão - PAN) trafeguem livremente pela porta serial do pinpad, esta especificação implementa uma modalidade de trabalho denominada “PAN Criptografado”.

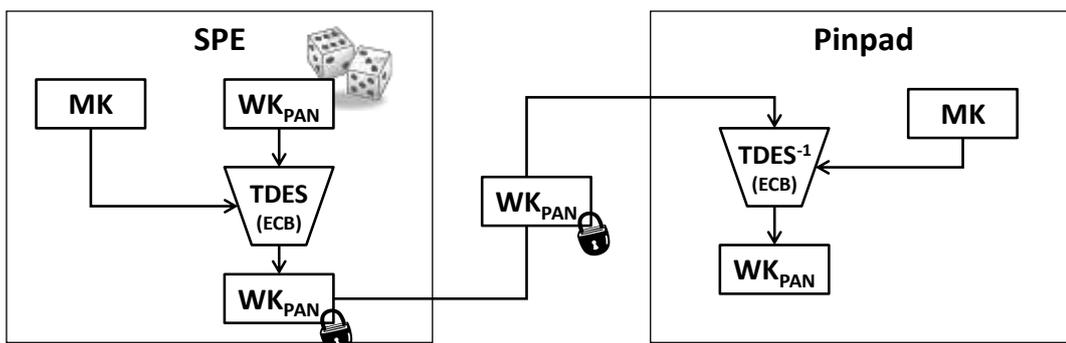
⚠ Esta modalidade é **obsoleta**, tendo sido substituída pela “Comunicação Segura” descrita na **seção 5.2**. O SPE deve usar esta modalidade somente se o pinpad não for reconhecido como um “Pinpad Abecs”.

Nesta modalidade, alguns dados transitam criptografados por uma chave ~~DES ou~~ TDES denominada **WK_{PAN}**.

Esta especificação prevê que a chave **WK_{PAN}** possa ser gerada de duas formas:

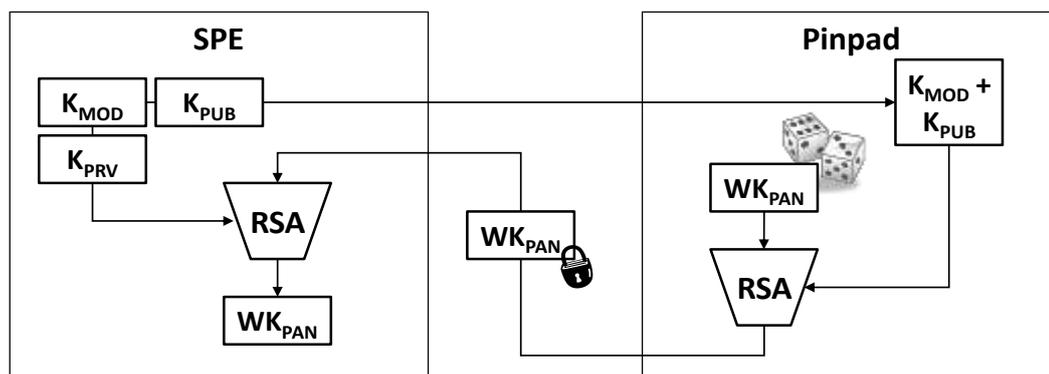
➡ Modalidade 1:

Se uma Master Key (TDES) do pinpad for conhecida, a **WK_{PAN}** poderá ser gerada externamente pelo SPE e fornecida ao pinpad criptografada por esta Master Key. ~~O tamanho da chave WK_{PAN} deverá respeitar o tamanho da Master Key conhecida (16 bytes para TDES).~~



➡ Modalidade 2:

Uma chave **WK_{PAN}** aleatória, ~~sempre TDES~~, poderá ser gerada internamente no pinpad e fornecida ao SPE através de um criptograma RSA, conforme descrito na **seção 5.3.3**.



A ativação da modalidade “PAN Criptografado”, assim como a definição da chave WK_{PAN} , é feita através do comando “DWK” (ver [seção 3.2.5](#)). A tabela a seguir lista os comandos e os dados afetados:

| Comando | Dados afetados | Algoritmo | Observação |
|----------------|---|--------------------|---|
| “ <u>CKE</u> ” | <u>CKE_TRK1</u> <u>CKE_TRK2</u> <u>CKE_TRK3</u> | TDES | O pinpad devolve o PAN das trilhas criptografado (<u>ou não</u>) de acordo com o definido na seção 5.3.1 . |
| “ <u>GCR</u> ” | <u>GCR_PAN</u> <u>GCR_TRK1</u> <u>GCR_TRK2</u> <u>GCR_TRK3</u> | TDES | O pinpad devolve o PAN (e o PAN das trilhas) criptografado (<u>ou não</u>) de acordo com o definido na seção 5.3.1 . |
| “ <u>ENB</u> ” | <u>ENB_INPUT</u> | TDES ⁻¹ | O SPE <u>sempre</u> criptografa o dado de entrada para enviá-lo ao pinpad (apesar de não ser necessariamente um PAN, o dado de entrada deste comando normalmente representa uma informação sensível). |
| “ <u>GPN</u> ” | <u>GPN_PAN</u> | TDES ⁻¹ | O PAN fornecido ao pinpad deve obrigatoriamente ser criptografado se tiver 16 ou mais dígitos. O PAN pode (<u>ou não</u>) ser criptografado pelo SPE se tiver menos de 16 dígitos. |

5.3.1. Codificação do PAN

A codificação do número do cartão deverá respeitar as seguintes regras:

- Somente os 16 dígitos menos significativos do PAN são criptografados, considerando-se que eles perfazem um bloco de 8 bytes em codificação BCD. Dado que os parâmetros dos comandos são em ASCII, os dígitos numéricos decimais do PAN podem ser substituídos diretamente pelos dígitos hexadecimais do criptograma gerado.
- Espaços em branco no meio do número do cartão (tipicamente na trilha 1 de alguns emissores) deverão ser convertidos para dígitos ‘E’ (em hexadecimal).
- A seguinte regra deve ser utilizada para identificação do PAN dentro das trilhas (seja 1, 2 ou 3):
 - ⇒ Da esquerda para a direita, localizar o primeiro caractere numérico (‘0’ a ‘9’) ou espaço em branco. Ele marca o início do PAN.
 - ⇒ Continuar percorrendo a trilha para localizar o caractere separador (“^” ou “=”) ou até atingir seu final.
- O PAN obtido não será criptografado nos seguintes casos:
 - ⇒ Se tiver menos de 13 dígitos.
 - ⇒ Se contiver algum caractere não numérico (‘0’ a ‘9’) e diferente de espaço em branco.
- Caso o PAN tenha menos de 16 dígitos, ele será acrescido de dígitos ‘F’ (em hexadecimal) à direita, até completar esse tamanho.

- A informação de tamanho do PAN ou trilha contida nos parâmetros de entrada e saída dos comandos deve respeitar o tamanho da informação enviada, incluindo a criptografia. A entidade que receber o dado criptografado, seja o SPE ou o pinpad, deverá eliminar eventuais 'F's ao final do PAN depois de decodificado e recalculer seu tamanho real.

⚠ Estas regras não se aplicam ao comando "**ENB**" mesmo que **ENB_INPUT** contenha a informação de um PAN, uma vez que ele se destina a criptografar dados genéricos que não são interpretados de nenhuma forma pelo pinpad.

➡ Exemplos

Os exemplos a seguir consideram uma WK_{PAN} tipo TDES de valor 'EA 52 8A 43 B0 26 52 FD EB 53 8B 42 B1 27 53 FC':

Exemplo 1: Trilha 1 devolvida pelo pinpad, com PAN contendo espaços em branco.

- Aberta (59 caracteres):
"B3764 361234 56006^NOME NOME NOME NOME NOME N^0905060640431"
- PAN Identificado (17 caracteres):
"3764 361234 56006"
- Codificação:
"764E361234E56006" ⇒ TDES ⇒ "5716A983F0E4643B"
- Criptografada (59 caracteres):
"B35716A983F0E4643B^NOME NOME NOME NOME NOME N^0905060640431"

Exemplo 2: PAN de 19 dígitos enviado pelo SPE ao pinpad.

- Aberto (19 caracteres):
"6234987432874320001"
- Codificação:
"4987432874320001" ⇒ TDES⁻¹ ⇒ "407E5D4F32598B98"
- Criptografado (19 caracteres):
"623407E5D4F32598B98"

Exemplo 3: Trilha 1 devolvida pelo pinpad, com PAN de 13 dígitos.

- Aberta (55 caracteres):
"B3764361234006^NOME NOME NOME NOME NOME N^0905060640431"
- PAN Identificado (13 caracteres):
"3764361234006"
- Codificação:
"3764361234006FFF" ⇒ TDES ⇒ "A4F4729D58CAA7DA"

- Criptografada (58 caracteres):

"BA4F4729D58CAA7DA"^NOME NOME NOME NOME NOME N^0905060640431"

Exemplo 4: PAN de 15 dígitos enviado pelo SPE ao pinpad.

- Aberto (15 caracteres):

"376436123456006"

- Codificação:

"376436123456006"F" ⇒ TDES⁻¹ ⇒ "431E6D386E688B0B"

- Criptografado (16 caracteres):

"431E6D386E688B0B"

Exemplo 5: Trilha 2 devolvida pelo pinpad, com PAN de 16 dígitos.

- Aberta (37 caracteres):

"6002938264523821=09050606404312376450"

- PAN Identificado (16 caracteres):

"6002938264523821"

- Codificação:

"6002938264523821" ⇒ TDES ⇒ "BC27B145C5DE8BEB"

- Criptografada (37 caracteres):

"BC27B145C5DE8BEB"=09050606404312376450"

Exemplo 6: Trilha 2 de 37 caracteres devolvida pelo pinpad, porém com PAN de 13 dígitos, resultando em 40 caracteres depois de criptografada.

- Aberta (37 caracteres):

"3827418937101=09050606404312376450123"

- PAN Identificado (13 caracteres):

"3827418937101"

- Codificação:

"3827418937101"FFF" ⇒ TDES ⇒ "1CCE9197C5C6E3FF"

- Criptografada (40 caracteres!!!):

"1CCE9197C5C6E3FF"=09050606404312376450123"

Exemplo 7: Trilha 3 devolvida pelo pinpad, com PAN de 19 dígitos.

- Aberta (104 caracteres):

"4916748362525378000==5300053205322056019300000010000004050=00000000000000000000000000000000=7=3012056"

- PAN Identificado (19 caracteres):
"4916748362525378000"
- Codificação:
"6748362525378000" ⇒ TDES ⇒ "FE8E271A114C1A35"
- Criptografada (104 caracteres):
"491FE8E271A114C1A35==5300053205322056019300000010000004050=00000000000000=000000000000000000=7=3012056"

Exemplo 8: Trilha 2 devolvida pelo pinpad, sem separador. Neste caso, para manter coerência com a regra definida, é como se a trilha inteira fosse o PAN.

- Aberta (37 caracteres):
"9823746589273648956239486587923497851"
- PAN Identificado (37 caracteres):
"9823746589273648956239486587923497851"
- Codificação:
"9486587923497851" ⇒ TDES ⇒ "2C05DF894573C7FA"
- Criptografada (37 caracteres):
"9823746589273648956232C05DF894573C7FA"

5.3.2. Decodificação das trilhas pelo SPE

Mesmo estando habilitada a modalidade "PAN Criptografado", em algumas situações (como explicado na [seção 5.3.1](#)) uma ou mais trilhas retornadas pelo pinpad podem não sofrer criptografia alguma, no caso em que não tenha sido possível isolar um PAN válido. Entretanto, esta especificação não prevê uma forma de informar ao SPE esta ocorrência, podendo gerar erros quando este tentar decifrar uma trilha recebida.

Esta seção procura definir uma regra padronizada para que o SPE possa identificar se a trilha contém de fato um PAN criptografado:

- Percorrer a trilha da esquerda para a direita até localizar um separador ("^" ou "=") ou até chegar ao seu final. O bloco de caracteres mais à direita deve ser considerado como um PAN criptografado.
 - ⇒ Se o bloco encontrado tiver menos de 16 caracteres, então não houve criptografia.
 - ⇒ Se o bloco encontrado tiver algum caractere fora da faixa hexadecimal ('0' a '9' / 'A' a 'F'), então não houve criptografia.
- Decifrar o bloco usando a chave **WK_{PAN}**. Somente caracteres numéricos ('0' a '9'), espaços em branco (codificados como 'E') ou o preenchimento ao final ('F', 'FF' ou 'FFF') são aceitos nesse resultado. Se ele não apresentar esta coerência, deduz-se que não houve criptografia.

5.3.3. Criptograma RSA

Quando requerida a “Modalidade 2” no comando “**DWK**”, o pinpad retornará um criptograma RSA gerado pela chave pública fornecida que, após “aberto”, possui o seguinte layout de 128 bytes:

| Formato | Descrição |
|---------|--|
| A1 | Cabeçalho do bloco (fixo = “T” / 54h). |
| N1 | Versão do <i>layout</i> (fixo = “1” / 31h). |
| N9 | Número sequencial gerado pelo pinpad para diversificação do criptograma. |
| H32 | Chave WK_{PAN} gerada aleatoriamente pelo pinpad. |
| N84 | Não usado (sequência de zeros = “00000...0000”). |
| A1 | Finalizador do bloco (fixo = “X”) |

Ao abrir o criptograma, o SPE deve verificar se o cabeçalho, a versão e o finalizador estão corretos, validando sua integridade. O número sequencial deve ser desprezado.

5.4. Criptografia “End-to-End”

A Criptografia “End-to-End” é um recurso através do qual o SPE não obtém as trilhas completas dos cartões (a menos que seja absolutamente necessário), trabalhando somente com as informações mínimas necessárias para o processamento local da transação.

Este processo parte dos seguintes princípios:

- Os comandos “**CEX**” e “**GCX**” nunca devolvem as trilhas completas dos cartões;
- O comando “**GTK**” é capaz de devolver as trilhas já criptografadas usando-se um método e uma chave definidos pela Rede Credenciadora, de forma que estas somente possam trafegar de forma segura na mensagem de autorização (opcionalmente pode-se usar uma chave aleatória K_{RAND} gerada pelo pinpad); e
- O comando “**GPN**” não necessita receber o PAN quando este é previamente obtido de um cartão e, portanto, já é conhecido do pinpad (desde que o comando “**GTK**” ainda não tenha sido usado).

5.4.1. Trilhas incompletas e mascaramento

Os campos **PP_TRK1INC**, **PP_TRK2INC** e **PP_TRK3INC** retornados pelo pinpad contêm as trilhas do cartão truncadas de forma que, normalmente, sejam devolvidas somente as seguintes informações necessárias ao processamento local do SPE:

- PAN (número do cartão), podendo ser mascarado de acordo com parâmetro **SPE_PANMASK**;
- Nome do portador (se trilha 1);
- Data de validade do cartão; e
- Código de Serviço (*Service Code*).

Para isso, deve-se adotar a seguinte regra ao montar os campos:

| | |
|-------------------|---|
| PP_TRK1INC | Percorrer a trilha 1 da esquerda para a direita e truncar em sete posições depois do segundo separador “^” (5Eh). Caso esta regra não seja possível, considerar as 19 primeiras posições. |
| PP_TRK2INC | Percorrer a trilha 2 da esquerda para a direita e truncar em sete posições depois do separador “=” (3Dh). Caso esta regra não seja possível, considerar as 19 primeiras posições. |
| PP_TRK3INC | Considerar sempre as 19 primeiras posições. |

Caso o parâmetro **SPE_PANMASK** tenha sido fornecido ao comando, o pinpad deverá efetuar o mascaramento do PAN da seguinte forma:

- Identificar como PAN a primeira sequência consecutiva de caracteres numéricos à esquerda do campo, ignorando eventuais espaços em branco.
- Acatar a definição de **SPE_PANMASK** que indica quantos dígitos numéricos devem ser mantidos abertos à direita (“dd”) e à esquerda (“ee”).

- ⇒ Caso a soma dos tamanhos “dd” e “ee” ultrapasse a quantidade de dígitos numéricos do PAN, não há mascaramento.
- ⇒ Os dígitos numéricos restantes devem ser substituídos por asteriscos (2Ah).
- ⇒ Eventuais espaços em branco no PAN não são considerados nesta contagem.

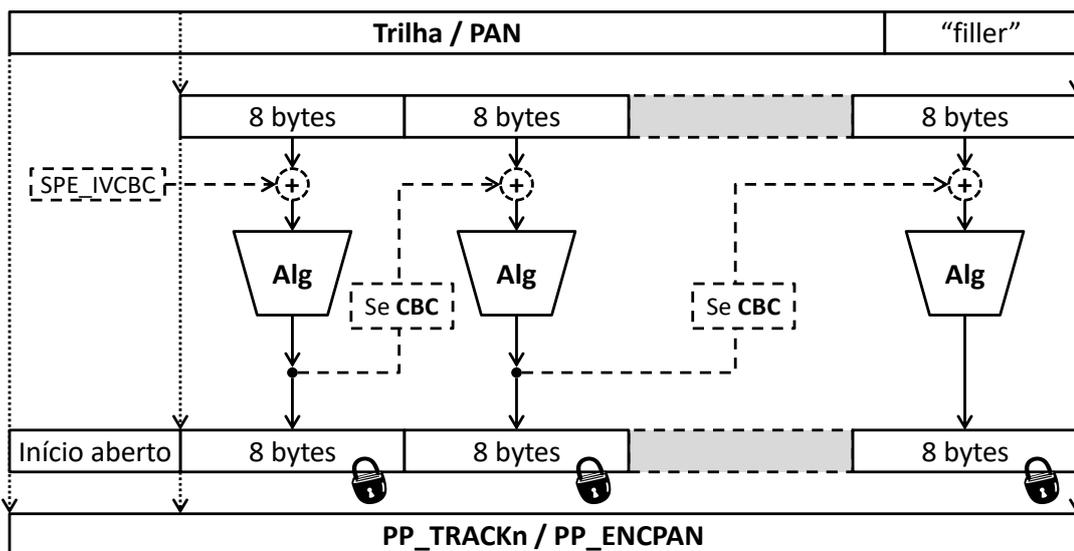
Este mascaramento também afeta o campo PP_PAN de retorno do comando “GCX”.

➔ Exemplos

- Supondo-se uma trilha 2 contendo “66733246732413=1512601234879534275432”, o valor de PP_TRK2INC seria “66733246732413=1512601”.
- Supondo-se uma trilha 1 contendo “B9994444333322221111^NOME^1512601234879”, o valor de PP_TRK1INC seria “B9994444333322221111^NOME^1512601”.
- Supondo-se uma trilha 2 contendo “667332467324131512601234879534275432”, o valor de PP_TRK2INC seria “6673324673241315126”.
- Supondo-se uma trilha 1 contendo “B3764 329710 01006^JOE^2108100265123756” e SPE_PANMASK = “0604”, o valor de PP_TRK1INC seria “B3764 32**** *1006^JOE^2108100”.
- Supondo-se uma trilha 2 contendo “4444333322221111=2212601019923625524” e SPE_PANMASK = “0700”, o valor de PP_TRK2INC seria “4444333*****=2212601”.
- Supondo-se uma trilha 1 contendo “A756325325535^PROPRIETARYFORMAT=6562532” e SPE_PANMASK = “0005”, o valor de PP_TRK1INC seria “A*****25535^PROPR”.

5.4.2. Criptografia de trilhas

Caso o SPE solicite as trilhas criptografadas no comando “GTK”, deve-se codificá-las de acordo com o apresentado no diagrama a seguir:



O algoritmo a ser usado na criptografia (“Alg”) é estipulado em SPE_MTHDDAT, usando a chave SPE_KEYIDX. Entretanto, quando SPE_MTHDDAT = “9x”, deve-se adotar a seguinte regra:

- A criptografia será feita usando **TDES** com uma chave aleatória **K_{RAND}** gerada pelo próprio pinpad. Esta chave deve ser gerada a cada execução do comando **"GTK"** e não pode ser reutilizada.
- O SPE deve fornecer uma chave pública RSA nos campos de entrada **SPE_PBKMOD** e **SPE_PBKEXP**.
- A chave **K_{RAND}** é criptografada pelo pinpad usando-se a chave pública RSA, no mesmo formato PKCS #1 apresentado na **seção 5.2.1**, gerando o campo de saída **PP_ENCKRAND**.

5.4.2.1. Trilha 1

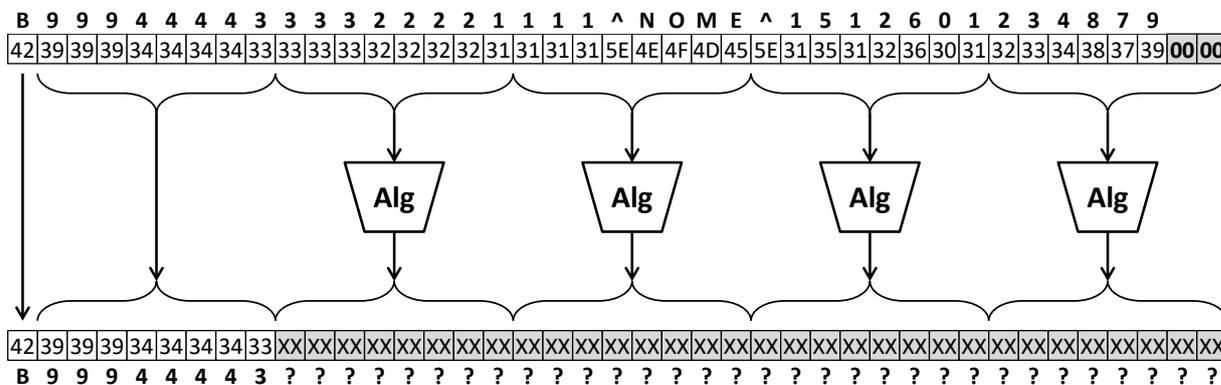
A trilha 1 permite caracteres alfanuméricos, então ela é sempre tratada como uma informação em codificação ASCII, sendo que cada símbolo ocupa um byte.

Desta forma, adota-se a seguinte regra:

- O pinpad preserva "em claro" os caracteres iniciais da trilha 1, de acordo com a quantidade solicitada em **SPE_OPNDIG**, desconsiderando-se o caractere de formato (normalmente "B"), que é sempre devolvido em claro.
- O bloco a ser criptografado deve ter tamanho múltiplo de 8 (oito) bytes. Caso necessário, ele deve ser preenchido com bytes **00h** ao final ("*filler*").

➡ Exemplo

O diagrama a seguir ilustra a criptografia de uma trilha de 39 caracteres ("**B9994444333322221111^NOME^1512601234879**") usando-se criptografia de bloco tipo **ECB**, em que se deseja preservar "em claro" os 8 primeiros caracteres:



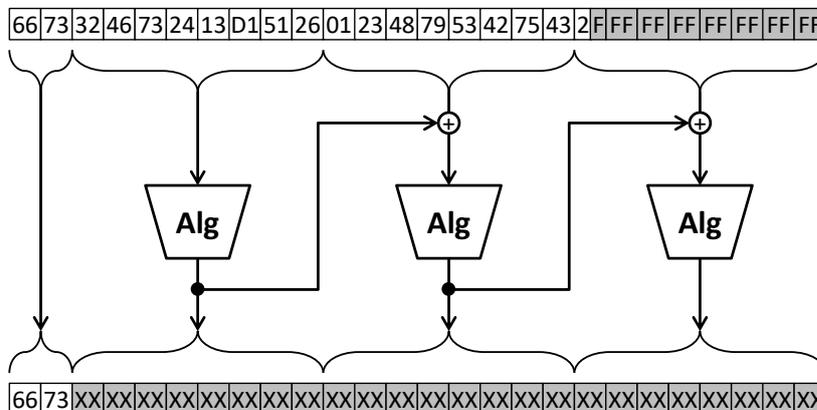
5.4.2.2. PAN e trilhas 2/3

O PAN, a trilha 2 e a trilha 3 seguem a mesma codificação, em que cada símbolo ocupa um *nibble* (meio byte). Desta forma, adota-se a seguinte regra:

- O pinpad preserva "em claro" os dígitos iniciais, de acordo com a quantidade solicitada em **SPE_OPNDIG**, levando-se em conta que cada byte representa dois dígitos.
- O bloco a ser criptografado deve ter tamanho múltiplo de 8 (oito) bytes. Caso necessário, ele deve ser preenchido com *nibbles* **Fh** ao final ("*filler*").

➔ Exemplo

O diagrama a seguir ilustra a criptografia de uma trilha de 37 posições (“66733246732413D1512601234879534275432F”**FF**FFFF) usando-se criptografia de bloco tipo **CBC**, sem “IV” (*Initialization Vector*), em que se deseja preservar “em claro” os 4 primeiros dígitos:



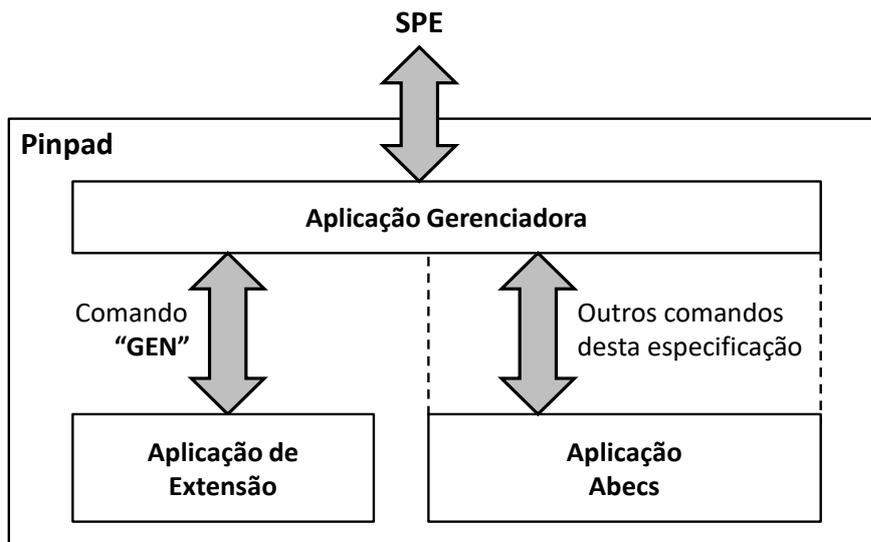
6. Funcionamento Interno do Pinpad

Este capítulo define as regras de funcionamento interno dos pinpads e destina-se aos seus fabricantes e desenvolvedores, sendo seu conhecimento dispensável aos fornecedores de SPE.

6.1. Arquitetura de *software*

6.1.1. Estrutura de aplicações

Esta especificação considera que o pinpad possua internamente aplicações independentes, conforme diagrama:



- A aplicação “Gerenciadora” é responsável por gerenciar o protocolo de comunicação e encaminhar os comandos recebidos para execução nas aplicações inferiores.
- A aplicação “Abecs” é responsável por executar todos os comandos descritos nesta especificação, com exceção dos comandos descritos na **seção 3.8**.
- A aplicação de “Extensão” é responsável por executar exclusivamente os comandos descritos na **seção 3.8** (“GEN”).
- As aplicações “Gerenciadora” e “Abecs” podem, opcionalmente, ser aglutinadas em uma única aplicação, dependendo das necessidades de arquitetura do pinpad.
- Cada uma das três aplicações possui uma identificação de versão distinta (informadas por **PP_MANVERS**, **PP_APPVERS** e **PP_GENVERS** na resposta do comando “GIX”). Caso as aplicações “Gerenciadora” e “Abecs” estejam aglutinadas, **PP_MANVERS** e **PP_APPVERS** serão idênticos.

6.1.2. Capacidades mínimas requeridas

Esta especificação prevê que os pinpads tenham capacidade para suportar os seguintes requisitos mínimos:

- Até **128** (cento e vinte e oito) AIDs simultâneos para o processo de seleção de aplicação EMV (ver **seções 6.8.1 e 6.9.1**), para cada tipo de tecnologia (ICC e CTLS).
- Até 99 Redes Credenciadoras (índices de “01” a “99”).

- Armazenamento e capacidade de substituição de no mínimo a seguinte quantidade total de Tabelas EMV (contemplando-se uma ou mais redes credenciadoras):
 - ⇒ **160** (cento e sessenta) registros de Tabelas de AID (ver **seção 4.1.1**).
 - ⇒ **80** (oitenta) registros de Tabelas de CAPK (ver **seção 4.1.2**).

Estima-se que o espaço de memória não volátil necessário para o armazenamento destas tabelas, sem nenhuma otimização, gira em torno de 96 KBytes. Deve-se também reservar espaço (volátil ou não) para a atualização total desta mesma quantidade de tabelas (ver **seções 6.7.3 e 6.7.4**).
- Não há uma quantidade máxima de registros de Tabelas de AID ou CAPK por rede credenciadora. Apenas a quantidade total de registros é relevante para o armazenamento e substituição, assim como a capacidade máxima de AIDs simultâneos que de fato entram no processamento (definido acima).
- No mínimo **64 chaves** de criptografia (**índices “00” a “63”**) para os seguintes tipos:
 - ⇒ Master Key Triple-DES para PIN (MK:TDES:PIN);
 - ⇒ Master Key Triple-DES para dados (MK:TDES:DAT);
 - ⇒ DUKPT Triple-DES para PIN (DUKPT:TDES:PIN); e
 - ⇒ DUKPT Triple-DES para dados (DUKPT:TDES:DAT).

6.1.3. Protocolos adicionais

Por questão de compatibilidade com sistemas legados, os pinpads podem também processar protocolos adicionais (proprietários do fabricante) não regidos por esta especificação. Entretanto, é extremamente recomendável que estes processamentos sejam feitos na Aplicação de Extensão, permitindo que o fabricante dê manutenção nestas funcionalidades sem afetar Aplicação Abecs.

|| ⚠ Esta opção é considerada obsoleta por esta especificação e será retirada em versões futuras.

6.2.2. Nível de Aplicação

No **Nível de Aplicação** o pinpad efetua o processamento do comando recebido em **PKTDATA** para, ao final, devolver uma mensagem de resposta.

Os três primeiros caracteres de **PKTDATA** (ou de **CLRDATA**, no caso de “Comunicação Segura”) compõem o código do comando **CMD_ID**, sendo que a forma de processamento de cada um dos comandos está descrita a partir da **seção 6.4** deste capítulo.

➔ Observações

- Caso o comando (**CMD_ID**) não seja conhecido pelo pinpad, este deve retornar **RSP_ID** = “ERR” e **RSP_STAT** = ↵ST_INVCALL, conforme descrito na **seção 2.3.4**.
- Caso o pinpad identifique um erro de integridade na estrutura de um Comando Abecs recebido (ver estrutura na **seção 3.1.3.1**), este deve finalizar o processamento com **RSP_STAT** = ↵ST_INVPARAM.
- Caso o SPE envie em um Comando Abecs mais de um parâmetro com o mesmo **CMD_PARID** e isso não for uma característica do comando (como é o caso de “**MNU**”), o pinpad deve simplesmente acatar o primeiro valor e ignorar os demais.
- Caso o SPE envie em um Comando Abecs um parâmetro não previsto (ou desconhecido), este parâmetro deve ser simplesmente desprezado.

6.3. Segurança

6.3.1. Mapeamento de chaves

De acordo com as exigências do PCI, as chaves de “PIN” e de “dados” devem estar armazenadas de forma separada.

A tabela a seguir ilustra qual o tipo de chave a ser usada pelos comandos desta especificação:

| Chave de PIN | Chave de dados |
|----------------|----------------|
| “ <u>GDU</u> ” | “ <u>DWK</u> ” |
| “ <u>GPN</u> ” | “ <u>EBX</u> ” |
| “ <u>GOC</u> ” | “ <u>ENB</u> ” |
| “ <u>GOX</u> ” | “ <u>GTK</u> ” |

6.3.2. Comunicação Segura

Adicionalmente aos procedimentos descritos na **seção 5.2**, deve-se levar em conta:

- Se o pinpad receber um pacote criptografado sem que tenha sido estabelecida a “Comunicação Segura”, ele deve retornar “**ERR003**” (↳ST_NOSEC, conforme descrito na **seção 2.3.4**).
- Se o pinpad receber um pacote “em claro” com a “Comunicação Segura” estabelecida, excetuando-se os comandos “OPN” (clássico e seguro), ele deve simplesmente retornar ↳ST_ERRPKTSEC para o comando em questão (a resposta deve ser “em claro”, de forma a ser compreendida pelo SPE).
- Depois de estabelecida a “Comunicação Segura”, todas as mensagens retornadas pelo pinpad ao SPE devem ser criptografadas, excetuando-se as respostas aos comandos “**CLO**” e “**CLX**”, que são responsáveis por desativar esse processo.
- O pinpad deve sempre acatar o comando “**OPN**” (clássico ou seguro) quando em “Comunicação Segura”, desabilitando este processo e eventualmente estabelecendo uma nova chave K_{SEC} no caso de um “**OPN**” (seguro). Isto é necessário pois o SPE deve ser capaz de iniciar o processamento sem conhecer o estado do pinpad.

➔ Casos especiais:

- Se o pinpad detectar inconsistências em um comando criptografado conforme descrito na **seção 5.2.2.2**, ele deve retornar “**ERR009**” (↳ST_ERRPKTSEC, conforme descrito na **seção 2.3.4**).
- Considerando a “Comunicação Segura” já estabelecida, se o pinpad receber o comando “**OPN**” e este vier criptografado, o pinpad deve devolver a resposta “**OPN010**” (↳ST_INVCALL).

Para estes casos, o pinpad deve efetuar os mesmos procedimentos descritos no comando “**CLO/CLX**” (**seção 6.4.5**), de forma a desativar a “Comunicação Segura” e considerar o pinpad como “fechado” (o SPE deverá efetuar um novo “**OPN**” para restabelecer o diálogo com o pinpad).

As respostas descritas devem ser devolvidas “em claro”, uma vez que a “Comunicação Segura” foi desativada.

6.3.3. PAN Criptografado

O processo de “PAN Criptografado” está detalhado na **seção 5.3**.

Observações quanto ao processamento interno ao pinpad:

- Nas respostas aos comandos “GCR” e “CKE”, caso a trilha 1 ou 3 recodificada ultrapasse seu tamanho máximo (76 ou 104 caracteres respectivamente), ela simplesmente não deve ser devolvida, evitando-se assim que a informação seja truncada ou que haja invasão do campo subsequente. Neste caso, tudo se passa como se a trilha não tivesse sido lida.

6.3.4. Criptografia “End-to-End”

O processo de criptografia “End-to-End” está detalhado na **seção 5.4**, porém alguns pontos importantes devem ser observados na implementação da aplicação do pinpad:

6.3.4.1. Trilhas Incompletas

Para a devolução incompleta das trilhas, o pinpad deve observar os separadores de forma a identificar o *Service Code*, da seguinte forma:

- **Trilha 1:** Retornar os dados da trilha até o 7º dígito depois do segundo separador “^” (5Eh).
- **Trilha 2:** Retornar os dados da trilha até o 7º dígito depois do primeiro separador “=” (3Dh).
- **Trilha 3:** Não havendo um formato padrão para os cartões de pagamento, o pinpad deve sempre devolver os primeiros caracteres da trilha (no máximo 19).

Na impossibilidade de se identificar os separadores nas trilhas, o pinpad deverá devolver os 19 (dezenove) primeiros dígitos ou caracteres da trilha em questão.

Depois de devolver a trilha incompleta, o pinpad armazena as trilhas completas para devolução no comando “GTK”.

6.3.4.2. Criptografia DUKPT

➔ Modo ECB

Cada criptografia DUKPT “consome” um número de série de chave (“KSN”). Para evitar este “desperdício”, o pinpad deve concatenar as informações requeridas na resposta ao comando “GTK” (dados “em claro” de PP_ENCPAN, PP_TRACK1, PP_TRACK2 e PP_TRACK3, nesta ordem) e aplicar a criptografia sobre o bloco total resultante.

Ao final, os campos PP_ENCPAN, PP_TRACK1, PP_TRACK2 e PP_TRACK3 a serem devolvidos são preenchidos com os dados criptografados, sendo que os campos PP_ENCPANKSN, PP_TRK1KSN, PP_TRK2KSN e PP_TRK3KSN acabam recebendo um valor único de KSN.

Esta medida “economiza” chaves DUKPT e, dependendo da tecnologia, pode reduzir o tempo de processamento do pinpad.

➔ Modo CBC

Nesta modalidade os dados “em claro” não podem ser concatenados, dado que o um bloco criptografado influenciaria no cálculo do bloco seguinte.

Assim sendo, no modo CBC os campos **PP_ENCSPAN**, **PP_TRACK1**, **PP_TRACK2** e **PP_TRACK3** devem ser criptografados individualmente, gerando valores de KSN (**PP_ENCANKSN**, **PP_TRK1KSN**, **PP_TRK2KSN** e **PP_TRK3KSN**) diferentes e independentes.

6.3.5. Proteção contra ataques de *replay* em ICC

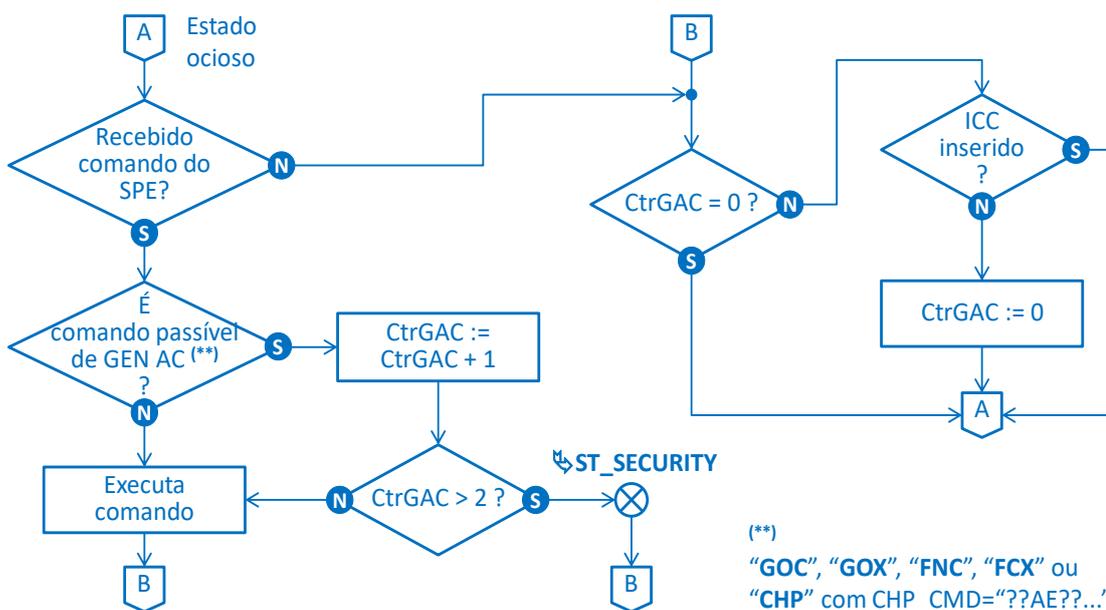
A norma EMV para ICC ([EMV#3](#)) prevê que o comando GENERATE APPLICATION CRYPTOGRAM (ou “GENERATE AC”) seja enviado ao cartão até duas vezes durante o processamento de uma transação. Caso o SPE envie comandos ao pinpad que resultem na submissão de mais de dois GENERATE AC ao cartão em uma única transação, isso deve ser caracterizado como tentativa de ataque de *replay*.

Para identificar essa situação, o pinpad deve contar quantos GENERATE AC estão sendo enviados ao cartão durante uma transação e retornar o erro ↵ST_SECURITY como resposta ao comando que faria terceira tentativa.

⚠ Para este fim, define-se como uma transação todo o período em que um ICC está inserido no pinpad, independentemente de quantas vezes o chip foi ativado (*reset*) ou quais comandos foram a ele submetidos. Uma nova transação somente será iniciada quando o cartão for removido e inserido novamente.

De acordo com esta especificação, o comando GENERATE AC somente é enviado ao ICC nos comandos “GOC”, “GOX”, “FNC”, “FCX” ou de forma explícita no comando “CHP” (se o byte INS dentro de **CHP_CMD** tiver o valor “AE”).

O fluxograma a seguir ilustra uma maneira simples de se implementar essa proteção, valendo-se de uma variável “CtrGAC” para contar quantos GENERATE AC foram submetidos dentro da mesma transação:



- ⚠ Nenhum outro comando deve influenciar o valor do “CtrGAC”, sendo que este contador é mantido inalterado mesmo na execução dos comandos “OPN”, “CLO” ou “CLX”.

6.4. Processamento dos comandos de controle

Esta seção descreve os tratamentos internos do pinpad para os comandos descritos na **seção 3.2**.

6.4.1. Comando “OPN”

Este comando inicializa recursos do equipamento (se necessário) e ativa o “backlight” do *display* (se existente). Por uma questão de robustez, o pinpad sempre deve acatar este comando, mesmo que o comando “**CLO/CLX**” não tenha sido recebido.

Caso o comando não contenha dados, assume-se a modalidade “clássica”, em que não há “Comunicação Segura”. Caso sejam recebidos corretamente os dados da chave pública, assume-se a modalidade “segura” e o pinpad deve retornar o criptograma **CRKSEC** (ver **seção 5.2.1**).

- ⚠ O comando “**OPN**” nunca pode ser enviado pelo SPE criptografado em “Comunicação Segura”, sendo que somente deve ser aceito pelo pinpad se vier “em claro”. Se isso ocorrer, deve-se seguir o procedimento descrito na **seção 6.3.2**.
- ⚠ Caso o pinpad já esteja operando em “Comunicação Segura” ou “PAN Criptografado” e receba um novo “**OPN**” (seja este clássico ou seguro) sem ter sido precedido por “**CLO/CLX**”, esses recursos de segurança devem ser automaticamente desativados, como se um “**CLO/CLX**” tivesse de fato sido recebido. A “Comunicação Segura” será restabelecida caso seja solicitada no novo “**OPN**” recebido.
- ⚠ Por uma questão de robustez e compatibilidade com o parque instalado de SPE, o comando “**OPN**” (clássico) é opcional para o funcionamento do pinpad. Caso o pinpad receba qualquer comando sem ter recebido previamente um “**OPN**”, ele deve antes efetuar internamente os processamentos equivalentes a um “**OPN**” (clássico) de forma que o comando recebido possa ser processado normalmente, assim como eventuais comandos subsequentes.
- ⚠ Por uma questão de robustez e compatibilidade com o parque instalado de SPE, o comando “**OPN**” (clássico) deve ser aceito também com **CMD_LEN1 = “000”** (ou seja, no formato “OPN000”).

➡ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_INVCALL | Comando foi recebido criptografado via “Comunicação Segura”. |

6.4.2. Comando “GIN”

Este comando é de processamento simples e não há nenhuma especificidade a ser descrita nesta seção.

↪ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |

6.4.3. Comando “GIX”

Ao receber este comando, o pinpad deve devolver as diversas informações solicitadas. Cabe ressaltar que qualquer identificador desconhecido fornecido em SPE_IDLIST deve simplesmente ser desprezado.

Por uma questão de simplicidade de processamento, o pinpad deve devolver as informações solicitadas na ordem em que os identificadores aparecem em SPE_IDLIST e, caso o SPE solicite informações repetidas, estas devem ser devolvidas normalmente, sem nenhuma crítica.

Caso o tamanho das informações geradas ultrapasse 999 bytes, os dados de resposta devem ser distribuídos dois ou mais blocos conforme descrito na **seção 3.1.3.2**.

↪ Situações de exceção:

| RSP_STAT | Situação |
|-----------------------|---|
| ↪ST_MANDAT | <u>SPE_IDLIST</u> não foi fornecido. |
| ↪ST_INVPARM | <u>SPE_IDLIST</u> possui tamanho ímpar. |
| ↪ST_RSPOVFL | Tamanho da resposta ultrapassa máximo permitido pelo protocolo. |

6.4.4. Comando “DWK”

Ao receber este comando, o pinpad deve efetuar o processo descrito na **seção 5.3**, de forma a estabelecer e armazenar a chave **WK_{PAN}** em sua memória volátil.

↪ Situações de exceção:

| RSP_STAT | Situação |
|-------------|--|
| ↪ST_INVCALL | Pinpad está em modo de “Comunicação Segura”. |

| RSP_STAT | Situação |
|-------------|--|
| ↪ST_INVPARM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ Se for “modalidade 1” e ENB_MKIDX estiver fora da faixa usada pelo pinpad. ▪ Se for “modalidade 2” e o primeiro byte de DWK_RSAMOD for menor ou igual a que 54h. |
| ↪ST_ERRKEY | Se for “modalidade 1” e a chave indicada por ENB_MKIDX não estiver carregada no pinpad |

6.4.5. Comandos “CLO” e “CLX”

Ao receber este comando o pinpad pode liberar recursos do equipamento (se necessário) e, além disso, deve:

- Desativar ou **esmaecer** o “backlight” do *display* (se existente).
- Se a modalidade “Comunicação Segura” estiver ativa (ver **seção 5.2**), esta deve ser desativada e a chave **K_{SEC}** deve ser apagada da memória volátil. A mensagem de resposta ao comando deve, portanto, ser sempre devolvida “em claro”.
- Eventuais trilhas armazenadas na memória volátil (para recuperação no comando “**GTK**”) devem ser apagadas.
- Se a modalidade “PAN Criptografado” estiver ativa, esta deve ser desativada e a chave **WK_{PAN}** deve ser apagada da memória volátil.
- A interface de ICC deve ser desligada.
- A antena CTLS deve ser desativada.

No caso do comando “**CLO**”, eventuais inconsistências nos dados da mensagem (**CLO_MSG**) não devem causar erro, devendo-se fazer os mesmos tratamentos do comando “**DSP**” (ver **seção 6.5.5**).

No caso do comando “**CLX**”, se o parâmetro **SPE_MFNAME** for recebido, deve-se apresentar o conteúdo no *display* conforme processamento descrito no comando “**DSI**” (ver **seção 6.6.6**). Em caso de falha, deve-se usar a mensagem informada em **SPE_DSPMSG**. Se o pinpad não suportar comandos multimídia, o parâmetro **SPE_MFNAME** deve ser simplesmente desprezado.

↻ Situações de exceção:

Não há.

6.5. Processamento dos comandos básicos

Esta seção descreve os tratamentos internos do pinpad para os comandos descritos na **seção 3.3**.

6.5.1. Comando “**CEX**”

Este comando é uma evolução do “**CKE**”, tornando-o mais flexível. Seu processamento é praticamente idêntico, excetuando-se a questão da devolução das trilhas magnéticas incompletas para cumprir com a Criptografia “*End-to-End*” (ver **seção 6.3.4.1**).

Observações:

- Deve-se desprezar eventuais cartões magnéticos ou teclas pressionadas antes da execução do comando (o pinpad não deve “guardar” estes eventos).
- Ao receber o comando, independentemente dos eventos sendo requisitados, o pinpad deve sempre limpar eventuais trilhas que estejam armazenadas para leitura através de “**GTK**”.
- Ao ler um cartão magnético com sucesso, o pinpad deve soar um único “*beep*”. No caso de erro de leitura, em que nenhuma trilha é lida com sucesso, o pinpad deve soar dois “*beeps*”.
 - ⇒ Se uma trilha possuir tamanho inválido (por exemplo, trilha 1 com mais de 76 posições), a trilha em questão é considerada como “não lida” (erro de leitura).
 - ⇒ Diferentemente do “**CKE**”, o erro na leitura de um cartão magnético não incorre em situação de exceção.
- Este comando não deve mudar o estado do ICC, ligando-o ou desligando-o. Caso o ICC seja selecionado para geração de evento, somente o seu sensor de presença deve ser verificado.
- Caso a detecção de CTLS tenha sido requerida em um pinpad que suporta essa interface, deve-se obedecer às regras a seguir:
 - ⇒ A antena sempre deve ser desligada ao final do processamento (mesmo que o evento detectado seja outro).
 - ⇒ Para proteger a antena, o pinpad deve finalizar o comando se atingidos 2 (dois) minutos de ociosidade, retornando **PP_EVENT** = “93”. Caso **SPE_TIMEOUT** tenha sido fornecido com valor superior a 2 (dois) minutos, ele acabará sendo ignorado neste caso.
- Um pinpad que não suporta CTLS deve simplesmente desprezar a requisição deste evento (mesmo que seja o único).
- Este comando também prevê o retorno de teclas [↑] e [↓], auxiliando o SPE a criar menus de seleção no pinpad. Caso o pinpad não possua estas teclas, devem-se elencar teclas alternativas que representem estas funções, de forma a retornar os valores “02” e “03” em **PP_EVENT**.
- Por uma questão de robustez, o pinpad deve aceitar **SPE_CEXOPT** com qualquer tamanho, simplesmente desprezando os eventos eventualmente não cobertos pelo dado recebido. Da mesma forma, qualquer caractere recebido que esteja fora da faixa definida deve ser considerado como “0” (ignora evento).
- Caso **SPE_TIMEOUT** tenha sido recebido, o pinpad deve finalizar a operação com **ST_TIMEOUT** se o tempo de espera por um evento ultrapassar o valor definido.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|--|
| ↳ST_MANDAT | <u>SPE_CEXOPT</u> ausente. |
| ↳ST_TIMEOUT | Esgotado tempo de espera para a ocorrência de um evento. |

6.5.2. Comando “CHP”

Este comando opera em quatro modalidades distintas, de acordo com CHP_OPER.

6.5.2.1. Desligar o cartão (CHP_OPER = “0”)

Ao receber este comando, o pinpad deve simplesmente desligar a interface de ICC (mesmo que o cartão não esteja presente) indicada por CHP_SLOT.

Se CHP_SLOT = “9”, deve-se desligar a antena CTLS.

Os campos CHP_CMD, CHP_PINFMT e CHP_PINMSG são ignorados.

Não há dados de resposta (CHP_RSLEN = “000”).

➔ Situações de exceção:

Não há.

6.5.2.2. Ligar o cartão (CHP_OPER = “1”)

Ao receber este comando, o pinpad deve ativar a interface de ICC indicada por CHP_SLOT, através de um “cold reset”. Se a operação for bem-sucedida, o ATR completo do cartão é devolvido em CHP_RSP.

Se CHP_SLOT = “9”, deve-se primeiramente ativar a antena e depois o cartão CTLS. Se não houver cartão presente, deve-se retornar ↳ST_NOCARD imediatamente.

- ⚠ A antena CTLS permanecerá ativa até que seja solicitado o seu desligamento (CHP_OPER = “0”), ou deve ser desligada automaticamente pelo pinpad no caso de erro de comunicação com o CTLS ou se atingidos 2 (dois) minutos de ociosidade.
- ⚠ No caso de CTLS, o ATR deve ser montado conforme definido no padrão **PC/SC parte 3**, dado que este tipo de cartão não retorna esta informação como um ICC.

Os campos CHP_CMD, CHP_PINFMT e CHP_PINMSG são ignorados nesta modalidade.

➤ Situações de exceção:

| RSP_STAT | Situação |
|------------------|---|
| ↵ST_NOCARD | Não há cartão presente no acoplador ou antena. Se a antena CTLS estiver ativa, esta deve ser desativada. |
| ↵ST_DUMBCARD | ICC inserido, mas não responde (“mudo”). |
| ↵ST_ERRCARD | Erro de comunicação entre o pinpad e o cartão. |
| ↵ST_CTLSMULTIPLE | Mais de um CTLS foi apresentado ao leitor simultaneamente (a antena deve ser desativada). |
| ↵ST_CTLSCOMMERR | Erro de comunicação entre o pinpad (antena) e o CTLS (a antena deve ser desativada). |
| ↵ST_INVPARAM | Referência fornecida em CHP_SLOT não é suportada pelo equipamento (slot de SAM inexistente ou CTLS não suportado). |

6.5.2.3. Troca de comando (**CHP_OPER = “2”**)

Ao receber este comando, o pinpad deve enviar ao cartão indicada por **CHP_SLOT** o APDU recebido em **CHP_CMD**, seja ICC ou CTLS.

A resposta do cartão deve ser devolvida em **CHP_RSP**, com os bytes de status (SW1/SW2) ao final.

Os campos **CHP_PINFMT** e **CHP_PINMSG** são ignorados nesta modalidade.

⚠ No caso de ICC com protocolo T=0, o pinpad não deve tratar internamente os casos em que o cartão devolve status SW1/SW2 = **61xxh** ou **6Cxxh**. Os bytes de status devem ser devolvidos ao SPE para que o tratamento seja feito externamente.

➤ Situações de exceção:

| RSP_STAT | Situação |
|-----------------|---|
| ↵ST_NOCARD | Não há cartão presente no acoplador ou antena. Se a antena CTLS estiver ativa, esta deve ser desativada. |
| ↵ST_DUMBCARD | ICC inserido, mas não responde (“mudo”) (não se aplica a CTLS). |
| ↵ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC, ou cartão não foi ativado através de CHP_OPER = “1” . |
| ↵ST_CTLSCOMMERR | Erro de comunicação entre o pinpad (antena) e o CTLS (a antena deve ser desativada). |
| ↵ST_INVPARAM | Referência fornecida em CHP_SLOT não é suportada pelo equipamento (slot de SAM inexistente ou CTLS não suportado). |
| ↵ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.5.2.4. Verificação de PIN (CHP_OPER = “3”)

Ao receber este comando, o pinpad solicitará uma captura de PIN apresentando a mensagem informada em CHP_PINMSG. Caso não haja nenhuma ação no teclado por 1 minuto, o comando falha por “timeout”.

Depois de capturado, o PIN é codificado de acordo com o formato indicado em CHP_PINFMT e anexado ao final do APDU informado em CHP_CMD (incluindo-se o byte Lc, que depende do formato).

O resto do processamento se passa exatamente como o descrito para CHP_OPER = “2”.

↪ Situações de exceção:

| RSP_STAT | Situação |
|-----------------|---|
| ↪ST_NOCARD | Não há cartão presente no acoplador ou antena. Se a antena CTLS estiver ativa, esta deve ser desativada. |
| ↪ST_DUMBCARD | ICC inserido, mas não responde (“mudo”) (não se aplica a CTLS). |
| ↪ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC, ou cartão não foi ativado através de <u>CHP_OPER</u> = “1”. |
| ↪ST_CTLSCOMMERR | Erro de comunicação entre o pinpad (antena) e o CTLS (a antena deve ser desativada). |
| ↪ST_CANCEL | Portador pressionou a tecla [CANCELAR] na tela de captura de PIN. |
| ↪ST_TIMEOUT | Tempo máximo de ociosidade atingido (1 minuto sem ação do operador). |
| ↪ST_INVPARAM | Referência fornecida em <u>CHP_SLOT</u> não é suportada pelo equipamento (slot de SAM inexistente ou CTLS não suportado). |

6.5.3. Comando “CKE”

Ao receber este comando, o pinpad permanece aguardando a ocorrência de qualquer um dos eventos selecionados pelo SPE.

- Deve-se desprezar eventuais cartões magnéticos ou teclas pressionadas antes da execução do comando (o pinpad não deve “guardar” estes eventos).
- Ao ler um cartão magnético com sucesso, o pinpad deve soar um único “beep”. No caso de erro de leitura, em que nenhuma trilha é lida com sucesso, o pinpad deve soar dois “beeps” e retornar ↪ST_MCDATAERR.
 - ⇒ Se uma trilha possuir tamanho inválido (por exemplo, trilha 1 com mais de 76 posições), a trilha em questão é considerada como “não lida” (erro de leitura).
- Este comando não deve mudar o estado do ICC, ligando-o ou desligando-o. Caso o ICC seja selecionado para geração de evento, somente o seu sensor de presença deve ser verificado.
- Caso a detecção de CTLS tenha sido requerida em um pinpad que suporta essa interface, deve-se obedecer às regras a seguir:

- ⇒ A antena sempre deve ser desligada ao final do processamento (mesmo que o evento detectado seja outro).
- ⇒ Para proteger a antena, o pinpad deve finalizar o comando se atingidos 2 (dois) minutos de ociosidade, retornando CKE_CTLSSTAT = "0".
- Um pinpad que não suporta CTLS deve simplesmente desprezar a requisição deste evento (mesmo que seja o único).
- Na modalidade "PAN Criptografado", **a trilha 2 pode ultrapassar 37 caracteres** em algumas situações (tipicamente quando o PAN tem menos do que 16 dígitos). Desta forma, o campo CKE_TRK3LEN não é preenchido pelo pinpad nesta modalidade, mesmo que a trilha 3 tenha sido lida com sucesso (somente o campo CKE_TRK3 é preenchido).
- Por uma questão de robustez, qualquer caractere recebido em CKE_KEY, CKE_MAG, CKE_ICC ou CKE_CTLS que esteja fora da faixa definida deve ser considerado como "0" (ignora evento).

↪ Situações de exceção:

| RSP_STAT | Situação |
|---------------|---|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_TIMEOUT | Esgotado tempo de espera para a apresentação de um CTLS. |
| ↪ST_MCDATAERR | Um cartão magnético foi passado, mas nenhuma trilha pôde ser lida com sucesso. |

6.5.4. Comando "DEX"

Ao receber este comando, o pinpad deve substituir o conteúdo do *display* pela nova mensagem recebida, posicionando-a no canto superior esquerdo de acordo com o alinhamento definido em DEX_OPTIONS.

Caso o pinpad possua diferentes resoluções de *display* para apresentação de texto (linhas x colunas), deve-se selecionar a menor resolução possível para apresentação da mensagem recebida (caracteres maiores), de modo que esta fique mais visível.

Por uma questão de robustez, qualquer caractere menor do que 20h (espaço) recebido em DEX_MSG deve ser considerado como quebra de linha.

Para o tratamento "tipo de mensagem" em DEX_OPTIONS, deve-se utilizar recursos de cor e/ou imagens para melhor indicar a intenção do SPE. Por exemplo, para uma mensagem de erro, pode-se apresentar o texto em branco sobre um fundo vermelho, com um ícone "☒".

A tabela a seguir indica o som a ser emitido pelo pinpad em cada situação:

| <u>DEX_OPTIONS</u> | Som a ser emitido |
|-------------------------------|---|
| "xx0xxx" = Genérica (default) | Nenhum. |
| "xx1xxx" = Sucesso | Um ou dois tons mais agudos de curta duração. |
| "xx2xxx" = Erro | Tom mais grave de longa duração. |

| <u>DEX_OPTIONS</u> | Som a ser emitido |
|------------------------|----------------------------------|
| "xx3xxx" = Alerta | Tom mais grave de curta duração. |
| "xx4xxx" = Informativa | Nenhum. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|--|
| ↳ST_INVPARM | <ul style="list-style-type: none"> ▪ <u>DEX_MSGLEN</u> não corresponde ao tamanho de <u>DEX_MSG</u>. ▪ <u>DEX_MSGLEN</u> > 160. |

6.5.5. Comando "DSP"

Ao receber este comando, o pinpad deve substituir o conteúdo do *display* pela nova mensagem recebida, centralizando-a da melhor forma dentro das capacidades do dispositivo.

Por uma questão de robustez, devem-se abrir as seguintes exceções:

- Qualquer caractere menor do que 20h (espaço) recebido em DSP_MSG deve ser considerado como 20h (espaço).
- Caso o pacote recebido faça com que DSP_MSG tenha menos do que 32 caracteres, deve-se preencher a parte faltante com espaços.
- Deve-se ignorar eventuais caracteres excedentes no pacote recebido.

➔ Situações de exceção:

Não há.

6.5.6. Comando "EBX"

Este comando é de processamento simples e não há nenhuma especificidade a ser descrita nesta seção.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|--|
| ↳ST_INVPARM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ <u>SPE_KEYIDX</u> está fora da faixa usada pelo pinpad. |
| ↳ST_MANDAT | <ul style="list-style-type: none"> ▪ Ausência de um ou mais parâmetros mandatórios (M). ▪ <u>SPE_MTHDDAT</u> = "1x" e <u>SPE_WKENC</u> não foi recebido. |

| RSP_STAT | Situação |
|------------|--|
| ↳ST_ERRKEY | A chave indicada por <u>SPE_KEYIDX</u> não está carregada no pinpad. |

6.5.7. Comando “ENB”

Este comando é de processamento simples e não há nenhuma especificidade a ser descrita nesta seção. Cabe ressaltar que o campo ENB_INPUT pode ser recebido criptografado, conforme descrito na [seção 5.3](#).

↻ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↳ST_INVPARAM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ <u>ENB_MKIDX</u> está fora da faixa usada pelo pinpad. |
| ↳ST_ERRKEY | A chave indicada por <u>ENB_MKIDX</u> não está carregada no pinpad. |

6.5.8. Comando “GCD”

Neste comando o pinpad efetua uma captura de dados, respeitando-se as seguintes regras:

- As mensagens definidas na [seção 3.3.8](#) propositalmente não possuem formatação, de forma que cada pinpad deve apresentá-las da melhor forma possível dentro das capacidades de seu *display*.
- Os dados digitados devem sempre ser apresentados “em aberto” (nunca devem ser mascarados por asteriscos ou outro símbolo).
- Os dados digitados devem ser apresentados sob a mensagem fixa, com alinhamento à direita.
- No caso de captura de dados alfanuméricos (SPE_GCDOPT = “1xx”), a forma como isso é feito depende dos recursos do pinpad e não é objeto desta especificação.
- Caso a quantidade de caracteres digitados ultrapasse a quantidade de colunas do *display*, os dados devem ser rotacionados para a esquerda.
- Ao final do processamento, o pinpad sempre apaga o *display*, seja a captura bem ou malsucedida.

↻ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↳ST_MANDAT | <u>SPE_MSGIDX</u> ausente. |
| ↳ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↳ST_TIMEOUT | Esgotado tempo definido por <u>SPE_TIMEOUT</u> . |

| RSP_STAT | Situação |
|------------|--|
| ↳ST_CANCEL | Portador pressionou a tecla [CANCELA]. |

6.5.9. Comando “GDU”

Ao receber este comando, o pinpad deve devolver o KSN atual da chave DUKPT indicada.

Por “KSN atual”, entende-se o valor que ser retornado no próximo uso da chave.

↪ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↳ST_INVPARAM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ GDU_IDX está fora da faixa usada pelo pinpad. |
| ↳ST_ERRKEY | A chave indicada por GDU_IDX não está carregada no pinpad. |

6.5.10. Comando “GKY”

Este comando é de processamento simples e não há nenhuma especificidade a ser descrita nesta seção.

↪ Situações de exceção:

| RSP_STAT | Situação |
|----------|---|
| --- | As teclas pressionadas retornam em RSP_STAT , apesar de não constituírem situações de exceção. |

6.5.11. Comando “GPN”

Ao receber este comando, o pinpad efetua uma captura de dados criptografada segundo os padrões estabelecidos pela ANSI X9.8 e X9.24.

- Este comando usa somente chave de PIN.
- Caso não haja nenhuma ação no teclado por 1 minuto (60 segundos), o comando falha por “timeout” (↳ST_TIMEOUT).
- Ao final do processamento, o pinpad sempre apaga o *display*, seja a captura bem ou malsucedida.
- Na modalidade “PAN Criptografado”, deve-se acatar PAN com menos de 16 dígitos (**GPN_PANLEN** < “16”) e considerá-lo “em claro”, dado que a informação recebida não pode constituir um criptograma ~~DES~~/TDES.

- Deve-se verificar a existência da chave de criptografia e, em caso de ausência, retornar erro antes da captura dos dados.
- A mensagem **GPN_MSG1** deve ser apresentada da melhor forma possível dentro das capacidades do *display*. Os dados digitados devem ser apresentados sob a mensagem, com qualquer alinhamento (esquerda, direita ou centralizado), sempre mascarados com asteriscos. Durante a digitação a mensagem **GPN_MSG1** deve ser mantida intacta, não podendo ser sobreposta ou apagada.
- Este comando aceita PAN de 2 a 19 dígitos. Dado que as normas consideram nos cálculos os “12 últimos dígitos do PAN excetuando-se o dígito verificador”, o pinpad deve preencher com zeros à esquerda qualquer PAN recebido com menos de 13 dígitos.

Exemplos:

⇒ O PAN “409127890417894231” de 18 dígitos entra no cálculo como “789041789423”.

⇒ O PAN “670192387” de 9 dígitos entra no cálculo como “000067019238”.

- Para aprimorar a segurança, o PAN não precisa ser fornecido ao pinpad (**GPN_PANLEN** = "00") quando o comando “**CEX**” ou “**G CX**” foi chamado previamente e os dados do cartão ainda estão preservados na memória. Neste caso, a criptografia do PIN deve utilizar o PAN do cartão lido, extraído da trilha no caso de cartão magnético.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↳ST_INVPARM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ GPN_KEYIDX está fora da faixa usada pelo pinpad. ▪ GPN_MIN1 é inferior a “04”. ▪ GPN_ENTRIES é diferente de “1” ▪ O PAN fornecido (dado “em claro” no caso de “PAN Criptografado”) possui caracteres não numéricos (espaços, letras, etc). |
| ↳ST_TIMEOUT | Tempo máximo de ociosidade atingido (1 minuto sem ação do operador). |
| ↳ST_ERRKEY | A chave indicada por GPN_KEYIDX não está carregada no pinpad. |
| ↳ST_INVCALL | Foi passado um PAN “vazio” (GPN_PANLEN = "00") sem que houvesse uma chamada prévia a “ CEX ” ou “ G CX ”, ou os dados do cartão não estão mais na memória (o comando “ G TK ” já foi chamado). |

6.5.12. Comando “**G TK**”

As trilhas de cartão lidas nos comandos “**CEX**” ou “**G CX**” são preservadas pelo pinpad em memória volátil para serem recuperadas através deste comando, conforme detalhado na **seção 5.4**.

- Este comando somente pode ser executado uma única vez depois de “**CEX**” ou “**G CX**”. Depois da execução deste comando, as trilhas de cartão magnético devem ser apagadas da memória (zeradas) para cumprir os requisitos de segurança PCI.

 A execução deste comando não deve afetar dados armazenados nos Kernels EMV ICC/CTLS, apesar de eles conterem informações de PAN e trilha.

- Por uma questão de robustez, o pinpad deve aceitar **SPE_TRACKS** com qualquer tamanho. Qualquer caractere que não tenha sido fornecido ou que esteja fora da faixa definida deve ser considerado como “0” (não devolve a trilha/dado).
- Características específicas de criptografia DUKPT estão descritas na **seção 6.3.4.2**.
- Em caso de erro de leitura do cartão magnético em “**CEX**” ou “**GCX**”, este comando deve seguir o mesmo comportamento e simplesmente retornar ↵ST_OK, sem os campos que contêm dados de cartão.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|--|
| ↵ST_INVCALL | <ul style="list-style-type: none"> ▪ Um comando “CEX” ou “GCX” não foi executado previamente com sucesso. ▪ O comando “GTK” já foi utilizado. |
| ↵ST_INVPARM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ SPE_KEYIDX está fora da faixa usada pelo pinpad. |
| ↵ST_MANDAT | <ul style="list-style-type: none"> ▪ SPE_MTHDDAT está presente com valor diferente de “9x” e SPE_KEYIDX não foi fornecido. ▪ SPE_MTHDDAT = “1x” e SPE_WKENC não foi fornecido. ▪ SPE_MTHDDAT = “9x” e SPE_PBKMOD ou SPE_PBKEXP não foram fornecidos. |
| ↵ST_ERRKEY | A chave indicada por SPE_KEYIDX não está carregada no pinpad. |

6.5.13. Comando “**MNU**”

Ao receber este comando, o pinpad deve apresentar no *display* um menu de opções para seleção do portador. A forma de implementação deste menu fica a cargo do desenvolvedor, de forma a melhor utilizar os recursos do pinpad (*display* gráfico, *touchscreen*, teclas de navegação, etc.).

- O pinpad deve permitir a escolha da opção mediante o uso de teclas [↑] [↓] ou “*touchscreen*”. Durante o processo, o pinpad pode apresentar somente as opções que cabem no *display*, de forma que o operador possa “*paginá-las*” ou “*rolá-las*”, porém destacando de alguma forma a opção a ser escolhida pela tecla [OK/ENTRA].
- Caso o pinpad não tenha capacidade de mostrar todos os 24 caracteres nas opções do menu, estas podem ser cortadas à direita, porém a opção em destaque deve, de alguma forma, ser totalmente visível para o operador (“*rolagem*” para a esquerda, por exemplo).
- Havendo somente uma opção, o pinpad pode simplesmente apresentá-la em destaque no menu, não permitindo ao operador nenhuma ação usando as teclas [↑] [↓].

- Opções iniciadas com os caracteres de “0” (30h) a “9” (39h) podem ser selecionadas diretamente através das teclas numéricas correspondentes (caso mais de uma opção seja iniciada pelo mesmo caractere numérico, o pinpad escolherá a que tiver menor índice).
- O pinpad não deve retornar erro caso o título informado em **SPE_DSPMSG** seja grande a ponto de não deixar espaço no *display* para as opções do menu. Neste caso, esse título deve ser truncado para que seja possível apresentar ao menos uma opção para seleção.
- Ao final do processamento, o pinpad sempre apaga o *display*, seja a seleção bem ou malsucedida.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|--|
| ↪ST_INVPARM | Alguma opção do menu ultrapassa o tamanho máximo definido ou o SPE não enviou nenhuma opção. |
| ↪ST_TIMEOUT | Atingido tempo máximo de ociosidade definido por SPE_TIMEOUT . |
| ↪ST_CANCEL | Operador pressionou a tecla [CANCELA]. |
| ↪ST_MANDAT | Nenhuma opção (SPE_MNUOPT) foi fornecida. |

6.5.14. Comando “RMC”

Ao receber este comando, o pinpad deve apresentar a mensagem **RMC_MSG** seguindo as mesmas regras de apresentação (e robustez) definidas para o comando “DSP”. Além disso:

- A interface ICC deve ser desligada.
- Caso um ICC não esteja presente, o comando retorna imediatamente.
- Caso um ICC esteja presente, deve-se alternar **RMC_MSG** com a mensagem “RETIRE O CARTÃO”, até que este seja retirado (deve-se esperar indefinidamente).
 - ⇒ A mensagem “RETIRE O CARTÃO” deve ser disposta no *display* da melhor forma possível para o dispositivo.
 - ⇒ As mensagens devem ser alternadas a cada 1,5s (um segundo e meio).
 - ⇒ Pinpads com *display* gráfico podem utilizar recursos mais sofisticados, a gosto do fornecedor, podendo conter animações indicando a remoção do cartão.
 - ⇒ A mensagem **RMC_MSG** deve ser deixada no *display* depois da remoção do cartão.
 - ⇒ Como todo comando “blocante”, este comando pode ser cancelado pelo SPE através do envio de um byte «CAN» (caso em que o pinpad responde com «EOT»). Neste caso o *display* deve ser limpo e o comando finalizado.

➔ Situações de exceção:

Não há.

6.6. Processamento dos comandos multimídia

Esta seção descreve os tratamentos internos do pinpad para os comandos descritos na **seção 3.4**.

6.6.1. Comando “MLI”

Ao receber este comando, o pinpad deve iniciar o processo de carga de um arquivo multimídia utilizando um arquivo temporário ou área de memória volátil. Eventuais arquivos temporários resultantes de processos anteriores inacabados devem ser excluídos.

- O pinpad não deve criticar neste momento o tipo de arquivo sendo carregado, mesmo que este não seja suportado pelo dispositivo. Esta crítica somente será feita no momento do uso.
- Por uma questão de robustez, deve-se aceitar SPE_MFINFO com qualquer tamanho maior ou igual a 7 bytes.

↪ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_MANDAT | <u>SPE_MFNAME</u> ou <u>SPE_MFINFO</u> estão ausentes. |

6.6.2. Comando “MLR”

Ao receber este comando, o pinpad deve simplesmente concatenar as informações recebidas nos blocos SPE_DATAIN, na ordem em que estão presentes no comando, e armazená-las no arquivo temporário (ou memória volátil).

↪ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↪ST_INVCALL | Comando “ <u>MLI</u> ” não foi executado previamente com sucesso. |
| ↪ST_MANDAT | Nenhum campo <u>SPE_DATAIN</u> foi recebido no comando. |
| ↪ST_INTERR | Falta de memória para gerenciamento dos dados recebidos. |

6.6.3. Comando “MLE”

Ao receber este comando, o pinpad avalia os dados recebidos através dos comandos “MLR”, verificando o tamanho e o CRC inicialmente informados em “MLI”. Caso estejam corretos, o arquivo resultante é armazenado em memória não volátil.

Independentemente do resultado (sucesso ou erro), eventuais arquivos temporários devem ser apagados para liberar espaço no pinpad.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↳ST_INVCALL | Comando “MLI” não foi executado previamente com sucesso. |
| ↳ST_MFERR | Tamanho do arquivo recebido ou CRC não correspondem às informações fornecidas no comando “MLI” (SPE_MFINFO). |
| ↳ST_INTERR | Falta de memória para gerenciamento dos dados recebidos. |

6.6.4. Comando “LMF”

Ao receber este comando, o pinpad pesquisa os arquivos multimídia armazenados com sucesso, devolvendo uma lista contendo seus nomes. Caso nenhum arquivo esteja armazenado, o comando não retorna erro.

Como os nomes dos arquivos não são “*case sensitive*”, o pinpad devolvê-los sempre em maiúsculas, mesmo que estes tenham sido fornecidos de forma diferente no comando “MLI”.

A mensagem de resposta do protocolo comporta mais de 160 nomes de arquivo. Na eventualidade de esta quantidade ser ultrapassada, o pinpad deve retornar sucesso e ignorar os arquivos cujos nomes não couberem na resposta.

➔ Situações de exceção:

Não há.

6.6.5. Comando “DMF”

Ao receber este comando, o pinpad verifica os nomes recebidos e exclui os arquivos cujos nomes forem idênticos aos informados.

Por uma questão de robustez, eventuais nomes inválidos (tamanho incorreto ou caracteres fora da faixa permitida) ou desconhecidos devem simplesmente ser desprezados.

➔ Situações de exceção:

| RSP_STAT | Situação |
|------------|---|
| ↪ST_MANDAT | Nenhum campo SPE_MFNAME foi recebido no comando. |

6.6.6. Comando “DSI”

Ao receber este comando, o pinpad apresenta um arquivo de imagem no *display*, caso seu formato seja suportado.

- A imagem deve substituir totalmente o conteúdo anterior do *display*.
- A imagem deve ser centralizada no display caso suas dimensões sejam inferiores à capacidade do dispositivo. Neste caso, a cor a ser utilizada nas bordas fica a gosto do fabricante.

➔ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↪ST_MANDAT | SPE_MFNAME está ausente. |
| ↪ST_MFNFOUND | Não foi localizado arquivo multimídia com o nome informado. |
| ↪ST_MFERRFMT | Formato do arquivo não aceito pelo pinpad, ou suas dimensões superam a capacidade do <i>display</i> . |

6.7. Processamento dos comandos para manutenção de tabelas

Esta seção descreve os tratamentos internos do pinpad para os comandos descritos na **seção 3.5**.

Todos os comandos se valem do mecanismo de versionamento de tabelas descrito na **seção 4.2**.

6.7.1. Comando “GTS”

Ao receber este comando, o pinpad retorna em **GTS_TABVER** a versão das tabelas correspondentes à Rede Credenciadora indicada em **GTS_ACQIDX**.

Se **GTS_ACQIDX** = “00”, deve-se retornar a versão do conjunto completo de tabelas (ver conceito na **seção 4.2.1**).

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |

6.7.2. Comando “TLI”

Ao receber este comando, o pinpad inicia o processo de atualização de Tabelas EMV, limpando arquivos temporários que eventualmente existam.

Este comando não faz críticas quanto ao conteúdo de **TLI_TABVER**, apenas retorna status diferenciado por uma questão informativa. Ela deve sempre iniciar o processo de carga de tabelas, acatando incondicionalmente a demanda do SPE.

Por uma questão de compatibilidade com os sistemas atualmente em operação, até mesmo um valor zerado (“0000000000”) deve ser aceito como válido em **TLI_TABVER**, apesar de ele indicar a ausência da tabela no retorno de “**GTS**”.

➔ Situações de exceção:

| RSP_STAT | Situação |
|---------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_TABVERDIF | Informativo: o processo de carga de tabelas foi iniciado com sucesso, porém a versão apresentada difere das tabelas carregadas. |

6.7.3. Comando “TLR”

Este comando carrega um ou mais registros das Tabelas EMV recebidos do SPE, armazenando-os de forma temporária. Isso é importante para preservar as tabelas anteriores em caso de erro na operação de atualização.

Ao processar este comando, o pinpad pode apresentar no *display* a mensagem “**ATUALIZANDO TABELAS**”, formatada da forma mais adequada para as capacidades do equipamento.

➔ Observações

- Caso o pinpad receba um registro de tabela com **TAB_ACQ** ≠ **TLI_ACQIDX** e **TLI_ACQIDX** ≠ “00”, esse registro deve ser simplesmente descartado, uma vez que ele não se refere à Rede Credenciadora cujas tabelas estão sendo atualizadas.
- O pinpad não deve retornar erro caso receba um registro de tabela desconhecido, inconsistente ou com campos inválidos. Um registro inválido deve ser simplesmente ignorado.

➔ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_INVCALL | Comando “TLI” não foi chamado previamente. |

6.7.4. Comando “TLE”

Ao receber este comando, o pinpad finaliza o processo de carga (ou atualização) de tabelas, fazendo com que os registros fornecidos através de “TLR” sejam armazenados de forma definitiva, substituindo as tabelas anteriores da Rede Credenciadora indicada por **TLI_ACQIDX** (caso existentes), ou todas as tabelas se **TLI_ACQIDX** = “00”. Nesse momento, a versão fornecida em **TLI_TABVER** passa a vigorar para as novas tabelas.

Caso “TLR” não seja chamado entre “TLI” e “TLE”, o conjunto de tabelas referenciado é simplesmente apagado (se existente).

Se o comando “TLR” apresentar mensagens no display, este deve ser apagado ao final do processamento deste comando.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|--|
| ↪ST_INVCALL | Comando “TLI” não foi chamado previamente. |
| ↪ST_TABERR | Erro ao tentar armazenar os registros (falta de memória, por exemplo). |

6.8. Processamento dos comandos de cartão (obsoletos)

Esta seção descreve os tratamentos internos do pinpad para os comandos descritos na **seção 3.6**. Para a sua compreensão, é imprescindível um conhecimento aprofundado das normas EMV (ICC e CTLS).

Os comandos "**GCR**", "**GOC**" e "**FNC**" devem sempre ser usados nesta sequência, porém o pinpad deve poder aceitar quaisquer outros comandos entre eles, ou seja:

- Qualquer comando pode ser usado entre "**GCR**" e "**GOC**" (com exceção de "**FNC**").
- Qualquer comando pode ser usado entre "**GOC**" e "**FNC**" (com exceção do próprio "**GOC**").

6.8.1. Comando “GCR”

Este comando pede a passagem de um cartão magnético, a inserção ou a aproximação de um cartão com *chip*.

Ao recebê-lo, o pinpad deve inicialmente pesquisar suas Tabelas de AID de forma a identificar quais aplicações de *chip* estão sendo requisitadas pelo SPE. Para identificar quais registros serão envolvidos no processamento, o pinpad deve aplicar a seguinte regra:

- Incluir somente registros que tenham **T1_ICCSTD = “03”** (EMV).
- Avaliar o valor informado em **GCR_APPTYPREQ**:
 - ⇒ Se igual a “99”, aceitar registros das tabelas com qualquer valor de **T1_APPTYPE**.
 - ⇒ Se maior do que “00”, considerar somente os registros em que **T1_APPTYPE** possuam o mesmo valor.
 - ⇒ Se igual a “00”, considerar somente os registros das tabelas que estão informados explicitamente na lista indicada por **GCR_QTDAPP/GCR_IDAPPx**.
- Se **GCR_ACQIDXREQ** e **GCR_APPTYPREQ** forem ambos diferentes de “00”, considerar somente as tabelas em que **TAB_ACQ = GCR_ACQIDXREQ**.

Para simplificar este documento, os registros identificados serão doravante denominados “registros candidatos”.

▲ O Kernel EMV (ICC e CTLS) do pinpad deve suportar uma lista de até **128** (cento e vinte e oito) “registros candidatos”.

Após esse processamento, o pinpad apresenta no *display* a seguinte mensagem, formatada adequadamente de acordo com os recursos do equipamento:

| | GCR_AMOUNT = 0 (zero) | GCR_AMOUNT ≠ 0 (zero) |
|--------------------|--|--|
| CTLS não suportado | INSIRA OU PASSE O CARTÃO | VALOR: 9.999.999.999,99 INSIRA OU PASSE O CARTÃO |
| CTLS suportado (*) |  APROXIME, INSIRA OU PASSE O CARTÃO |  VALOR: 9.999.999.999,99 APROXIME, INSIRA OU PASSE O CARTÃO |

(*) Considera-se que CTLS é suportado somente se todas as condições seguintes forem válidas:

- O equipamento suporte cartões sem contato;
- O parâmetro **GCR_CTLSON** for igual a “1” (ou estiver ausente);
- Ao menos um dos “registros candidatos” indique essa tecnologia (**T1_CTLSMODE** com valor válido entre “1” e “9”);

- Se **GCR_AMOUNT** estiver zerado, ao menos um dos “registros candidatos” indique a possibilidade de processamento *online* neste caso (**T1_CTLSZEROAM** = “1”); e
- Ao menos um dos “registros candidatos” possua o parâmetro **T1_CTLSTRNLIM** ou **T1_CTLSMBTLIM** com valor igual ou superior a **GCR_AMOUNT**.

➔ Observações:

- Deve-se desprezar eventuais cartões magnéticos ou teclas pressionadas antes da execução do comando (o pinpad não deve “guardar” estes eventos).
- Caso já exista um ICC inserido, nenhuma mensagem deve ser mostrada e a interface CTLS não deve ser ativada.
- Enquanto o pinpad aguarda um cartão, o comando pode ser cancelado pelo operador (através da tecla [CANCELA]).
- Enquanto o pinpad aguarda um cartão, o comando pode ser cancelado pelo SPE através do envio de um byte «CAN». (caso em que o pinpad responde com «EOT»). Neste caso o *display* deve ser limpo e o comando finalizado.
- Caso a detecção de CTLS tenha sido requerida, o pinpad deve finalizar o comando se atingidos 2 minutos de ociosidade, caso contrário o pinpad espera indefinidamente por um dos eventos.
- Caso a detecção de CTLS tenha sido requerida, porém outro evento tenha sido detectado, a antena deve ser desligada.
- Após a inserção, passagem ou aproximação do cartão, a mensagem deve ser imediatamente apagada do *display*.
- O comando “**GCR**” pode ser chamado pelo SPE mais de uma vez, independentemente do acionamento de outros comandos (“**GOC**”, “**FNC**”, etc.). Caso isso ocorra, o processamento iniciado pela chamada anterior é desprezado e um novo é iniciado, sem a necessidade da remoção do cartão.
- Se **GCR_APPTYPREQ** for diferente “00”, o pinpad deve poder aceitar o comando sem o campo **GCR_QTDAPP** (que passa a ser opcional). Isto se deve à compatibilidade com sistemas muito antigos.
- Na modalidade “PAN Criptografado”, **a trilha 2 pode ultrapassar 37 caracteres** em algumas situações (tipicamente quando o PAN tem menos do que 16 dígitos). Desta forma, o campo **GCR_TRK3LEN** não é preenchido pelo pinpad nesta modalidade, mesmo que a trilha 3 tenha sido lida com sucesso (somente o campo **GCR_TRK3** é preenchido).
- Em qualquer caso de erro, o conteúdo do *display* deve ser apagado.

Dependendo do tipo de cartão utilizado pelo portador, o processamento é diferente e descrito a seguir.

6.8.1.1. Cartão magnético

Caso um cartão magnético seja passado com sucesso, seus dados são simplesmente retornados conforma tabela a seguir:

| Id. do Campo | Dado retornado |
|---------------------------------------|--|
| <u>GCR_CARDTYPE</u> | Tipo de cartão lido: “00” = Magnético. |
| <u>GCR_STATCHIP</u> | Status do último processamento de ICC (**). |
| <u>GCR_TRK1LEN</u> <u>GCR_TRK1</u> | Dados da trilha 1, se lida com sucesso. |
| <u>GCR_TRK2LEN</u> <u>GCR_TRK2</u> | Dados da trilha 2, se lida com sucesso. |
| <u>GCR_TRK3LEN</u> <u>GCR_TRK3</u> | Dados da trilha 3, se lida com sucesso. |
| Outro... | Retornar valor zerado (“000...”) para os campos numéricos e espaços em branco para os alfanuméricos. |

(**) Este campo tem o propósito de informar ao SPE o que ocorreu com o último processamento de ICC, de forma que este possa tomar a decisão de aceitar ou não uma transação de “fallback” para a tarja magnética lida.

| <u>GCR_STATCHIP</u> | Situação |
|---------------------|--|
| “2” | O comando “ <u>GCR</u> ” imediatamente anterior retornou ↪ST_CARDAPPNAV. |
| “1” | O comando “ <u>GCR</u> ” imediatamente anterior retornou ↪ST_DUMBCARD ou ↪ST_ERRCARD. |
| “1” | O comando “ <u>GCR</u> ” ou “ <u>GOC</u> ” imediatamente anterior retornou erro ↪ST_ERRFALLBACK. |
| “0” | Outra situação, seja sucesso (↪ST_OK) ou falha. |

Ao ler um cartão magnético com sucesso, o pinpad deve soar um único “beep”. No caso de erro de leitura, em que nenhuma trilha é lida com sucesso, o pinpad deve soar dois “beeps”.

Se uma trilha possuir tamanho inválido (por exemplo, trilha 1 com mais de 76 posições), a trilha em questão é considerada como “não lida” (erro de leitura).

➔ Situações de exceção:

| <u>RSP_STAT</u> | Situação |
|-----------------|---|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_CANCEL | A tecla [CANCEL] foi pressionada pelo portador enquanto o pinpad aguardava um cartão. |
| ↪ST_MCDATAERR | Um cartão magnético foi passado, mas nenhuma trilha pôde ser lida com sucesso. |
| ↪ST_TIMEOUT | Atingido tempo de ociosidade de 2 minutos caso CTLS seja requerido. |

6.8.1.2. Cartão com *chip* de contato (ICC EMV)

Caso seja inserido um ICC, o pinpad deve efetuar os seguintes processos iniciais:

- Apresenta a mensagem “**PROCESSANDO...**” no *display*.
- Verifica o valor recebido em **GCR_TABVER** conforme processo descrito na **seção 4.2**. Se a versão diferir, preserva os parâmetros recebidos no comando (em memória não volátil) e retorna ↵ST_TABVERDIF.

Depois de ativado o cartão, o pinpad efetuará os seguintes processamentos da norma **EMV#3**:

- *Application Selection*;
- *Initiate Application Processing*; e
- *Read Application Data*.

➔ Application Selection

Neste processo, o pinpad deve fornecer ao seu Kernel EMV a lista de AIDs dos “registros candidatos”, de forma que este possa efetuar o processamento da seleção, utilizando o conceito de “*partial match*” (ver **EMV#1**). Caso o Kernel EMV necessite do *Additional Terminal Capabilities* (tag 9F40h), não há como utilizar o valor de **T1_ADDTRMCP**, dado que ainda não se sabe qual aplicação será utilizada. Desta forma, o pinpad deve usar um valor qualquer (aceito pelo Kernel) que tenha o bit “*Code table 1*” ativo no 5º byte.

Caso o número de “registros candidatos” supere a capacidade definida por esta especificação (conforme **seção 6.1.2**), o pinpad deve retornar ↵ST_ERRMAXAID.

Caso exista mais de uma aplicação compatível no cartão, ou caso a aplicação (se única) exija confirmação do portador, o pinpad deverá apresentar um menu de seleção contendo as etiquetas das aplicações (*Application Label* ou *Application Preferred Name*, se existente e o *Issuer Code Table Index* for 01h), com o título “**SELECIONE:**” O *layout* do menu é livre de forma a usar melhor os recursos de cada equipamento, lembrando sempre que as etiquetas podem ter até 16 caracteres.

- Enquanto o pinpad aguarda a seleção:
 - ⇒ O comando pode ser cancelado pelo operador através da tecla [CANCELA]. Neste caso o *display* deve ser limpo e o comando finalizado com ↵ST_CANCEL.
 - ⇒ O comando pode ser cancelado pelo SPE através do envio de um byte «**CAN**». (caso em que o pinpad responde com «**EOT**»). Neste caso o *display* deve ser limpo e o comando finalizado.
- Durante o processamento do menu, o pinpad sempre deverá enviar uma mensagem de notificação ao SPE informado qual opção está “ativa” (em destaque), enviando-a novamente caso o portador mude a seleção. A mensagem de notificação deve possuir o seguinte formato, sendo que o dado “XXX...X” representa a etiqueta apresentada no menu:

| NTM_MSG [1..16] | NTM_MSG [17..32] |
|-----------------|----------------------|
| SELECIONADO: | XXXXXXXXXXXXXXXXXXXX |

Caso o cartão possua somente uma aplicação compatível e esta é selecionada automaticamente pelo Kernel EMV, a mesma mensagem de notificação deve ser enviada.

Ao final do processo, a mensagem “SELECIONADO: XXX..X” deve ser deixada no *display*, também com *layout* livre.

➔ Initiate Application Processing / Read Application Data

Determinada a aplicação a ser utilizada e, conseqüentemente, o registro correspondente na Tabela de AIDs, o processamento do Kernel EMV deve prosseguir utilizando os seguintes parâmetros (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---|-------|--|
| <i>Amount, Authorized (binary)</i> | 81h | <u>GCR_AMOUNT</u> (ver seção 6.8.6.2) |
| <i>Amount, Authorized (numeric)</i> | 9F02h | <u>GCR_AMOUNT</u> |
| <i>Transaction Date</i> | 9Ah | <u>GCR_DATE</u> |
| <i>Transaction Time</i> | 9F21h | <u>GCR_TIME</u> |
| <i>Application Version Number</i> | 9F09h | <u>T1_APPVER1</u> |
| <i>Terminal Country Code</i> | 9F1Ah | <u>T1_TRMCNTRY</u> |
| <i>Transaction Currency Code</i> | 5F2Ah | <u>T1_TRNCURR</u> |
| <i>Transaction Currency Exponent</i> | 5F36h | <u>T1_TRNCRREXP</u> |
| <i>Merchant Identifier</i> | 9F16h | <u>T1_MERCHID</u> |
| <i>Merchant Category Code</i> | 9F15h | <u>T1_MCC</u> |
| <i>Terminal Identification</i> | 9F1Ch | <u>T1_TRMID</u> |
| <i>Terminal Capabilities</i> | 9F33h | <u>T1_TRMCAPAB</u> |
| <i>Additional Terminal Capabilities</i> | 9F40h | <u>T1_ADDTRMCP</u> |
| <i>Terminal Type</i> | 9F35h | <u>T1_TRMTYP</u> |
| <i>Terminal Floor Limit</i> | 9F1Bh | <u>T1_FLRLIMIT</u> |
| <i>Transaction Category Code</i> | 9F53h | <u>T1_TCC</u> |
| <i>Transaction Sequence Counter</i> | 9F41h | Contador <u>regido internamente pelo pinpad</u> . |
| <i>Transaction Type</i> | 9Ch | 00h (não há como saber se haverá <i>cashback</i> neste momento). |

Caso o processo EMV indique a exclusão da aplicação selecionada da lista de candidatas e, havendo mais de uma (ou seja, foi apresentado um menu), o pinpad deve:

- Apresentar a mensagem “APLICAÇÃO INVÁLIDA”, devidamente formatada para o *display*.
- Enviar a seguinte mensagem de notificação ao SPE:

| <u>NTM_MSG</u> [1..16] | <u>NTM_MSG</u> [17..32] |
|------------------------|-------------------------|
| APLICACAO | INVALIDA |

- Aguardar 1,5s (um segundo e meio) e retornar o processamento à etapa “*Application Selection*”.

➔ Dados de Saída

Caso o ICC seja processado com sucesso, a resposta ao comando “**GCR**” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|---|---|
| <u>GCR_CARDTYPE</u> | Tipo de cartão lido: “03” = ICC EMV. |
| <u>GCR_STATCHIP</u> | “0” (sucesso). |
| <u>GCR_APPTYPE</u> | <u>T1_APPTYPE</u> |
| <u>GCR_ACQIDX</u> | <u>TAB_ACQ</u> (Tabela de AID) |
| <u>GCR_RECIDX</u> | <u>TAB_RECIDX</u> (Tabela de AID) |
| <u>GCR_TRK2LEN</u> <u>GCR_TRK2</u> | <i>Track 2 Equivalent Data (tag 57h)</i> , se existente no cartão. |
| <u>GCR_PANLEN</u> <u>GCR_PAN</u> | <i>Application PAN - Primary Account Number (tag 5Ah)</i> . |
| <u>GCR_PANSEQNO</u> | <i>PAN Sequence Number (tag 5F34h)</i> . Se ausente, retornar “00” . |
| <u>GCR_APPLABEL</u> | Mesma etiqueta da aplicação apresentada no processo de seleção, podendo ser o <i>Application Label (tag 50h)</i> ou <i>Application Preferred Name (tag 9F12h)</i> . |
| <u>GCR_SRVCODE</u> | <i>Service Code (tag 5F30h)</i> , se existente no cartão. |
| <u>GCR_CHNAME</u> | <i>Cardholder Name (tag 5F20h)</i> , com espaços à direita. |
| <u>GCR_CARDEXP</u> | <i>Application Expiration Date (tag 5F24h)</i> , se existente no cartão. |
| <u>GCR_ISSCNTRY</u> | <i>Issuer Country Code (tag 5F28h)</i> , se existente no cartão. |
| <u>GCR_ACQRDLEN</u> <u>GCR_ACQRD</u> | Ver definição na seção 3.6.1 . |
| Outro... | Retornar valor zerado (“000...”) para os campos numéricos e espaços em branco para os alfanuméricos. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|---------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_CANCEL | A tecla [CANCELAR] foi pressionada pelo portador enquanto o pinpad aguardava um cartão ou durante o menu de seleção de aplicação. |
| ↪ST_TIMEOUT | Atingido tempo de ociosidade de 2 minutos caso CTLS seja requerido. |
| ↪ST_TABVERDIF | Versão das Tabelas EMV difere da esperada. |

| RSP_STAT | Situação |
|-------------------|--|
| ↪ST_NOCARD | ICC foi removido durante a apresentação do menu de seleção. |
| ↪ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↪ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↪ST_CARDINVALIDAT | ▪ A <u>única aplicação compatível</u> no cartão está invalidada (SELECT retornou SW1/SW2 = 6283h). |
| ↪ST_CARDBLOCKED | Comando SELECT retorna erro SW1/SW2=6A81h (cartão foi bloqueado). |
| ↪ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↪ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ↪ST_CARDAPPAV | O ICC não possui nenhuma aplicação compatível para o processamento. |
| ↪ST_CARDAPPNAUT | ▪ A <u>única aplicação compatível</u> no cartão retornou erro SW1/SW2 = 6985h no comando GET PROCESSING OPTIONS. ▪ A <u>única aplicação compatível</u> no cartão retornou erro no comando SELECT final. |
| ↪ST_ERRFALLBACK | O cartão reportou no GET PROCESSING OPTIONS um status (SW1/SW2) cujo comportamento não é regido pela norma EMV. |
| ↪ST_ERRMAXAID | Número de AIDs candidatos supera a capacidade de tratamento do Kernel EMV. |

6.8.1.3. Cartão com *chip* sem contato (CTLS)

Caso seja apresentado um CTLS ao pinpad, este deverá filtrar os “registros candidatos” de forma a fornecer ao Kernel EMV CTLS uma lista contendo somente os AIDs provenientes dos registros que cumpram os seguintes requisitos:

- O campo T1_CTLSMODE deve ter valor válido entre “1” e “9”;
- Se GCR_AMOUNT estiver zerado, o campo T1_CTLSZEROAM deve ser igual a “1”;
- O campo T1_CTLSTRNLIM ou T1_CTLSMBTLIM deve ter valor maior ou igual a GCR_AMOUNT.

Caso o número de “registros candidatos” supere a capacidade definida por esta especificação (conforme [seção 6.1.2](#)), o pinpad deve retornar ↪ST_ERRMAXAID.

▲ Caso o cartão contenha mais de uma aplicação compatível, a aplicação de maior prioridade será selecionada automaticamente.

Ao final do processo de seleção, uma mensagem de notificação deve ser enviada ao SPE no seguinte formato, sendo que o dado “XXX...X” representa a etiqueta da aplicação (*Application Label* ou *Application Preferred Name*, se existente e o *Issuer Code Table Index* for 01h):

| | |
|------------------------|-------------------------|
| NTM_MSG [1..16] | NTM_MSG [17..32] |
| SELECIONADO: | XXXXXXXXXXXXXXXXXX |

A mensagem “**SELECIONADO: XXX..X**” deve ser deixada no *display*, com *layout* livre de acordo com as capacidades do equipamento.

Identificada a aplicação a ser usada, deve-se verificar o valor de **T1_CTLSMODE**, de forma a efetuar os processamentos específicos de cada “bandeira” (Visa, Mastercard, American Express, Discover ou [Union Pay](#)), conforme descrito em [VCPS](#), [PPMChip](#), [ExpPay](#), [D-PAS](#) ou [QckPass](#).

Pelas características do CTLS, todo o processamento é feito já na função “**GCR**”, em um único “toque”. Caso a transação requiera verificação do portador (PIN *online*), isso deverá ser feito em “**GOC**”.

A tabela a seguir lista os objetos genéricos que devem ser fornecidos ao Kernel EMV CTLS (com as devidas conversões de formato), independentemente do valor de **T1_CTLSMODE**:

| Objeto EMV | Tag | Origem (Id. do Campo) |
|--|------------|--|
| <i>Amount, Authorized (binary)</i> | 81h | GCR_AMOUNT (ver seção 6.8.6.2) |
| <i>Amount, Authorized (numeric)</i> | 9F02h | GCR_AMOUNT |
| <i>Transaction Date</i> | 9Ah | GCR_DATE |
| <i>Transaction Time</i> | 9F21h | GCR_TIME |
| <i>Application Version Number</i> | 9F09h | T1_APPVER1 |
| <i>Terminal Country Code</i> | 9F1Ah | T1_TRMCNTRY |
| <i>Transaction Currency Code</i> | 5F2Ah | T1_TRNCURR |
| <i>Transaction Currency Exponent</i> | 5F36h | T1_TRNCRREXP |
| <i>Merchant Identifier</i> | 9F16h | T1_MERCHID |
| <i>Merchant Category Code</i> | 9F15h | T1_MCC |
| <i>Terminal Identification</i> | 9F1Ch | T1_TRMID |
| <i>Terminal Capabilities</i> | 9F33h | T1_CTLSTRMCP |
| <i>Additional Terminal Capabilities</i> | 9F40h | T1_CTLSADDTC |
| <i>Terminal Type</i> | 9F35h | T1_TRMTYP |
| <i>Terminal/Reader Contactless Transaction Limit</i> | | T1_CTLSTRNLIM |
| <i>Terminal/Reader Contactless Floor Limit</i> | | T1_CTLSFLRLIM |
| <i>Terminal/Reader CVM Required Limit</i> | | T1_CTLSCVMLIM |
| <i>Contactless Term. Action Code – Default</i> | | T1_CTLSTACDEF |
| <i>Contactless Term. Action Code – Denial</i> | | T1_CTLSTACDEN |

| Objeto EMV | Tag | Origem (Id. do Campo) |
|--|-----|--|
| Contactless Term. Action Code – Online | | T1_CTLSTACONL |
| Transaction Type | 9Ch | 00h (não há como saber se haverá <i>cashback</i> neste momento). |

▲ Durante o processamento do CLTS, o pinpad deve apresentar os indicadores visuais (LEDs) e sonoros (*beeps*) conforme definido em **EMV#CtIsA** (Capítulo 9).

➡ Parâmetros específicos - Visa payWave

Se **Visa payWave** (**T1_CTLSMODE** = “1” ou “2”), os seguintes parâmetros adicionais devem ser fornecidos ao Kernel EMV CTLS, segundo **VCPS**:

| Objeto | Tag | Valor |
|---------------------------------------|-------|---|
| Terminal Transaction Qualifiers (TTQ) | 9F66h | Ver configuração de alguns bits relevantes na tabela a seguir (outros bits podem ser preenchidos pelo próprio Kernel). |
| Zero Amount Allowed Flag | -- | T1_CTLSZEROAM |

Terminal Transaction Qualifiers:

| Byte | Bit | Descrição | Valor |
|-------|-----|--|---|
| 1 | 8 | MSD Supported | 0 (fixo) |
| | 7 | RFU | 0 (fixo) |
| | 6 | qVSDC supported | 1 (fixo) |
| | 5 | EMV contact chip supported | Bit 6 do 1º byte de T1_TRMCPAB |
| | 4 | Offline-only reader | Ativar se T1_TRMTYP = “x3” ou “x6” |
| | 3 | Online PIN supported | Bit 7 do 2º byte de T1_CTLSTRMCP |
| | 2 | Signature supported | Bit 6 do 2º byte de T1_CTLSTRMCP |
| 2 | 1 | Offline Data Authentication for Online Authorizations supported. | 0 (fixo) |
| | 8 | Online cryptogram required | Preenchido pelo Kernel EMV CTLS. |
| | 7 | CVM required | Preenchido pelo Kernel EMV CTLS. |
| | 6 | (Contact Chip) Offline PIN supported | 1 se o bit 8 ou o bit 5 do 2º byte de T1_TRMCPAB estiverem ativos. |
| 5 a 1 | RFU | 00000 (fixo) | |

| Byte | Bit | Descrição | Valor |
|------|-------|---|--------------------------------------|
| 3 | 8 | <i>Issuer Update Processing supported</i> | Ativar se T1_CTLSSISSCR = "1" |
| | 7 | <i>Mobile functionality supported (Consumer Device CVM)</i> | 1 (fixo) |
| | 6 a 1 | <i>RFU</i> | 000000 (fixo) |
| 4 | 8 a 1 | <i>RFU</i> | 00000000 (fixo) |

➤ Parâmetros específicos - Mastercard PayPass

Se **Mastercard PayPass** (**T1_CTLSMODE** = "3" ou "4"), os seguintes parâmetros adicionais devem ser fornecidos ao Kernel EMV CTLS, segundo  **PPMChip**:

| Objeto | Tag | Valor |
|---|------------------|--|
| <i>Card Data Input Capability</i> | DF8117h | Primeiro byte de T1_CTLSTRMCP . |
| <i>CVM Capability – No CVM Required</i> | DF8119h | Fixo 08h (somente o "No CVM required" é habilitado). |
| <i>CVM Capability – CVM Required</i> | DF8118h | Segundo byte de T1_CTLSTRMCP , desabilitando-se os bits: b8 - Plaintext PIN for ICC verification; b6 ^(*) - Signature (paper); b5 - Enciphered PIN for offline verification; b4 = No CVM required. (*) Desabilitar se débito (Maestro), pois este não deve suportar assinatura em papel. |
| <i>Security Capability</i> | DF811Fh | Terceiro byte de T1_CTLSTRMCP . |
| <i>Kernel Configuration</i> | DF811Bh | Ver tabela a seguir . |
| <i>PayPass Mag Stripe App. Version Number</i> | 9F6Dh | T1_CTLSAPPVER |
| <i>Reader Contactless Transaction Limit (On-device CVM)</i> | DF8125h | T1_CTLSMBTLIM |
| <i>Terminal Risk Management Data</i> | 9F1Dh | Ver tabela a seguir . |
| <i>Transaction Category Code</i> | 9F53h | T1_TCC |

Kernel Configuration:

| Byte | Bit | Descrição | Valor |
|------|-----|--|----------|
| 1 | 8 | <i>Only EMV mode transactions supported</i> | 1 (fixo) |
| | 7 | <i>Only mag-stripe mode transactions supported</i> | 0 (fixo) |

| Byte | Bit | Descrição | Valor |
|------|-----|--|---|
| | 6 | <i>On device cardholder verification supported</i> | Ativar se <u>T1_MOBCVM</u> = "1" 1 (fixo) |
| | 5 | <i>Relay resistance protocol (RRP) supported</i> | 1 (fixo) |

Terminal Risk Management Data:

| Byte | Bit | Descrição | Valor |
|------|-----|---|--|
| 1 | 7 | <i>Enciphered PIN verified online (Contactless)</i> | Bit 7 de <i>CVM Capability</i> – <i>CVM Required (tag DF8118h)</i> ; |
| | 6 | <i>Signature (paper) (Contactless)</i> | Bit 6 de <i>CVM Capability</i> – <i>CVM Required (tag DF8118h)</i> ; |
| | 4 | <i>No CVM required (Contactless)</i> | 1 (fixo) |
| | 3 | <i>CDCVM (Contactless)</i> | Ativar se <u>T1_MOBCVM</u> = "1" 1 (fixo). |
| 2 | 7 | <i>Enciphered PIN verified online (Contact)</i> | Bit 7 do 2º byte de <u>T1_TRMCAPAB</u> |
| | 6 | <i>Signature (paper) (Contact)</i> | Bit 6 do 2º byte de <u>T1_TRMCAPAB</u> |
| | 5 | <i>Enciphered PIN verification performed by ICC (Contact)</i> | Bit 5 do 2º byte de <u>T1_TRMCAPAB</u> |
| | 4 | <i>No CVM required (Contact)</i> | Bit 4 do 2º byte de <u>T1_TRMCAPAB</u> |
| | 2 | <i>Plaintext PIN verification performed by ICC (Contact)</i> | Bit 8 do 2º byte de <u>T1_TRMCAPAB</u> |
| 3 | 8 | <i>Mag-Stripe mode contactless transactions not supported</i> | 1 (fixo) |
| | 7 | <i>EMV mode contactless transactions not supported</i> | 0 (fixo) |

➤ Parâmetros específicos - Amex ExpressPay

Se **Amex ExpressPay** (T1_CTLSMODE = "5" ou "6"), os seguintes parâmetros adicionais devem ser fornecidos ao Kernel EMV CTLS, de acordo com  **ExpPay**:

| Objeto | Tag | Valor |
|---|-------|---|
| <i>Contactless Reader Capabilities</i> | 9F6Dh | C0h (fixo). |
| <i>Enhanced Contactless Reader Capabilities</i> | 9F6Eh | Ver configuração de alguns bits relevantes na tabela a seguir (outros bits podem ser preenchidos pelo próprio Kernel). |
| <i>Zero Amount Allowed Flag</i> | -- | <u>T1_CTLSZEROAM</u> |

Enhanced Contactless Reader Capabilities:

| Byte | Bit | Descrição | Valor |
|------|-----|--|---|
| 1 | 8 | Contact mode supported | 1 (fixo) |
| | 7 | Contactless Mag-Stripe Mode supported | 1 (fixo) |
| | 6 | Contactless EMV full online mode not supported | 0 (fixo) |
| | 5 | Contactless EMV partial online mode supported | 1 (fixo) |
| | 4 | Contactless Mobile Supported | 1 (fixo) |
| | 3 | Try Another Interface after a decline. | 0 (fixo) |
| | 2 | RFU | 0 (fixo) |
| | 1 | RFU | 0 (fixo) |
| 2 | 8 | Mobile CVM supported | Ativar se T1_MOBCVM = "1" 1 (fixo). |
| | 7 | Online PIN supported | Bit 7 do 2º byte de T1_CTLSTRMCP |
| | 6 | Signature | Bit 6 do 2º byte de T1_CTLSTRMCP |
| | 5 | Plaintext Offline PIN | 0 (fixo) |
| 3 | -- | -- | 00h (a ser preenchido pelo kernel) |
| 4 | -- | -- | 00h (a ser preenchido pelo kernel) |

~~Parâmetros específicos – Pure Contactless~~

Se ~~Pure Contactless (T1_CTLSMODE = "7")~~, os seguintes parâmetros adicionais devem ser fornecidos ao Kernel EMV-CTLS, de acordo com ~~EMV Pure~~:

| Objeto | Tag | Valor |
|--|----------------|---|
| Contactless POS Implementation Options | = | 00h (fixo) |
| Additional Tag Object List (ATOL) | = | 9Fh 02h 9Fh 03h 9Fh 26h 82h 9Fh 36h 9Fh 27h 9Fh 10h 9Fh 1Ah 95h 5Fh 2Ah 9Ah 9Ch 9Fh 37h 9Fh 35h 57h 9Fh 34h 84h 5Fh 34h 5Ah 9Fh 1Fh 5Fh 20h 9Fh 77h (fixo) |
| Mandatory Tag Object List (MTOL) | = | 8Ch 57h (fixo) |
| Contactless Application/Kernel Capabilities | = | 36h 00h 40h 03h F9h (fixo) |

➔ Parâmetros específicos - Discover D-PAS

Se **Discover D-PAS** (**T1_CTLSMODE** = "8" ou "9"), os seguintes parâmetros adicionais devem ser fornecidos ao Kernel EMV CTLS, de acordo com **D-PAS**:

| Objeto | Tag | Valor |
|--|-------|---|
| <i>Terminal Transaction Qualifiers (TTQ)</i> | 9F66h | Ver configuração de alguns bits relevantes na tabela a seguir (outros bits podem ser preenchidos pelo próprio Kernel). |
| <i>Zero Amount Allowed Flag</i> | -- | T1_CTLSZEROAM |

Terminal Transaction Qualifiers:

| Byte | Bit | Descrição | Valor |
|------|-------|---|--|
| 1 | 8 | <i>Mag stripe mode supported</i> | 0 (fixo) |
| | 7 | <i>RFU</i> | 0 (fixo) |
| | 6 | <i>EMV mode supported</i> | 1 (fixo) |
| | 5 | <i>EMV contact chip supported</i> | Bit 6 do 1º byte de T1_TRMCAPAB |
| | 4 | <i>Offline-only Contactless Reader</i> | Ativar se T1_TRMTYP = "x3" ou "x6" |
| | 3 | <i>Online PIN supported</i> | Bit 7 do 2º byte de T1_CTLSTRMCP |
| | 2 | <i>Signature supported</i> | Bit 6 do 2º byte de T1_CTLSTRMCP |
| | 1 | <i>Offline Data Authentication for Online Authorizations supported.</i> | 0 (fixo) |
| 2 | 8 | <i>Online cryptogram required</i> | Preenchido pelo Kernel EMV CTLS. |
| | 7 | <i>CVM required</i> | Preenchido pelo Kernel EMV CTLS. |
| | 6 | <i>(Contact Chip) Offline PIN supported</i> | 1 se o bit 8 ou o bit 5 do 2º byte de T1_TRMCAPAB estiverem ativos. |
| | 5 | <i>RFU</i> | 0 (fixo) |
| | 4 | <i>Fast Mode Supported</i> | 0 (fixo) |
| | 3 | <i>Transit Terminal</i> | 0 (fixo) |
| | 2 a 1 | <i>RFU</i> | 00 (fixo) |
| 3 | 8 | <i>Issuer Update Processing supported</i> | Ativar se T1_CTLSISSSCR = "1" |
| | 7 | <i>Consumer Device CVM supported</i> | 1 (fixo) |
| | 6 | <i>Consumer Device CVM for Transit MCC</i> | 0 (fixo) |
| | 5 a 1 | <i>RFU</i> | 00000 (fixo) |
| 4 | 8 a 1 | <i>RFU</i> | 00000000 (fixo) |

➔ Parâmetros específicos – Union Pay Quick Pass

Se **Union Pay Quick Pass** (**T1_CTLSMODE** = “B”), os seguintes parâmetros adicionais devem ser fornecidos ao Kernel EMV CTLS, de acordo com **QckPass**:

| Objeto | Tag | Valor |
|--|-------|---|
| <i>Terminal Transaction Qualifiers (TTQ)</i> | 9F66h | Ver configuração de alguns bits relevantes na tabela a seguir (outros bits podem ser preenchidos pelo próprio Kernel). |

Terminal Transaction Qualifiers:

| Byte | Bit | Descrição | Valor |
|------|-------|---|---|
| 1 | 8 | <i>RFU</i> | 0 (fixo) |
| | 7 | <i>Full transaction flow in Contactless interface Support</i> | 0 (fixo) |
| | 6 | <i>EMV mode supported</i> | 1 (fixo) |
| | 5 | <i>Full transaction flow in contact interface Support</i> | Bit 6 do 1º byte de T1_TRMCAPAB |
| | 4 | <i>Offline-only terminal</i> | Ativar se T1_TRMTYP = “x3” ou “x6” |
| | 3 | <i>Online PIN Supported</i> | Bit 7 do 2º byte de T1_CTLSTRMCP |
| | 2 | <i>Signature supported</i> | Bit 6 do 2º byte de T1_CTLSTRMCP |
| | 1 | <i>Offline Data Authentication for Online Authorizations support.</i> | 0 (fixo) |
| 2 | 8 | <i>Online cryptogram required</i> | Preenchido pelo Kernel EMV CTLS. |
| | 7 | <i>CVM required</i> | Preenchido pelo Kernel EMV CTLS. |
| | 6 a 1 | <i>RFU</i> | 000000 (fixo) |
| 3 | 8 | <i>RFU</i> | 0 (fixo) |
| | 7 | <i>Consumer Device CVM supported</i> | 1 (fixo) |
| | 6 a 1 | <i>RFU</i> | 000000 (fixo) |
| 4 | 8 | <i>fDDA v1.0 Supported</i> | 1 (fixo) |
| | 7 a 1 | <i>RFU</i> | 0000000 (fixo) |

➔ Offline Data Authentication

Para o processo de autenticação *offline*, o pinpad deve fornecer ao Kernel EMV CTLS, antes do processamento, determinadas chaves públicas de Autoridade Certificadora disponíveis em suas Tabelas de CAPK. Entretanto, como as Tabelas de CAPK **podem** ser “aglutinadas” pelo SPE conforme descrito na **seção 4.1.2**, esse processo pode seguir duas lógicas distintas.

Caso o SPE tenha aglutinado chaves em uma tabela com **TAB_ACQ** = “00”, deve-se adotar o seguinte procedimento:

- Partindo-se do princípio que o SPE já fez corretamente o tratamento descrito na **seção 4.1.2**, deve-se utilizar somente as chaves da tabela aglutinada (**TAB_ACQ = "00"**), desprezando eventuais outras tabelas (**TAB_ACQ ≠ "00"**).

Caso o SPE **não** tenha aglutinado chaves, deve-se adotar o seguinte procedimento:

- Utilizar somente chaves das redes credenciadoras que geraram "registros candidatos".
- Dependendo das características do Kernel EMV CTLS, o pinpad deverá eliminar eventuais duplicidades (chaves com mesmo **T2_RID** e **T2_CAPKIDX**), porém as regras para isso não estão contempladas nesta especificação.

➔ Dados de Saída (CTLS simulando tarja)

Caso o CTLS seja processado com sucesso na modalidade ~~PayPass Mag Stripe, Expresspay Magstripe Mode~~ ou ~~D-PAS MS Mode~~ *magstripe* (simulação de tarja magnética), a resposta ao comando "GCR" deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|---------------------------------------|---|
| GCR_CARDTYPE | Tipo de cartão lido: "05" = CTLS simulando tarja. |
| GCR_STATCHIP | "0" (sucesso). |
| GCR_APPTYPE | T1_APPTYPE |
| GCR_ACQIDX | TAB_ACQ (Tabela de AID) |
| GCR_RECIDX | TAB_RECIDX (Tabela de AID) |
| GCR_TRK1LEN GCR_TRK1 | Dados da trilha 1, montados de acordo com a especificação da "bandeira". |
| GCR_TRK2LEN GCR_TRK2 | Dados da trilha 2, montados de acordo com a especificação da "bandeira". |
| GCR_APPLABEL | Mesma etiqueta da aplicação apresentada no processo de seleção, podendo ser o <i>Application Label (tag 50h)</i> ou <i>Application Preferred Name (tag 9F12h)</i> . |
| Outro... | Retornar valor zerado ("000...") para os campos numéricos e espaços em branco para os alfanuméricos. |

➔ Dados de Saída (CTLS EMV)

Caso o CTLS seja processado com sucesso na modalidade ~~eVSDC, PayPass M/Chip, Expresspay EMV Mode, D-PAS EMV Mode~~ *EMV full* (não simulando tarja magnética) a resposta ao comando "GCR" deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|---------------------|---------------------------------------|
| GCR_CARDTYPE | Tipo de cartão lido: "06" = CTLS EMV. |
| GCR_STATCHIP | "0" (sucesso). |
| GCR_APPTYPE | T1_APPTYPE |
| GCR_ACQIDX | TAB_ACQ (Tabela de AID) |

| Id. do Campo | Dado retornado |
|--|---|
| <u>GCR_RECIDX</u> | <u>TAB_RECIDX</u> (Tabela de AID) |
| <u>GCR_TRK2LEN</u> <u>GCR_TRK2</u> | Dados da trilha 2, montados de acordo com a especificação da “bandeira”. |
| <u>GCR_PANLEN</u> <u>GCR_PAN</u> | <i>Application PAN - Primary Account Number (tag 5Ah)</i> . Caso este objeto não exista no cartão, extraí-lo de <u>GCR_TRK2</u> . |
| <u>GCR_PANSEQNO</u> | <i>PAN Sequence Number (tag 5F34h)</i> . Se ausente, retornar “00”. |
| <u>GCR_APPLABEL</u> | Mesma etiqueta da aplicação apresentada no processo de seleção, podendo ser o <i>Application Label (tag 50h)</i> ou <i>Application Preferred Name (tag 9F12h)</i> . |
| <u>GCR_SRVCODE</u> | <i>Service Code (tag 5F30h)</i> , se existente no cartão. |
| <u>GCR_CHNAME</u> | <i>Cardholder Name (tag 5F20h)</i>, com espaços à direita |
| <u>GCR_CARDEXP</u> | <i>Application Expiration Date (tag 5F24h)</i> , se existente no cartão. |
| <u>GCR_ISSCNTRY</u> | <i>Issuer Country Code (tag 5F28h)</i> , se existente no cartão. |
| <u>GCR_ACQRLEN</u> <u>GCR_ACQRD</u> | Ver definição na seção 3.6.1 . |
| Outro... | Retornar valor zerado (“000...”) para os campos numéricos e espaços em branco para os alfanuméricos. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|--------------------------------|--|
| ↳ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↳ST_TIMEOUT | Atingido tempo de ociosidade de 2 minutos ao aguardar o cartão. |
| ↳ST_CTLSMULTIPLE | Mais de um CTLS foi apresentado ao leitor simultaneamente. |
| ↳ST_CTLSCOMMERR | <ul style="list-style-type: none"> ▪ Erro de comunicação entre o pinpad (antena) e o CTLS. • O Kernel CTLS solicitou a verificação do portador no dispositivo móvel (Outcome = “Try Again”) e <u>T1_MOBCVM</u> = “1”.³ |
| ↳ST_CARDBLOCKED | Comando SELECT retorna erro SW1/SW2=6A81h (cartão foi bloqueado). |
| ↳ST_CTLSINVALIDAT ✘ | Comando SELECT retorna erro SW1/SW2=6A81h ou 6283h. |
| ↳ST_CTLSPROBLEMS | CTLS com problemas. Esse status é válido para muitas ocorrências no processamento onde o CTLS não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↳ST_CTLSAPPNAV ✘ | O CTLS não possui nenhuma aplicação compatível para o processamento. |

³ Isto é feito para se manter compatibilidade com um SPE antigo que não conhece ~~↳ST_CTLSXTCVM~~.

| RSP_STAT | Situação |
|----------------------------|--|
| ↳ST_CTLSPNAUT ✘ | O cartão retornou erro SW1/SW2 = 6985h no comando GET PROCESSING OPTIONS. |
| ↳ST_ERRMAXAID | Número de AIDs candidatos supera a capacidade de tratamento do Kernel EMV. |
| ↳ST_CTLSEXTCVM | O Kernel CTLS solicitou a verificação do portador no dispositivo móvel (<i>Outcome</i> = "Try Again") e T1_MOBCVM = "1" . |
| ↳ST_CTLSIFCHG | <ul style="list-style-type: none"> ▪ Kernel CTLS solicitou "mudança de interface" para processamento usando ICC ou cartão magnético (<i>Outcome</i> = "Try Another Interface"). ▪ Se GCR_AMOUNT ≥ T1_CTLSTRNLIM e cartão Discover D-PAS ou Visa payWave. |

- ✘ De acordo com [EMV#CtlsB](#) seções 3.3.2.6 até 3.3.3.5, qualquer erro no comando SELECT faz com que o AID em questão seja retirado da lista de candidatos e, se a lista estiver vazia, deve-se solicitar mudança de interface. Assim sendo, estas situações acabam acarretando ~~↳ST_CTLSIFCHG~~.

6.8.2. Comando “GCR” (vazio)

Caso o pinpad receba o “GCR” sem parâmetros, ele deve recuperar os parâmetros armazenados na última chamada a “GCR” que retornou ↵ST_TABVERDIF e executar o comando normalmente, conforme **seção 6.8.1**.

Se a última chamada a “GCR” não retornou ↵ST_TABVERDIF, esta chamada deverá retornar ↵ST_INVCALL.

6.8.3. Comando “CNG”

Ao processar este comando, pinpad deve armazenar os objetos TLV recebidos para uso no processamento de “GOC” e “FNC”, de acordo com as seguintes regras:

- Os objetos recebidos podem ser proprietários ou pertencentes à norma EMV. Caso existam objetos correspondentes aos da **Tabela de AID**, eles terão prioridade no processamento.
- Cabe ao SPE garantir a consistência dos dados enviados. No caso da recepção de objetos repetidos, por exemplo, o pinpad poderá usar qualquer um deles.
- O SPE pode enviar mais de um comando “CNG” após o processamento de “GCR”. Desta forma, o pinpad deve acumular os dados recebidos a cada comando, não os sobrepondo.
- A lista de objetos mantida por este comando deve ser limpa ao final do processamento de “FNC”, bem como no início do processamento do próximo “GCR”.
- O pinpad deve acatar qualquer objeto TLV recebido, sendo que o comando deve retornar ↵ST_INVPARAM somente se a estrutura TLV estiver corrompida.
- O pinpad não deve utilizar objetos que, pela norma, sabidamente são originados no cartão (ver **seção 6.8.6.4**).

➔ Situações de exceção:

| RSP_STAT | Situação |
|--------------|--|
| ↵ST_INVPARAM | Estrutura de dados recebida não segue corretamente as regras do BER-TLV (ver seção 7.1). |
| ↵ST_INVCALL | A chamada anterior de “GCR” não processou com sucesso um cartão ICC EMV. |

6.8.4. Comando “GOC”

Este comando continua o processamento de cartões ICC EMV (GCR_CARDTYPE = “03”) ou CTLS EMV (GCR_CARDTYPE = “06”).

Os tratamentos envolvidos são diferentes dependendo do tipo de *chip*, descritos a seguir.

➔ Observações:

- Ao final do processamento do comando, o pinpad deve apagar o *display* em caso de erro.

6.8.4.1. Cartão com *chip* de contato (ICC EMV)

O processamento de ICC EMV deve continuar a seguir os processamentos estipulados pela norma EMV:

- *Processing Restrictions*;
- *Offline Data Authentication*;
- *Cardholder Verification*;
- *Terminal Risk Management*;
- *Terminal Action Analysis*; e
- *Card Action Analysis*.

Para que o Kernel EMV possa continuar o processamento, os seguintes parâmetros devem ser fornecidos a ele, além dos que já foram estipulados no comando “GCR” (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|-------------------------------------|-------|---|
| <i>Amount, Authorized (binary)</i> | 81h | <u>GOC_AMOUNT</u> (ver seção 6.8.6.2) |
| <i>Amount, Authorized (numeric)</i> | 9F02h | <u>GOC_AMOUNT</u> |
| <i>Amount, Other (binary)</i> | 9F04h | <u>GOC_CASHBACK</u> (ver seção 6.8.6.2). |
| <i>Amount, Other (numeric)</i> | 9F03h | <u>GOC_CASHBACK</u> |
| <i>Transaction Type</i> | 9Ch | 09h → Se <u>GOC_CASHBACK</u> diferente de zero; ou 00h → Outras situações. |

➔ Processing Restrictions

Para esta etapa do processamento, devem-se usar os seguintes objetos de dados:

| Objeto EMV | Tag | Origem (Id. do Campo) |
|-----------------------------------|-------|---|
| <i>Application Version Number</i> | 9F09h | T1_APPVER1 , T1_APPVER2 ou T1_APPVER3 (o que coincidir com a versão do cartão, ou T1_APPVER1 se não houver coincidência). |

Além disso, o campo **GOC_EXCLIST** indica se o número do cartão está em uma *Exception List*.

➔ Offline Data Authentication

Para o processo de autenticação *offline*, o pinpad deve fornecer ao Kernel EMV a chave pública da Autoridade Certificadora mediante pesquisa de **T2_RID** e **T2_CAPKIDX** nas Tabelas de CAPK.

- Se **T2_CHKSTAT** = "1" e **T2_CHECKSUM** não for coerente, a autenticação deve simplesmente falhar.

As Tabelas de CAPK podem ser "aglutinadas" pelo SPE, conforme descrito na **seção 4.1.2**. Desta forma, deve-se adotar a seguinte regra:

- Primeiro pesquisar os registros da Tabela de CAPK em que **TAB_ACQ** = **GCR_ACQIDX**.
- Caso o registro não seja encontrado, pesquisar a tabela em que **TAB_ACQ** = "00".

Além disso, fornecer ao Kernel EMV os registros da Tabela de Certificados Revogados (ver **seção 4.1.3**) em que **TAB_ACQ** = **GCR_ACQIDX**.

Outros parâmetros que devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|--|-----|-----------------------|
| <i>Default Dynamic Data Authentication Data Object List (DDOL)</i> | --- | T1_DDOLDEF |

➔ Cardholder Verification

Caso a verificação de portador indique a necessidade de validação de PIN, deve-se seguir o detalhamento indicado na **seção 6.8.6.1**.

Para o caso de PIN *online*:

- Deve-se usar os parâmetros **GOC_METHOD**, **GOC_KEYIDX** e **GOC_WKENC**.
- No caso de problemas com a chave indicada, o pinpad deve abortar a operação com ↵ST_ERRKEY, não dando continuidade ao processamento EMV.
- O valor do PAN usado no cálculo do PIN criptografado deve ser obtido diretamente do Kernel EMV, sendo que sua existência não é afetada por um eventual uso prévio do comando "**GTK**".

Se **GCR_ACQIDX** = "01" e **GOC_ACQPR** = "xx1", deve-se impedir o *bypass* do PIN, seja este online ou offline.

O pinpad deve finalizar a operação com ↵ST_TIMEOUT se o tempo de inatividade em uma tela de captura de PIN ultrapassar 1 minuto (60 segundos).

Enquanto o pinpad aguarda a digitação de um PIN, o comando pode ser cancelado pelo SPE através do envio de um byte «CAN».

➔ Terminal Risk Management

Esta etapa do processamento EMV sempre será efetuada, independentemente do valor de GOC_RISKMAN (obsoleto).

Para este processo, os seguintes parâmetros devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---|-------|-----------------------|
| <i>Terminal Floor Limit</i> (em centavos) | 9F1Bh | <u>GOC_FLRLIMIT</u> |
| <i>Target Percentage to be used for Biased Random Selection</i> | -- | <u>GOC_TPBR</u> |
| <i>Threshold Value for Biased Random Selection</i> (em centavos) | -- | <u>GOC_TVBR</u> |
| <i>Maximum Target Percentage to be used for Biased Random Selection</i> | -- | <u>GOC_MTPBR</u> |

➔ Terminal Action Analysis

Para este processo, os seguintes parâmetros devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---------------------------------------|-----|-----------------------|
| <i>Terminal Action Code – Default</i> | --- | <u>T1_TACDEF</u> |
| <i>Terminal Action Code – Denial</i> | --- | <u>T1_TACDEN</u> |
| <i>Terminal Action Code – Online</i> | --- | <u>T1_TACONL</u> |

⚠ Se GOC_CONNECT = "1", o Kernel EMV nunca poderá sugerir aprovação offline (TC) ao cartão!
Essa implementação não pode usar o bit "Merchant Forced Transaction Online" do TVR para impedir a aprovação offline, pois isso não é aceito nas certificações de bandeiras.

➔ Card Action Analysis.

Para este processo, os seguintes parâmetros devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|--|-----|-----------------------|
| <i>Default Transaction Certificate Data Object List (TDOL)</i> | --- | <u>T1_TDOLDEF</u> |

➔ Dados de Saída

Caso o ICC seja processado com sucesso, a resposta ao comando “**GOC**” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|--|--|
| GOC_DECISION | Decisão do cartão no comando 1st GENERATE AC: “0” → Cartão retornou TC (aprovada <i>offline</i>). “1” → Cartão retornou AAC (negada <i>offline</i>). “2” → Cartão retornou ARQC (requer autorização <i>online</i>). |
| GOC_SIGNAT | “1” → O <i>Cardholder Verification</i> decidiu pela obtenção de assinatura em papel. |
| GOC_PINOFF | “1” → O PIN <i>offline</i> foi verificado com sucesso no <i>Cardholder Verification</i> (comando VERIFY retornou SW1/SW2 = 9000h) |
| GOC_ERRPINOFF | Quantidade de vezes que o comando VERIFY retornou SW1/SW2 = 63C _x h nesta transação. Retornar “9” caso este valor seja ≥ 10, dado que o campo permite somente um dígito numérico. |
| GOC_PBLOCKED | “1” → O comando VERIFY retornou SW1/SW2 = 63C0h nesta transação. |
| GOC_PINONL | “1” → PIN capturado para verificação online no <i>Cardholder Verification</i> . |
| GOC_PINBLK | PIN criptografado para validação <i>online</i> (somente se GOC_PINONL = “1”). |
| GOC_KSN | KSN da chave DUKPT usada na criptografia de PIN <i>online</i> (somente se GOC_PINONL = “1” e GOC_METHOD = “3”). |
| GOC_EMVDTLEN GOC_EMVDAT | Dados definidos pela <i>tags</i> em GOC_TAGS1 e GOC_TAGS2 , no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3 . IMPORTANTE: <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando “GCR” não foi chamado previamente. ▪ Comando “GCR” foi chamado previamente, porém retornou GCR_CARDTYPE diferente de “03” e “06”. ▪ Comando “GOC” já foi chamado. |

| RSP_STAT | Situação |
|------------------|--|
| ↪ST_ERRKEY | Foi requerida captura de PIN <i>online</i> , mas a chave indicada está ausente ou corrompida. |
| ↪ST_CANCEL | A tecla [CANCELA] foi pressionada pelo portador durante a tela de captura de PIN. |
| ↪ST_TIMEOUT | Atingido tempo máximo de ociosidade (1 minuto sem ação do operador) na tela de captura de PIN (seja <i>online</i> ou <i>offline</i>). |
| ↪ST_NOCARD | ICC foi removido durante a tela de captura de PIN. |
| ↪ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↪ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↪ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↪ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ↪ST_ERRFALLBACK | O comando 1st GENERATE AC retornou SW1/SW2 diferente de 9000h. |
| ↪ST_INVAMOUNT | O cartão pediu informação de valor no formato "b4" e este supera a capacidade campo. |
| ↪ST_CARDAPPNAUT | O objeto <i>Cryptogram Information Data</i> (tag '9F27') retornado pelo cartão indica situação "Service not allowed". |
| ↪ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.8.4.2. Cartão com *chip* sem contato (CTLS EMV)

O cartão sem contato é processado inteiramente no comando "**GCR**". Desta forma, o valor da transação (**GOC_AMOUNT**) não pode ser alterado em "**GOC**".

Caso o processamento decida por pedir PIN *online*, isso é feito da mesma forma descrita na seção 6.8.4.1 (*Cardholder Verification*).

⚠ Caso a verificação de portador requisite PIN Online, o *bypass* não deve ser permitido para cartões CTLS (a tecla [OK/ENTRA] deve ser ignorada se o campo de entrada estiver vazio).

➡ Dados de Saída

Caso o processamento seja bem-sucedido, a resposta ao comando "**GOC**" deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|---------------------|--|
| GOC_DECISION | Decisão do processamento do cartão CTLS (já efetuado em " GCR "): <ul style="list-style-type: none"> "0" → Aprovada <i>offline</i>. "1" → Negada <i>offline</i>. "2" → Requer autorização <i>online</i>. |

| Id. do Campo | Dado retornado |
|--|--|
| <u>GOC_SIGNAT</u> | "1" → O <i>Cardholder Verification</i> decidiu pela obtenção de assinatura em papel. |
| <u>GOC_PINOFF</u> | "0" (sempre) |
| <u>GOC_ERRPINOFF</u> | "0" (sempre) |
| <u>GOC_PBLOCKED</u> | "0" (sempre) |
| <u>GOC_PINONL</u> | "1" → PIN capturado para verificação online. |
| <u>GOC_PINBLK</u> | PIN criptografado para validação <i>online</i> (somente se <u>GOC_PINONL</u> = "1"). |
| <u>GOC_KSN</u> | KSN da chave DUKPT usada na criptografia de PIN <i>online</i> (somente se <u>GOC_PINONL</u> = "1" e <u>GOC_METHOD</u> = "3"). |
| <u>GOC_EMVDTLEN</u> <u>GOC_EMVDAT</u> | Dados definidos pela <i>tags</i> em <u>GOC_TAGS1</u> e <u>GOC_TAGS2</u> , no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3 . IMPORTANTE: <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |

↻ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↵ST_INVPARM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ O valor da transação foi alterado. |
| ↵ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando "GCR" não foi executado previamente com sucesso. ▪ Comando "GCR" foi chamado previamente, porém retornou <u>GCR_CARDTYPE</u> diferente de "03" e "06". |
| ↵ST_ERRKEY | Foi requerida captura de PIN <i>online</i> , mas a chave indicada está ausente ou corrompida. |
| ↵ST_CANCEL | A tecla [CANCELA] foi pressionada pelo portador durante a tela de captura de PIN. |
| ↵ST_TIMEOUT | Atingido tempo máximo de ociosidade (1 minuto sem ação do operador) na tela de captura de PIN (seja <i>online</i> ou <i>offline</i>). |

6.8.5. Comando “FNC”

Este comando é chamado pelo SPE caso “GOC” tenha requerido aprovação *online* ou, opcionalmente, caso a transação já tenha sido aprovada ou negada *offline*. Os tratamentos envolvidos são diferentes dependendo do tipo de chip, descritos a seguir.

OBS: No caso de ICC, este comando deve sempre desligar sua alimentação ao final do processamento.

6.8.5.1. ICC EMV - encerrada *offline*

Caso a transação já tenha sido aprovada ou negada *offline* pelo cartão em “GOC”, o SPE pode acionar este comando apenas para compatibilização de fluxo.

➔ Dados de Saída

Neste caso, a resposta ao comando “FNC” deve simplesmente devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|--------------|---|
| FNC_DECISION | Mesmo valor de GOC_DECISION (“0” ou “1”). |
| FNC_EMVDTLEN | “000” |
| FNC_EMVDAT | (vazio) |
| FNC_ISRLLEN | “00” |
| FNC_ISR | (vazio) |

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando “GOC” não foi executado previamente com sucesso. ▪ Comando “FNC” já foi chamado. |
| ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.8.5.2. ICC EMV - impossibilidade de conexão *online*

Caso o comando receba FNC_COMMST com valor “1”, a conexão com a Rede Credenciadora não foi bem-sucedida e os campos FNC_ARC, FNC_ISSDATLEN e FNC_ISSDAT devem ser ignorados.

➔ Completion

Neste caso, o pinpad deve acionar o processo conhecido como *Unable to Go Online* no Kernel EMV, fornecendo o objeto *Terminal Action Code – Default (T1_TACDEF)*.

⚠ Se GOC_CONNECT = "1", o Kernel EMV deverá sugerir negação (AAC) ao cartão.

➔ Dados de Saída

A resposta ao comando "FNC" deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|--|--|
| <u>FNC_DECISION</u> | Decisão do cartão no comando 2nd GENERATE AC: "0" → Cartão retornou TC (aprovada <i>offline</i>). "1" → Cartão retornou AAC (negada <i>offline</i>). |
| <u>FNC_EMVDTLEN</u> <u>FNC_EMVDAT</u> | Dados definidos pela <i>tags</i> em <u>FNC_TAGS</u> , no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3 . IMPORTANTE: <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |
| <u>FNC_ISRLEN</u> | "00" |
| <u>FNC_ISR</u> | (vazio) |

➔ Situações de exceção:

| RSP_STAT | Situação |
|------------------|---|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando "<u>GOC</u>" não foi executado previamente com sucesso. ▪ Comando "<u>FNC</u>" já foi chamado. |
| ↪ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↪ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↪ST_NOCARD | O cartão foi removido. |
| ↪ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |

| RSP_STAT | Situação |
|----------------|---|
| ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.8.5.3. ICC EMV - autorização *online* bem-sucedida

Caso o comando receba **FNC_COMMST** com valor diferente de “1”, considera-se que a conexão com a Rede Credenciadora foi bem-sucedida e, portanto, uma resposta de autorização foi recebida.

➔ Issuer Authentication

Caso o campo **FNC_ISSDAT** contenha o objeto TLV de *tag* 91h (*Issuer Authentication Data*), este deve ser fornecido ao Kernel EMV para o processo de *Issuer Authentication*.

⚠ Caso o cartão suporte *Issuer Authentication* em seu *AIP* e, caso o comando EXTERNAL AUTHENTICATE retorne qualquer status SW1/SW2 de 9000h (por exemplo 6985h), o pinpad não deverá interromper a transação, seguindo com o processamento considerando falha na autenticação.

➔ Issuer Script Processing

Caso o campo **FNC_ISSDAT** contenha um ou mais objetos TLV de *tag* 71h e 72h (*Issuer Scripts*), estes devem ser fornecidos ao Kernel EMV para a execução do *Issuer Script Processing*.

➔ Completion

O Kernel EMV deverá solicitar a autorização final ao cartão de acordo com a decisão da Rede Credenciadora:

- Se **FNC_COMMST** = “0”, considera-se que a Rede Credenciadora aprovou a transação se **FNC_ARC** = “00”, sendo que qualquer outro valor de **FNC_ARC** indica a negação.
- Se **FNC_COMMST** > “1”, considera-se que a Rede Credenciadora aprovou a transação, independentemente do valor de **FNC_ARC**.

Em qualquer dos casos, o valor de **FNC_ARC** deve ser fornecido ao Kernel EMV como *Authorization Response Code* (*tag* 8Ah).

➔ Dados de Saída

A resposta ao comando “FNC” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|--|---|
| <u>FNC_DECISION</u> | Decisão do cartão no comando 2nd GENERATE AC: "0" → Cartão retornou TC (aprovada). "1" → Cartão retornou AAC (negada), porém Rede Credenciadora havia aprovado a transação. "2" → Cartão retornou AAC (negada), acatando a decisão da Rede Credenciadora. |
| <u>FNC_EMVDTLEN</u> <u>FNC_EMVDAT</u> | Dados definidos pela <i>tags</i> em <u>FNC_TAGS</u> , no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3 . IMPORTANTE: <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |
| <u>FNC_ISRLEN</u> <u>FNC_ISR</u> | Resultado do processamento de scripts (<i>Issuer Script Results</i>), se existente. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|------------------|---|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando "GOC" não foi executado previamente com sucesso. ▪ Comando "FNC" já foi chamado. |
| ↪ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↪ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↪ST_NOCARD | O cartão foi removido. |
| ↪ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↪ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ↪ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.8.5.4. Cartão com *chip* sem contato (CTLS EMV)

Este comando será acionado pelo SPE caso a transação tenha requerido aprovação *online*. Neste caso, o pinpad não faz mais nada, somente "rebatendo" a decisão informada no comando.

➔ Dados de Saída

Neste caso, a resposta ao comando “**FNC**” deve simplesmente devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|---------------------|---|
| FNC_DECISION | “0” → Se FNC_COMMST > “1”. “0” → Se FNC_COMMST = “0” e FNC_ARC = “00”. “1” → Se FNC_COMMST = “1”. “2” → Se FNC_COMMST = “0” e FNC_ARC ≠ “00” |
| FNC_EMVDTLEN | “000” |
| FNC_EMVDAT | (vazio) (*) |
| FNC_ISRLLEN | “00” |
| FNC_ISR | (vazio) |

(*) Caso o SPE passe uma lista de *tags* em **FNC_TAGS**, ela deve ser simplesmente ignorada.

➔ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando “GOC” não foi executado previamente com sucesso. ▪ Comando “FNC” já foi chamado. |

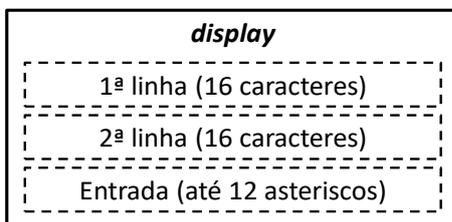
6.8.6. Regras gerais

6.8.6.1. Telas de captura de PIN

Durante o processamento de um cartão EMV, pode haver a necessidade de captura de PIN, seja *online* ou *offline*. Durante este processo, o pinpad poderá apresentar mensagens em três idiomas no *display*, de acordo com a configuração do cartão.

Para isso, o pinpad deve percorrer o objeto *Language Preference* (*tag* 5F2Dh) de forma a localizar o primeiro código ISO639-1 que corresponda aos suportados pelo pinpad: “**pt**” (português), “**en**” (inglês) e “**es**” (espanhol).

A tela de captura de PIN deve seguir o seguinte formato:



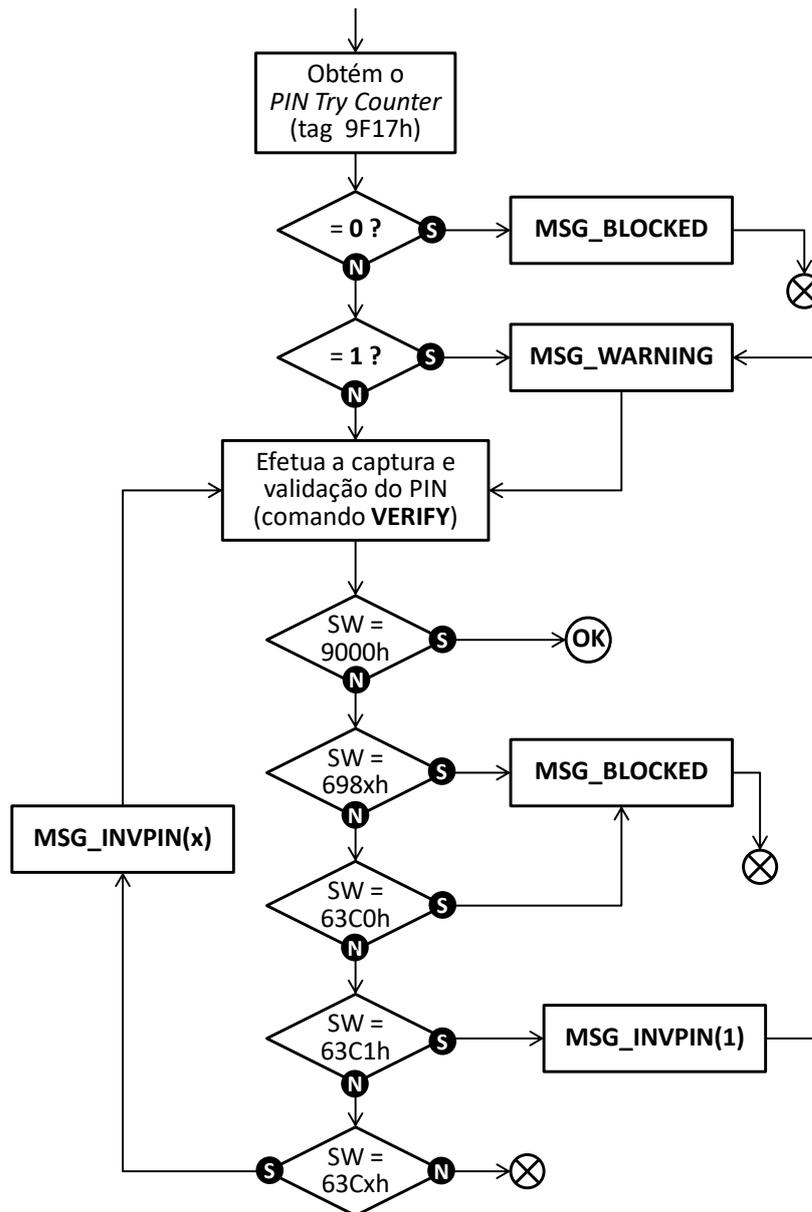
A tabela a seguir ilustra as mensagens a serem utilizadas para cada idioma, sempre com o valor total da transação (**GOC_AMOUNT/SPE_AMOUNT**) alinhado à direita:

| | Português | Inglês | Espanhol |
|---------------------------------|-------------------------------------|-----------------------------------|--|
| Se valor < 100.000,00 | VALOR: nn.nnn,nn SENHA: ***** | AMOUNT:nn.nnn,nn PIN: ***** | MONTO: nn.nnn,nn CONTRASEÑA: ***** |
| Se valor ≥ 100.000,00 | n.nnn.nnn.nnn,nn SENHA: ***** | n.nnn.nnn.nnn,nn PIN: ***** | n.nnn.nnn.nnn,nn CONTRASEÑA: ***** |
| Se valor = 0,00 | DIGITE A SENHA | ENTER YOUR PIN | INGRESE SU CONTRASEÑA |

O *design* do campo de entrada do PIN fica a cargo do desenvolvedor do pinpad, de forma a melhor aproveitar as características de seu *display*, podendo inclusive utilizar recursos gráficos mais sofisticados caso suportado.

⚠ Antes de apresentar uma mensagem de captura de PIN no *display*, o pinpad deve enviar ao SPE a seguinte mensagem de notificação: “**SOLICITE A SENHA**”.

No caso de captura de **PIN offline**, deve-se seguir o seguinte fluxo:



A tabela a seguir apresenta as mensagens a serem utilizadas no fluxo, ressaltando que o *design* fica a cargo do desenvolvedor do pinpad, de forma a melhor aproveitar as características de seu *display*:

| | Português | Inglês | Espanhol |
|--------------------|---|--|--|
| MSG_INVPIN | SENHA INCORRETA (RESTAM n TENTATIVAS) | INVALID PIN (n REMAINING ATTEMPTS) | CONTRASEÑA INVÁLIDA (RESTAN n INTENTOS) |
| MSG_BLOCKED | SENHA BLOQUEADA | PIN BLOCKED | CONTRASEÑA BLOQUEADA |
| MSG_WARNING | CUIDADO! O PRÓXIMO ERRO BLOQUEARÁ A SENHA | WARNING! THE NEXT ERROR BLOCKS THE PIN | ATENCIÓN! EL ERROR SIGUIENTE BLOQUEARÁ LA CONTRASEÑA |

Para cada mensagem apresentada no *display* durante o fluxo, o pinpad deve enviar uma mensagem de notificação ao SPE, formatada conforme tabela a seguir (somente em português, sem acentos):

| | NTM_MSG [1..16] | NTM_MSG [17..32] |
|--------------------|------------------------|-------------------------|
| MSG_INVPIN | SENHA INVALIDA | (+n TENTATIVAS) |
| MSG_BLOCKED | SENHA BLOQUEADA | |
| MSG_WARNING | PROX. ERRO | BLOQUEIA SENHA |

⚠ Ao final da captura de PIN, seja ela bem ou malsucedida, o *display* do pinpad deve ser sempre apagado.

6.8.6.2. Valores da Transação

A norma EMV prevê dois formatos de objeto para representar os valores da transação:

| | "b4" (<i>binary</i>) | "n12" (<i>numeric</i>) |
|---------------------------|------------------------|--------------------------|
| <i>Amount, Authorized</i> | <i>tag 81h</i> | <i>tag 9F02h</i> |
| <i>Amount, Other</i> | <i>tag 9F04h</i> | <i>tag 9F03h</i> |

Entretanto, o valor máximo que se pode representar no tipo "b4" é FFFFFFFFh, ou seja, **42.949.672,95**. Caso o cartão solicite um dos objetos acima de tipo "b4" (*binary*) e o valor em questão seja superior a 42.949.672,95, o processamento do comando deve ser encerrado com ↵ST_INVAMOUNT, uma vez que o dado não pode ser repassado corretamente ao cartão.

⚠ Um valor zerado deve ser aceito normalmente pelo pinpad, cabendo ao cartão a decisão de aceitá-lo.

6.8.6.3. Dados protegidos

Determinados comandos desta **seção 6.8** e da **seção 6.9** permitem o fornecimento de uma lista de *tags* para obtenção de dados EMV. Por questão de segurança, os seguintes objetos nunca devem ser devolvidos desta forma, mesmo que conhecidos:

- *PAN - Application Primary Account Number (tag 5Ah)*;
- *Track 1 Discretionary Data (tag 9F1Fh)*;
- *Track 2 Discretionary Data (tag 9F20h)*;
- *Track 1 Data (tag 56h)*; e
- *Track 2 Equivalent Data (tag 57h)*.

6.8.6.4. Objetos do cartão

Determinados comandos desta **seção 6.8** e da **seção 6.9** permitem o fornecimento externo de objetos EMV para uso no processamento.

O pinpad deve sempre ignorar objetos que, pela norma, sabidamente são originados no cartão (como o PAN, por exemplo), além do *TVR*, *TSI* e *CVM Results*. A tabela a seguir lista as *tags* dos parâmetros que, caso recebidos, não devem ser fornecidos ao Kernel EMV para uso no processamento:

| | | | | | | | | | |
|-------|-------|-------|-------|-------|-------|-------|-------|-------|-------|
| 42h | 4Fh | 50h | 56h | 57h | 5Ah | 5F20h | 5F24h | 5F25h | 5F28h |
| 5F2Dh | 5F30h | 5F34h | 5F50h | 5F53h | 5F54h | 5F55h | 5F56h | 61h | 6Fh |
| 70h | 73h | 77h | 80h | 82h | 84h | 87h | 88h | 8Ch | 8Dh |
| 8Eh | 8Fh | 90h | 92h | 93h | 94h | 95h | 97h | 9Bh | 9Dh |
| 9F05h | 9F07h | 9F08h | 9F0Bh | 9F0Dh | 9F0Eh | 9F0Fh | 9F10h | 9F11h | 9F12h |
| 9F13h | 9F14h | 9F17h | 9F1Fh | 9F20h | 9F23h | 9F26h | 9F27h | 9F2Dh | 9F2Eh |
| 9F2Fh | 9F32h | 9F34h | 9F36h | 9F38h | 9F3Bh | 9F42h | 9F43h | 9F44h | 9F45h |
| 9F46h | 9F47h | 9F48h | 9F49h | 9F4Ah | 9F4Bh | 9F4Ch | 9F4Dh | 9F4Fh | A5h |
| BF0Ch | | | | | | | | | |

6.9. Processamento dos comandos Abecs de cartão

Esta seção descreve os tratamentos internos do pinpad para os comandos descritos na **seção 3.7**. Para a sua compreensão, é imprescindível um conhecimento aprofundado da norma EMV (ICC e CTLS).

Os comandos **“GCX”**, **“GOX”** e **“FCX”** devem sempre ser usados nesta sequência, porém o pinpad deve poder aceitar quaisquer outros comandos entre eles, ou seja:

- Qualquer comando pode ser usado entre o **“GCX”** e o **“GOX”** (com exceção do **“FCX”**).
- Qualquer comando pode ser usado entre o **“GOX”** e o **“FCX”** (com exceção do próprio **“GOX”**).

6.9.1. Comando “GCX”

Este comando pede a passagem de um cartão magnético, a inserção ou a aproximação de um cartão com *chip*.

Ao recebê-lo, o pinpad deve inicialmente pesquisar suas Tabelas de AID de forma a identificar quais aplicações de *chip* estão sendo requisitadas pelo SPE. Para identificar quais registros serão envolvidos no processamento, o pinpad deve aplicar a seguinte regra:

- Incluir somente registros que tenham **T1_ICCSTD** = “03” (EMV).
- Se **SPE_AIDLIST** foi fornecido, considerar somente os registros das tabelas em que **TAB_ACQ** e **TAB_RECIDX** correspondem à lista indicada por ele.
- Se **SPE_AIDLIST** estiver ausente:
 - ⇒ Se **SPE_APPTYPE** foi fornecido, considerar somente os registros em que **T1_APPTYPE** possua um dos valores designados, caso contrário ignorar o valor de **T1_APPTYPE**.
 - ⇒ Se **SPE_ACQREF** foi fornecido, considerar somente os registros em que **TAB_ACQ** possua o mesmo valor, caso contrário ignorar o valor de **TAB_ACQ**.
 - ⇒ Na ausência de **SPE_ACQREF** e **SPE_APPTYPE**, todos os registros das Tabelas de AID serão aceitos.

Para simplificar este documento, os registros identificados serão doravante denominados “registros candidatos”.

▲ O Kernel EMV (ICC e CTLS) do pinpad deve suportar uma lista de até **128** (cento e vinte e oito) “registros candidatos”.

Após esse processamento, o pinpad apresenta no *display* a mensagem fornecida pelo SPE no parâmetro **SPE_DSPMSG** (se existente) ou mensagens pré-determinadas conforme tabela a seguir:

| | SPE_DSPMSG recebido | SPE_DSPMSG não recebido | |
|--------------------|--|--|--|
| | | SPE_GCXOPT = “x1xxx” ou SPE_AMOUNT = 0 (zero) | SPE_GCXOPT ≠ “x1xxx” e SPE_AMOUNT ≠ 0 (zero) |
| CTLS não suportado | <texto de SPE_DSPMSG > | INSIRA OU PASSE O CARTÃO | VALOR: 9.999.999.999,99 INSIRA OU PASSE O CARTÃO |
| CTLS suportado (*) |  <texto de SPE_DSPMSG > |  APROXIME, INSIRA OU PASSE O CARTÃO | VALOR: 9.999.999.999,99  APROXIME, INSIRA OU PASSE O CARTÃO |

(*) Considera-se que CTLS é suportado somente se todas as condições seguintes forem válidas:

- O equipamento suporta cartões sem contato;

- O parâmetro SPE_GCXOPT for igual a “1xxxx”;
- Ao menos um dos “registros candidatos” indique essa tecnologia (T1_CTLSMODE com valor válido entre “1” e “9”);
- Se SPE_AMOUNT estiver zerado, ao menos um dos “registros candidatos” indique a possibilidade de processamento *online* neste caso (T1_CTLSZEROAM = “1”); e
- Ao menos um dos “registros candidatos” possua o parâmetro T1_CTLSTRNLIM ou T1_CTLSMBTLIM com valor igual ou superior a SPE_AMOUNT.
- Caso os objetos *Terminal/Reader Contactless Transaction Limit* (tag DF8124h) ou *Terminal/Reader Contactless Transaction Limit - Mobile* (tag DF8125h) tenham sido fornecidos pelo SPE no parâmetro SPE_EMVDATA, estes valores terão prioridade sobre T1_CTLSTRNLIM ou T1_CTLSMBTLIM.

➔ Observações:

- Deve-se desprezar eventuais cartões magnéticos ou teclas pressionadas antes da execução do comando (o pinpad não deve “guardar” estes eventos).
- Ao receber o comando, o pinpad deve sempre limpar eventuais trilhas que estejam armazenadas para leitura através de “GTK”.
- Caso já exista um ICC inserido, nenhuma mensagem deve ser mostrada e a interface CTLS não deve ser ativada.
- Enquanto o pinpad aguarda um cartão, o comando pode ser cancelado pelo operador (através da tecla [CANCELA]).
- Enquanto o pinpad aguarda um cartão, o comando pode ser cancelado pelo SPE através do envio de um byte «**CAN**» (caso em que o pinpad responde com «**EOT**»). Neste caso o *display* deve ser limpo e o comando finalizado.
- Caso SPE_TIMEOUT tenha sido recebido, o pinpad deve finalizar a operação com ↵ST_TIMEOUT se o tempo de espera por um cartão ultrapassar o valor definido.
- Caso a detecção de CTLS tenha sido requerida e SPE_TIMEOUT não tenha sido recebido, o pinpad deve finalizar o comando se atingidos 2 minutos de ociosidade, caso contrário o pinpad espera indefinidamente por um dos eventos.
- Caso a detecção de CTLS tenha sido requerida, porém outro evento tenha sido detectado, a antena deve ser desligada.
- Após a inserção, passagem ou aproximação do cartão, a mensagem deve ser imediatamente apagada do *display*.
- O comando “GCX” pode ser chamado pelo SPE mais de uma vez, independentemente do acionamento de outros comandos (“GOX”, “FCX”, etc.). Caso isso ocorra, o processamento iniciado pela chamada anterior é desprezado e um novo é iniciado, sem a necessidade da remoção do cartão.
- Em qualquer caso de erro, o conteúdo do *display* deve ser apagado.

Dependendo do tipo de cartão utilizado pelo portador, o processamento é diferente e descrito a seguir.

6.9.1.1. Cartão magnético

Caso um cartão magnético seja passado com sucesso, seus dados são simplesmente retornados conforma tabela a seguir:

| Id. do Campo | Dado retornado |
|--------------------|--|
| PP_CARDTYPE | Tipo de cartão lido: "00" = Magnético. |
| PP_ICCSTAT | Status do último processamento de ICC (**). |
| PP_TRK1INC | Dados <u>incompletos</u> da trilha 1, se lida com sucesso (ver seção 6.3.4.1). |
| PP_TRK2INC | Dados <u>incompletos</u> da trilha 2, se lida com sucesso (ver seção 6.3.4.1). |
| PP_TRK3INC | Dados <u>incompletos</u> da trilha 3, se lida com sucesso (ver seção 6.3.4.1). |

(**) Este campo tem o propósito de informar ao SPE o que ocorreu com o último processamento de ICC, de forma que este possa tomar a decisão de aceitar ou não uma transação de "fallback" para a tarja magnética lida.

| PP_ICCSTAT | Situação |
|------------|---|
| "2" | O comando " GCX " imediatamente anterior retornou ↪ST_CARDAPPNAV. |
| "1" | O comando " GCX " imediatamente anterior retornou ↪ST_DUMBCARD ou ↪ST_ERRCARD. |
| "1" | O comando " GCX " ou " GOX " imediatamente anterior retornou ↪ST_ERRFALLBACK. |
| "0" | Outra situação, seja sucesso (↪ST_OK) ou falha. |

Ao ler um cartão magnético com sucesso, o pinpad deve soar um único "beep". No caso de erro de leitura, em que nenhuma trilha é lida com sucesso, o pinpad deve soar dois "beeps".

Se uma trilha possuir tamanho inválido (por exemplo, trilha 1 com mais de 76 posições), a trilha em questão é considerada como "não lida" (erro de leitura).

↻ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_CANCEL | A tecla [CANCEL] foi pressionada pelo portador enquanto o pinpad aguardava um cartão. |
| ↪ST_TIMEOUT | Esgotado tempo de espera para a apresentação de um cartão. |

⚠ Caso um cartão tenha sido passado, mas nenhuma trilha pode ser lida com sucesso, este comando retorna ↪ST_OK, porém sem os campos **PP_TRK1INC**, **PP_TRK2INC** e **PP_TRK3INC**.

6.9.1.2. Cartão com *chip* de contato (ICC EMV)

Caso seja inserido um ICC, o pinpad deve apresentar a mensagem “**PROCESSANDO...**” no *display* e ativar o cartão para efetuar os seguintes processamentos da norma EMV#3:

- *Application Selection*;
- *Initiate Application Processing*; e
- *Read Application Data*.

➔ Application Selection

Neste processo, o pinpad deve fornecer ao seu Kernel EMV a lista de AIDs dos “registros candidatos”, de forma que este possa efetuar o processamento da seleção, utilizando o conceito de “*partial match*” (ver EMV#1).

-  Caso existam “registros candidatos” com AIDs conflitantes⁴, o pinpad deverá eliminar os conflitos de forma a fornecer uma lista sem repetições ao Kernel EMV. Caso o Kernel EMV solicite parâmetros das Tabelas de AID durante o processo de seleção, estes devem ser fornecidos de acordo com a regra descrita na **seção 6.9.5.2**.

Caso o número de “registros candidatos” supere a capacidade definida por esta especificação (conforme **seção 6.1.2**), o pinpad deve retornar .

Caso exista mais de uma aplicação compatível no cartão, ou caso a aplicação (se única) exija confirmação do portador, o pinpad deverá apresentar um menu de seleção contendo as etiquetas das aplicações (*Application Label* ou *Application Preferred Name*, se existente e o *Issuer Code Table Index* for 01h), com o título “**SELECIONE:**” O *layout* do menu é livre de forma a usar melhor os recursos de cada equipamento, lembrando sempre que as etiquetas podem ter até 16 caracteres.

- Enquanto o pinpad aguarda a seleção:
 - ⇒ O comando pode ser cancelado pelo operador através da tecla [CANCELAR]. Neste caso o *display* deve ser limpo e o comando finalizado com .
 - ⇒ O comando pode ser cancelado pelo SPE através do envio de um byte «CAN». (caso em que o pinpad responde com «EOT»). Neste caso o *display* deve ser limpo e o comando finalizado.
 - ⇒ Caso **SPE_TIMEOUT** tenha sido recebido e o tempo de espera por uma ação do operador ultrapassar este valor, o pinpad deve limpar o *display* e finalizar o comando com .
- Durante o processamento do menu, o pinpad sempre deverá enviar uma mensagem de notificação ao SPE informado qual opção está “ativa” (em destaque), enviando-a novamente caso o portador mude a seleção. A mensagem de notificação deve possuir o seguinte formato, sendo que o dado “XXX...X” representa a etiqueta apresentada no menu:

⁴ Dois AIDs são conflitantes quando forem exatamente iguais ou quando o AID de menor tamanho estiver inteiramente contido à esquerda do AID de maior tamanho. Por exemplo, os AIDs ‘A0 00 00 00 03 10 10’ e ‘A0 00 00 00 03 10 10 98 76’ são conflitantes.

| | |
|------------------------|-------------------------|
| NTM_MSG [1..16] | NTM_MSG [17..32] |
| SELECIONADO: | XXXXXXXXXXXXXXXXXXXX |

Caso o cartão possua somente uma aplicação compatível e esta é selecionada automaticamente pelo Kernel EMV, a mesma mensagem de notificação deve ser enviada.

Ao final do processo, a mensagem “SELECIONADO: XXX..X” deve ser deixada no *display*, também com *layout* livre.

➔ Initiate Application Processing / Read Application Data

Determinada a aplicação a ser utilizada e, conseqüentemente, o registro correspondente na Tabela de AIDs, o processamento do Kernel EMV deve prosseguir utilizando os seguintes parâmetros (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---|-------|--|
| <i>Amount, Authorized (binary)</i> | 81h | SPE_AMOUNT (ver seção 6.8.6.2) |
| <i>Amount, Authorized (numeric)</i> | 9F02h | SPE_AMOUNT |
| <i>Amount, Other (binary)</i> | 9F04h | SPE_CASHBACK (ver seção 6.8.6.2) |
| <i>Amount, Other (numeric)</i> | 9F03h | SPE_CASHBACK |
| <i>Transaction Date</i> | 9Ah | SPE_TRNDATE |
| <i>Transaction Time</i> | 9F21h | SPE_TRNTIME |
| <i>Application Version Number</i> | 9F09h | T1_APPVER1 |
| <i>Terminal Country Code</i> | 9F1Ah | T1_TRMCNTRY |
| <i>Transaction Currency Code</i> | 5F2Ah | SPE_TRNCURR (se não fornecido, usar T1_TRNCURR) |
| <i>Transaction Currency Exponent</i> | 5F36h | T1_TRNCRREXP |
| <i>Merchant Identifier</i> | 9F16h | T1_MERCHID |
| <i>Merchant Category Code</i> | 9F15h | T1_MCC |
| <i>Terminal Identification</i> | 9F1Ch | T1_TRMID |
| <i>Terminal Capabilities</i> | 9F33h | T1_TRMCAPAB |
| <i>Additional Terminal Capabilities</i> | 9F40h | T1_ADDTRMCP |
| <i>Terminal Type</i> | 9F35h | T1_TRMTYP |
| <i>Terminal Floor Limit</i> | 9F1Bh | T1_FLRLIMIT |
| <i>Transaction Category Code</i> | 9F53h | T1_TCC |
| <i>Transaction Sequence Counter</i> | 9F41h | Contador <u>regido internamente pelo pinpad</u> . |

| Objeto EMV | Tag | Origem (Id. do Campo) |
|-------------------------|-----|--|
| <i>Transaction Type</i> | 9Ch | SPE_TRNTYPE . Se ausente, adotar: 09h → Se SPE_CASHBACK presente e diferente de zero; ou 00h → Outras situações. |

- ⚠ Se **SPE_EMVDATA** foi fornecido pelo SPE, seus objetos TLV são usados no processamento e possuem prioridade sobre as definições da tabela acima (ver restrições na **seção 6.8.6.4**).

Caso o processo EMV indique a exclusão da aplicação selecionada da lista de candidatas e, havendo mais de uma (ou seja, foi apresentado um menu), o pinpad deve:

- Apresentar a mensagem “APLICAÇÃO INVÁLIDA”, devidamente formatada para o *display*.
- Enviar a seguinte mensagem de notificação ao SPE:

| NTM_MSG [1..16] | NTM_MSG [17..32] |
|------------------------|-------------------------|
| APLICACAO | INVALIDA |

- Aguardar 1,5s (um segundo e meio) e retornar o processamento à etapa “*Application Selection*”.

➡ Dados de Saída

Caso o ICC seja processado com sucesso, a resposta ao comando “**GCX**” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|----------------------|---|
| PP_CARDTYPE | Tipo de cartão lido: “03” = ICC EMV. |
| PP_AIDTABINFO | Concatenação dos campos TAB_ACQ , TAB_RECIDX e T1_APPTYPE do(s) registro(s) das Tabelas de AID usado(s) no processamento. IMPORTANTE: No caso de ter havido conflito de AIDs para a aplicação selecionada, este campo retorna uma lista contendo os dados dos diversos registros das Tabelas de AID que originaram o conflito. |
| PP_PAN | <i>Application PAN - Primary Account Number (tag 5Ah).</i> |
| PP_PANSEQNO | <i>PAN Sequence Number (tag 5F34h).</i> Se ausente, retornar “00” . |
| PP_TRK2INC | <i>Track 2 Equivalent Data (tag 57h)</i> , se existente no cartão, devolvida incompleta conforme descrito na seção 6.3.4.1 . |
| PP_CHNAME | <i>Cardholder Name (tag 5F20h)</i> , se existente no cartão. |
| PP_LABEL | Mesma etiqueta da aplicação apresentada no processo de seleção, podendo ser o <i>Application Label (tag 50h)</i> ou <i>Application Preferred Name (tag 9F12h)</i> . |
| PP_ISSCNTRY | <i>Issuer Country Code (tag 5F28h)</i> , se existente no cartão. |
| PP_CARDEXP | <i>Application Expiration Date (tag 5F24h)</i> , se existente no cartão. |

| Id. do Campo | Dado retornado |
|-------------------|--|
| PP_EMVDATA | <p>Dados definidos pela <i>tags</i> em SPE_TAGLIST, no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3</p> <p>Este campo é mandatário sempre que SPE_TAGLIST existir no comando, mesmo que nenhum objeto seja encontrado (caso em que é retornado com tamanho zerado).</p> <p>IMPORTANTE:</p> <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |

➤ Situações de exceção:

| RSP_STAT | Situação |
|-------------------|--|
| ↳ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↳ST_CANCEL | A tecla [CANCELA] foi pressionada pelo portador enquanto o pinpad aguardava um cartão ou durante o menu de seleção de aplicação. |
| ↳ST_TIMEOUT | Esgotado tempo de espera para a apresentação de um cartão. |
| ↳ST_NOCARD | ICC foi removido durante a apresentação do menu de seleção. |
| ↳ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↳ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↳ST_CARDINVALIDAT | A <u>única aplicação compatível</u> no cartão está invalidada (SELECT retornou SW1/SW2 = 6283h). |
| ↳ST_CARDBLOCKED | Comando SELECT retorna erro SW1/SW2=6A81h (cartão foi bloqueado). |
| ↳ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↳ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ↳ST_CARDAPPAV | O ICC não possui nenhuma aplicação compatível para o processamento. |
| ↳ST_CARDAPPNAUT | <ul style="list-style-type: none"> ▪ A <u>única aplicação compatível</u> no cartão retornou erro SW1/SW2 = 6985h no comando GET PROCESSING OPTIONS. ▪ A <u>única aplicação compatível</u> no cartão retornou erro no comando SELECT final. |
| ↳ST_ERRFALLBACK | O cartão reportou no GET PROCESSING OPTIONS um status (SW1/SW2) cujo comportamento não é regido pela norma EMV. |

| RSP_STAT | Situação |
|---------------|--|
| ↳ST_ERRMAXAID | Número de AIDs candidatos supera a capacidade de tratamento do Kernel EMV. |

6.9.1.3. Cartão com *chip* sem contato (CTLS)

Caso seja apresentado um CTLS ao pinpad, este deverá “filtrar” os “registros candidatos” de forma a fornecer ao Kernel EMV CTLS uma lista contendo somente os AIDs provenientes dos registros que cumpram os seguintes requisitos:

- O campo **T1_CTLSMODE** deve ter valor válido entre “1” e “9”;
- Se **SPE_AMOUNT** estiver zerado, o campo **T1_CTLSZEROAM** deve ser igual a “1”;
- O campo **T1_CTLSTRNLIM** ou **T1_CTLSBTMLIM** deve ter valor maior ou igual a **SPE_AMOUNT**.

Caso o número de “registros candidatos” supere a capacidade definida por esta especificação (conforme **seção 6.1.2**), o pinpad deve retornar ↳ST_ERRMAXAID.

- ⚠ Caso os “registros candidatos” resultantes conttenham AIDs conflitantes⁵ (de mesmo valor), o pinpad deverá eliminar os conflitos de forma a fornecer uma lista sem repetições ao Kernel EMV CTLS.
- ⚠ Caso o cartão contenha mais de uma aplicação compatível, a aplicação de maior prioridade será selecionada automaticamente.

Ao final do processo de seleção, uma mensagem de notificação deve ser enviada ao SPE no seguinte formato, sendo que o dado “XXX..X” representa a etiqueta da aplicação (*Application Label* ou *Application Preferred Name*, se existente e o *Issuer Code Table Index* for 01h):

| NTM_MSG [1..16] | NTM_MSG [17..32] |
|-----------------|----------------------|
| SELECIONADO: | XXXXXXXXXXXXXXXXXXXX |

A mensagem “SELECIONADO: XXX..X” deve ser deixada no *display*, com *layout* livre de acordo com as capacidades do equipamento.

Identificada a aplicação a ser usada, deve-se verificar o valor de **T1_CTLSMODE**, de forma a efetuar os processamentos específicos de cada “bandeira” (Visa, Mastercard, American Express ou Discover), conforme descrito em [VCPS](#), [PPMChip](#), [ExpPay](#), [D-PAS](#) e [QckPass](#).

Pelas características do CTLS, todo o processamento é feito já na função “**GcX**” em um único “toque”. Caso a transação requeira verificação do portador (PIN *online*), isso deverá ser feito em “**Gox**”.

⁵ Dois AIDs são conflitantes quando forem exatamente iguais ou quando o AID de menor tamanho estiver inteiramente contido à esquerda do AID de maior tamanho. Por exemplo, os AIDs ‘A0 00 00 00 03 10 10’ e ‘A0 00 00 00 03 10 10 98 76’ são conflitantes.

▲ Para cartões Visa payWave e Discover D-PAS há ainda a possibilidade de processamento de *Issuer Scripts* em “**FCX**”, situação descrita na **seção 6.9.4.4**.

A tabela a seguir lista os objetos genéricos que devem ser fornecidos ao Kernel EMV CTLs (com as devidas conversões de formato), independentemente do valor de **T1_CTLSMODE**:

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---|------------|--|
| <i>Amount, Authorized (binary)</i> | 81h | SPE_AMOUNT (ver seção 6.8.6.2) |
| <i>Amount, Authorized (numeric)</i> | 9F02h | SPE_AMOUNT |
| <i>Transaction Date</i> | 9Ah | SPE_TRNDATE |
| <i>Transaction Time</i> | 9F21h | SPE_TRNTIME |
| <i>Application Version Number</i> | 9F09h | T1_APPVER1 |
| <i>Terminal Country Code</i> | 9F1Ah | T1_TRMCNTRY |
| <i>Transaction Currency Code</i> | 5F2Ah | T1_TRNCURR |
| <i>Transaction Currency Exponent</i> | 5F36h | T1_TRNCREXP |
| <i>Merchant Identifier</i> | 9F16h | T1_MERCHID |
| <i>Merchant Category Code</i> | 9F15h | T1_MCC |
| <i>Terminal Identification</i> | 9F1Ch | T1_TRMID |
| <i>Terminal Capabilities</i> | 9F33h | T1_CTLSTRMCP |
| <i>Additional Terminal Capabilities</i> | 9F40h | T1_CTLSADDTC |
| <i>Terminal Type</i> | 9F35h | T1_TRMTYP |
| <i>Terminal/Reader Contactless Transaction Limit</i> | DF8124h | T1_CTLSTRNLIM |
| <i>Terminal/Reader Contactless Transaction Limit - Mobile</i> | DF8125h | T1_CTLSMBTLIM |
| <i>Terminal/Reader Contactless Floor Limit</i> | DF8123h | T1_CTLSFLRLIM |
| <i>Terminal/Reader CVM Required Limit</i> | DF8126h | T1_CTLSCVMLIM |
| <i>Contactless Term. Action Code – Default</i> | DF8120h | T1_CTLSTACDEF |
| <i>Contactless Term. Action Code – Denial</i> | DF8121h | T1_CTLSTACDEN |
| <i>Contactless Term. Action Code – Online</i> | DF8122h | T1_CTLSTACONL |
| <i>Transaction Type</i> | 9Ch | SPE_TRNTYPE . Se ausente, adotar: 09h → Se SPE_CASHBACK presente e diferente de zero; ou 00h → Outras situações. |

- ▲ Caso existam “registros candidatos” com AIDs conflitantes (de mesmo valor), os parâmetros a serem fornecidos ao Kernel EMV CTLS devem respeitar as regras descritas na **seção 6.9.5.2**.
- ▲ Se **SPE_EMVDATA** foi fornecido pelo SPE, seus objetos TLV são usados no processamento e possuem prioridade sobre as definições da tabela acima (ver restrições na **seção 6.8.6.4**).
- ▲ Durante o processamento do CLTS, o pinpad deve apresentar os indicadores visuais (LEDs) e sonoros (*beeps*) conforme definido em **EMV#CtIsA** (Capítulo 9).

➔ Tratamentos específicos

- Se **T1_CTLSMODE** = “1” ou “2”, os mesmos parâmetros adicionais definidos na **seção 6.8.1.3 (Parâmetros específicos - Visa payWave)** devem ser fornecidos ao Kernel EMV CTLS.
- Se **T1_CTLSMODE** = “3” ou “4”, os mesmos parâmetros adicionais definidos na **seção 6.8.1.3 (Parâmetros específicos - Mastercard PayPass)** devem ser fornecidos ao Kernel EMV CTLS.
- Se **T1_CTLSMODE** = “5” ou “6”, os mesmos parâmetros adicionais definidos na **seção 6.8.1.3 (Parâmetros específicos - Amex ExpressPay)** devem ser fornecidos ao Kernel EMV CTLS.
- ~~Se **T1_CTLSMODE** = “7”, os mesmos parâmetros adicionais definidos na **seção 6.8.1.3 (Parâmetros específicos - Pure Contactless)** devem ser fornecidos ao Kernel EMV CTLS.~~
- Se **T1_CTLSMODE** = “8” ou “9”, os mesmos parâmetros adicionais definidos na **seção 6.8.1.3 (Parâmetros específicos - Discover D-PAS)** devem ser fornecidos ao Kernel EMV CTLS.
- Se **T1_CTLSMODE** = “B”, os mesmos parâmetros adicionais definidos na **seção 6.8.1.3 (Parâmetros específicos - Discover D-PAS)** devem ser fornecidos ao Kernel EMV CTLS.

➔ Offline Data Authentication

O processo de autenticação *offline*, o pinpad deve fornecer ao Kernel EMV CTLS, antes do processamento, determinadas chaves públicas de Autoridade Certificadora disponíveis em suas Tabelas de CAPK. Entretanto, como as Tabelas de CAPK podem ser “aglutinadas” pelo SPE conforme descrito na **seção 4.1.2**, esse processo pode seguir duas lógicas distintas.

Caso o SPE tenha aglutinado chaves em uma tabela com **TAB_ACQ** = “00”, deve-se adotar o seguinte procedimento:

- Partindo-se do princípio que o SPE já fez corretamente o tratamento descrito na **seção 4.1.2**, deve-se utilizar somente as chaves da tabela aglutinada (**TAB_ACQ** = “00”), desprezando eventuais outras tabelas (**TAB_ACQ** ≠ “00”).

Caso o SPE **não** tenha aglutinado chaves, deve-se adotar o seguinte procedimento:

- Utilizar somente chaves das redes credenciadoras que geraram “registros candidatos”.
- Dependendo das características do Kernel EMV CTLS, o pinpad deverá eliminar eventuais duplicidades (chaves com mesmo **T2_RID** e **T2_CAPKIDX**), porém as regras para isso não estão contempladas nesta especificação.

➔ Dados de Saída (CTLS simulando tarja)

Caso o CTLS seja processado com sucesso na modalidade ~~de PayPass Mag Stripe, Expresspay Magstripe Mode ou D-PAS MS Mode~~ *magstripe* (simulação de tarja magnética), a resposta ao comando “GCX” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|----------------------|---|
| <u>PP_CARDTYPE</u> | Tipo de cartão lido: “05” = CTLS simulando tarja. |
| <u>PP_AIDTABINFO</u> | Concatenação dos campos <u>TAB_ACQ</u> , <u>TAB_RECIDX</u> e <u>T1_APPTYPE</u> do(s) registro(s) das Tabelas de AID usado(s) no processamento. IMPORTANTE: No caso de ter havido conflito de AIDs para a aplicação selecionada, este campo retorna uma lista contendo os dados dos diversos registros das Tabelas de AID que originaram o conflito. |
| <u>PP_TRK1INC</u> | Dados da trilha 1, montados de acordo com a especificação da “bandeira”, devolvida incompleta conforme descrito na seção 6.3.4.1 . |
| <u>PP_TRK2INC</u> | Dados da trilha 2, montados de acordo com a especificação da “bandeira”, devolvida incompleta conforme descrito na seção 6.3.4.1 . |
| <u>PP_LABEL</u> | Mesma etiqueta da aplicação apresentada no processo de seleção, podendo ser o <i>Application Label (tag 50h)</i> ou <i>Application Preferred Name (tag 9F12h)</i> . <i>Importante:</i> retornar campo vazio (tamanho zero) se o <i>Application Label</i> estiver ausente no cartão. |
| <u>PP_DEVTYPE</u> | Depende da modalidade/bandeira processada (ver “Dados de Saída - CTLS EMV”). |

➔ Dados de Saída (CTLS EMV)

Caso o CTLS seja processado com sucesso na modalidade ~~de VSDC, PayPass M/Chip, Expresspay EMV Mode ou D-PAS EMV Mode~~ *EMV full* (não simulando tarja magnética), a resposta ao comando “GCX” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|----------------------|---|
| <u>PP_CARDTYPE</u> | Tipo de cartão lido: “06” = CTLS EMV. |
| <u>PP_AIDTABINFO</u> | Concatenação dos campos <u>TAB_ACQ</u> , <u>TAB_RECIDX</u> e <u>T1_APPTYPE</u> do(s) registro(s) das Tabelas de AID usado(s) no processamento. IMPORTANTE: No caso de ter havido conflito de AIDs para a aplicação selecionada, este campo retorna uma lista contendo os dados dos diversos registros das Tabelas de AID que originaram o conflito. |
| <u>PP_PAN</u> | <i>Application PAN - Primary Account Number (tag 5Ah)</i> . Caso este objeto não exista no cartão, extraí-lo de <u>PP_TRK2INC</u> . |
| <u>PP_PANSEQNO</u> | <i>PAN Sequence Number (tag 5F34h)</i> . Se ausente, retornar “00”. |
| <u>PP_TRK2INC</u> | Dados da trilha 2, montados de acordo com a especificação da “bandeira”, devolvida incompleta conforme descrito na seção 6.3.4.1 . |

| Id. do Campo | Dado retornado |
|--|--|
| PP_CHNAME | Cardholder Name (tag 5F20h), se existente no cartão. |
| PP_LABEL | Mesma etiqueta da aplicação apresentada no processo de seleção, podendo ser o <i>Application Label (tag 50h)</i> ou <i>Application Preferred Name (tag 9F12h)</i> . Importante: retornar campo vazio (tamanho zero) se o <i>Application Label</i> estiver ausente no cartão. |
| PP_ISSCNTRY | <i>Issuer Country Code (tag 5F28h)</i> , se existente no cartão. |
| PP_CARDEXP | <i>Application Expiration Date (tag 5F24h)</i> , se existente no cartão. |
| PP_EMVDATA | Dados definidos pela <i>tags</i> em SPE_TAGLIST , no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3 Este campo é mandatário sempre que SPE_TAGLIST existir no comando, mesmo que nenhum objeto seja encontrado (caso em que é retornado com tamanho zerado). IMPORTANTE: <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |
| PP_DEVTYPE (Visa payWave) | De acordo com os bits de 5 a 1 do 1º byte do <i>Form Factor Indicator (tag 9F6Eh)</i> : 0000 (<i>Standard card</i>) → “00” (Cartão) 00001 (<i>Mini-card</i>) → “00” (Cartão) 00011 (<i>Consumer mobile phone</i>) → “01” (Telefone móvel) 00100 (<i>Wrist-worn device</i>) → “05” (Pulseira) outro valor → “99” (não definido) |
| PP_DEVTYPE (Mastercard PayPass) | <i>Device Type - quinto e sexto bytes de Third Party Data (tag ‘9F6E’)</i> . |
| PP_DEVTYPE (Amex Expresspay) | Se o bit 4 do 1º byte de <i>Card Interface and Payment Capabilities (tag 9F70h)</i> estiver ativo (“ <i>Mobile Interface Supported</i> ”), retornar “01” (Telefone móvel), caso contrário retornar “00” (Cartão). |
| PP_DEVTYPE (Pure Contactless) | Quinto byte o objeto de tag 9F6Eh, no mesmo formato do Mastercard PayPass. |
| PP_DEVTYPE (Discover D-PAS) | Se o bit 5 do 1º byte de <i>Card Processing Requirement (tag 9F71h)</i> estiver ativo (“ <i>Consumer Device CVM Performed</i> ”), retornar “01” (Telefone móvel), caso contrário retornar “00” (Cartão). |

| Id. do Campo | Dado retornado |
|---|--|
| <u>PP_DEVTYPE</u> (Union Pay Quick Pass) | Se o bit 8 do 2º byte de <i>Card Transaction Qualifiers</i> (tag 9F6Ch) estiver ativo (“ <i>Consumer Device CVM Performed</i> ”), retornar “01” (Telefone móvel), caso contrário retornar “00” (Cartão). |

➤ Situações de exceção:

| RSP_STAT | Situação |
|---------------------|--|
| ↪ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_TIMEOUT | Esgotado tempo de espera para a apresentação de um cartão. |
| ↪ST_CTLSMULTIPLE | Mais de um CTLS foi apresentado ao leitor simultaneamente. |
| ↪ST_CTLSCOMMERR | <ul style="list-style-type: none"> ▪ Erro de comunicação entre o pinpad (antena) e o CTLS. ▪ O Kernel CTLS solicitou a verificação do portador no dispositivo móvel (<i>Outcome = “Try Again”</i>) e <u>T1_MOBCVM</u> ≠ “1”.⁶ |
| ↪ST_CARDBLOCKED | Comando SELECT retorna erro SW1/SW2=6A81h (cartão foi bloqueado). |
| ↪ST_CTLSINVALIDAT ✘ | Comando SELECT retorna erro SW1/SW2=6A81h ou 6283h. |
| ↪ST_CTLSPROBLEMS | CTLS com problemas. Esse status é válido para muitas ocorrências no processamento onde o CTLS não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↪ST_CTLSAPPNAV ✘ | O CTLS não possui nenhuma aplicação compatível para o processamento. |
| ↪ST_CTLSAPPNAUT ✘ | O cartão retornou erro SW1/SW2 = 6985h no comando GET PROCESSING OPTIONS. |
| ↪ST_ERRMAXAID | Número de AIDs candidatos supera a capacidade de tratamento do Kernel EMV. |
| ↪ST_CTLSEXTCVM | O Kernel CTLS solicitou a verificação do portador no dispositivo móvel (<i>Outcome = “Try Again”</i>) e <u>T1_MOBCVM</u> = “1”. |
| ↪ST_CTLSIFCHG | <ul style="list-style-type: none"> ▪ Kernel CTLS solicitou “mudança de interface” para processamento usando ICC ou cartão magnético (<i>Outcome = “Try Another Interface”</i>). ▪ Se <u>GCR_AMOUNT</u> ≥ <u>T1_CTLSTRNLIM</u> e cartão Discover D-PAS ou Visa payWave. |

✘ De acordo com [EMV#CtlsB](#) seções 3.3.2.6 até 3.3.3.5, qualquer erro no comando SELECT faz com que o AID em questão seja retirado da lista de candidatos e, se a lista estiver vazia, deve-se solicitar mudança de interface. Assim sendo, estas situações acabam acarretando ↪ST_CTLSIFCHG.

⁶ Isto é feito para se manter compatibilidade com um SPE antigo que não conhece ↪ST_CTLSEXTCVM.

6.9.2. Comando “GED”

Este comando é chamado pelo SPE para obter dados do processamento EMV, desde que o comando “GCX” tenha sido executado previamente com sucesso para um ICC EMV (**PP_CARDTYPE** = “03”), CTLS emulando tarja (**PP_CARDTYPE** = “05”) ou CTLS EMV (**PP_CARDTYPE** = “06”).

Ao recebê-lo, o pinpad deve pesquisar no Kernel EMV os dados requeridos através das *tags* fornecidas em **SPE_TAGLIST** e retornar a estrutura TLV correspondente em **PP_EMVDATA**.

Objetos que não sejam conhecidos do Kernel EMV simplesmente não são devolvidos, assim como os objetos descritos na **seção 6.8.6.3**.

- ▲ O SPE pode solicitar objetos proprietários do cartão cujas *tags* não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam.
- ▲ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, se não existir, devolver o valor do registro da tabela que foi usado na parametrização da transação.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ST_INVCALL | Comando “GCX” não foi executado previamente com sucesso para ICC/CTLS. |
| ST_RSPOVRF | Tamanho dos dados EMV ultrapassa máximo permitido para PP_EMVDATA. |

6.9.3. Comando “GOX”

O comando “GOX” continua o processamento de cartões ICC EMV (**PP_CARDTYPE** = “03”) ou CTLS EMV (**PP_CARDTYPE** = “06”), forçando o uso do registro da Tabela de AID indicada por **SPE_ACQREF**.

Os tratamentos envolvidos são diferentes dependendo do tipo de *chip*, descritos a seguir.

➔ Observações:

- Ao final do processamento do comando, o pinpad deve apagar o *display* em caso de erro.

6.9.3.1. Cartão com *chip* de contato (ICC EMV)

O processamento de ICC EMV deve continuar a seguir os processamentos estipulados pela norma EMV:

- *Processing Restrictions*;
- *Offline Data Authentication*;
- *Cardholder Verification*;
- *Terminal Risk Management*;
- *Terminal Action Analysis*; e
- *Card Action Analysis*.

Para que o Kernel EMV possa continuar o processamento, os seguintes parâmetros devem ser fornecidos a ele, além dos que já foram estipulados no comando “GCX” (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|--------------------------------------|-------|---|
| <i>Amount, Authorized (binary)</i> | 81h | SPE_AMOUNT (ver seção 6.8.6.2) |
| <i>Amount, Authorized (numeric)</i> | 9F02h | SPE_AMOUNT |
| <i>Amount, Other (binary)</i> | 9F04h | SPE_CASHBACK (ver seção 6.8.6.2). |
| <i>Amount, Other (numeric)</i> | 9F03h | SPE_CASHBACK |
| <i>Terminal Country Code</i> | 9F1Ah | T1_TRMCNTRY |
| <i>Transaction Currency Code</i> | 5F2Ah | SPE_TRNCURR (se não fornecido, usar SPE_TRNCURR fornecido em “GCX”, se existente, ou T1_TRNCURR ^(*)) |
| <i>Transaction Currency Exponent</i> | 5F36h | T1_TRNCRREXP ^(*) |
| <i>Merchant Identifier</i> | 9F16h | T1_MERCHID ^(*) |
| <i>Merchant Category Code</i> | 9F15h | T1_MCC ^(*) |
| <i>Terminal Identification</i> | 9F1Ch | T1_TRMID ^(*) |

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---|---------|--|
| <i>Terminal Capabilities</i> | 9F33h | <u>T1_TRMCAPAB</u> (*) |
| <i>Additional Terminal Capabilities</i> | 9F40h | <u>T1_ADDTRMCP</u> (*) |
| <i>Terminal Type</i> | 9F35h | <u>T1_TRMTYP</u> (*) |
| <i>Terminal Action Code – Default</i> | DF9F0Dh | <u>T1_TACDEF</u> |
| <i>Terminal Action Code – Denial</i> | DF9F0Eh | <u>T1_TACDEN</u> |
| <i>Terminal Action Code – Online</i> | DF9F0Fh | <u>T1_TACONL</u> |
| <i>Terminal Floor Limit</i> | 9F1Bh | <u>T1_FLRLIMIT</u> (*) |
| <i>Transaction Category Code</i> | 9F53h | <u>T1_TCC</u> (*) |
| <i>Transaction Type</i> | 9Ch | <u>SPE_TRNTYPE</u> . Se ausente, adotar: 09h → Se <u>SPE_CASHBACK</u> presente e diferente de zero; ou 00h → Outras situações. |

(*) Caso tenha havido conflito de AID no comando “GCTX” (ver [seção 6.9.5.2](#)), devem ser considerados agora os parâmetros inalterados da [Tabela de AID](#) da rede indicada por SPE_ACOREF.

⚠ Se SPE_EMVDATA foi fornecido pelo SPE, seus objetos TLV são usados no processamento e possuem prioridade sobre as definições da tabela acima (ver restrições na [seção 6.8.6.4](#)).

➡ Processing Restrictions

Para esta etapa do processamento, devem-se usar os seguintes objetos de dados:

| Objeto EMV | Tag | Origem (Id. do Campo) |
|-----------------------------------|-------|---|
| <i>Application Version Number</i> | 9F09h | <u>T1_APPVER1</u> , <u>T1_APPVER2</u> ou <u>T1_APPVER3</u> (o que coincidir com a versão do cartão, ou <u>T1_APPVER1</u> se não houver coincidência). |

Além disso, se SPE_GOLOPT = “1xxxx”, deve-se informar ao Kernel EMV que o número do cartão está em uma *Exception List*.

➡ Offline Data Authentication

Para o processo de autenticação *offline*, o pinpad deve fornecer ao Kernel EMV a chave pública da Autoridade Certificadora mediante pesquisa de T2_RID e T2_CAPKIDX nas Tabelas de CAPK.

- Se T2_CHKSTAT = “1” e T2_CHECKSUM não for coerente, a autenticação deve simplesmente falhar.

As Tabelas de CAPK podem ser “aglutinadas” pelo SPE, conforme descrito na [seção 4.1.2](#). Desta forma, deve-se adotar a seguinte regra:

- Primeiro pesquisar os registros da Tabela de CAPK em que **TAB_ACQ = SPE_ACQREF**.
- Caso o registro não seja encontrado, pesquisar a tabela em que **TAB_ACQ = "00"**.

Além disso, fornecer ao Kernel EMV os registros da Tabela de Certificados Revogados (ver **seção 4.1.3**) em que **TAB_ACQ = SPE_ACQREF**.

Outros parâmetros que devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|--|-----|-----------------------|
| <i>Default Dynamic Data Authentication Data Object List (DDOL)</i> | --- | T1_DDOLDEF |

➔ Cardholder Verification

Caso a verificação de portador indique a necessidade de validação de PIN, deve-se seguir o detalhamento indicado na **seção 6.9.5.1**.

Para o caso de PIN *online*:

- Deve-se usar os parâmetros **SPE_MTHDPIN**, **SPE_KEYIDX** e **SPE_WKENC**.
- No caso de problemas com a chave indicada, o pinpad deve abortar a operação com ↵ST_ERRKEY, não dando continuidade ao processamento EMV.
- O valor do PAN usado no cálculo do PIN criptografado deve ser obtido diretamente do Kernel EMV, sendo que sua existência não é afetada por um eventual uso prévio do comando "**GTK**".

Enquanto o pinpad aguarda a digitação de um PIN, o comando pode ser cancelado pelo SPE através do envio de um byte «CAN».

Se **SPE_GOXOPT = "xx1xx"**, deve-se impedir o *bypass* do PIN, seja este online ou offline.

O pinpad deve finalizar a operação com ↵ST_TIMEOUT se o tempo de inatividade em uma tela de captura de PIN ultrapassar o valor definido em **SPE_TIMEOUT**, ou 1 minuto (60 segundos) se este não tiver sido fornecido.

➔ Terminal Risk Management

Para este processo, os seguintes parâmetros devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---|-------|--------------------------|
| <i>Terminal Floor Limit (em centavos)</i> | 9F1Bh | SPE_TRMPAR [1..4] |
| <i>Target Percentage to be used for Biased Random Selection</i> | -- | SPE_TRMPAR [5] |
| <i>Threshold Value for Biased Random Selection (em centavos)</i> | -- | SPE_TRMPAR [6..9] |
| <i>Maximum Target Percentage to be used for Biased Random Selection</i> | -- | SPE_TRMPAR [10] |

▲ Se **SPE_TRMPAR** não for fornecido pelo SPE, deve-se considerar os valores como zerados.

➔ Terminal Action Analysis

Para este processo, os seguintes parâmetros devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|---------------------------------------|-----|-----------------------|
| <i>Terminal Action Code – Default</i> | --- | T1_TACDEF |
| <i>Terminal Action Code – Denial</i> | --- | T1_TACDEN |
| <i>Terminal Action Code – Online</i> | --- | T1_TACONL |

▲ Se **SPE_GOXOPT** = "x1xxx", o Kernel EMV nunca poderá sugerir aprovação offline (TC) ao cartão. Essa implementação não pode usar o bit "Merchant Forced Transaction Online" do TVR para impedir a aprovação offline, pois isso não é aceito nas certificações de bandeiras.

➔ Card Action Analysis.

Para este processo, os seguintes parâmetros devem ser fornecidos ao Kernel EMV (com as devidas conversões de formato):

| Objeto EMV | Tag | Origem (Id. do Campo) |
|--|-----|-----------------------|
| <i>Default Transaction Certificate Data Object List (TDOL)</i> | --- | T1_TDOLDEF |

➔ Dados de Saída

Caso o ICC seja processado com sucesso, a resposta ao comando "**GOX**" deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|------------------|--|
| PP_GOXRES | Decisão do cartão no comando 1st GENERATE AC: "0xxxxx" → Cartão retornou TC (aprovada <i>offline</i>); "1xxxxx" → Cartão retornou AAC (negada <i>offline</i>); ou "2xxxxx" → Cartão retornou ARQC (requer autorização <i>online</i>). Resultado do <i>Cardholder Verification</i> : "x1xxxx" = Deve-se coletar assinatura em papel; "xx1xxx" = PIN foi verificado com sucesso <i>offline</i> ; "xx2xxx" = PIN capturado para verificação <i>online</i> . |
| PP_PINBLK | PIN criptografado para validação <i>online</i> (somente se PP_GOXRES = "xx2xxx"). |
| PP_KSN | KSN da chave DUKPT usada na criptografia de PIN <i>online</i> (somente se PP_GOXRES = "xx2xxx" e SPE_MTHDPIN = "2" ou "3"). |

| Id. do Campo | Dado retornado |
|-------------------|--|
| <u>PP_EMVDATA</u> | <p>Dados definidos pela <i>tags</i> em <u>SPE_TAGLIST</u>, no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3</p> <p>Este campo é mandatário sempre que <u>SPE_TAGLIST</u> existir no comando, mesmo que nenhum objeto seja encontrado (caso em que é retornado com tamanho zerado).</p> <p>IMPORTANTE:</p> <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |

➤ Situações de exceção:

| RSP_STAT | Situação |
|------------------|--|
| ↵ST_INVPARM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ O valor de <u>SPE_ACQREF</u> não corresponde a nenhum informado na resposta do "<u>GCX</u>" (campo <u>PP_AIDTABINFO</u>). |
| ↵ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando "<u>GCX</u>" não foi chamado previamente. ▪ Comando "<u>GCX</u>" foi chamado previamente, porém retornou <u>PP_CARDTYPE</u> diferente de "03" e "06". ▪ Comando "<u>GOX</u>" já foi chamado. |
| ↵ST_TIMEOUT | Ultrapassado tempo de inatividade em uma tela de captura de PIN. |
| ↵ST_ERRKEY | Foi requerida captura de PIN <i>online</i> , mas a chave indicada está ausente ou corrompida. |
| ↵ST_CANCEL | A tecla [CANCEL] foi pressionada pelo portador durante a tela de captura de PIN. |
| ↵ST_NOCARD | ICC foi removido durante a tela de captura de PIN. |
| ↵ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↵ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↵ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↵ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ↵ST_ERRFALLBACK | O comando 1st GENERATE AC retornou SW1/SW2 diferente de 9000h. |

| RSP_STAT | Situação |
|----------------|--|
| ST_INVAMOUNT | O cartão pediu informação de valor no formato “b4” e este supera a capacidade campo. |
| ST_CARDAPPNAUT | O objeto <i>Cryptogram Information Data</i> (tag ‘9F27’) retornado pelo cartão indica situação “ <i>Service not allowed</i> ”. |
| ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.9.3.2. Cartão com *chip* sem contato (CTLS EMV)

O cartão sem contato é processado inteiramente no comando “**GCX**”. Desta forma, os seguintes parâmetros não podem ser alterados em “**GOX**”:

- Valor da transação (**SPE_AMOUNT**);
- Valor do saque (**SPE_CASHBACK**); e
- Tipo de transação (**SPE_TRNTYPE**).

Caso o processamento decida por pedir PIN *online*, isso é feito da mesma forma descrita na seção 6.9.3.1 (*Cardholder Verification*).

⚠ Caso a verificação de portador requisite PIN Online, o *bypass* não deve ser permitido para cartões CTLS (a tecla [OK/ENTRA] deve ser ignorada se o campo de entrada estiver vazio).

➡ Dados de Saída

Caso o processamento seja bem-sucedido, a resposta ao comando “**GOX**” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|--------------|--|
| PP_GOXRES | Decisão do processamento do cartão CTLS (já efetuado em “ GCX ”): “0xxxxx” = Transação aprovada <i>offline</i> ; “1xxxxx” = Transação negada; ou “2xxxxx” = Transação requer autorização <i>online</i> . Resultado da verificação de portador: “x1xxxx” = Deve-se coletar assinatura em papel. “xx2xxx” = PIN capturado para verificação <i>online</i> . “xxx1xx” = Verificação de portador efetuada no dispositivo móvel. |
| PP_PINBLK | PIN criptografado para validação <i>online</i> (somente se PP_GOXRES = “xx2xxx”). |
| PP_KSN | KSN da chave DUKPT usada na criptografia de PIN <i>online</i> (somente se PP_GOXRES = “xx2xxx” e SPE_MTHDPIN = “2” ou “3”). |

| Id. do Campo | Dado retornado |
|-------------------|--|
| <u>PP_EMVDATA</u> | <p>Dados definidos pela <i>tags</i> em <u>SPE_TAGLIST</u>, no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3</p> <p>Este campo é mandatário sempre que <u>SPE_TAGLIST</u> existir no comando, mesmo que nenhum objeto seja encontrado (caso em que é retornado com tamanho zerado).</p> <p>IMPORTANTE:</p> <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |

➤ Situações de exceção:

| RSP_STAT | Situação |
|--------------|--|
| ↵ST_INVPARAM | <ul style="list-style-type: none"> ▪ O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. ▪ <u>SPE_AMOUNT</u>, <u>SPE_CASHBACK</u> ou <u>SPE_TRNTYPE</u> foram alterados. |
| ↵ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando "<u>GCX</u>" não foi executado previamente com sucesso. ▪ Comando "<u>GCX</u>" foi chamado previamente, porém retornou <u>PP_CARDTYPE</u> diferente de "03" e "06". ▪ Comando "<u>GOX</u>" já foi chamado. |
| ↵ST_ERRKEY | Foi requerida captura de PIN <i>online</i> , mas a chave indicada está ausente ou corrompida. |
| ↵ST_CANCEL | A tecla [CANCELA] foi pressionada pelo portador durante a tela de captura de PIN. |

6.9.4. Comando “FCX”

Este comando é chamado pelo SPE caso “GOX” tenha requerido aprovação *online* ou, opcionalmente, caso a transação já tenha sido aprovada ou negada *offline* e haja necessidade de processamento de *Issuer Scripts*. Os tratamentos envolvidos são diferentes dependendo do tipo de chip, descritos a seguir.

OBS: No caso de ICC, este comando deve sempre desligar sua alimentação ao final do processamento.

6.9.4.1. ICC EMV - encerrada *offline*

Caso a transação já tenha sido aprovada ou negada *offline* pelo cartão em “GOX” (PP_GOXRES ≠ “2xxxx”), o SPE pode acionar este comando para compatibilização de fluxo ou para efetuar *Issuer Script Processing* em ICC.

Esta segunda funcionalidade é específica de algumas “bandeiras” ou Redes Credenciadoras que definem processos de atualização de cartão em transação não financeira, caso em que é feita uma conexão com a Rede Credenciadora independentemente do resultado do *Card Action Analysis* (1st GENERATE AC). Desta forma, se o campo SPE_EMVDATA for recebido, o pinpad deve efetuar os seguintes processos:

- **Issuer Authentication**

Caso o campo SPE_EMVDATA contenha o objeto TLV de *tag 91h (Issuer Authentication Data)*, este deve ser fornecido ao Kernel EMV para o processo de *Issuer Authentication*.

- **Issuer Script Processing**

Caso o campo SPE_EMVDATA contenha um ou mais objetos TLV de *tag 71h (Issuer Script Template 1)*, estes devem ser fornecidos ao Kernel EMV para a execução do *Issuer Script Processing*.

- **Completion**

Dado que a transação foi finalizada *offline* pelo cartão no 1st GENERATE AC, o pinpad deve ignorar o resultado do comando 2nd GENERATE AC, mesmo que este retorne erro.

Se SPE_EMVDATA não for recebido, nenhuma operação é feita com o ICC.

➔ Dados de Saída

Neste caso, a resposta ao comando “FCX” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|--------------|--|
| PP_FCXRES | “0xx” → Se PP_GOXRES = “0xxxxx” e SPE_FCXOPT = “0xxx”; “1xx” → Outra situação. |
| PP_ISRESULTS | Resultado do processamento de scripts (<i>Issuer Script Results</i>), somente se houver. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↪ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando “GOX” não foi executado previamente com sucesso. ▪ Comando “FCX” já foi chamado. |
| ↪ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

⚠ Nesta situação, este comando nunca deverá retornar erro de processamento com o cartão, mesmo que este seja retirado. Qualquer problema de processamento deve ser refletido somente em **PP_ISRESULTS**.

6.9.4.2. ICC EMV - impossibilidade de conexão *online*

Caso o comando receba **SPE_FCXOPT** com valor “2xxx”, a conexão com a Rede Credenciadora não foi bem-sucedida.

Neste caso, o pinpad deve acionar o processo conhecido como *Unable to Go Online* no Kernel EMV.

⚠ Se **SPE_GOXOPT** = “x1xxx”, o Kernel EMV deverá sugerir negação (AAC) ao cartão!

➔ Dados de Saída

A resposta ao comando “**FCX**” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|-------------------|--|
| PP_FCXRES | Decisão do cartão no comando 2nd GENERATE AC: “0xx” → Cartão retornou TC (aprovada <i>offline</i>); “1xx” → Cartão retornou AAC (negada <i>offline</i>). |
| PP_EMVDATA | Dados definidos pela <i>tags</i> em SPE_TAGLIST , no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3 Este campo é mandatário sempre que SPE_TAGLIST existir no comando, mesmo que nenhum objeto seja encontrado (caso em que é retornado com tamanho zerado). IMPORTANTE: <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|------------------|---|
| ↳ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↳ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando “GOX” não foi executado previamente com sucesso. ▪ Comando “FCX” já foi chamado. |
| ↳ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↳ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↳ST_NOCARD | O cartão foi removido. |
| ↳ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↳ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |
| ↳ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.9.4.3. ICC EMV - autorização *online* bem-sucedida

Caso o comando receba SPE_FCXOPT com diferente de “2xxx”, considera-se que a conexão com a Rede Credenciadora foi bem-sucedida e uma resposta de autorização foi recebida.

➔ Issuer Authentication

Caso o campo SPE_EMVDATA contenha o objeto TLV de *tag* 91h (*Issuer Authentication Data*), este deve ser fornecido ao Kernel EMV para o processo de *Issuer Authentication*.

⚠ Caso o cartão suporte *Issuer Authentication* em seu *AIP* e, caso o comando EXTERNAL AUTHENTICATE retorne qualquer status SW1/SW2 de 9000h (por exemplo 6985h), o pinpad não deverá interromper a transação, seguindo com o processamento considerando falha na autenticação:

➔ Issuer Script Processing

Caso o campo SPE_EMVDATA contenha um ou mais objetos TLV de *tag* 71h e 72h (*Issuer Scripts*), estes devem ser fornecidos ao Kernel EMV para a execução do *Issuer Script Processing*.

➔ Completion

O Kernel EMV deve fornecer ao Kernel EMV a decisão da Rede Credenciadora:

- Se SPE_FCXOPT = “0xxx”, sugerir aprovação ao cartão; ou
- Se SPE_FCXOPT ≠ “0xxx”, sugerir negação ao cartão.

Em qualquer dos casos, o valor de **SPE_ARC** deve ser fornecido ao Kernel EMV como *Authorization Response Code* (tag 8Ah).

➔ Dados de Saída

A resposta ao comando “**FCX**” deve devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|---------------------|--|
| PP_FCXRES | Decisão do cartão no comando 2nd GENERATE AC: “0xx” → Cartão retornou TC (aprovada); “1xx” → Cartão retornou AAC (negada). |
| PP_EMVDATA | Dados definidos pela <i>tags</i> em SPE_TAGLIST , no formato TLV (se existentes), na mesma ordem em que foram solicitados. Ver restrições na seção 6.8.6.3 Este campo é mandatório sempre que SPE_TAGLIST existir no comando, mesmo que nenhum objeto seja encontrado (caso em que é retornado com tamanho zerado). IMPORTANTE: <ul style="list-style-type: none"> ▪ O SPE pode solicitar objetos proprietários do cartão cujas <i>tags</i> não são definidas pelas normas EMV, sendo que o pinpad deve devolvê-los corretamente caso existam. ▪ Caso o SPE solicite um objeto da Tabela de AID, o pinpad deve procurá-lo primeiramente no kernel EMV e, <u>se não existir</u>, devolver o valor do registro da tabela que foi usado na parametrização da transação. |
| PP_ISRESULTS | Resultado do processamento de scripts (<i>Issuer Script Results</i>), somente se houve. |

➔ Situações de exceção:

| RSP_STAT | Situação |
|------------------|---|
| ↳ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↳ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando “GOX” não foi executado previamente com sucesso. ▪ Comando “FCX” já foi chamado. |
| ↳ST_DUMBCARD | ICC inserido, mas não responde (<i>chip</i> com defeito ou ausente). |
| ↳ST_ERRCARD | Erro de comunicação entre o pinpad e o ICC. |
| ↳ST_NOCARD | O cartão foi removido. |
| ↳ST_CARDPROBLEMS | ICC com problemas. Esse status é válido para muitas ocorrências no processamento onde o ICC não se comporta conforme o esperado e a transação deve ser finalizada. |
| ↳ST_CARDINVDATA | O ICC comporta-se corretamente, porém possui dados inválidos ou inconsistentes. |

| RSP_STAT | Situação |
|-------------|---|
| ST_SECURITY | Tentativa de ataque de <i>replay</i> (ver seção 6.3.5). |

6.9.4.4. Cartão com *chip* sem contato (CTLS EMV)

Este comando será acionado pelo SPE caso a transação tenha requerido aprovação *online*. Como a decisão do CTLS já é tomada no comando “GCX”, normalmente o pinpad não faz mais nada, somente “rebatendo” a decisão informada no comando.

➔ Issuer Script Processing

Para cartões Visa payWave e Discover D-PAS, entretanto, há ainda a possibilidade de se efetuar o processamento de *Issuer Scripts* para manutenção do cartão. Esta operação será efetuada somente se:

- O campo **SPE_EMVDATA** tenha sido recebido;
- **T1_CTLSISSSCR** = “1”; e
- O CTLS processado seja:
 - ⇒ Um VISA payWave que retorne no 2º byte do CTQ (tag 9F6Ch) o bit 7 indicando “*Card supports Issuer Update Processing at the POS*”; ou
 - ⇒ Um Discover D-PAS que retorne no 2º byte do *Card Processing Requirements* (tag ‘9F71’) o bit 5 indicando “*Issuer Update Processing supported*”

Exclusivamente neste caso o pinpad mostrará a seguinte mensagem solicitando a reaproximação do cartão (considerando o objeto *Language Preference*, da mesma forma que na seção 6.8.6.1):

| Português | Inglês | Espanhol |
|---|---|--|
|  <p>POR FAVOR REAPROXIME O CARTÃO</p> |  <p>PLEASE PRESENT CARD AGAIN</p> |  <p>POR FAVOR ACERQUE TARJETA NUEVAMENTE</p> |

Neste momento, o pinpad envia a seguinte mensagem de notificação ao SPE de forma a alertar o operador:

| NTM_MSG [1..16] | NTM_MSG [17..32] |
|-----------------|------------------|
| SOLICITE CARTAO | NOVAMENTE |

Observações:

- Enquanto o pinpad aguarda um cartão, o comando pode ser cancelado pelo SPE através do envio de um byte «CAN» (caso em que o pinpad responde com «EOT»). Neste caso o display deve ser limpo e o comando finalizado.

- Enquanto o pinpad aguarda um cartão, o processo pode ser cancelado pelo operador (através da tecla [CANCELA]).
- O pinpad deve simplesmente desistir desta operação se o tempo de espera por um cartão ultrapassar o valor determinado (SPE_TIMEOUT ou 2 minutos se este campo não foi recebido).
- O *display* deve ser apagado assim que apresentado um cartão ou em caso cancelamento ou outro erro.

⚠ Este processo não deve influenciar a decisão da transação, sendo somente um procedimento de manutenção do emissor. Desta forma, o comando "**FCX**" deve sair com sucesso (↵ST_OK) mesmo que ocorra algum erro de processamento, "timeout" ou que a [operação seja interrompida pela tecla \[CANCELA\]](#).

➡ Dados de Saída

Neste caso, a resposta ao comando "**FCX**" deve simplesmente devolver os seguintes campos:

| Id. do Campo | Dado retornado |
|---------------------|---|
| <u>PP_FCXRES</u> | "0xx" → Se <u>SPE_FCXOPT</u> = "0xxx" "1xx" → Se <u>SPE_FCXOPT</u> ≠ "0xxx" |
| <u>PP_EMVDATA</u> | Retornar este campo <u>vazio</u> somente se <u>SPE_TAGLIST</u> existir no comando, caso contrário ele não deve ser retornado. |
| <u>PP_ISRESULTS</u> | <i>Issuer Script Results</i> , se houve processamento de scripts. |

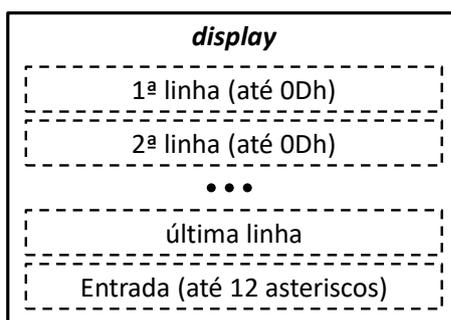
➡ Situações de exceção:

| RSP_STAT | Situação |
|--------------|---|
| ↵ST_INVPARAM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |
| ↵ST_INVCALL | <ul style="list-style-type: none"> ▪ Comando "GOX" não foi executado previamente com sucesso. ▪ Comando "FCX" já foi chamado. |

6.9.5. Regras gerais

6.9.5.1. Telas de captura de PIN

As telas de captura de PIN seguem a mesma regra dos comandos obsoletos (ver [seção 6.8.6.1](#)), lembrando-se que, no comando “GOX”, a mensagem de captura de PIN pode vir opcionalmente do SPE através do campo **SPE_DSPMSG**. Neste caso, a tela de captura de PIN passa a ser mais flexível, conforme figura a seguir:



6.9.5.2. Tabelas de AID (resolução de conflitos)

As diversas Redes Credenciadoras podem processar cartões ICC/CTLS das mesmas “bandeiras”, portanto os registros das **Tabelas de AID** associados às diferentes redes podem conter os mesmos AIDs (ou seja, registros com campo **T1_AID** idêntico para diferentes **T1_ACQ**). Dependendo da forma como é feita a filtragem dos “registros candidatos” no comando “GCX”, a lista resultante pode conter AIDs conflitantes e estes não podem ser simplesmente repassados ao Kernel EMV, que não saberá como tratá-los.

Para evitar esta situação indesejável no processamento EMV, o pinpad deverá desfazer os eventuais conflitos na lista de “registros candidatos” e, caso o processo de seleção de aplicação indique um AID aceito por mais de uma Rede Credenciadora, esta situação é informada ao SPE através do campo **PP_AIDTABINFO**.

Adicionalmente, os registros “conflitantes” podem possuir parâmetros diferentes entre as Redes Credenciadoras, sendo que o Kernel EMV pode precisar de alguns deles logo no processamento do comando “GCX”, antes mesmo de saber qual Rede Credenciadora será usada para concluir o processamento no comando “GOX” (especialmente no caso de CTLS). Para resolver essa questão, o pinpad deve “aglutinar” os parâmetros destes registros ao fornecer os dados ao Kernel EMV de acordo com as seguintes regras:

| Id. do Campo | Dado retornado |
|---|--|
| T1_APPVER1 T1_APPVER2 T1_APPVER3 | Usar os valores fornecido pelos registros conflitantes, caso seja os mesmos. Em caso de divergência, adotar “0000”. |
| T1_TRMCNTRY | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar “076” (Brasil). |

| Id. do Campo | Dado retornado |
|---------------------------------|---|
| <u>T1_TRNCURR</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "986" (Real). |
| <u>T1_TRNCRREXP</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "2". |
| <u>T1_MERCHID</u> | Adotar qualquer um dos valores. |
| <u>T1_MCC</u> | Adotar qualquer um dos valores. |
| <u>T1_TRMID</u> | Adotar qualquer um dos valores. |
| <u>T1_TRMCAPAB</u> | "E" ("AND") binário dos valores dos registros conflitantes. |
| <u>T1_ADDTRMCP</u> | "E" ("AND") binário dos valores dos registros conflitantes. |
| <u>T1_TRMTYP</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "21". |
| <u>T1_TACDEF</u> | Não importa. |
| <u>T1_TACDEN</u> | Não importa. |
| <u>T1_TACONL</u> | Não importa. |
| <u>T1_FLRLIMIT</u> | Adotar o menor valor entre os registros conflitantes. |
| <u>T1_TCC</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "R". |
| <u>T1_CTLSZEROAM</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "0". |
| <u>T1_CTLSMODE</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar qualquer um dos valores. |
| <u>T1_CTLSTRNLIM</u> | Adotar o menor valor entre os registros conflitantes. |
| <u>T1_CTLSFLRLIM</u> | Adotar o menor valor entre os registros conflitantes. |
| <u>T1_CTLSCVMLIM</u> | Adotar o menor valor entre os registros conflitantes. |
| <u>T1_CTLSAPPVER</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "0000". |
| <u>T1_TDOLDEF</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "0000...00". |
| <u>T1_DDOLDEF</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "9F370400...00". |
| <u>T1_CTLSTACDEF</u> | "OU" ("OR") binário dos valores dos registros conflitantes. |
| <u>T1_CTLSTACDEN</u> | "OU" ("OR") binário dos valores dos registros conflitantes. |
| <u>T1_CTLSTACONL</u> | "OU" ("OR") binário dos valores dos registros conflitantes. |
| <u>T1_CTLSTRMCP</u> | "E" ("AND") binário dos valores dos registros conflitantes. |
| <u>T1_MOBCVM</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "0". |
| <u>T1_CTLSADDTTC</u> | "E" ("AND") binário dos valores dos registros conflitantes. |

| Id. do Campo | Dado retornado |
|----------------------|--|
| <u>T1_CTLSMBTLIM</u> | Adotar o menor valor entre os registros conflitantes. |
| <u>T1_CTLSISSCR</u> | Usar o valor fornecido pelos registros conflitantes, caso seja o mesmo. Em caso de divergência, adotar "0". |

 Este processo de "aglutinação" é necessário apenas para o comando "GCX", momento em que a Rede Credenciadora ainda não está definida. No comando "GOX" devem-se usar os parâmetros inalterados da Rede Credenciadora definida por SPE_ACQREF.

6.9.5.3. Valor da transação, dados protegidos, objetos do cartão

As regras definidas para o comando "GCR" nas seções 6.8.6.2 ("Valores da Transação") 6.8.6.3 ("Dados protegidos") e 6.8.6.4 ("Objetos do cartão") também devem ser observadas para o comando "GCX".

6.10. Processamento dos comandos genéricos

Esta seção descreve os tratamentos internos do pinpad para os comandos descritos na **seção 3.8**.

6.10.1. Comando “GEN/02/K3”

Este comando é de processamento simples e não há nenhuma especificidade a ser descrita nesta seção.

➔ Situações de exceção:

| RSP_STAT | Situação |
|-------------|---|
| ↪ST_INVPARM | O formato do comando ou o conteúdo dos parâmetros recebidos não estão condizentes com esta especificação. |

6.10.2. Comando “GEN/03/02”

Este comando faz a mesma função dos comandos “GED” e “GEN/04/04”, porém com diferentes *layouts* de dados (ver **seção 6.9.2**).

Diferentemente de “GED”, este comando só pode ser utilizado para cartão ICC EMV após a execução bem-sucedida de um comando “GCR”.

6.10.3. Comando “GEN/04/01”

Este comando é similar ao comando “CHP” para CHP_OPER = “0” ou “1”, porém com diferentes *layouts* de dados (ver **seções 6.5.2.1 e 6.5.2.2**), além de permitir o “*warm reset*” e a consulta do status do cartão.

No recebimento de um pedido de “*power on*” (G0401_OPER = “1”), o pinpad realiza um processo de ativação conforme norma ISO7816, incluindo o tratamento do ATR e uma negociação dos parâmetros de protocolo, caso possível. Caso o ICC esteja ativo, deve ser efetuado o processo de desativação (“*power off*”) antes de efetuar o “*power on*”.

Ao receber um pedido de “*power off*” (G0401_OPER = “3”), o pinpad realiza o processo de desativação do chip conforme norma ISO7816.

No caso de um pedido de “*reset*” (G0401_OPER = “2”), o pinpad efetua um “*warm reset*” (através do pino de *reset* do ICC) e prossegue para o tratamento do ATR e negociação conforme descrito acima. Se um “*warm reset*” não for possível ou se ocorrer algum erro, deve ser efetuado um “*cold reset*” (através de uma sequência “*power off*” e “*power on*”).

6.10.4. Comando “GEN/04/02”

Este comando faz a mesma função do comando “CHP” para CHP_OPER = “2”, porém com diferentes *layouts* de dados (ver **seção 6.5.2.3**).

6.10.5. Comando “GEN/04/03”

Este comando faz a mesma função do comando “CHP” para CHP_OPER = “3”, porém com diferentes *layouts* de dados (ver **seção 6.5.2.4**).

6.10.6. Comando “GEN/04/04”

Este comando faz a mesma função dos comandos “GED” e “GEN/03/02”, porém com diferentes *layouts* de dados (ver **seção 6.9.2**).

Diferentemente de “GED”, este comando só pode ser utilizado para cartão ICC EMV após a execução bem-sucedida de um comando “GCR”.

6.11. Teclas especiais

Considerando-se que o pinpad é multiaplicação (conforme descrito na **seção 6.1.1**), existem instalados aplicativos independentes no mesmo equipamento, cada um com uma versão diferente, podendo ter sido certificados em momentos distintos. Dessa maneira, para facilitar a operação de técnicos de campo, é importante que exista uma forma simples de verificar as versões das aplicações instaladas.

Assim, deve-se permitir a consulta das versões quando o pinpad estiver em estado ocioso (aguardando comando pelo protocolo serial) através da digitação de uma sequência de teclas pré-determinada, conforme tabela:

| Aplicação | Sequência de teclas | Informações no <i>display</i> |
|--------------|-----------------------------------|--|
| Gerenciadora | [CLEAR/BACKSP] → [OK/ENTRA] → [0] | <u>PP_MODEL</u> <u>PP_MANVERS</u> <u>PP_SPECVER</u> |
| Abecs | [CLEAR/BACKSP] → [OK/ENTRA] → [1] | <u>PP_APPVERS</u> <u>PP_SPECVER</u> <u>PP_KRNLVER</u> <u>PP_CTLsver</u> |
| Extensão | [CLEAR/BACKSP] → [OK/ENTRA] → [9] | <u>PP_GENVERS</u> <u>PP_SPECVER</u> |

Pressionada a sequência correta, o pinpad deve soar um “bip” e apresentar no *display* as informações descritas na tabela acima (uma por linha). As informações permanecem no *display* indefinidamente, até que um evento qualquer o atualize.

OBSERVAÇÃO: O fabricante do pinpad pode definir outras sequências de teclas para consulta de informações proprietárias alheias a esta especificação.

7. Informações Complementares

As seções deste capítulo apresentam informações complementares úteis para a compreensão desta especificação.

7.1. Codificação TLV

Como definido pela norma ISO/IEC 8825, um objeto de dados BER-TLV consiste de 2 a 3 campos consecutivos:

- O campo “tag” (T) consiste de um ou mais bytes consecutivos.
- O campo “length” (L) consiste de um ou mais bytes consecutivos. Ele indica o tamanho do campo seguinte.
- O campo “value” (V) indica o valor do objeto de dados. Se L = 00h, o campo “value” não está presente.

Os subitens a seguir definem a codificação destes campos.

7.1.1. Codificação do campo “tag” (T)

A tabela a seguir descreve o primeiro byte do campo “tag” de um objeto BER-TLV:

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Significado |
|----|----|----|------------------|----|----|----|----|-------------------------|
| x | x | x | | | | | | Classe e tipo de objeto |
| | | | 1 | 1 | 1 | 1 | 1 | Ver bytes subsequentes |
| | | | Outro valor < 31 | | | | | Número da “tag” |

De acordo com a ISO/IEC 8825, a tabela a seguir define as regras de codificação dos bytes subsequentes de uma “tag” BER-TLV quando os bits b5 a b1 do primeiro byte são iguais a ‘11111’:

| b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Significado |
|----|----|----|--------------------|----|----|----|----|----------------------------|
| 1 | | | | | | | | Há outro byte seguinte |
| 0 | | | | | | | | Último byte da “tag” |
| | | | Qualquer valor > 0 | | | | | (Parte do) número da “tag” |

Antes, entre, ou depois de objetos de dados codificados em TLV, podem existir bytes 00h sem nenhum significado.

7.1.2. Codificação do campo “length” (L)

Quando o bit “b8” do byte mais significativo do campo “length” é **0**, o campo “length” consiste de apenas um byte. Os bits “b7” a “b1” codificam o tamanho do campo “value”, na faixa de 1 a 127.

Quando o bit “b8” do byte mais significativo do campo “length” é **1**, os bits “b7” a “b1” codificam o número de bytes subsequentes no campo “length”. Estes bytes subsequentes codificam um valor inteiro que representa o tamanho do campo “value”. Desta forma, são necessários dois bytes para representar até 255 bytes no campo “value”.

7.2. Cálculo de CRC

Sempre que esta especificação se referir ao cálculo de CRC, está sendo considerado o **CRC-16-CCITT**, com polinômio gerador $x^{16} + x^{12} + x^5 + x^0$.

O código em linguagem C a seguir ilustra esta implementação:

```
#define CRC_MASK 0x1021          /* x^16 + x^12 + x^5 + x^0 */

UINT16 CRC_Calc (unsigned char *pbData, int iLength)
{
    UINT16 wData, wCRC = 0;
    int i;

    for ( ; iLength > 0; iLength--, pbData++) {
        wData = (UINT16) (((UINT16) *pbData) << 8);
        for (i = 0; i < 8; i++, wData <<= 1) {
            if ((wCRC ^ wData) & 0x8000)
                wCRC = (UINT16) ((wCRC << 1) ^ CRC_MASK);
            else
                wCRC <<= 1;
        }
    }
    return wCRC;
}
```

7.3. Display do pinpad

7.3.1. Uso pelos comandos

Os comandos especificados neste documento podem usar ou não o *display* do pinpad para apresentação de mensagens, dependendo da situação. A tabela a seguir lista todos os comandos que podem utilizar o display e de que forma.

| Comando | Uso do <i>display</i> |
|---------|---|
| "OPN" | O <i>display</i> é apagado e o <i>backlight</i> ativado. |
| "CLO" | O <i>backlight</i> é desativado e a mensagem CLO_MSG é deixada no <i>display</i> . |
| "CLX" | O <i>backlight</i> é desativado e a mensagem SPE_DSPMSG ou a imagem SPE_MFNAME são deixadas no <i>display</i> . |
| "CHP" | O <i>display</i> somente é usado se houver captura de PIN (CHP_OPER = "3"), sendo apagado ao final, seja a captura bem ou malsucedida. Nas outras modalidades do comando (CHP_OPER ≠ "3") o <i>display</i> não deve ser modificado ou apagado. |
| "DEX" | A mensagem DEX_MSG é deixada no <i>display</i> . |
| "DSP" | A mensagem DSP_MSG é deixada no <i>display</i> . |
| "GCD" | O <i>display</i> é usado no processo de captura de dados e é sempre apagado ao final, seja a captura bem ou malsucedida. |
| "GPN" | O <i>display</i> é usado no processo de captura de PIN e é sempre apagado ao final, seja a captura bem ou malsucedida. |
| "MNU" | O <i>display</i> é usado para apresentação do menu e é sempre apagado ao final, seja a seleção bem ou malsucedida. |
| "RMC" | Ao final do processamento, a mensagem RMC_MSG é deixada no <i>display</i> . |
| "DSI" | A imagem indicada em SPE_MFNAME é deixada no <i>display</i> . |
| "TLR" | Opcionalmente, o pinpad pode deixar no <i>display</i> uma mensagem informativa indicando a carga de tabelas em curso. |
| "TLE" | Apaga o <i>display</i> somente se este foi modificado em "TLR", caso contrário não o modifica. |
| "GCR" | Usa o <i>display</i> para solicitar cartão e para apresentar o menu de seleção de aplicação. <ul style="list-style-type: none"> ▪ Para ICC ou CTLS processados com sucesso, deixa no <i>display</i> uma mensagem indicando a aplicação selecionada. ▪ Em caso de erro, apaga o <i>display</i> ao final. |
| "GOC" | Se requerido, usa o <i>display</i> para captura de PIN, apagando-o ao final, seja a captura bem ou malsucedida. Se não houver captura de PIN, o <i>display</i> |

| | |
|--------------------|---|
| | não deve ser modificado. |
| "GCX" | Usa o <i>display</i> para solicitar cartão e para apresentar o menu de seleção de aplicação. <ul style="list-style-type: none"> ▪ Para ICC ou CTLS processados com sucesso, deixa no <i>display</i> uma mensagem indicando a aplicação selecionada. ▪ Em caso de erro, apaga o <i>display</i> ao final. |
| "GOX" | Se requerido, usa o <i>display</i> para captura de PIN, apagando-o ao final. Se não houver captura de PIN, o <i>display</i> não deve ser modificado. |
| "FCX" | Pode usar o <i>display</i> para solicitação de cartão no caso de CTLS com <i>Issuer Script Processing</i> (ver seção 6.9.4.4), sendo apagado ao final. Para outras situações, o <i>display</i> não deve ser modificado. |
| "GEN/04/03" | O <i>display</i> é usado no processo de captura de PIN e é sempre apagado ao final, seja a captura bem ou malsucedida. |

▲ Os demais comandos que não estão listados nesta tabela não devem apagar nem modificar o conteúdo do *display*.

7.3.2. Tabela de Caracteres

Para apresentação de mensagens de *display* no pinpad, esta especificação prevê o uso do *codepage ISO/IEC 8859-1*, cujos principais símbolos estão definidos na tabela a seguir:

| | | | | | | | | | |
|----------|---|----------|---|----------|---|----------|---|----------|----|
| 032(20h) | | 033(21h) | ! | 034(22h) | " | 035(23h) | # | 036(24h) | \$ |
| 037(25h) | % | 038(26h) | & | 039(27h) | ' | 040(28h) | (| 041(29h) |) |
| 042(2Ah) | * | 043(2Bh) | + | 044(2Ch) | , | 045(2Dh) | - | 046(2Eh) | . |
| 047(2Fh) | / | 048(30h) | 0 | 049(31h) | 1 | 050(32h) | 2 | 051(33h) | 3 |
| 052(34h) | 4 | 053(35h) | 5 | 054(36h) | 6 | 055(37h) | 7 | 056(38h) | 8 |
| 057(39h) | 9 | 058(3Ah) | : | 059(3Bh) | ; | 060(3Ch) | < | 061(3Dh) | = |
| 062(3Eh) | > | 063(3Fh) | ? | 064(40h) | @ | 065(41h) | A | 066(42h) | B |
| 067(43h) | C | 068(44h) | D | 069(45h) | E | 070(46h) | F | 071(47h) | G |
| 072(48h) | H | 073(49h) | I | 074(4Ah) | J | 075(4Bh) | K | 076(4Ch) | L |
| 077(4Dh) | M | 078(4Eh) | N | 079(4Fh) | O | 080(50h) | P | 081(51h) | Q |
| 082(52h) | R | 083(53h) | S | 084(54h) | T | 085(55h) | U | 086(56h) | V |
| 087(57h) | W | 088(58h) | X | 089(59h) | Y | 090(5Ah) | Z | 091(5Bh) | [|
| 092(5Ch) | \ | 093(5Dh) |] | 094(5Eh) | ^ | 095(5Fh) | _ | 096(60h) | ` |
| 097(61h) | a | 098(62h) | b | 099(63h) | c | 100(64h) | d | 101(65h) | e |
| 102(66h) | f | 103(67h) | g | 104(68h) | h | 105(69h) | i | 106(6Ah) | j |
| 107(6Bh) | k | 108(6Ch) | l | 109(6Dh) | m | 110(6Eh) | n | 111(6Fh) | o |

| | | | | | | | | | |
|----------|---|----------|---|----------|---|----------|---|----------|---|
| 112(70h) | p | 113(71h) | q | 114(72h) | r | 115(73h) | s | 116(74h) | t |
| 117(75h) | u | 118(76h) | v | 119(77h) | w | 120(78h) | x | 121(79h) | y |
| 122(7Ah) | z | 123(7Bh) | { | 124(7Ch) | | 125(7Dh) | } | 126(7Eh) | ~ |
| 192(C0h) | À | 193(C1h) | Á | 194(C2h) | Â | 195(C3h) | Ã | 199(C7h) | Ç |
| 200(C8h) | È | 201(C9h) | É | 202(CAh) | Ê | 205(CDh) | Í | 209(D1h) | Ñ |
| 211(D3h) | Ó | 212(D4h) | Ô | 213(D5h) | Õ | 218(DAh) | Ú | 220(DCh) | Ü |
| 224(E0h) | à | 225(E1h) | á | 226(E2h) | â | 227(E3h) | ã | 231(E7h) | ç |
| 232(E8h) | è | 233(E9h) | é | 234(EAh) | ê | 237(EDh) | í | 241(F1h) | ñ |
| 243(F3h) | ó | 244(F4h) | ô | 245(F5h) | õ | 250(FAh) | ú | 252(FCh) | ü |

⚠ Caso o pinpad não suporte este *codepage*, ele deverá “traduzir” as mensagens antes da apresentação no *display*, de forma a retirar acentos e cedilha.

7.4. Diferenças em relação à Biblioteca Compartilhada

Esta especificação preserva total compatibilidade com  **BibComp** para garantir o correto funcionamento do Pinpad Abecs na base legada de SPE.

Entretanto, esta especificação define diversas modificações e evoluções no pinpad, listadas a seguir:

- Incluídos diversos novos comandos, identificados como “**Comandos Abecs**”.
- Retiradas todas as referências a cartões TIBC e VISA Cash, bem como o retorno dos dados do SAM no comando “**GIN**”.
- Incluídas todas as particularidades dos anexos Rede Amex, Redecard e Cielo de  **BibComp**.
- Excluído suporte às chaves tipo DES.
- **Capítulo 6**: O processamento interno de todos os comandos foi revisado de forma a melhor padronizar o comportamento entre os pinpads dos diferentes fornecedores, além de cobrir situações deixadas em aberto em  **BibComp**.
- **Seção 2.2.1**: Os pacotes de dados tiveram seu tamanho aumentado de 1024 para 2049 bytes, porém somente para os “Comandos Abecs”.
- **Seção 2.2.1**: Incluído uso de byte «**DC3**» para trafegar bytes de controle no meio do pacote.
- **Seção 2.3.3**: Todas as mensagens de notificação enviadas ao SPE devem ser formatadas em 2 linhas de 16 colunas.
- **Seção 2.3.4**: Incluída resposta “**ERR**” para situações de exceção.
- **Seções 3.2.2 e 5.2**: Incluído processo de “Comunicação Segura”.
- **Seção 3.3.2**: Comando “**CHP**” também permite a codificação do PIN como uma sequência de dígitos ASCII.
- **Seção 3.3.2**: O tratamento dos casos ‘61xx’ e ‘6Cxx’ do protocolo T=0 não é mais opcional. O pinpad não poderá tratá-los, deixando essa responsabilidade ao SPE.
- **Seções 3.3.11**: O comando “**GPN**” aceita PAN de “02” a “19” dígitos.
- **Seções 3.6.1 e 4.1.1**: O “tipo de aplicação” (**T1_APPTYPE**) passa a ser livre e definido pelo SPE.
- **Seções 3.6.2**: Melhorada explicação sobre quais objetos TLV podem ser enviados ao pinpad no comando “**CNG**”.
- **Seções 3.6.3 e 3.6.4**: Os comandos “**GOC**” e “**FNC**” não retornam objetos EMV que contenham informações de trilhas ou PAN.
- **Seção 3.6.4**: Excluída referência a “emissor *partial grade*” no comando “**FNC**”.
- **Seções 3.6.4**: O comando “**FNC**” pode ser chamado mesmo em caso de negação *offline* em “**GOC**”. Isso é importante para o processamento de *Issuer Scripts*, caso definido pela Rede Credenciadora.
- **Seção 4.1.1**: Para CTLS, retirada a opção de menu de seleção caso exista mais de uma aplicação compatível (a aplicação CTLS sempre é selecionada automaticamente de acordo com sua prioridade).

- **Seção 4.2:** As versões de tabelas (antes nomeadas “*time-stamps*”) passam a ser alfanuméricas, permitindo qualquer formato definido pelo SPE.
- **Seção 5.1:** Definição detalhada quanto ao mapeamento de chaves de criptografia de PIN e dados (ausente  **BibComp**).
- **Seção 6.2.1:** O recebimento de uma mensagem válida também aborta comandos blocantes.
- **Seção 6.8.1.3:** O pinpad simplesmente não deve conferir versão de Tabelas EMV no comando “**GCR**” para CTLS, uma vez que o processamento completo deste tipo de cartão é feito imediatamente após sua detecção.
- **seção 7.3.2:** Todas as mensagens enviadas do SPE ao pinpad para apresentação no *display* permitem caracteres acentuados e cedilha.