Utilização de dispositivos de hardware portáteis em criptografia

Módulo IV

Manuel Barbosa @ DI Universidade do Minho

Setembro de 2001

Introdução

Parte I-A

Introdução

- Porquê utilizar dispositivos de hardware portáteis em aplicações de criptografia?
  - Para o armazenamento de material criptográfico sensivel, nomeadamente chaves privadas.
  - Para permitir a utilização de identidades digitais em qualquer lugar e não apenas no posto normal de trabalho.
- Dispositivos mais utilizados:
  - Smart-cards
  - iButtons
  - Computadores de bolso

iButtons

O que é um iButton?

- Dispositivo de hardware portátil fabricado pela Dallas Semiconductors.
- Chip + embalagem de aço inoxidável cilíndrica que disponibiliza os contactos do chip ao exterior.
- Aposta na miniaturização e simplificação da interface HW => leitores pequenos e baratos.
- Alimentação através da *parasitic power* das linhas de dados => apenas dois contactos: 1-Wire protocol.
- Cada iButton é identificado unicamente através de um número de série.

Que tipos de iButton existem?

- iButtons de Memória
  - DS1990A – É o mais básico: apenas inclui o numero de série de 64 bits comum a toda a família.
  - Série DS197x – EEPROM de tamanho variável.
  - Série DS198x – EPROM programável incrementalmente.
  - DS1963 – iButton para aplicações monetárias.
  - DS1991 – 1 kbit de memória r/w dividida em três àreas protegidas individualmente por password.
  - DS1994 – Memória não protegida e relógio.
  - DS192x – Memória e sensor de temperatura.
  - Restante da família DS199x – Apenas memória nõ volátil.

Que tipos de iButton existem?

- iButtons baseados na tecnologia JAVA Card para desenvolvimento de aplicações criptográficas:
  - Armazenamento de chaves privadas.
  - Identificação na Internet.
  - Assinatura digital de documentos.
  - Cifragem/Decifragem de mensagens.
  - Existe um anel especial que permite ao utilizador usar sempre o seu iButton !!!

- iButton Thermocron – Para acoplar a produtos e controlar o tempo e as condições de temperatura a que estiveram sujeitos e assim controlar a sua qualidade.

Vantagens e Desvantagens

+ Baixo custo: o iButton para criptografia custa apenas 15€.
+ Dimensões muito reduzidas.
+ Cada botão contém um código de ID único.
+ Podem ter alimentação autónoma e relógio.
+ Construidos em material bastante resistente.
+ Leitores muito baratos.
- São fabricados por apenas uma empresa.
- O utilizador comum não está familiarizado com eles.
- São fabricados nos EUA e portanto não podemos obter os botões com suporte para criptografia!!!

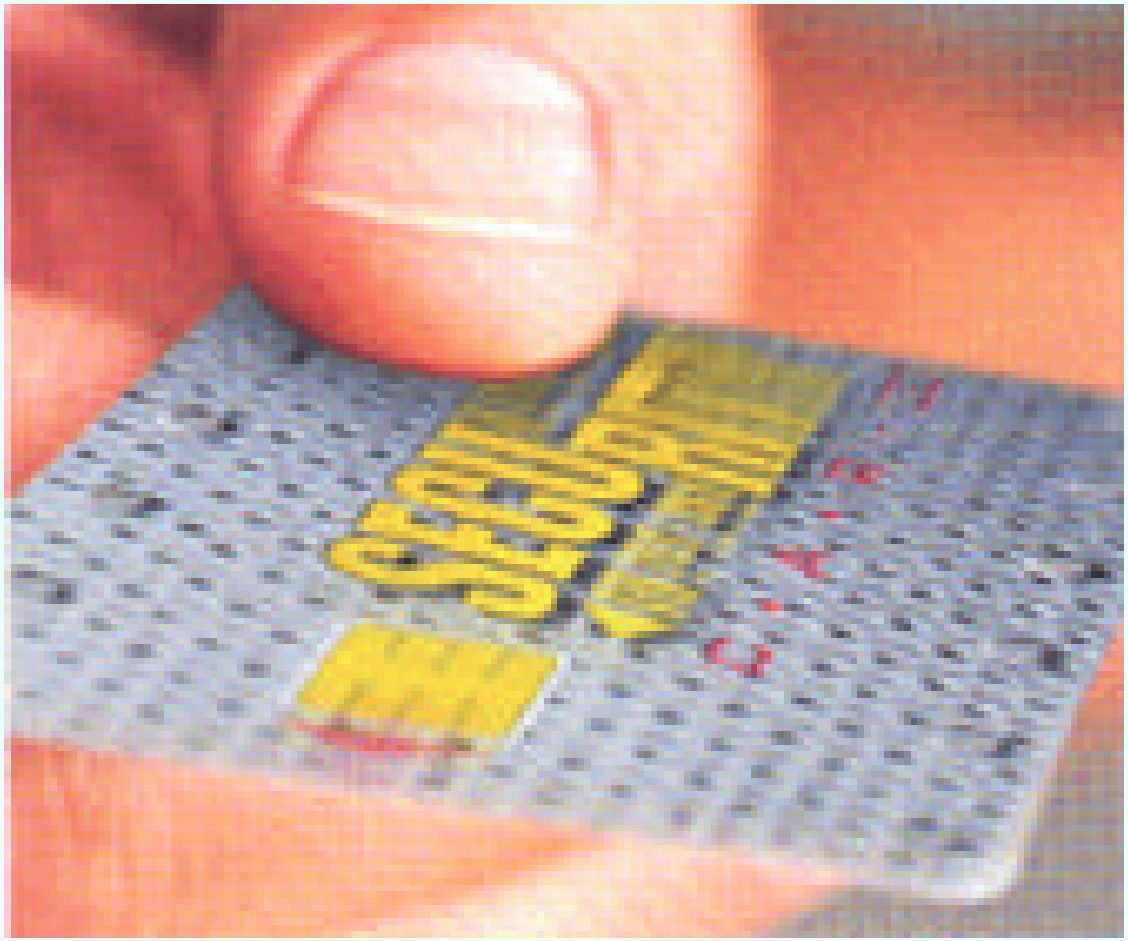
Computadores de Bolso

Todos temos uma ideia do que são...

Vantagens e Desvantagens

+ Possuem capacidade de processamento suficiente para executar todo o tipo de algoritmos criptográficos.
+ No desenvolvimento de aplicações é semelhante ao PC.
+ O porte de aplicações criptográficas existentes é relativamente simples.
+ Podem ser reutilizados para vários tipos de aplicações.
- Custo comparativamente muito elevado.
- Ainda não são suficientemente pequenos.
- Autonomia reduzida.
- Não existe um standard único (Windows CE vs. PalmOS)
- Não possuem um código de identificação único.

Smart-cards



O que são?

- O termo *smart-card* é um termo ambíguo, utilizado em vários contextos diferentes.
- Os smart-cards aqui referidos são os que a ISO (International Standards Organization) designa de Integrated Circuit Card (ICC).
- Um ICC é um cartão plástico de identificação com as medidas (85.6mm x 53.98mm x 0.76mm) que contém um circuito integrado.
- O termo “smart” advém da existência deste IC que confere ao cartão capacidade de processamento autónoma.

Tipos de Smart-cards

- Smart-cards com contactos / sem contactos.
- Cartões apenas com memória, utilizados para armazenamento de dados não confidenciais.
- Cartões com memória protegida, utilizados para armazenamento de dados com protecção através de um sistema de chaves com bloqueamento.
- Cartões com memória e CPU, utilizados para executar aplicações dentro do cartão.

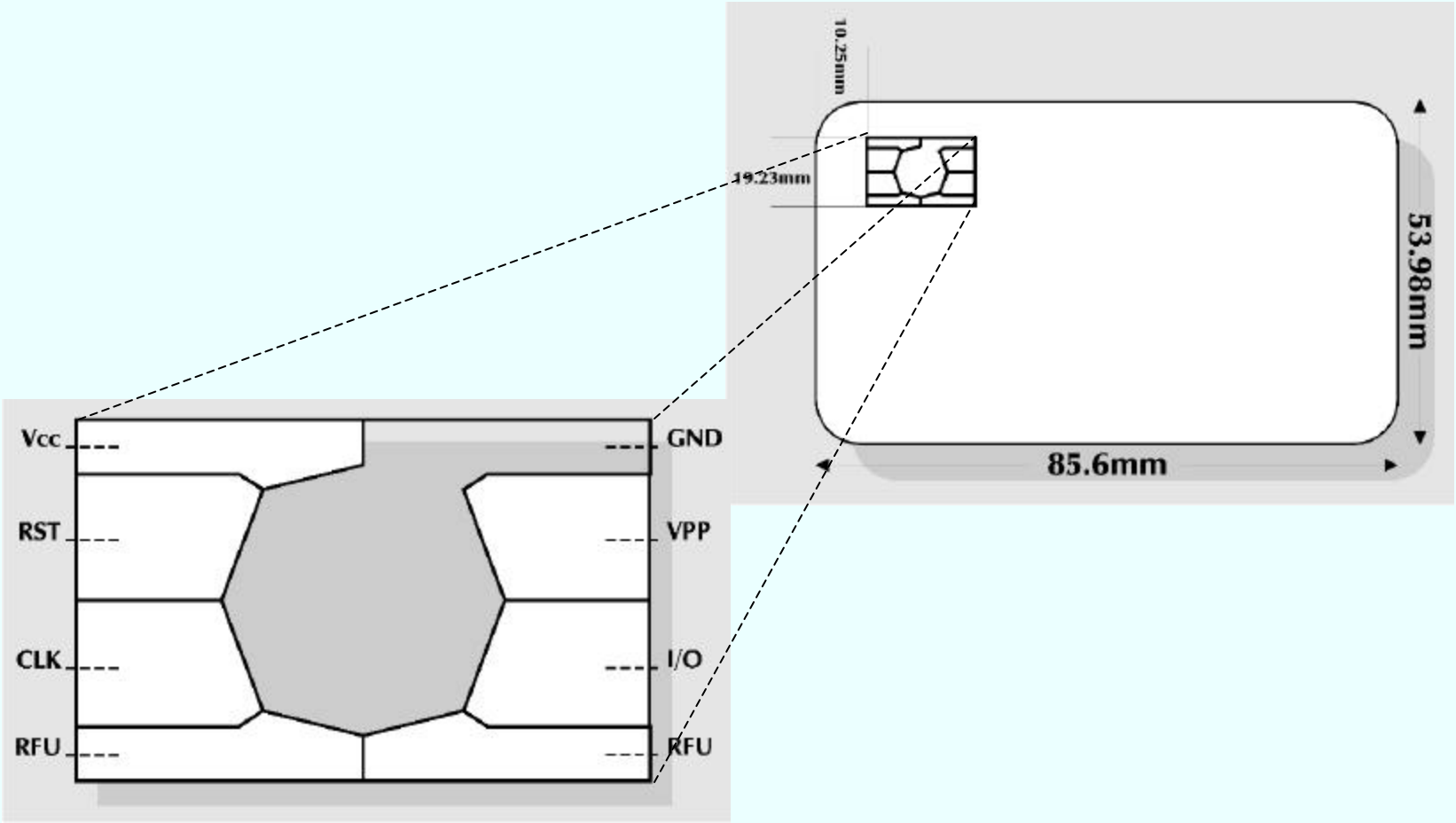
Vantagens e Desvantagens

+ Baixo custo: um Java Card com suporte para extensões de criptografia custa cerca de 15€.
+ Tem um formato familiar para o utilizador comum.
+ Encontram-se normalizados e são produzidos por vários fabricantes.
+ Não possuem alimentação própria e, por isso, a autonomia é apenas limitada pelo desgaste físico.
+ Existem vários standards para o desenvolvimento de aplicações que interagem com cartões.
- Não possuem alimentação própria e, por isso, não podem ter relógio.
- Não possuem um código de identificação único.
- Ainda não possuem capacidade de processamento suficiente para executar certos algoritmos criptográficos.

Smart cards: ISO7816

Parte I-B

Características Mecânicas

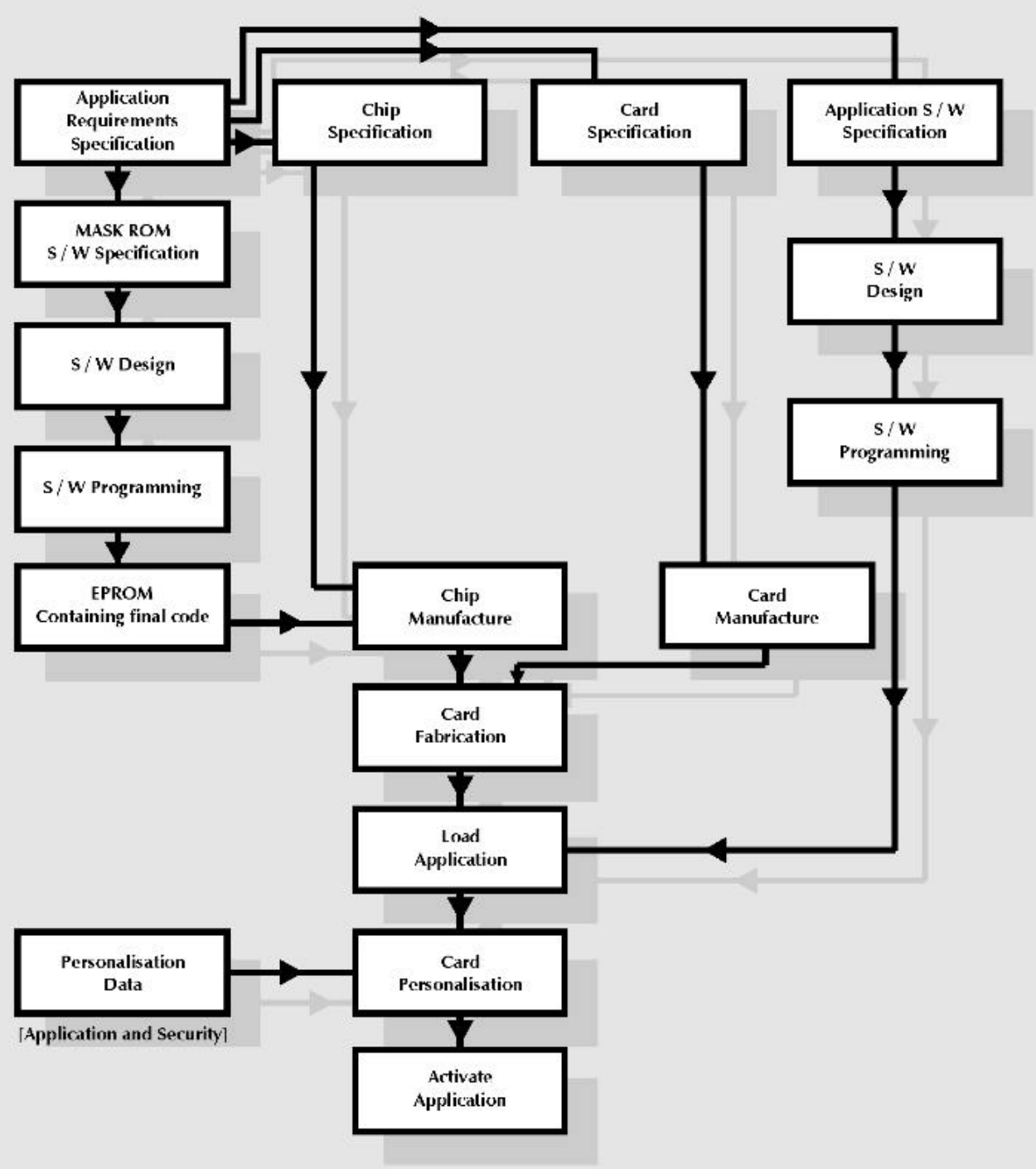


Conteudo do Chip

- A componente fundamental do chip é a memória.
- Em geral, um chip pode incluir memória de alguns destes tipos: ROM, PROM, EPROM, EEPROM e RAM.
- Lógica de controlo para assegurar as operações de comunicação básicas e, nos cartões um pouco mais evoluidos, assegurar a protecção da memória.
- Os smart-cards para aplicações mais exigentes contêm também uma CPU.

Processo de Fabrico de um Smart-card

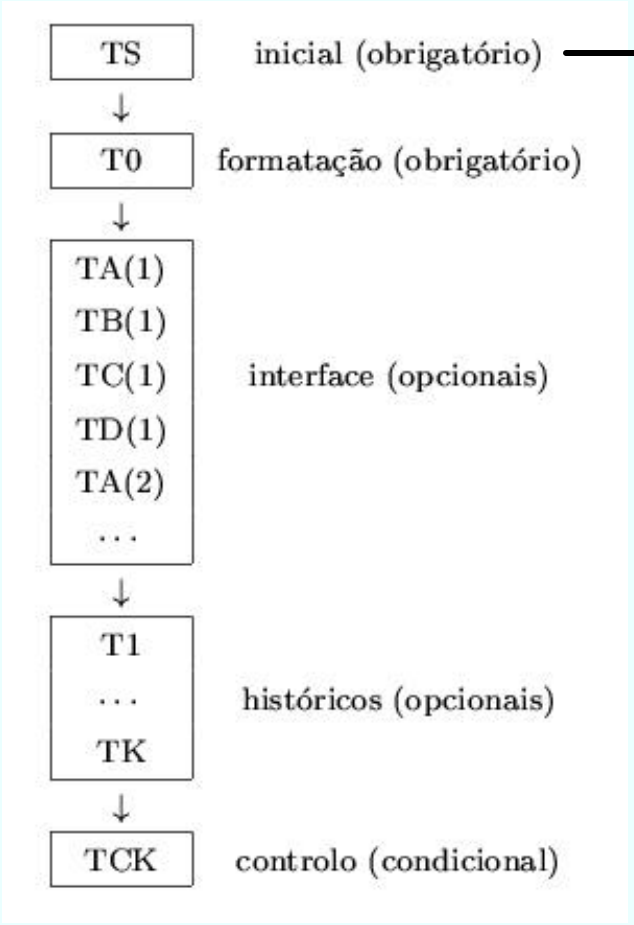
É preciso definir:
- CPU (6805/8051)
- Tamanho da ROM
- Tamanho da RAM
- Tipo e tamanho da memória não volátil (EPROM, EEPROM)
- Co-processador (criptografia).

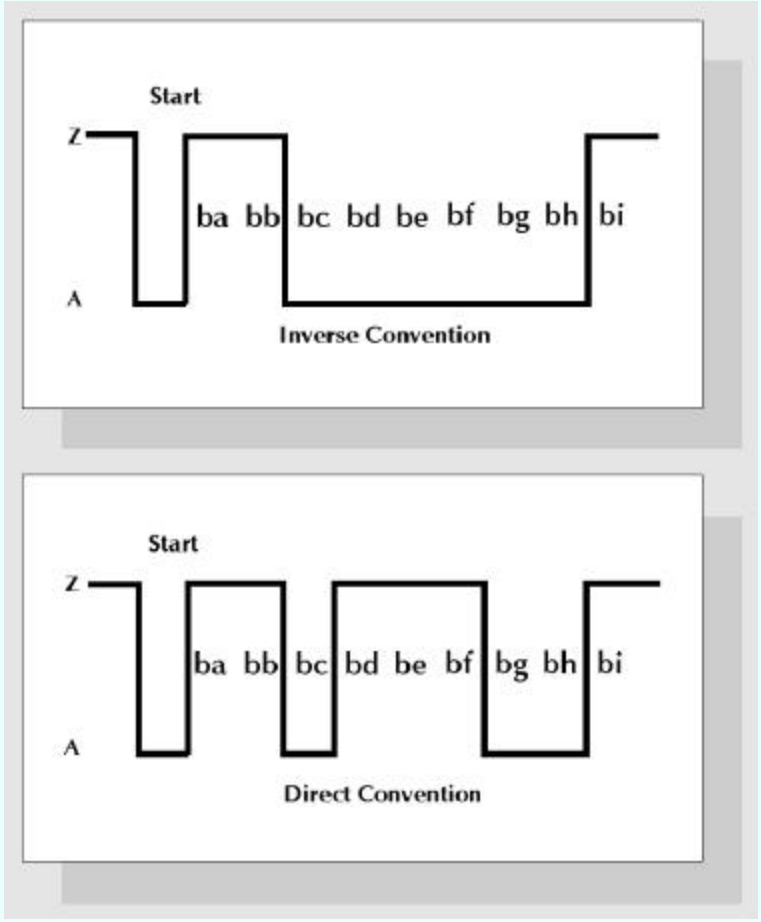


Interoperabilidade do cartão

- Depende de vários factores:
  - Parâmetros eléctricos (alimentação, velocidade de relógio, voltagem de programação, etc.)
  - Protocolo série de transmissão de caracteres (nivel físico).
  - Mecanismo de resposta ao Reset (Answer To Reset).
  - Protocolo(s) de transmissão de dados (nivel lógico): T0/T1.
  - Possibilidade de seleccionar o protocolo de transmissão de dados: Protocol Type Selection.

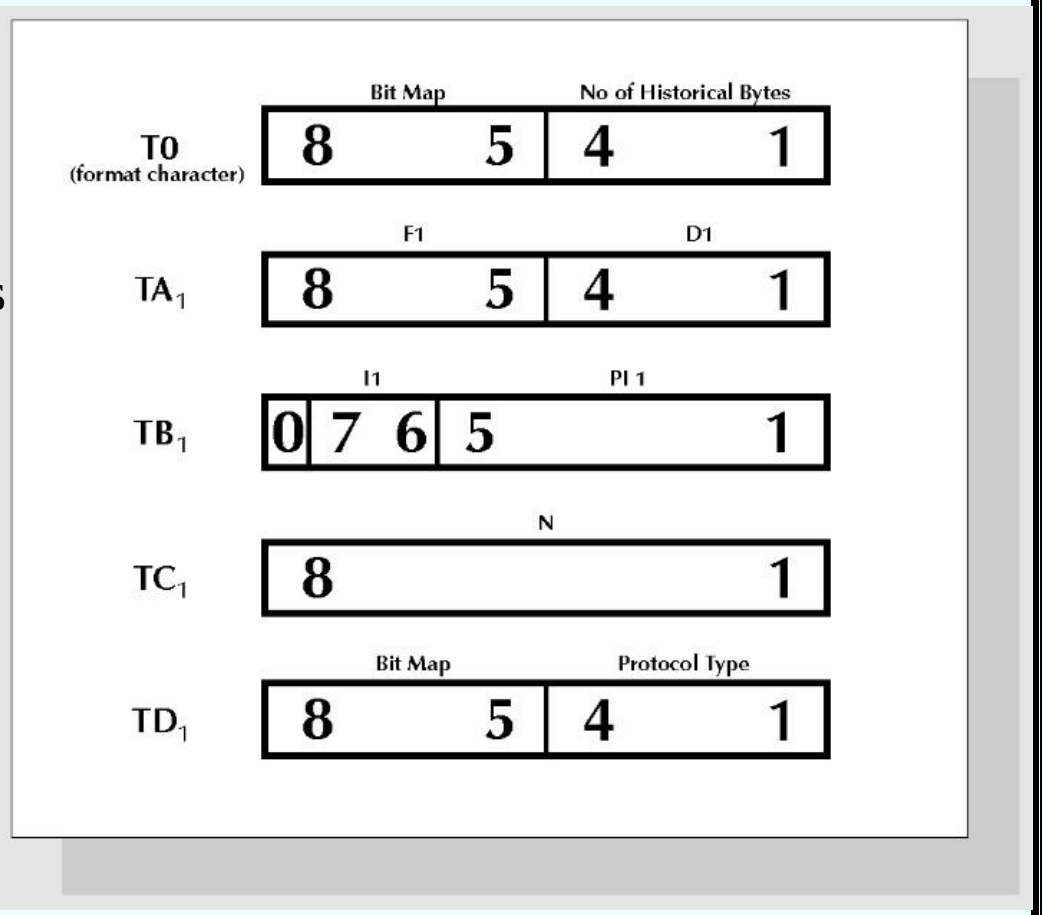
Answer To Reset (ATR)





Answer To Reset (ATR)

- O byte T0 determina a presença dos bytes de interface iniciais, bem como o número de bytes históricos.
- O número de bytes históricos pode ir de 0 a 15.
- O Bit Map indica, bit a bit, a presença dos bytes TA1, TB1, TC1 e TD1.
- O byte TD1 indica o protocolo T de nivel lógico, bem como a presença dos bytes de interface seguintes.

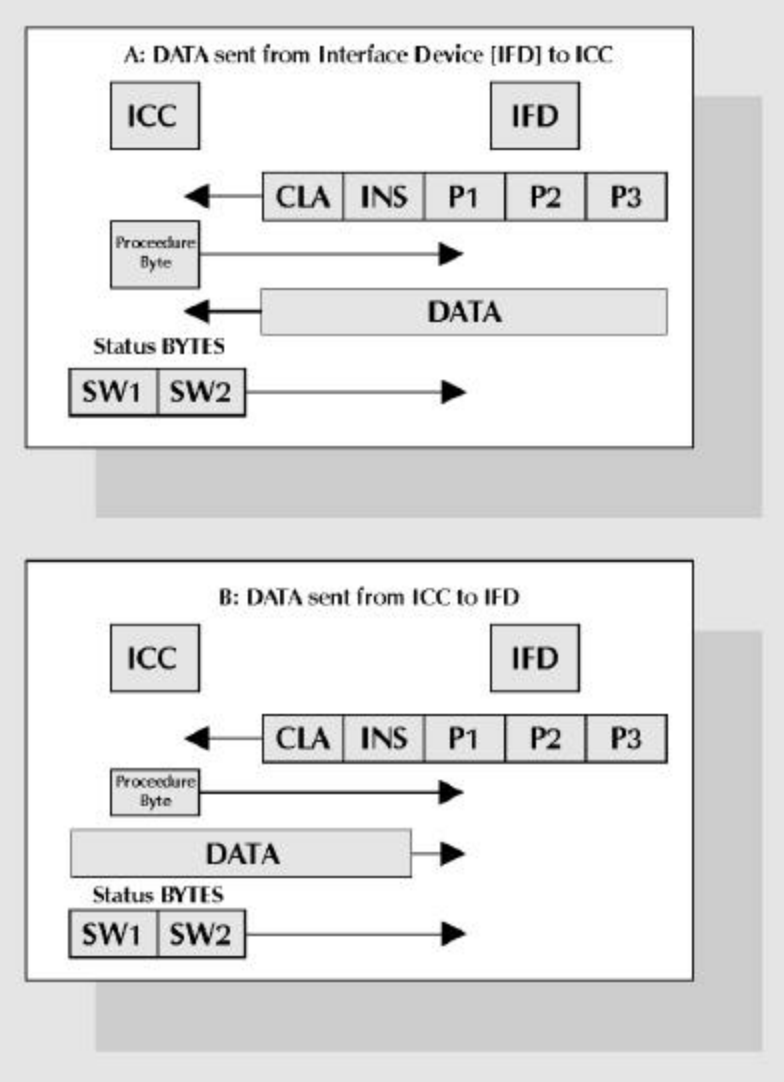


Protocolo T=0

- Transferência half-duplex assíncrona de caracteres.
- Neste protocolo o cartão é passivo. Apenas responde a comandos enviados pelo leitor.
- Para cada comando os dados só podem fluir numa direcção: ou estão incluidos no comando ou na resposta.
- No caso de uma transferência bi-direccional utiliza-se um comando "get response" para obter a resposta do cartão.
- O protocolo não inclui mecanismos para controlar a direcção de transferência. Assume-se que ela é conhecida, para cada comando, por ambas as partes.

Protocolo T=0

- O cabeçalho contém 5 bytes que especificam o comando a ser executado: CLA, INS, P1, P2.
- O cartão responde com um byte de controlo que indica a validade, ou não, do comando.
- Depois são transferidos os dados.
- Finalmente, o cartão envia dois bytes de status que indicam o resultado da operação.



Protocolo T=1

- Transferência half-duplex assíncrona de blocos de caracteres.
- Além das funcionalidades oferecidas pelo protocolo T=0, oferece:
  - Flow control
  - Segmentação
  - Recuperação de erros (retransmissão de blocos)
- Desaparece a limitação de funcionamento Master/Slave.

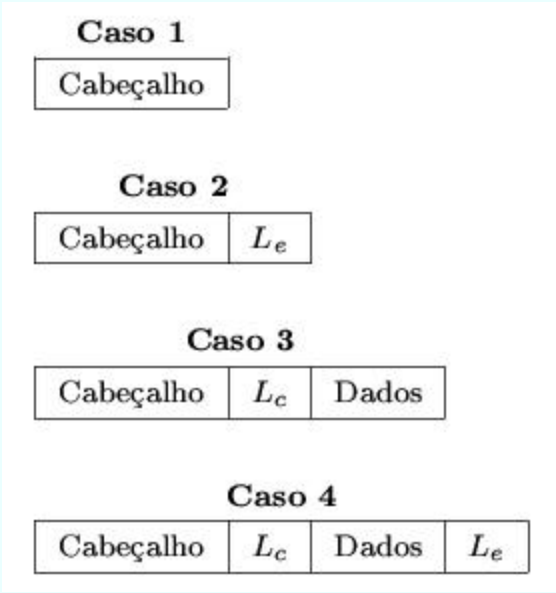
APIs e APDUs

- O ISO7816 define uma abstracção da comunicação leitor/cartão que deve ser utilizada para construir APIs independentes do hardware.
- A comunica com a API através de Aplication Protocol Data Units (APDUs).
- A API traduz entre as APDUs e os comandos e respostas trocados entre o leitor e o cartão, de acordo com o protocolo de nível lógico que estiver a ser utilizado.
- Este mapeamento está definido no standard ISO7816.
- Para a aplicação, a utilização do protocolo T=0 ou T=1 é transparente.

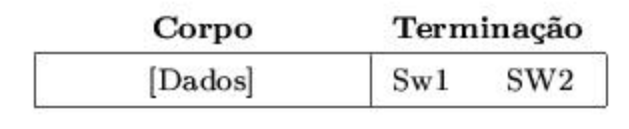
APIs e APDUs

Caso	Comando	Resposta
1	Sem dados	Sem dados
2	Sem dados	Com dados
3	Com dados	Sem dados
4	Com dados	Com dados

Comando

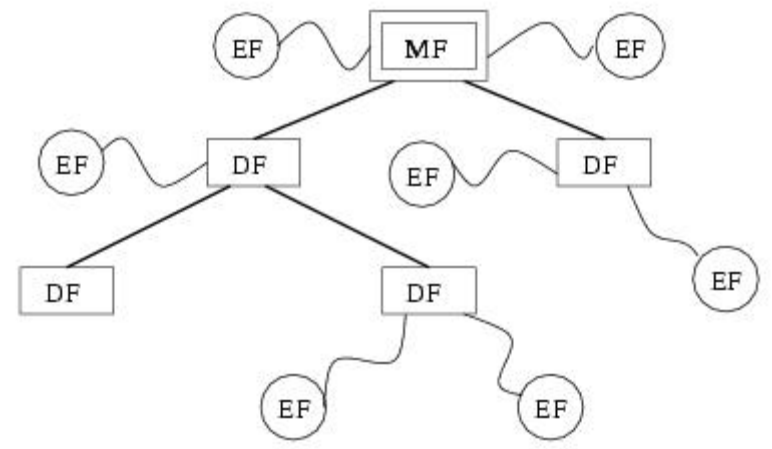


Resposta



Sistema de Ficheiros

- Alguns cartões possuem um sistema de ficheiros incorporado.
- O ISO7816 define a estrutura desse sistema e os comandos necessários para o manipular.
- Três tipos de ficheiros:
  - Master File
  - Dedicated Files
  - Elementary Files
    - Internos
    - Trabalho



Sistema de ficheiros

- Os Efs podem ter quatro tipos de estrutura:
  - Transparente – Sequência não estruturada de bytes
  - Registos individualmente referenciáveis organizados
    - De forma linear com tamanho fixo
    - De forma linear com tamanho variável
    - De forma cíclica com tamanho fixo