Teil 7

Chipkarten
mit synchroner †bertragung

Teil 3: Anwendung von Inter-industry Commands

Entwurf

20.07.95

DIN NI-17.4

Inhalt

1. Zweck

Ziel dieser Spezifikation ist, die Verwendung von ISO/IEC 7816-4 Inter-industry Commands fŸr Chipkarten mit synchroner †bertragung zu beschreiben und ihre Umsetzung in Chip-spezifische Aktionen zu spezifizieren. Diese Spezifikation gilt nur fŸr Chipkarten, deren Datenbereiche nach der DIN-Spezifikation «Chipkarten mit synchroner †bertragung, Teil 1: ATR und Datenbereiche« codiert sind. Die Verwendung von ISO/IEC 7816-4 Inter-industry Commands an einem CardTerminal Application Programming Interface setzt neben der Einhaltung der Struktur des ATR und der Datenbereiche auch die FŠhigkeit des Kartenterminals zur Umsetzung von Inter-industry Commands in Chip-spezifische Aktionen voraus.

2. Normative Verweisungen

DIN NI-17: 1995
Chipkarten mit synchroner †bertragung
Teil 1: ATR und Datenbereiche

DIN NI-17.4: 1995
Chipkarten mit synchroner †bertragung
Teil 2: †bertragungsprotokolle

ISO/IEC 7816-3: 1989
Identification cards - Integrated circuit(s) cards with contacts
Part 3 - Electronic Signals and transmission protocols
AM 1: Clause 9: Protocol T=1, asynchronous half duplex block transmission protocol
AM 2: Protocol type selection
(Verwendung findet der WD vom Oktober 1994, in dem AM 1 und AM 2 integriert sind)

ISO/IEC 7816-4: 1995
Identification cards - Integrated circuit(s) cards with contacts
Part 4 - Inter-industry commands for interchange

3. AbkŸrzungen

AID = Application Identifier
ATR = Answer-to-Reset
BCD = Binary Coded Digits
CT = CardTerminal
FID = File Identifier
ICC = Integrated Circuit(s) Card
PIN = Personal Identification Number
TLV = Tag, Length, Value
VD = Verification Data

4. Das Umsetzungsprinzip

Um die Ansteuerung von Chipkarten mit synchroner †bertragung einerseits so einfach wie mšglich und andererseits weitgehend kompatibel mit der Ansteuerung von Prozessorchipkarten zu machen, werden bestimmte Inter-industry Commands im Kartenterminal (bzw. in der zum Kartenterminal gehšrenden Software) auf Interaktionen mit der entsprechenden synchronen Chipkarte abgebildet. FŸr alle Chipkarten sind folgende Kommandos zu unterstŸtzen:

- SELECT FILE
- READ BINARY und
- UPDATE BINARY.

FŸr Chipkarten mit Verification Data sind zusŠtzlich die Kommandos

- VERIFY und
- CHANGE VERIFICATION DATA

zu unterstŸtzen. Das Setzen von Protection Flags (falls der Chip mit einem Protection Memory ausgestattet ist) wird in der Regel bei der Personalisierung vorgenommen und ist nicht im Leistungsumfang der nachfolgend beschriebenen Kommandos enthalten.

5. Inter-industry Commands fŸr Grundfunktionen

5.1 SELECT FILE

5.1.1 Funktion

a) Selektieren einer Anwendung

Zum Selektieren einer Anwendung wird das ISO/IEC 7816-4 SELECT FILE-Kommando verwendet. Hierbei wird der Application Identifier (AID) im Datenfeld Ÿbergeben. Das Kartentermi-

nal liest den DIR-Datenbereich und prŸft, ob die AID dort zu finden ist (Strukturen des DIR-Datenbereichs entsprechend Teil 1 «ATR und Datenbereiche«). Wenn ja, wird die Anwendung und damit der Anwendungsdatenbereich selektiert und als Return-Code «9000« zurŸckgegeben. Der Anwendungsdatenbereich beginnt bei Mono-Application Cards direkt hinter dem DIR-Datenbereich, in Multi-Application Cards wird der Anfang des Anwendungsdatenbereichs im Path-Element des entsprechenden Application Templates angezeigt.

b) Selektieren von Datenbereichen

Ein Datenbereich in einer synchronen Karte ist wie ein File in einer Mikroprozessorchipkarte Ÿber File-Identifier (FIDs) bzw. Ÿber den File-Name mit dem SELECT FILE-Kommando selektierbar:

- der ATR-Datenbereich hat wie der ATR-File als FID «2F01« und beginnt nach dem ATR auf Byte-Adresse «04«
- der DIR-Datenbereich hat wie der DIR-File als FID «2F00« und beginnt auf der Byte-Adresse, die in Byte H4 (siehe Teil 1: «ATR und Datenbereiche«) angegeben ist
- der Anwendungs-Datenbereich hat als Kennung den Application Identifier und wird daher mit dem SELECT FILE-Kommando mit Angabe der AID selektiert, wie in a) beschrieben. Hierbei wird der Pointer auf das erste Byte des Anwendungs-Datenbereichs eingestellt.

Um auch den gesamten Datenspeicher bei Bedarf selektieren zu kšnnen, wird er als eine Sequenz von Datenbereichen bzw. Files gesehen, die im Master-File enthalten bzw. diesem untergeordnet sind; daher wird als FID fŸr den gesamten Datenspeicher die MF-FID «3F00« verwendet (Adresse des ersten Bytes: «00«).

5.1.3 Kommando-Struktur

CLA	«00«
INS	«A4« (= SELECT FILE)
P1	Selection control «00« = FID in data field «04« = AID in data field
P2	«00«
Lc field	Length of subsequent data field
Data field	AID or FID («3F00«= MF (= total data memory), «2F00«= DIR data section, «2F01« = ATR data section)
Le field	Empty

Tab. 1: SELECT FILE-Kommando

Anmerkung: Das Kommando ist hier nur mit den benštigten Parameter-Codierungen dargestellt.

5.1.4 Antwort-Struktur

Data	Empty
SW1-SW2	Status bytes

Tab. 2: SELECT FILE-Response

5.1.5 Status Bytes

«9000« = Command successful
«6A82« = File not found

5.2 READ BINARY

5.2.1 Funktion

Mit dem READ BINARY-Kommando kšnnen Daten aus dem zuvor selektierten Datenbereich gelesen werden. Das erste Byte des Datenbereichs hat die logische Adresse «0000«. Die LŠnge des Datenbereichs ergibt sich aus der LŠnge des ersten DOs (siehe Teil 1: «ATR und Datenbereiche«).

5.2.2 Anwendungsbedingungen

Der zu lesende Datenbereich mu§ zuvor slektiert worden sein.

5.2.3 Kommando-Struktur

CLA	«00«
INS	«B0« (= READ BINARY)
P1, P2	Offset («0000« = Logical start address of the file)
Lc field	Empty
Data field	Empty
Le field	Length of data to be read or «00« (= read available data)

Tab. 3: READ BINARY-Kommando

Anmerkung: Das Kommando ist hier nur mit den benštigten Parameter-Codierungen dargestellt.

5.2.4 Antwort-Struktur

Data	Data to be read
SW1-SW2	Status bytes

Tab. 4: READ BINARY-Response

5.2.5 Status Bytes

«9000« = Command successful
«6281« = Data corrupted
«6282« = Warning, end of file reached before reading Le bytes
«6501« = Memory failure

5.3 UPDATE BINARY

5.3.1 Funktion

Mit dem UPDATE BINARY-Kommando kšnnen Daten aus dem zuvor selektierten Datenbereich geŠndert werden. Das erste Byte des Datenbereichs hat die logische Adresse «0000«. Die LŠnge des Datenbereichs ergibt sich aus der LŠnge des ersten DOs (siehe Teil 1: «ATR und Datenbereiche«).

5.3.2 Anwendungsbedingungen

Der zu beschreibende Datenbereich mu§ zuvor selektiert worden sein. Bei Chipkarten, die eine €nderung des Datenspeichers (bzw. Teile davon) nur nach vorheriger erfolgreicher PrŠsentation des Sicherheitscodes erlauben, ist zuvor die entsprechende Authentisierung (siehe VERIFY command) durchzufŸhren. Bei Offset «0000« ist der komplette Inhalt des betreffenden Datenbereichs zurŸckzuschreiben und als LŠnge des Datenbereichs die entsprechende neue LŠnge einzutragen.

5.3.3 Kommando-Struktur

CLA	«00«
INS	«D6« (= UPDATE BINARY)
P1, P2	Offset («0000« = Logical start address of the file)
Lc field	Length of subsequent data field
Data field	Data to be written
Le field	Empty

Tab. 5: UPDATE BINARY-Kommando

Anmerkung: Das Kommando ist hier nur mit den benštigten Parameter-Codierungen dargestellt.

5.3.4 Antwort-Struktur

Data	Empty
SW1-SW2	Status bytes

Tab. 6: UPDATE BINARY-Response

5.3.5 Status Bytes

«9000« = Command successful
«6200« = Error

6. Inter-industry Commands fŸr Sicherheitsfunktionen

6.1 VERIFY

6.1.1 Funktion

Das VERIFY-Kommando veranla§t den Vergleich der Verification Data mit den gespeicherten Reference Data. Der Ablauf ist in Anhang A dargestellt. Verification Data sind BCD-codiert, wenn es sich um eine PIN handelt, ansonsten wird «hexadecimal coding« verwendet.

6.1.2 Anwendungsbedingungen

Das Kommando ist nur bei Chipkarten mit entsprechender Sicherheitsfunktion zulŠssig.

6.1.3 Kommando-Struktur

CLA	«00«
INS	«20« (= VERIFY)
P1	«00«
P2	«00«
Lc field	Length of subsequent data field
Data field	Verification data (Byte1, Byte 2,

	Byte 3) Note: a PIN is BCD-coded possibly padded with one or more «F«
Le field	Empty

Tab. 7: VERIFY-Kommando

Anmerkung: Das Kommando ist hier nur mit den benštigten Parameter-Codierungen dargestellt.

6.1.4 Antwort-Struktur

Data	Empty
SW1-SW2	Status bytes

Tab. 8: VERIFY-Response

6.1.5 Status Bytes

«9000« = Command successful
«63Cx«= Verification unsuccessfull, x = number of possible retries
«6983« = Verification method blocked

6.2 CHANGE VERIFICATION DATA

6.2.1 Funktion

Mit dem CHANGE VERIFICATION DATA-Kommando kšnnen Verification Data geŠndert werden.
Verification Data sind BCD-codiert, wenn es sich um eine PIN handelt, ansonsten wird «hexadecimal coding« verwendet.

6.2.2 Anwendungsbedingungen

Das Kommando ist nur bei Chipkarten mit entsprechender Sicherheitsfunktion zulŠssig.

6.2.3 Kommando-Struktur

CLA	«00«
INS	«24« (= CHANGE VERIFICATION DATA)
P1, P2	«0000«
Lc field	Length of subsequent data
Data field	Old verification data (Byte1, Byte 2, Byte 3), new reference data (Byte1, Byte 2, Byte 3) Note: PINs are BCD-coded possibly padded with one or more «F«
Le field	Empty

Tab. 9: CHANGE VERIFICATION DATA-Kommando

Anmerkung: Das Kommando ist hier nur mit den benštigten Parameter-Codierungen dargestellt.

6.2.4 Antwort-Struktur

Data	Empty
SW1-SW2	Status bytes

Tab. 10: CHANGE VERIFICATION DATA-Response

6.2.5 Status Bytes

«9000« = Command successful
«63Cx«= Verification unsuccessful, x = number of possible retries
«6983« = Verification method blocked

Anhang A (normativ)

Abbildung der Kommandos VERIFY und CHANGE VERIFICATION DATA

Die folgenden Abbildung zeigen die Abbildung des ISO/IEC 7816-4 VERIFY-Kommandos und des CHANGE VERIFICATION DATA-Kommandos auf die Kommandofolge von Chipkarten mit 2WB-Protokoll (S = 10) und entsprechender Sicherheitsfunktion.

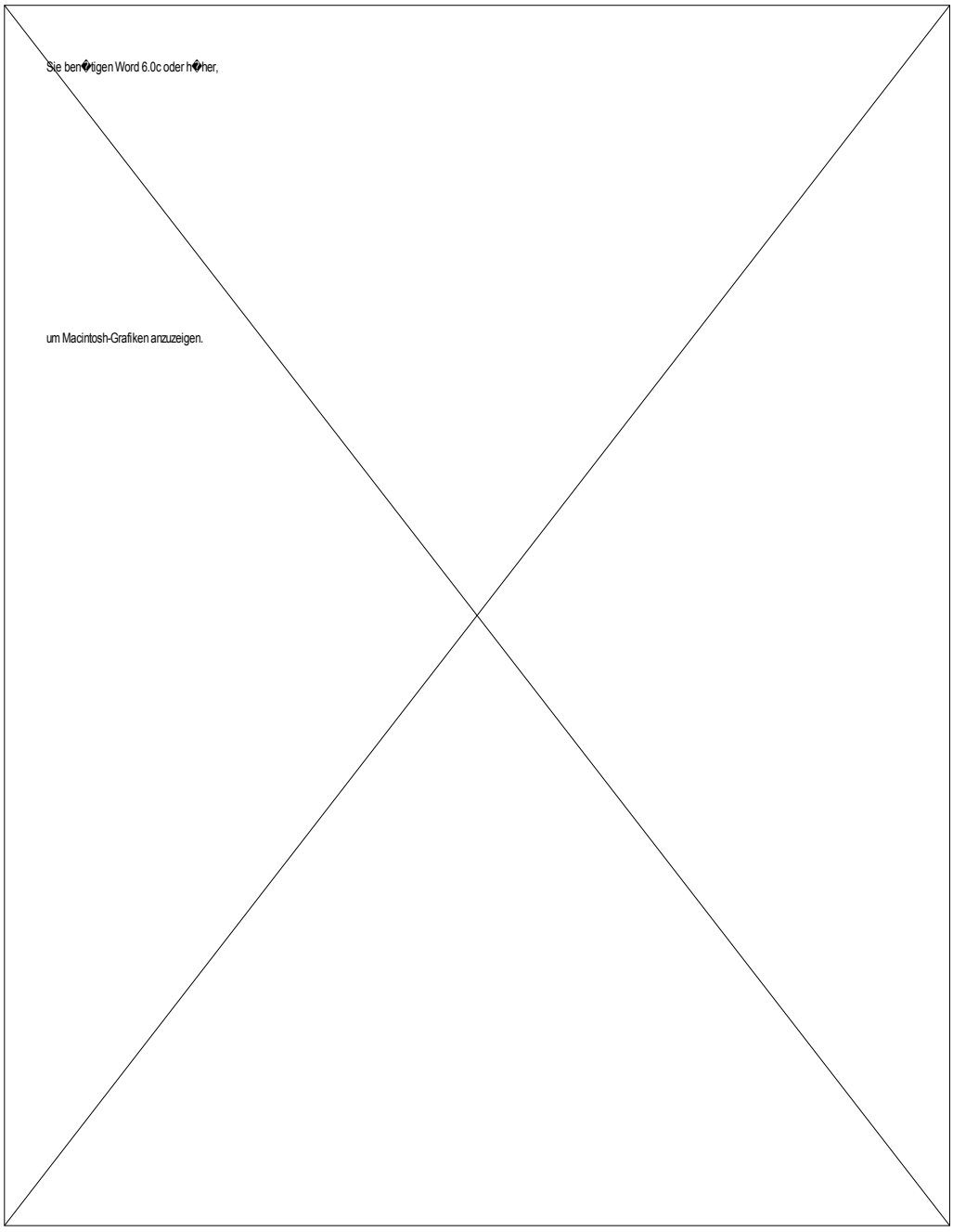

Abb. 1: Flu§diagramm zum Ablauf des VERIFY-Kommandos

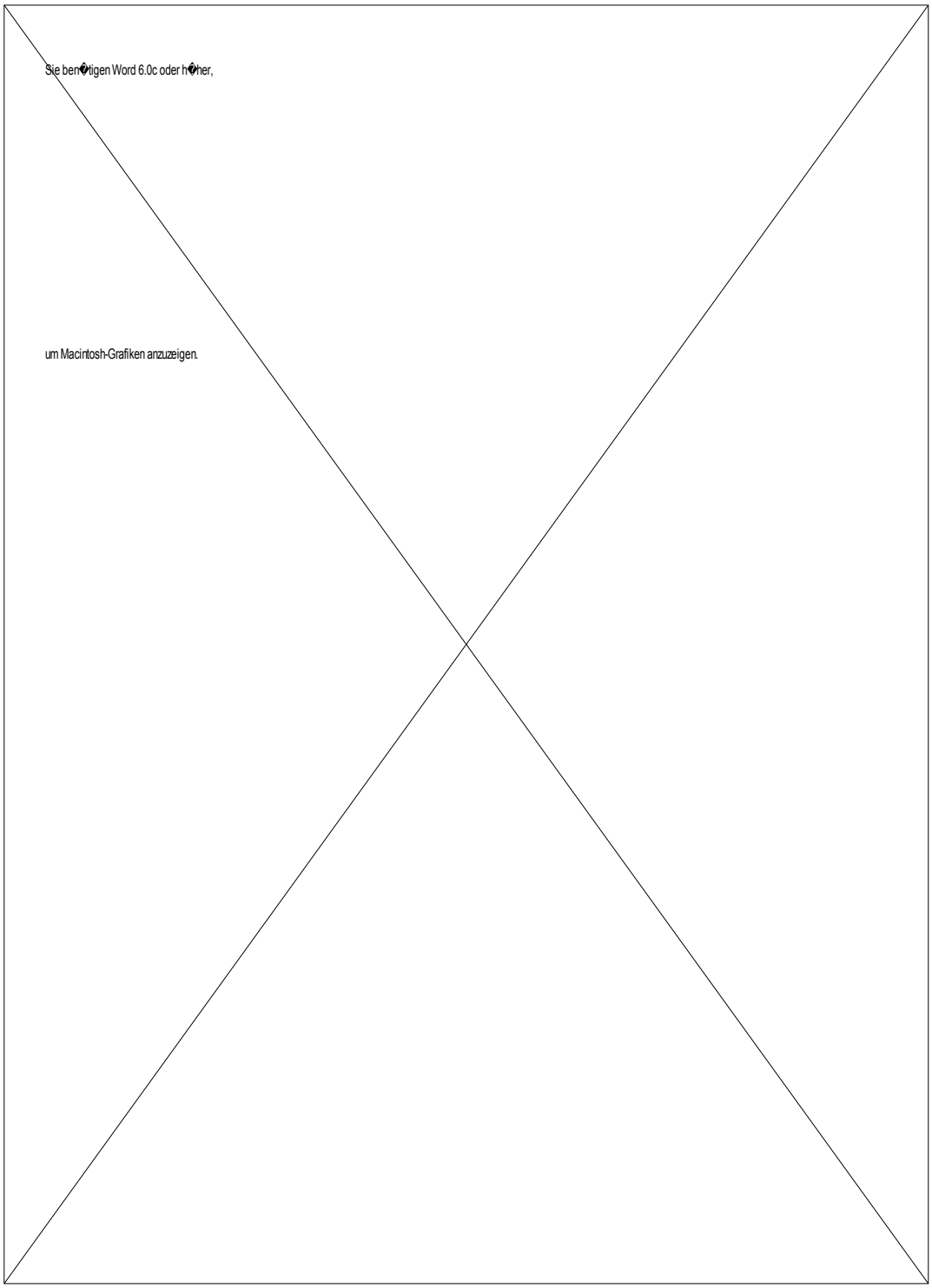

Abb. 2: Flu§diagramm zum Ablauf des CHANGE VERIFICATION DATA-Kommandos