

APPLICATION NOTE

**TDA8006 mask n°12
DOUBLE SMART CARD READER
USING TDA8006 AND TDA8004 INTERFACES**

AN/97074

ABSTRACT

This document describes the software specifications that have been developed for the double smart card reader CAKE 611A and the demoboard CAKE-609A. They use the card interfaces devices TDA8006 and TDA8004.

The software, embedded in the TDA8006, is able to handle the communication between a host system (called also system controller further in this document) and two asynchronous smart cards according to T=0 or T=1 protocols. The TDA8006 functions are controlled by the host through a serial interface.

APPLICATION NOTE

TDA8006 mask n°12 DOUBLE SMART CARD READER USING TDA8006 AND TDA8004 INTERFACES

**Author(s):
Jean-Luc LUONG**

**Application Laboratory - Paris
France**

Keywords

TDA8006

TDA8004

ISO7816-3, protocols T=0, T=1

EMV standards

GSM 11-11 standards

Double smart card reader

Serial interface at 19200 bauds

Date : October 1997

CONTENTS

| | |
|---|-----------|
| I. REFERENCE HARDWARE | 6 |
| I.1. EXPLANATION OF THE PINS | 6 |
| I.2. MAIN CHARACTERISTICS OF THE DOUBLE READER INTERFACE CONTROLLER | 6 |
| II. FUNCTIONALITIES..... | 8 |
| II.1. POWER MANAGEMENT | 8 |
| II.2. CARDS ACTIVATION AND COMMUNICATIONS | 9 |
| II.2.1. Activation..... | 9 |
| II.2.2. Deactivation | 9 |
| II.2.3. Card communication commands | 10 |
| II.3. CARD MANAGEMENT : ADDITIONAL FEATURES | 11 |
| II.3.1. Card presence information | 11 |
| II.3.2. Card clock frequency choice | 11 |
| II.3.3. PTS request..... | 11 |
| II.4. SYSTEM COMMUNICATION AND ADDITIONAL GENERAL COMMANDS..... | 12 |
| III. SERIAL INTERFACE..... | 13 |
| III.1. PHYSICAL LINES | 13 |
| III.2. PHYSICAL LAYER..... | 13 |
| III.2.1. Mode of operation..... | 13 |
| III.2.2. Baud rate..... | 13 |
| III.3. PROTOCOL..... | 14 |
| III.4. DIALOG STRUCTURE | 14 |
| III.4.1. System command..... | 14 |
| III.4.2. Messages initiated by the interface controller..... | 15 |
| IV. COMMANDS DESCRIPTION..... | 16 |
| IV.1. APPLICATION AND INSTRUCTION OPERATING CODES | 16 |
| IV.2. GENERAL COMMANDS (AOPC = 0)..... | 17 |
| IV.2.1. Receipt (Control = 00h)..... | 17 |
| IV.2.2. Error detection (Control = 01h)..... | 17 |
| IV.2.3. Set in power down (Control = 02h)..... | 17 |
| IV.2.4. Alive (Control = 04h)..... | 18 |
| IV.2.5. Software restart (Control = 05h)..... | 18 |
| IV.3. CARD COMMANDS (AOPC = 1 OR 2)..... | 19 |
| IV.3.1. Reset card (Control = 10h or 20h)..... | 19 |
| IV.3.2. Disconnect card (Opcode = 11h or 21h)..... | 20 |
| IV.3.3. Send data to card (Opcode = 12h or 22h)..... | 20 |
| IV.3.4. Request data from card (Control = 13h or 23h)..... | 20 |
| IV.3.5. Answer from card (Control = 17h or 27h)..... | 21 |
| IV.3.6. Card communication error (Control = 1Bh or 2Bh)..... | 21 |
| IV.3.7. PTS request (Control = 1Ch or 2Ch)..... | 23 |
| IV.3.8. Set card 1 clock (Control = 14h)..... | 23 |
| IV.3.9. Card presence active (Control = 19h or 29h) | 23 |
| V. DEMOBOARD DESCRIPTION AND DIRECTIONS FOR USE..... | 24 |

- V.1. HARDWARE DESCRIPTION 24
 - V.1.1. Schematics..... 24
 - V.1.2. Power supply..... 24
 - V.1.3. Interface with the PC..... 24
 - V.1.4. Clock frequencies 24
 - V.1.5. Smart card features..... 24
- V.2. DIRECTIONS FOR USE 25
 - V.2.1. Presentation of the PC driver « double64 » 25
 - V.2.2. Installation of « double64 » 25
 - V.2.3. Starting «double64»..... 26
 - V.2.4. Description of the interface windows of « double64 »..... 27
- VI. ANNEX : CAKE 611-A 29**

I. Reference hardware

The schematic given in the next page shows the masked TDA8006 with the hereby described software in a typical application with the TDA8004.

I.1. Explanation of the pins

| PINS | DESCRIPTION |
|-------|--|
| RESET | Reset pin of the tda8006 |
| RX | Reception line for serial communication |
| TX | Transmission line for serial communication |
| VDD | 5V power supply |
| GND | Ground |

I.2. Main characteristics of the double reader interface controller

The main characteristics of the card reader 1 - driven by the TDA 8006 - and of the card reader 2 - controlled by the tda 8004 - are listed below :

| CARD READER 1 | CARD READER 2 |
|---|--|
| Support asynchronous protocols T=0 and T=1 in accordance with ISO 7816-3, EMV and GSM 11-11 requirements. | |
| Control and communication through a standard serial interface with a baudrate of 19200 bauds. | |
| Automatic hardware protections in the event of card take-off, supply voltage drop, shortcircuit or overheating. | |
| Support 3V and 5V smart cards | |
| Hardware and software driven ISO UART supporting 9600, 19200, ...,115000 baud rates | Fully software driven ISO UART supporting 9600 and 19200 baud rates. |
| Low power consumption mode | -- |
| 3 selectable card clock frequencies:1.84 Mhz, 3.68 Mhz and 7.37 Mhz | Fixed card clock frequency : 3.68 Mhz |
| Card clock stopped in high state, in low state or running at 1.25 Mhz frequency for card power-down mode. | -- |

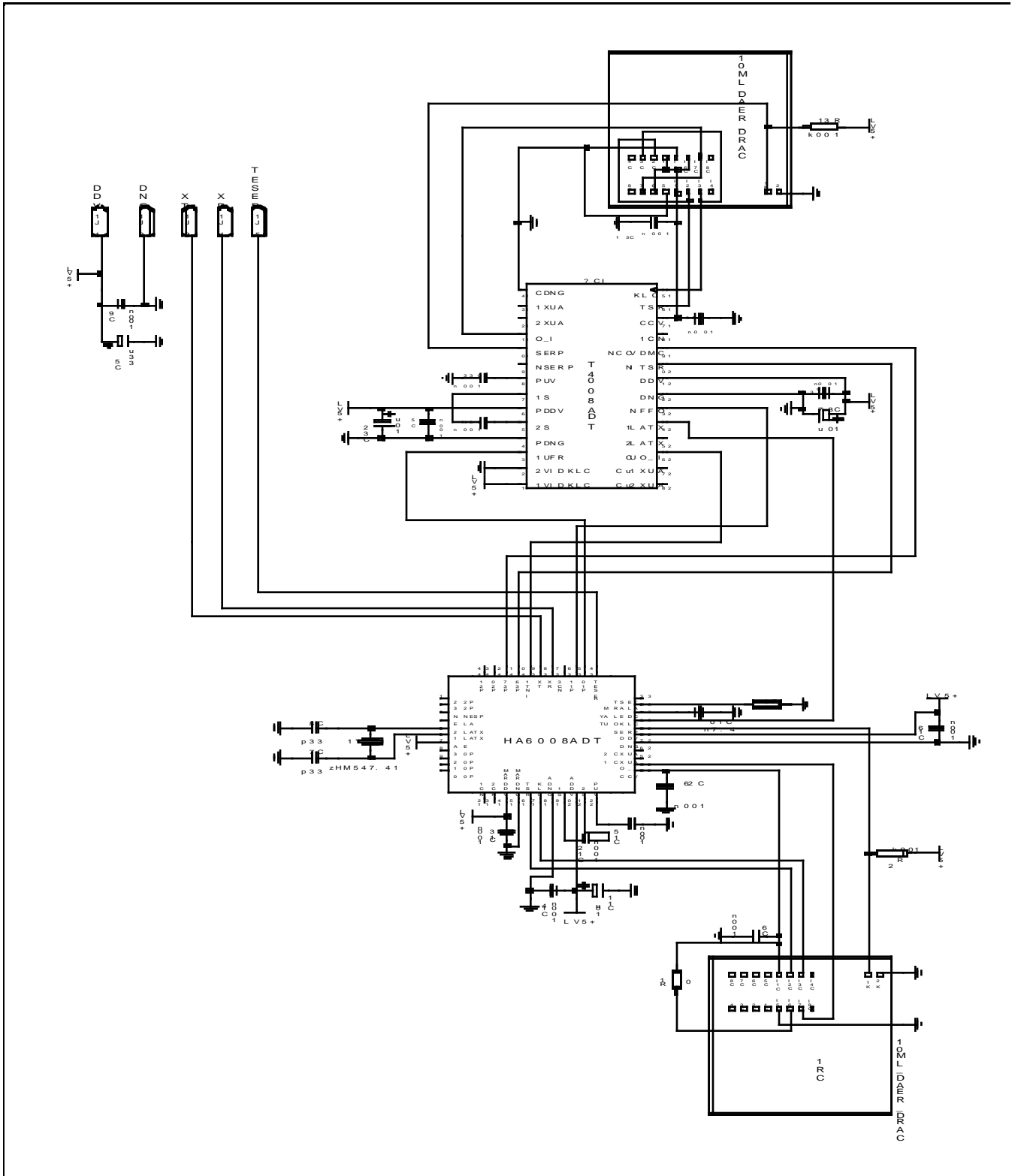


Figure 1 : Double smart card reader CAKE-611A using TDA8006 and TDA8004 interface devices

II. Functionalities

This application is able to handle a communication between a system controller and two smart cards.

It is particularly suited to applications where one card (card 1) is inserted as long as the system is on, and the other card (card 2) is inserted on demand.

The host system communicates with the double reader via a serial Rx/Tx interface.

One card is attached to the TDA8006 (card 1). It is handled by the software embedded in the TDA8006. The other card (card 2) is attached to the TDA8004. The management of the TDA8004 and the card 2 is handled by the TDA8006 through the following pins :

| TDA8004 PINS | TDA8006 PINS |
|--------------|--------------|
| CMDVCC | P3.7 |
| RSTIN2 | P3.6 |
| IO_uC2 | P3.3 (INT1) |
| OFF2 | P1.1 |
| RFU2 | P1.0 |
| XTAL2 | CLKOUT |

Rem : the names "card 1" or "slot 1" will be used for the card or card connector attached to the TDA8006, and the names "card 2" or "slot 2" will be used for the card or card connector attached to the TDA8004.

II.1. Power management

The power management of the card 1 and 2 readers are independant each other. The card 1 reader, driven by the interface controller TDA8006, has two operating modes :

- *Normal operating mode*: The card 1 is active i.e. its clock is set to a selected frequency (1.84 MHz, 3.68 MHz or 7.37 MHz)
- *Power down mode*: The TDA8006 is put in idle mode. The internal clock is gated off to the CPU but not to the Interrupt, Timer and Serial Port. The port pins hold the logical states they had.

The card 2 reader, driven by the TDA8004, has only an operating mode which is *the normal operating mode*. So the card 2 is either inactive or active with a fixed clock frequency of 3.68 MHz.

II.2. Cards activation and communications

For the two cards, the following functions are defined :

II.2.1. Activation

This function is initiated by the commands "Reset Card 1" or "Reset Card 2" that have one parameter indicating some activation options (see command description).

The Answer To Reset (abbreviated ATR) is expected from the card according to the ISO standard activation sequence. The parameters of ATR are decoded and analysed. If the ATR is correctly received, it is completely sent back to the system through an "Answer From Card" message.

If its decoding shows an erroneous parameter (depending on the activation options or on the interfaces restrictions), a "card communication error" message is sent to the system and no deactivation is performed. Moreover, if the work waiting time is elapsed and no ATR is received, the card is considered as « mute » and deactivated.

The activation is always initiated at a card clock frequency of 3.68 MHz (the card clock frequency is get by dividing by 4 the 14,74 MHz crystal frequency of the TDA8006) for both cards, with no guard time and with the default work waiting time of 10. The timing parameters i.e. extra guard time , work waiting time (applicable for T = 0 protocol), character waiting time and block waiting time (applicables for T = 1) are automatically extracted from the ATR and memorised. They will be used for all further communications with the card.

II.2.2. Deactivation

This function is initiated either :

- by the commands "Disconnect Card 1" or "Disconnect Card 2".
- when the events like card take-off, supply voltage drop, shortcircuit or overheating occur. In this case, it is done in an automatical way by the hardware of the TDA8004 or of the TDA8006.
- *if the card is considered as absent or muted during a card activation.*

This function disconnects the card contacts according to the ISO standard activation sequence.

II.2.3. Card communication commands

Except the answer to reset and the PTS procedure, two kinds of card communication commands are defined :

- Card commands sending data to the card (or "write" commands)
- Card commands requesting data from the card (or "read" commands)

- In the first case, the card command structure is the following :

CLA INS P1 P2 L data ...

The number of data is given by the parameter L. *A maximum number of 255 data can be sent to the card.*

The card answers two parameters SW1 and SW2.

The system command used to send such a command to the card is "Send data to Card", with the complete card command (header + data bytes) as parameters.

- In the second case, the card command structure is the following :

CLA INS P1 P2 L

The L data are expected from the card. *If L is equal to 0, a maximum of 256 data can be requested from the card.*

The card may answer either the L data plus the parameters SW1 and SW2, or only SW1,SW2 if there was an error in the command.

The system command used to send such a command to the card is "Request data from card", with the header as parameters.

II.3. Card management : additional features

II.3.1. Card presence information

The system controller is automatically informed on the cards presence or absence :

- at power on,
- after a restart software command,
- every time a card is inserted or extracted.

There is no command foreseen for the system to ask for card presence. If it is needed, the system can send a card communication command (reset card, send data or request data). If it gets only a card receipt message, it means that the card is not present.

II.3.2. Card clock frequency choice

The setting of card clock frequency is made through the command "Set card 1 clock". The chosen frequency is set at the same time, in the "Clock status" parameter.

After the activation, six clock card frequencies are available for the card 1:

- | | | |
|--|---|--|
| <ul style="list-style-type: none"> - Fclk = Fxtal / 2 - Fclk = Fxtal / 4 - Fclk = Fxtal / 8 | } | these frequencies can be selected in active mode |
| <ul style="list-style-type: none"> - Fclk = Fint/ 2 (≈ 1.2 MHz) - Fclk = stop low - Fclk = stop high | } | these frequencies are used to put the card in sleep mode |

For the card 2, only 3.68 MHz (Fxtal/4) frequency is available. So, there is no clock programming command for card 2.

The default card frequencies are Fxtal / 4 in active mode.

The cards activation and the answers to reset (even for warm resets) are always handled with Fclk = Fxtal / 4. The settings are reset (to default values) after card deactivation.

II.3.3. PTS request

If the card is able to handle different protocols (T=0 and T=1) and if the card supports to work at a higher baud rate on I/O line (Fi/Di values are different from default values) a PTS request can be initiated just after an ATR.

A PTS request to the card is performed using the « Pts request » command. Only values 0 (for protocol T=0) or 1 (for protocol T=1) are allowed for the parameter of this command.

II.4. System communication and additional general commands

- The communication with the system controller is handled via a standard serial link (Rx/Tx lines) at a fixed baud rate. The serial link and the chosen protocol are described in the paragraph "Serial interface".

The software version and the TDA8006 mask number informations are available through the command "Alive".

- A "restart software" command is able to restart the software initialisations for the serial communications and the interrupts, without resetting the cards.
In the case the TDA8006 is blocked into a system command processing, the restart software command may be able to release it : At reception of the command control byte a flag is set that allows to go through some communication waiting loops. If it occurs, the TDA8006 may send on the TX line some bytes without sense, until it sends the receipt frame and the cards presence messages.

III. Serial Interface

The serial interface is the link between the system controller and the interface device TDA8006.

III.1. Physical lines

The realisation of the serial interface is a RS232 full duplex interface from the 80C52 with the two lines RXD and TXD.

RXD is used to receive data from the system controller, TXD is used to send data to the system controller.

III.2. Physical layer

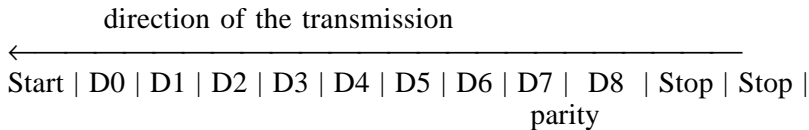
III.2.1. Mode of operation

The mode of operation is the mode 3 of the 8052 serial interface, with the following features :

12 bits transmitted or received :

- start bit
- 8 data bits, with LSB transmitted first (D0...D7)
- parity bit (D8)
- 2 stop bit(s)

The parity bit D8 is used for parity check. *The defined parity is even.*



III.2.2. Baud rate

The baud rate of the TDA8006 serial UART in mode 3 is given by the following formula :

$$baudrate = \frac{2^{SMOD}}{32} \times \frac{Crystal\ Frequency}{12 \times (256 - TH1)}$$

The chosen values for TH1 and SMOD give a baud rate of 19200 bauds for a crystal frequency of 14.745 MHz.

III.3.Protocol

The communication protocol has the following frame structure :

| | | |
|--|--|------------------------------|
| - 1 PATTERN byte (7Eh) | | 5 bytes used for protocol |
| - 1 FRAME_NR byte | | |
| - 2 LENGTH bytes (of the complete frame) | | |
| - 1 CONTROL byte | | |
| - data bytes | | |

| PATTERN | FRAME NF | MSB LENGTH | LSB LENGTH | CONTROL | 1st DATABYTE |
|------------|------------|------------|------------|-------------|--------------|
| 01111110 P | xxxxxxxx P | xxxxxxxx P | xxxxxxxx P | AOPC IOPC P | xxxxxxxx P |

Note: P stands for parity bit

The beginning of each frame is marked by the PATTERN byte, which hexadecimal value is 7E.

The second byte is the frame number byte (FRAME_NR). Each transmitter increments its own frame number byte at every transmission. The received frame number is stored by the interface controller for the receipt (see forward).

The next two bytes are the LENGTH, coded in hexadecimal, of the complete frame including 5 bytes of protocol and data bytes. Its minimal value is 5 (no data byte), and its maximal value is 263 (5 bytes of protocol + 256 bytes of data + 2 status bytes SW1, SW2).

The fifth byte is the CONTROL byte. It consists of two nibbles with the following meaning :

- high nibble : AOPC (Application OpCode)
- low nibble : IOPC (Instruction OpCode)

This protocol is used in both directions, for commands and answers.

III.4.Dialog structure

III.4.1.System command

Most procedures are initiated by a host command. After each command sent by the system controller, the interface controller first runs the corresponding process, and then sends possibly a command answer and -at least - a receipt frame.

A command sent by the system controller is assumed to be finished when the interface controller sends a receipt to the system controller. If the command needs an answer from the interface controller, the receipt is transmitted after this answer: The receipt is the last transmitted frame in a response to a system command.

Remark : A receipt message is a normal frame, with a length of 6 bytes, a control byte equal to 00h and a parameter which repeats the frame_nr of the command.

If the system command reception has failed, due for example to a parity error or an unknown OpCode, an Error Detection message is sent instead of a receipt at the end of the reception of the complete system command. In case of a parity error detection, the system controller will have to repeat the whole frame.

So, the system controller should not send any further command before it has received either a receipt or an error detection message to its previous command.

III.4.2. Messages initiated by the interface controller

The card presences generate a message without any command from the system controller. No answer is expected from the system controller.

IV. Commands description

IV.1. Application and instruction Operating Codes

Three AOPCs are defined for using the interface controller:

- AOPC = 0 : General commands

These commands are used to manage the interface controller. The following table shows the corresponding instructions :

| GENERAL COMMANDS | AOPC IOPC | DIRECTION | PARAMETER BYTES |
|-------------------|-----------|-----------|---------------------|
| Receipt | 0 0 | IFD → SC | Receipt frame |
| Error detection | 0 1 | IFD → SC | Error detection nr |
| Set in power down | 0 2 | SC → IFD | -- |
| Alive | 0 4 | SC → IFD | Mask nr, version nr |
| Software restart | 0 5 | SC → IFD | -- |

- AOPC = 1 : Card 1 commands

These commands are used to perform the complete dialog with the card connected to the TDA8006 interface device. The following instructions are defined :

| CARD 1 COMMANDS | AOPC IOPC | DIRECTION | PARAMETER BYTES |
|----------------------------|-----------|-----------|-----------------------------|
| Reset card 1 | 1 0 | SC → IFD | Activation options byte |
| Disconnect card 1 | 1 1 | SC → IFD | -- |
| Send data to card 1 | 1 2 | SC → IFD | N bytes: CLA...P3 + P3 data |
| Req. data from card 1 | 1 3 | SC → IFD | 5 bytes: CLA...P3 |
| Set card 1 clock | 1 4 | SC → IFD | Clock status byte |
| Answer from card 1 | 1 7 | IFD → SC | See command description |
| Card 1 presence active | 1 9 | IFD → SC | -- |
| Card 1 presence passive | 1 A | IFD → SC | -- |
| Card 1 communication error | 1 B | IFD → SC | Error code byte |
| Pts 1 request | 1 C | SC → IFD | Protocol type (0 or 1) |

- AOPC = 2 : Card 2 commands

These commands are used to perform the complete dialog with the card connected to the TDA8004 interface device :

| CARD 2 COMMANDS | AOPC IOPC | DIRECTION | PARAMETER BYTES |
|----------------------------|-----------|-----------|-----------------------------|
| Reset card 2 | 2 0 | SC → IFD | Activation options byte |
| Disconnect card 2 | 2 1 | SC → IFD | -- |
| Send data to card 2 | 2 2 | SC → IFD | N bytes: CLA...P3 + P3 data |
| Req. data from card 2 | 2 3 | SC → IFD | 5 bytes: CLA...P3 |
| Answer from card 2 | 2 7 | IFD → SC | See command description |
| Card 2 presence active | 2 9 | IFD → SC | -- |
| Card 2 presence passive | 2 A | IFD → SC | -- |
| Card 2 communication error | 2 B | IFD → SC | Error code byte |
| Pts 2 request | 2 C | SC → IFD | Protocol type (0 or 1) |

IV.2.General commands (AOPC = 0)**IV.2.1.Receipt (Control = 00h)**

After complete and errorfree reception by the interface controller of a command and after its fulfilled execution, a receipt is given to the system controller.

IV.2.2.Error detection (Control = 01h)

The error detection message is *the negative receipt*. It is sent by the interface controller to the system controller if the command is not totally understood.

It has one parameter which codes in ASCII the kind of error found by the interface controller :

| Error detection parameter | Meaning |
|---------------------------|---------------------------|
| 'A' (41h) | wrong AOPC |
| 'I' (49h) | wrong IOPC |
| 'P' (50h) | parity error in reception |
| 'O' (4Fh) | any other problem |

IV.2.3.Set in power down (Control = 02h)

This command is used to set the interface controller tda8006 in low power consumption mode. This mode is left as soon as the system controller sends *any commands* to the interface controller.

IV.2.4. Alive (Control = 04h)

This command is used in test to know the mask number (coded in decimal) and the software version (version 1.0 is coded 10h).

It may also be used by the system controller to check the presence and the functionality of the interface controller. The answer from the interface controller is a « ALIVE » message with 2 parameters.

IV.2.5. Software restart (Control = 05h)

A receipt is sent to the system controller once this command is received. The software is completely restarted except that the ram is not cleared and the state of cards remains unchanged. The cards presence messages are sent.

Are totally reset or restarted :

- the card presence, in order to send the card presence message
- the external I/O ports
- the serial UART

The system should not send any command before it has received the cards presence messages.

IV.3. Card commands (AOPC = 1 or 2)

IV.3.1. Reset card (Control = 10h or 20h)

This command is used by the system controller to activate the card and to initiate the communication with the card. This command has one parameter (called activation options) that codes the optional possibilities :

activation options = D7 D6 D5 **D4** **D3** D2 **D1** D0

- **D1 = no warm reset (1)/warm reset (0):** If this bit is set, a warm reset is automatically made if the ATR is considered as erroneous.
- **D3 = no 5V vpp (1)/ 5V vpp (0):** If this bit is set (resp. not set), cards needing a Vpp greater than 5V or equal to 5 V (resp. greater than 5V) are pointed out by a « card communication error » message and remains active.
- **D4 = 3V activation (1) / 5V activation (0):** If this flag is set (resp. not set), a 3V (resp. 5V) activation is made

First, an activation sequence conform to ISO7816-3 normalisation is performed. If the contacts were already active, only the RST pin is switched.

If the card has answered to the reset sequence, the parameters TS, T0, TA1, TB1, TC1, TD1, TA2, TB2, TC2, TD2, TA3, TB3 and TC3 are evaluated by the software. Depending on the activation options, the following evaluations and reactions are made :

| Paramete | Description |
|------------|--|
| TS | 3Bh or 3Fh, depending on the communication convention. Any other values generate a « card communication error » message. |
| T0 | Gives the presence of the following level 1 bytes and the number of historical bytes |
| TA1 | For the tda8004, the F/D ratio values 372 (9600 bauds) or 186 (19200 bauds) are accepted. Any other values generate a « card communication error » message. For the tda8006, any values of F and D, except the RFU values, are accepted. |
| TB1 | If the activation option bit D3 is set (resp. not set), PI1 is equal to 5 (resp. greater than 5) generates a « card communication error » message. |
| TC1 | Codes the extra-guard-time, all values are supported; TC1 = 255 means no guard time (transmission at 12 etu in protocol T = 0, 11 etu in protocol T = 1) |
| TD1 | Indicates the 1st offered protocol that can be used for communicating with the card. <i>If the protocol is different from T = 0 or T =1, a « card communication error » is generated.</i> |
| TA2 | If this parameter is present (resp. not present), the card is in specific mode (resp. negociable mode) |
| TB2 | If activation option bit D3 is set (resp. not set), PI1 greater than 0 (resp. greater than 50) generates a « card communication error. |
| TC2 | Codes Work waiting time Integer (WI) which is used to compute the work waiting time. This time is specific for protocol type T = 0 |

| | |
|------------|--|
| TD2 | Indicates a 2nd offered protocol |
| TA3 | Codes Interface Field Size Integer (ISFI) of the card. This integer is specific for protocol T = 1 |
| TB3 | Codes the Character Waiting Integer (CWI) and the Block Waiting Integer (BWI). These integers are used to compute the Character Waiting Time and the Block Waiting Time which are specific for protocol type T = 1 |
| TC3 | Indicates the type of block error detection code to be used. <i>The Cyclic Redundancy Check is not supported and generates an error message</i> |
| TCK | Gives the value of the exclusive-ORing of all bytes from T0 to the last historical byte. This integer is transmitted for protocol T = 1 |

After a correct decoding, the parameters of ATR are memorised. Moreover, the whole ATR is sent to the system controller in an « answer from card » message. If there is an erroneous parameter, a « card communication error » message is transmitted to the system controller.

IV.3.2. Disconnect card (Opcode = 11h or 21h)

This command deactivates the card according to ISO7816-3 standards. A receipt is returned afterwards.

IV.3.3. Send data to card (Opcode = 12h or 22h)

This command is used to transfer data to the card. It must be used for card commands whose data are expected by the card, like file selection, write operations, a.s.o.

The parameters are the standard data (CLA ... P3 + data bytes).

For a length (P3) of 0, no data are transferred to the card. In this case, the card may or not send an acknowledge byte before the status words. *The maximum value of P3 is 255.*

The result of this command is an « answer from card » message with the status words from the card as parameters.

IV.3.4. Request data from card (Control = 13h or 23h)

This command is used to get data from the card. It must be used for card commands whose data are sent by the card to the interface controller, like status or read operations.

Therefore the parameters are the "header" data (CLA ... P3), where P3 codes the number of data that are expected. The card answers either with the procedure byte, the requested data bytes and the status words, or only with the status words. The procedure byte is analyzed but never transferred to the system controller.

P3 = 0 means 256 data expected by the system controller.

IV.3.5. Answer from card (Control = 17h or 27h)

This command is used by the interface controller to transmit the data received from the card to the system controller. These data - parameters of the command - are either :

- the Answer To Reset
- the Answer of a PTS request
- the status words SW1, SW2 after a send data to card command
- the data and the status words after a request data from card command.

IV.3.6. Card communication error (Control = 1Bh or 2Bh)

This command is sent by the interface controller to the system controller when any problem occurs that may either stop or disturb the communication with the card. It has one byte parameter which gives an error description and its consequences.

The following parameters have been defined in the table below :

| Card communication error parameter | Description of the error |
|------------------------------------|---|
| 'A' (41h) | ATR not valid. There is a wrong parameter of the ATR sent by the card |
| 'B' (42h) | Card mute |
| 'C' (43h) | Automatic emergency deactivation of the card in the event of card take-off, supply voltage drop, shortcircuit or overheating. |
| 'D' (44h) | Bad node address in a prologue field of a block frame in protocol T= 1 |
| 'E' (45h) | Bad protocol control byte in a prologue field of a block frame in protocol T = 1 |
| 'F' (46h) | Bad LRC in an epilogue field of a block frame in protocol T = 1 |
| 'G' (47h) | Card chain abort |
| 'H' (48h) | Card frame error |
| 'I' (49h) | 3 consecutive parity errors in the reception of a character |
| 'J' (4Ah) | Apdu too long. The P3 parameter of the apdu CLA INS P1 P2 P3 D1...Dn is greater than n. |
| 'K' (4Bh) | Apdu too short. The apdu header lacks parameter (f.i. CLA INS P1) |
| 'L' (4Ch) | Card resynchronised in a protocol T = 1 |
| 'M' (4Dh) | PTS request not allowed (TA2 parameter is present in the ATR so the card is already in specific mode) |
| 'N' (4Eh) | Card protocol is neither T=0 nor T=1 |
| 'O' (4Fh) | The PTS request for protocol T=1 is not accepted |
| 'P' (50h) | Procedure byte error. This message is sent if the procedure byte coming from the card is not correct (i.e. <> INS, SW1, NULL). The card remains active. |
| 'Q' (51h) | The PTS answer does not match with the PTS request |
| 'R' (52h) | Checksum error |
| 'S' (53h) | SW1 error, the status byte SW1 is different from 6x or 9x. The card remains active |
| 'T' (54h) | IO line locked |
| 'U' (55h) | The VPP voltage of the card is greater than 5V |
| 'V' (56h) | The VPP voltage of the card is equal 5V (when the activation bit D3 is set) |
| 'W' (57h) | Card absent |
| 'X' (58h) | The F/D ratio proposed in the ATR is not supported. Default F/D ratio 1/1 is selected. |
| 'Y' (59h) | Bad class of card |
| 'Z' (5Ah) | Bad TS (TS is neither 3Bh or 3Fh) |
| 5Bh | The Cyclic Redundancy Check (CRC), used for protocol T=1, is not supported. |
| 5Ch | The number of data expected exceeds 256 bytes |

IV.3.7. PTS request (Control = 1Ch or 2Ch)

This command is used to send a PTS request frame to the card in order to change either the communication protocol or the Fi Di parameters. The Fi Di values (different from the default values) given by the ATR are used for the negotiation. Only the protocol has to be given as a parameter.

When the PTS request is processed, a string of 4 bytes is sent to the smart card :

- PTSS(FFh),
- PTS0(contains the PTS request parameter 0 or 1),
- PTS1(Fi Di given in the ATR of the card)
- and PCK(checksum of 3 previous bytes)

The PTS response is evaluated. If it's matched with the PTS request, it is transferred in an « answer from card » message. Otherwise, a « card communication error » message is sent to the system controller to warn the failure of the PTS request.

IV.3.8. Set card 1 clock (Control = 14h)

This command is used by the system controller to select a new card 1 clock frequency. The chosen configurations are given to the interface controller through the clock status parameter, coded as follows :

| Clock status parameter in hex. | Clock frequency |
|--------------------------------|-----------------------------|
| 0h | clock stopped in low state |
| 1h | Fxtal/2 (7.37 Mhz) |
| 2h | Fxtal/4 (3.68 Mhz) |
| 4h | Fxtal/8 (1.84 Mhz) |
| 8h | Fint/2 (1.2 Mhz) |
| 10h | clock stopped in high state |

At the beginning of a card session, the clock status parameter is initialized at 2 (Fclk = Fxtal/4 -3.68 MHz- during the ATR phase) and remains unchanged until a « set card 1 clock » command is used to change the clock frequency of the card 1.

IV.3.9. Card presence active (Control = 19h or 29h)

Card presence passive (Control = 1Ah or 2Ah)

If the card is inserted (resp. removed) the command card_presence_active (resp. card_presence_passive) is sent to the system controller.

The presence status of the card is controlled during all modes of the TDA8006 even in power-down mode.

The messages are sent in the following cases :

- after power on
- after a hardware or after the general command software restart (see V.2.5)
- after every change of the card presence status.

V. Demoboard description and directions for use

V.1. Hardware description

V.1.1. Schematics

The schematics and hardware informations of the demoboard CAKE 611-A are given in annex.

V.1.2. Power supply

The demoboard CAKE 611-A power supply between 4V and 6V ; 5V supply voltage is recommended.

V.1.3. Interface with the PC

The communication between the PC and the TDA8006 is achieved according to the asynchronous transmission protocol. The software has to translate the informations from the PC into control actions or smart card protocol. It also gives the status of the card and returns card data to the PC.

An interface circuit MAX232C is placed between the interface device TDA8006 and the serial port of the PC. This circuit allows to separate ingoing and outgoing signals.

The board may be connected to a PC via a serial SUB-D connector. The signals level on the SUB-D connector is -10 / +10 V on pins 2 and 3.

V.1.4. Clock frequencies

The circuits are clocked by the on-board 14.745 MHz X-Tal, ensuring a communication baud rate with the PC of 19200 bauds.

V.1.5. Smart card features

The smart card reader 2 accepts ISO and AFNOR position but the smart card reader 1 accepts only ISO position for the card. As the TDA8006 and TDA8004 can not provide a specific VPP programming voltage, the VPP contacts of the card connector are directly linked to the VCC card supply contacts, so that a 5v voltage is provided. It is possible to separate both signals by removing the corresponding strap.

Accepted cards are 3V or 5V asynchronous smart cards using T=0 or T=1 protocols complying with ISO 7816 standards.

V.2. Directions for use

V.2.1. Presentation of the PC driver « double64 »

The figure 2 shows the complete system: host system (PC), smart card interface (CAKE-609A) and smart cards. A driver software called 'Double64' installed in the PC plays the role of the host system. The control and the communication of the interface controller CAKE-611A by the PC driver are made through a standard serial cable.

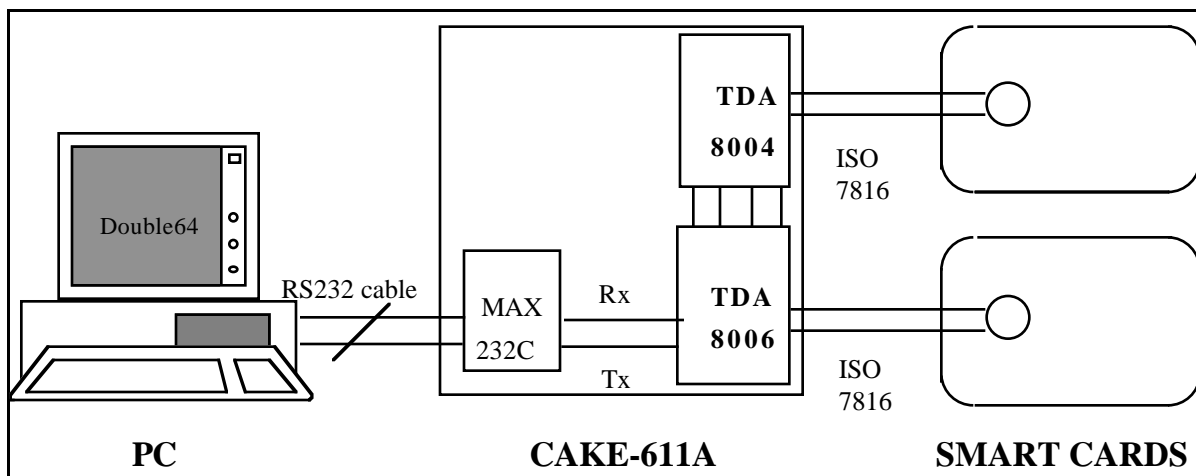


Figure 2: The complete system: PC, CAKE-611A and smart cards

V.2.2. Installation of « double64 »

Once the demonstration board CAKE 611-A is linked to the PC by a SUB-D cable, it can be powered on. In order to process communication between the PC and the smart cards, the DOS software 'DOUBLE64' should be installed on the PC. A floppy disk including the following files are provided with each demo board :

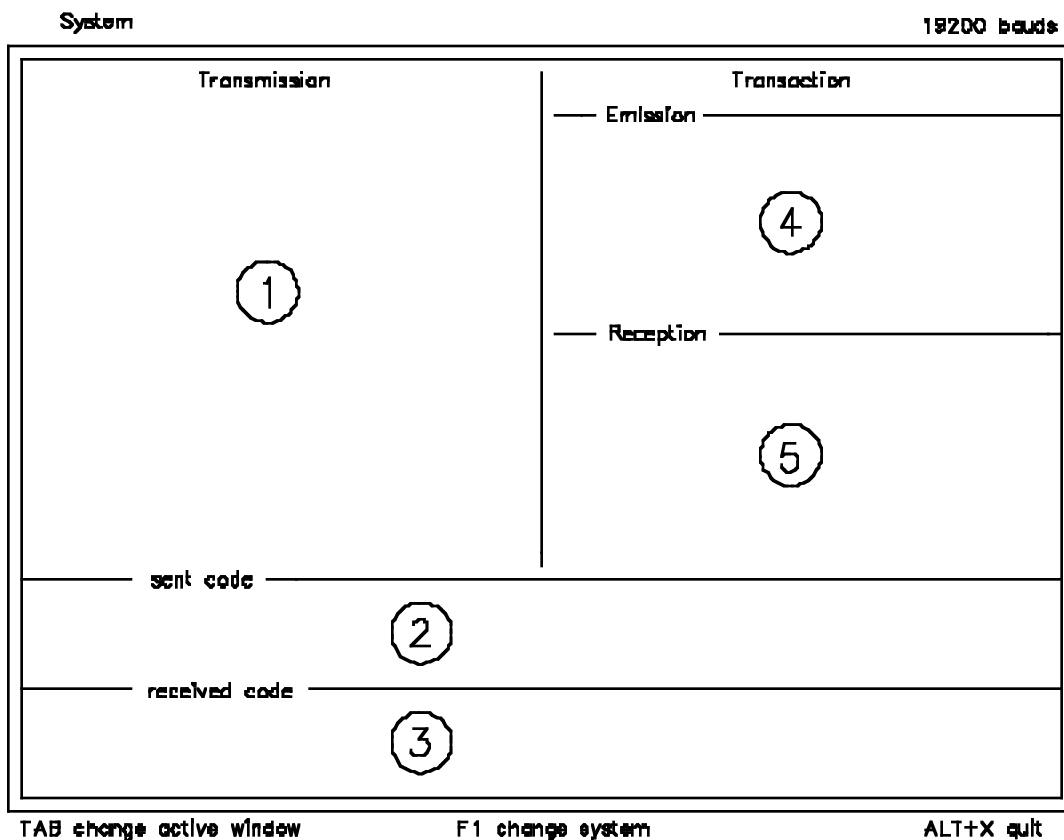
- an executable file DOUBLE64.EXE
- a data file CODE.TEX

V.2.3. Starting «double64»

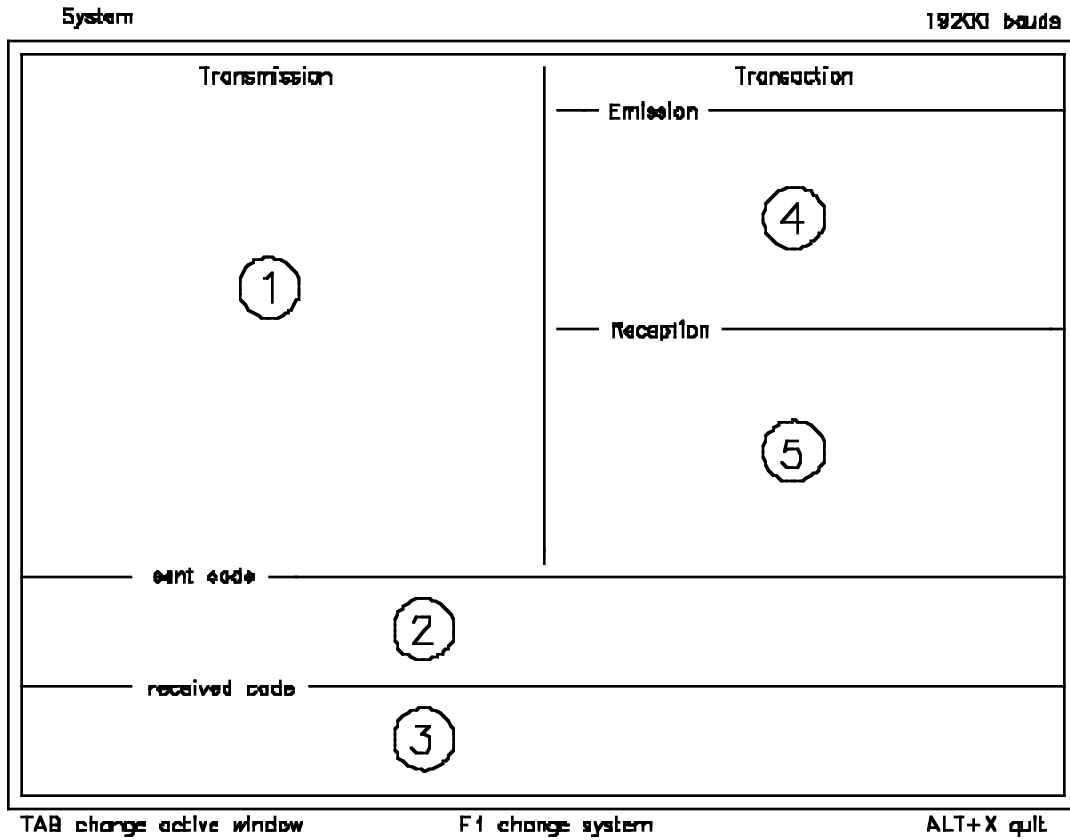
To run this program, type « **double64** » :

- a first window gives the references of the version;
→ press **enter**
- a second window appears with :
What is the name and the path of your file ?
→ *Code.tex*
→ press **enter**
- a third window asks you a question concerning the system
→ type 'd'

The main screen appears as follows:



V.2.4. Description of the interface windows of « double64 »



Window 1 displays the menu where you will find the different commands that can be sent to the cards interface. (see chapter **VI**).

The transaction areas 4 and 5 give the last transmitted or received messages and their hexadecimal values are displayed on windows 2 or 3.

To change from one to the other press **TAB**.

The selected command in window 1 is highlighted in green color. With the up and down arrows the user can move into the window and select the right command. Once the command is selected, press **enter** to send it to the reader.

If necessary, a window appears on the screen in which the instruction bytes or data can be entered. When you have written your data, press **enter** to send the complete message.

Last received or transmitted messages

When the program receives a message, its name is displayed on window 5, and the hexadecimal code in window 3.

When the program sends a message, its name is displayed on window 4 and the corresponding hexadecimal code on window 2.

On windows 2 and 3 the hexadecimal messages are including the protocol bytes as described in § IV.1

Window 4 gives the six last transmitted commands and window 5 the seven last received messages.

To exit the program press **Alt x**.

VI. ANNEX : CAKE 611-A