US 20050009543A1

(54) **ADAPTABLE CHIP CARD**

(75) Inventor: **Adriano Huber**, Locarno (CH)

Correspondence Address:
**PEARNE & GORDON LLP
1801 EAST 9TH STREET
SUITE 1200
CLEVELAND, OH 44114-3108 (US)**

Publication Classification

(57) **ABSTRACT**

Chip card (5) capable of communicating with an external device (3) over a logical interface (4), standardized data units (20) being defined for this communication. The chip card has a ROM and an EEPROM. The functionality that is made available through the chip card when a data unit (20) is received over said logical interface (4) is determined in the EEPROM, for example through an APDU handler (512) implemented in the EEPROM.

Fig. 1

Fig. 2

Fig. 3

Fig. 4

1

2

ME 3

4

51

52

RAM

512

EEPROM
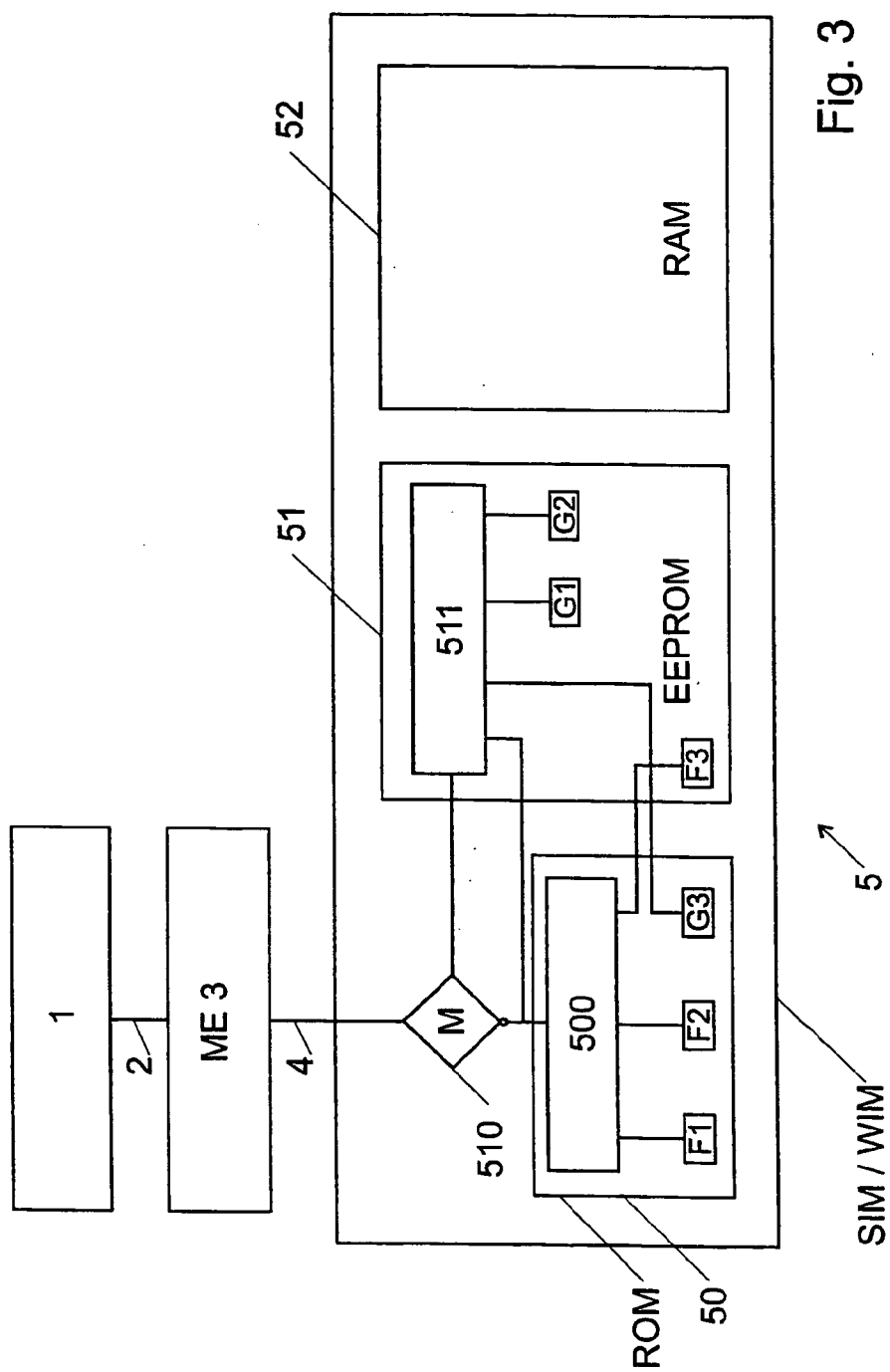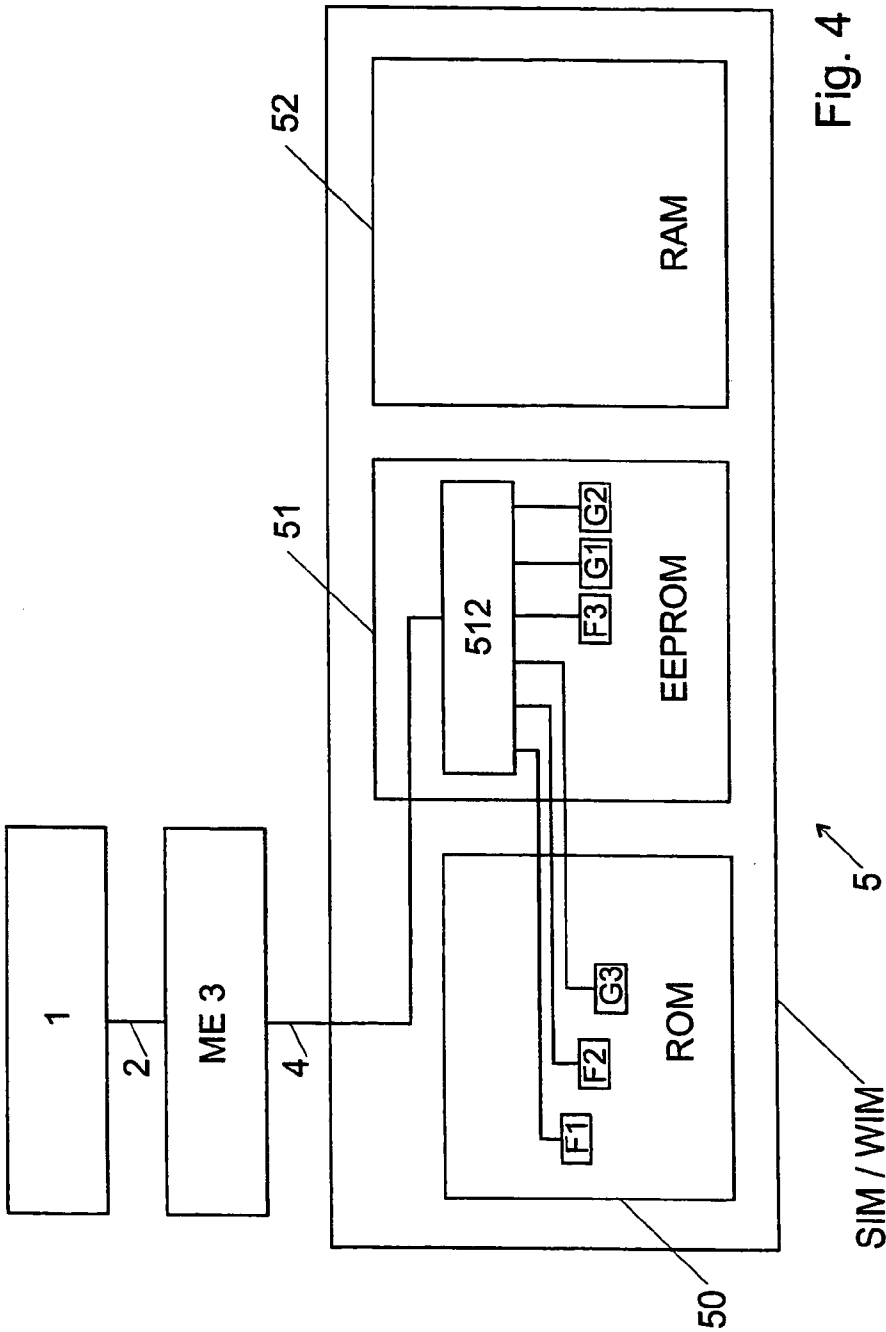
F3 G1 G2

ROM

F1 F2 G3

SIM / WIM

5

50
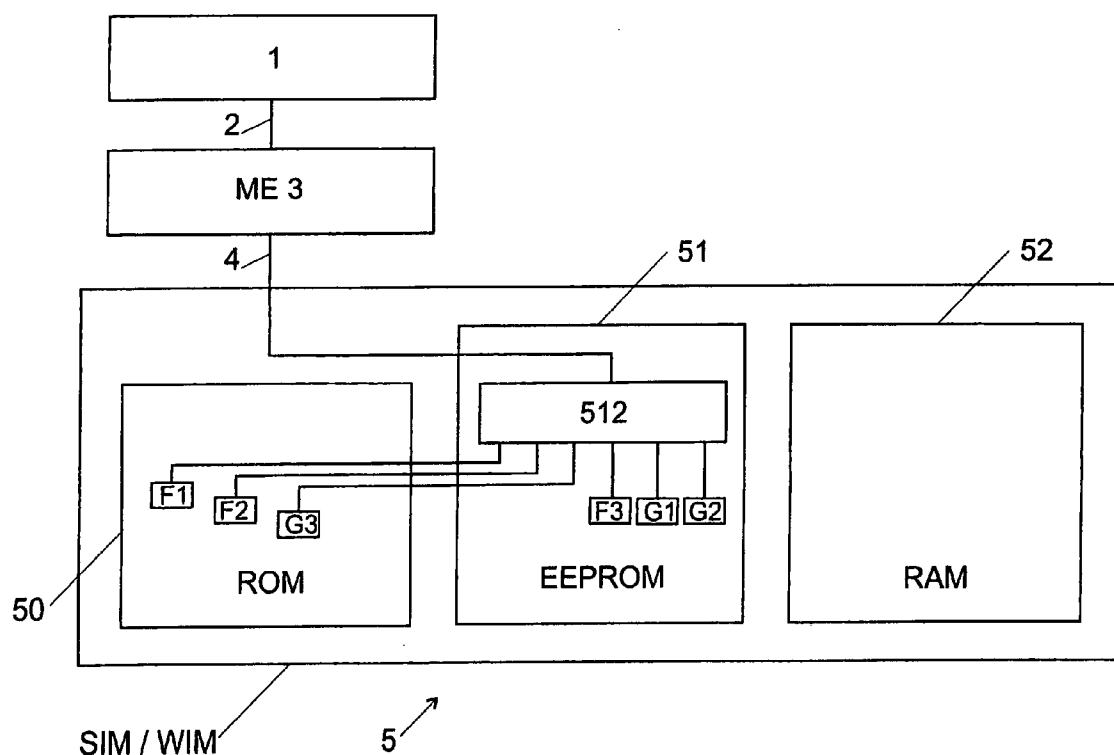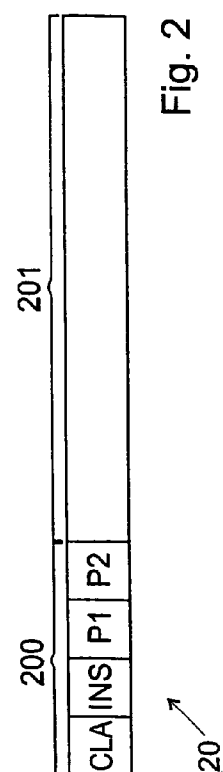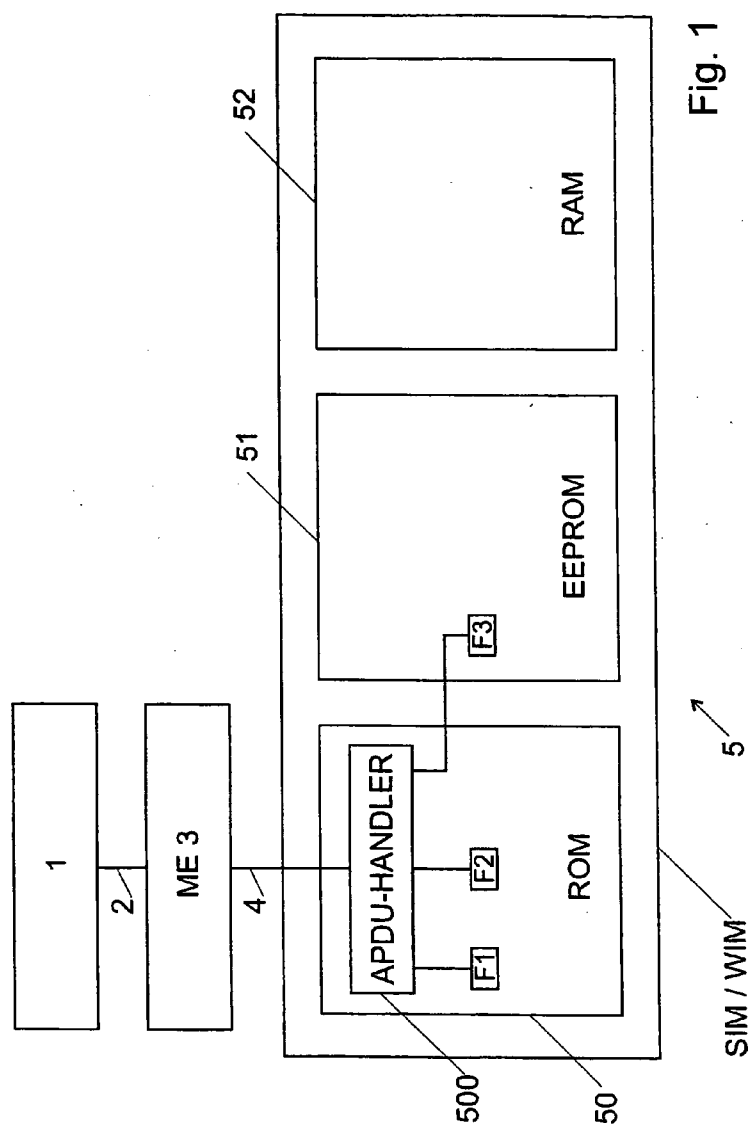
ADAPTABLE CHIP CARD

REFERENCE DATA

[0001] The present application is a continuation of U.S. application Ser. No. 10/150,334 filed May 16, 2002, itself a Continuation under 35 USC§111 of PCT/CH99/00553 filed on Nov. 19, 1999, the contents of which are included by reference.

FIELD OF THE INVENTION

[0002] The present invention concerns a chip card, in particular a chip card that can be adapted after its manufacture and distribution.

RELATED ART

[0003] Chip cards communicate with external devices, for example with a mobile telephone, over a logical interface often called API (Application Protocol Interface). The external device wanting to access a functionality of the chip card sends an order over this API interface. The card that receives an order from the external device can decode it and trigger the corresponding action.

[0004] Such orders are usually encoded as structured data units, called APDU (Application Packet Data Unit). An APDU consists usually of a header and a body, the header indicating the instruction to be performed and the instruction parameters whilst the body, which is not always present, contains data. This protocol is described among others in the ISO norm ISO/IEC 7816, part 4. Further orders are specified in system specific documents (e.g. GSM 11.11, GSM 11.14).

[0005] The definition of the API interface and the standardization of the APDU so as to be recognized by chip cards are developed mainly in standardization forums or are proposed by individual firms. For reasons of compatibility between chip cards and devices of different manufacturers, a strong international standardization of the API interface is a desirable aim.

[0006] The standardization of APDU orders is however a process that often happens more slowly than the technical development of the chip cards. For this reason, conventional chip cards cannot fulfil all the wishes of the users and in particular not all the wishes of the services providers or of the telecommunication network operators. Functionalities of a chip card that can technically already be realized and are even sometimes implemented in the card can often be offered only after the next standardization of the APDU specification.

[0007] WO9819237 (Schlumberger) describes a chip card with a virtual Java machine in which different applications are stored in an EEPROM. Applications can thus be modified or replaced. The instruction handler is however stored in the ROM, so that no new instructions can be added to the existing set of instructions.

[0008] It is therefore an aim of this invention to propose a method with which an external device can access the functionalities of a card for which the APDU has been standardized only after the manufacture of the chip card.

BRIEF SUMMARY OF THE INVENTION

[0009] According to the present invention, these aims are achieved in particular through the characteristics of the independent claims. Further advantageous embodiments are moreover described in the dependent claims and in the description.

[0010] In particular, these aims are achieved in that the functionality made available through the chip card when a data unit is received over the API interface is decoded, interpreted and performed by the application logic in the EEPROM part of the chip card.

[0011] In this way, the so-called APDU handler in the chip card and which determines the reaction to incoming data units can be modified, for example completed, renewed or replaced, at any moment in order to take into account newly defined, proprietary or non standardized APDU.

DESCRIPTION OF THE DRAWINGS

[0012] Hereafter, preferred embodiments of the invention will be described in more detail with the aid of the attached figures, in which:

[0013] FIG. 1 shows a diagrammatic view of a system with a conventional chip card.

[0014] FIG. 2 shows a diagrammatic view of an APDU order according to the invention.

[0015] FIG. 3 shows a diagrammatic view of a system with a chip card according to a first embodiment of the invention.

[0016] FIG. 4 shows a diagrammatic view of a system with a chip card according to a second embodiment of the invention.

DETAILED DESCRIPTION OF THE INVENTION

[0017] The following description refers to the special case of a chip card that is used as an identification module in a digital cellular mobile telephone, for example a SIM (Subscriber Identification Module) card or a WIM (Wireless Application protocol Identification Module) card, or WIM/SM (SIM with WIM application) that can be used for example in a GSM (Global System for Mobile Telecommunication), WAP (Wireless Application Protocol) or UMTS mobile telephone. The one skilled in the art will however understand that the present invention can also be applied to other types of chip cards, for example with JAVA (trademark of SUN Microsystems) or with OpenCard cards.

[0018] FIG. 1 shows diagrammatically an example of a known conventional system. The system comprises a terminal 3 (Mobile Equipment ME) and a conventional SIM or WIM card 5. An application 1, for example a browser or an application program, for example a program from a service provider, is executed by data processing means (not represented) in the mobile equipment 3. The application communicates with the mobile equipment 3 over an API (Application Protocol Interface) 2 that is here not detailed.

[0019] The mobile equipment 3 communicates with the SIM card 5 over another API interface 4. The mobile equipment 3 usually takes over the role of master of the communication over this interface whilst the SIM card 5 answers as slave. Protocol variants are however also known (for example according to SIM-Toolkit, i.e. GSM 11.14) in which the SIM card takes over, at least temporarily, the role of master.

2

[0020] A step in the API protocol consists in an order being sent to the chip card **5**, in the card executing the order and if necessary in a reply being sent to the mobile equipment **3**. Thus either orders or replies are exchanged over the interface **4**. Data can be contained both in the orders and/or in the replies.

[0021] These questions and orders are encoded with standardized APDU (Application Protocol Data Unit). Some standardized APDU for ISO chip cards are described in the norm ISO/IEC 7816-4:1995(E). Additional APDU have been defined for SIM cards and for WIM cards in order to expand the functionalities of the API interface **4**.

[0022] **FIG. 2** shows the typical structure of an APDU **20**. The data unit comprises a compulsory header **200** with four bytes and an optional body **201** of variable length.

[0023] The header contains a first byte (CLA) used to indicate among others which API version the data unit complies with. A second byte (INS) gives the instruction that is to be executed or has just been executed by the chip card **5**. Depending on the instructions, two further parameters P1 and P2 can be indicated. If an instruction does not require a parameter P1 and/or P2, this parameter must be set to zero.

[0024] The optional body **201** contains dada and at least one byte indicating the length of these data and/or the maximal length of the expected reply.

[0025] The conventional chip card **5** comprises generally a ROM **50**, an EEPROM **51** and a RAM **52**, as well as data processing means (not representted). The ROM nowadays typically has 64 Kbytes and usually includes the operating system, a Java virtual machine (JVM, trademark of SUN Micro-systems) and different components F1, F2, . . . , for example JavaBeans, Applets or programs for making available the card's different functionalities. Components are also stored in the EEPROM **51**, for example components that have been downloaded as applet after the personalization of the card as well as personal user data, for example a telephone directory, an electronic certificate, etc. The EEPROM can for example comprise 16 or 32 Kbytes. Temporary data are stored in the RAM **52**, for example temporary variables. Applications (applets) can furthermore be stored in the EEPROM.

[0026] A data unit handler **500**, often called APDU handler, receives the APDU of the mobile equipment **3** received over the API interface **4** and analyses the header **200**, in particular the bytes CLA and INS, in order to forward the order to the appropriate component F1, F2, . . . that can execute the order. The APDU handler **500** can then receive the reply of this component and send this reply over the API interface **4** to the mobile equipment **3**.

[0027] The APDU handler **500** is usually implemented in the ROM part **50**. The program is executed by the data processing means when an incoming APDU is detected. It is however also possible to devise the APDU handler as a circuit. A problem with this configuration is that the mobile equipment **3** and the application cannot access new functionalities of the card made available through new components G1, G2, G3 in the ROM, EEPROM or RAM, as long as no APDU **20** has been standardized for this functionality. Even when such an APDU has been defined and standardized, chip cards **5** that have already been distributed cannot offer this functionnality if they only have a conventional

APDU handler **500** unable to recognize this APDU that has been defined later, even if the required component G1, G2, . . . is contained in the card. For these reasons, new functionalities that would be possible through the continuous and fast development of chip cards can be distributed only slowly.

[0028] **FIG. 3** shows an example of a first embodiment of a chip card **5** according to the invention, for example a SIM or WIM card that can solve these problems. In this embodiment, the chip card has a test module **510** that checks all APDU **20** received over the API interface **4**. If the test module detects that the received APDU was already provided at the time the ROM part **50** was manufactured, it forward this APDU to a conventional APDU handler **500** in the ROM part **50** that makes available the corresponding functionality F1, F2, . . . as described above. If, on the other hand, the test module detects that the received APDU **20** was not provided at the time the ROM part **50** was manufactured, for example because it was standarddized only in a later version of the API interface **4**, it forwards this APDU to a new, modified APDU handler **511** that calls up the component G1, G2, G3, . . . provided for this new APDU in the ROM or EEPROM.

[0029] The test as to whether the received APDU should be sent to the conventional APDU handler **500** or to a new APDU handler **511** can be performed in various ways. In a first embodiment, the module **510** contains a list of APDU that are to be forwarded to the conventional APDU handler **500** and/or a list of newly defined APDU that are to be directed to the new APDU. This list is preferably located also in the EEPROM and can thus be adapted. In another embodiment, the APDU handler **500** directs the received APDU to the new APDU handler **511** only if it receives an error message from the conventional APDU handler, namely if the conventional APDU handler cannot process this APDU. In another embodiment, the test module **510** can itself determine, on the basis of the CLA or INS bytes, to which APDU handler the received APDU **20** is to be forwarded. In yet another embodiment, newly defined APDU that are to be forwarded to the new APDU handler **511** are specially marked, for example by means of being misused, for example by containing special, impossible or very improbable parameters P1, P2 or data **201**.

[0030] The new APDU handler **511** is preferably located in the EEPROM **51**. In this manner, it can also be altered after the manufacture of the ROM **50**, in order for example to ensure access to new components G1, G2, G3 that have meanwhile been implemented. Preferably, the test module **510** is also located in the EEPROM so that the test conditions can be set after the manufacture of the ROM and/or after the distribution of the chip cards. The test module **510** could however also be located in the ROM **50** or even in the RAM **52**.

[0031] The new components G1, G2, G3, . . . can be contained in the ROM **50** or in the EEPROM **51**. Further components temporarily stored in the RAM **52** can also be made available. If the chip card **5** can play the role of master of the interface **4** at least temporarily, it can with this mechanism access also components, for example applets, that are made available by the mobile equipment **3**. It is even possible for the APDU handler **501** to access components that are located in an external device connected with the

mobile equipment **3**, for example over a contactless interface, for example an infrared, RFID or Bluetooth interface.

[0032] In a variant embodiment of the invention (not represented), the APDU handler **500** is modified in the ROM part **50** so that it automatically forwards all received APDU that are not known to a new adaptable APDU handler **511** in the EEPROM **51**. This embodiment makes it possible to forgo the test module **510**: that which cannot be processed by the conventional APDU handler **500** is automatically transmitted to the new APDU handler **511**.

[0033] **FIG. 4** shows a further variant embodiment of a chip card according to the invention. The same or similar features are referred to in all figures with the same reference numbers and will not be described again unless this is necessary. In this embodiment, the normal APDU handler usually located in the ROM **50** has been replaced with a modified APDU handler **512** in the EEPROM. In this manner, it can be adapted at any time in order to take into account newly defined APDU.

[0034] This variant embodiment allows a great flexibility, since new components G1, G2, G3 and new APDU can be implemented at any time in the card **5**.

[0035] In a variant embodiment (not represented) the modified APDU handler is also located in the RAM **52** so that it can easily be modified through the chip card's operating system or through the application **1**.

[0036] In this manner, an application **1** can access functionalities of a chip card **5** that were not provided at the time of manufacture of the card and for which no APDU has been standardized, for example functionalities G1, G2, G3 ... that were downloaded as applet over an air interface only after the personalization or even after the distribution of the card.

[0037] For example, the test module **510** and/or the new APDU handler **511** resp. **512** can be adapted with a message over a mobile radio network, for example with a SMS message, a USSD message a GPRS packet, an IP packet, etc. The test module **510** and/or the APDU handler can also be modified or completed by an application **1**, for example a browser, that has been loaded for example over the air interface or with a data carrier.

[0038] In a variant embodiment of the invention, each APDU can trigger the execution of several different functions in the chip card **5**. A special APDU defines in which state the card is and which set of functions is to be used in future. This variant allows for all functionalities of the chip card to be replaced by sending a single APDU, by means of incoming APDU being processed by a new APDU handler. In this manner, for example, all parameters and components stored in the EEPROM **51** can temporarily be replaced. Another special APDU can then return the chip card to its normal state.

1. A method for adapting a chip card after its manufacture to a new standardization of the API interface, comprising the steps of:

    adapting a module in a memory of the chip card to determine a proper functionality when said certain data unit is received; and

    receiving said certain data unit over said API interface.

2. The method of claim 1, wherein said module is adapted via a message received over a mobile radio network.

3. The method of claim 2, wherein said message is an SMS message.

4. The method of claim 2, wherein said message is a USSD message.

5. The method of claim 2, wherein said message is a GPRS packet.

6. The method of claim 2, wherein said message is an IP packet.

* * * * *