

Method of modifying the instruction set of a smart card.

BACKGROUND OF THE INVENTION

The present invention relates to a method of modifying the instruction set of a smart card. More particularly, the present invention relates to a method of modifying individual instructions of a smart card.

5 A smart card, also called IC card (IC = Integrated Circuit), is an electronic device for effecting payments. The integrated circuit of the card contains a microprocessor or the like for processing data. The processor of a smart card is normally provided with an instruction set comprising instructions (commands) such as READ, WRITE, UPDATE, etc., for operating upon the (payment) data contained in the card. Often standardized instruction sets are used, such as e.g. defined by European Norm EN726. However, such an instruction set is necessarily limited with respect to the number and the scope of the instructions.

10 As electronic payment systems involving smart cards develop, the need often arises to add new functions to the card, or to alter existing functions. Such changes usually imply a modification of the instruction set of the smart card. This would normally involve a redesign of the integrated circuit of the card, which is not only expensive but renders existing cards obsolete. Furthermore, altering a standardized instruction set would jeopardize its compatibility with the standard.

For these and other reasons, the need arises to be able to modify the instruction set of smart cards without substantially deviating from the existing set.

SUMMARY OF THE INVENTION

It is thus an object of the invention to overcome the above-mentioned and other disadvantages of the prior art and to provide a method which makes it possible to modify an instruction set of a smart card without adding commands. It is a further object of the invention to add new functions to a smart card, the new functions being based on the original instruction set.

In order to achieve these and other objectives, the invention provides a method of modifying the instruction set of a processor of a smart card, the smart card further comprising a memory for storing files and groups of attributes associated with the files, each

attribute specifying properties of an instruction of the set, the method comprising the step of inserting into a group of attributes a new attribute corresponding with an instruction to be modified, said new attribute specifying a modified property of the respective instruction.

That is, according to the present invention instructions are modified by modifying their attributes. As the attributes may be organized in groups associated with specific files, the method of the present invention allows instructions to be selectively modified, i.e. modified for specific files. An attribute typically specifies an access condition or a specific key to be used by the instruction concerned. However, the attribute may contain machine code (or a pointer to machine code) specifying a further operation, the operation performing a specific function (such as a cryptographic procedure). By inserting a new operation, or a modified operation, in an attribute, the function of an instruction may be modified without altering the instruction proper.

It will be understood that the inserting of an attribute may involve the replacement of an existing attribute. In general, memory locations may be reserved for storing attributes. The insertion of a new attribute generally involves the storing of the new attribute in the corresponding memory location, in which case any information previously stored in the memory location is overwritten.

The method according to the invention may be applied in situations where the groups of attributes of the instruction set comprise a first table containing first attributes and a second table containing second attributes, the first table containing references to the second table and wherein a new attribute is inserted in the second table. Such a first table may e.g. comprise said specific key, while the second table contains procedures (operations) to be used in conjunction with the instruction concerned. It should be noted that the first table may be referred to by a base table containing references. By inserting a new attribute in the second table, the modification is achieved with a minimal change in the attributes.

Modern smart cards may have at least two different data files, e.g. for different applications of the smart card. In such a case, a group of attributes may comprise subgroups associated with the respective data files. The invention allows a new attribute to be

inserted in a limited number of subgroups, thus making instruction modifications specific for individual data files. This way, a great flexibility of instructions may be achieved.

5 BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 shows, in perspective, a smart card as used for payment and identification purposes.

Fig. 2 schematically shows the integrated circuit of the smart card of Fig. 1.

10 Fig. 3 schematically shows an attribute table of an instruction set of a smart card, e.g. the smart card of Fig. 2.

Fig. 4 schematically shows a key table associated with the attribute table of Fig. 3.

15 Fig. 5 schematically shows a procedure table associated with the attribute table of Fig. 3 and the key table of Fig. 4.

Fig. 6 schematically shows an example of a hierarchical file structure of a smart card.

EXEMPLARY EMBODIMENTS

20 The smart card or IC card 1 shown schematically and by way of example in Fig. 1 comprises a substrate 2, in which an integrated circuit is embedded. The integrated circuit is provided with contacts 3 for contacting a card reader or the like. It should be noted that the present invention can also be applied in the case of so-called
25 contactless smart cards.

The integrated circuit 10 shown schematically and by way of example in Fig. 2 comprises a processor 11, a memory 12 and an input/output circuit 13. The memory may comprise a volatile (RAM) memory part for temporarily storing data and a non-volatile (ROM) memory part for permanently or semi-permanently storing data. The
30 latter part is preferably an EEPROM (Electrically Erasable Read Only Memory) type memory. The data stored in the non-volatile part may contain both programming data (instructions, programs) and payment data, i.e. data relating to monetary transactions. It will be
35 understood that a separate memory (not shown) may be provided to store the instruction set of the processor 11.

The processor 11 may be a microprocessor having a standardized instruction set, for example in accordance with European Norm EN726.

It will be understood that other standard, or non-standard instruction sets, may also be used. The instruction set will be further explained with reference to Fig. 3.

5 The input/output circuit 13 handles the communication between the processor 11 and external devices (not shown) such as card readers. The input-output-circuit 13, which may contain suitable buffers, is connected with the processor 11 via a data bus. The processor 11 and the memory 12 may be connected by means of the same data bus.

10 The attribute table shown schematically and by way of example in Fig. 3 and denoted in general by the reference numeral 100 comprises several parts, associated with different files (a possible file structure of the smart card will later be explained with reference to Fig. 6). A first part, associated with file A (21 in Fig. 6),
15 comprises the fields 111, 121 and 131. The first field 111 contains the file indicator, here denoted by "File A". The second field 121 contains an instruction table, i.e a table containing the instruction set of the smart card, or at least references to the instructions of the instruction set. It will be understood that the instructions
20 proper comprise the machine code executed by the processor. In the example shown, the instructions "READ", "WRITE" and "UPDATE" are shown. The third field 131 contains attributes of the instructions in field 121. That is, for each instruction of the second field 121 there may be an entry in the third field 131, the entry constituting a group
25 of attributes which further specify the functioning of the instruction. The attributes may for example comprise access conditions and key numbers, denoted in Fig. 3 by AC and KEYNR, respectively. Access conditions may comprise ALWAYS, NEVER, PROTECTED, and the like. Key numbers may refer to entries in the Key Table depicted in Fig. 4.
30 The third field 131 of the table 100 may thus constitute a base table of attributes, which refers to further tables (depicted in Fig. 4).

Although the first, second and third fields of the attributes table 100 are shown in Fig. 3 for the sake of clarity as adjacent fields, in actual embodiments the fields may very well be stored at
35 different, non-adjacent locations of the smart card's memory 12.

The attributes listed in field 131 all correspond with File A, as specified in field 111. That is, the attributes of field 131 are only valid for File A. Such a structure allows instructions to have

different functionalities for different (data) files. Such a file may contain data with respect to a specific application (function) of the smart card. Typical applications are "purse", "loyalty program", and "access", for making or receiving payments, collecting loyalty credits, and gaining access to buildings and the like, respectively. Even within such an application more than one data file may be used, and thus more than one associated group of attributes may exist. As a result of the attributes being specific to a data file, the function of an instruction may, at least for the parts defined by attributes, be specific to a data file. Thus, the access conditions of the instruction "READ" may vary between a file containing e.g. purse data (storage of electronic money) and a file containing personal data of the user of the smart card. This will further be explained with reference to Figs. 4 and 5.

The actual access conditions, shown symbolically in Fig. 3 by AC, may be contained in the base attributes table 100, or may be contained in a separate table (not shown). Similarly, the keys (KEY 1, KEY 2, ...) shown in Fig. 4 may be contained in the attributes table 100. For the sake of the explanation of the invention, it will be assumed that the entries KEYNR, which may simply numbers (1, 2, ...), refer to the Key Table 200 of Fig. 4.

Each entry in the Key Table 200 of Fig. 4 contains a key number (KEYNR.), a key (KEY 1, KEY 2, ...) and a procedure identification or procedure number (denoted PRONR in Figs. 4 and 5). Each procedure identification refers in turn to an entry in the Procedure Table 300. Each entry in the Procedure Table 300 contains a procedure number and a procedure (operation), denoted in Fig. 5 by e.g. DES or RSA. Such a procedure may be a cryptographic procedure, such as DES and RSA, or some other operation on data.

In accordance with the present invention, the instructions of a smart card may be modified by modifying their attributes. Thus by modifying an entry in the Procedure Table 300, a new function or a modification of an existing function may be achieved. In Fig. 5, a new function (e.g. multiply data with a common factor, or subtract one from data) is entered at the second entry of the Procedure Table. Thus if in the Key Table 200 Key nr. 3 is selected, the Procedure Number refers to procedure nr. 2, being the new function. Each time an instruction is executed which involves the use of KEY 3, the new

function will be invoked.

It will be understood that references to a new function can be made in various ways, e.g. directly from the table 100 of Fig. 3. Also, the Key Table 200 may contain "dummy" keys so as to allow the use of the table without actually using a key.

Instead of inserting the new function in the Procedure Table 300, a pointer may be inserted which points to a memory location where the new or modified code is stored.

As explained before, in the base attribute table (100 in Fig. 3) different attributes may exist for different files. The method according to the invention allows for a flexible instruction set by modifying attributes for individual data files.

Smart cards may comprise a hierarchical directory structure for organizing their files. An example of such a structure is shown in Fig. 6. A main file 20 has three subfiles 21, 22 and 23. That is, the main file may contain references to the subfiles, or such references are stored elsewhere (e.g. in a common directory). A group of attributes is associated with the main file 20. It is possible to have individual groups of attributes for all three subfiles 21, 22, and 23, as depicted in Fig. 3. However, in case a group of attributes is associated with file 20, said group of attributes may be applicable to all files lower in the hierarchy relative to the main file 20. That is, the same group of attributes may be associated with files 21, 22 and 23 as well. In other words, if an instruction is changed by modifying an attribute (of a group of attributes) associated with a certain file, the modification may hold for all files hierarchically lower relative to said file. This allows an instruction to be altered for e.g. a specific card application where this application involves several files.

As is explained above, the modification of instructions is achieved in accordance with the present invention by replacing and/or modifying attributes. The method of the invention thus enables the use of standard instructions (standard card functions) to effect non-standard operations. The invention also provides a modified instruction set of a smart card, as well as a smart card provided with a modified instruction set. The actual inserting of a new attribute into an attributes table is done by storing the new attribute at an appropriate memory location. The data representing the new attribute

may be transferred from a computer via a card reader/writer to the memory of the card. Referring to Fig. 2, the data are transferred via the I/O unit 13 to the memory 12 in a usual manner.

- 5 It will be understood by those skilled in the art that the embodiments described above are given by way of example only and that many modifications and additions are possible without departing from the scope of the present invention.

CLAIMS

1. Method of modifying the instruction set of a processor (11) of a smart card (1), the smart card further comprising a memory (12) for storing files (21, 22) and groups of attributes (e.g. AC, KEYNR, KEY
5 1) associated with the files, each attribute specifying properties of an instruction (e.g. READ) of the set, the method comprising the step of inserting into a group of attributes a new attribute (e.g. NEW FUNCTION) corresponding with an instruction to be modified, said new attribute specifying a modified property of the respective
10 instruction.
2. Method according to claim 1, wherein the groups of attributes are organized in a first table (100) referring to a further table (200, 300)
and wherein a new attribute (NEW FUNCTION) is inserted in a further
15 table (e.g. 300).
3. Method according to claim 1 or 2, wherein the insertion of a new attribute involves the replacement of an existing attribute.
4. Method according to any of the preceding claims, wherein the memory (12) of the smart card (1) comprises at least two files (e.g.
20 21, 22), a group of attributes (e.g. 131) being associated with each file, a new attribute being inserted in a limited number of groups.
5. Method according to any of the preceding claims, wherein the files (e.g. 21, 22) are hierarchically organized, each group of attributes being associated with at least one file (e.g. 20) and all
25 files (e.g. 21, 22) hierarchically lower relative to said file (20).
6. Method according to any of the preceding claims, wherein a set of attributes is associated with all files of the smart card (1).
7. Instruction set of a smart card (1), modified by the method according to any of the preceding claims.
- 30 8. Smart card (1), comprising a memory (12) and a processor (11) having an instruction set modified by the method according to any of the claims 1 through 6.

