Protection of a Programmable Memory Against Unauthorized Modification

An apparatus comprises a programmable memory (24), a data write path (30) for writing data into the memory (24) and a data read path (38) for reading data from the memory (24). The memory (24) comprises at least one protected memory field (42). The data write path (30) comprises a decryption unit (32) that is adapted for receiving encrypted data (ED), decrypting the encrypted data (ED), and writing resulting plain data (PD) into the at least one protected memory field (42). The data read path (38) is adapted for reading out the plain data (PD) stored in the protected memory field (42). The at least one protected memory field (42) is only writable by applying the data to be written into the at least one protected memory field (42) in encrypted form to the data write path (30). The invention further comprises an electronic device (10), an external data source (44), a method for accessing the programmable memory (24), and a method for providing an update to an electronic device (10). The invention provides a technique for protecting the integrity of the electronic device (10) by preventing attacks in which unauthorized data - the data being program code and/or other information - is stored in the programmable memory (24).
PROTECTING A PROGRAMMABLE MEMORY AGAINST UNAUTHORIZED MODIFICATION

FIELD OF THE INVENTION

The present invention relates generally to the field of protecting the integrity of programmed electronic devices. In particular, the present invention relates to the field of protecting a programmable memory against unauthorized modification of its contents.

BACKGROUND OF THE INVENTION

Programmed electronic devices have become ubiquitous. Most of these devices contain a programmable memory like, for example, a Flash memory or an EEPROM memory. It is generally desirable to provide at least some level of assurance of the integrity of the contents of the programmable memory. These contents may comprise program code for execution by the programmed electronic device and/or other information like, for example, identification data, configuration data, and user data.

Any unauthorized modification of the contents of the programmable memory may have undesired or even potentially disastrous consequences. For example, if the programmed electronic device is an automotive control system, any tampering with the software stored in the device may be very dangerous. As another example, if the programmed electronic device provides a media playback function, an unauthorized software modification may circumvent digital rights management settings or other restrictions. As a further example, any possibility to change a serial number or similar identifying information stored in a mobile device - for example, the IMEI of a mobile telephone - might be used for fraudulent purposes.
It is known at least in the field of mobile telephones to perform an integrity check at the time of starting up the device. This integrity check may cover program code and/or other critical information. The integrity check comprises calculating a signature of the data to be checked and comparing the calculated signature with a signature that is stored in the device. The signature calculation is performed using a cryptographic method that ensures that it is impossible to alter the data without altering the calculated signature at the same time. A number of suitable methods like, for example, the RSA, DSA and HMAC methods are well known in the art. International patent application No. PCT/EP 2006/009690 of Agere Systems Inc., filed on October 06, 2006, describes further details of a startup integrity checking method.

Performing an integrity check only when starting up the device is not effective against unauthorized modifications that occur after the integrity check. This difficulty could be overcome by performing the integrity check periodically during operation of the device. However, such regular integrity checks are difficult to implement in architectures that do not provide a non-maskable interrupt. Furthermore, a periodic integrity check would cause a significant processor load and increase the power consumption of the device. This is especially a problem for mobile devices because such devices have limited processor resources and limited battery capacity. A further problem in connection with all kinds of integrity checks is that the software that implements the integrity check must be trusted and must be stored in an un-modifiable memory like, for example, a ROM.

OBJECTS AND SUMMARY OF THE INVENTION

It is therefore an object of the present invention to provide a technique for protecting the integrity of programmed electronic devices. It is a further object of the invention to provide a technique for preventing attacks in which unauthorized data - the data being program code and/or other information - is stored in a programmable memory of a device.
The present invention is defined by the independent claims. The dependent claims concern optional features of some embodiments of the invention.

The present invention is based on the fundamental idea to provide a data write path of a programmable memory with a decryption unit. At least one protected memory field - and in some embodiments the entire programmable memory - is only programmable via this decryption unit. In other words, in order to write data into the protected memory field, the data must be encrypted before it is applied to the data write path. The decryption unit will then decrypt the encrypted data and provide the decrypted data to the protected memory field where the data is stored in decrypted form, i.e., as a plain data. Of course, the encryption must match the decryption performed by the decryption unit if the desired plain data is to be written into the memory field.

The present invention ensures that a useful modification of the contents of the at least one protected memory field is only possible if a suitably encrypted version of the plain data to be written into the memory is available. An attacker who does not have the necessary information to prepare this encrypted version cannot modify the memory contents in any meaningful way. It might still be possible for the attacker to write data into the memory, but this data - being the result of the decryption process performed in the data write path - will essentially be random information and can easily be identified. For example, this data will not be executable program code. Only an authorized entity that has access to the secret key and other information used in the decryption process is able to create the required encrypted version of the data.

In the present document, the term "data" comprises all information stored in the programmable memory. For example, such data may be program code for execution by a CPU or other information stored in the programmable memory.

In some embodiments of the invention, there is no performance penalty for the data read operation because the plain data is stored in the memory and can easily
be accessed. The data read path is free of any cryptographic processing elements in these embodiments.

Writing data into the memory entails a decryption and therefore requires some processing time. However, in some embodiments this processing is performed concurrently with the physical memory write operation. Depending on the memory technology used, the speed of the physical memory write operation may actually be the limiting factor in some embodiments. Furthermore, if the invention is used to write blocks of consecutive memory cells - like, for example, when performing a software update - , some embodiments provide for an efficient decryption by using a block operation mode.

In some embodiments, the decryption - and the corresponding encryption when generating the encrypted data - are performed by a symmetric method with a secret key. This secret key may be stored in a dedicated decryption key storage that is only readable by the decryption unit. It is understood that the entity that creates the encrypted data - like, for example, an external data source - must also have access to the secret key. For example, in some embodiments of the invention the secret key may also be stored in an external database.

The electronic device of embodiments of the present invention may be a mobile device and/or a communication device and/or an embedded device and/or an authentication device. Examples of an authentication device are a SIM (subscriber identity module) or an RFID tag or device.

BRIEF DESCRIPTION OF THE DRAWINGS

Further features, objects and advantages of the invention will become apparent when studying the following detailed description, in connection with the annexed drawings, in which:
Fig. 1 is a schematic block diagram of a device comprising a programmable memory and an external data source, according to a first embodiment of the present invention.

Fig. 2 is a sequence diagram showing a memory write access and a memory read access in the device of Fig. 1, and Fig. 3 is a schematic block diagram as in Fig. 1, according to a second embodiment of the invention.

DETAILED DESCRIPTION OF EMBODIMENTS OF THE INVENTION

Fig. 1 shows an example of a device 10 comprising a controller unit 12 and a memory unit 14. In the present sample embodiment, the device 10 is a mobile telephone, the controller unit 12 is a baseband integrated circuit, and the memory unit 14 is an integrated Flash memory circuit. However, the present invention is not limited to the field of mobile telephones. The invention is applicable to all kinds of programmed electronic devices that are to provide some level of protection against execution of unauthorized software and/or against unauthorized modification of data stored therein. Such devices include, without limitation, mobile communication devices, media playback devices, embedded systems, devices for automotive and medical use, external memory devices, authentication devices, and so on. Depending on its type, the device 10 may comprise further components like, for example, a power supply, input and output elements, a high frequency unit, and so on. These further components are not shown in Fig. 1.

The controller unit 12 comprises a CPU (central processing unit) 16 that is connected to a memory interface 18 via an internal memory access path 20. The controller unit 12 generally comprises a number of further integrated components, which are not shown in Fig. 1. For example, these further components may include an integrated memory, a digital signal processor, and so on.
The memory unit 14 of the device 10 comprises an internal memory controller 22 and a programmable memory 24. In the present embodiment, the programmable memory 24 is a Flash memory with a large number of Flash memory cells that are arranged in a plurality of sectors. However, it is apparent that the programmable memory 24 may also be configured in another technology like, for example, as an EEPROM or FRAM. The programmable memory 24 will in many embodiments be a non-volatile memory, but the invention is not limited to non-volatile memories.

The controller unit 12 and the memory unit 14 are connected via an address bus 26 and a data bus 28. More particularly, the address and data busses 26, 28 run between the memory interface 18 of the controller unit 12 and the internal memory controller 22 of the memory unit 14. The address and data busses 26, 28 comprise address and data lines as well as control lines for controlling the communication between the controller unit 12 and the memory unit 14. The internal memory controller 22 decodes memory addresses arriving via the address bus 26 and controls all operations within the memory unit 14. These operations will be described below in detail.

The programmable memory 24 is programmed via a data write path 30, which runs from the internal memory controller 22 to the programmable memory 24. The data write path 30 comprises a decryption unit 32, a decryption key storage 34, and an initialization vector storage 36. The decryption unit 32 receives encrypted data ED from the internal memory controller 22, decrypts the data to obtain plain - i.e., decrypted - data PD, and provides the plain data PD to the programmable memory 24 for storage therein.

The decryption key storage 34 and the initialization vector storage 36 hold a decryption key K and an initialization vector IV that are used in the decryption process. In the presently described embodiment, the storages 34, 36 are implemented as one time programmable (OTP) memories that can only be read out by the decryption unit 32. The decryption key K and the initialization vector IV are programmed into these storages 34, 36 at the time of producing the device 10. It is
apparent that other memory configurations are possible, as long as the decryption key K and the initialization vector IV cannot be changed by an attacker and cannot be read out, except by the decryption unit 32. For example, the storages 34, 36 may be implemented as a section of the programmable memory 24 or as a section of RAM memory that is initialized at the time of starting up the device 10 under control of a ROM based startup routine. In such embodiments, suitable precautions must be taken to ensure that only the ROM based startup routine can write into the storages 34, 36.

The decryption unit 32 performs the decryption process according to any one of a number of cryptographic methods that are, as such, known in the art. In many embodiments, a symmetric block cipher method is used, but the invention is not limited to symmetric methods or to block cipher methods. Examples of suitable methods are the well known AES, IDEA, DES and 3DES methods. These methods are preferably used in a block operation mode like, for example, one of the ECB and CBC block operation modes for DES. It is understood that, for a symmetric method, there is no difference between the encryption and decryption steps. The term "decryption" will nevertheless be used in the present document in order to clarify that the decryption unit 32 takes the encrypted data ED and outputs the plain data PD to the programmable memory 24.

In many embodiments of the present invention the encrypted data ED is decrypted during or in connection with the process of writing the resulting plain data PD into the programmable memory 24. Writing data into a Flash memory or another non-volatile memory is rather slow because of physical constraints. Therefore, if the decryption and the writing operation are performed concurrently, the decryption will in many embodiments not require any additional time, other than the time that is needed in any case for the operation of writing the data into the memory. While in many embodiments the timing of the decryption operation is not critical, there are also embodiments in which a suitable buffer - e.g., a FIFO queue - is provided within the internal memory controller 22 and/or the data write path 30 in order to decouple any timing constraints.
In many embodiments, the invention will be used for updating software stored in the programmable memory 24. This involves a sequential writing operation into consecutive memory cells of the programmable memory 24. The corresponding decryption in the decryption unit 32 incurs little overhead, especially if one of the above-mentioned block modes of operation is used.

Data is read out from the programmable memory 24 via a data read path 38, which connects the programmable memory 24 to the internal memory controller 22. In the present embodiment, the data read path 38 does not contain any cryptographic elements and therefore outputs any data - for example, the plain data PD - as it is stored in the programmable memory 24. Because there are no complex data manipulation steps, the timing of the read operation is only determined by the programmable memory 24. In other words, the present embodiment achieves the desired protection against manipulation without any performance penalty for memory read operations. This is true both for sequential and for random access read operations.

The internal memory controller 22 is further adapted to erase parts of the programmable memory 24 by applying suitable signals to an erase signal line 40. The erase function is used for preparing a data write operation in a way that is customary for Flash memories.

Fig. 1 shows a protected memory field 42 of the programmable memory 24 into which the plain data PD is written. In some embodiments of the invention, only parts of the programmable memory 24 are protected against unauthorized modification while other parts - outside the protected memory field 42 - are writable without any protection. For example, write operations to memory cells outside the protected memory field 42 may bypass the decryption unit 32 within the data write path 30 or may use an additional, direct write path (not shown). However, in other embodiments of the invention, there are no unprotected memory
portions such that the protected memory field 42 covers the entire programmable memory 24.

Some embodiments of the invention may have more than one protected memory field 42 within the programmable memory 24. The size and arrangement of the one or more protected memory fields 42 may be fixed or settable. For example, in some embodiments the memory unit 14 may contain a register (not shown) that determines the regions - e.g., sectors or groups of sectors - of the programmable memory 24 that are to be included into or excluded from the one or more protected memory fields 42. This register may, for example, be formed as a one time programmable (OTP) memory, and it may be programmed at the time of producing the device 10.

The possibility of excluding certain regions of the programmable memory 24 from the protection scheme of the present invention is useful or necessary in embodiments in which, for example, the memory unit 14 contains some kind of Flash file system (FFS) for storing persistent data. In such embodiments, it must be possible for the controller unit 12 and/or the internal memory controller 22 to write administrative data of the file system into the programmable memory 24. However, since the controller unit 12 and/or the internal memory controller 22 in many embodiments cannot access the key K and therefore cannot encrypt the administrative data, there must be a non-protected portion of the programmable memory 24 into which the administrative data can freely be written.

All in all, the one or more protected memory fields 42 of the programmable memory 24 can only be modified in a meaningful way if an encrypted version of the data to be written into the programmable memory 24 is available. Creating this encrypted version requires knowledge of the cryptographic method, the key K and the initialization vector IV. Since a symmetric cipher is used in the presently described embodiment, the same key K and initialization vector IV are used both for the encryption and for the subsequent decryption. In other words, the key K and the
initialization vector IV that are contained in the storages 34, 36 must also be available when generating the encrypted data ED.

There may be embodiments in which the controller unit 12 or another component of the device 10 has access to the key K and the initialization vector IV. In such embodiments, it would be possible to generate suitably encrypted data ED within the device 10. However, such embodiments may offer less than the optimum security if the device 10 is tampered with or if fraudulent software is executed by the device 10. Consequently, in many other embodiments there are no provisions within the device 10 to encrypt data to be written into the programmable memory 24. In such embodiments, the device 10 must receive the encrypted data ED from an external data source 44 like, for example, an external service provider or a mobile network operator or an authorized service center.

Fig. 1 shows an example of the external data source 44 that sends the encrypted data ED to the device 10 via a communication path 46. The communication path 46 may be a wireless communication channel or a wire-bound data transmission link. The external data source 44 comprises an encryption unit 48 implementing a cryptographic method that corresponds to that of the decryption unit 32. The encryption unit 48 receives the plain data PD to be encrypted from a file system 50, and it receives the key K and the initialization vector IV from a database 52.

For example, the database 52 may contain an individual record for each device 10 that has ever been manufactured, each record containing the key K and the initialization vector IV of the device 10, as well as other administrative information. The serial number of the device 10 or other suitable identification data may serve as an index for accessing this information. In other embodiments, the database 52 may contain fewer records like, for example, only one record for each manufacturing batch or even only one record for each type of the device 10.

Having a variety of keys K ensures that software updates are properly matched to the various devices 10 and also increases the overall security of the protection scheme in case one of the keys K is compromised. However, the invention also
includes embodiments in which only a single key \( K \) and/or a single initialization vector \( IV \) are used. In such embodiments, no database 52 is necessary.

The key \( K \) must be kept secret in order to ensure that an unauthorized attacker cannot create a properly encrypted version of some unauthorized data for storage in the protected memory field 42. The presently described embodiment is especially well protected in this respect because the key \( K \) is neither part of nor accessible to any software of the device 10 that could be monitored or analyzed by an attacker. In particular, the decryption key storage 34 is only connected to the decryption unit 32 and cannot be read out by the controller unit 12 or any other entity of the device 10.

In order to maintain the secrecy of the key \( K \), the cipher method used by the decryption unit 32 should have the property that no useful information about the key \( K \) can be obtained even if a number of decryption processes are observed, i.e., pairs of encrypted data \( ED \) and corresponding plain data \( DP \) are known. The cipher methods mentioned above and other known cipher methods are suitable in this respect.

In some embodiments, the decryption unit 32 uses a derived key \( DK \) instead of the key \( K \) that is contained in the decryption key storage 34. The derived key \( DK \) may be obtained from the stored key \( K \) by any method, and further information may be incorporated into the derived key \( DK \) in this process. For example, the derived key \( DK \) may be obtained by applying a cryptographic hash function \( CH \) to both the stored key \( K \) and an address \( ADR \) of the memory write operation as follows:

\[
DK = CH(K, ADR)
\]

It is understood that the same derived key \( DK \) must also be used when preparing the encrypted data \( ED \).
Using a derived key $DK$ like, for example, the one specified above further increases the security of the method against manipulation because a different key will be used for memory write operations to each address. Even if an attacker manages to spy out one derived key $DK$, it will be useless for subsequent write operations to different memory addresses.

In some embodiments, not only the key $K$, but also the initialization vector $IV$ is kept secret. However, this is not strictly necessary from a cryptographic point of view, and consequently there are also embodiments in which the initialization vector $IV$ is not a secret value. For example, in some embodiments a unique serial number of the device 10 is used as the initialization vector $IV$.

Generally, there are a number of possible ways for individualizing the initialization vector $IV$, and any of these ways may be used in various embodiments of the invention. For example, the initialization vector $IV$ may be specific to the individual devices 10 - as in the example given in the previous paragraph - , or it may be specific to the individual users, or it may be specific to the individual write operations. In some embodiments, the initialization vector $IV$ may depend on or be identical to an address of the memory write operation like, for example, the start sector address.

Fig. 2 shows an exemplary flow sequence comprising the steps of writing data into the programmable memory 24 and reading data from the programmable memory 24.

When the data - for example, application software - stored in the programmable memory 24 is to be updated, the external data source 44 executes a step 54 of creating the encrypted data $ED$, using the new plain data $PD$ and suitable values for the decryption key $K$ and initialization vector $IV$, as described above.

In step 56, the encrypted data $ED$ is transferred to the device 10 - more particularly, to its controller unit 12 - via the communication path 46. Fig. 2 shows
an example of a command UPDATE(ADR,ED) that instructs the device 10 to perform an update operation and includes the update start address ADR and the encrypted data ED. In step 58, the controller unit 12 forwards this information in a write command WRITE(ADR,ED) to the memory unit 14 - more particularly, to the internal memory controller 22.

In response to receiving the write command WRITE(ADR,ED), the internal memory controller 22 executes the steps necessary for programming the programmable memory 24. First, one or more sectors of the programmable memory 24 are erased in step 60. Then, in step 62, the encrypted data ED to be programmed is passed through the decryption unit 32 in the data write path 30 such that the actual plain data PD is written into the protected memory field 42. Steps 60 and 62 may be repeated as often as necessary if there is further data to be written into the programmable memory 24; this possibility is shown in Fig. 2 by dashed arrow 64.

After completion of the update process, the protected memory field 42 contains the plain data PD. This data can now be read out in the usual way. For example, a memory read command READ(ADR) may be issued by the controller unit 12 in step 66. The internal memory controller 22 performs a corresponding memory read operation in step 68. In step 70, the programmable memory 24 outputs the plain data PD via the data read path 38 to the internal memory controller 22. Finally, the internal memory controller 22 forwards the plain data PD to the controller unit 12 in step 72. For example, the plain data PD may be application program code that is executed by the CPU 16 of the controller unit 12.

In the embodiment of Fig. 1, all elements of the memory unit 14 are integrated into a single semiconductor chip or a single semiconductor package. In particular, the decryption unit 32, the programmable memory 24 and the lines by which the plain data PD is programmed into the memory 24 are all contained in this semiconductor chip or package. This configuration provides particularly good protection against physical attacks that attempt to circumvent the decryption unit 32 by connecting data write lines directly to the programmable memory 24.
Fig. 3 shows an alternative embodiment in which the decryption unit 32 and the associated elements of the data write path 30 are integrated into the controller unit 12. Furthermore, the memory interface 18 and the internal memory controller 22 of the embodiment of Fig. 1 are integrated into a combined memory controller 74 in the embodiment of Fig. 3. An advantage of the embodiment of Fig. 3 is that the memory unit 14 may be a standard memory device like, for example, a Flash memory chip. Furthermore, the controller unit 12 may be designed such that the CPU 16 and the decryption unit 32 share common elements like, for example, arithmetic or logic processing units. In the embodiment of Fig. 3, care should be taken that an attacker cannot physically connect data write lines directly to the memory unit 14.

All in all, the present invention protects the data stored in the protected memory field 42 of the programmable memory 24 against unauthorized manipulation. If the protected data comprises software that is to be executed on the device 10, then this software can be considered as trusted without any integrity check. It is understood that suitable precautions should be taken against possible attacks that involve physically removing - e.g., unsoldering - or replacing the memory unit 14.

The particulars contained in the above description of sample embodiments should not be construed as limitations of the scope of the invention, but rather as exemplifications of some embodiments thereof. Accordingly, the scope of the invention should be determined not by the embodiments illustrated, but by the appended claims and their legal equivalents.
CLAIMS

1. An apparatus comprising a programmable memory (24), a data write path (30) for writing data into the memory (24) and a data read path (38) for reading data from the memory (24),
   the memory (24) comprising at least one protected memory field (42),
   the data write path (30) comprising a decryption unit (32) that is adapted for receiving encrypted data (ED), decrypting the encrypted data (ED), and writing resulting plain data (PD) into the at least one protected memory field (42),
   the data read path (38) being adapted for reading out the plain data (PD) stored in the protected memory field (42),
   whereby the at least one protected memory field (42) is only writable by applying the data to be written into the at least one protected memory field (42) in encrypted form to the data write path (30).

2. The apparatus of claim 1, wherein the data read path (38) is free of any cryptographic processing elements.

3. The apparatus of claim 1 or claim 2, further comprising a decryption key storage (34) containing a secret key (K).

4. The apparatus of claim 3, wherein the decryption unit (32) is adapted for using, in the decryption process, one of the secret key (K) and a derived key (DK) that incorporates information from the secret key (K).

5. The apparatus of claim 3 or claim 4, wherein the decryption key storage (34) is adapted to be only readable by the decryption unit (32).

6. The apparatus of any one of claims 1 - 5, wherein the decryption unit (32) is adapted for using a symmetric block cipher method for obtaining the plain data (PD) from the encrypted data (ED).
7. The apparatus of claim 6, wherein the decryption unit (32) is adapted for using an individualized initialization vector (IV) in the decryption process.

8. The apparatus of any one of claims 1 - 7, wherein the programmable memory (24) is a Flash memory.

9. The apparatus of any one of claims 1 - 8, wherein the at least one protected memory field (42) is a single protected memory field (42) that takes up the entire programmable memory (24).

10. The apparatus of any one of claims 1 - 9, wherein the apparatus is an integrated semiconductor memory unit (14).

11. An electronic device (10) comprising an apparatus according to any one of claims 1 - 10, the device (10) comprising a controller unit (12) and a memory unit (14).

12. The electronic device (10) of claim 11, wherein the memory unit (14) comprises the programmable memory (24), the data write path (30), the data read path (38), and an internal memory controller (22).

13. The electronic device (10) of claim 11, wherein the controller unit (12) comprises the decryption unit (32), and wherein the memory unit (14) comprises the programmable memory (24).

14. The electronic device (10) of any one of claims 11 - 13, wherein the controller unit (12) comprises a CPU (16), and the at least one protected memory field (42) of the memory unit (14) comprises program code for execution by the CPU (16).
15. The electronic device (10) of any one of claims 11 - 14, wherein the device (10) is one of a mobile communication device, a media playback device, an embedded system, a device for automotive use, an authentication device, and a device for medical use.

16. An external data source (44) comprising an encryption unit (48), the external data source (44) being adapted for providing encrypted data (ED) for processing by the apparatus of any one of claims 1 - 10.

17. A method for accessing a programmable memory (24), the memory (24) comprising at least one protected memory field (42), wherein:
   a write access is performed via a data write path (30) comprising a decryption unit (32), the data write path (30) receiving encrypted data (ED), decrypting the encrypted data (ED), and writing resulting plain data (PD) into the at least one protected memory field (42), and
   a read access is performed via a data read path (38), the data read path (38) reading out the plain data (PD) stored in the protected memory field (42), whereby the at least one protected memory field (42) is only writable by applying the data to be written into the at least one protected memory field (42) in encrypted form to the data write path (30).

18. A method for providing an update to an electronic device (10) according to any one of claims 11 - 14, the method comprising:
   creating (54) encrypted data (ED) on the basis of plain data (PD) to be written into the electronic device (10), and
   providing the encrypted data (ED) to the electronic device (10) for decryption and storage within the electronic device (10).
Fig. 3
### A. CLASSIFICATION OF SUBJECT MATTER

**INV. G06F12/14**

According to International Patent Classification (IPC) or to both national classification and IPC

### B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

**EPO-Internal**

### C. DOCUMENTS CONSIDERED TO BE RELEVANT

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>DE 199 33 263 A1 (SIEMENS AG [DE]) 25 January 2001 (2001-01-25) abstract column 1, line 55 - line 59 column 2, line 33 - column 4, line 59 figure 1</td>
<td>1-18</td>
</tr>
</tbody>
</table>

### D

Further documents are listed in the continuation of Box C

### X

See patent family annex

- Special categories of cited documents
  - A* document defining the general state of the art which is not considered to be of particular relevance
  - E* earlier document but published on or after the international filing date
  - L* document which may throw doubts on priorly claim(s) or which is cited to establish the publication date of another citation or other special reason
  - O* document referring to an oral disclosure, use, exhibition or other means
  - P* document published prior to the international filing date but later than the priority date claimed
  - T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
  - X* document of particular relevance, the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
  - Y* document of particular relevance, the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
  - & member of the same patent family

Date of the actual completion of the international search: 3 August 2007

Date of mailing of the international search report: 22/08/2007

Name and mailing address of the ISA/
European Patent Office, P B 5818 Patentlaan 2 NL-2280 HN Rijswijk
Tel (+31-70) 340-2040, Tx 31 651 epo nl, Fax (+31-70) 340-3016

Authorized officer
Knutsson, Frederic
<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td>DE 19933263 A1</td>
<td>25-01-2001</td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td>US 2003051090 A1</td>
<td>13-03-2003</td>
<td>NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>CN 1229513 A</td>
<td>22-09-1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DE 19781829 C2</td>
<td>22-02-2001</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DE 19781829 TO</td>
<td>17-06-1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GB 2330228 A</td>
<td>14-04-1999</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 200205978 A1</td>
<td>30-05-2002</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 6363463 B1</td>
<td>26-03-2002</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 5778070 A</td>
<td>07-07-1998</td>
</tr>
</tbody>
</table>