Title: DEVICE AND METHOD FOR TRANSFERRING DATA BETWEEN A SOURCE AND A STORAGE MEANS

Abstract: The invention relates to a method and a device for transferring data between a source (15) and at least one storage means (8, 12), the data having to be stored in an encrypted manner on the said storage means (8, 12). The device comprises at least two means (4, 5) of direct memory access transfer, one of which is able to encrypt the data at the moment of their transfer. According to the invention, so as to transfer the data from the said source to the said storage means (8, 12), the direct memory access transfer means (4, 5) transfer each of the data blocks alternately, in such a way that the data recorded on the said storage means (8, 12) are recorded by alternating encrypted data blocks and non-encrypted data blocks.
For two-letter codes and other abbreviations, refer to the “Guidance Notes on Codes and Abbreviations” appearing at the beginning of each regular issue of the PCT Gazette.
DEVICE AND METHOD FOR TRANSFERRING DATA BETWEEN A SOURCE AND A STORAGE MEANS

The invention relates to a device and a method for transferring data between a source and a storage means.

The invention relates more particularly to encrypted data transfers.

The implementation of hard disks in digital decoders leads to content protection problems. Specifically, content diffusers permit the recording of videos but in a limited manner, while wanting to control the number of copies. The proliferation of the means of storage of hard disk, DVD and other types can lead to an illegal proliferation of contents. This is why, increasingly, videos must be encrypted before being recorded.

The context of the invention is the transfer of encrypted data on a data bus, between a source and a destination that may possibly be a storage means.

The encryption of the data and their transfer are expensive in terms of time. This transfer of encrypted data therefore monopolizes the data buses for too long a time.

It is therefore desirable to minimize this transfer time, encrypting the data so as to guarantee the fastest possible transfer times.

For this purpose, the invention proposes a device for transferring data between a source and at least one storage means, the data having to be stored in an encrypted manner on the storage means comprising at least two means of direct memory access transfer, one of which is able to encrypt the data at the moment of their transfer.

According to the invention, so as to transfer the data from the source to the storage means, the direct memory access transfer means transfer each of the data blocks alternately, in such a way that the data recorded on the storage means are recorded by alternating encrypted data blocks and non-encrypted data blocks.

In this way, a single part of the data is encrypted. This suffices to render the encryption sufficiently robust for it not to be possible for the thus
recorded data to be read by other reading means and recopied so as to be read and decrypted subsequently.

Thus, the encrypted data transfer time can be reduced and the rate of occupancy of the transfer bus can also be reduced.

According to a preferred embodiment, the size of the non-encrypted data blocks is determined as a function of the bandwidth available between the source and the storage means.

In this way, it may be possible to encrypt more or less data as a function of the availability of the data transfer buses and therefore to increase or to decrease more or less the relative size of the encrypted and non-encrypted data blocks.

Advantageously, the data being coded in the form of groups of pictures, the size of the data blocks is not aligned with these groups of pictures.

This strengthens the security of the recorded data and limits the risks of piracy since the non-alignment of the structure of the data with their encryption markedly accentuates the difficulty of reconstructing the data.

According to a preferred embodiment, the data being coded in the form of groups of pictures, the size of the data blocks is aligned with these groups of pictures.

According to a preferred embodiment, the size of the encrypted and non-encrypted data blocks is less than the size of a picture.

Advantageously, information relating to the size of the blocks is recorded in the storage means.

According to a particular embodiment, the data of a group of pictures being coded in a dependent manner such that the coding of a first picture is used to code at least one other picture, only the pictures whose coding is independent of the other pictures are transferred by the means of direct transfer able to encrypt the data, the other pictures being transferred by the means of transfer not able to encrypt the data.
Advantageously, the size of the blocks is aligned with the size of the blocks used for the encryption of the data by the direct memory access transfer means able to encrypt the data.

Preferably, the data comprising a recurrent pattern, the size of the encrypted and non-encrypted data blocks is a multiple of the said pattern. This advantageously makes it possible not to record information relating to the size of the blocks in the storage means since the size of the blocks is easily detectable by searching for the pattern in the recorded data. In this way, it is possible readily to retrieve the encrypted and non-encrypted data blocks in the data recorded by aligning oneself with the pattern.

According to another aspect, the invention also relates to a method for transferring data between a source and at least one storage means, the data having to be stored in an encrypted manner on the storage means comprising a step of direct memory access transfer and of simultaneous encryption of the data at the moment of their transfer and a step of direct memory access transfer without simultaneous encryption of the data.

According to the invention, to transfer the data from the source to the storage means, the steps of direct memory access transfer with simultaneous encryption and without simultaneous encryption are alternated, in such a way that the data recorded on the storage means are recorded by alternating encrypted data blocks and non-encrypted data blocks.

The invention will be better understood and illustrated by means of wholly nonlimiting advantageous exemplary embodiments and modes of implementation, with reference to the appended figures in which:

- Figure 1 represents a device according to a preferred embodiment of the invention,
- Figure 2 represents an example of data transfer,
- Figure 3 represents a preferred embodiment of the invention.
In the description which follows, the modules represented are functional units, which may or may not correspond to physically distinguishable units. For example, these modules or some of them may be grouped together in a single component, or constitute functionalities of one and the same software. Conversely, some modules may possibly be composed of separate physical entities.

The preferred embodiment illustrated below is based on a stream in accordance with the H264 or MPEG-2 standard but the invention is not limited to these standards and can apply to other types of data.

Figure 1 represents a device according to a preferred embodiment of the invention.

A "front-end" module 1 is linked to an external network which may be of cable, satellite, terrestrial, IP type. This module in particular demodulates the signals received and transmits a stream of digital data to a data processing module 9.

The module 9 receives the data coming from the front-end 1. This module processes the data received so as to decode them, to display them on a television 16 or to record them on an internal hard disk 8 or on an external hard disk 12.

The module 9 is typically known as a digital television decoder. This decoder is equipped with various functionalities such as the recording of audio video data, the possibility of playing back audio video data according to various special modes, rewind, fast forward, deferred playing of the data received while maintaining simultaneous recording. These various functionalities produce significant data transfers on the various data buses between the hard disk 8 and the processor 2, between a RAM memory 6 and the hard disk 8 and to a host USB interface 3.

The significant occupancy of these data buses can sometimes slow the special reading modes (fast forward or fast rewind, etc.) which require a great deal of bandwidth.

Moreover, to create for oneself a personal library of videos for example, an external storage module 10 is linked to the decoder by way of a USB-type
connection. The host USB interface 3 of the decoder is linked by a cable 11 to the USB interface 13 of the storage module 10. This storage module 10 also comprises a microcontroller 14 and the storage means 12 represented by a hard disk.

This storage module 10 is intended to record the audio video data received by the module 9 and that the user desires to preserve. Of course, the internal hard disk 8 is also able to record audio video data but its capacity is limited and it is therefore advantageous to benefit from an external and tailorable storage unit if possible, so as to increase the storage capacities.

The external storage module 10 can also be of the USB key type.

For content protection and preservation of copyright reasons, the data must be recorded on the hard disk in an encrypted manner so that they cannot be pirated. The data received are sometimes received already encrypted according to the encryption used by the DVB protocol.

The encryption of the data is performed by an encryption module 4 which also fulfils a data transfer function of DMA type (acronym standing for "direct memory access") which allows fast data transfers between a memory and a peripheral or between two memories. The encryption module 4 is a module integrated with the host processor 2. The latter also comprises a DMA module 5 which does not carry out the encryption function.

The various types of applications performed by the decoder are:
- the recording of data on the external hard disk 12 (or internal 8),
- the playback of data on the external hard disk 12 (or internal 8),
- the simultaneous reading and recording of data on the external hard disk 12 (or internal 8),
- the deferred reading of a programme currently being recorded (referred to as timeshift) and the recording of another programme on the other hard disk,
- the archiving of the data on the external disk 12.

The archiving of the data on the external hard disk 12, for example previously recorded on the internal hard disk 8, takes time and can be regarded as a background task which must be rendered transparent to the
user. This task must therefore be carried out in the shortest possible time, to avoid overly long occupancy of the data buses and thus penalize the other transfer times. The invention is therefore particularly advantageous during this archiving application but is not limited to this function.

The DMA 4 which ensures the encryption function has markedly lower performance than the straightforward DMA 5 in terms of data transfer time. The host processor 2, in order to increase the bandwidth in the decoder will therefore programme the two DMAs 4 and 5 in such a way that the data to be encrypted and to be transferred are partially encrypted.

The data written to the destination hard disk 8 or 12 are therefore not all encrypted but they are encrypted blockwise. Thus, the two DMAs are programmed to transfer each of the blocks of a predetermined size. This size depends essentially on the bandwidth available in the decoder. It can also depend on the other applications current at the moment of the transfer/encryption.

The application or the characteristics of the decoder very often require predetermined bit rates to obtain likewise predetermined transfer times. Thus, when the bit rate required to transfer data between the internal hard disk 8 and the external hard disk 12 is 200 Mbit/s, it is possible to easily calculate the share of transfer performed by the DMA 4 and the share of transfer done by the DMA 5 according to the following equations:

\[ V = \frac{\alpha VDMA4 + \beta VDMA5}{\alpha + \beta} \]

With \( V \) the total bit rate on the bus, \( VDMA4 \) the maximum bit rate of the DMA 4 and \( VDMA5 \) the maximum bit rate of the DMA5. \( \alpha \) and \( \beta \) are coefficients for weighting each of the bit rates so as to arrive at the bit rate \( V \).

When the application requires a bit rate \( Vobj \), lower than the bit rate \( V \), we then obtain:

\[ Vobj < \frac{\alpha VDMA4 + \beta VDMA5}{\alpha + \beta} \]

i.e. therefore
\[ \frac{\alpha}{\beta} < \frac{VDMA5 - Vobj}{Vobj - VDMAA} \]

i.e., in the exemplary embodiment, having

\[
V_{obj} = 200 \text{ Mbit/s} \\
V_{DMA4} = 150 \text{ Mbit/s} \\
V_{DMA5} = 300 \text{ Mbit/s}.
\]

We then obtain:

\[
\frac{\alpha}{\beta} < 2
\]

We choose \( \alpha + \beta = 20K \), as a function of the available bandwidth and of the various concurrent applications.

Thus, we obtain

\[
\alpha = 6.666 \\
\beta = 13.333
\]

Thus, the DMA 5 is programmed to transfer blocks of 7 Kbytes and the DMA 4 is programmed to transfer blocks of 13 Kbytes. Thus, the data present in the hard disk are composed alternately of blocks of 7 kbytes of encrypted data and of blocks of data of 13 Kbytes that are non-encrypted.

This encryption rate is sufficient to cause the person who desires to view the archived video not to be able to see correctly and this therefore avoids piracy.

During playback of the data in the hard disk, the data must be identifiable as encrypted or non-encrypted data. For this purpose, metadata are recorded on the hard disk with the useful data in the course of the recording or at the start of recording. These metadata make it possible to identify the sectors in which the data are encrypted and are represented for example by a simple indication of the size of the blocks, i.e. 7k and 13k.

In other embodiments the relative size of the encrypted and non-encrypted blocks can take other values.
The use of DMAs makes it possible to transfer data rapidly on the bus. In order that the use of the DMA is effective, it is important that the data blocks to be transferred are of fairly significant size in order that the reprogramming time for the two DMAs between each transfer remains negligible with respect to the duration of a data block transfer.

The alternating transfer of an encrypted byte then of a non-encrypted byte would ensure very good security at the encryption level but would take too much time and would not solve the bus transfer time problem. It is necessary therefore to make a compromise between the size of the data blocks and the encryption security desired. Moreover, some DMA encryptors also have requirements and sometimes need to encrypt several bytes simultaneously.

Figure 2 represents an exemplary organization of the data during a transfer such as described in Figure 1.

Figure 2 relates to the transfer of data between the internal hard disk 8 and the external hard disk 12. The data are previously recorded on the internal hard disk 8 and when the user desires to transfer them to his external hard disk 12, for long-term backup, the data are encrypted before being backed up.

The data received on the processing module 9 are encoded data in the MPEG-2 or MPEG-4 format and structured in the form of GOPs (the acronym standing for "group of pictures").

The size of a GOP is variable as a function of the complexity of the picture data, as a function of the motion etc. For example, in the case of a bit rate of 4 Mbits/s, the size of a GOP can be 256 Kbytes.

The data are transferred to the external hard disk 12 in blocks of 7 encrypted kbytes and of 13 non-encrypted kbytes.

The GOP 1, whose size is greater than 8 Kbytes, is therefore transferred partially in an encrypted manner and partially in a non-encrypted manner. Thus the size of a GOP does not correspond to the size of a DMA
transfer. This further strengthens the concept of security and complicates data piracy.

In another embodiment (not represented), not only is a GOP not transferred in a totally encrypted manner or totally non-encrypted manner but a picture itself is transferred in a fragmented manner, certain parts being encrypted while others are not. The size of the blocks transferred is then less than the size of a coded picture. Thus, the pictures are all transferred, an encrypted part alternating with a non-encrypted part and so on and so forth according to the size of the picture and of the transferred blocks. This is in particular the case when the bit rate is 256 Kbytes and the size of an I picture is twice the size of a P picture and the size of a P picture is twice the size of a B picture. When the GOP comprises 12 pictures as indicated in Figure 3, we have the following equation:

\[ 256KO = I + 3P + SB = 4B + 6B + SB = 18B \]

The following picture sizes are therefore obtained:

- \( B = U \) Kbytes
- \( P = 28 \) Kbytes
- \( I = 56 \) Kbytes

In this case, when the encrypted data blocks are of size 13 Kbytes and the non-encrypted data blocks are of size 7 Kbytes, the I, P and B pictures are all transferred and recorded partially encrypted and non-encrypted since the size of the blocks is therefore not aligned with the size of the pictures. This makes it impossible to view them clearly.

In other embodiments, to align the size of the GOPs with the size of the pictures, it is for example envisageable to take a block size that is a multiple of 14 Kbytes but in this case, certain pictures are totally non-encrypted. On the other hand, they are coded according to the MPEG-2 standard and in this case must be decoded before being viewed. They do not therefore appear as
plain text even if they are not encrypted since their decoding depends on encrypted pictures.

Figure 3 represents a GOP coded according to the H264 standard. This example illustrates the particular embodiment in which only the I pictures (intra pictures) are encrypted and transferred by the DMA 4.

This particular exemplary embodiment requires that the location of the I pictures in the data stream to be transferred and to be encrypted be known. Moreover, the size of the I pictures not being constant, it depends on the complexity of coding of the picture, the size of the blocks to be transferred then varies from one GOP to another thereby making it harder to programme the two DMAs 4 and 5. It is indeed necessary to know the size of the various pictures so as to be able to programme the DMA, and this may be expensive in terms of time.

In another embodiment (not represented), the GOPs are transferred in an alternating manner by the DMA 4 and the DMA 5. This embodiment also requires that the size of each GOP be known so as to parametrize the DMA transfers.

In order to play back the data which have been transferred as described previously, it is necessary to be able to retrieve in the storage means which blocks are encrypted and which blocks are non-encrypted. Thus, metadata are recorded on the storage means with the stored data with which they are associated.

With each file thus recorded on the hard disk 12, a metadata file is associated. This file comprises the size of the blocks used to transfer the data. It therefore comprises a size of blocks for the encrypted data and a size of blocks for the non-encrypted data. During playback of the data in the storage module 10, the metadata file corresponding to the data file to be read is read before reading the metadata.

In another embodiment, it is also possible not to store the size of the blocks in the external hard disk but to store a code representative of the block size, thereby strengthening the securing of the data. This code is determined
by the host processor 2 which associates with each block size a particular code that it can decode during the playback of the metadata file in the hard disk 12.

When the data are recorded using the TS format (acronym standing for "transport stream"), packet format according to the MPEG-2 standard, then the data comprise every 188 bytes a byte whose value is "47" in hexa. The data therefore comprise a recurrent pattern comprising a byte with the value 47 and 187 subsequent bytes. It is therefore also possible to align oneself with this byte and to retrieve it in the recorded data stream. For this purpose, the size of the encrypted and non-encrypted data blocks must be a multiple of 188 bytes.

Thus, during playback the bytes of value "47" are searched for. When a byte of value "47" is no longer encountered 188 bytes after having encountered the previous one, it is detected that the 188 bytes that will follow are encrypted data and so on and so forth.

In another embodiment, the size of the data blocks varies regularly, so as to complicate the structure of the data recorded in memory and therefore to complicate piracy thereof. A series of blocks of data of size 7 encrypted kbytes and 13 non-encrypted Kbytes is for example transferred then a series of blocks of 5 encrypted Kbytes and 15 non-encrypted Kbytes is transferred thereafter.
1. Device for transferring data between a source (15) and at least one storage means (8, 12), the data having to be stored in an encrypted manner on the said storage means (8, 12), comprising at least two means (4, 5) of direct memory access transfer, one of which is able to encrypt the data at the moment of their transfer, characterized in that, so as to transfer the data from the said source to the said storage means (8, 12), the direct memory access transfer means (4, 5) transfer each of the data blocks alternately, in such a way that the data recorded on the said storage means (8, 12) are recorded by alternating encrypted data blocks and non-encrypted data blocks.

2. Device according to Claim 1 characterized in that the size of the non-encrypted data blocks is determined as a function of the bandwidth available between the source (15) and the storage means (8, 12).

3. Device according to Claim 1 characterized in that the data being coded in the form of groups of pictures (GOP), the size of the data blocks is not aligned with these groups of pictures (GOP).

4. Device according to Claim 1 characterized in that the data being coded in the form of groups of pictures (GOP), the size of the data blocks is aligned with these groups of pictures (GOP).

5. Device according to Claim 1 characterized in that the size of the encrypted and non-encrypted data blocks is less than the size of a picture.

6. Device according to one of Claims 1 to 3 characterized in that information relating to the size of the blocks is recorded in the storage means (8, 12).
7. Device according to Claim 1 characterized in that the data of a group of pictures (GOP) being coded in a dependent manner such that the coding of a first picture is used to code at least one other picture, only the pictures (I) whose coding is independent of the other pictures are transferred by the means (4) of direct transfer able to encrypt the data, the other pictures being transferred by the means (5) of transfer not able to encrypt the data.

8. Device according to Claim 1 characterized in that the information relating to the size of the blocks is recorded in a coded manner in the said storage means (8, 12).

9. Device according to Claim 1 characterized in that, the data comprising a recurrent pattern, the size of the encrypted and non-encrypted data blocks is a multiple of the said pattern.

10. Method for transferring data between a source (15) and at least one storage means (8, 12), the data having to be stored in an encrypted manner on the said storage means (8, 12), comprising a step of direct memory access transfer and of simultaneous encryption of the data at the moment of their transfer and a step of direct memory access transfer without simultaneous encryption of the data, characterized in that, to transfer the data from the said source (15) to the said storage means (8, 12), the steps of direct memory access transfer with simultaneous encryption and without simultaneous encryption are alternated, in such a way that the data recorded on the said storage means (8, 12) are recorded by alternating encrypted data blocks and non-encrypted data blocks.