BESCHWERDEKAMMERN
DES EUROPÄISCHEN
PATENTAMTS

BOARDS OF APPEAL OF
THE EUROPEAN PATENT
OFFICE

CHAMBRES DE RECOURS
DE L'OFFICE EUROPEEN
DES BREVETS

Internal distribution code:
(A) [ ] Publication in OJ
(B) [ ] To Chairmen and Members
(C) [ ] To Chairmen
(D) [X] No distribution

Datasheet for the decision of 29 May 2009

Case Number:	T 1460/06 - 3.5.01
Application Number:	98111553.8
Publication Number:	0887735
IPC:	G06F 12/02, G11C 16/34
Language of the proceedings:	EN

Title of invention:
Memory management method for a flash memory

Applicant:
Sony Corporation

Opponent:
-

Headword:
Memory management/SONY

Relevant legal provisions:
EPC Art. 123(2)

Relevant legal provisions (EPC 1973):
EPC Art. 54(1), 56, 84

Keyword:
"Clarity (no - all requests)"

Decisions cited:
-

Catchword:
-

Case Number: T 1460/06 - 3.5.01

DECISION
of the Technical Board of Appeal 3.5.01
of 29 May 2009

Appellant:

Sony Corporation
7-35, Kitashinagawa 6-chome
Shinagawa-ku
Tokyo (JP)

Representative:

Melzer, Wolfgang
Mitscherlich & Partner
Patent- und Rechtsanwälte
Postfach 33 06 09
80066 München (DE)

Decision under appeal:

Decision of the Examining Division of the European Patent Office posted 28 March 2006 refusing European patent application No. 98111553.8 pursuant to Article 97(1) EPC 1973.

Composition of the Board:

Chairman: S. Steinbrener
Members: S. Wibergh
P. Schmitz

Summary of Facts and Submissions

I. This appeal is against the decision of the examining division to refuse European patent application No. 98111553.8.

II. The following document will be referred to:

D3: US-A-5 485 595.

III. According to the decision appealed, the subject-matter of claim 1 of the then main request and first auxiliary request did not involve an inventive step. The claims according to a second auxiliary request were not admitted under Rule 86(3) EPC 1973.

IV. In the statement setting out the grounds of appeal the appellant requested grant of a patent based on the claims on file.

V. In a communication the Board commented in particular on the clarity of the claims and the possible obviousness of their subject-matter.

VI. By letter dated 30 April 2009 the appellant filed amended claims according to a main and two auxiliary requests.

VII. Oral proceedings were held on 29 May 2009. The appellant submitted a new claim 1 of the main request and requested that the decision under appeal be set aside and a patent be granted on the basis of claim 1 of the main request submitted during the oral proceedings before the Board and claims 2 to 8 as filed by letter dated 30 April 2009 to be accordingly adapted, or on the basis of auxiliary request 1 or 2 filed by letter dated 30 April 2009.

VIII. Claim 1 according to the *main request* reads:

"A memory management method for a memory (22a..22d) having a storage area divided into a plurality of blocks, so that data in each of said blocks are erased at once when the block is initialized, wherein each of said blocks has a redundant area, said memory management method comprising steps of:
connecting said memory to a host computer executing said management,
determining by said host computer all blocks having a flag being set to a state that data should be erased; and
executing (ST3) initialization processing to all such blocks and setting the flag of each said block to a writable state,
characterized by
further determining (ST1) by said host computer whether the erasing process should be executed in a hurry, wherein, if the erasing process should be executed in a hurry, instead of an initialization of all blocks having a flag being set to a state that data should be erased is executed an erasing process of one such blocks */sic/* is executed (ST2)".

IX. Claim 1 according to *auxiliary request 1* reads:

"A memory management method for a memory (22a..22d) having a storage area divided into a plurality of blocks, so that data in each of said blocks are erased

at once when the block is initialized, wherein each of said blocks has a redundant area, said memory management method comprising steps of:
determining a block whose data should be erased; and executing (ST3) initialization processing to said block and setting the flag of the block to an initial state, characterized by
further determining (ST1) whether an other processing than said initialization processing has to be executed, wherein if an other processing than initialization processing has to be executed, instead of an initialization of said block an erasing process is executed (ST2) wherein an erase flag of the block is set to an erase state indicating that data contained in said block are to be erased and the erase operation is terminated temporarily;
and
wherein after said erasing process instead of the initialization of said block is executed (ST2), and if no other processing than the initialization processing has to be executed, the initialization processing is executed to said block (ST3) and the flag of the block is set from the erase state to the initial state".

X. Claim 1 according to *auxiliary request 2* reads:

"A memory management method for a memory (22a..22d) having a storage area divided into a plurality of blocks, so that data in each of said blocks are erased at once when the block is initialized, wherein each of said blocks has a redundant area containing a plurality of flags, said memory management method comprising steps of:

determining a block whose data should be erased and setting a flag to an erase state; and
executing (ST3) initialization processing to said block and setting the flag of said block to an initial state, characterized by
further determining (ST14) whether a writing processing should be executed in a hurry, wherein, if the writing processing is to be executed in a hurry, the erase flags are executed successively and repeatedly until an erase flag in an initial state is found, whereupon writing processing is carried out wherein, if an erase flag is found, the erase flag of a block is set to a further erase state indicating that data contained in said block are to be erased and the erase operation is terminated temporarily".

XI. At the end of the oral proceedings the Board announced its decision.

Reasons for the Decision

Main request

1. The invention

1.1 The invention concerns flash memories. A flash memory block can only be written if it has previously been initialized, meaning that all its data locations have been set to "1" (see A-publication, col. 1, l. 55 to col. 2, l. 13 and col. 6, l. 15-22). The initialization takes a relatively long time. The invention therefore proposes to determine whether the processing should be executed "in a hurry". If no, the block is initialized.

If yes, an "erasing process" is instead performed. The block is then not initialized but merely flagged as being in an "erase state" (ie containing "unnecessary data"). The initialization will be performed at some later time.

1.2 In the present application the word "erasing" sometimes covers initialization and sometimes does not (see eg col. 7, l. 57 to col. 8, l. 5). For clarity, in the present decision the term "erase" will be used only in the sense of setting a flag of a block to the "erase state", reserving the word "initialization" for the setting of all the data of a block to the state "1".

2. The prior art

D3 (col. 2-4) describes how the initialization techniques for flash memories have developed. An early proposal (SunDisk 1991) consisted in always initializing a block immediately before rewriting it with an update of data. The initialization step was however slow. An improvement was to rewrite modified data not to the same memory location but to a different one that was already initialized. Such a location could be found by inspection of the associated flags (col. 5, l. 49-64). The location containing the previous version of the data was flagged as "old" (ie "erased" as the term is used in the present decision). This updating process was quicker than the earlier one since the initialization step was avoided. Only when no more initialized locations were available did a collective initialization of all the "old" memory locations take place.

3. Clarity

3.1 The characterising part of claim 1 reads:

"further determining (ST1) by said host computer whether the erasing process should be executed in a hurry, wherein, if the erasing process should be executed in a hurry, instead of an initialization of all blocks having a flag being set to a state that data should be erased is executed an erasing process of one such blocks */sic/* is executed (ST2)".

As has already been mentioned, in the application the word "erase" sometimes includes the process of initialization and sometimes excludes it. This is true also for claim 1. The feature "... if the erasing process should be executed in a hurry... an erasing process of one */of/* such blocks is executed" would be trivial unless the first instance of "erasing process" were understood as covering both initialization and setting of flags and the second instance as indicating only the setting of flags. But the ambiguity renders the claim obscure.

Furthermore, the antecedent of "such blocks" can only be the immediately preceding word group "all blocks having a flag being set to a state that data should be erased". However, since these blocks have already had their flags set (ie been erased) they cannot be erased again. This is also obscure.

3.2 The characterising part of claim 1 states not only what happens when the erasing process should be executed in a hurry but also what does not happen ("instead of an initialization of all blocks having a flag being set to a state that data should be erased is executed..."). Judging by the description (see col. 7, l. 47 to col. 8, l. 5) the intended meaning is that what does not happen when in a hurry (ie initialization), happens when not in a hurry. But this should have been explicitly stated.

3.3 It follows that claim 1 is not clear and the main request must be refused (Article 84 EPC 1973).

3.4 The Board might add that even if the clarity objections were ignored and claim 1 was interpreted with the aid of fig. 4 and the accompanying description, the invention would not involve an inventive step (Article 56 EPC 1973). The invention's main addition to the art known from D3 is the choice between initialization and erasure. In D3 these techniques are described as different design options, erasure being preferred until initialized blocks are no longer available. A method using the two techniques on an ad hoc basis would only involve an inventive step if it produced a (surprising) synergistic effect. A synergistic effect has however been neither argued nor disclosed. Instead, the method known to be faster is - not surprisingly - used when "in a hurry", and the slower one otherwise. Why the slower option should at all be available seems not to be clearly explained in the description. In any case, if the "in a hurry" state prevailed until all blocks were erased the claimed method would end up with a global initialization step as in D3. The idea of allowing some initialization in any idle states appears to be a straightforward housekeeping measure. The present invention is therefore regarded as an obvious use of two known techniques.

Auxiliary request 1

4. Clarity

According to this request the expression "whether... in a hurry" is replaced by "determining (ST1) whether an other processing than said initialization processing has to be executed". This is obscure since the "other processing" is not defined. The description (col. 8, l. 7-10) suggests that some kind of processing on the blocks is intended ("in a case when a processing other than initialization is to be carried out to predetermined blocks containing a data which has become unnecessary"). But what this processing consists of remains in the dark (unless, of course, erasing is intended, but this interpretation would lead to the claim trivially requiring that a block be erased if it is determined that it has to be erased). Thus claim 1 is not clear and auxiliary request 1 must also be refused (Article 84 EPC 1973).

Auxiliary request 2

5. Clarity

Claim 1 contains the expression "the erase flags are executed successively and repeatedly until an erase flag in an initial state is found, whereupon writing processing is carried out wherein, if an erase flag is found, the erase flag of a block is set to a further erase state". Thus the condition that an erase flag is found occurs twice, each time with different consequences. This is not clear and therefore the request cannot be allowed (Article 84 EPC 1973). Obscure is furthermore the expression "erase flags are executed" since flags (data) cannot be "executed".

6. Support for claim 1 in the description

The characterising part of claim 1 concerns the "writing processing". It includes the step that "the erase flag of a block is set to a further erase state indicating that data contained in said block are to be erased and the erase operation is terminated temporarily". However, the "further erase state" is not mentioned in the description. Moreover, the flow chart in fig. 5 defining the writing process indicates no step of setting flags to the erase state. Thus, these features could not be considered to be originally disclosed (Article 123(2) EPC).

7. Novelty

Notwithstanding the objections above it appears possible to establish that claim 1 contains no initialization step not preceded by erasing. Thus, the claim cannot be understood as containing a choice between initialization and erasing but concerns erasing only. Therefore, as far as the claim can be understood its subject-matter would not be new with respect to D3 (Article 54(1) EPC 1973).

Order

For these reasons it is decided that:

The appeal is dismissed.

The Registrar:	The Chairman:
T. Buschek	S. Steinbrener