

**Internal distribution code:**

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

**Datasheet for the decision  
of 10 December 2025**

**Case Number:** T 0446/24 - 3.5.06

**Application Number:** 17207638.2

**Publication Number:** 3404838

**IPC:** H03M7/30, G06F9/448

**Language of the proceedings:** EN

**Title of invention:**

METHODS FOR ACCELERATING HASH-BASED COMPRESSION AND  
APPARATUSES USING THE SAME

**Applicant:**

Shanghai Zhaoxin Semiconductor Co., Ltd.

**Headword:**

Hash-based compression/Shanghai Zhaoxin

**Relevant legal provisions:**

EPC Art. 56

RPBA 2020 Art. 12(4)

**Keyword:**

Inventive step - (no) - mixture of technical and non-technical  
features

Amendment to case - amendment overcomes objection (no)

**Decisions cited:**

G 0001/19, T 0641/00, T 1539/09

**Catchword:**



**Beschwerdekammern**  
**Boards of Appeal**  
**Chambres de recours**

Boards of Appeal of the  
European Patent Office  
Richard-Reitzner-Allee 8  
85540 Haar  
GERMANY  
Tel. +49 (0)89 2399-0

Case Number: T 0446/24 - 3.5.06

**D E C I S I O N**  
**of Technical Board of Appeal 3.5.06**  
**of 10 December 2025**

**Appellant:** Shanghai Zhaoxin Semiconductor Co., Ltd.  
(Applicant) Room 301, No. 2537, Jinke Road  
Zhangjiang Hi-Tech Park  
201203 Shanghai (CN)

**Representative:** Michalski Hüttermann & Partner  
Patentanwälte mbB  
Kaistraße 16A  
40221 Düsseldorf (DE)

**Decision under appeal:** **Decision of the Examining Division of the  
European Patent Office posted on 24 November  
2023 refusing European patent application No.  
17207638.2 pursuant to Article 97(2) EPC.**

**Composition of the Board:**

**Chairman** M. Müller  
**Members:** T. Alecu  
K. Kerber-Zubrzycka

## Summary of Facts and Submissions

I. The appeal lies from the decision of the Examining Division to refuse the application for lack of inventive step. The decision referred inter alia to documents:

D1: BELL T ET AL: "LONGEST-MATCH STRING SEARCHING FOR ZIV-LEMPERL COMPRESSION", 1993

D3: US 9584155 B1, 28 February 2017

II. The Appellant requested that the decision of the Examining Division be set aside and that a patent be granted on the basis of a main request, which is identical to the main request underlying the decision, or on the basis of one of three auxiliary requests, all filed with the statement of grounds of appeal.

III. The decision was taken in oral proceedings before the Board.

IV. Claim 1 of the main request defines:

*A method for performing hash-based compression, performed in a compression accelerator (20), wherein the compression accelerator comprises an FSM, Finite-State Machine (240), comprising the steps:*

*fetching a string to be compressed from a data buffer (210) by a fetch unit (220);*

*storing a plurality of instances corresponding to the string, in an order of the string, in an intermediary buffer (230) by the fetch unit (220),*

*wherein each instance comprises an associated substring of the string and comprises a state, a match length and a match offset for the associated substring;*

issuing a hash request to a hash matcher (250) for each instance,

issuing a data request to an LSM, longest string matcher, (260) when a first reply sent by the hash matcher (250) comprises a match message, and

updating the state, the match length and the match offset of the instance stored in the intermediary buffer (230) according to a determination of whether a length of a second reply sent by the LSM (260) is shorter than a maximum match length;

outputting a result to a formatter (270) according to the state, the match length and the match offset of each instance in the original order of the associated substrings that appeared in the string, thereby enabling the formatter (270) to compress the string accordingly, and

compressing the string by the formatter (270);

wherein the step of updating the state, the match length and the match offset of the instance according to a determination of whether a length of a second reply sent by the LSM (260) is shorter than a maximum match length comprises:

updating the state of the instance with a partial match state and updating the match length of the instance with the length of the second reply when the length of the second reply is shorter than the maximum match length; and

updating the state of the instance with a full match state, updating the match length of the instance with the maximum match length, and issuing a further data request corresponding to the instance to the LSM (260) to continue the following-string comparison when the length of the second reply is not shorter than the maximum match length

wherein the FSM (240) comprises a hash request procedure, a hash reply procedure, a data reply procedure and a retirement procedure, wherein the hash request procedure issues the hash request to the hash matcher (250) for each instance, wherein the hash reply procedure issues the data request to the LSM (260) according to the first reply sent by the hash matcher (250),

wherein the data reply procedure updates the state, the match length and the match offset of the instance according to the second reply sent by the LSM (260),

wherein the retirement procedure outputs the result to the formatter (270) according to the state, the match length and the match offset of each instance in the original order of the associated substrings that appeared in the string; and

wherein the hash request procedure and the retirement procedure are performed in parallel and have higher priorities than the other procedures.

- V. Claim 1 of the first auxiliary request differs from that of the main request by an amendment of the following feature (addition underlined by the Board):

*issuing a hash request to a hash matcher (250) for each instance to obtain information indicating whether a string with a length of 3 that is associated with the instance matches a hash key of a hash table, wherein the hash table is distributed in four memory banks*

- VI. Claim 1 of the second auxiliary request differs from that of the first auxiliary request by addition of the following feature:

*wherein the steps for issuing a hash request to a hash matcher (250) for each instance and issuing a data*

*request to an LSM, longest string matcher, (260) when a first reply sent by the hash matcher (250) comprises a match message comprises: updating the state of the instance with a no-match state when the first reply comprises a no-match message; and updating the state of the instance with a data request state, updating the match length of the instance with a length of the substring, and updating the match offset of the instance with an offset of the first reply when the first reply comprises the match message.*

- VII. Claim 1 of the third auxiliary request differs from that of the second auxiliary request by addition of the following feature:

*wherein the step of outputting a result to a formatter (270) according to the state, the match length and the match offset of each instance in the original order of the associated substrings that appeared in the string comprises: outputting the result comprising a character of the instance when the state of the instance is a no-match state; and outputting the result comprising the match length and the match offset of the instance when the state of the instance is a partial match state.*

## **Reasons for the Decision**

### *The application*

1. The application relates to methods and apparatuses for accelerating hash-based compression (paragraphs 1 and 2). The type of compression addressed is Lempel-Ziv coding (see paragraph 26), wherein a string of characters is replaced with a pointer to its previous

occurrence in the text, if any (see also D1, second page, first full paragraph).

- 1.1 The most time consuming step in these methods is that of finding the longest matching string (see D1, last phrase of the Introduction, bridging pages 758 and 749, and the application, paragraph 2), because in principle substrings of all possible lengths from the incoming ("lookahead") string need to be matched against the preceding text.
- 1.2 One known way of accelerating this process is hash-based compression, in which a hash function indexes strings of some predefined length. When a matching string in the hash table is found, its position is stored at the corresponding index in a hash table. Any incoming substring is first matched against the entries in that hash table, and when - but only when - a match is found corresponding to the predefined length, a longest matching string is sought using other means (application, paragraph 4, see also D1 section "Hash table" on pages 762-763).
2. The application proposes an accelerator for hash-based compression which relies on parallelism and comprises a Finite State Machine (FSM).
- 2.1 "Instances" corresponding to the incoming string are stored in a buffer memory. More precisely, each instance corresponds to a starting position in the incoming string, comprises the symbol at that position, a "state" variable, and match results (see Table 1 on page 7). The state indicates the state of the matching process for substrings starting at that position, i.e. whether or not a hash request was sent to a hash matcher, whether a request for a longest string match

"by other means" was sent, and whether a match was found (paragraphs 23 to 25).

- 2.2 The FSM contains procedures for the different steps: a hash request procedure (for hash matching), a hash reply procedure, a data reply procedure (longest string match result), and a retirement procedure (see e.g. paragraph 23).
- 2.3 The retirement procedure (see paragraph 26) advances through the instances in order. If the state of an instance indicates an end of processing (e.g. no match in the hash table or other LSM result), the instance is "retired" and the result is sent to a formatter which performs the coding accordingly.
- 2.4 Multiple hash requests are executed in parallel (paragraphs 5, 32, 32); the application further states that the request and retirement procedures are also performed in parallel, and with higher priorities than any other operations (paragraph 23).

*Main request*

*The decision*

3. The Examining Division refused the main request for a lack of inventive step in view of D3. This document teaches a hash-based data compression method comprising inter alia the steps of determining a hash match in a hash table, searching for a longest string by other means if an indexed match is found, and providing the results to an encoder (see figures 4, 5A and 5B, columns 17 to 21).

4. The differences between the claimed invention and D3 are, according to the Examining Division and the Appellant (see the decision, page 7, bottom, and the statement of grounds of appeal, page 2), those related to the use of an FSM and the parallelization of the procedures, as recited in the following features (numbering by the Board):

*(1) the compression accelerator comprises a Finite State Machine,*

*(2) the FSM (240) comprises a hash request procedure, a hash reply procedure, a data reply procedure and a retirement procedure,*

*wherein the hash request procedure issues the hash request to the hash matcher (250) for each instance,*

*wherein the hash reply procedure issues the data request to the LSM (260) according to the first reply sent by the hash matcher (250),*

*wherein the data reply procedure updates the state, the match length and the match offset of the instance according to the second reply sent by the LSM (260),*

*wherein the retirement procedure outputs the result to the formatter (270) according to the state, the match length and the match offset of each instance in the original order of the associated substrings that appeared in the string; and*

*(3) wherein the hash request procedure and the retirement procedure are performed in parallel and have higher priorities than the other procedures.*

5. The Appellant had argued before the Examining Division that the technical effect of these differences was an

acceleration of the hash-based compression because of the parallelisation implemented by the FSM.

- 5.1 The Examining Division considered (decision, page 8) that an FSM was notoriously an abstract model, which in the claim was defined to implement procedures that were otherwise disclosed in D3 using different terminology. The claim defined that the FSM implemented desiderata of parallelisation and prioritisation of operations, but this effect was not supported by the claim wording, which did not define the technical features needed to achieve it. So the use of an FSM as claimed could not *"be translated into any technical effect"*.

*The Appellant's arguments*

6. The Appellant stressed that the contentious issue was inventive step and that the claim only needed to recite the features that distinguished it from the prior art in a non-obvious manner.
- 6.1 The technical effect associated with those features, here acceleration of hash-based compression, was derivable from the specification (paragraphs 20 ff.), which provided sufficient details for the skilled person to implement the invention and achieve it at least in a plausible manner (see e.g. statement of grounds of appeal, page 4, first full paragraph).
7. In the context of the specification, the person skilled in the art would understand that the claim defined hardware features for parallelisation.
- 7.1 The claim defined a compression accelerator comprising an FSM executing four distinct procedures, of which two were executed in parallel and with higher priorities.

This was not an abstract concept, but a concrete structural arrangement determining how multiple hardware components operated concurrently.

7.2 Storing "a plurality of instances corresponding to the string, in an order of the string, in an intermediary buffer" implied that multiple instances were stored and processed simultaneously, as the specification detailed further. The last feature of the claim (difference (3)) explicitly stated that the hash request and the hash retirement procedure were executed in parallel. This implied that at least two instances were processed in parallel. This provided the technical effect of accelerating hash-based compression (see above point 6.1).

7.3 For the invention to be found obvious, it had to be shown that the skilled person *would* - and not only *could* - have modified D3 so as to arrive at the claimed invention. There was no prompt in D3 leading the person skilled in the art towards the claimed features, in particular to the specific parallelisation defined in the last feature.

#### *The Board's opinion*

#### *Differences*

8. The Board agrees with the differences as identified above, but notes the following.

8.1 In difference (1) the compression accelerator is not further specified, other than by the fact that it comprises an FSM. D3 discloses a data compressor which, as the Examining Division stated (see decision top of page 5), aims to accelerate the string matching

procedure, and therefore constitutes a compression accelerator.

- 8.2 The various procedures in (2) are implied in D3, as the Examining Division also stated in its decision, by the different steps of the program depicted in figures 4, 5A and 5B.
- 8.3 The differences (1) and (2) taken together amount therefore to the formalisation of these procedures in terms of an FSM and the corresponding state variable(s) (which may be said to correspond with the state of the program execution in D3 in figure 4, or figures 5A and 5B) and to the implementation of the FSM in hardware.

*Inventive step and obviousness*

9. The Board notes first that features which are disclosed in the application but are not claimed cannot be taken into account in favour of inventive step of the claimed invention. This holds even if, as the appellant states, these features (further) contribute to solving the same technical problem that the claimed invention solves.
10. Regarding differences (1) and (2), the formal definition of an FSM and its corresponding procedures cannot support the presence of a technical effect as an FSM is tantamount to a computer program and the provision of an FSM therefore a mere matter of computer programming which cannot support the presence of an inventive step (see T 641/00, headnote I, and T 1539/09, headnote). The mere fact that the FSM is implemented in hardware cannot establish an inventive step either, as implementing algorithms in hardware is commonplace.

- 10.1 A specific implementation might be inventive, but neither the claim nor the description provide any specific details. In particular, the "concrete structural arrangement" to which the Appellant refers is not defined in the claim.
- 10.2 A technical effect might also be acknowledged if the use of the FSM could be considered to constitute a "specific adaptation of the computer" within the meaning of G 1/19 (reasons 85), as the Appellant correctly noted (see statement of grounds of appeal, page 3, last paragraph).
11. In this respect the Appellant referred to two features, namely the storage of a plurality of instances in a buffer, and the parallelisation and prioritisation of the hash request and retirement procedures (difference (3)), and argued that an acceleration effect was achieved by the parallel processing of instances.
12. The Board agrees only in part.
  - 12.1 First, while the simultaneous storage of several instances is a prerequisite for their parallel processing, the former most certainly does not imply the latter. For example, processing queues, i.e. buffers storing several similar data items simultaneously, are notoriously known structures for sequential processing.
  - 12.2 Furthermore, the Appellant chose to claim parallelism only for two procedures (hash request and retirement), and not, for instance, for the hash or LSM requests for multiple instances or for other procedures.
  - 12.3 Therefore, the only parallelism that can be taken into account in the assessment of inventive step is the one

specifically claimed, namely between the hash request procedure and the retirement procedure.

13. The acceleration effect caused by this specific parallelisation is unrelated to the fact that an FSM is used for the implementation. Either way, the same procedures are carried out. The use of an FSM remains therefore an algorithmic choice with no associated technical effect, which is therefore not to be considered for obviousness.

14. Thus the only differentiating feature that may be said to be contributing towards solving a technical problem considering the context of the other claimed features, and therefore be pertinent for inventive step, is the last claimed feature, which the Board repeats here for convenience:

*wherein the hash request procedure and the retirement procedure are performed in parallel and have higher priorities than the other procedures.*

15. As established above, D3 discloses a method for hash-based compression performing all the claimed steps, albeit sequentially. A problem that the person skilled in the art is always expected to address is how to speed up the execution of a given algorithm.

15.1 The Board considers that the skilled person would always consider parallelism as a means to achieve that purpose. The Appellant did not appear to challenge this specific point.

15.2 The difficult problem for the person skilled in the art is to determine how (and whether) a given algorithm can be parallelised. Typical approaches include data-

parallel execution of one subroutine or parallel execution of different sub-routines.

- 15.3 A straightforward choice of sub-routines that are suitable candidates for parallelisation are sub-routines which do not interact with each other. The skilled person would realise that this is the case for the initial matching stage (i.e. hash request, see D3, e.g. figure 4, step 405) and the output (i.e. retirement, see D3, e.g. figure 4, step 430). Indeed, once the matching is completed for an instance, the result can be output and, at the same time, the matching for a new instance can begin.
- 15.4 Also prioritisation is a known measure for program execution which has been used in the art for several reasons, for instance increased throughput or responsiveness of a given computation. Per se, the use of "prioritisation" is a routine measure for the person skilled in the art, and the claim lacks any detail regarding the intended prioritisation.
16. The Board concludes that claim 1 of the main request lacks inventive step starting from D3 in view of the common general knowledge in the art.

*Auxiliary requests*

17. The first, second and third auxiliary requests amend, in identical way, the main, the first auxiliary and the second auxiliary request underlying the decision.
18. The amendments introduce features which aim at further delimiting the claimed invention from D3. They specify, on the one hand, hash tables for substrings of length

three and, on the other hand, four memory banks (see the statement of grounds of appeal, pages 5 to 8).

19. It is not clear to the Board why the Appellant did not file these amendments before the Examining Division. The Appellant states that they are a "*direct response*" (letter of 6 November 2025, II.1) to that decision, but does not provide any further explanation.
20. More importantly, the amendments do not appear to address the above inventive-step objection to the main request, as they do not establish any parallelism beyond the specifically claimed one between the hash request and retirement procedures, and appear unsuitable to establish an inventive step for other reasons as well.
  - 20.1 Indeed, the use of three-byte substrings is known from D3 (see column 7, lines 5 to 25) and the use of four memory banks appears to be an arbitrary design choice when considering the lack of specification of the said parallelism (see above point 12).
  - 20.2 The features which distinguish the auxiliary requests from the main request (and which were already present in the auxiliary requests underlying the decision) appear to define, as the Examining Division stated (see the corresponding sections on pages 10 and 11 of the decision) conventional processing in hash-based Lempel-Ziv compression and are implicitly disclosed in D3.
21. Therefore, the Board decides to not admit these requests (Article 12(4) RPBA).

**Order**

**For these reasons it is decided that:**

The appeal is dismissed.

The Registrar:

The Chairman:



L. Stridde

Martin Müller

Decision electronically authenticated