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

Datasheet for the decision of 9 January 2013

Case Number:	T 1003/10 - 3.5.02
Application Number:	02797323.9
Publication Number:	1459450
IPC:	H03M 13/29
Language of the proceedings:	EN

Title of invention:
Method and apparatus for coding bits of data in parallel

Applicant:
Qualcomm, Incorporated

Opponent:
-

Headword:
-

Relevant legal provisions:
-

Keyword:
"Sufficiency of disclosure - (yes)"
"Inventive step - after amendment (yes)"

Decisions cited:
-

Catchword:
-

This datasheet is not part of the Decision.
It can be changed at any time and without notice.

Europäisches Patentamt | European Patent Office | Office européen des brevets

Beschwerdekammern | Boards of Appeal | Chambres de recours

Case Number: T 1003/10 - 3.5.02

DECISION
of the Technical Board of Appeal 3.5.02
of 9 January 2013

Appellant:
(Applicant)

Qualcomm, Incorporated
5775 Morehouse Drive
San Diego, CA 92121 (US)

Representative:

Heselberger, Johannes
Bardehle Pagenberg Partnerschaft
Patentanwälte, Rechtsanwälte
Postfach 86 06 20
D-81633 München (DE)

Decision under appeal:

Decision of the Examining Division of the European Patent Office posted 23 October 2009 refusing European patent application No. 02797323.9 pursuant to Article 97(2) EPC.

Composition of the Board:

Chairman: M. Ruggiu
Members: M. Léouffre
R. Moufang

Summary of Facts and Submissions

I. The applicant lodged an appeal against the decision of the examining division, posted 23 October 2009, to refuse the application. The statement setting out the grounds of appeal was received on 2 March 2010.

II. The examining division held that claims 1 and 14 then on file were not supported by the description, contrary to Article 84 EPC.

The following documents of the state of the art were cited during the procedure before the first instance
D1: EP 1 085 660 A; and
D2: Benedetto & al. "A Soft-Input Soft-Output Maximum A Posteriori (MAP) Module to Decode Parallel and Serial Concatenated Codes" TDA PROGRESS REPORT, vol. 42, no. 127, 15 November 1996, pages 1 to 20

III. In an annex to a communication dated 20 September 2012 summoning the appellant to oral proceedings the board referred to a further document
D3 = WO 00/13323 A1
and expressed a preliminary opinion regarding Articles 84, 54, 56 and 83 EPC.

IV. With a letter dated 21 November 2012, the appellant filed three new requests replacing the previous request and taking account of document D3 as the closest prior art.

V. Oral proceedings before the board took place on 9 January 2013. The appellant requested that the decision under appeal be set aside and that a patent be granted in the following version:

- description pages 1, 2, 2a and 3 to 42 filed at the oral proceedings;
- claims 1 to 16 of new main request filed at the oral proceedings;
- drawing sheets 1/19 and 2/19, 8/19 to 12/19 and 14/19 to 19/19 of the published application and drawing sheets 3/19 to 7/19 and 13/19 as filed with the letter dated 21 November 2012.

VI. Claim 1 of the new main request reads as follows:

"A method of encoding data with an encoder (1200) of a wireless communication system, wherein the encoder is clocked, the method comprising:

receiving a plurality of input information bits I[0]:I[3] at the encoder wherein the method processes said plurality of input information bits, **I**, and performs the following steps during a single clock cycle:

- calculating state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ for the encoder, based on the plurality of input information bits I[0]:I[3] and initial or stored state values $S_0[0]$, $S_1[0]$ and $S_2[0]$; and
- generating a set of encoded output values X[0]:X[3], $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$ using the calculated state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ and the plurality of input information bits I[0]:I[3] and the initial or stored state values $S_0[0]$ and $S_1[0]$ by recursively applying:

  $X[n] = I[n]$;

  $Y_0[n] = I[n] \oplus S1[n] \oplus S0[n]$; and

  $Y_1[n] = I[n] \oplus S0[n]$;

  wherein I[n] represents a n-th bit element of said plurality of input information bits I[0]:I[3], X[n] represents an output of the encoder equal to n-th input information bit;

$S_0[n]$ and $S_1[n]$ represent n-th bit elements of a first and a second state value of the state values $S_0[0]:S_0[3]$, $S_1[0]:S_1[3]$,

$Y_0[n]$ represents the n-th first parity bit element of said set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$, $Y_1[n]$ represents the n-th second parity bit element of said set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$,

$\oplus$ represents digital logic XOR operation; and n represents an iteration index of the recursive operation,

wherein said step of calculating said state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ uses:

$S_0[n+1] = I[n] \oplus S_1[n] \oplus S_2[n]$;

$S_1[n+1] = S_0[n]$; and

$S_{\mathbf{2}}[n+1] = S_1[n]$,

wherein $S_2[n]$ represents a n-th bit element of a third state value $S_2[0]:S_2[3]$."

Claims 2 to 8 are dependent on claim 1.

Claim 9 of the new main request reads as follows:

"An encoder apparatus (1200) of a wireless communication system, wherein the encoder apparatus is clocked, comprising:

- means for receiving a plurality of input information bits $I[0]:I[3]$, wherein the encoder apparatus is adapted to process said plurality of input information bits, **I**;
- means for calculating state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ for the encoder, based on the plurality of input information bits $I[0]:I[3]$

and initial or stored values $S_0[0]$, $S_1[0]$ and $S_2[0]$; and

- means for generating a set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$ using the calculated state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ and the plurality of input information bits $I[0]:I[3]$ and the initial or stored state values S0[0] and S1[0] by recursively applying:

  $X[n] = I[n]$;

  $Y_0[n] = I[n] \oplus S1[n] \oplus S0[n]$; and

  $Y_1[n] = I[n] \oplus S0[n]$;

  wherein $I[n]$ represents a n-th bit element of said plurality of input information bits $I[0]:I[3]$, $X[n]$ represents an output of the encoder equal to n-th input information bit;

  $S_0[n]$ and $S_1[n]$ represent n-th bit elements of a first and a second state value of said state values $S_0[0]:S_0[3]$, $S_1[0]:S_1[3]$, $S_2[0]:S_2[3]$,

  $Y_0[n]$ represents the n-th first parity bit element of said set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$, $Y_1[n]$ represents the n-th second parity bit element of said set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$, $\oplus$ represents digital logic XOR operation, and n represents an iteration index of the recursive operation,

wherein said state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$) are calculated using:

$S_0[n+1] = I[n] \oplus S_1[n] \oplus S_2[n]$;

$S_1[n+1] = S_0[n]$; and

$S_2[n+1] = S_1[n]$,

wherein $S_2[n]$ represents a n-th bit element of a third state value $S_2[0]:S_2[3]$, wherein the means for calculating a set of state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ and the means for generating a set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$ are enabled during a single clock cycle."

Claims 10 to 16 are dependent on claim 9.

VII. The appellant essentially argued as follows:

(a) Claim 1 defined a method of encoding data wherein a number of steps were performed during a single clock cycle and wherein formulas defining the intermediate steps of the encoder, i.e. the state values, were applied iteratively during a single clock cycle. This iteration was referred to by the term "recursively". The state values $S_0[0]:S_0[3]$, $S_1[0]:S_1[3]$, $S_2[0]:S_2[3]$ were vectors S_0, S_1, S_2 and the index "n" a vector index. The formulas defining the state values were applied index per index to each vector component "recursively" during a single clock cycle.

Section [1148] of the published application recited: "The parallel encoder receives the 4 input bits and generates three outputs, X, Y_0, Y_1. Each of the outputs is a 4 bit output, wherein the parallel encoder uses recursive processing to produce 4 bit outputs for each cycle". This section provided support for the above interpretation of the claim. The "recursive operation" was further mentioned in sections [1150] and [1157] in connection with figure 14.

The invention could be better understood in the light of figure 14 describing in details one of the encoders referenced 1132 and 1144 in figure 13. Figure 14 disclosed a "look ahead state generator" 1202 performing an iterative process wherein the formulas enabling the calculation of the state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ were applied recursively. The thus calculated state values and initial state values $S_0[0]:S_0[3]$ were then provided to multi-bit output generators 1206 and 1208 (cf. section [1150], lines 8 to 10) issuing each a four bits output at each clock cycle. The whole calculating and generating process happened between two edges of the clock, i.e. during a single clock cycle.

(b) D3 (figure 1) disclosed a turbo encoder comprising two recursive serial encoders performing the same encoding functions as the encoder of the invention. No parallel implementation of the encoders was disclosed therein. D2 did not disclose an encoder applying the same encoding functions.

D1 proposed a parallel implementation of a conventional encoder but the four bits encoder given as an example and referred to in paragraph 3.2 (cf. page 15) was a rate 1 encoder with a latency of two clock cycles and with an output at each cycle based on state values calculated during previous cycles.

The subject-matter of the invention wherein, during a single clock cycle, state values were calculated to generate a set of encoded output values was therefore new.

(c) Starting from D3 and considering the objective problem of increasing the encoding speed, a person skilled in the art would have increased the frequency of the clock of D3 to reduce the latency of the encoder. He would not have modified the architecture of the encoder because there was no hint to modify it. Even if he had been aware of D1 and would have considered applying the teaching of D1 to the encoder of D3, he would not have arrived at the method according to the invention, because the encoder output values of D1 were not calculated during a single clock cycle and were not based on state values calculated during the same single clock cycle. The invention was therefore not obvious having regard to the combination of D3 and D1.

Reasons for the Decision

1. The appeal is admissible.

2. *Amendments (Article 123(2) EPC)*

A basis for independent claim 9 may be found in original claims 23 to 25 which were directed to the embodiment described in sections [1148] to [1157] of the published application and in figures 13 and 14. This embodiment discloses a parallel implementation of the serial encoder shown in figure 11.
The expression "look ahead generator" mentioned in original claim 23, which had no general accepted meaning, has been changed to "means for calculating state values". This "means for calculating state values" constitutes the essential part of the invention rather than the structure of the output generators 1206 and 1208. The "two output generators" mentioned in original claim 23 were therefore renamed "means for generating a set of encoded output values".
The original features of claims 24 and 25 aimed at specifying the functions of the encoder shown in figure

11. These functions and the different output state values have been clarified using an iteration index "n" as stated in section [1157] of the published application. The parity bits have been clarified accordingly.
An essential feature of the invention is that the state values are recursively calculated during a single clock cycle (cf. in particular last sentence of section [1154] and section [1155]). This aspect is emphasized by the expression "n represents an iteration index of the recursive operation".

Finally the feature defining the possibility of having stored state values instead of initial values is supported by section [1150] where it is mentioned that the state values $S_0[4]:S_2[4]$ used for the next encoding cycle are stored.

Method claim 1 corresponding to apparatus claim 9 has been amended accordingly.

The description of the application has been amended to be consistent with the claims and acknowledges the background art disclosed in document D1, D2 and D3.

Thus, the amendments to the application do not contravene Article 123(2) EPC.

3. *Clarity (Article 84 EPC)*

The method of encoding data comprises generating the set of encoded output values by recursively applying equations

$$X[n] = I[n];$$

$Y_0[n] = I[n] \oplus S1[n] \oplus S0[n]$; and

$Y_1[n] = I[n] \oplus S0[n]$.

The term "recursively" should not be interpreted as implying calculation steps performed over a plurality of clock cycles. With the added feature that "n represents an iteration index of the recursive operation", it is apparent that the term "recursively" relates to the definition of a chain of logical operations in accordance with the above equations. Obtaining the result of this chain of operations does not necessarily require a clock. The term "recursively" and the expression "during a single clock cycle" are therefore not in contradiction.

4. *Sufficiency of disclosure (Article 83 EPC) and one way of carrying out the invention (Rule 42(1)e) EPC)*

4.1 The invention aims at reducing the processing delays of serial encoders (cf. sections [1006] and [1038] of the published application). The solution proposes to transform the serial architecture of a constituent encoder in an architecture receiving and coding multiple bits in parallel (cf. section [1039] of the published application). The invention applies to the constituent encoders 1502, 1552 of figure 11. No detailed embodiment of a parallel architecture for an encoder as shown in figure 11 is disclosed.

4.2 However, an example of a parallel architecture for the serial encoder shown in figure 3 of the application and corresponding to "U.S. Patent Application Serial No. 08/963,386" is given in figure 4 (cf. section [1065]: "The encoder architecture shown in FIG.4 can be used to implement, for example, outer convolutional

encoder 310 or inner convolutional encoder 340 in FIG.3").
A specific design of the parallel encoder of figure 4 is shown in figures 5A and 5B (cf. last line of section [1069] and sections [1070] to [1075]).

In this specific embodiment each of the eight parity bits $Y^b{}_0$:$Y^b{}_7$ results from a digital logic combination of the encoder initial state values (the initial state values of variables x_1:x_4) with some of the eight input information bits u_0 to u_7 received in parallel. The digital logic combinations are given in the right hand column of table 2 (cf. section [1059]). These digital logic combinations are designed taking account of the intermediate state values of variables x_1:x_4, given in rows 2 to 9 of table 2. These intermediate state values, which are necessary to calculate the parity bits, are however not provided in and by the circuit of figure 5A. Only the final intermediate state values of variables x_1:x_4 listed in the last row of table 2 (row 10) are calculated by the encoder state machine 510 and stored in registers 514A to 514D shown in figure 5A (cf. page 15, sections [1070], [1071]). The calculation of the final state values and the generation of the encoded output values, namely the parity bits, are performed in a single clock cycle.

4.3 According to the invention, the calculations of all the intermediate state values S_0[1]:S_0[3], S_1[1]:S_1[3], S_2[1]:S_2[3] of the encoder shown in figure 11 (which correspond to the variables x_1 to x_4 for the encoder of figure 3) are executed during the same single clock cycle.
If the invention were to be applied to an encoder as

shown in figure 3, it would mean that all the intermediate state values of variables x_1 to x_4 listed on rows 3 to 10 of table 2 would be provided by the circuit of figure 5A. This would happen in a look ahead state generator similar to the look ahead generator 1202 of figure 14 while the values of the last column of table 2 would be generated as a function of the calculated intermediate state values instead of the initial values, and provided by a multi-bit output generator similar to the generators 1206 and 1208 of figure 14.

Digital logic combinations are not necessarily clocked. Therefore the results of the digital logic combinations may be available during the same single clock cycle. Thus, the board considers that there is no undue burden for a person of ordinary skill to design a digital logic circuit providing all the intermediate state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ of the encoder shown in figure 11 during a single clock cycle. The conditions set out in Article 83 EPC are therefore considered as fulfilled and there appears to be no need for a more detailed example of a way of carrying out the invention than what is given in figure 14 (Rule 42(1)e) EPC).

5. *Novelty (Article 54 EPC)*

5.1 D3 discloses a method (cf. title of D3) of encoding data with a (constituent) encoder 10 of a turbo encoder for CDMA communication channels (cf. D3, figure 1 and page 1, lines 8 to 25).
The encoder 10 receives a plurality of input information bits and calculates a first set of state

values S_0, S_1 and S_2 (shift registers 18, 21 and 22), based on the plurality of information input bits. It generates a first set of encoder output values X(t), Y_0(t) and Y_1(t) using the first set of state values and the plurality of input information bits.

From figure 1, it is unambiguously derivable that, given an iteration index "n", the following equations are valid.

X(t) = I[n] (input information bits or tail bits),

$Y_0[n] = (I[n]\oplus(S_1[n]\oplus S_2[n]))\oplus S_0[n]\oplus S_2[n]$;

$Y_1[n]= (I[n]\oplus S_1[n]\oplus S_2[n]))\oplus S_0[n]\oplus S_1[n]\oplus S_2[n]$;

$S_0[n+1] = I[n]\oplus S_1[n]\oplus S_2[n]$

$S_1[n+1] = S_0[n]$;

$S_2[n+1] = S_1[n]$.

As in the present application, the first two equations defining the parity bits are equivalent to

$Y_0[n]= I[n]\oplus S_0[n]\oplus S_1[n]$;

$Y_1[n]= I[n]\oplus S_0[n]$.

wherein $\oplus$ represents digital logic XOR operation.

While D3 calculates two parity bits Y_0 and Y_1 at each clocked iteration, claim 1, respectively claim 9, differs therefrom at least in that, "during a single clock cycle", the state values $S_0[1]:S_0[3]$, $S_1[1]:S_1[3]$, $S_2[1]:S_2[3]$ are calculated based on the plurality of input information bits and the initial or stored state values and the set of encoded output values X[0]:X[3], $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$ are generated using the calculated state values and the plurality of input information bits and the initial or stored state values.

5.2 D1 generates, "during a single clock cycle", a set of encoded output values using a plurality of input bits provided in parallel (see in particular sections [0048] and [0051] of D1) and initial state values of the encoder. However none of the encoders shown in D1 implements the claimed encoding functions.

The subject-matter of claims 1 and 9 is therefore new.

6. *Inventive step (Article 56 EPC)*

D3 discloses in figure 1 a constituent encoder apparatus 10 which is identical with the constituent encoder 1502 of figure 11 of the application. D3 is therefore considered as the closest prior art.
Starting from D3, a person of ordinary skill facing the need to increase the encoding speed would consider document D1, which solves the same problem (cf. sections [0010] and [0014]). D1 teaches how to convert the serial architecture of a constituent encoder in an architecture receiving and coding multiple bits in parallel.

The serial/parallel conversion disclosed in D1 is based on the same approach as the serial/parallel conversion disclosed in the present application in relation with figures 3, 4, 5A and 5B (cf. item 4.3 above and section [0054] of D1). The encoded output bits are function of the initial state values of the encoder and the input bits which are received in parallel and processed in one single clock cycle (cf. section [0052]). No intermediate state values between the initial state values and the final state values are calculated and provided by the circuits shown in D1.
For example, the four bit parallel turbo coding block shown in figure 6 of D1 provides four information bits

I0(p), I1(p), I2(p) and I3(p) in parallel and generates a set of four encoded output values Q0(p) to Q3(p) as parity bits which are stored in registers Q0 to Q3. The parity bits are function of the input information bits I0(p), I1(p), I2(p) and I3(p) and of initial state values stored in registers X_1 to X_3. The output values are calculated as follows (cf. page 15 in paragraph 3.2):

$Q0(p) = I0(p-1) \oplus I1(p-1) \oplus I2(p-1) \oplus I3(p-1) \oplus x_3(p-1)$

$Q1(p) = I1(p-1) \oplus I2(p-1) \oplus I3(p-1) \oplus x_1(p-1) \oplus x_3(p-1)$

$Q2(p) = I2(p-1) \oplus I3(p-1) \oplus x_1(p-1) \oplus x_2(p-1) \oplus x_3(p-1)$

$Q3(p) = I3(p-1) \oplus x_1(p-1) \oplus x_2(p-1)$

The state values stored in registers X_1 to X_3 are expressed as follows (cf. paragraph 3.2 at page 15):

$x_1(p) = I0(p-1) \oplus I2(p-1) \oplus I3(p-1) \oplus x_1(p-1) \oplus x_3(p-1)$

$x_2(p) = I1(p-1) \oplus I3(p-1) \oplus x_1(p-1) \oplus x_2(p-1) \oplus x_3(p-1)$

$x_3(p) = I2(p-1) \oplus x_1(p-1) \oplus x_2(p-1)$

The index p in D1 corresponds to the clock cycle index of the parallel encoder (cf. D1, section [0048]).
The registers Q0 to Q3 introduce a delay of one clock cycle between the calculated parity bits and their delivery at the output of the registers. Nevertheless as shown by the equation listed above, each parity bit at the input of the registers Q0 to Q3 is calculated in one cycle as a function of the input information bits and the state values stored in registers X_1 to X_3. The state values are either initial state values or state values calculated during the previous parallel encoding cycle. The state values x_1 to x_3 are calculated, updated for the next parallel encoding cycle, and stored during the same clock cycle in which the parity bits are calculated. The initial or stored state values of D1 correspond to the initial or stored state values $S_0[0]$

to $S_2[0]$ and $S_0[4]$ to $S_2[4]$ shown in figure 14 of the present application.

Like the parallel encoder shown in figure 5A of the application, the parallel encoder of D1 does not provide intermediate state values similar to the state values $S_0[1]:S_0[3]$, $S_1[0]:S_1[3]$ and $S_2[1]:S_2[3]$ provided by the look ahead state generator 1202 of figure 14.

Hence, applying the teaching of D1 to the encoder of D3 would not lead to
- an encoder apparatus comprising means for calculating a set of state values $S_0[1]:S_0[3]$, $S_1[0]:S_1[3]$ and $S_2[1]:S_2[3]$ and means for generating a set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$ enabled during a single clock cycle, or to
- a method of encoding data performing the steps of calculating state values $S_0[1]:S_0[3]$, $S_1[0]:S_1[3]$ and $S_2[1]:S_2[3]$ for the encoder, based on the plurality of input information bits $I[0]:I[3]$ and initial or stored state values $S_o[0]$, $S_1[0]$ and $S_2[0]$; and generating a set of encoded output values $X[0]:X[3]$, $Y_0[0]:Y_0[3]$, $Y_1[0]:Y_1[3]$ using the calculated state values $S_0[1]:S_0[3]$, $S_1[0]:S_1[3]$ and $S_2[1]:S_2[3]$ and the plurality of input information bits $I[0]:I[3]$ and the initial or stored state values $S_o[0]$ and $S_1[0]$, during a single clock cycle.
Calculating said state values $S_0[1]:S_0[3]$, $S_1[0]:S_1[3]$ and $S_2[1]:S_2[3]$ allows using them in common for generating both the first and second parity bits elements $Y_0[0]:Y_0[3]$ and $Y_1[0]:Y_1[3]$.
The method of encoding data according to claim 1 and the encoder apparatus according to claim 9 are therefore not obvious in the light of the combination

of documents D3 and D1 (Article 56 EPC).

The subject-matter of dependent claims 2 to 8 and 10 to 16 therefore also involves an inventive step.

Order

For these reasons it is decided that:

1. The decision under appeal is set aside.

2. The case is remitted to the department of first instance with the order to grant a patent in the following version:

   - description pages 1, 2, 2a and 3 to 42 filed at the oral proceedings of 9 January 2013;

   - claims 1 to 16 of new main request filed at the oral proceedings of 9 January 2013;

   - drawing sheets 1/19 and 2/19, 8/19 to 12/19 and 14/19 to 19/19 of the published application and drawing sheets 3/19 to 7/19 and 13/19 as filed with the letter dated 21 November 2012.

The Registrar: The Chairman:

U. Bultmann M. Ruggiu