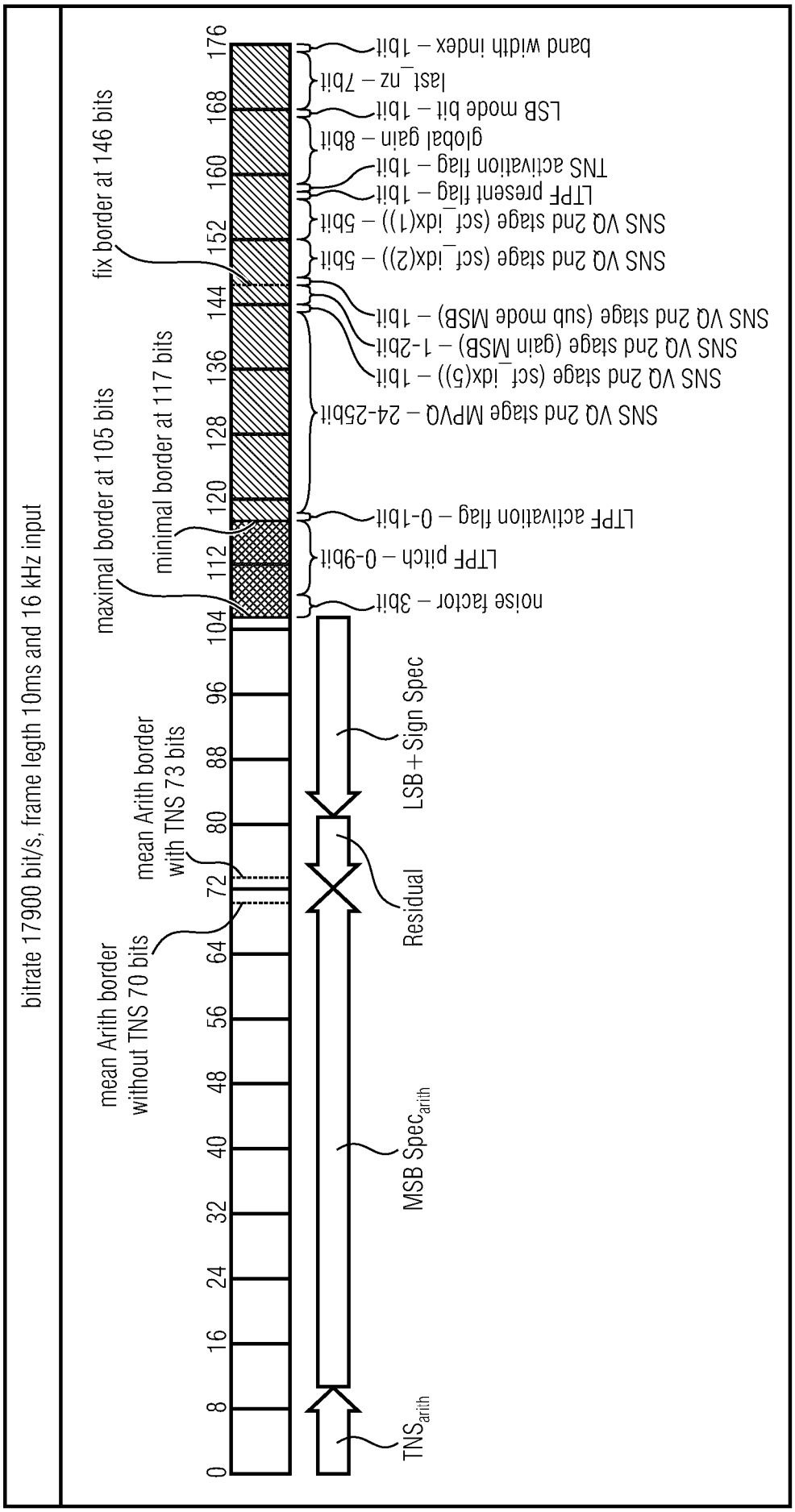
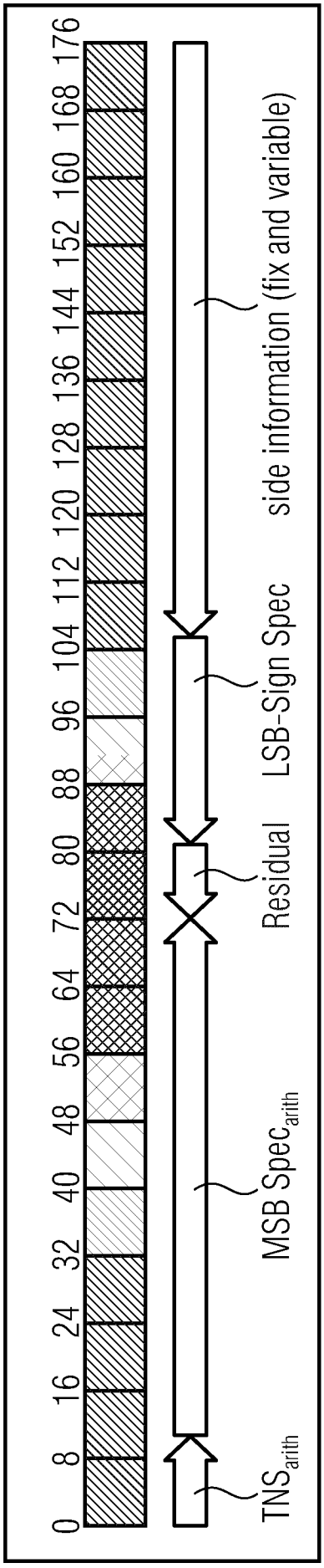


example of original LC3 bitstream payload

Fig. 1



distribution of bits of the example LC3 bitstream payload given in Fig. 1 based on their psychoacoustic relevancy

Fig. 2

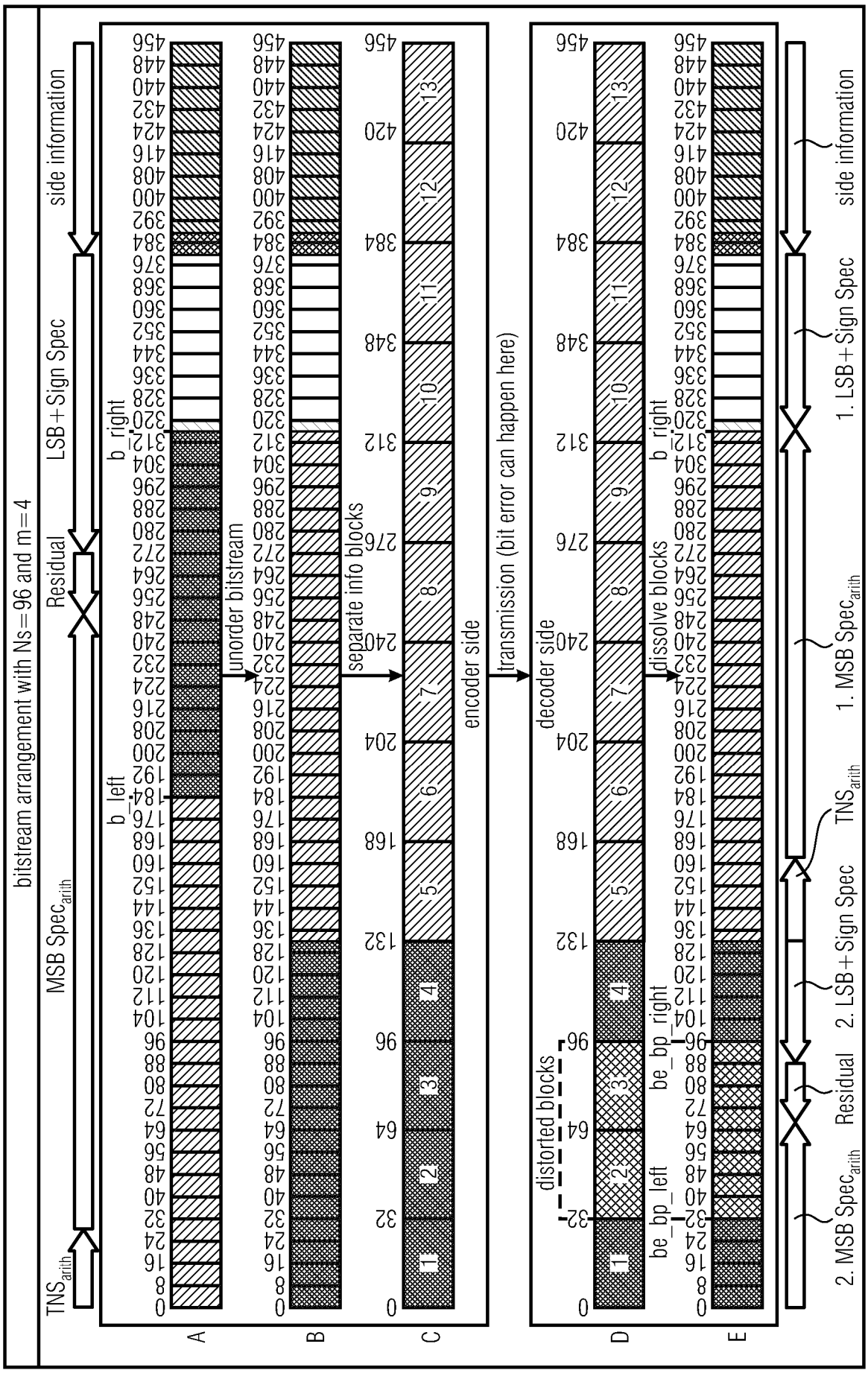


Fig. 3 example of LC3 bitstream payload arrangement

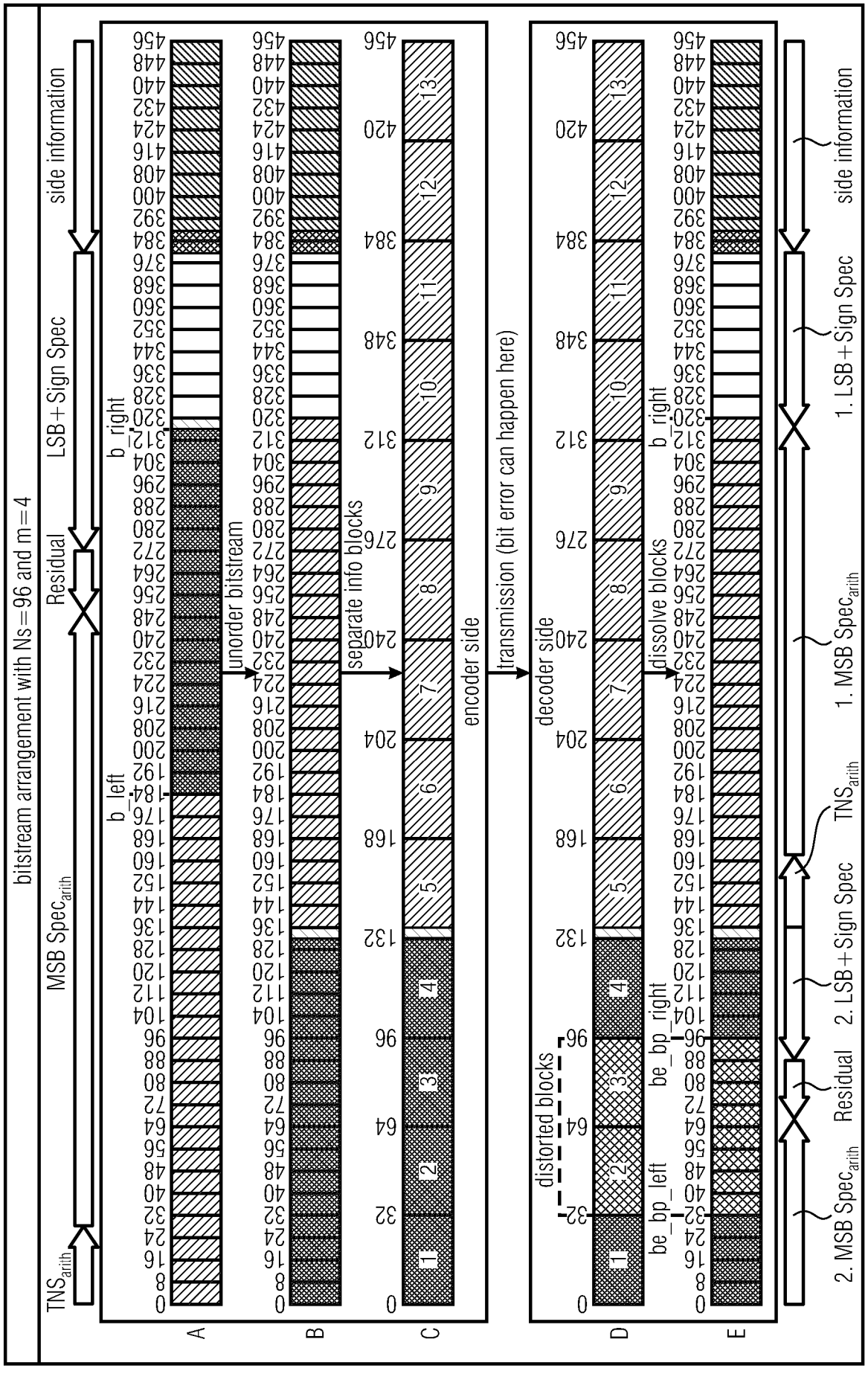


Fig. 4 example of LC3 bitstream payload rearrangement with arithmetic encoder/decoder operating on byte granularity

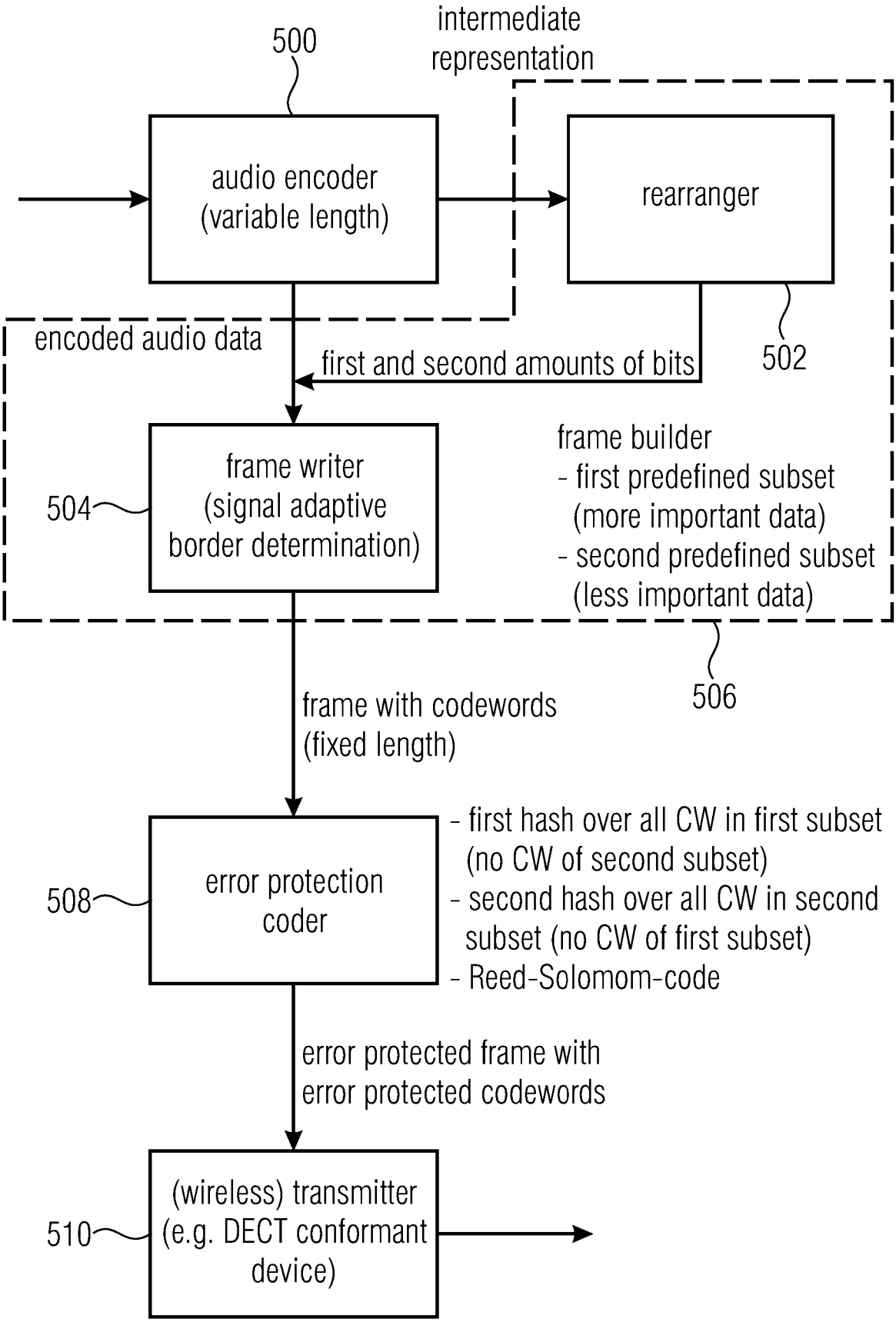

500
intermediate representation
audio encoder (variable length)
rearranger
encoded audio data
first and second amounts of bits
502
frame writer (signal adaptive border determination)
504
frame builder
- first predefined subset (more important data)
- second predefined subset (less important data)
506
frame with codewords (fixed length)
error protection coder
508
- first hash over all CW in first subset (no CW of second subset)
- second hash over all CW in second subset (no CW of first subset)
- Reed-Solomom-code
error protected frame with error protected codewords
(wireless) transmitter (e.g. DECT conformant device)
510

Fig. 5

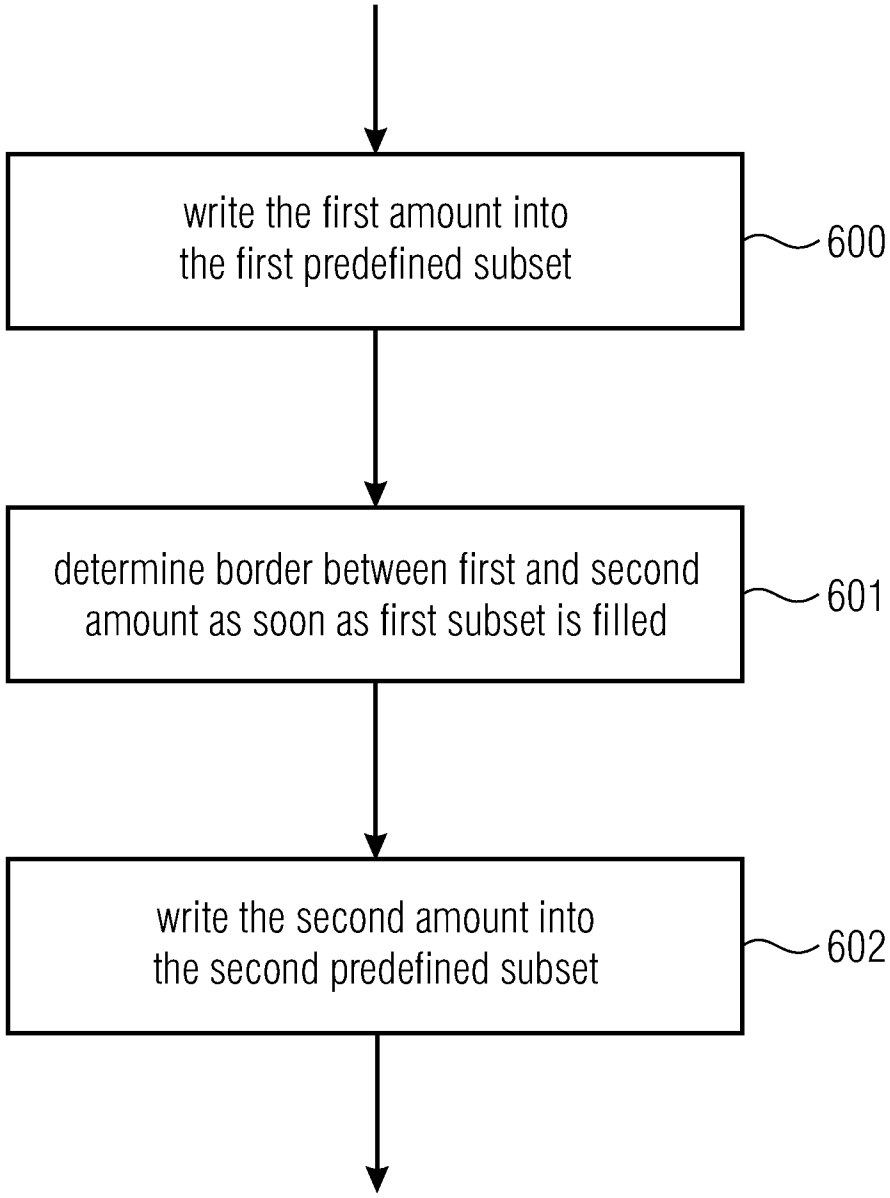

write the first amount into the first predefined subset
600
determine border between first and second amount as soon as first subset is filled
601
write the second amount into the second predefined subset
602

Fig. 6

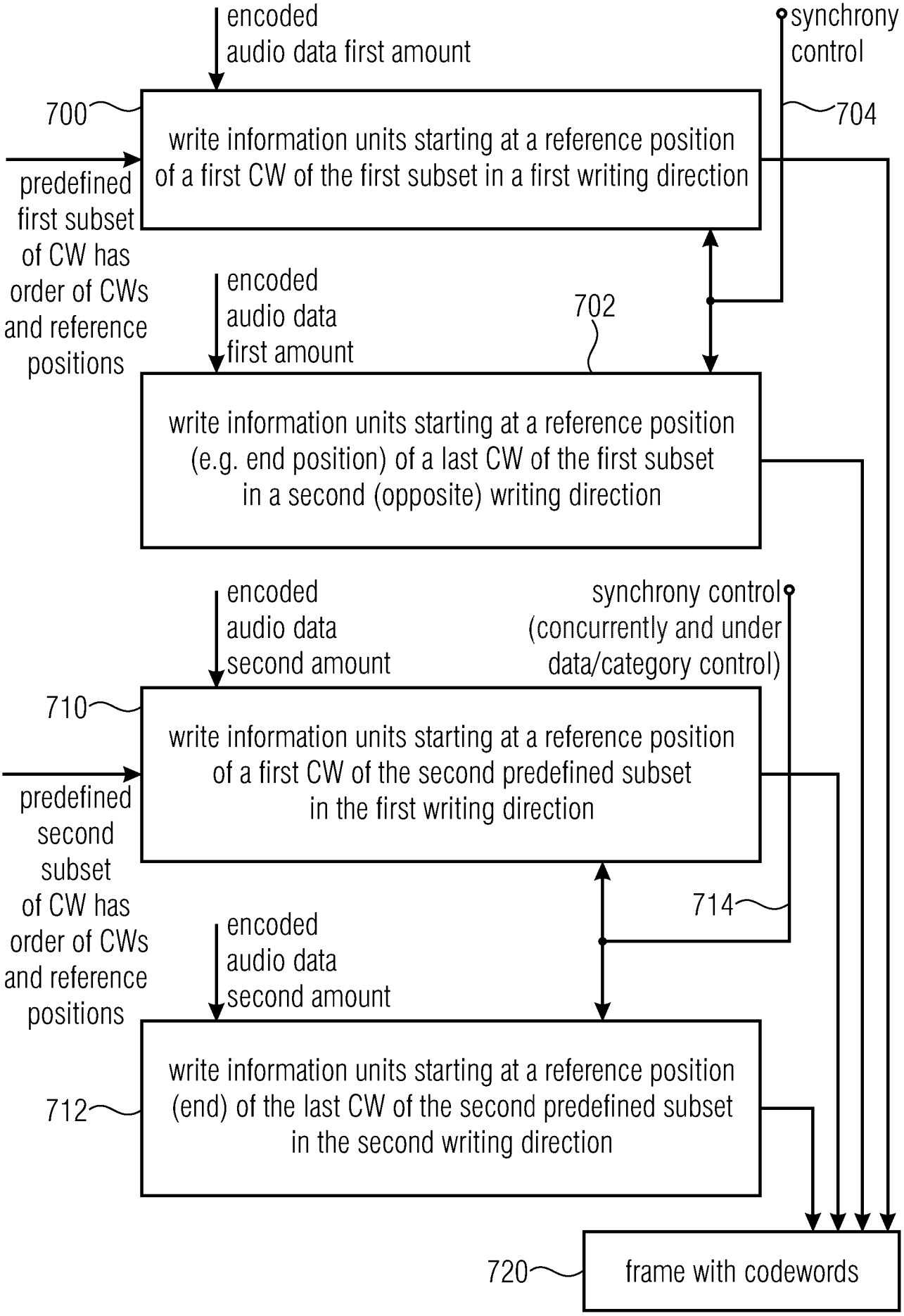
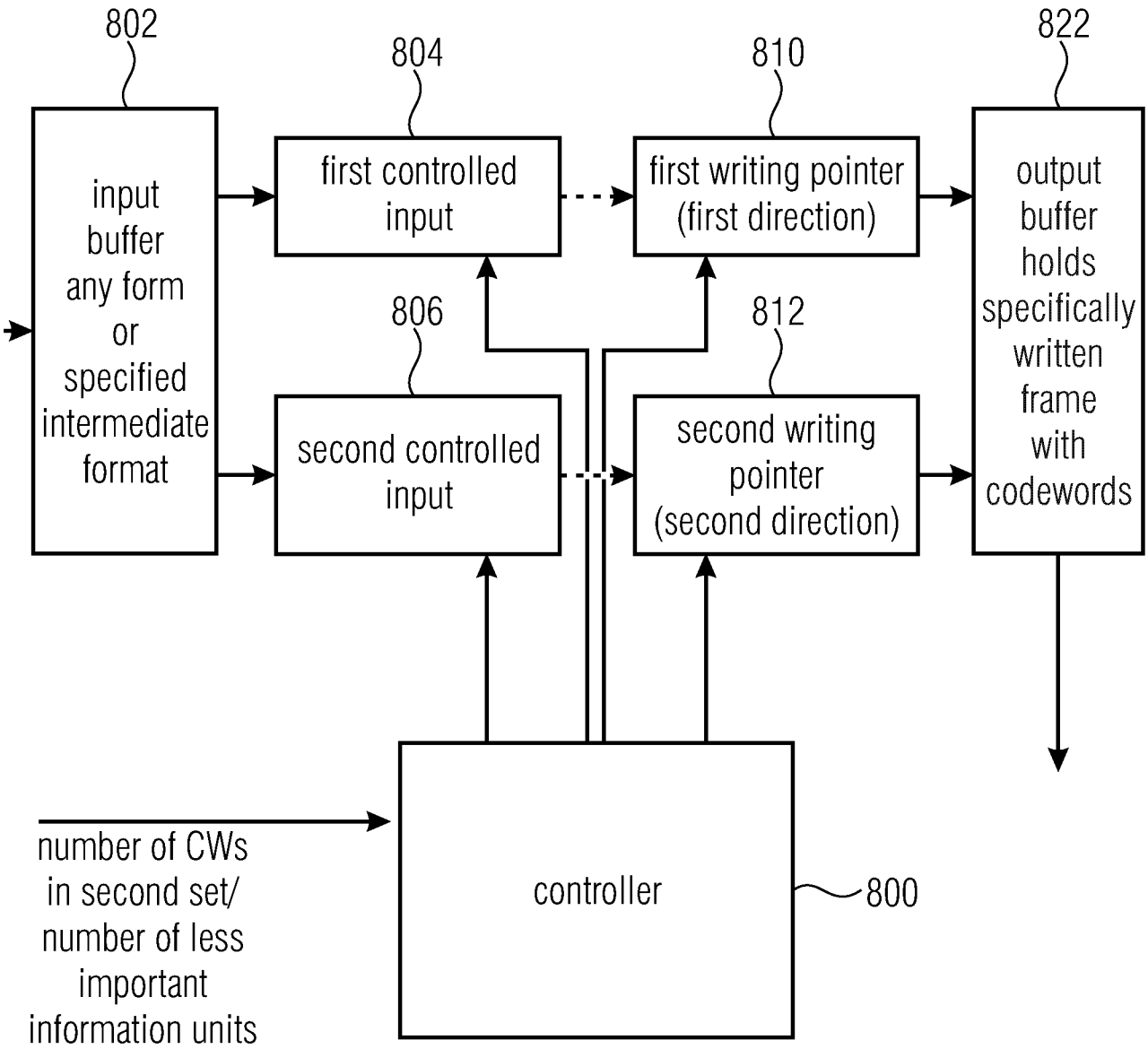

encoded audio data first amount
synchrony control
700
write information units starting at a reference position of a first CW of the first subset in a first writing direction
704
predefined first subset of CW has order of CWs and reference positions
encoded audio data first amount
702
write information units starting at a reference position (e.g. end position) of a last CW of the first subset in a second (opposite) writing direction
encoded audio data second amount
synchrony control (concurrently and under data/category control)
710
write information units starting at a reference position of a first CW of the second predefined subset in the first writing direction
predefined second subset of CW has order of CWs and reference positions
714
encoded audio data second amount
write information units starting at a reference position (end) of the last CW of the second predefined subset in the second writing direction
712
720
frame with codewords

Fig. 7



-controls inputs
- sets first and second pointers to start positions/ addresses in frame for first amount of information units
- synchronously increments writing pointers
- detects that all CW of the first subset are written into the output buffer
- sets first and second pointers‘ start positions/addresses for the second amount of information units
- synchronously increments writing pointers

Fig. 8

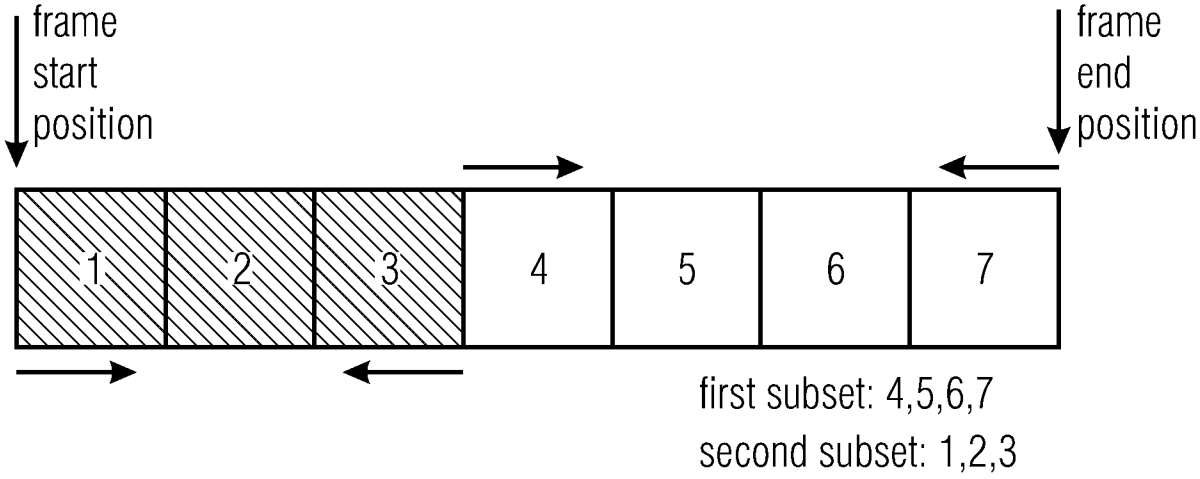


Fig. 9a

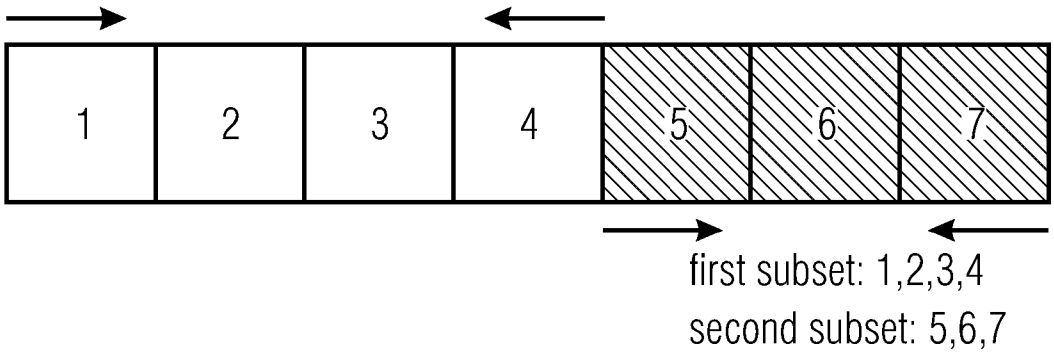


Fig. 9b

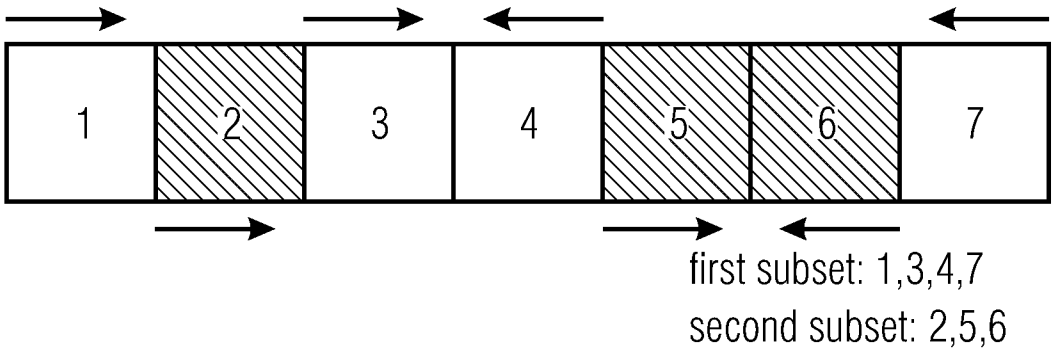


Fig. 9c

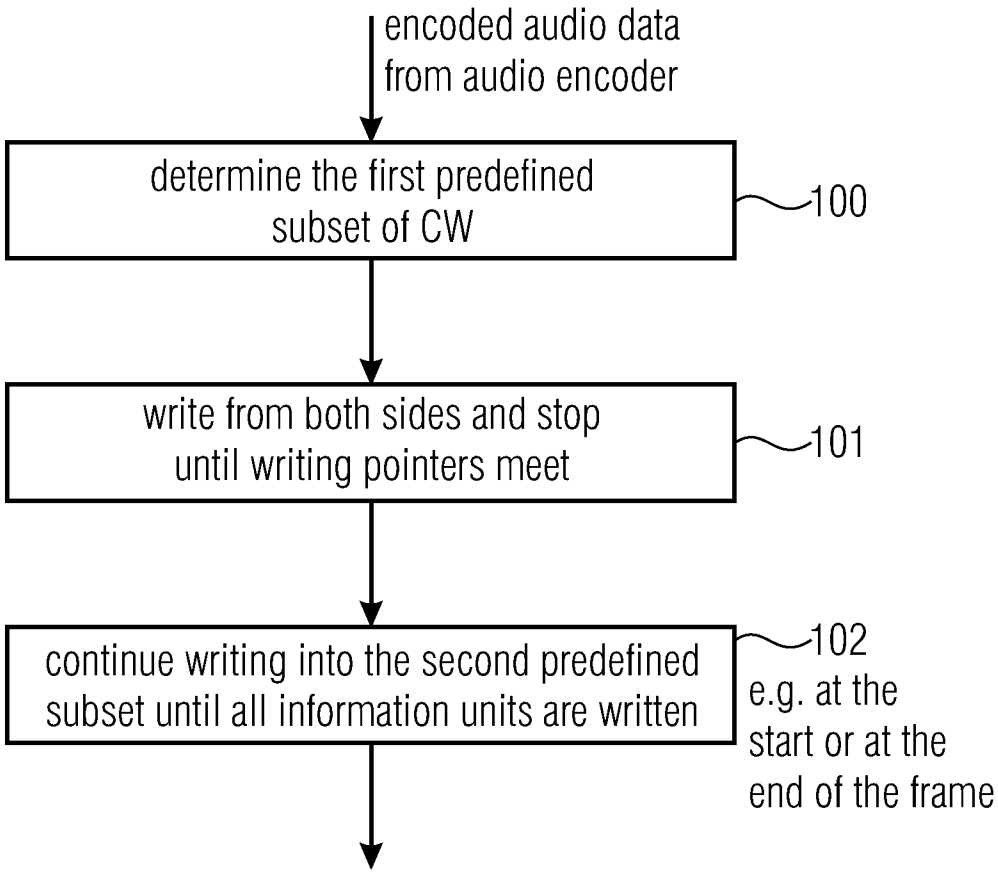


Fig. 10
(DIRECT WRITING)

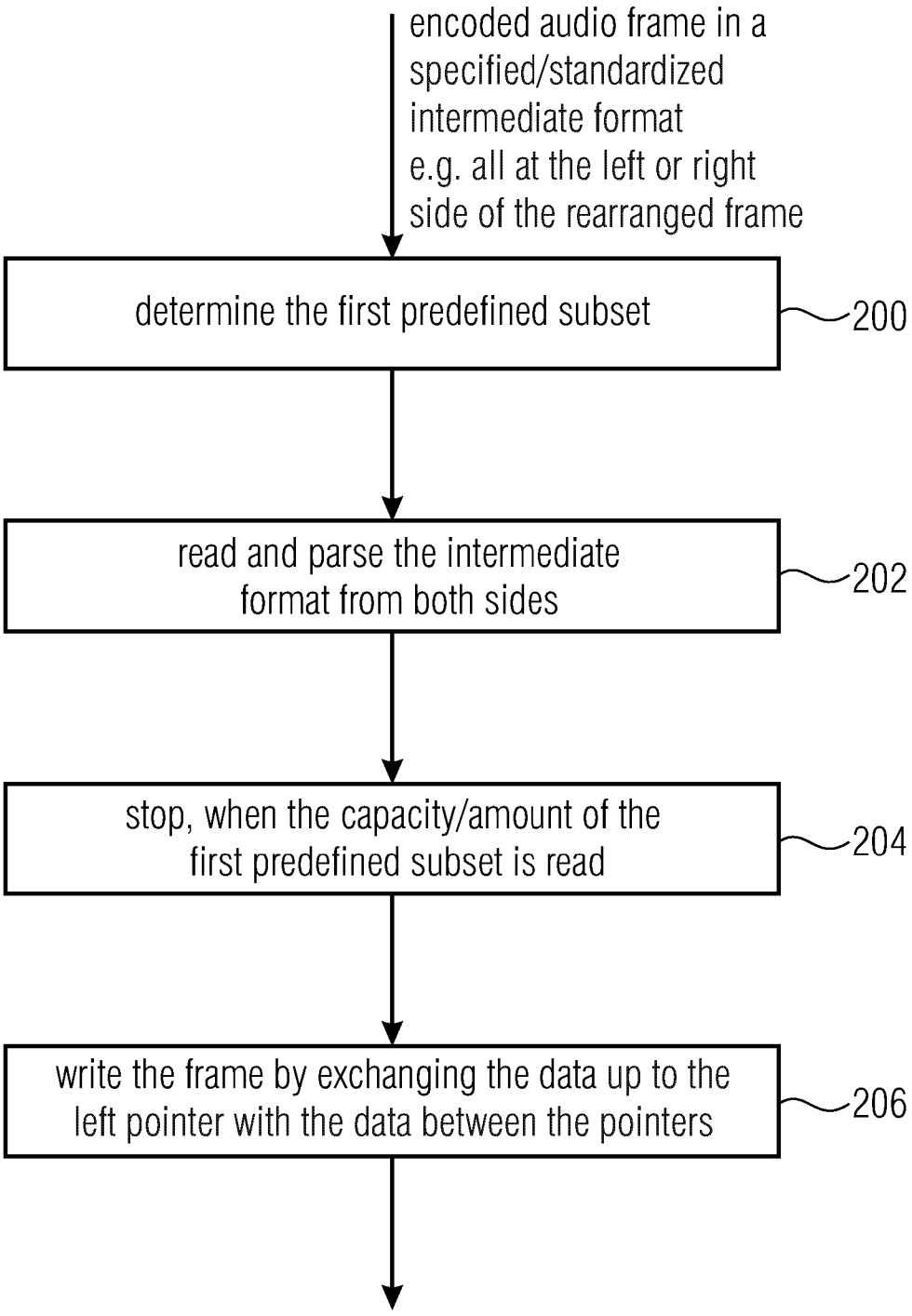


Fig. 11
(REARRANGING)

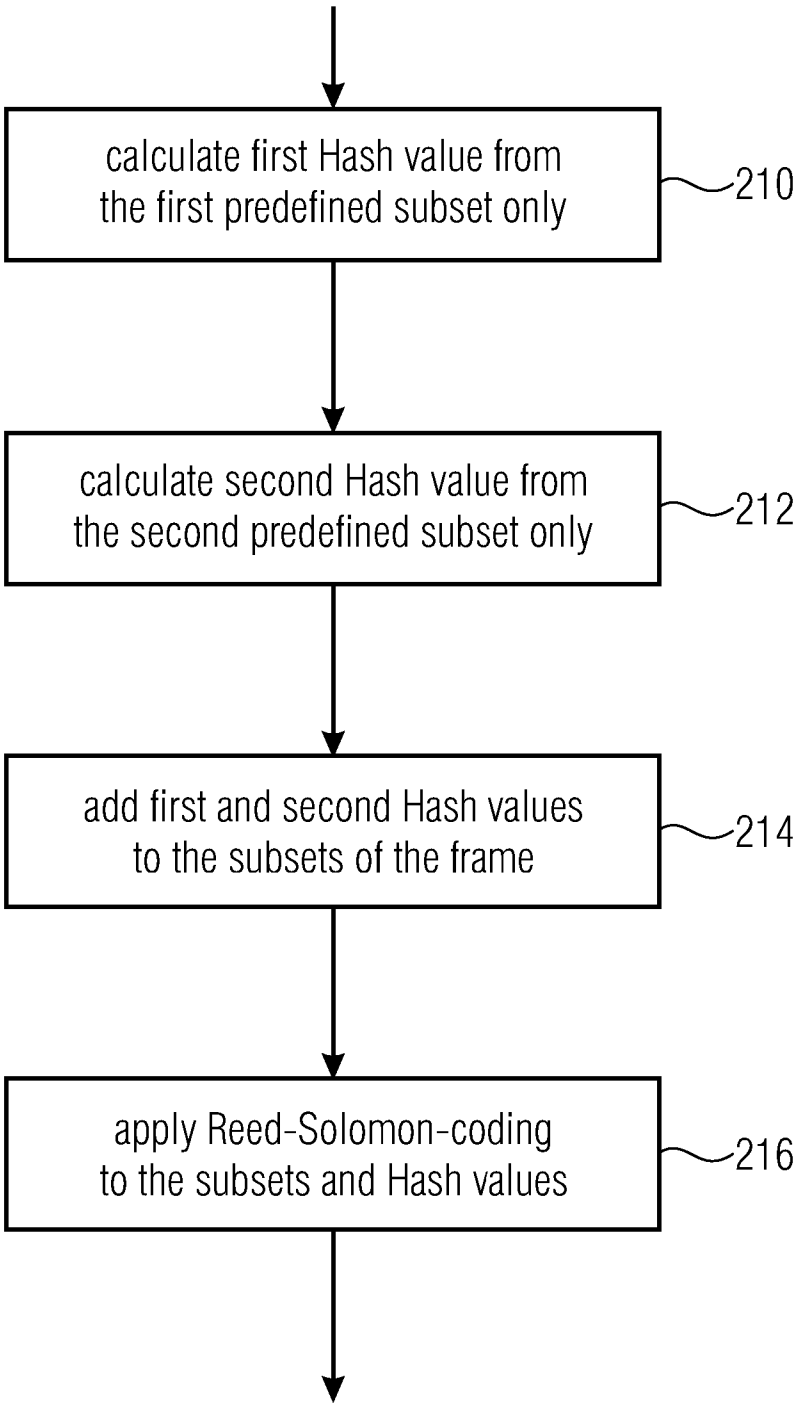

calculate first Hash value from the first predefined subset only
210
calculate second Hash value from the second predefined subset only
212
add first and second Hash values to the subsets of the frame
214
apply Reed-Solomon-coding to the subsets and Hash values
216

Fig. 12

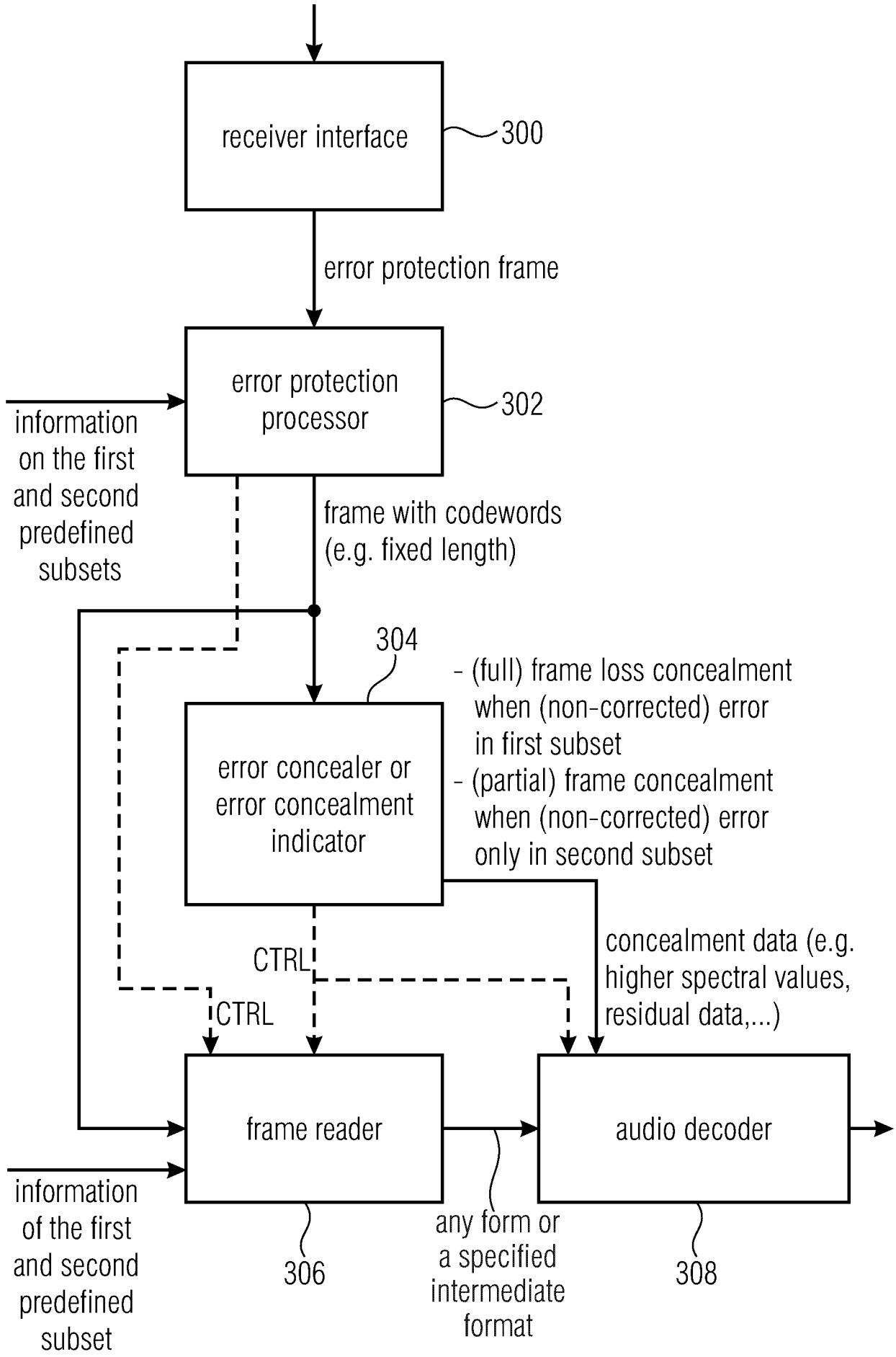

receiver interface
300
error protection frame
error protection processor
302
information on the first and second predefined subsets
frame with codewords (e.g. fixed length)
304
- (full) frame loss concealment when (non-corrected) error in first subset
- (partial) frame concealment when (non-corrected) error only in second subset
error concealer or error concealment indicator
CTRL
CTRL
concealment data (e.g. higher spectral values, residual data,...)
frame reader
audio decoder
information of the first and second predefined subset
306
any form or a specified intermediate format
308

Fig. 13

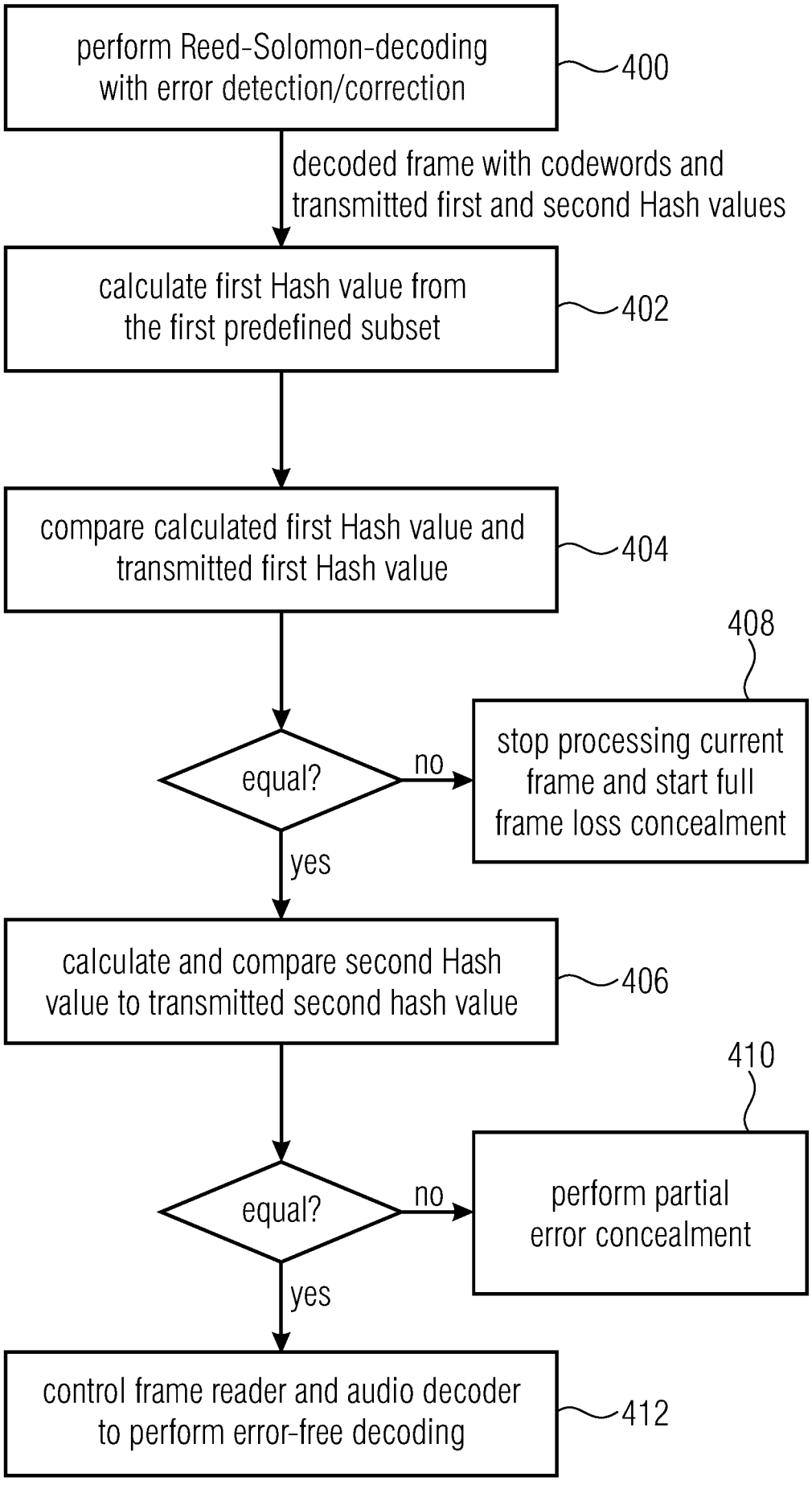

perform Reed-Solomon-decoding with error detection/correction
400
decoded frame with codewords and transmitted first and second Hash values
calculate first Hash value from the first predefined subset
402
compare calculated first Hash value and transmitted first Hash value
404
408
equal?
no
stop processing current frame and start full frame loss concealment
yes
calculate and compare second Hash value to transmitted second hash value
406
410
equal?
no
perform partial error concealment
yes
control frame reader and audio decoder to perform error-free decoding
412

Fig. 14

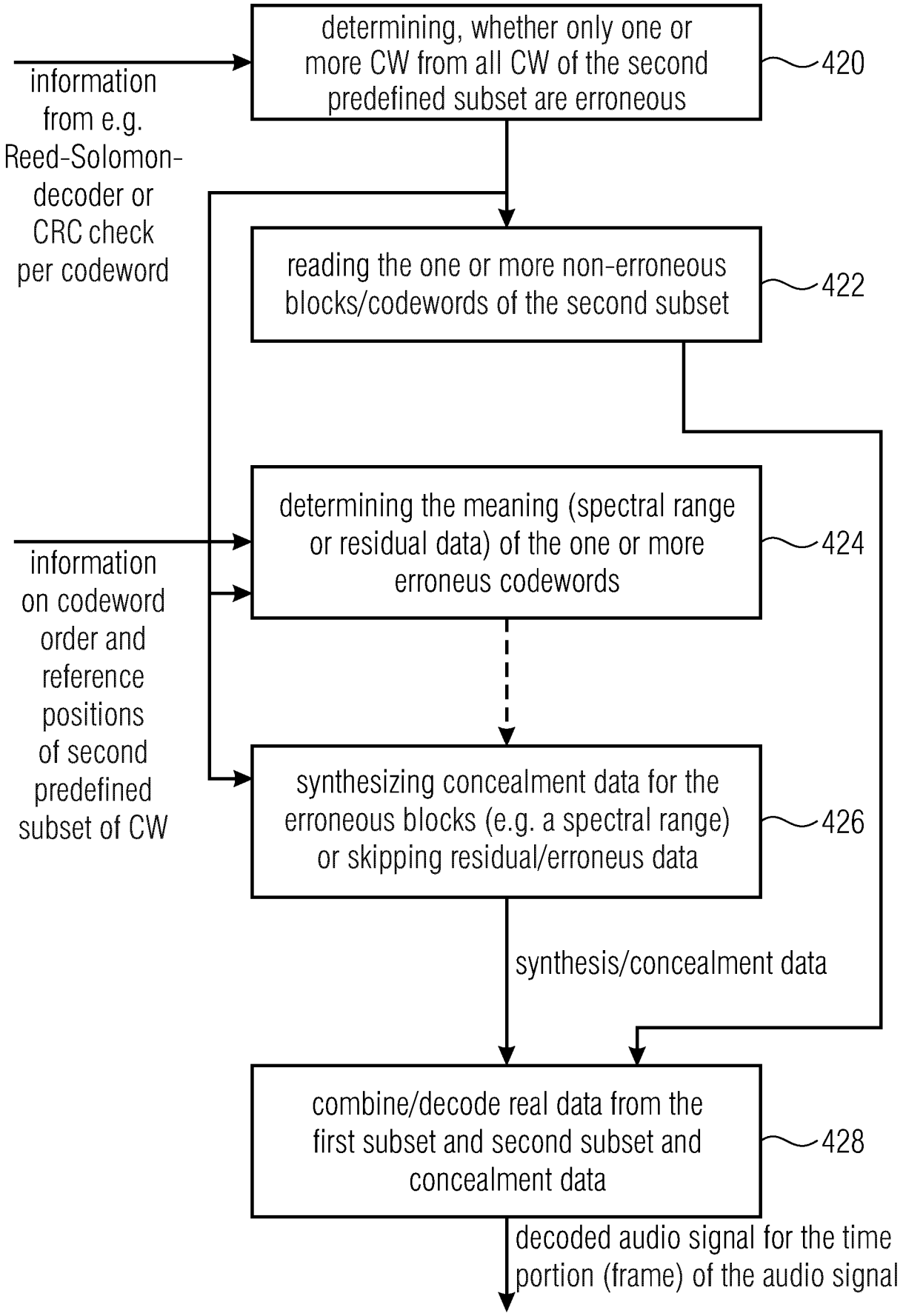

determining, whether only one or more CW from all CW of the second predefined subset are erroneous
420
information from e.g. Reed-Solomon-decoder or CRC check per codeword
reading the one or more non-erroneous blocks/codewords of the second subset
422
determining the meaning (spectral range or residual data) of the one or more erroneus codewords
424
information on codeword order and reference positions of second predefined subset of CW
synthesizing concealment data for the erroneous blocks (e.g. a spectral range) or skipping residual/erroneus data
426
synthesis/concealment data
combine/decode real data from the first subset and second subset and concealment data
428
decoded audio signal for the time portion (frame) of the audio signal

Fig. 15

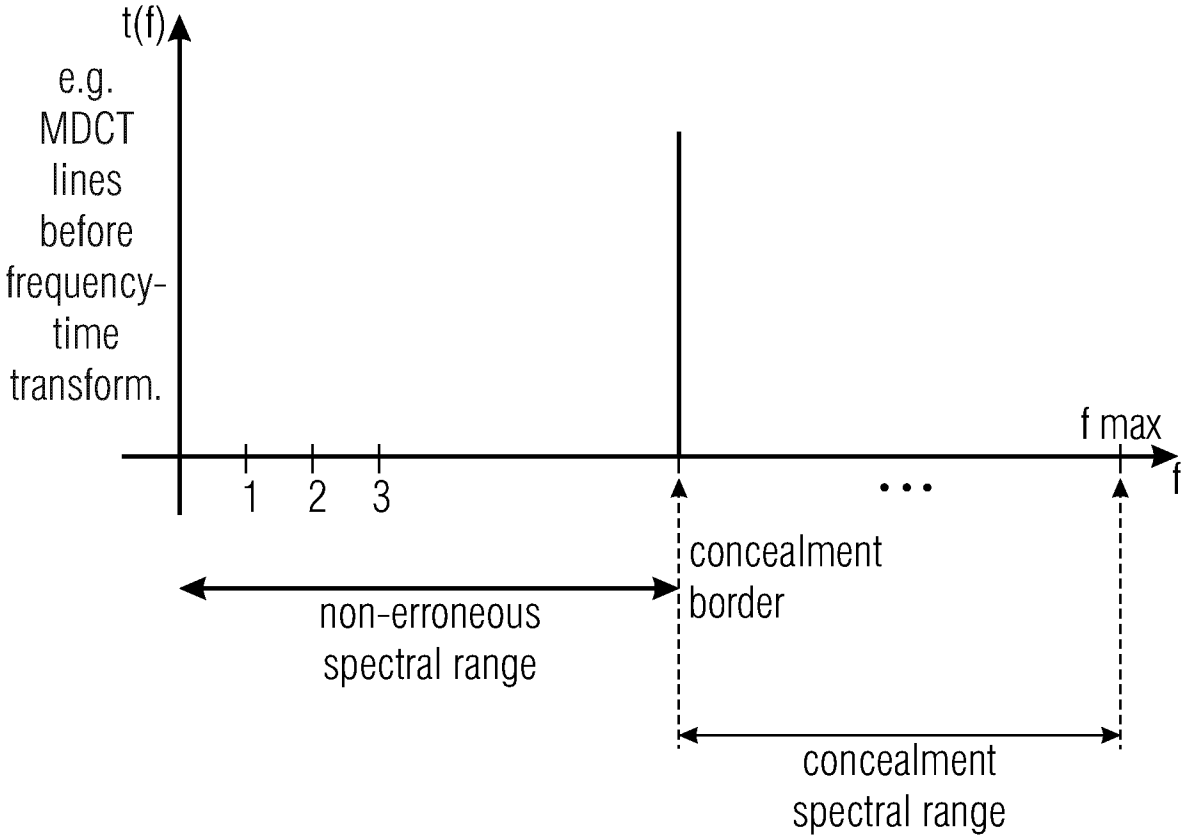

t(f)
e.g.
MDCT
lines
before
frequency-
time
transform.
f max
f
1
2
3
concealment
border
non-erroneous
spectral range
concealment
spectral range

Fig. 16

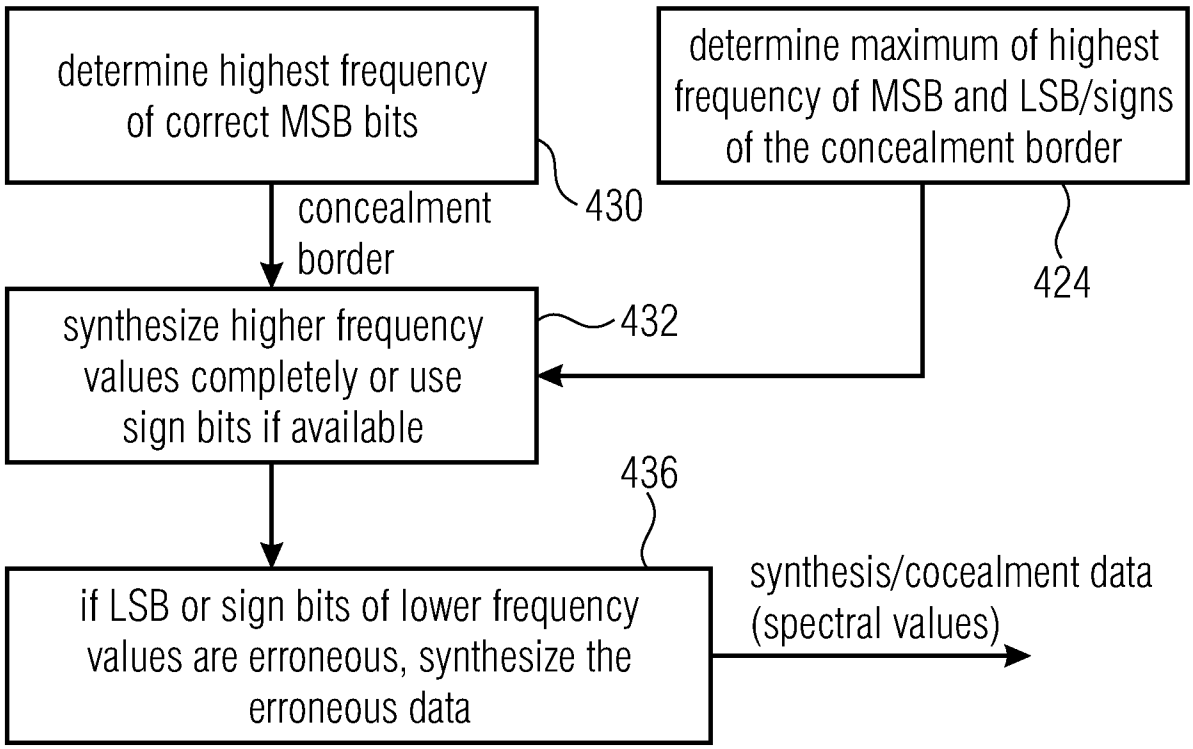

determine highest frequency of correct MSB bits
430
concealment
border
determine maximum of highest frequency of MSB and LSB/signs of the concealment border
424
synthesize higher frequency values completely or use sign bits if available
432
436
if LSB or sign bits of lower frequency values are erroneous, synthesize the erroneous data
synthesis/cocealment data (spectral values)

Fig. 17

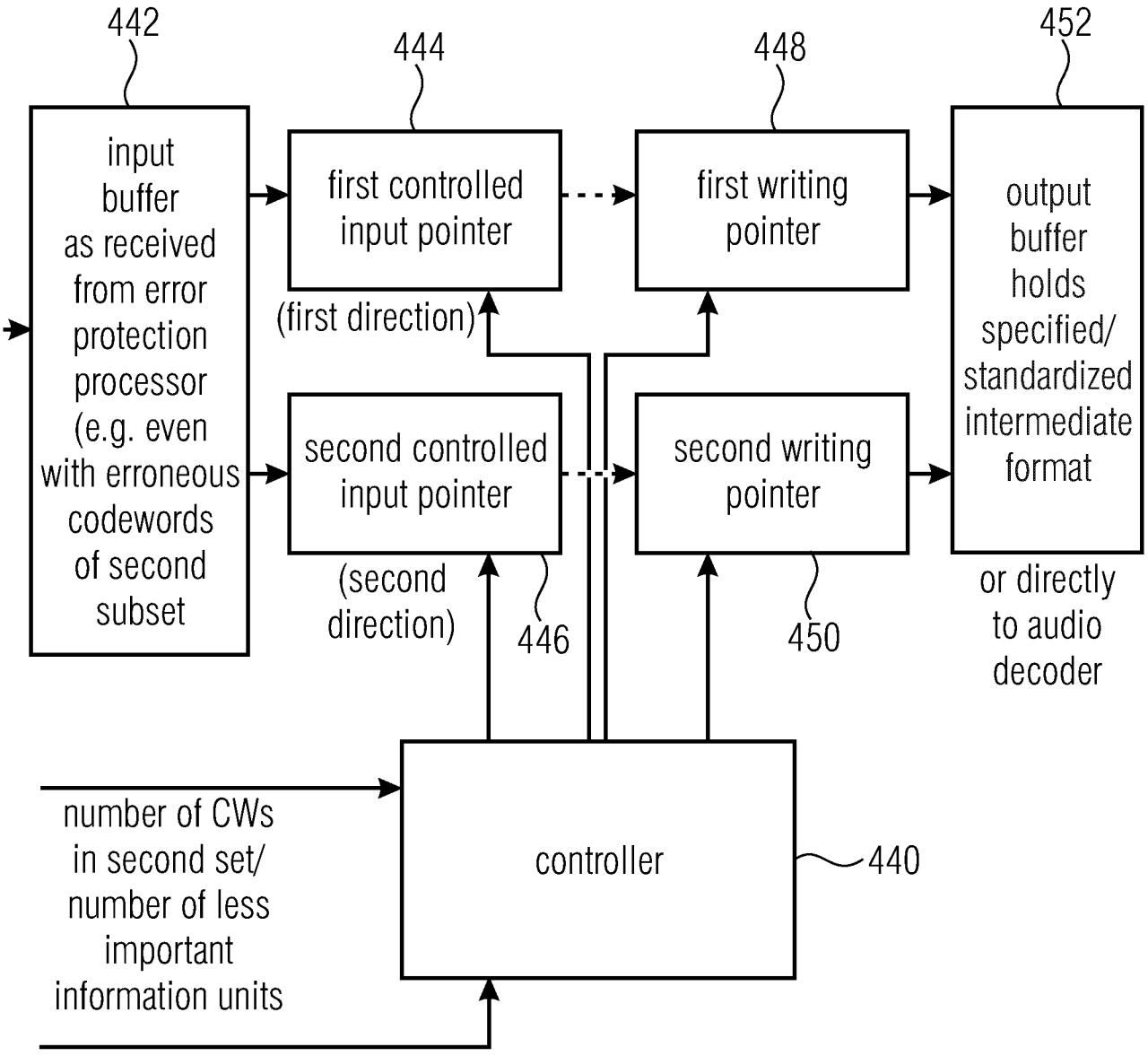

442
444
448
452
input buffer as received from error protection processor (e.g. even with erroneous codewords of second subset
first controlled input pointer
first writing pointer
output buffer holds specified/ standardized intermediate format
(first direction)
second controlled input pointer
second writing pointer
(second direction)
446
450
or directly to audio decoder
number of CWs in second set/ number of less important information units
controller
440
indication of erroneous CW of second subset from error protection processor
- controls writing pointers (e.g. for data exchange or by increments
- sets first and second pointers to start positions/addresses in frame for first amount of information units
- synchronously increments input (reading) pointers
- detects that all CW of the first subset are read from the intput buffer
- sets first and second pointers‘ start positions/addresses for the second amount of information units
- synchronously increments reading pointers

Fig. 18

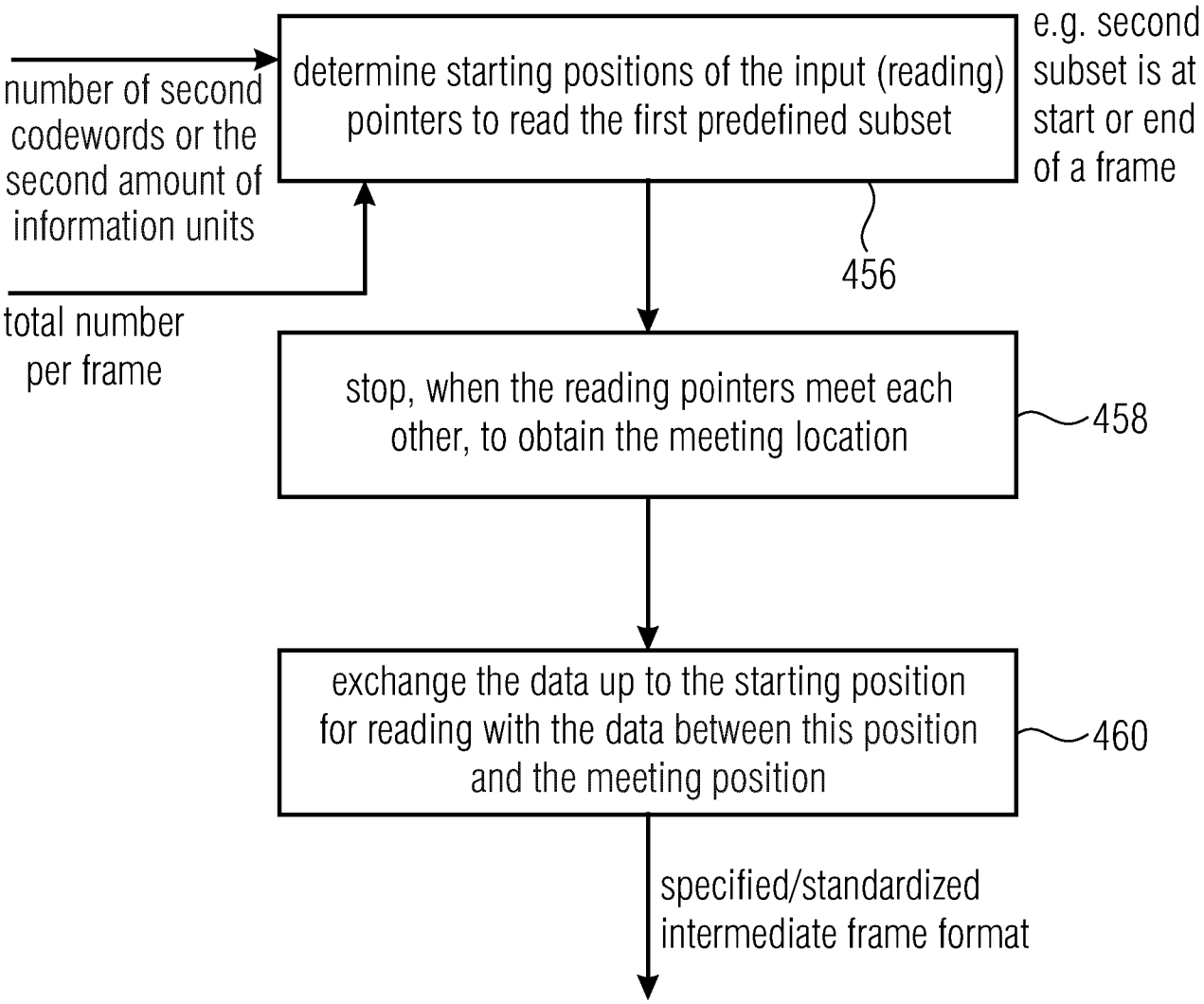


Fig. 19
(REARRANGING)

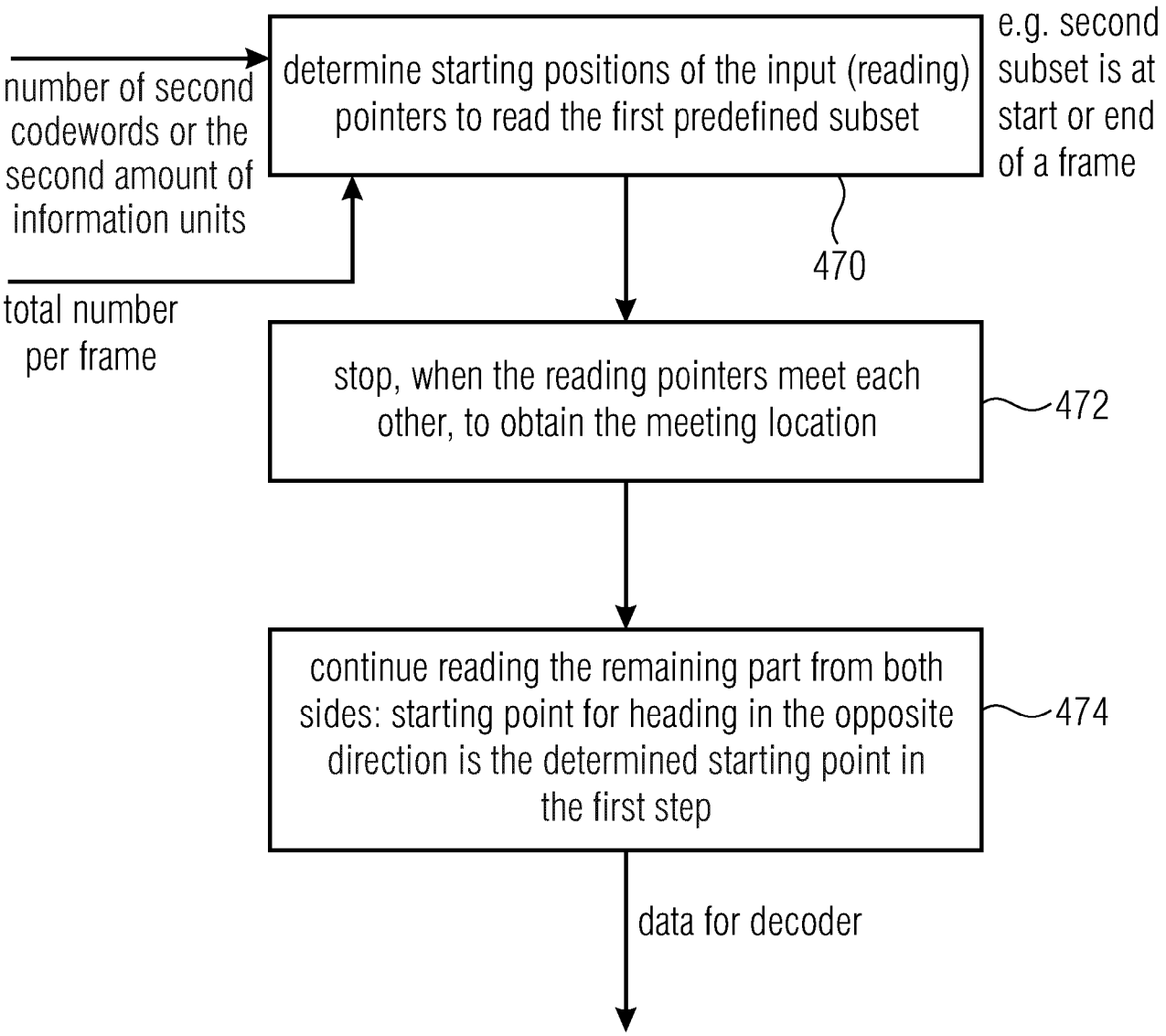


Fig. 20
(DIRECT DECODING)

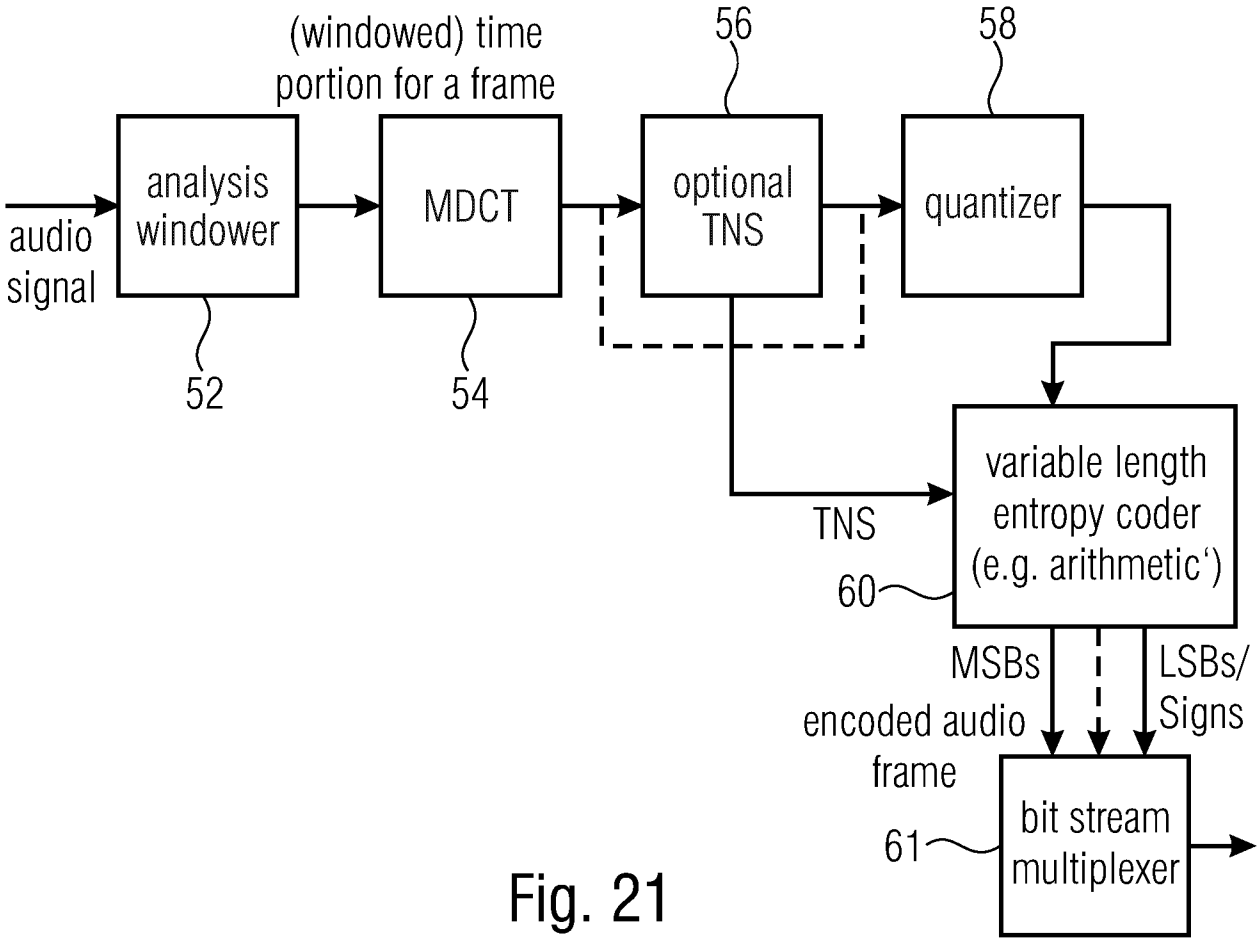

(windowed) time portion for a frame
56
58
audio signal
analysis windower
MDCT
optional TNS
quantizer
52
54
TNS
variable length entropy coder (e.g. arithmetic')
60
MSBs
LSBs/ Signs
encoded audio frame
61
bit stream multiplexer

Fig. 21

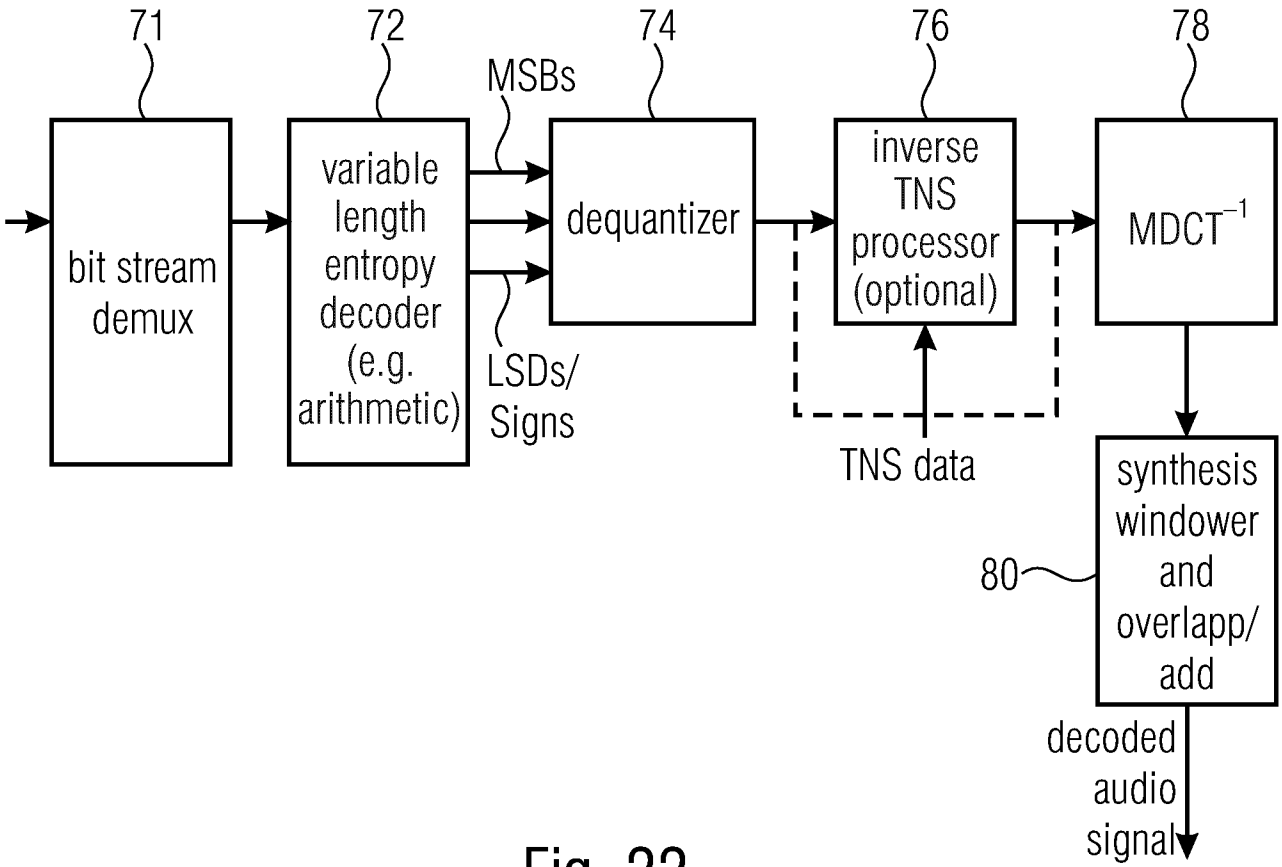

71
72
MSBs
74
76
78
bit stream demux
variable length entropy decoder (e.g. arithmetic)
dequantizer
inverse TNS processor (optional)
MDCT^{-1}
LSDs/ Signs
TNS data
synthesis windower and overlapp/ add
80
decoded audio signal

Fig. 22