Claims

1. Encoder ~~(100)~~, comprising:

a quantization stage ~~(102)~~ configured to quantize an input signal ~~(140)~~ using a dead zone in order to obtain a plurality of quantized values ~~(142)~~;

an entropy encoder ~~(104)~~ configured to encode the plurality of quantized values ~~(142)~~ using an entropy encoding scheme in order to obtain a plurality of entropy encoded values ~~(144)~~;

a residual quantization stage ~~(106)~~ configured to quantize a residual signal caused by the quantization stage ~~(102)~~, wherein the residual quantization stage ~~(106)~~ is configured to determine, for a non-zero quantized value, at least one quantized residual value ~~(146)~~ in dependence on a width of the dead zone of the quantization stage ~~(102)~~; and

a coded signal former ~~(108)~~ configured to form a coded signal ~~(148)~~ from the plurality of entropy encoded values ~~(144)~~ and the at least one quantized residual value ~~(146)~~;

wherein the coded signal former ~~(108)~~ is configured to form the coded signal ~~(148)~~ by appending the at least one quantized residual value ~~(146)~~ or a plurality of quantized residual values ~~(146)~~ to the plurality of entropy encoded values ~~(144)~~ until the coded signal ~~(148)~~ comprises a maximum length available for a transfer to a decoder;

wherein the residual quantization stage ~~(106)~~ is configured to determine the at least one quantized residual value ~~(146)~~ such that the quantized residual value ~~(146)~~ comprises 1 bit for a non-zero quantized value, wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ based on the comparison

$$x[i] < x_Q[i]$$

wherein x[i] is the input signal ~~140~~, wherein x_Q[i] is the inverse quantized input signal ~~(152)~~, and wherein i is an index that is incremented by 1 for each obtained quantized value ~~(142)~~;

wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ such that the quantized residual value ~~(146)~~ comprises 2 bits for a zero quantized value, wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ based on the comparisons

$$|x[i]| < (C \cdot x_Q[i]) \text{ and } x[i] < 0$$

wherein C depends on the width of the dead zone of the quantization stage ~~(102)~~, wherein x[i] is the input signal ~~(140)~~, wherein x_Q[i] is the inverse quantized input signal ~~(152)~~, and wherein i is an index that is incremented by 1 for each obtained quantized value ~~(142)~~;

wherein the at least one quantized residual value ~~(146)~~ is not entropy encoded.

2. Encoder ~~(100)~~ according to claim 1, wherein the residual quantization stage ~~(106)~~ comprises:

an inverse quantizer ~~(160)~~ configured to inverse quantize the plurality of quantized values ~~(112)~~ in dependence on the dead zone of the quantization stage ~~(102)~~ in order to obtain an inverse quantized input signal ~~(152)~~;

wherein the residual quantization stage ~~(106)~~ is configured to determine the at least one quantized residual value ~~(146)~~ such that the quantized residual value ~~(146)~~ comprises 1 bit for a non-zero quantized value; and

wherein the residual quantization stage ~~(106)~~ is configured to determine, for the non-zero quantized value, that the quantized residual value ~~(146)~~ comprises a logic 0 if the input signal ~~(140)~~ is smaller than the inverse quantized input signal ~~(152)~~ for the non-zero quantized value, and to determine, for the non-zero quantized value, that the quantized residual value ~~(146)~~ comprises a logic 1 else.

3. Encoder ~~(100)~~ according to one of the claims 1 to 2, wherein the residual quantization stage ~~(106)~~ comprises:

an inverse quantizer ~~(160)~~ configured to inverse quantize the plurality of quantized values ~~(142)~~ in dependence on the dead zone in order to obtain an inverse quantized input signal ~~(152)~~; and

a comparer ~~(162)~~ configured to compare the input signal ~~(140)~~ and the inverse quantized input signal ~~(152)~~.

4. Encoder ~~(100)~~ according to claim 3, wherein the comparer ~~(162)~~ is configured to compare the input signal ~~(140)~~ and the inverse quantized input signal ~~(152)~~ in order to obtain the residual signal ~~(154)~~;

wherein the residual quantization stage ~~(106)~~ is configured to quantize the residual signal ~~(154)~~ in dependence on the dead zone.

5. Encoder ~~(100)~~ according to claim 4, wherein the residual quantization stage ~~(106)~~ is configured to determine the at least one quantized residual value ~~(146)~~ such that the quantized residual value ~~(146)~~ comprises 1 bit for a non-zero quantized value; and

wherein the residual quantization stage is configured to determine, for the non-zero quantized value, that the quantized residual value ~~(146)~~ comprises a logic 0 if the residual signal ~~(154)~~ is negative for the non-zero quantized value, and to determine, for the non-zero quantized value, that the quantized residual value ~~(146)~~ comprises a logic 1 else.

6. Encoder ~~(100)~~ according to one of the claims 1 to 5, wherein the residual quantization stage ~~(106)~~ is configured to determine the at least one quantized residual value ~~(146)~~ such that the quantized residual value ~~(146)~~ comprises 1 bit for a non-zero quantized value, wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ based on the syntax

$$prm[n] = \begin{cases} 0 \; if \; x[i] < x_Q[i] \\ 1 \; otherwise \end{cases}$$

wherein prm is a bit stream generated by the residual quantization stage ~~(106)~~ using the quantized residual value ~~(146)~~, wherein x[i] is the input signal ~~140~~, wherein x_Q[i] is the inverse quantized input signal ~~(152)~~, wherein n is an index that is incremented by 1 for each non-zero quantized value, and wherein i is an index that is incremented by 1 for each obtained quantized value ~~(142)~~.

7. Encoder ~~(100)~~ according to one of the claims 1 to 6, wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ such that

the quantized residual value ~~(146)~~ comprises 2 bits for a zero quantized value, wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ based on the syntax

$$prm[n] = \begin{cases} 0 \; if \; |x[i]| < (C \cdot x_Q[i]) \\ 1 \; otherwise \end{cases}$$

$$if \; prm[n] == 1 \; then$$

$$prm[n+1] = \begin{cases} 0 \; if \; x[i] < 0 \\ 1 \; otherwise \end{cases}$$

wherein C depends on the dead zone of the quantization stage ~~(102)~~, wherein prm is a bit stream generated by the residual quantization stage ~~(106)~~ using the quantized residual value ~~(146)~~, wherein x[i] is the input signal ~~(140)~~, wherein x_Q[i] is the inverse quantized input signal ~~(152)~~, wherein n is an index that is incremented by 1 for each zero quantized value that is requantized to a zero quantized value and incremented by 2 for each zero quantized value that is requantized to a non-zero quantized value, and wherein i is an index that is incremented by 1 for each obtained quantized value ~~(142)~~.

8. Encoder ~~(100)~~ according to one of the claims 1 to 7, wherein the coded signal former ~~(108)~~ is configured to provide a bit stream as the coded signal ~~(148)~~, wherein the coded signal former ~~(108)~~ is configured to form the bit stream from the plurality of entropy encoded values ~~(144)~~ and the plurality of quantized residual values ~~(146)~~, wherein the coded signal former ~~(108)~~ is configured to append the quantized residual values ~~(146)~~ to the entropy encoded values ~~(144)~~, wherein the residual quantization stage ~~(106)~~ comprises:

a residual quantizer ~~(106')~~; and

an adjuster ~~(164)~~ configured to control the residual quantizer ~~(106')~~ to quantize the residual signal in dependence on a width of the dead zone used in the quantization stage ~~(102)~~ for obtaining the plurality of quantized values ~~(142)~~;

wherein the adjuster ~~(164)~~ is configured to obtain a number of target bits and a number of consumed bits; and

wherein the adjuster ~~(164)~~ is configured to control the residual quantization stage to stop determining quantized residual values when the bit stream comprises the number of target bits.

9. Decoder ~~(120)~~, comprising:

a coded signal parser ~~(122)~~ configured to parse a coded signal ~~(148)~~ in order to obtain a plurality of entropy encoded values ~~(144)~~ and at least one quantized residual value ~~(146)~~;

an entropy decoder ~~(124)~~ configured to decode the plurality of entropy encoded values ~~(144)~~ using an entropy decoding scheme in order to obtain a plurality of quantized values ~~(142)~~; and

an inverse quantization stage ~~(126)~~ configured to inverse quantize the plurality of quantized values ~~(142)~~ in order to obtain an output signal ~~(150)~~;

wherein the inverse quantization stage ~~(126)~~ is configured to refine an inverse quantization level ~~(172)~~ used for obtaining the output signal ~~(150)~~ in dependence on the quantized residual value ~~(146)~~;

wherein the inverse quantization stage ~~(126)~~ is configured to refine an inverse quantization level ~~(172)~~ for a non-zero quantized value in dependence on a quantized residual value ~~(146)~~ and a width of the dead zone.

10. Decoder ~~(120)~~ according to claim 9, wherein the inverse quantization stage ~~(126)~~ is configured to refine the inverse quantization level ~~(172)~~ by determining a refined inverse quantization level ~~(174, 176)~~ in dependence on the dead zone.

11. Decoder ~~(120)~~ according to claim 10, wherein the inverse quantization stage ~~(126)~~ is configured to determine two refined inverse quantization levels ~~(174, 176)~~ for a non-zero quantized value, wherein the inverse quantization stage ~~(126)~~ is configured to obtain the output signal ~~(150)~~ by using one out of the two refined inverse quantization levels ~~(174, 176)~~ indicated by quantized residual value.

12. Decoder ~~(120)~~ according to claim 11, wherein the inverse quantization stage ~~(126)~~ is configured to increase a normalized absolute value of the inverse quantization level

~~(172)~~ by an increase value in order to obtain a first of the two refined inverse quantization levels ~~(174)~~;

wherein the inverse quantization stage ~~(126)~~ is configured to decrease a normalized absolute value of the inverse quantization level ~~(172)~~ by a decrease value in order to obtain a second of the two refined inverse quantization levels ~~(174)~~; and

wherein the increase value and the decrease value are different from each other.

13. Decoder ~~(120)~~ according to one of the claims 11 or 12, wherein the inverse quantization stage ~~(126)~~ is configured to determine the two refined inverse quantization levels ~~(174, 176)~~ for a non-zero quantized value based on the two factors:

fac_p = 0,25 * dz

fac_m = 0,5 * (1 - 0,5 * dz)

wherein fac_p indicates a normalized absolute value by which a normalized absolute value of the inverse quantization level ~~(172)~~ is to be increased in order to obtain a first of the two refined inverse quantization levels ~~(174)~~, wherein fac_m indicates a normalized absolute value by which the normalized absolute value of the inverse quantization level ~~(172)~~ is to be decreased in order to obtain a second of the two refined inverse quantization levels ~~(176)~~, and wherein dz is a normalized width of the dead zone.

14. Decoder ~~(120)~~ according to one of the claims 9 to 13, wherein the inverse quantization stage ~~(126)~~ is configured to determine two refined inverse quantization levels ~~(174, 176)~~ for a zero quantized value, wherein the inverse quantization stage ~~(126)~~ is configured to obtain the output signal ~~(150)~~ by using one out of the inverse quantization level ~~(172)~~ and two refined inverse quantization levels ~~(174, 176)~~ indicated by quantized residual value.

15. Decoder ~~(120)~~ according to claim 14, wherein the inverse quantization stage ~~(126)~~ is configured to increase a normalized absolute value of the inverse quantization level ~~(172)~~ by an increase value in order to obtain a first of the two refined inverse quantization levels ~~(174)~~;

wherein the inverse quantization stage ~~(126)~~ is configured to decrease a normalized absolute value of the inverse quantization level ~~(172)~~ by a decrease value in order to obtain a second of the two refined inverse quantization levels ~~(174)~~.

16. Decoder ~~(120)~~ according to one of the claims 14 or 15, wherein the inverse quantization stage ~~(126)~~ is configured to determine the two refined inverse quantization levels ~~(174, 176)~~ for the zero quantized value based on the factor:

fac_z = dz / 3

wherein fac_z indicates a normalized absolute value by which a normalized absolute value of the inverse quantization level ~~(172)~~ is to be increased in order to obtain a first of the two refined inverse quantization levels ~~(174)~~ and a normalized absolute value by which a normalized absolute value of the inverse quantization level ~~(172)~~ is to be decreased in order to obtain a second of the two refined inverse quantization levels ~~(176)~~, and wherein dz is a normalized width of the dead zone.

17. System, comprising:

an encoder ~~(100)~~ according to one of the claims 1 to 8; and

a decoder ~~(120)~~ according to one of the claims 9 to 16.

18. Method ~~(200)~~ for encoding, the method comprising:

quantizing ~~(202)~~ an input signal in order to obtain a plurality of quantized values using a dead zone;

encoding ~~(204)~~ the plurality of quantized values using an entropy encoding scheme in order to obtain a plurality of entropy encoded values;

quantizing ~~(206)~~ a residual signal caused by quantizing the input signal, wherein quantizing ~~(206)~~ the residual signal comprises determining, for a non-zero quantized value, at least one quantized residual value in dependence on a width of the dead zone of the quantization stage; and

forming ~~(208)~~ a bit stream from the plurality of entropy encoded values and the plurality of quantized residual values;

wherein forming ~~(208)~~ the bitstream comprises appending the at least one quantized residual value ~~(146)~~ or a plurality of quantized residual values ~~(146)~~ to the plurality of entropy encoded values ~~(144)~~ until the coded signal ~~(148)~~ comprises a maximum length available for a transfer to a decoder;

wherein quantizing ~~(206)~~ the residual signal comprises determining the at least one quantized residual value ~~(146)~~ such that the quantized residual value ~~(146)~~ comprises 1 bit for a non-zero quantized value, wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ based on the comparison

$$0 \; if \; x[i] < x_Q[i]$$

wherein x[i] is the input signal ~~140~~, wherein x_Q[i] is the inverse quantized input signal ~~(152)~~, and wherein i is an index that is incremented by 1 for each obtained quantized value ~~(142)~~;

wherein quantizing ~~(206)~~ the residual signal comprises determining the quantized residual value ~~(146)~~ such that the quantized residual value ~~(146)~~ comprises 2 bits for a zero quantized value, wherein the residual quantization stage ~~(106)~~ is configured to determine the quantized residual value ~~(146)~~ based on the comparisons

$$|x[i]| < (C \cdot x_Q[i]) \text{ and } x[i] < 0$$

wherein C depends on the width of the dead zone of the quantization stage ~~(102)~~, wherein x[i] is the input signal ~~(140)~~, wherein x_Q[i] is the inverse quantized input signal ~~(152)~~, and wherein i is an index that is incremented by 1 for each obtained quantized value ~~(142)~~;

wherein the at least one quantized residual value ~~(146)~~ is not entropy encoded.

19. Method ~~(220)~~ for decoding, the method comprising:

parsing ~~(222)~~ a coded signal in order to obtain a plurality of entropy encoded values and a quantized residual value;

decoding ~~(224)~~ the plurality of entropy encoded values using an entropy decoding scheme in order to obtain a plurality of quantized values;

inverse quantizing ~~(226)~~ the plurality of quantized values in order to obtain an output signal; and

refining ~~(228)~~ an inverse quantization level used for obtaining the output signal in dependence on a width of a dead zone and the quantized residual value;

wherein refining ~~(228)~~ the inverse quantization level comprises refining an inverse quantization level ~~(172)~~ for a non-zero quantized value in dependence on a quantized residual value ~~(146)~~ and a width of the dead zone.

20. Computer program for performing a method according to claim 18 or 19.