"nlz"

s1_31 int32_t →signed 2's complement: can be shifted max zero bitpos to the left (*)

Get_NumLeadingZeros(abs)	[31]	[30]	[29]	[28]	[27]	[26]	[25]	[24]	[23]	[22]	[21]	[20]	[19]	[18]	[17]	[16]	[15]	[14]	[13]	[12]	[11]	[10]	[9]	[8]	[7]	[6]	[5]	[4]	[3]	[2]	[1]	[0]	signed, int, int32_t
Examples																																	
1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	INT_MAX (2exp31)-1 2147483647 0x7fffffff
31	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
32	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
abs(-1)=1 31	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	-1
abs(INT_MIN) NOT DEFINED	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	INT_MIN -2exp31 -2147483648 0x80000000

The s1_31 sign field takes 1 bit pos (int32_t, signed)
The s9_23 sign field takes 9 bit pos.

sign plus 23 significant value bits = 24 bits = [31]..[8]

SAMPLE FROM MIC @16kHz per sample	S	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0	Mic samples from task_m to task_a (24 topmost bits)

nlz_AGC_normalised

s9_23 want_num_bits_with_sign = 24 (8 LSB zero): can be shifted max 8 bitpos to the left (*)

FOR 32 ms NOT FULL RANGE	S	s	s	s	s	s	s	s	S	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0	

want_num_leading_zeros_with_sign 9

6-9=-3: immediate attack with nlz_AGC_to_normal=(-3) is ">>3 "

got_num_bits_with_sign = 27 (8 LSB zero)

"LESS" VOLUME WANTED IMMEDIATE ATTACK	S	s	s	s	s	s	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0	>>3 of all for 32 ms

num_leading_zeros_with_sign_now 6

(11-9)=2: Start slow decay with 0 to nlz_AGC_to_normal=2 is "<<2"

got_num_bits_with_sign = 22 (8 LSB zero)

"MORE" VOLUME WANTED SLOW DECAY POSSIBLE	S	s	s	s	s	s	s	s	s	s	s	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0	<<2 of all for 32 ms

num_leading_zeros_with_sign_now 11

got_num_bits_with_sign = 32

"LESS" VOLUME WANTED IMMEDIATE ATTACK	S	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	0	0	0	0	0	0	0	0	>>8 of all for 32 ms

1-9=-8: immediate attack with nlz_AGC_to_normal=(-8) is ">>8 "

The Q (number format) like q4.24 (1 sign bit, 3 integer bits and 24 fraction bits) contains integer info even after "shifting the sign in and down". The magic of two's complement is that those very "sign bits" except for the leftmost will indeed also have a say for the integer value, for negative values meaning zero * (-2exp(n)-1). This goes for all Q formats, from q1_31 (int or signed) and "down" in formats. If "sign" bits =1 they represent that something is not present, that's why they can have this "double meaning". Example:

0x80000000 (q4.28 -128.000000 signed -2147483648) (INT_MIN), divided by 256 or shifted right 8 bits is
0xFF800000 (q4.28 -0.500000 signed -8388608) with 9 "sign bits" as well as 1 sign bit + 8 value bits.

0xBFFFFFFF (q4.28 -64.000000 signed -1073741825) has first integer bit zero, meaning present (-2exp(30))-1=-1073741824-1
0xFFFFFFFF (signed -1) has 31 "sign bits" and 1 integer bit

4-9=-5: immediate attack with nlz_AGC_to_normal=(-5) is ">>5 "

s4_28 -0.9986839 most negative 11110000 00000101 01100100 00000000: can be shifted max 3 bitpos to the left (*)

Sine in "audio_sine_table.h"	1	1	1	0	0	0	0	0	0	0	0	0	1	0	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	0	0	0	>>5 of all for 32 ms

s4_28 0.9986839 most positive 00001111 11111010 10011100 00000000: can be shifted max 3 bitpos to the left (*)

--"--	0	0	0	1	1	1	1	1	1	1	1	1	0	1	0	1	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	>>5 of all for 32 ms

All entries end in "00,", so it looks like this has been generated as s12_20 and shifted left 8 bitpos

(*) See "test_signed_shift_left_overflow"

2024 05 17 signed int int32_t twos complement overview and scaling of mic for FFT.pages by Øyvind Teig