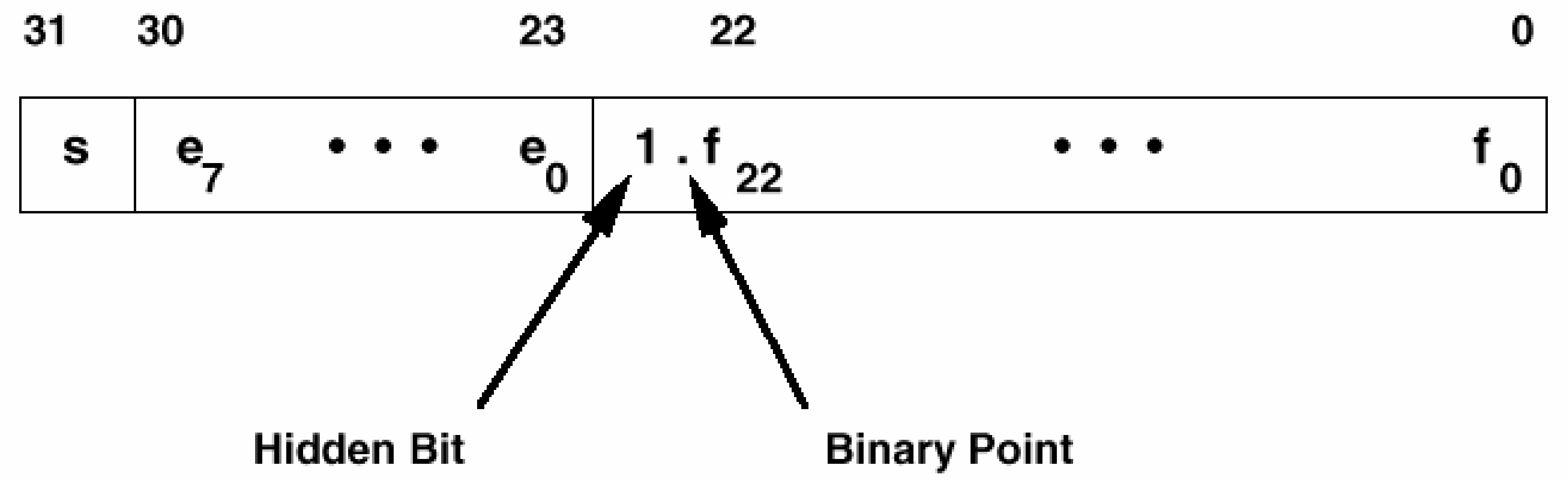
ADSP 2106x

Zahlenformate
Auszüge aus
ADSP-2106x Sharc Users Manual
Analog Devices, Inc.

Übersicht Zahlenformate

IEEE SINGLE-PRECISION FLOATING-POINT DATA FORMAT



Übersicht Zahlenformate

IEEE SINGLE-PRECISION FLOATING-POINT DATA FORMAT

- **The unsigned exponent e can range between 1 ≤ e ≤ 254 for normal numbers in the single-precision format.**
- **This exponent is biased by +127 (254 : 2).**
- **To calculate the true unbiased exponent, 127 must be subtracted from e.**

Übersicht Zahlenformate

IEEE SINGLE-PRECISION FLOATING-POINT DATA FORMAT

The IEEE Standard also provides for several special data types in the single-precision floating-point format:

- **An exponent value of 255 (all ones) with a nonzero fraction is a Not-A-Number (NAN). NANs are usually used as flags for data flow control, for the values of uninitialized variables, and for the results of invalid operations such as 0 * ∞**
- **Infinity is represented as an exponent of 255 and a zero fraction. Note that because the fraction is signed, both positive and negative Infinity can be represented.**
- **Zero is represented by a zero exponent and a zero fraction. As with Infinity, both positive Zero and negative Zero can be represented.**

Übersicht Zahlenformate

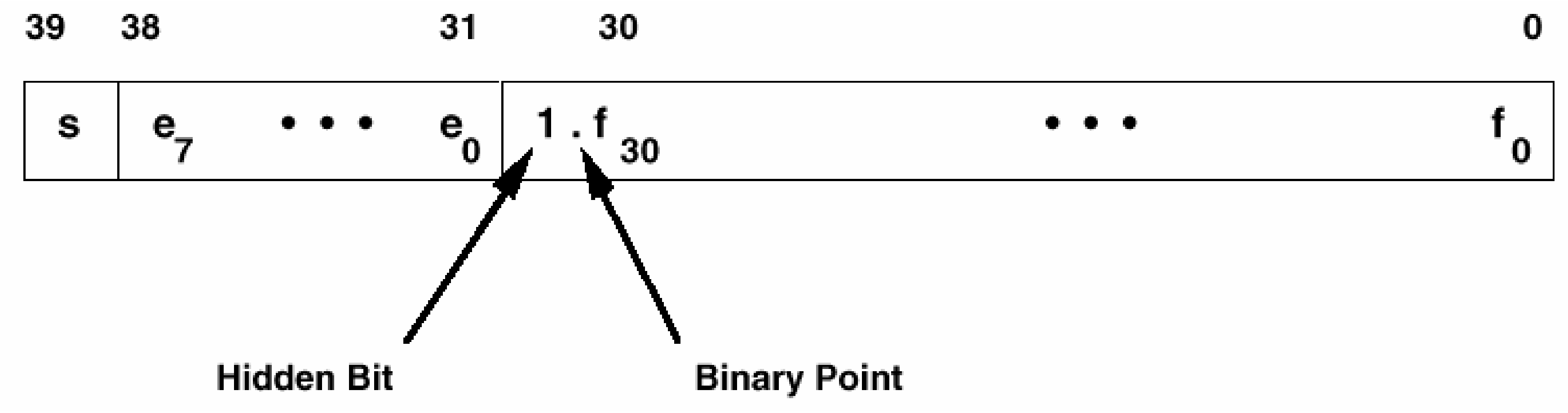
IEEE SINGLE-PRECISION FLOATING-POINT DATA FORMAT

The IEEE single-precision floating-point data types supported by the ADSP-2106x and their interpretations are :

Type	Exponent	Fraction	Value
NAN	255	Nonzero	Undefined
Infinity	255	0	(–1)s Infinity
Normal	1 ≤ e ≤ 254	Any	(–1)s (1.f 22-0) 2 e–127
Zero	0	0	(–1)s Zero

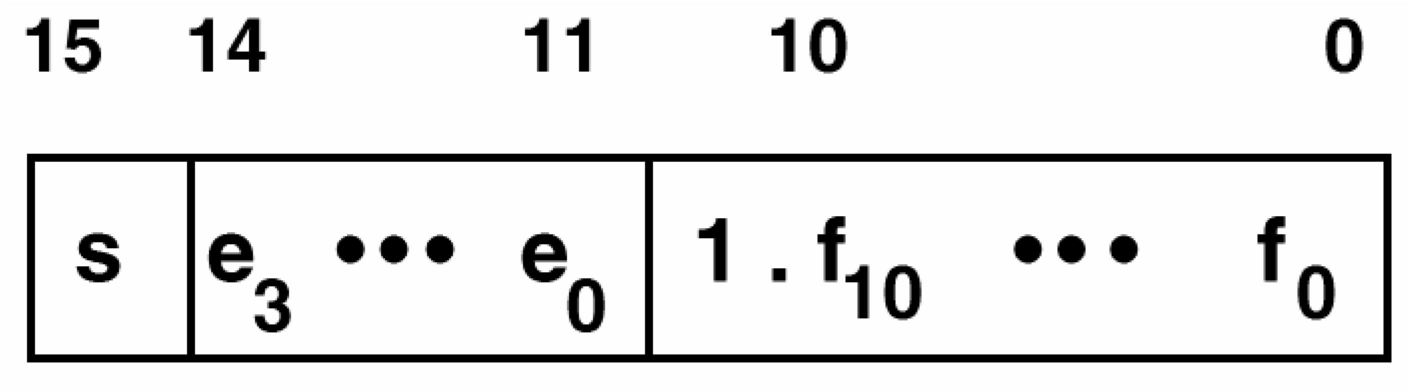
Übersicht Zahlenformate

EXTENDED PRECISION FLOATING-POINT FORMAT



Übersicht Zahlenformate

SHORT WORD FLOATING-POINT FORMAT



15	14	11	10	0
s	e_3 ••• e_0		$1 . f_{10}$ ••• f_0	

Übersicht Zahlenformate

- The ADSP-2106x supports a 16-bit floating-point data type and provides conversion instructions for it.
- The short float data format has an 11-bit mantissa with a four-bit exponent plus sign bit, as shown
- The 16-bit floating-point numbers reside in the lower 16 bits of the 32-bit floating-point field.

Übersicht Zahlenformate

Two shifter instructions, FPACK and FUNPACK, perform the packing and unpacking conversions between 32-bit floating-point words and 16-bit floating-point words.

Übersicht Zahlenformate

- The FPACK instruction converts a 32-bit IEEE floating-point number to a 16-bit floating-point number.
- FUNPACK converts the 16-bit floating-point numbers back to 32-bit IEEE floating-point.

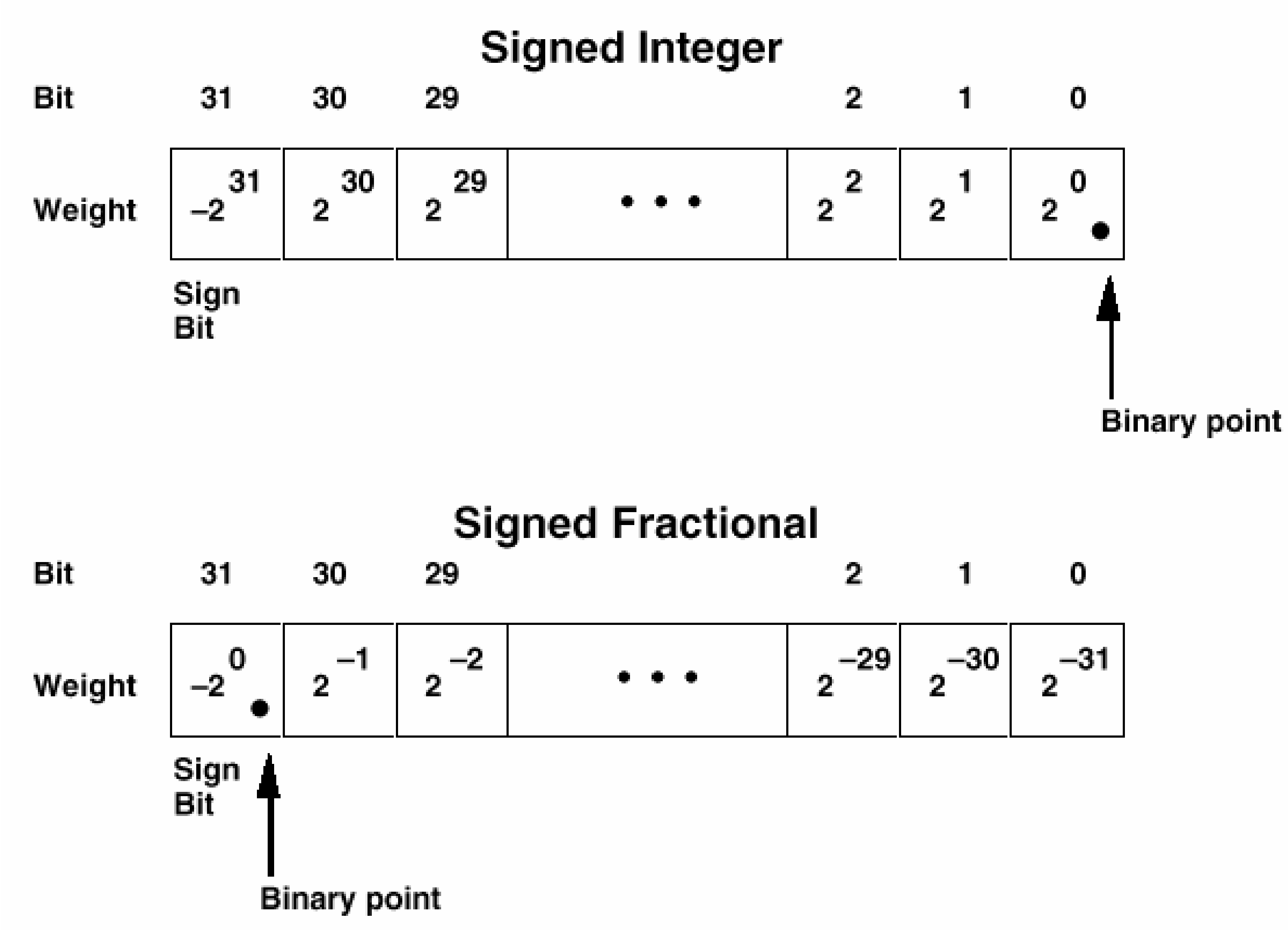
Übersicht Zahlenformate

FIXED-POINT FORMATS

The ADSP-2106x supports two 32-bit fixed-point formats:

- **signed and unsigned fractional and**
- **signed and unsigned integer.**

Übersicht Zahlenformate



Signed Integer

Bit	31	30	29		2	1	0
Weight	-2^{31}	2^{30}	2^{29}	• • •	2^{2}	2^{1}	2^{0} ●

Sign Bit

Binary point

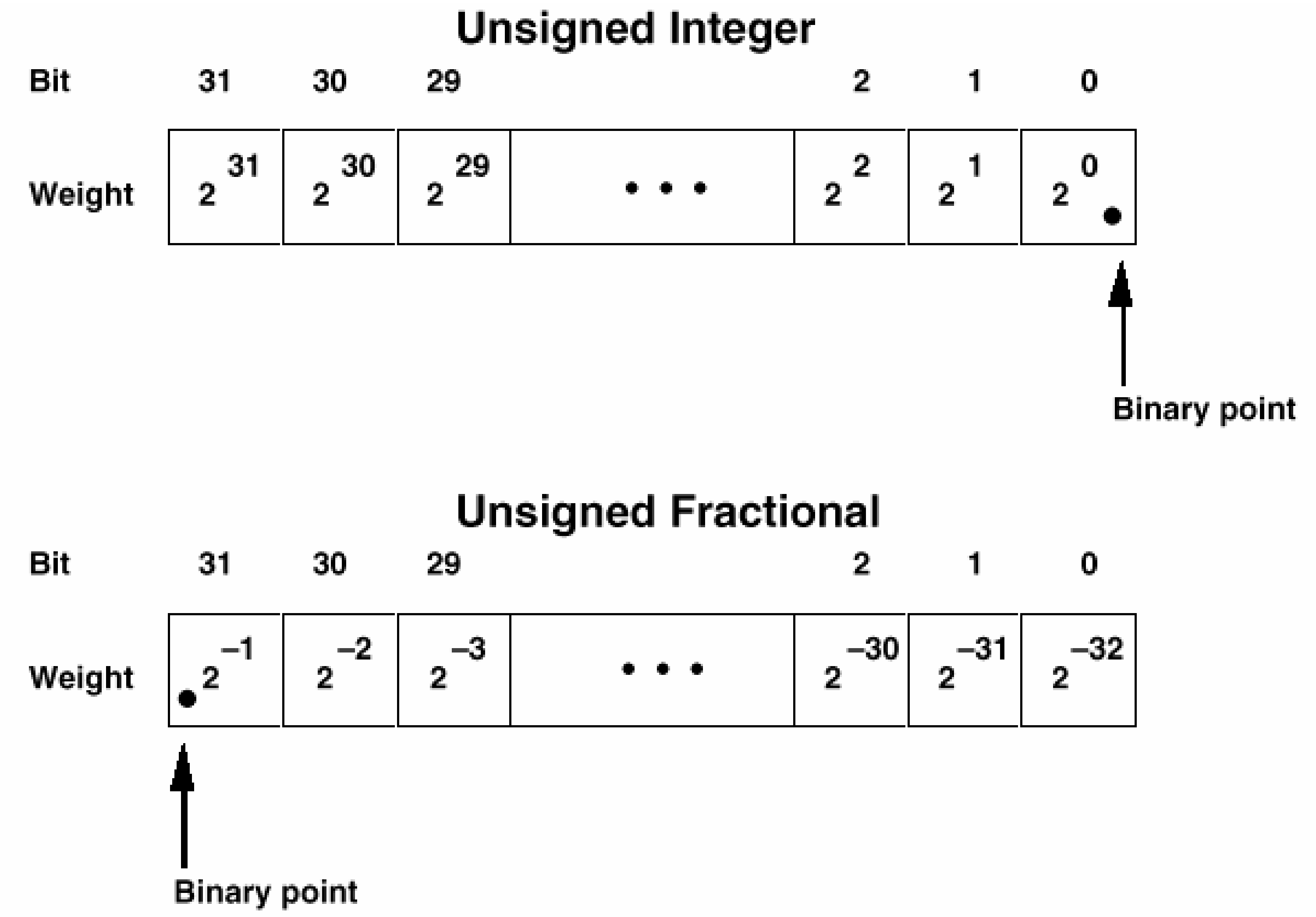
Signed Fractional

Bit	31	30	29		2	1	0
Weight	-2^{0} ●	2^{-1}	2^{-2}	• • •	2^{-29}	2^{-30}	2^{-31}

Sign Bit

Binary point

Übersicht Zahlenformate



Unsigned Integer

Bit	31	30	29		2	1	0
Weight	2^{31}	2^{30}	2^{29}	• • •	2^{2}	2^{1}	2^{0} ●

Binary point

Unsigned Fractional

Bit	31	30	29		2	1	0
Weight	● 2^{-1}	2^{-2}	2^{-3}	• • •	2^{-30}	2^{-31}	2^{-32}

Binary point

Übersicht Zahlenformate

FIXED-POINT FORMATS

- **ALU outputs always have the same width and data format as the inputs.**

Übersicht Zahlenformate

FIXED-POINT FORMATS

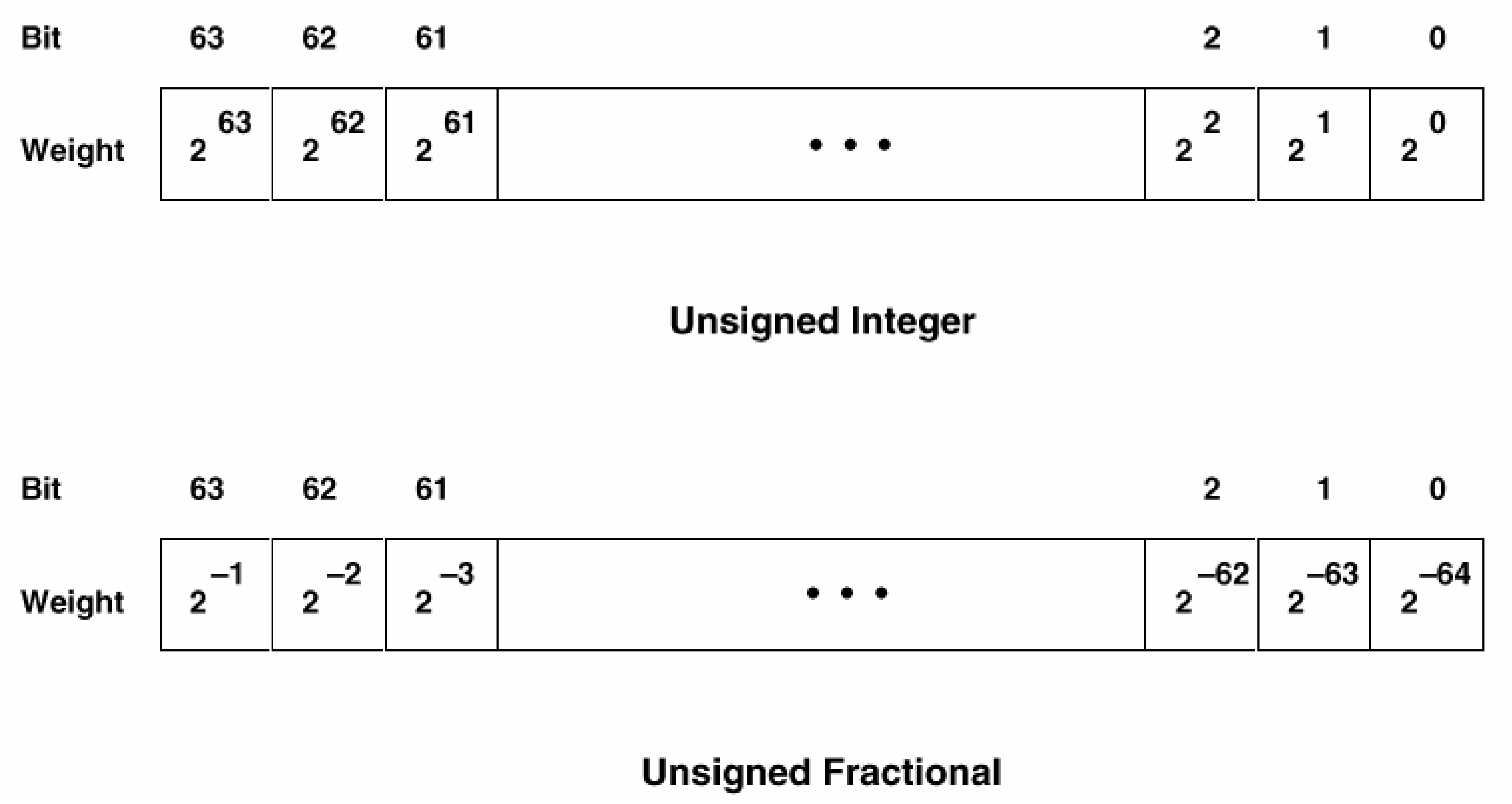
- **The multiplier produces a 64-bit product from two 32-bit inputs.**
- **If both operands are unsigned integers, the result is a 64-bit unsigned integer.**
- **If both operands are unsigned fractions, the result is a 64-bit unsigned fraction.**

Übersicht Zahlenformate

FIXED-POINT FORMATS

- **If one operand is signed and the other unsigned, the result is signed.**
- **If both inputs are signed, the result is signed and automatically shifted left one bit.**
- **The LSB becomes zero and bit 62 moves into the sign bit position.**
- **Normally bit 63 and bit 62 are identical when both operands are signed.**

Übersicht Zahlenformate



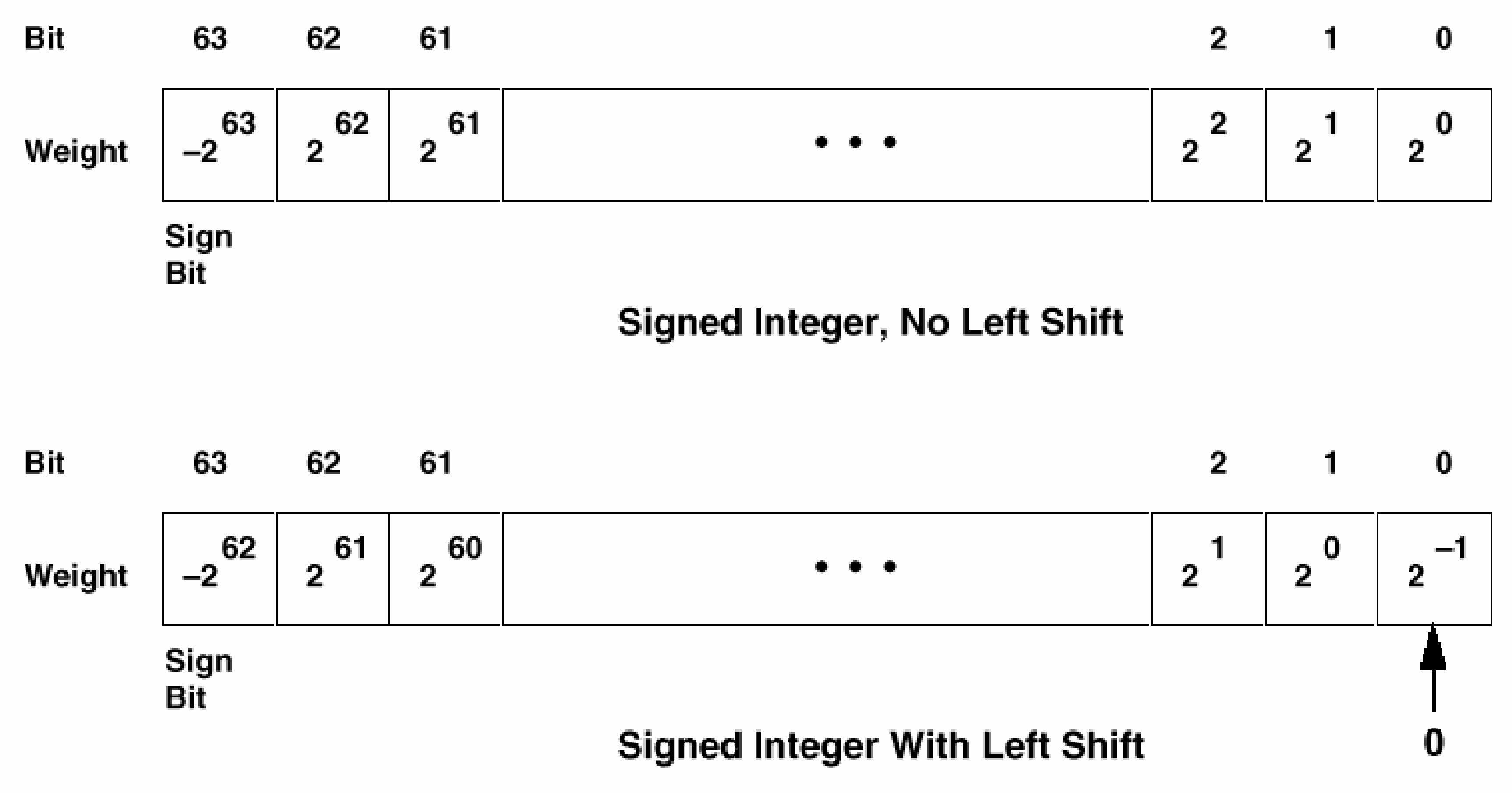
Bit	63	62	61		2	1	0
Weight	2^{63}	2^{62}	2^{61}	• • •	2^{2}	2^{1}	2^{0}

Unsigned Integer

Bit	63	62	61		2	1	0
Weight	2^{-1}	2^{-2}	2^{-3}	• • •	2^{-62}	2^{-63}	2^{-64}

Unsigned Fractional

Übersicht Zahlenformate



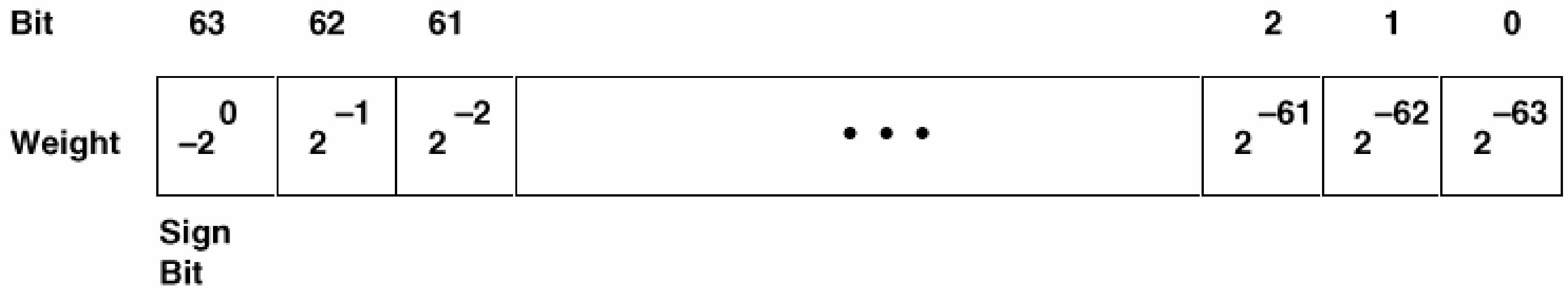
Bit	63	62	61		2	1	0
Weight	-2^{63}	2^{62}	2^{61}	• • •	2^{2}	2^{1}	2^{0}
	Sign Bit						

Signed Integer, No Left Shift

Bit	63	62	61		2	1	0
Weight	-2^{62}	2^{61}	2^{60}	• • •	2^{1}	2^{0}	2^{-1}
	Sign Bit						0

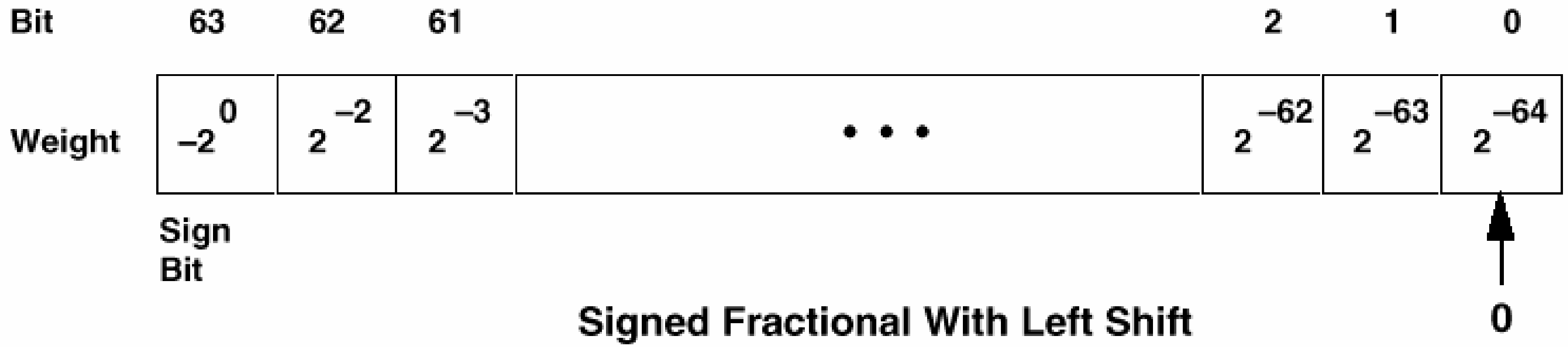
Signed Integer With Left Shift

Übersicht Zahlenformate



Bit	63	62	61		2	1	0
Weight	-2^{0}	2^{-1}	2^{-2}	• • •	2^{-61}	2^{-62}	2^{-63}
	Sign Bit						

Signed Fractional, No Left Shift



Bit	63	62	61		2	1	0
Weight	-2^{0}	2^{-2}	2^{-3}	• • •	2^{-62}	2^{-63}	2^{-64}
	Sign Bit						0

Signed Fractional With Left Shift