Floating Points Representation

Example: Convert 44.703125_{10} to a floating point.

I. Convert whole number 44_{10} to a binary number.

128	64	32	16	8	4	2	1
0	0	1	0	1	1	0	0

$$44_{10} = 101100_2$$

II. Convert the fraction 0.703125_{10} to a binary number.

Fraction part × 2	Whole number
$0.703125 \times 2 = 1.40625$	1
$0.40625 \times 2 = 0.8125$	0
$0.8125 \times 2 = 1.625$	1
$0.625 \times 2 = 1.25$	1
$0.25 \times 2 = 0.5$	0
$0.5 \times 2 = 1.0$	1
0 until zero	0.101101_2

III. Combine two parts:

$$44.703125_{10} = 101100.101101_2$$

IV. Write into Scientific Notation:

$$101100.101101_2 = 1.01100101101_2 \times 2^5$$
$$= 1.01100101101_2 \times 2^{101_2}$$

V. Write the number $1.01100101101_2 \times 2^{101_2}$ into floating points.

a. Number's sign is positive = 0

b. Exponent's sign is positive = 0

c. 7 bits exponent will be 0000101

d. 23 bits fraction will be 01100101101000000000000

VI. Conclusion: 44.703125_{10} will be stored as

000000101 01100101101000000000000

Sign bits + exponent + fraction

Example 2: Convert 0.8_{10} to a binary number.

Fraction part $\times$ 2	Whole number
$0.8 \times 2 = 1.6$	1
$0.6 \times 2 = 1.2$	1
$0.2 \times 2 = 0.4$	0
$0.4 \times 2 = 0.8$	0
$0.8 \times 2 = 1.6$	1
$0.6 \times 2 = 1.2$	1
$0.2 \times 2 = 0.4$	0
$0.4 \times 2 = 0.8$	0
	…
The answer will be repeated	$0.11001100\ldots_2$

Example 3: Convert 0.101101_2 to a decimal number.

$$
\begin{aligned}
1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3} + 1 \times 2^{-4} + 0 \times 2^{-5} + 1 \times 2^{-6} \\
= 0.5 + \frac{1}{8} + \frac{1}{16} + \frac{1}{64} \\
= 0.5 + 0.125 + 0.0625 + 0.015625 = 0.703125
\end{aligned}
$$

Question:

Why can we not get an exact float point for 0.1?