

Introduction to Computer Science—Honors I

CS181—Fall 2008

Instructor: Antonio R. Nicolosi

10 September 2008

Handout—A Toy 16-Bit Floating-Point Format

[Adapted from a description of the IEEE-754 standard in “Floating Point Numbers in Java,” by Michael L. Overton.]

\pm	$a_1 a_2 a_3 a_4 a_5$	$b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10}$
-------	-----------------------	--

[Sign bit + 5-bit exponent in ‘punctured’ excess-15 signed-notation + 10-bit mantissa.]

If exponent bitstring $a_1 a_2 a_3 a_4 a_5$ is	Then numerical value represented is
$(00000)_2 = (0)_{10}$	$\pm(0.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^{-14}$
$(00001)_2 = (1)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^{-14}$
$(00010)_2 = (2)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^{-13}$
$(00011)_2 = (3)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^{-12}$
↓	↓
$(01111)_2 = (15)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^0$
$(10000)_2 = (16)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^1$
↓	↓
$(11100)_2 = (28)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^{13}$
$(11101)_2 = (29)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^{14}$
$(11110)_2 = (30)_{10}$	$\pm(1.b_1 b_2 b_3 b_4 b_5 b_6 b_7 b_8 b_9 b_{10})_2 \times 2^{15}$
$(11111)_2 = (31)_{10}$	$\pm\infty$ if $b_1 = \dots = b_{10} = 0$, NaN otherwise

[‘Exponent *vs.* represented value’ chart.]

Format	Mantissa length	Exponent length	Bias (‘excess- <i>N</i> ’)
16-bit toy format	10	5	15
32-bit IEEE single format (Java float)	23	8	127
64-bit IEEE double format (Java double)	52	11	1023

[Comparison of the floating-point formats discussed in class.]