

Trabalho Final - SSC0109

Descrição

Implemente um hardware para calcular uma das funções matemáticas presentes no final deste documento:

- Entrada:
 - X[7..0]: número real na representação em ponto-fixa, diferente de zero quando aplicável (por exemplo em funções onde X é usado como divisor).
- Saída:
 - Y[7..0]: resultado da função, número real na representação em ponto-fixa.
- **Número real: pode ser positivo ou negativo (utilizar complemento de 2)!**

Ponto-fixa

Neste trabalho, utilizaremos a representação em ponto-fixa, onde é possível representar números reais utilizando um número fixo de bits para parte inteira e fracionária. Utilizaremos 4 bits para parte inteira (X[7..4] e Y[7..4]) e 4 bits para parte fracionária (X[3..0] e Y[3..0]).

Da mesma maneira que o bit menos significativo inteiro vale 2^0 , o segundo menos significativo vale 2^1 e assim por diante, o primeiro bit fracionário (mais à esquerda) vale 2^{-1} , o próximo 2^{-2} e assim sucessivamente. Para números negativos, o complemento de 2 funciona igual a números inteiros.

Exemplos de números nesta representação:

- $5.000 = 01010000 = 2^2 + 2^0$
- $7.500 = 01111000 = 2^2 + 2^1 + 2^0 + 2^{-1}$
- $2.250 = 00100100 = 2^1 + 2^{-2}$
- $0.625 = 00001010 = 2^{-1} + 2^{-3}$
- $-7.625 = 10000110$

Para este trabalho, os módulos aritméticos deverão ser expandidos:

- Somador/subtrator: expandir de 4 para 8 bits. Exemplo de operação:
 - $4.5 + 2.5 \rightarrow 01001000 + 00101000 \rightarrow 01110000 \rightarrow 7$
- Multiplicador: expandir as entradas para 8 bits (a saída terá 16 bits). Dos 16 bits da saída, realizar um shift de 4 bits para direita. Exemplo de operação:
 - $2.5 * 2.5 \rightarrow 00101000 * 00101000 \rightarrow 110010000000$ (shift) $\rightarrow 01100100 \rightarrow 6.25$
- Divisor: expandir as entradas para 16 bits (a saída terá 8 bits). O valor do primeiro operando (dividendo) deverá ter um shift para a esquerda de 4 bits. Exemplo de operação:
 - $2.5 / 2 \rightarrow 0000000000101000 / 0000000000100000 \rightarrow 0000001010000000 / 0000000000100000$ (shift no dividendo) $\rightarrow 00010100 \rightarrow 1.25$

Caso haja algum operando negativo na operação de multiplicação ou divisão, convertê-los para positivo, realizar a operação e ajustar o sinal no resultado:

- Se os dois operandos forem positivos, a saída é positiva
- Se os dois operandos forem negativos, a saída é positiva
- Se os dois operandos diferem em seus sinais, a saída é negativa

Exemplo de divisão onde um dos operandos é negativo:

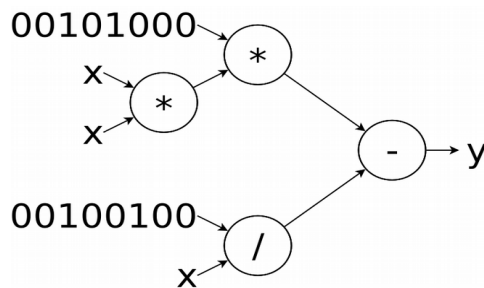
- $-4.5 / 1.5 \rightarrow 0000000001001000$ (convertido para positivo) / 000000000011000
 $\rightarrow 0000010010000000 / 000000000011000$ (shift no dividendo) $\rightarrow 00110000 \rightarrow$
 11010000 (convertido para negativo) $\rightarrow -3$

Observações

- Entre na wiki da disciplina e veja quais funções já foram escolhidas. Escolha uma função que não tenha sido escolhida e mande um e-mail para make.vhdl.not.war@gmail.com no formato:
 - Assunto: Projeto SSC-109 2017/1 Ter/Qui
 - Conteúdo: Nome e NUSP dos integrantes da dupla e a função escolhida
- Caso algum ponto da função cause overflow, este deve ser indicado por um LED
- Os trabalhos serão avaliados no horário de aula:
 - Terça: 27/06
 - Quinta: 29/06
- Dupla de 2 alunos (também conhecido como dupla)
- Colar = 0
- Não é possível mudar de equação após o e-mail ter sido enviado e confirmado!
- O projeto será executado nas placas (utilizar os switches para entrada de dados e LEDs ou displays de sete segmentos para o resultado)

Exemplo

Considere a equação: $f(x) = 2.5x^2 - 2.25/x$. O hardware para esta função:



Aplicativo

Ao utilizar 4 bits para parte fracionária no ponto fixo, a precisão é baixa. Portanto, o seu hardware possivelmente gerará resultados bem diferentes dos esperados em uma calculadora. Para comparar os resultados, será disponibilizado na wiki uma ferramenta denominada *evaluator*. Este programa faz cálculos com precisão similar ao do hardware. Para utilizá-lo:

- Instale Python em sua máquina, caso já não tenha
- Para avaliar $x = 0.5$ na função $f(x) = ax^b + cx^d$, execute:
 - `python evaluator.py 0 a b c d -1.75`
- A saída será algo similar a:
 - DEC:
 - $(-3.5x^{-3.0}) + (3.0625x^{1.0})$
 - $f(-1.75) = -4.6875$
 - BIN:
 - $(11001000x^{11010000}) + (110001x^{10000})$
 - $f(11100100) = 10110101$
- O primeiro argumento é a precedência em parcelas da soma com expoentes negativos. Como a precisão do hardware é baixa, isto gera pequenas diferenças no resultado final. Os modos disponíveis são (considerando $b < 0$):
 - 0: $(1/x^b) * a$
 - 1: a/x^b

Funções

1. $(2.0625x^{2.0}) + (-2.8125x^{-3.0})$
2. $(-3.8125x^{3.0}) + (3.25x^{-1.0})$
3. $(-3.375x^{1.0}) + (2.25x^{-2.0})$
4. $(-2.375x^{2.0}) + (2.125x^{-3.0})$
5. $(3.1875x^{-2.0}) + (-2.125x^{1.0})$
6. $(2.5625x^{2.0}) + (-3.625x^{-3.0})$
7. $(-2.0625x^{-2.0}) + (2.4375x^{1.0})$
8. $(2.5625x^{3.0}) + (-3.0x^{-1.0})$
9. $(-2.5x^{-2.0}) + (2.5x^{3.0})$
10. $(3.0x^{-1.0}) + (-3.25x^{3.0})$
11. $(3.75x^{3.0}) + (-2.0x^{-1.0})$
12. $(3.1875x^{2.0}) + (-2.3125x^{-1.0})$
13. $(2.1875x^{-2.0}) + (-3.625x^{1.0})$
14. $(-3.0x^{1.0}) + (2.25x^{-3.0})$
15. $(-3.875x^{3.0}) + (2.125x^{-1.0})$
16. $(2.5625x^{1.0}) + (-2.25x^{-3.0})$
17. $(-2.875x^{3.0}) + (2.25x^{-1.0})$
18. $(2.75x^{-1.0}) + (-3.5x^{2.0})$
19. $(-2.375x^{-1.0}) + (2.0x^{3.0})$
20. $(-2.5625x^{-1.0}) + (2.125x^{2.0})$
21. $(2.875x^{1.0}) + (-3.0625x^{-2.0})$
22. $(-3.4375x^{1.0}) + (3.0625x^{-3.0})$
23. $(2.5625x^{2.0}) + (-3.4375x^{-1.0})$
24. $(-2.3125x^{-3.0}) + (3.5625x^{2.0})$
25. $(3.875x^{1.0}) + (-2.875x^{-2.0})$
26. $(-2.125x^{3.0}) + (2.375x^{-1.0})$
27. $(-2.5x^{2.0}) + (2.125x^{-1.0})$
28. $(-2.5x^{3.0}) + (2.125x^{-1.0})$
29. $(-3.4375x^{-2.0}) + (3.4375x^{1.0})$
30. $(-2.375x^{2.0}) + (2.625x^{-3.0})$
31. $(2.5x^{-3.0}) + (-2.625x^{1.0})$
32. $(-3.1875x^{1.0}) + (3.5x^{-3.0})$
33. $(-3.0625x^{-3.0}) + (2.0625x^{1.0})$
34. $(-2.4375x^{3.0}) + (2.0x^{-1.0})$
35. $(-2.0625x^{2.0}) + (3.625x^{-1.0})$
36. $(-2.3125x^{3.0}) + (2.5625x^{-2.0})$
37. $(-2.1875x^{-2.0}) + (3.8125x^{1.0})$
38. $(-2.375x^{-1.0}) + (2.8125x^{2.0})$
39. $(2.8125x^{-3.0}) + (-3.1875x^{1.0})$
40. $(2.0625x^{1.0}) + (-2.375x^{-3.0})$
41. $(-2.3125x^{-1.0}) + (3.8125x^{3.0})$
42. $(3.3125x^{-1.0}) + (-3.875x^{2.0})$
43. $(-3.0625x^{-1.0}) + (3.75x^{2.0})$
44. $(-2.0x^{3.0}) + (3.0x^{-1.0})$
45. $(2.75x^{1.0}) + (-3.5x^{-3.0})$
46. $(2.0x^{-1.0}) + (-2.125x^{3.0})$
47. $(-2.6875x^{-2.0}) + (2.6875x^{1.0})$
48. $(2.25x^{1.0}) + (-2.5x^{-3.0})$
49. $(-3.0x^{1.0}) + (2.625x^{-3.0})$
50. $(2.0625x^{-1.0}) + (-2.8125x^{2.0})$
51. $(-2.4375x^{3.0}) + (2.125x^{-2.0})$
52. $(2.8125x^{1.0}) + (-2.625x^{-3.0})$
53. $(3.3125x^{-2.0}) + (-3.25x^{1.0})$
54. $(-3.875x^{1.0}) + (3.5625x^{-3.0})$

55. $(2.1875x^{2.0}) + (-3.4375x^{-1.0})$
56. $(-2.875x^{-2.0}) + (2.1875x^{1.0})$
57. $(-2.3125x^{3.0}) + (3.6875x^{-1.0})$
58. $(-3.625x^{1.0}) + (3.0x^{-3.0})$
59. $(-3.5625x^{-3.0}) + (3.125x^{1.0})$
60. $(2.5x^{-1.0}) + (-3.75x^{3.0})$
61. $(2.375x^{2.0}) + (-2.375x^{-1.0})$
62. $(-3.3125x^{1.0}) + (2.5x^{-3.0})$
63. $(-3.625x^{-3.0}) + (2.3125x^{1.0})$
64. $(2.3125x^{2.0}) + (-3.375x^{-3.0})$
65. $(3.375x^{1.0}) + (-2.0625x^{-2.0})$
66. $(-3.5x^{1.0}) + (3.375x^{-2.0})$
67. $(-3.75x^{-3.0}) + (2.1875x^{1.0})$
68. $(2.0625x^{2.0}) + (-2.0625x^{-1.0})$
69. $(-2.75x^{-1.0}) + (3.1875x^{2.0})$
70. $(3.3125x^{-2.0}) + (-3.4375x^{3.0})$
71. $(3.0625x^{-1.0}) + (-2.0625x^{3.0})$
72. $(2.0x^{-1.0}) + (-3.125x^{2.0})$
73. $(-2.875x^{1.0}) + (3.4375x^{-2.0})$
74. $(2.3125x^{1.0}) + (-2.9375x^{-3.0})$
75. $(-3.625x^{-2.0}) + (2.25x^{3.0})$
76. $(-2.6875x^{-1.0}) + (3.125x^{2.0})$
77. $(2.75x^{-2.0}) + (-2.8125x^{1.0})$
78. $(2.5625x^{3.0}) + (-2.9375x^{-1.0})$
79. $(2.25x^{-1.0}) + (-3.5x^{3.0})$
80. $(-2.3125x^{3.0}) + (3.25x^{-1.0})$
81. $(3.0625x^{3.0}) + (-2.0x^{-2.0})$
82. $(3.375x^{-1.0}) + (-3.875x^{2.0})$
83. $(3.6875x^{1.0}) + (-2.375x^{-2.0})$
84. $(3.6875x^{-3.0}) + (-2.625x^{2.0})$
85. $(2.1875x^{1.0}) + (-2.0x^{-3.0})$
86. $(-2.9375x^{2.0}) + (2.0625x^{-1.0})$
87. $(-3.5x^{1.0}) + (3.625x^{-2.0})$
88. $(2.3125x^{1.0}) + (-3.3125x^{-3.0})$
89. $(-3.0625x^{1.0}) + (2.0x^{-2.0})$
90. $(2.1875x^{-1.0}) + (-2.0x^{3.0})$
91. $(3.875x^{-1.0}) + (-3.125x^{3.0})$
92. $(-3.5625x^{1.0}) + (3.1875x^{-3.0})$
93. $(2.125x^{1.0}) + (-3.375x^{-3.0})$
94. $(-3.375x^{1.0}) + (2.375x^{-2.0})$
95. $(2.9375x^{-2.0}) + (-2.6875x^{1.0})$
96. $(-2.0x^{3.0}) + (2.9375x^{-2.0})$
97. $(-2.625x^{-3.0}) + (2.625x^{1.0})$
98. $(2.875x^{-1.0}) + (-2.3125x^{2.0})$
99. $(2.0x^{2.0}) + (-2.8125x^{-3.0})$
100. $(3.0625x^{-3.0}) + (-2.625x^{1.0})$