

Números em Ponto Fixo e Ponto Flutuante

Disciplina: Introdução à Arquitetura de Computadores

Luciano Moraes Da Luz Brum

Universidade Federal do Pampa – Unipampa – Campus Bagé

Email: lucianobrum18@gmail.com

Tópicos

- Números em Ponto Fixo;
- Operações aritméticas com números em Ponto Fixo;
- Números em Ponto Flutuante;
- Operações aritméticas com números em Ponto Flutuante;
- Resumo

Números em Ponto Fixo

- Os números tratados até aqui não possuíam vírgula explicitamente;
- Utilizando-se bits para representar frações, diminuem-se os bits para representar inteiros;
- A notação em ponto fixo determina quantos bits são usados para representar a parte inteira e quantos bits são usados para representar a parte fracionária;

Números em Ponto Fixo

➤ Para a mesma cadeia de bits, tem-se:

0	1	1	0	0	1	1,	1 ₂	Decimal = 51,5 ₁₀ (7 bits para inteiro, 1 bit para fração)
0	1	1	0	0	1,	1	1 ₂	Decimal = 25,75 ₁₀ (6 bits para inteiro, 2 bit para fração)
0	1	1	0	0,	1	1	1 ₂	Decimal = 12,875 ₁₀ (5 bits para inteiro, 3 bit para fração)
0	1	1,	0	0	1	1	1 ₂	Decimal = 3,21875 ₁₀ (3 bits para inteiro, 5 bit para fração)
0,	1	1	0	0	1	1	1 ₂	Decimal = 0,8046875 ₁₀ (1 bit para inteiro 7 bit para fração)

Números em Ponto Fixo

- A vírgula não é representada explicitamente;
- Dos 'n' bits: 't' bits ($t \geq 0$) são usados para parte inteira e 'f' bits ($f \geq 0$) são usados para parte fracionária ($t + f = n$);
- A quantidade total de valores representáveis segue a mesma, independente da posição da vírgula (2^n);

Números em Ponto Fixo

- A faixa de valores depende da posição da vírgula;
- A fórmula é a mesma utilizada para complemento de B e B-1, sinal-magnitude e inteiros positivos, porém, todos divididos por um fator de 2^f ;
- Ex: Complemento de B $\rightarrow \left[\frac{-2^{n-1}}{2^f}, \frac{(2^{n-1}-1)}{2^f} \right]$;

Números em Ponto Fixo

- Os números fracionários estão separados entre si por uma diferença de 2^{-f} ;

n	t	f	Qtia de n°s	Menor n°	Maior n°	Intervalo
8	8	0	256	-128	127	1
8	7	1	256	-64	63,5	0,5
8	6	2	256	-32	31,75	0,25
8	5	3	256	-16	15,875	0,125
8	4	4	256	-8	7,9375	0,0625
8	3	5	256	-4	3,96875	0,03125
8	2	6	256	-2	1,984375	0,015625
8	1	7	256	-1	0,9921875	0,0078125
8	0	8	256	-0,5	0,49609375	0,00390625

Tabela 1: Possibilidades de números em ponto fixo de 8 bits. Fonte: Adaptado de Weber, 2001.

Tópicos

- Números em Ponto Fixo;
- Operações aritméticas com números em Ponto Fixo;
- Números em Ponto Flutuante;
- Operações aritméticas com números em Ponto Flutuante;
- Resumo;

Operações com Números em PF

- Soma e subtração:
- Efetua-se da mesma maneira que para números inteiros, porém os números devem ter a mesma posição para a vírgula ('t' e 'f' iguais para todos operandos);
- Caso tenham diferentes posições, converte-se um dos números para a representação do outro;
- Ex: $m1 (t1 \text{ e } f1) + m2 (t2 \text{ e } f2) = m3 (t1 \text{ e } f1 \text{ ou } t2 \text{ e } f2)$

Operações com Números em PF

- 1º Caso: 't1' > 't2' (t1 parte inteira de m1 e t2 parte inteira de m2).
 - Parte Inteira: t2 deve ser **estendido** para t1 bits, mantendo **sinal** e valor do **número**.
 - Para Inteiros-positivos, (t1-t2) zeros são colocados à esquerda de m2;
 - Para números em complemento de B, o sinal deve ser duplicado à esquerda por (t1-t2) bits;
- Parte Fracionária: f2 deve ser **reduzida** para f1 bits.
 - Truncagem: f2-f1 bits a direita de m2 são eliminados;
 - Arredondamento: somar $2^{-(f_1+1)}$ a m2 e após, truncagem;

Operações com Números em PF

- Na tabela 2, considere que o número final deve ser representado em **4 bits de parte inteira e 4 bits de parte fracionária**;

Nº Original	Representação	Parte Inteira (t)	Fração (T)	Fração (A)
01,101101 ₂	Inteiro-Positivo	0001 ₂	1011 ₂	1011 ₂
11,101101 ₂	Inteiro-Positivo	0011 ₂	1011 ₂	1011 ₂
010,00111 ₂	Inteiro-Positivo	0010 ₂	0011 ₂	0100 ₂
01,101101 ₂	Complemento de dois	0001 ₂	1011 ₂	1011 ₂
11,101101 ₂	Complemento de dois	1111 ₂	1011 ₂	1011 ₂
010,00111 ₂	Complemento de dois	0010 ₂	0011 ₂	0100 ₂

Tabela 2: Exemplos de redução de fração. Fonte: Adaptado de Weber, 2001.

Operações com Números em PF

- 2º Caso: 't1' < 't2' (t1 parte inteira de m1 e t2 parte inteira de m2).
 - Parte Inteira: t2 deve ser **reduzido** para t1 bits, mantendo **sinal** e valor do **número**.
 - Isso somente se for possível representar o mesmo número com t1 bits, caso contrário, ocorre estouro de representação;
 - Parte Fracionária: f2 deve ser **estendida** para f1 bits.
 - Basta acrescentar f1-f2 bits em zero a direita da fração de m1;

Operações com Números em PF

- Na tabela 2, considere que o número final deve ser representado em 4 bits de parte inteira e 4 bits de parte fracionária;

Nº Original	Representação	Parte Inteira (t)	Fração (f)
01101,101 ₂	Inteiro-Positivo	1101 ₂	1010 ₂
111011,01 ₂	Inteiro-Positivo	Estouro	0100 ₂
01000,111 ₂	Inteiro-Positivo	1000 ₂	1110 ₂
0110110,1 ₂	Complemento de dois	Estouro	1000 ₂
11101,101 ₂	Complemento de dois	1101 ₂	1010 ₂
0001011,1 ₂	Complemento de dois	Estouro	1000 ₂

Tabela 3: Exemplos de redução de mantissa. Fonte: Adaptado de Weber, 2001.

Operações com Números em PF

➤ Exemplo: (Inteiro Positivos) 1º operando (3t e 3f) e 2º operando (1t e 5f):

➤ (t1 > t2)

$$100111_2 + 111010_2 =$$

$$100,111_2 + 1,11010_2 =$$

$$100,111_2 \qquad 100,111_2$$

$$+ \quad \underline{001,110}_2 \text{ (T)} \quad + \quad \underline{001,111}_2 \text{ (A)}$$

$$110,101_2$$

$$110,110_2$$

Operações com Números em PF

- Multiplicação em Ponto Fixo: Procede-se da mesma forma que para números inteiros, porém cuidando agora a posição da vírgula.
- $M1 (t1 + f1) \times M2 (t2 + f2) = M3 (t1+t2 \text{ de parte inteira e } f1 + f2 \text{ de parte fracionária});$

Operações com Números em PF

- Divisão em Ponto Fixo: Procede-se da mesma forma que para números inteiros, porém cuidando agora a posição da vírgula do dividendo e divisor.
- $M1 (2.t + 2.f) / M2 (t + f) =$
 - $Q = n$ bits ('t' de parte inteira e 'f' de parte fracionária);
 - $R = n$ bits ('2f' de parte fracionária);

Tópicos

- Números em Ponto Fixo;
- Operações aritméticas com números em Ponto Fixo;
- **Números em Ponto Flutuante;**
- Operações aritméticas com números em Ponto Flutuante;
- Resumo;

Números em Ponto Flutuante

- A faixa de números representáveis pelos números em ponto fixo é insuficiente para aplicações científicas;
- A representação de números em ponto flutuante é a versão binária da notação científica;
- $N = m \times b^e$, onde: N = número, m = mantissa, b = base, e = expoente;

Números em Ponto Flutuante

- A **precisão** de um número é determinado pelo número de bits da **mantissa**;
- A faixa de representação **R** depende do número de bits do **expoente**;
- Números em Ponto Flutuante são inerentemente **redundantes**;
- Ex: $1,0 \times 10^{18}$ ou $0,1 \times 10^{19}$ ou $0,01 \times 10^{20}$;
- Devemos **normalizar** as mantissas para evitar redundâncias;

Números em Ponto Flutuante

- Definição: A mantissa está **normalizada** quando:
 - É constituída somente de uma parte fracionária, **sem parte inteira**, e quando o **primeiro dígito após a vírgula é diferente de 0**;
 - No caso de números em complemento de B:
 - Positivos: começam por $0,1_2$;
 - Negativos: começam por $1,0_2$;
- Normalizar um número envolve apenas deslocamentos da mantissa e incremento ou decrementos no expoente;

Números em Ponto Flutuante

- Estouro na mantissa é resolvido com um deslocamento para a direita e corrigindo o sinal;
- Estouro no expoente indica estouro de capacidade de representação:
 - Se o expoente ultrapassou o maior n° positivo: overflow.
 - Se o expoente ultrapassou o maior n° negativo: underflow.

Números em Ponto Flutuante

- Formato de um número em Ponto Flutuante (recomendado pela IEEE):

	Simples	Duplo	Quádruplo
Campos:			
S = sinal	1 bit	1 bit	1 bit
E = expoente	8 bits	11 bits	15 bits
L = primeiro bit	(não representado)	(não representado)	1 bit
F = fração	23 bits	52 bits	111 bits
Expoente:			
Excesso de	127	1023	16383
Maior valor	255	2047	32767
Menor valor	0	0	0

Tabela 4: Formatos IEEE. Fonte: Adaptado de Weber, 2001.

Números em Ponto Flutuante

- Representar o número $4,75_{10}$: (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,75_{10} \times 2_{10}$	$1,5_{10}$	1_2
$0,5_{10} \times 2_{10}$	$1,0_{10}$	1_2
$4_{10} / 2_{10}$	2_{10}	0_2
$2_{10} / 2_{10}$	1_{10}	0_2
$1_{10} / 2_{10}$	0_{10}	1_2
Resultado: $100,11_2 \rightarrow$ Forma normalizada: $1,0011_2 \times 2^2$		

- 2º passo: Calcular o expoente em excesso de 127_{10} .
 - $2^2 = 2_{10} + 127_{10} = 129_{10} = 1000\ 0001_2$

Números em Ponto Flutuante

- 3º passo: Obter a mantissa.

Resultado: $100,11_2 \rightarrow$ Forma normalizada: $1,0011_2 \times 2^2$

Mantissa: $0011\dots_2$ (23 bits)

- 4º Passo: analisar sinal. Positivo = 0.

Sinal	Expoente	Mantissa
0_2	$1000\ 0010_2$	$001\ 1000\ 0000\ 0000\ 0000\ 0000_2$
Resposta: $01000000100110000000000000000000_2$		

Números em Ponto Flutuante

- Representar o número $-0,75_{10}$: (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,75_{10} \times 2_{10}$	$1,5_{10}$	1_2
$0,5_{10} \times 2_{10}$	$1,0_{10}$	1_2
Resultado: $-0,11_2 \rightarrow$ Forma normalizada: $-1,1_2 \times 2^{-1}$		

- 2º passo: Calcular o expoente em excesso de 127_{10} .
 - $2^{-1} = -1_{10} + 127_{10} = 126_{10} = 0111\ 1110_2$

Números em Ponto Flutuante

- 3º passo: Obter a mantissa.

Resultado: $-0,11_2 \rightarrow$ Forma normalizada: $-1,1_2 \times 2^{-1}$

Mantissa: $1..._2$ (23 bits)

- 4º Passo: analisar sinal. Negativo = 1.

Sinal	Expoente	Mantissa
1_2	$0111\ 1110_2$	$10000000000000000000000_2$
Resposta: $10111111010000000000000000000000_2$		

Números em Ponto Flutuante

- Representar o número 0.6875_{10} : (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado

- 2º passo:

Números em Ponto Flutuante

- Representar o número 0.6875_{10} : (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,6875_{10} \times 2_{10}$	$1,375_{10}$	1_2

- 2º passo:

Números em Ponto Flutuante

- Representar o número 0.6875_{10} : (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,6875_{10} \times 2_{10}$	$1,375_{10}$	1_2
$0,375_{10} \times 2_{10}$	$0,65_{10}$	0_2

- 2º passo:

Números em Ponto Flutuante

- Representar o número 0.6875_{10} : (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,6875_{10} \times 2_{10}$	$1,375_{10}$	1_2
$0,375_{10} \times 2_{10}$	$0,75_{10}$	0_2
$0,75_{10} \times 2_{10}$	$1,5_{10}$	1_2

- 2º passo:

Números em Ponto Flutuante

- Representar o número 0.6875_{10} : (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,6875_{10} \times 2_{10}$	$1,375_{10}$	1_2
$0,375_{10} \times 2_{10}$	$0,75_{10}$	0_2
$0,75_{10} \times 2_{10}$	$1,5_{10}$	1_2
$0,5_{10} \times 2_{10}$	$1,0_{10}$	1_2

- 2º passo:

Números em Ponto Flutuante

- Representar o número 0.6875_{10} : (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,6875_{10} \times 2_{10}$	$1,375_{10}$	1_2
$0,375_{10} \times 2_{10}$	$0,75_{10}$	0_2
$0,75_{10} \times 2_{10}$	$1,5_{10}$	1_2
$0,5_{10} \times 2_{10}$	$1,0_{10}$	1_2
Resultado: $0,1011_2 \rightarrow$ Forma normalizada: $1,011_2 \times 2^{-1}$		

- 2º passo:

Números em Ponto Flutuante

- Representar o número 0.6875_{10} : (1 bit sinal, 8 expoente, 23 mantissa)

- 1º passo:

Divisão/Multiplicação	Quociente/Resultado	Resto/Resultado
$0,6875_{10} \times 2_{10}$	$1,375_{10}$	1_2
$0,375_{10} \times 2_{10}$	$0,75_{10}$	0_2
$0,75_{10} \times 2_{10}$	$1,5_{10}$	1_2
$0,5_{10} \times 2_{10}$	$1,0_{10}$	1_2
Resultado: $0,1011_2 \rightarrow$ Forma normalizada: $1,011_2 \times 2^{-1}$		

- 2º passo: Calcular o expoente em excesso de 127_{10} .
 - $2^{-1} = -1_{10} + 127_{10} = 126_{10} = \mathbf{0111\ 1110_2}$

Números em Ponto Flutuante

- 3º passo: Obter a mantissa.

Resultado: $0,1011_2 \rightarrow$ Forma normalizada: $1,011_2 \times 2^{-1}$

- 4º Passo:

Sinal	Expoente	Mantissa

Números em Ponto Flutuante

- 3º passo: Obter a mantissa.

Resultado: $0,1011_2 \rightarrow$ Forma normalizada: $1,011_2 \times 2^{-1}$

Mantissa: 011...

- 4º Passo:

Sinal	Expoente	Mantissa

Números em Ponto Flutuante

- 3º passo: Obter a mantissa.

Resultado: $0,1011_2 \rightarrow$ Forma normalizada: $1,011_2 \times 2^{-1}$

Mantissa: 011...

- 4º Passo: analisar sinal. Positivo= 0.

Sinal	Expoente	Mantissa

Números em Ponto Flutuante

- 3º passo: Obter a mantissa.

Resultado: $0,1011_2 \rightarrow$ Forma normalizada: $1,011_2 \times 2^{-1}$

Mantissa: 011...

- 4º Passo: analisar sinal. Positivo= 0.

Sinal	Expoente	Mantissa
0_2	$0111\ 1110_2$	$011000000000000000000000_2$
Resposta: $0011111100110000000000000000000000000000_2$		

Números em Ponto Flutuante

IEEE precisão simples (32 bits)				
sinal	expoente	Mantissa	valor	
0	0000 0000 ₂	000 0000 0000 0000 0000 0000 ₂	+ 0	Zero
1	0000 0000 ₂	000 0000 0000 0000 0000 0000 ₂	- 0	
0	1111 1111 ₂	000 0000 0000 0000 0000 0000 ₂	+ ∞	Infinito
1	1111 1111 ₂	000 0000 0000 0000 0000 0000 ₂	- ∞	
0	1111 1111 ₂	000 0100 0100 0000 0000 0000 ₂	+ NaN	Não-Número
1	1111 1111 ₂	010 0000 0000 0000 0000 0000 ₂	- NaN	
0	0000 0000 ₂	Qualquer valor ≠ 0	Número não-normalizado	
1	0000 0000 ₂	Qualquer valor ≠ 0	Número não-normalizado	
0	0000 0001 ₂ até 1111 1110 ₂	Qualquer valor	Número normalizado	
1	0000 0001 ₂ até 1111 1110 ₂	Qualquer valor	Número normalizado	

Tabela 5: Diferentes grupos de números no formato simples da IEEE. Fonte: Elaborada pelo autor.

Números em Ponto Flutuante

Links úteis:

<http://carlosrafaelgn.com.br/aula/flutuante.html>

<http://www.h-schmidt.net/FloatConverter/IEEE754.html>

Tópicos

- Números em Ponto Fixo;
- Operações aritméticas com números em Ponto Fixo;
- Números em Ponto Flutuante;
- **Operações aritméticas com números em Ponto Flutuante;**
- Resumo;

Operações com Números em PF

- Soma e subtração:
- São feitas as operações sobre as mantissas e os expoentes são mantidos. Ambos operandos devem ter o mesmo expoente !
- Se são diferentes, o menor deve ser igualado ao maior e a mantissa deslocada para a direita para manter o valor representado pelo número.

Operações com Números em PF

- Sejam X e Y dois operandos: X_m e Y_m suas mantissas e X_e e Y_e seus expoentes;
 - 1. Se $X_e = Y_e$, então $X \pm Y = (X_m \pm Y_m) \cdot 2^{X_e}$
 - 2. Se $X_e < Y_e$, então $X \pm Y = (X_m \cdot 2^{(X_e - Y_e)} \pm Y_m) \cdot 2^{Y_e}$
 - 3. Se $X_e > Y_e$, então $X \pm Y = (X_m \pm Y_m \cdot 2^{(Y_e - X_e)}) \cdot 2^{X_e}$
- Após a operação, devemos normalizar a mantissa (pode ocorrer estouro);

Operações com Números em PF

- Multiplicação: multiplica-se as mantissas e somam-se os expoentes;
- A multiplicação segue as mesmas regras da multiplicação normal e a mantissa resultado terá o dobro de bits de comprimento;
- Para reduzir ao número normal de bits, basta realizar truncagem ou arredondamento;
- $X \times Y = (X_m \times Y_m) \cdot 2^{(X_e + Y_e)}$
- Após a multiplicação, normalizar o resultado (pode ocorrer estouro);

Operações com Números em PF

- Divisão: dividimos as mantissas e subtraímos os expoentes;
- A divisão segue as mesmas regras da divisão normal: a mantissa do dividendo é estendida para o dobro do n° de bits e depois as mantissas são divididas;
- O resto é desprezado e as etapas que corrigem quociente e resto podem ser eliminadas
- $X \div Y = (X_m \div Y_m) \cdot 2^{(X_e - Y_e)}$
- Após a divisão, normalizar o resultado (pode ocorrer estouro);

Tópicos

- Números em Ponto Fixo;
- Operações aritméticas com números em Ponto Fixo;
- Números em Ponto Flutuante;
- Operações aritméticas com números em Ponto Flutuante;
- **Resumo;**

Resumo

- Números em Ponto Fixo e Ponto Flutuante;
- Operações aritméticas com números em Ponto Fixo e Ponto Flutuante;
- Foi apresentado o Padrão IEEE 754 – 32 bits (simples);
- Exemplos de conversão de números decimais para ponto flutuante em binário;

Exercício

➤ Converta o número a seguir para decimal (IEEE 754 – single precision);

➤ $01000001001000000000000000000000_2$

➤ $10111111111110000000000000000000_2$



Dúvidas ?