

Ingeniería en Computación

Clase 4. Representaciones en Punto Fijo
Curso 2021

En esta clase:

- Concepto de Representación en Punto Fijo
- Precisión – Rango
- Operaciones en Punto Fijo

Representacion de Punto Fijo

- En los sistemas de numeración posicionales de cualquier base, se pueden diferenciar los valores fraccionarios de los enteros mediante el uso convencional de un simbolo que los separa (punto o coma decimal)
- En el caso binario por ejemplo el número representado con 9 bits en la siguiente figura sería:

2^4	2^3	2^2	2^1	2^0	●	2^{-1}	2^{-2}	2^{-3}	2^{-4}
1	0	0	1	0	●	1	1	1	1

$$2^4 + 2^1 + 2^{-1} + 2^{-2} + 2^{-3} + 2^{-4} = 16 + 2 + 0.5 + 0.25 + 0.125 + 0.0625 = 18.9375$$

- ¿Cuál sería el número más grande y el más chico (omitiendo el cero) que puede representarse en este caso?
 - Es fácil ver que el mayor sería 31.9375 y el menor 0.0625
 - tambien que la *resolución* o sea el incremento mínimo entre un valor y el siguiente también es $0.0625 = 2^{-4}$

Representacion de Punto Fijo

- Si ahora cambiamos la posición del punto, podemos obtener otro rango de representaciones y resolución.

•	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}
•	1	0	0	1	0	1	1	1	1

$$2^{-1}+2^{-4}+2^{-6}+2^{-7}+2^{-8}+2^{-9} = 1/2+1/16+1/64+1/128+1/256+1/512 = 0.591796875$$

- ¿Y ahora cuál sería el número más grande y el más chico (omitendo el cero) que puede representarse?
 - Es fácil ver que el mayor sería 0.998046875 y el menor 0.001953125
 - tambien que la **resolución** o sea el incremento mínimo entre un valor y el siguiente también es $0.001953125=2^{-9}$

Representacion de Punto Fijo

- Por último situemos el punto en otra posición. Veamos que rango de valores podemos representar.

2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	•
1	0	0	1	0	1	1	1	1	•

$$2^8 + 2^5 + 2^3 + 2^2 + 2^1 + 2^0 = 256 + 32 + 8 + 4 + 2 + 1 = 303$$

- ¿Y ahora cuál sería el número más grande y el más chico (omitiendo el cero) que puede representarse?
 - Es fácil ver que el mayor sería 511 y el menor 1.
 - También que la **resolución** o sea el incremento mínimo entre un valor y el siguiente también es $1=2^0$.

Representacion de Punto Fijo

- Representación de valores grandes

2^{12}	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	•
1	0	0	1	0	1	1	1	1	0	0	0	0	•

Aqui estamos usando los mismos 9 bits para representar el número, pero asumimos que hay 4 digitos menos significativos “ficticios” que no utilizamos.

- En el ejemplo anterior teníamos:

$$2^8+2^5+2^3+2^2+2^1+2^0 = 256+32+8+4+2+1 = 303$$

y en este tenemos

$$2^{12}+2^9+2^7+2^6+2^5+2^4 = 4096+512+128+64+32+16 = 4848$$

- ¿Y ahora cuál sería el número más grande y el más chico (omitiendo el cero) que puede representarse?
 - Es fácil ver que el mayor sería 8176 y el menor 16.
 - Tambien que la **resolución** o sea el incremento mínimo entre un valor y el siguiente también es $16=2^4$.
- Se puede verificar que los valores representados de esta manera son los mismos que se obtenían en el ejemplo anterior, pero multiplicados por 16 o sea 2^4 .

Representacion de Punto Fijo

- Si hacemos el cociente entre el valor mas alto y el mas pequeño que se pueden representar en todos los casos vistos podemos encontrar que:

$$31.9375 / 0.0625 = (2^5 - 2^{-4}) / 2^{-4} = 2^9 - 1 = 511$$

$$0.998046875 / 0.001953125 = (2^0 - 2^{-9}) / 2^{-9} = 2^9 - 1 = 511$$

$$511 / 1 = (2^9 - 2^0) / 2^0 = 2^9 - 1 = 511$$

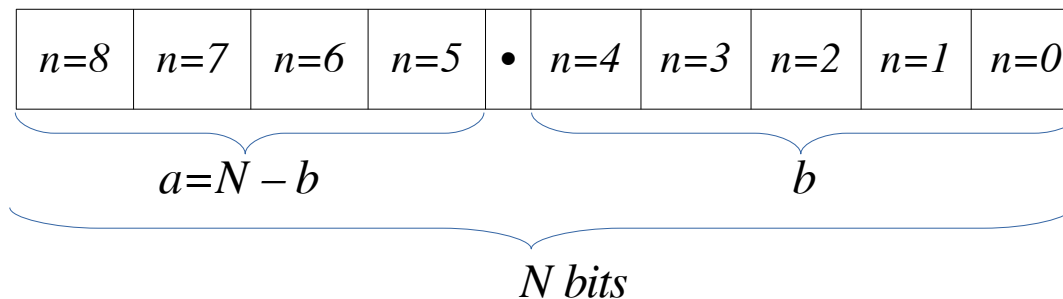
$$8176 / 16 = (2^{13} - 2^4) / 2^4 = 2^4 (2^9 - 1) / 2^4 = 511$$

- Podemos observar que:
 - Con la misma cantidad de bits, podemos representar distintos rangos de valores numéricos dependiendo de donde situemos el punto decimal que separa la parte fraccionaria de la parte entera.
 - La cantidad de valores que pueden representarse es la misma.
 - La resolución corresponde al minimo valor representable distinto que cero y esta dada por el valor asignado al LSB.
 - Esta idea se puede extender para representar valores muy pequeños o muy grandes
 - Puede interpretarse que la ubicación del punto con respecto al LSB puede modificarse multiplicando por una *escala* 2^M , donde M es un entero.

Representacion de Punto Fijo

Vamos a formalizar esto para representaciones de números positivos

- Sea una representación de punto fijo de N bits, que llamaremos $U(a, b)$.
Con $b \leq N$; $a = N - b$



- Esto se puede interpretar como ubicar el punto fraccionario entre los bits $n = b - 1$ y $n = b$ representando al conjunto de números:

$$P = \{p / 2^b \mid 0 \leq p \leq 2^N - 1, p \in \mathbb{Z}\}.$$

- El bit n tiene un peso de $2^n / 2^b = 2^{n-b}$ y la resolución es el menor valor representable 2^{-b}
- El valor de un numero x en particular está dado por $x = (1/2^b) \sum_{n=0}^{N-1} 2^n x_n$ dónde x_n representa el enésimo bit de x .
- El rango de la representación está dado por : $0 \leq x \leq (2^N - 1) / 2^b = 2^{N-b} - 2^{-b} = 2^a - 2^{-b}$

Rango: $0 \leq x \leq 2^a - 2^{-b}$

Resolución: 2^{-b}

Representacion de Punto Fijo: Ca1 y Ca2

- Si consideremos $U(N,0)$, $a=N$, $b=0$, nos queda que la resolución es 1 y el rango representable será $0 \leq x \leq 2^N - 1$
- Esto coincide con la representación binaria natural en N bits, y sabemos que si queremos representar números con signo, debemos utilizar el MSB para esta finalidad.
- Se puede verificar que si definimos $\tilde{x} = Ca1(x)$, entonces para $U(N,0)$ tendremos $\tilde{x} = 2^N - 1 - x$.
- Si definimos $\hat{x} = Ca2(x)$, entonces $\hat{x} = \tilde{x} + 1 = 2^N - x$
- Ahora si queremos representar números con signo, y el MSB nos da el signo, entonces podemos decir que el rango de números positivos, lo vamos a representar con N-1 bits, y el MSB será cero.
- Para representar los números negativos, usaremos Ca2, y los valores que obtendremos en ese caso estarán dados por $\hat{x} = \tilde{x} + 1 = 2^N - x$
- El rango de números que podemos representar estará dado por:

$$P = \{ p \mid -2^{N-1} \leq p \leq (2^{N-1} - 1), p \in \mathbb{Z} \}$$

Representacion de Punto Fijo con signo

- y el valor obtenido para una representación dada en N bits será:

$$x = \left[-2^{N-1} x_{N-1} + \sum_{n=0}^{N-2} 2^n x_n \right]$$

- Puede verse que si el MSB (ubicado en la posición $N-1$) es cero, la representación de x es la de un número positivo de $N-1$ bits.
- Si el MSB es 1, podemos comprobar que se está realizando la operación de Ca2 sobre x y queda representado un número negativo.
- Si combinamos esta representación con lo que vimos para punto fijo positivo, estaríamos pasando de $U(N,0)$ a una representación con signo, que llamaremos $Q(N-1,0)$.
- Si ahora tomamos como antes a bits para la parte entera y b bits para la parte fraccionaria, generalizamos la representación a $Q(a,b)$ en la cual $a=N-b-1$ y el conjunto de números que pueden representarse será:

$$P = \{ p / 2^b \mid -2^{N-1} \leq p \leq (2^{N-1} - 1), p \in \mathbb{Z} \}$$

Representacion de Punto Fijo con signo

- El valor x representado por una $Q(a,b)$ de N bits será entonces:

$$x = \left(\frac{1}{2^b}\right) \left[-2^{N-1} x_{N-1} + \sum_{n=0}^{N-2} 2^n x_n \right]$$

- El factor $\left(\frac{1}{2^b}\right)$ se llama **Escala** y el rango de representación será:
$$-2^{N-1-b} \leq x \leq +2^{N-1-b} - 1/2^b$$

- Nótese que $a+b = N-1$ porque se utiliza el MSB para el signo
 - Ejemplo 1: ¿Qué número representa 01011001 en $Q(3,4)$?*
 - Podemos ver que el número es positivo, entonces el valor corresponde a la representación entera del número 89 y debe multiplicarse por la **escala** dada por $1/2^4 = 1/16$ se obtiene 5,5625
 - Ejemplo 2: ¿Qué número representa 10010110 en $Q(2,5)$?*
 - El número es negativo (MSB=1) y la escala es $1/2^5 = 1/32$, entonces el número representado es $-106 / 32 = -3,3125$

Operaciones

- Largo sin signo
 - El número de bits para representar $U(a,b)$ es $a+b$
- Largo con signo
 - El número de bits para representar $Q(a,b)$ es $a+b+1$
- Rango sin signo.
 - El rango de $U(a,b)$ es $0 \leq x \leq 2^a - 2^{-b}$.
- Rango con signo
 - el rango de $Q(a,b)$ es $-2^a \leq x \leq 2^a - 2^{-b}$.
- Cambio de escala:
 - Si tengo una representación de un número x en $Q(a,b)$ donde la escala es $1/2^b$ y quiero pasarlo a otra representación $Q_1(a_1, b_1)$ con la misma cantidad de bits N , se debe cambiar la escala desplazando los bits de la representación hacia la izquierda o hacia la derecha según b_1 sea mayor o menor que b respectivamente. Esto equivale a multiplicar o dividir por 2 respectivamente.
 - Debe analizarse si no se pierden bits más significativos al realizar esta operación
- Operandos de adición
 - Dos números binarios deben tener igual escala para poder sumarlos.
 - Esto es, la operación $X(c,d) + Y(e,f)$ es solamente válida si $X = Y$ (ya sea Q o U) y además se tiene que cumplir que $c=e$ y $d=f$.