

Módulo 05

Aritmética de Punto Flotante (Pt. 1)



Organización de Computadoras
Depto. Cs. e Ing. de la Comp.
Universidad Nacional del Sur



Copyright

- Copyright © 2011-2023 A. G. Stankevicius
- Se asegura la libertad para copiar, distribuir y modificar este documento de acuerdo a los términos de la **GNU Free Documentation License**, Versión 1.2 o cualquiera posterior publicada por la Free Software Foundation, sin secciones invariantes ni textos de cubierta delantera o trasera
- Una copia de esta licencia está siempre disponible en la página <http://www.gnu.org/copyleft/fdl.html>
- La versión transparente de este documento puede ser obtenida de la siguiente dirección:

<http://cs.uns.edu.ar/~ags/teaching>

Contenidos

- Inconvenientes con la representación **FXP**
- Un problema con el escalamiento
- Formatos para la representación **FLP**
- Concepto de orden de magnitud cero (**OMZ**)
- Situaciones especiales a contemplar
- Operaciones aritméticas básicas

Aritmética de punto fijo

- La aritmética de punto fijo (FXP) es apropiada para representar números de base pequeña con órdenes de magnitud acotados
 - Por caso, operando con 32 bits de precisión, el rango a manejar esta acotado por $\pm(2^{31} - 1)$, lo cual corresponde en base 10 a $\pm(10^{10})$
- No obstante, este rango es inadecuado en un contexto ingenieril o para las aplicaciones científicas, donde se manejan magnitudes más grandes y también más pequeñas

Aritmética de punto flotante

- Una alternativa superadora consiste en hacer uso de la notación científica, también conocida como de punto flotante (FLP)
- Un cierto número real f se denota como el producto entre una mantisa m y un exponente e , de la siguiente forma:

$$f = m \times b^e$$

- Tanto m como e son números signados expresados en una cierta base, mientras que b es un entero positivo que denota la base de la representación

Aritmética de punto flotante

- Volviendo al ejemplo anterior, en caso de contar con 32 bits de precisión, se deben asignar estos bits para representar tanto a m como a e
 - Nótese que b no es almacenado, su valor se asume, queda implícito una vez acordada la representación de punto flotante a ser empleada
 - Una posibilidad consiste en reservar por caso los 22 bits más a la izquierda para m y los restantes 10 bits para e , haciendo uso de una base implicada $b = 2$

m (22 bits)	e (10 bits)
-------------	-------------

Aritmética de punto flotante

- ¿Habrá mejorado el rango al hacer uso de esta representación de **FLP** en lugar de **FXP**?
 - Recordemos, con **32 bits FXP** permite representar números enteros de unos **10 dígitos** en base **10**.
 - En cambio, en **FLP**, codificando el exponente por caso en **SM** permite representar número de unos **500 dígitos** en base **2**.
 - Es decir, números de unos **150 dígitos** en base **10**

m (22 bits)	e (10 bits)
-------------	-------------

Aritmética de punto flotante

- Por lo general, la mantisa **m** puede asumir cualquiera de los tres sistemas vistos para punto fijo
- Como se puede apreciar, el punto en la mantisa flota a partir de la magnitud del exponente
 - Es por esta razón que a esta notación se la refiere como de “punto flotante” (floating point)
- Para el caso del exponente **e**, también se puede usar cualquiera de los sistemas vistos o bien hacer uso de alguno de propósito específico

Aritmética de punto flotante

- El hardware para operar en **FLP** difiere en gran medida del que se usa para operar en **FXP**
 - Más adelante al repasar los algoritmos para operar en punto flotante esta diferencia se hará evidente
 - Es decir, las computadoras que permitan operar en **FLP** serán necesariamente más costosas
- Ahora bien, ¿no sería más razonable intentar resolver esos problemas ingenieriles y científicos usando directamente punto fijo?

Inconvenientes con FXP

- Inicialmente las primeras computadoras hacían uso exclusivo de aritmética **FXP**
 - Al realizar cálculos científicos, se redondeaban las magnitudes con frecuencia a fin de mantener acotado el número de dígitos significativos
- En estas aplicaciones es importante resolver correctamente con las cuestiones relativas al rango, a la precisión y al nivel de significancia de los valores representados

Inconvenientes con FXP

- Esas primeras computadoras hacían un uso intensivo del intervalo unitario **[-1, +1]**
 - La idea es que cuando la magnitud de un número se torne muy grande o muy pequeña durante el cómputo, el programador sea el responsable de seguir la posición de la coma en los resultados intermedios
 - Los fuera de rango se manejaban usualmente vía escalamiento por software, firmware o hardware
 - A su vez, las magnitudes con las que se debía operar usualmente no cabían en ese acotado rango

Inconvenientes con FXP

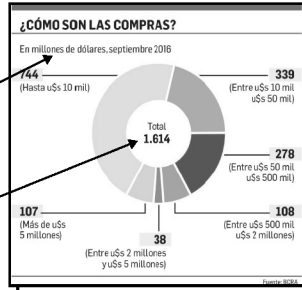
- Continúa:
 - En muchos casos, el número a ser representado debía ser escalado hacia arriba o hacia abajo para poder ser representado
 - Naturalmente, al final del cómputo el resultado debía ser transformado nuevamente al dominio requerido por el usuario del sistema
 - Téngase presente que sin estas transformaciones, el hardware **FXP** produciría resultados carentes sentido (producto de los múltiples overflows)

Un problema de escalamiento

- Los factores de escalamiento son un recurso habitual para escapar a las limitaciones cuando representamos información

siempre debe estar especificado el factor de escalamiento en uso

con sólo cuatro dígitos se logra representar al número 1.614.000.000



Organización de Computadoras - Mg. A. G. Stankevicius - 13

Un problema de escalamiento

- Para hacer uso de esta técnica, se debe explicitar el factor de escalamiento a utilizar:
 - Todo número en notación FXP de n dígitos de precisión en un base b al hacer uso de un factor de escala b^k representará un valor absoluto menor que b^k , resultando en un máximo error de $b^{-(n-k)} = b^{k-n}$
 - Como se puede apreciar, el factor de escalamiento también escalará al error
 - Esto se debe a que el intervalo $[-1, +1]$, cuyo máximo error para una precisión n es apenas b^{-n} , ahora representa magnitudes del intervalo $[-b^k, +b^k]$

Organización de Computadoras - Mg. A. G. Stankevicius - 14

Un problema de escalamiento

- Esta pérdida de precisión puede ser remediada de diversas formas
- Una posibilidad es hacer uso de aritmética de punto fijo de múltiple precisión
 - Es decir, la idea es usar múltiples palabras para cada magnitud representada
 - Naturalmente, el hacer uso de más de una palabra implica usualmente un mayor costo tanto en tiempo de ejecución como en espacio de almacenamiento

Organización de Computadoras - Mg. A. G. Stankevicius - 15

Un problema de escalamiento

- Para cálculos largos o complicados, este escalamiento y extensión de precisión implica un arduo análisis matemático y cómputo lateral para seguir los factores de escala o controlar las longitudes de las palabras
- Una complicación adicional surge al considerar que sólo se puede usar un único factor de escala sobre un dado conjunto de operandos
 - Nótese que la introducción del escalamiento único acarreará importantes pérdidas de significancia

Ejemplo

- En 2016 en Mercado Libre figuraba listada una Ferrari F-430 F1 Spyder a **\$528.000** dólares
- Supongamos que también quiero comprar un llavero original que cuesta **\$39** dólares



Ejemplo

- Si el hardware sólo puede representar tres dígitos fraccionarios base **10**, ¿qué factor de escalamiento conviene usar en cada caso?
 - Para la Ferrari, “millones de dólares”
 - Para el llavero, “miles de dólares”
- Ahora bien, si quisiera calcular el costo total, ahí se debe usar el mayor factor para representar ambas magnitudes:
 - La Ferrari se representa como **0.528...** ¿y el llavero?

Un problema de escalamiento

- Retomando:

- Como vimos, la diferencia actual entre un factor de escala común b^k para un orden de magnitud b^z , en un cierto número con algún $z < k$, creará $k - z - 1$ ceros a la izquierda
- En consecuencia, en vez de contar con n dígitos que aporten significancia sólo se contará con un máximo de $n - (k - z - 1)$ dígitos
- La pérdida sucesiva de significancia en una secuencia de operaciones desencadena situaciones singulares, que el hardware no estará en condiciones de resolver



Aritmética de punto flotante

- En respuesta a todos estos cuestionamientos, se introdujo la representación de punto flotante en la década del '40
- Una idea brillante fue simplemente permitir que cada magnitud sea acompañada por su propio factor de escala a lo largo de la computación
 - Es decir, las magnitudes preservarán hasta último momento la máxima precisión, ya que utilizarán en todo momento el menor factor de escala posible



¿Preguntas?