

DSP. Introducción

Dispositivos Lógicos Programables
Ingeniero de Telecomunicación

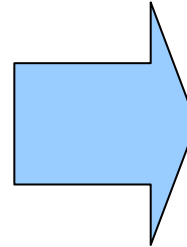


Universidad
de Oviedo

Introducción

Operaciones

Suma
Multiplicación

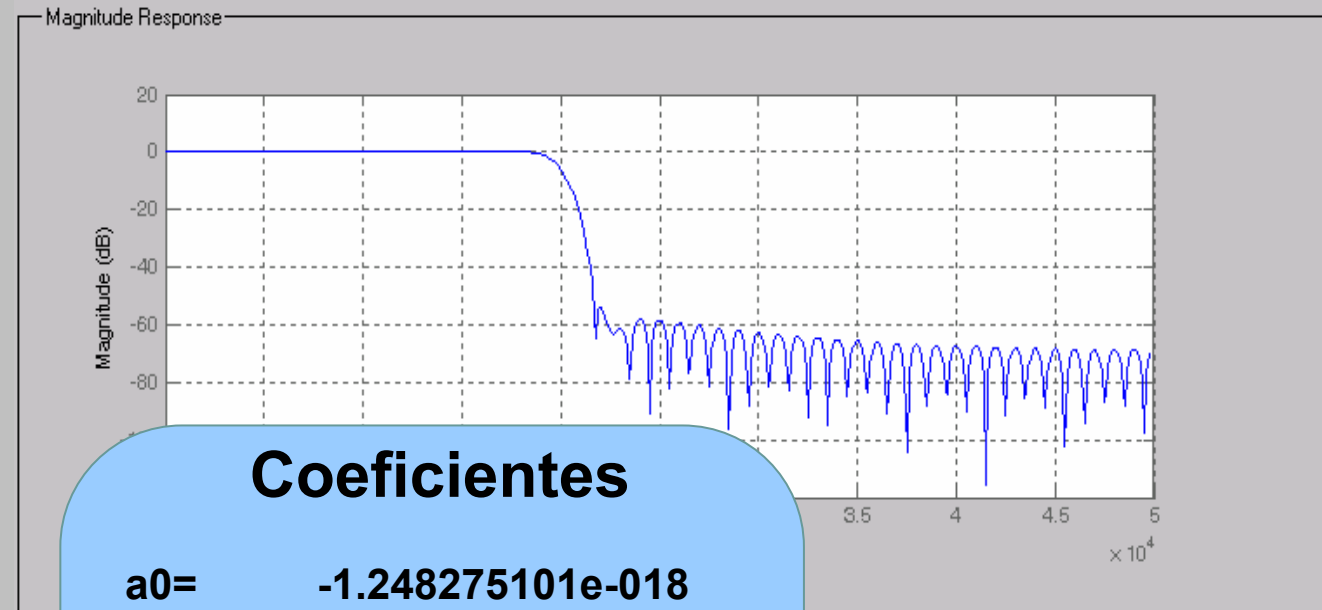


MAC
(Multiply
Add
Accumulate)


$$A \cdot B + C \cdot D$$

Ejemplo

**Filtro FIR
(Hamming,
pasabajos,
orden 100,
diseñado con
Filter Design &
Analysis Tool de
MATHLAB)**



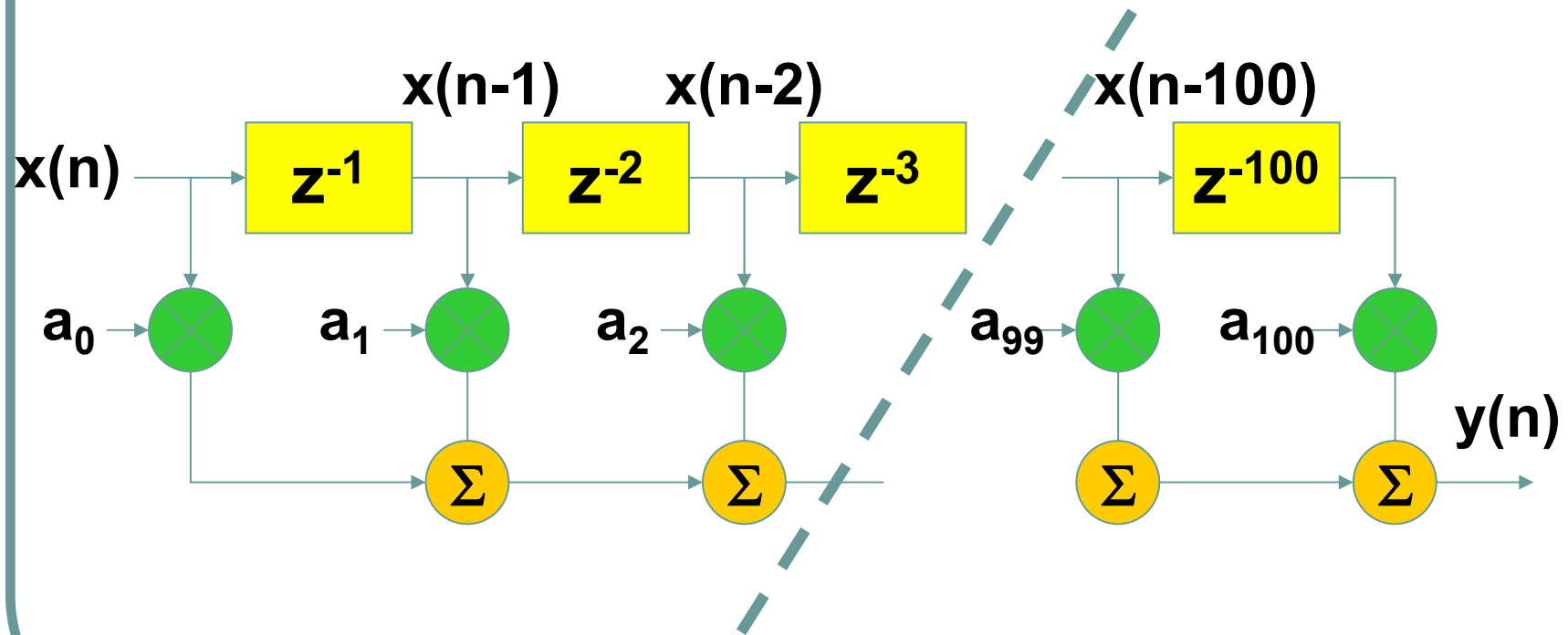
Coeficientes

a0=	-1.248275101e-018
a1=	-0.0005002068356
a2=	-0.0003261930251
a3=	0.0003511414689
a4=	0.0006220251671
a5=	-1.599571056e-018
.....	
a97=	0.0003511414689
a98=	-0.0003261930251
a99=	-0.0005002068356
a100=	-1.248275101e-018



Ejemplo

$$y(n] = \sum_{i=0}^{100} a_i z^{-i}$$



¿qué se necesita para implementar el filtro?

Convertidor A/D para leer la muestra

Memoria muestras leídas

X(n)
X(n-1)
X(n-3)
·
·
·
·
·
X(n-99)
X(n-100)

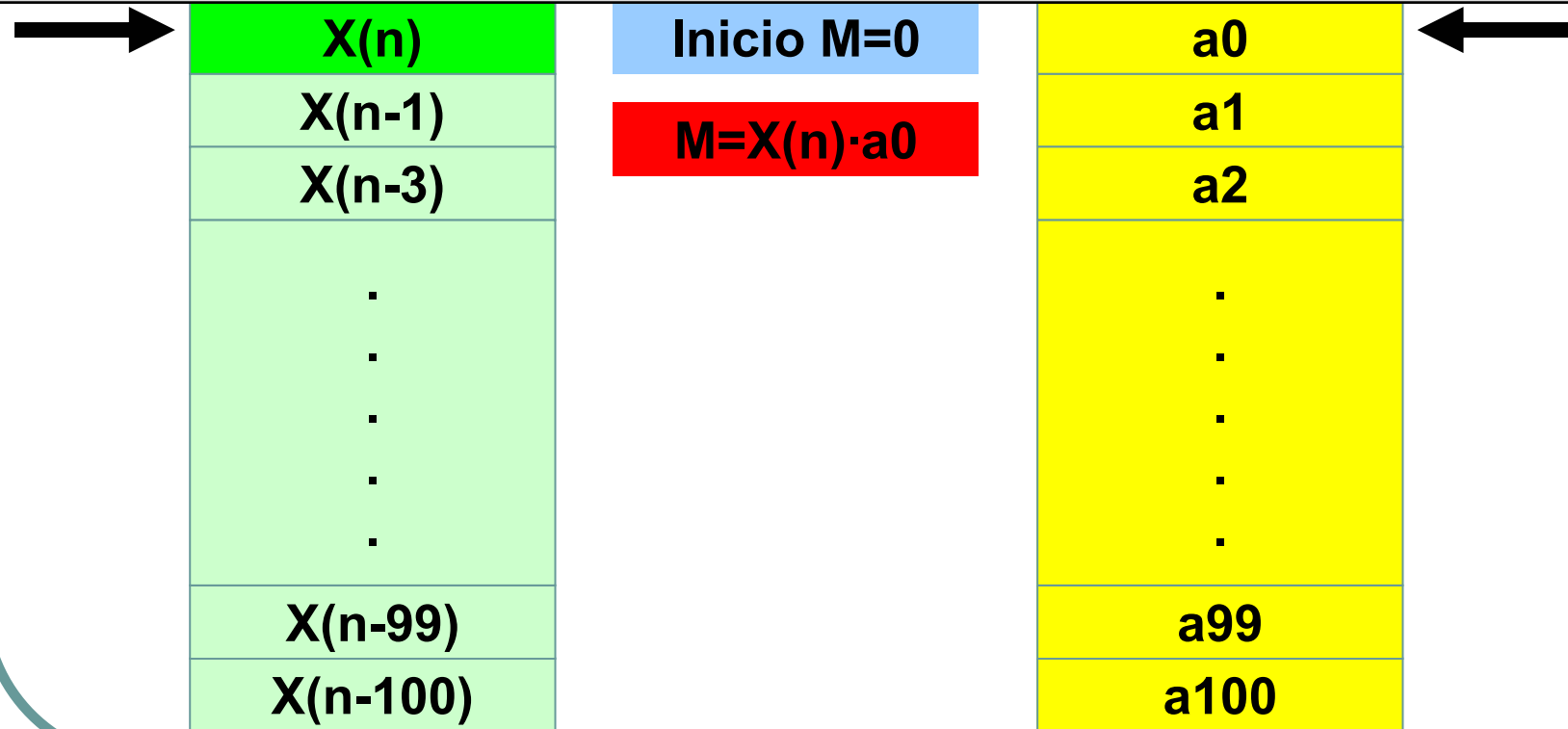
Memoria coeficientes

a0
a1
a2
·
·
·
·
·
a99
a100



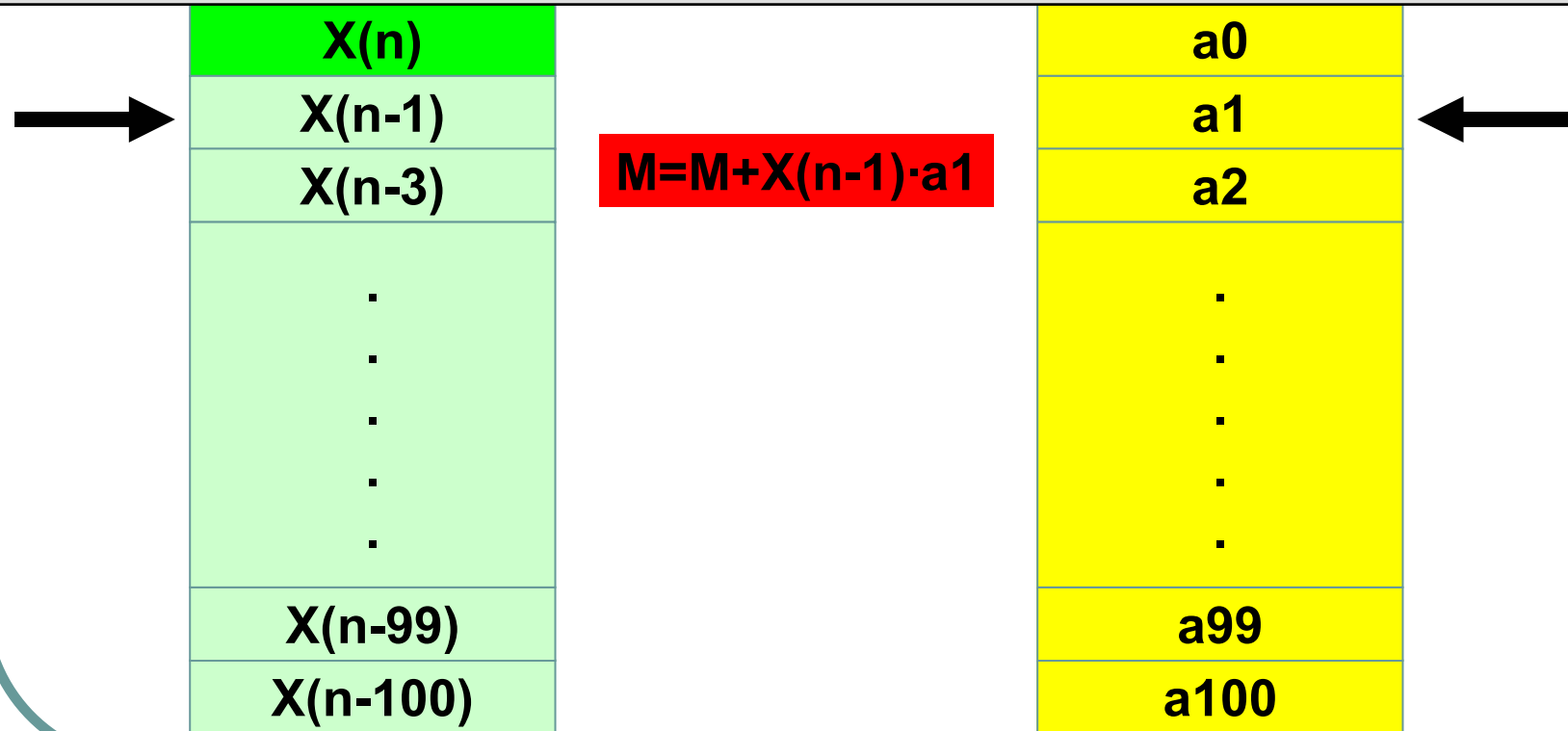
¿qué se necesita para implementar el filtro?

Paso 1: Apuntar a posición muestra, leerla
Apuntar a posición coeficiente, leerlo
Multiplicar

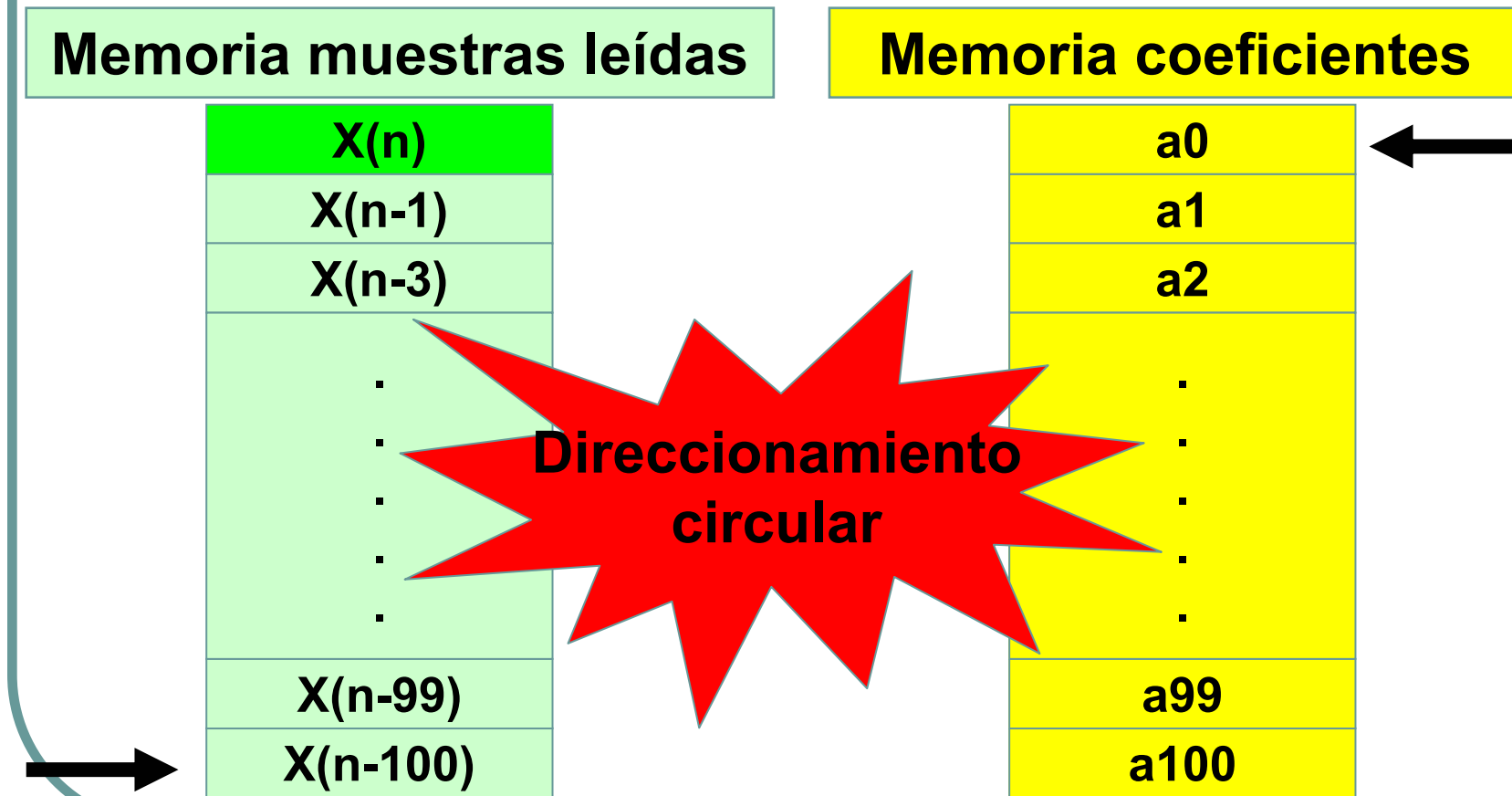


¿qué se necesita para implementar el filtro?

**Paso 2...: Avanzar una posición en muestra, leer muestra
Idem coeficientes
multiplicarlo y sumarlo a lo anterior. Repetir hasta el final**



¿qué se necesita para implementar el filtro?



Comparación

Microprocesador

Bus de direcciones grande
Formato de datos grande (p.e 32 bits)
Modos direccionamiento sofisticados
Coprocesador
Coste, tamaño, consumo elevados

Microcontrolador

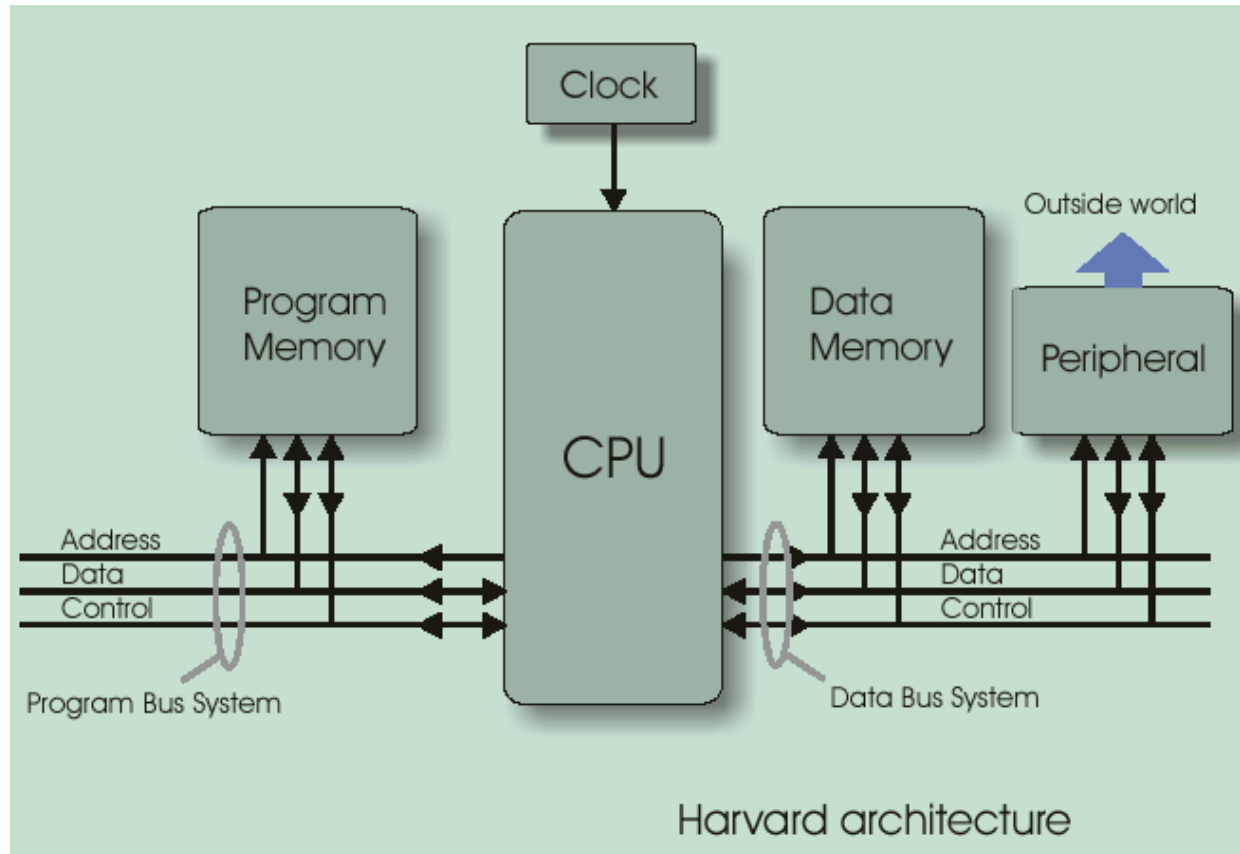
Memoria y periféricos integrados
Bus de direcciones pequeño
Datos pequeños (8, 16)
ALU sencilla
Modos direccionamiento sencillos
Consumo, tamaño, coste reducidos

DSP

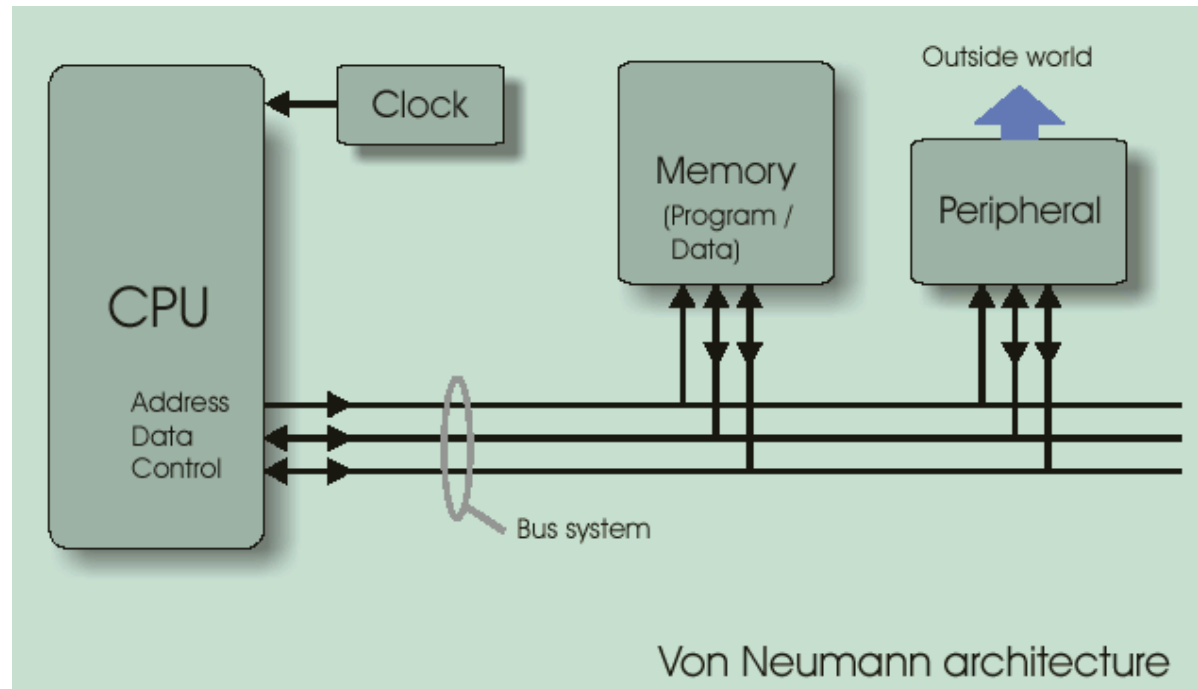
Coma fija o flotante
Bus de direcciones pequeño
Datos mediano (16, 32)
ALU sencilla
Periféricos integrados
Modos direccionamiento pensados para procesamiento
Consumo, tamaño, coste rel. reducidos



Arquitecturas



Arquitecturas



Coma fija (formato 1.15)

MSB		BIT WEIGHT												LSB	
-2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}	2^{-6}	2^{-7}	2^{-8}	2^{-9}	2^{-10}	2^{-11}	2^{-12}	2^{-13}	2^{-14}	2^{-15}

HEX	BINARY				DECIMAL
7FFF	0111	1111	1111	1111	+0.999969
0001	0000	0000	0000	0001	+0.000031
0000	0000	0000	0000	0000	+0.000000
FFFF	1111	1111	1111	1111	-0.000031
8000	1000	0000	0000	0000	-1.000000

© INTRODUCTION TO DIGITAL SIGNAL PROCESSORS. Brian L. Evans. The University of Texas at Austin



Ej. Coma fija (formato 1.15)

0.0010100000000000 (0,15625)

0.0010000000000000 (0,125)

0.0110100000000000 (0,40625)

0.0011100000000000 (0,21875)

Suma (0,90625)

0,1110000000000000 (0,875)

1,1110000000000000 (-0,125)

1,1000000000000000 (-0,5)

1,1111000000000000 (-0,0625)

Suma (0.1875)

$$0,5 \times -0,375 = -0,1875$$

$$-0,5 \times 0,5 = -0,25$$



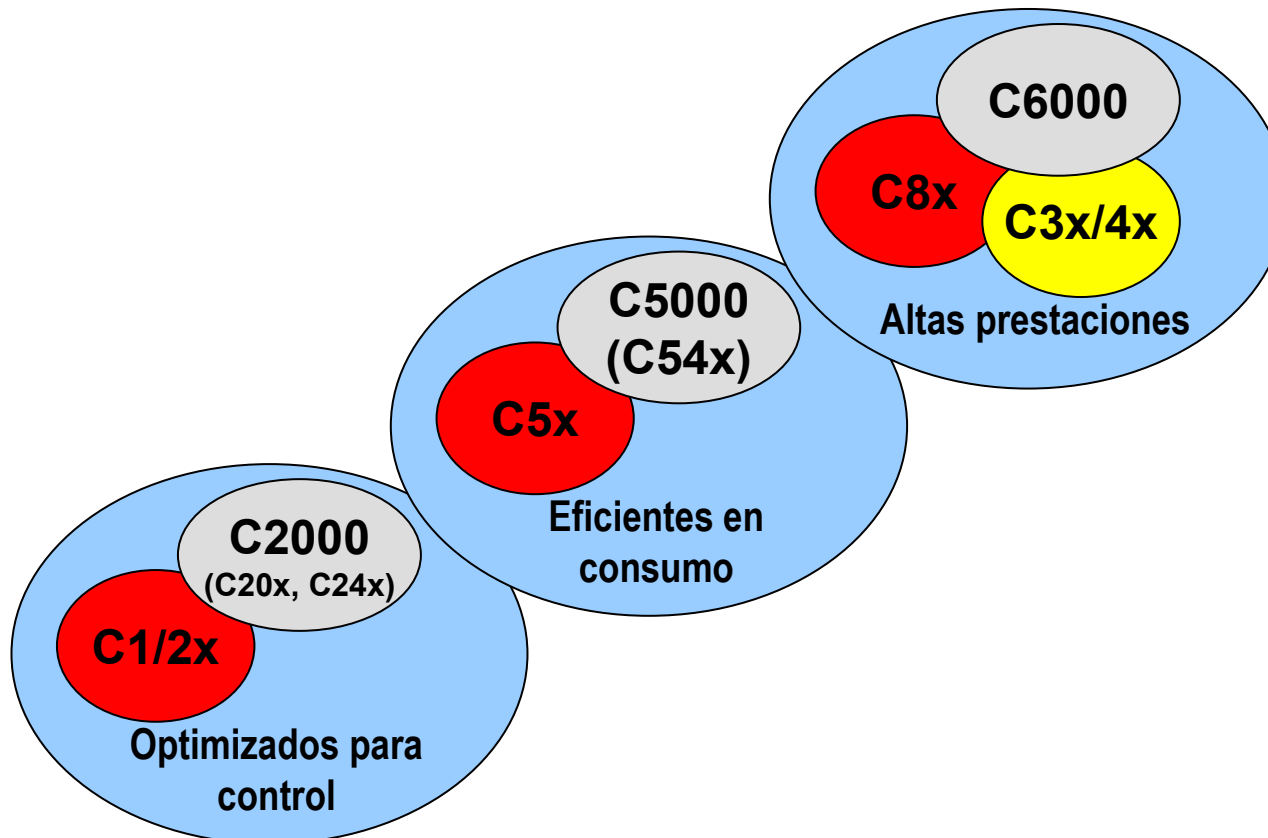
Coma fija-coma flotante

	<i>Coma fija</i>	<i>Coma flotante</i>
<i>Coste unitario</i>	\$5 - \$79	\$5 - \$381
<i>Arquitectura</i>	Accumulator	load-store or memory-register
<i>Registros</i>	2-4 data 8 address	8 or 16 data 8 or 16 address
<i>Palabra de datos</i>	16 or 24 bit integer and fixed-point	32 bit integer and fixed/floating-point
<i>Memoria interna</i>	2-64 kwords data 2-64 kwords program	8-64 kwords data 8-64 kwords program
<i>Espacio direcciones</i>	16-128 kw data 16-64 kw program	16 Mw – 4Gw data 16 Mw – 4 Gw program
<i>Compiladores</i>	C compilers; poor code generation	C, C++ compilers; better code generation
<i>Ejemplo</i>	TI TMS320C5x; Motorola 56000	TI TMS320C3x; Analog Devices SHARC

© INTRODUCTION TO DIGITAL SIGNAL PROCESSORS. Brian L. Evans. The University of Texas at Austin



Ej. Texas Instruments



TMS320C6000:
TMS320C62x
TMS320C64x
TMS320C67x

TMS320C5000:
TMS320C54x
TMS320C55x

TMS320C2000:
TMS320C24x
TMS320C28x

Otros TMS 320
TMS320C33
TMS320C1x
TMS320C2x
TMS320C3x
TMS320C4x
TMS320C5x
TMS320C8x

Ej. Texas Instruments

