

Introducción

- Vamos a resolver algoritmos utilizando **instrucciones vectoriales**.
- Debemos **conocer** las instrucciones que tenemos disponibles.
- y las **técnicas** para pensar algoritmos desde la operatoria vectoriales.

Registros y tipos de datos

- Registros:

XMM0 a XMM15 de 128 bits (16 bytes)

- Tipos de datos:

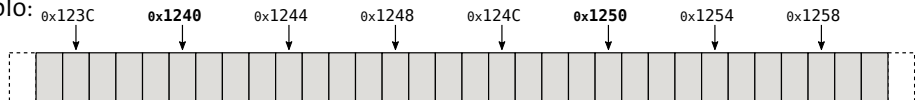
Enteros: 8, 16, 32, 64 y 128.

Float: 32 (Float) y 64 (Double).

Operaciones Load/Store

MOVD	MOVQ	Move Doubleword/Quadword
MOVSS	MOVSD	Moves a 32bits Single FP/64bits Double FP
MOVDQA	MOVDQU	Moves aligned/unaligned double quadword
MOVAPS	MOVUPS	Moves 4 aligned/unaligned 32bit singles
MOVAPD	MOVUPD	Moves 2 aligned/unaligned 64bit doubles

Ejemplo:



MOVD [0x123C], xmm0	✓	[0x123C] ← xmm0(32:0)
MOVQ xmm0, [0x1245]	✓	xmm0(64:0) ← [0x1245]
MOVDQA xmm0, [0x1245]	✗	Error dirección no alineada.
MOVDQA [0x1250], xmm0	✓	[0x1250] ← xmm0(128:0)
MOVSS xmm0, [0x1248]	✓	xmm0(32:0) ← [0x1248] ; sobre punto flotante
MOVUPS [0x1258], xmm0	✓	[0x1258] ← xmm0(128:0) ; sobre punto flotante

Operaciones Load/Store

PMOVSX ^B W	PMOVZX ^B W	packed sign/zero extension byte to word
PMOVSX ^B D	PMOVZX ^B D	packed sign/zero extension byte to dword
PMOVSX ^B Q	PMOVZX ^B Q	packed sign/zero extension byte to qword
PMOVSX ^W D	PMOVZX ^W D	packed sign/zero extension word to dword
PMOVSX ^W Q	PMOVZX ^W Q	packed sign/zero extension word to qword
PMOVSX ^D Q	PMOVZX ^D Q	packed sign/zero extension word to dqword

Ejemplos:

PMOVSXBD	xmm0, xmm0	✓	
PMOVZXWD	xmm0, [data]	✓	
PMOVZXDQ	xmm0, xmm1	✓	
PMOVZXQD	xmm0, xmm0	×	Instrucción invalida.
PMOVSXBD	[data], xmm0	×	Modo de direccionamiento invalido.

Operaciones Aritméticas

PADDB	PADDW	PADDQ	PADDQ	Add Integer
PSUBB	PSUBW	PSUBD	PSUBQ	Sub Integer
PMULHW	PMULLW			Mul Integer Word
PMULHD	PMULLD			Mul Integer Dword
PMINSB	PMAXSB	PMINUB	PMAXUB	Max and Min Integer
PMINSW	PMAXSW	PMINUW	PMAXUW	Max and Min Integer
PMINSD	PMAXSD	PMINUW	PMAXUD	Max and Min Integer

Ejemplos:

PADDQ xmm0, xmm1 ✓

PSUBW xmm0, [data] ✓

PMULLD xmm0, xmm1 ✓

PMAXSW xmm0, [data] ✓

PMINSB [data], xmm0 × Modo de direccionamiento invalido.

Operaciones Aritméticas

PABSB	Absolute for 8 bit Integers
PABSW	Absolute for 16 bit Integers
PABSD	Absolute for 32 bit Integers

Ejemplos:

PABSD xmm0, xmm0 ✓

PABSD xmm0, [data] ✓

PABSD [data], xmm0 × Modo de direccionamiento invalido.

Operaciones Aritméticas

PADD SB	PADD SW	Add Int saturation
PADD USB	PADD USW	Add Int unsigned saturation
PSUB SB	PSUB SW	Sub Int saturation
PSUB USB	PSUB USW	Sub Int unsigned saturation

Ejemplos:

PADD~~SW~~ xmm0, xmm0 ✓

PSUB~~USB~~ xmm0, [data] ✓

PSUB~~SW~~ [data], xmm0 ✗ Modo de direccionamiento invalido.

Operaciones Aritméticas

ADDPS	ADDSS	ADDPD	ADDSD	Addition of FP values
SUBPS	SUBSS	SUBPD	SUBSD	Subtraction of FP values
MULPS	MULSS	MULPD	MULSD	Multiply of FP values
DIVPS	DIVSS	DIVPD	DIVSD	Division of FP values
MAXPS	MAXSS	MINPS	MINSS	Max and Min of Single FP values
MAXPD	MAXSD	MINPD	MINSD	Max and Min of Double FP values

Ejemplos:

ADDPS xmm0, [data] ✓

ADDPD xmm0, [data] ✓

ADDSS xmm0, [data] ✓

ADDSD xmm0, [data] ✓

MINSD [data], xmm0 × Modo de direccionamiento invalido.

Operaciones Aritméticas

SQRT ^{SS}	SQRT ^{PS}	Square root of Scalar/Packed Single FP values
SQRT ^{SD}	SQRT ^{PD}	Square root of Scalar/Packed Double FP values

Ejemplos:

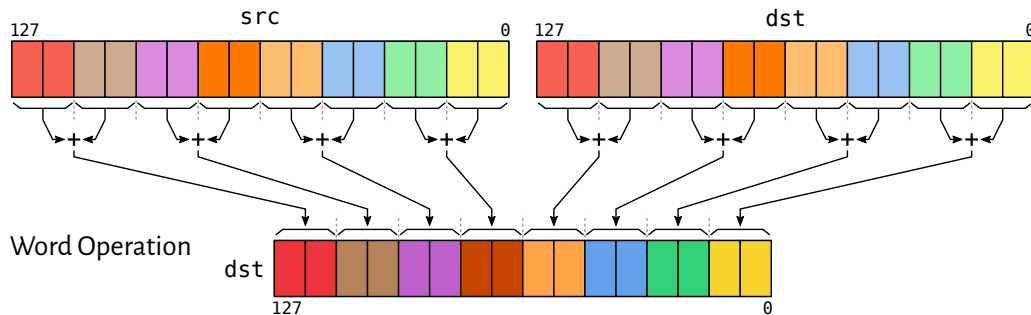
SQRTPS xmm0, [data] ✓

SQRTSS xmm0, [data] ✓

SQRTPD [data], xmm0 × Modo de direccionamiento invalido.

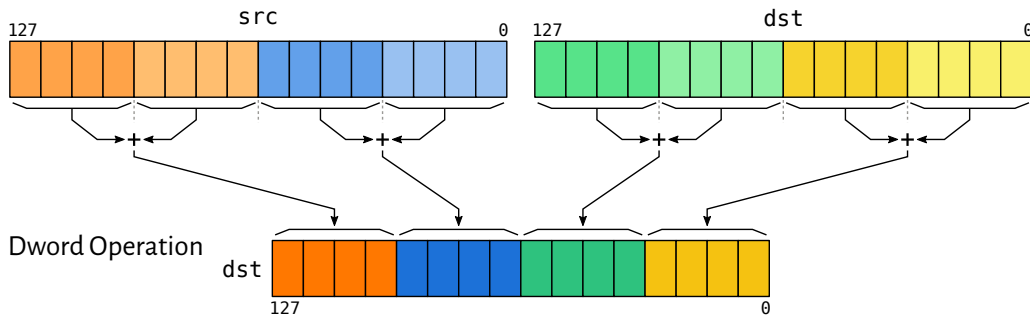
Operaciones Aritméticas

PHADDW	PHADD	Horizontal addition of unsigned 16bit/32bit integers
PHADDSW		Horizontal saturated addition of 16bit integers
PHSUBW	PHSUB	Horizontal subtraction of unsigned 16bit/32bit integers
PHSUBSW		Horizontal saturated subtraction of 16bit words
HADDPS	HADDPD	Packed Single/Double FP Horizontal Add
HSUBPS	HSUBPD	Packed Single/Double FP Horizontal Subtract



Operaciones Aritméticas

PHADDW	PHADD	Horizontal addition of unsigned 16bit/32bit integers
PHADDSW		Horizontal saturated addition of 16bit integers
PHSUBW	PHSUB	Horizontal subtraction of unsigned 16bit/32bit integers
PHSUBSW		Horizontal saturated subtraction of 16bit words
HADDPS	HADDPD	Packed Single/Double FP Horizontal Add
HSUBPS	HSUBPD	Packed Single/Double FP Horizontal Subtract



Operaciones Lógicas

PAND	PANDN	POR	PXOR	Operaciones lógicas para enteros.
AND ^{PS}	ANDN ^{PS}	OR ^{PS}	XOR ^{PS}	Operaciones lógicas para <i>float</i> .
AND ^{PD}	ANDN ^{PD}	OR ^{PD}	XOR ^{PD}	Operaciones lógicas para <i>double</i> .

- Actúan lógicamente sobre todo el registro, sin importar el tamaño del operando.
- La distinción entre ^{PS} y ^{PD} se debe a meta información para el procesador.

PSLLW	PSLLD	PSLLQ	PSLLDQ*
PSRLW	PSRLD	PSRLQ	PSRLDQ*
PSRAW	PSRAD		

- Todos los *shifts* operan de forma lógica como aritmética, tanto a derecha como izquierda.
- Se limitan a realizar la operación sobre cada uno de los datos dentro del registro según su tamaño.
- * En las operaciones indicadas, el parámetro es la cantidad de bytes del desplazamiento.

Técnica: Operatoria con mascarar

11111111 00000000 11111111 00000000

1. Calculo de la mascara

NOT (11111111 00000000 11111111 00000000)

11111111 00000000 11111111 00000000

AND

AND

Datos



00000000 00000000 00000000 00000000

00000000 00000000 00000000 00000000

2. Aplicación de la mascara

00000000 00000000 00000000 00000000

OR

00000000 00000000 00000000 00000000

3. Combinación de resultados



Operaciones de comparación

PCMPEQB	PCMPEQW	PCMPEQD	PCMPEQQ	Compare Packed Data for Equal
PCMPGTB	PCMPGTW	PCMPGTD	PCMPGTQ	Compare Packed Signed Int for Greater Than

Ejemplos:

PCMPEQB xmm0, [data] ✓

PCMPEQW xmm0, [data] ✓

PCMPEQD xmm0, [data] ✓

PCMPEQQ xmm0, [data] ✓

PCMPGTQ [data], xmm0 × Modo de direccionamiento invalido.

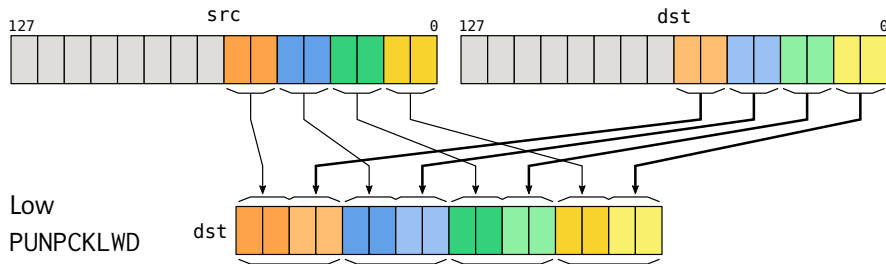
Operaciones de comparación

CMP xx PD	Compare Packed Double-Precision Floating-Point Values
CMP xx PS	Compare Packed Single-Precision Floating-Point Values
CMP xx SD	Compare Scalar Double-Precision Floating-Point Values
CMP xx SS	Compare Scalar Single-Precision Floating-Point Values
COMISD	Compare Scalar Ordered Double-Precision Floating-Point Values and Set EFLAGS
COMISS	Compare Scalar Ordered Single-Precision Floating-Point Values and Set EFLAGS

Ejemplos:			Acción	xx	CMPxxyy A, B	
			0	Igual	EQ	$A = B$
CMPEQPD xmm0, [data]	✓		1	Menor	LT	$A < B$
CMPLEPD xmm0, [data]	✓		2	Menor o Igual	LE	$A \leq B$
CMPORDPD xmm0, [data]	✓	; (Nan)	3	No Orden	UNORD	$A, B = unordered$
			4	Distinto	NEQ	$A \neq B$
			5	No Menor	NLT	$not(A < B)$
			6	No Meno o Igual	NLE	$not(A \leq B)$
			7	Orden	ORD	$A, B = Ordered$

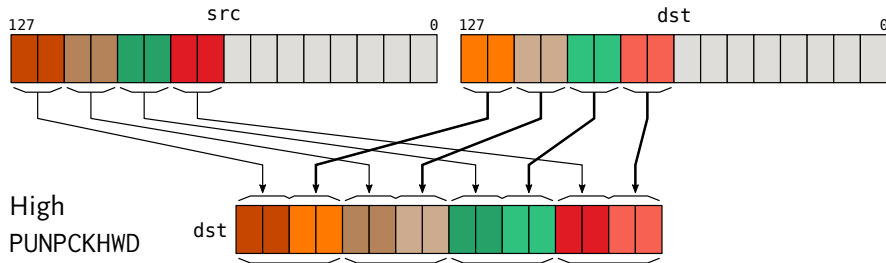
Operaciones de desempaquetado (Unpack)

PUNPCK ^L BW	PUNPCK ^H BW	Unpacks 8 enteros de 8 bits en words
PUNPCK ^L WD	PUNPCK ^H WD	Unpacks 4 enteros de 16 bits en dwords
PUNPCK ^L DQ	PUNPCK ^H DQ	Unpacks 2 enteros de 32 bits en qwords
PUNPCK ^L QDQ	PUNPCK ^H QDQ	Unpacks 1 entero de 64 bits en 128 bits
UNPCK ^L PS	UNPCK ^H PS	Unpacks Single FP
UNPCK ^L PD	UNPCK ^H PD	Unpacks Double FP



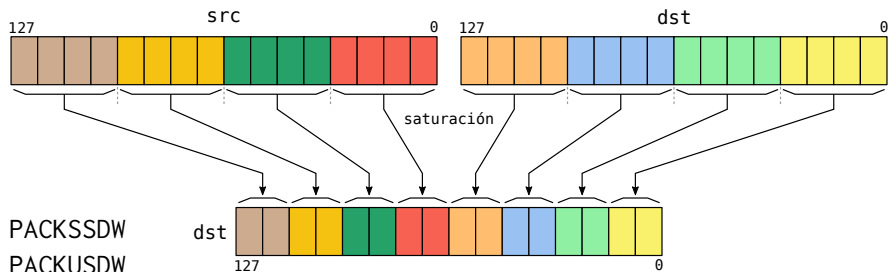
Operaciones de desempaqueado (Unpack)

PUNPCK ^L BW	PUNPCK ^H BW	Unpacks 8 enteros de 8 bits en words
PUNPCK ^L WD	PUNPCK ^H WD	Unpacks 4 enteros de 16 bits en dwords
PUNPCK ^L DQ	PUNPCK ^H DQ	Unpacks 2 enteros de 32 bits en qwords
PUNPCK ^L QDQ	PUNPCK ^H QDQ	Unpacks 1 entero de 64 bits en 128 bits
UNPCK ^L PS	UNPCK ^H PS	Unpacks Single FP
UNPCK ^L PD	UNPCK ^H PD	Unpacks Double FP



Operaciones de desempaquetado (Unpack)

PACKSSDW	Packs 32 bits (signado) a 16 bits (signado) usando saturation
PACKUSDW	Packs 32 bits (signado) a 16 bits (sin signo) usando saturation
PACKSSWB	Packs 16 bits (signado) a 8 bits (signado) usando saturation
PACKUSWB	Packs 16 bits (signado) a 8 bits (sin signo) usando saturation



Técnica: Operatoria de desempaquetado y empaquetado

