

Primer Parcial

- Introducción a lenguaje ensamblador.
- Instrucciones simples de la arquitectura x86.
- Objetivo:
 - Familiarizarse con las instrucciones básicas de programación.

MOV: move data

- **Syntax:**

MOV dest, orig

- **Operation:**

dest ← orig

-	-	-	-	-	-	-
---	---	---	---	---	---	---

of df sf zf af pf cf

CMOV cc : conditional move

- **Syntax:**

```
CMOV $cc$  dest, orig
```

- **Operation:**

```
if( $cc$ )  
     $dest \leftarrow orig$   
endif
```

- **Notes:** cc is any of the condition codes.

-	-	-	-	-	-	-
---	---	---	---	---	---	---

of df sf zf af pf cf

XCHG: exchange data

- **Syntax:**

XCHG *op1*, *op2*

- **Operation:**

temp ← *op1*

op1 ← *op2*

op2 ← *temp*

-	-	-	-	-	-	-
---	---	---	---	---	---	---

of df sf zf af pf cf

Arithmetic Instructions

- CLC
- STC
- CMC
- ADD
- ADC
- INC
- SUB
- SBB
- DEC
- NEG
- CMP
- MUL
- IMUL
- DIV
- IDIV
- CBW
- CWD
- CDQ
- CWDE
- MOVSX
- MOVZX

CLC: clear carry flag

- **Syntax:**

CLC

- **Operation:**

CF ← 0

-	-	-	-	-	-	0
---	---	---	---	---	---	---

of df sf zf af pf cf

STC: set carry flag

- **Syntax:**

STC

- **Operation:**

CF \leftarrow 1

-	-	-	-	-	-	1
---	---	---	---	---	---	---

of df sf zf af pf cf

CMC: complement carry flag

- **Syntax:**

CMC

- **Operation:**

$CF \leftarrow \sim CF$

-	-	-	-	-	-	X
---	---	---	---	---	---	---

of df sf zf af pf cf

ADD: add integers

- **Syntax:**

ADD dest, orig

- **Operation:**

dest ← *dest* + *orig*

X	-	X	X	X	X	X
---	---	---	---	---	---	---

of df sf zf af pf cf

ADD examples

- AX: 0075
- CX: 01A2

Add ax,cx

- Results:
 - AX: 0217
 - CX: 01A2
 - SF=ZF=CF=OF=0

ADD examples

- AX: 77AC
- CX: 4B35

add ax, cx

- Results:
 - AX: C2E1
 - CX: 4B35
 - SF=OF=1; ZF=CF=0

ADC: add with carry

- **Syntax:**

ADC dest, orig

- **Operation:**

dest ← *dest* + *orig* + CF

X	-	X	X	X	X	X
---	---	---	---	---	---	---

of df sf zf af pf cf

INC: increment integer

- **Syntax:**

`INC dest`

- **Operation:**

$dest \leftarrow dest + 1$

X	-	X	X	X	X	-
---	---	---	---	---	---	---

of df sf zf af pf cf

INC examples

- ECX: 00 00 01 A2
inc ecx
- Results:
- ECX= 00 00 01 A3
- SF=ZF=OF=0

INC examples

- EDX: 7F FF FF FF

inc edx

- Results:
- EDS: 80 00 00 00
- SF=OF=1; ZF=0

SUB: subtract integers

- **Syntax:**

SUB dest, orig

- **Operation:**

dest ← *dest* - *orig*

X	-	X	X	X	X	X
---	---	---	---	---	---	---

of df sf zf af pf cf

SUB examples

- EAX: 00 00 00 75
- ECX: 00 00 01 A2
- `sub eax, ecx`
- Results:
- EAX: FF FF FE D3
- ECX: 00 00 01 A2
- SF=1, ZF=CF=OF=0

SUB examples

- DX: FF 20
- Word at value FF 20
 sub dx, Value
- Results:
- DX:00 00
- Value: FF 20
- ZF=1, the rest are zero.

SBB: subtract with borrow

- **Syntax:**

SBB dest, orig

- **Operation:**

dest ← *dest* - *orig* - CF

X	-	X	X	X	X	X
---	---	---	---	---	---	---

of df sf zf af pf cf

DEC: decrement integer

- **Syntax:**

`DEC dest`

- **Operation:**

$dest \leftarrow dest - 1$

X	-	X	X	X	X	-
---	---	---	---	---	---	---

of df sf zf af pf cf

DEC examples

- BX: 00 01

dec bx

- Results:
- BX: 00 00
- ZF=1; SF=OF=0

DEC examples

- AL: F5

dec al

- Results:
- AL: F4
- SF=1; OF=ZF=0

NEG: negate

- **Syntax:**

NEG dest

- **Operation:**

dest \leftarrow $-$ *dest*

- **Notes:** Sets CF , unless *dest* is zero, y which case CF is cleared.

X	-	X	X	X	X	X
---	---	---	---	---	---	---

of df sf zf af pf cf

NEG examples

- BX: 01 A2

neg bx

- Results:
- BX: FE 5E
- SF=1; ZF=0

NEG examples

- DH: F5

neg dh

- Results:
- DH:0B
- SF=ZF=0

NEG examples

- EAX: 00 00 00 00

neg eax

- Results:
- EAX: 00 00 00 00
- SF=0; ZF=1