

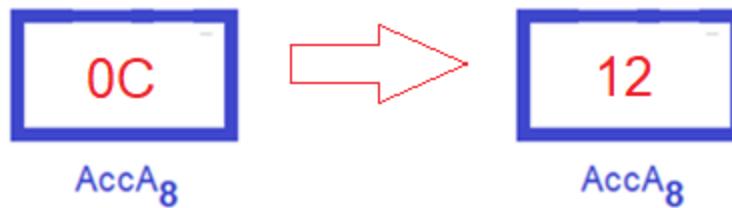
Set de instrucciones del **MC68HC11**

Instrucciones aritméticas de incremento, decremento, complemento a 1, complemento a 2 y puesto a cero.

DAA

Decimal adjust A

- Modo de direccionamiento **Inherente**
- **Opcode:** \$19 **Ciclo:**2 **Byte:**1
- Ajusta el contenido del registro acumulador A con el equivalente en BCD del acumulador A.
- Actualiza banderas N,Z,V y C.

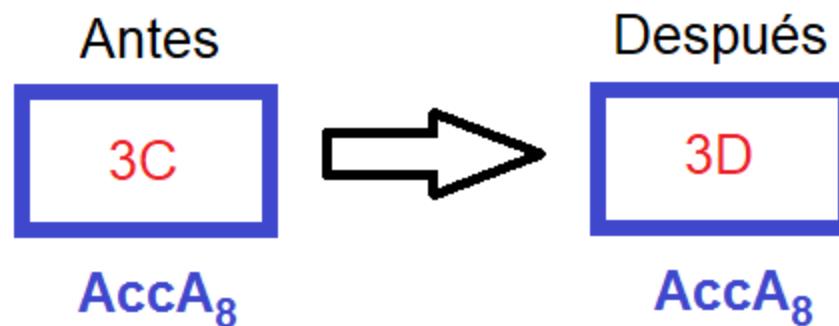


La actualización solo es posible si la bandera de H está en alto.

INCA

$A+1 \rightarrow A$

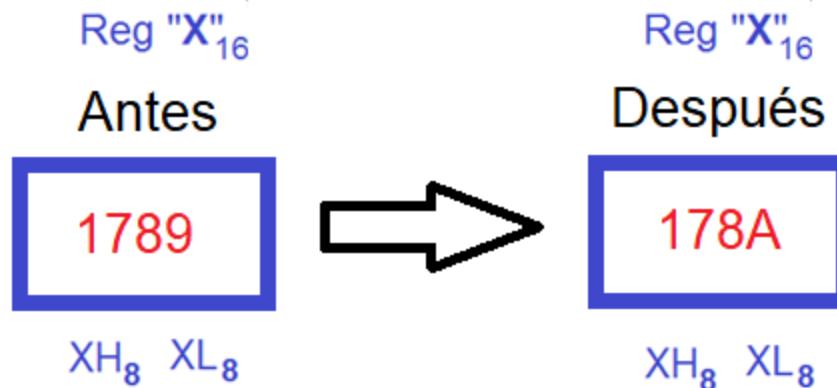
- Modo de direccionamiento **Inherente**
- **Opcode:** \$4C **Ciclo:**2 **Byte:**1
- Incrementa el contenido del registro acumulador A en una unidad.
- Actualiza banderas N, Z, y V



INX

$X+1 \rightarrow X$

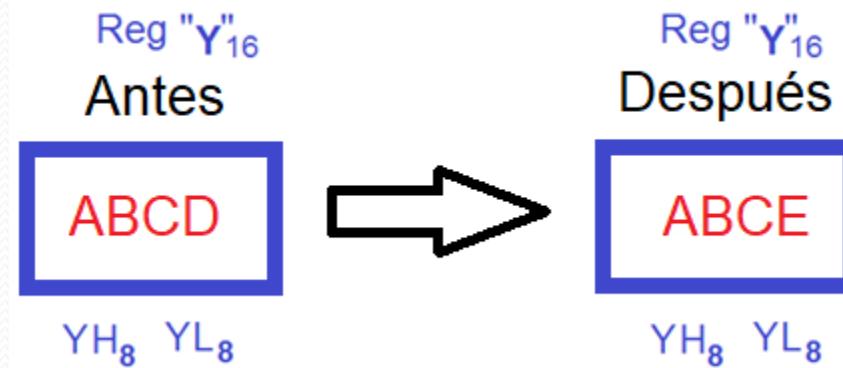
- Modo de direccionamiento **Inherente**
- **Opcode:** \$08 **Ciclo:**3 **Byte:**1
- Incrementa el contenido del registro X en una unidad.
- Actualiza bandera Z



INY

$Y+1 \rightarrow Y$

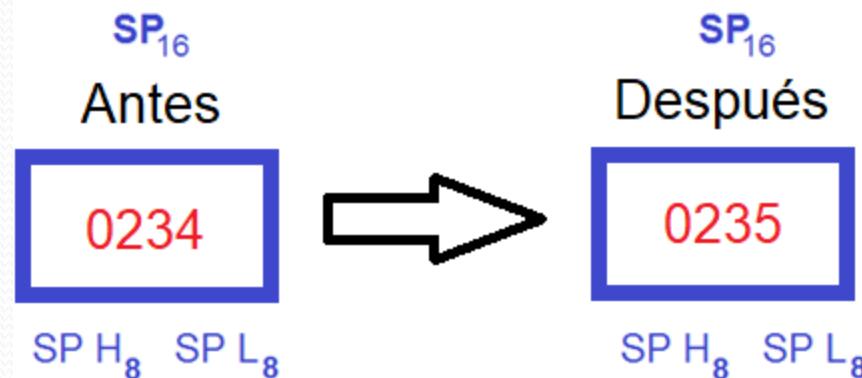
- Modo de direccionamiento Inherente
- **Opcode:** \$1808 **Ciclo:**3 **Byte:**2
- Incrementa el contenido del registro Y en una unidad.
- Actualiza bandera Z



INS

$SP+1 \rightarrow SP$

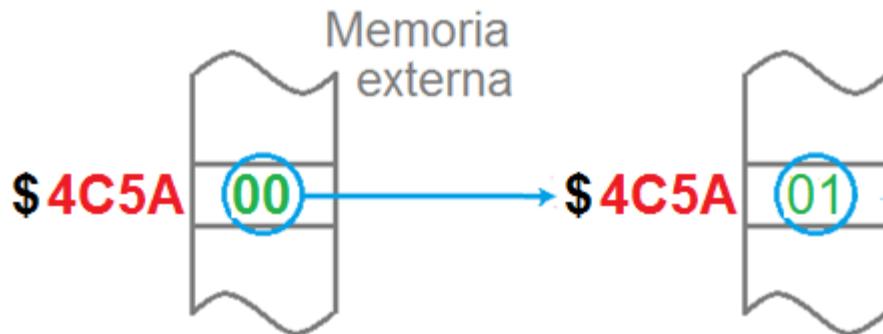
- Modo de direccionamiento **Inherente**
- **Opcode:** \$31 **Ciclo:**3 **Byte:**1
- Incrementa el contenido del registro apuntador de pila (Stack Pointer) en una unidad.
- No actualiza bandera alguna



INC \$4C5A

M+1 → M

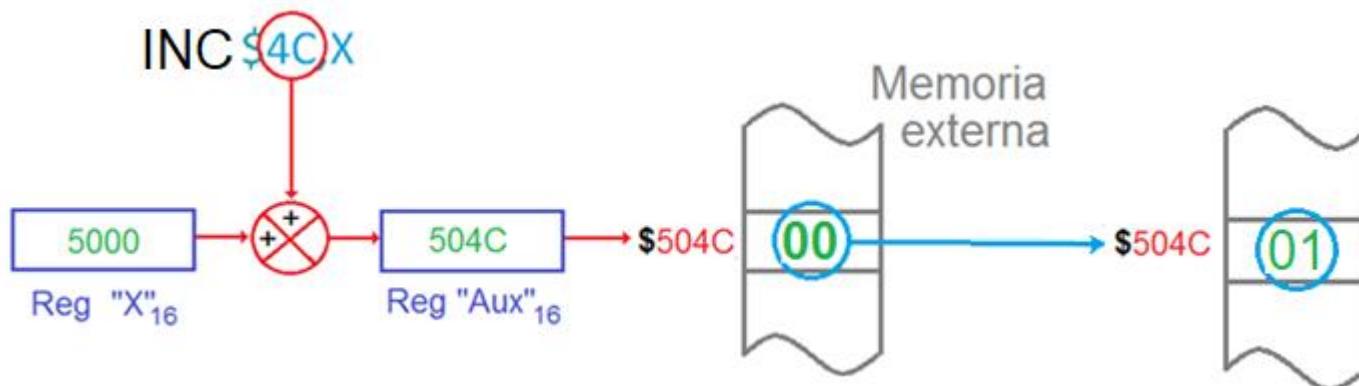
- Modo de direccionamiento **extendido**
- **Opcode:** \$7C **Ciclo:**6 **Byte:**3
- Incrementa el contenido de la memoria en una unidad (Debe ser localidad de memoria RAM).
- Actualiza banderas N, Z y V



INC \$4C,X

M+1 → M

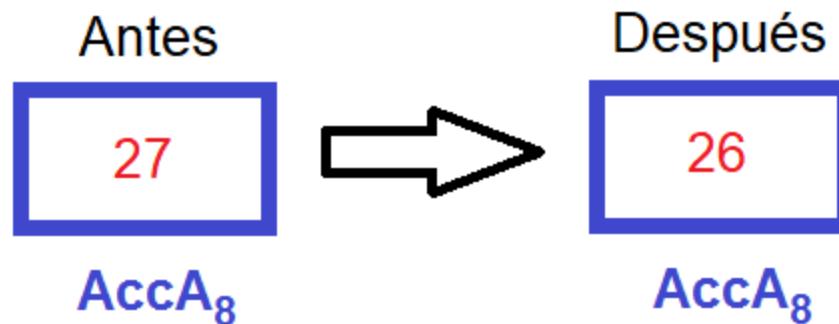
- Modo de direccionamiento **Indexado**
- **Opcode:** \$6C **Ciclo:**6 **Byte:**2
- Incrementa el contenido de la memoria en una unidad (Debe ser localidad de memoria RAM).
- Actualiza banderas N, Z y V



DECA

A-1 → A

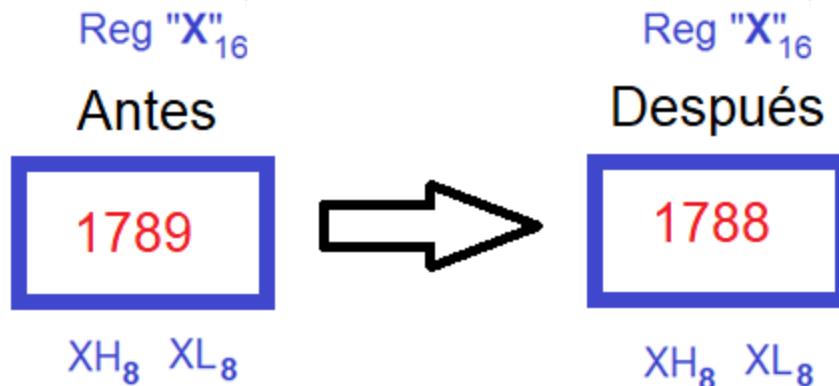
- Modo de direccionamiento Inherente
- **Opcode:** \$4A **Ciclo:**2 **Byte:**1
- Decrementa el contenido del registro acumulador A en una unidad.
- Actualiza banderas N, Z, y V



DEX

$X-1 \rightarrow X$

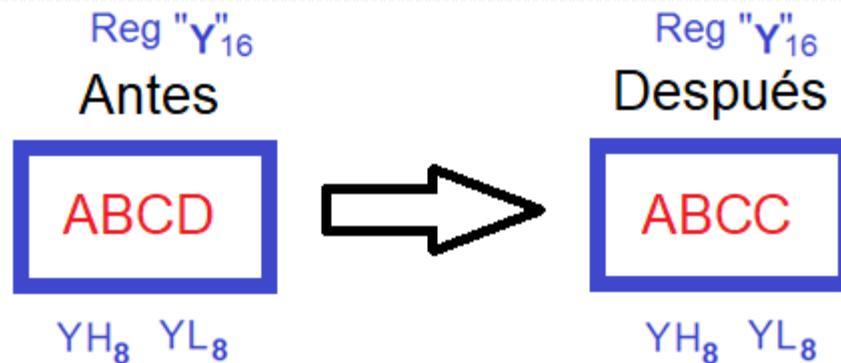
- Modo de direccionamiento **Inherente**
- **Opcode:** \$09 **Ciclo:**3 **Byte:**1
- Decrementa el contenido del registro X en una unidad.
- Actualiza bandera Z



DEY

$Y-1 \rightarrow Y$

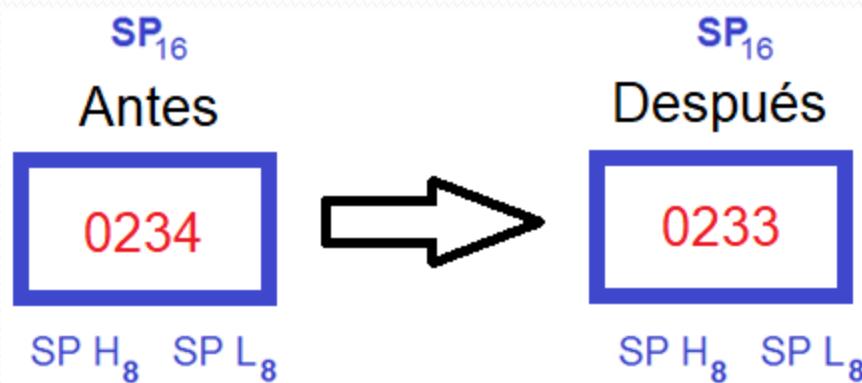
- Modo de direccionamiento **Inherente**
- **Opcode:** \$1809 **Ciclo:**3 **Byte:**2
- Decrementa el contenido del registro Y en una unidad.
- Actualiza bandera Z



DES

SP-1→ SP

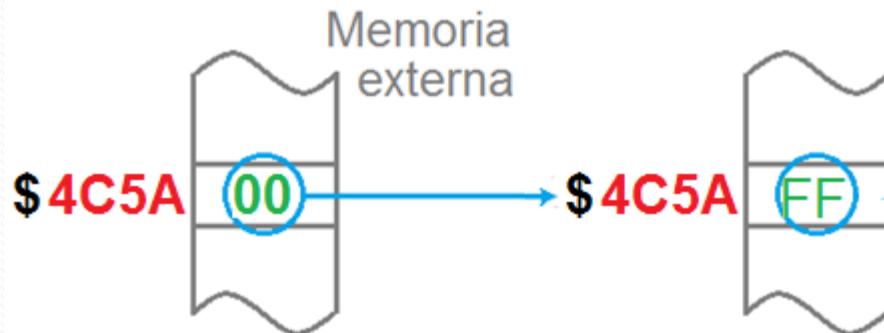
- Modo de direccionamiento **Inherente**
- **Opcode:** \$34 **Ciclo:**3 **Byte:**1
- Decrementa el contenido del registro apuntador de pila (Stack Pointer) en una unidad.
- No actualiza bandera alguna



DEC \$4C5A

M-1 → M

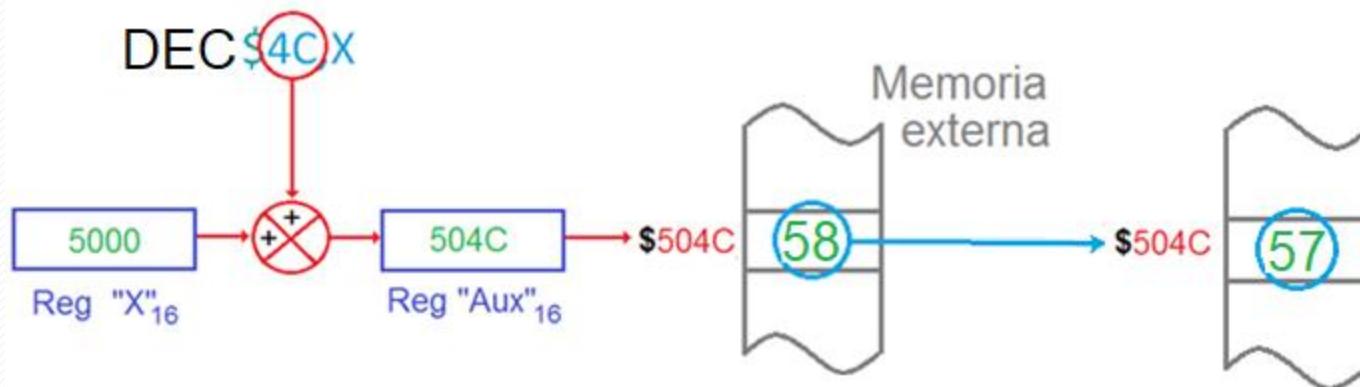
- Modo de direccionamiento **extendido**
- **Opcode:** \$7A **Ciclo:**6 **Byte:**3
- Decrementa el contenido de la memoria en una unidad (Debe ser localidad de memoria RAM).
- Actualiza banderas N, Z y V



DEC \$4C,X

M+1 → M

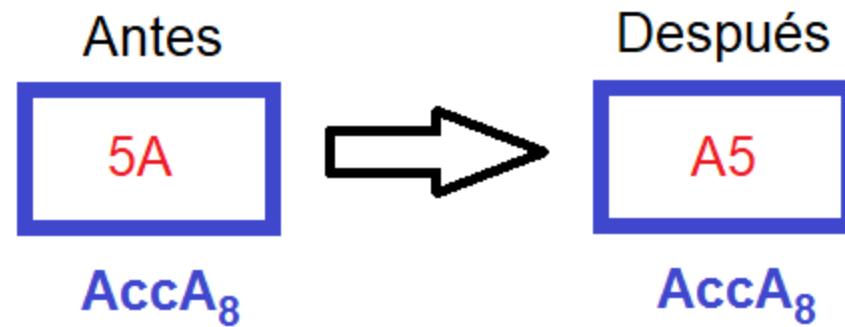
- Modo de direccionamiento **Indexado**
- **Opcode:** \$6A **Ciclo:**6 **Byte:**2
- Decrementa el contenido de la memoria en una unidad (Debe ser localidad de memoria RAM).
- Actualiza banderas N, Z y V



COMA

$\$FF-A \rightarrow A$

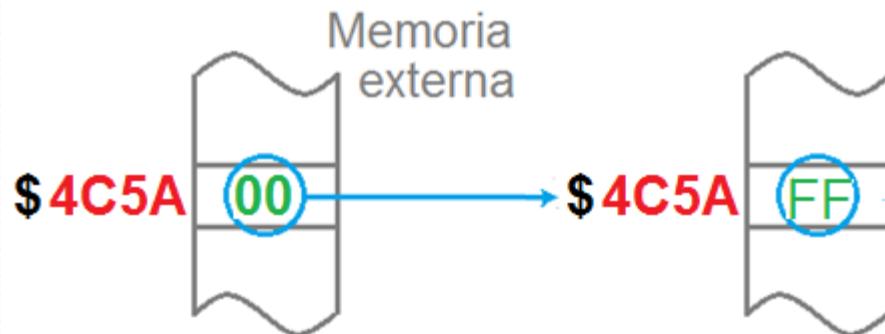
- Modo de direccionamiento **Inherente**
- **Opcode:** \$43 **Ciclo:**2 **Byte:**1
- Devuelve el complemento a uno del contenido del registro acumulador A.
- Actualiza banderas N, Z, y V=0, C=1



COM \$4C5A

\$FF-M → M

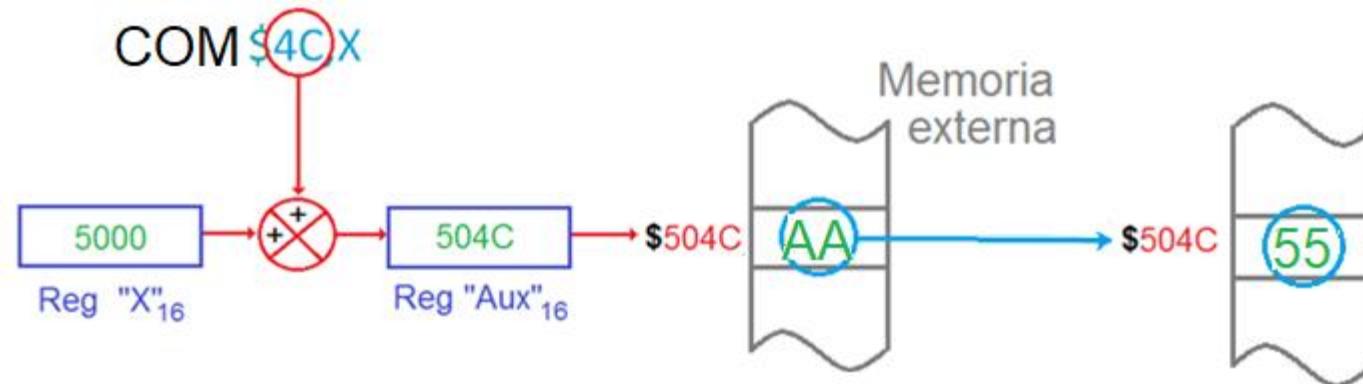
- Modo de direccionamiento **extendido**
- **Opcode:** \$73 **Ciclo:**6 **Byte:**3
- Devuelve el complemento a uno del contenido de la localidad de memoria RAM
- Actualiza banderas N, Z y V=0, C=1



COM \$4C,X

\$FF-M → M

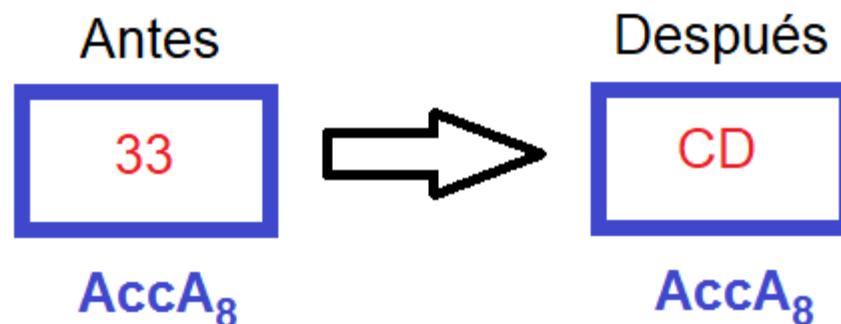
- Modo de direccionamiento **Indexado**
- **Opcode:** \$63 **Ciclo:**6 **Byte:**2
- Devuelve el complemento a uno del contenido de la localidad de memoria RAM
- Actualiza banderas N, Z y V=0, C=1



NEGA

\$00-A → A

- Modo de direccionamiento **Inherente**
- **Opcode:** \$40 **Ciclo:**2 **Byte:**1
- Devuelve el complemento a dos del contenido del registro acumulador A.
- Actualiza banderas N, Z,V,C



NEG \$4C5A

\$00-M → M

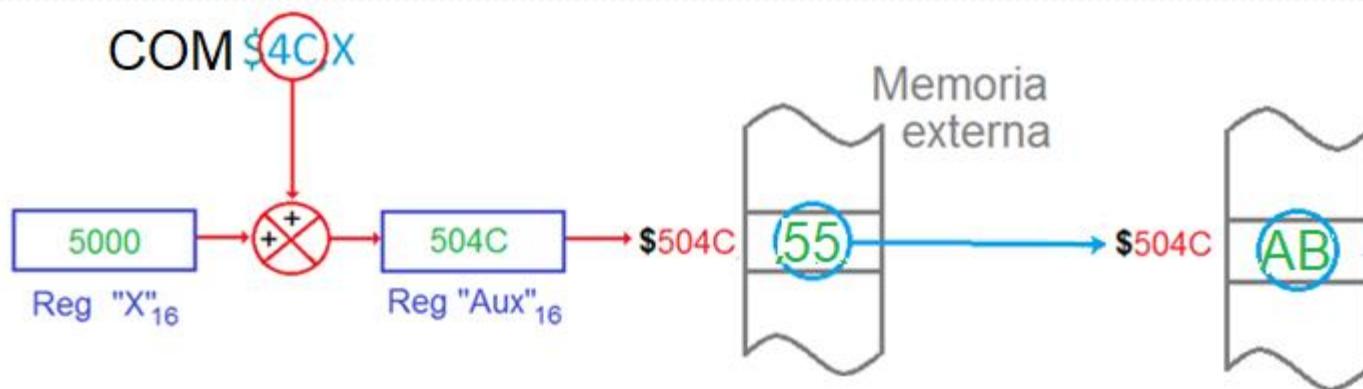
- Modo de direccionamiento **extendido**
- **Opcode:** \$70 **Ciclo:**6 **Byte:**3
- Devuelve el complemento a dos del contenido de la localidad de memoria RAM
- Actualiza banderas N, Z, V y C



NEG \$4C,X

\$00-M → M

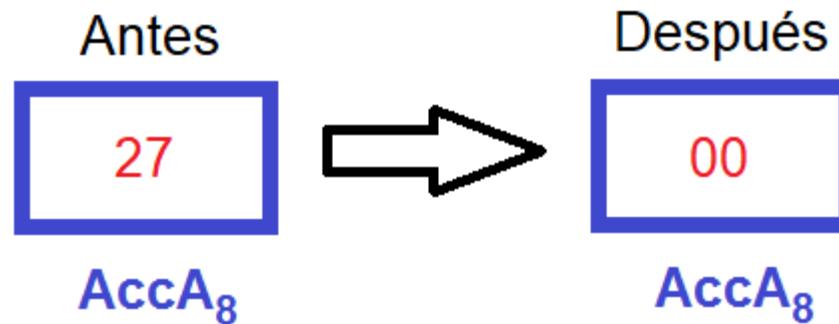
- Modo de direccionamiento **Indexado**
- **Opcode:** \$60 **Ciclo:**6 **Byte:**2
- Devuelve el complemento a dos del contenido de la localidad de memoria RAM
- Actualiza banderas N, Z, V y C



CLRA

\$00 → A

- Modo de direccionamiento **Inherente**
- **Opcode:** \$4F **Ciclo:**2 **Byte:**1
- Deja en cero el contenido del registro acumulador A.
- Actualiza banderas N=0, Z=1, V=0, C=0



CLR \$4C5A

\$00 → M

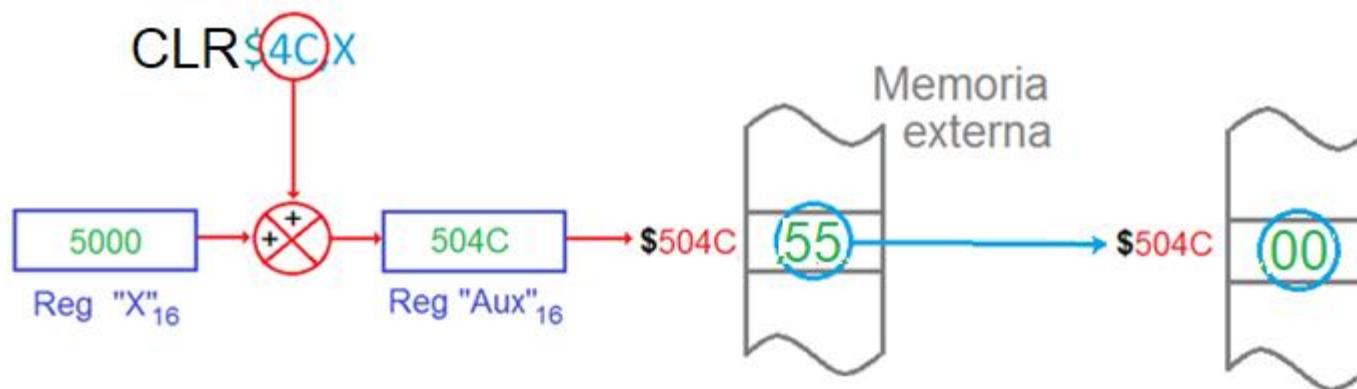
- Modo de direccionamiento **extendido**
- **Opcode:** \$70 **Ciclo:**6 **Byte:**3
- Deja en cero el contenido de la localidad de memoria RAM
- Actualiza banderas N=0, Z=1, V=0 y C=0



CLR \$4C,X

\$00 → M

- Modo de direccionamiento **Indexado**
- **Opcode:** \$6F **Ciclo:**6 **Byte:**2
- Deja en cero el contenido de la localidad de memoria RAM
- Actualiza banderas N=0, Z=1, V=0 y C=0



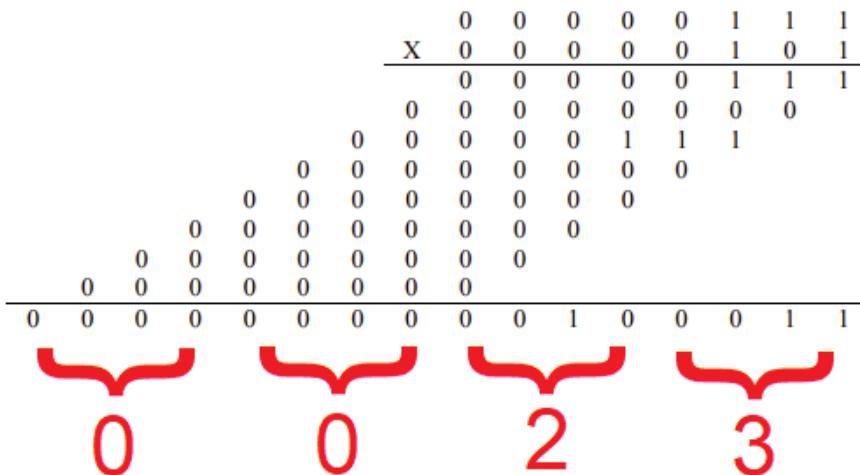
Set de instrucciones del MC68HC11

Instrucciones aritméticas de multiplicación y división

MUL

(A)(B) → D

- Modo de direccionamiento **Inherente**
- **Opcode:** \$3D **Ciclo:** 10 **Byte:** 1
- Multiplica el contenido del registro acumulador A por el contenido del registro acumulador B y deja el resultado en el contenido del registro acumulador D.
- Actualiza bandera C



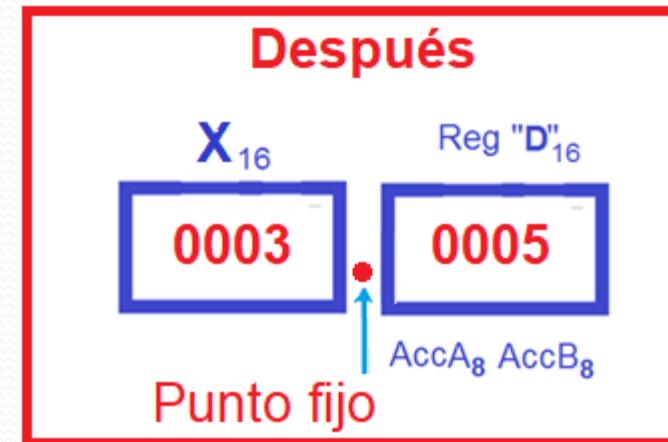
Reg "D"₁₆

$$\boxed{07}_{\text{AccA}_8} \times \boxed{05}_{\text{AccB}_8} = \boxed{0023}_{\text{AccA}_8 \text{ AccB}_8}$$

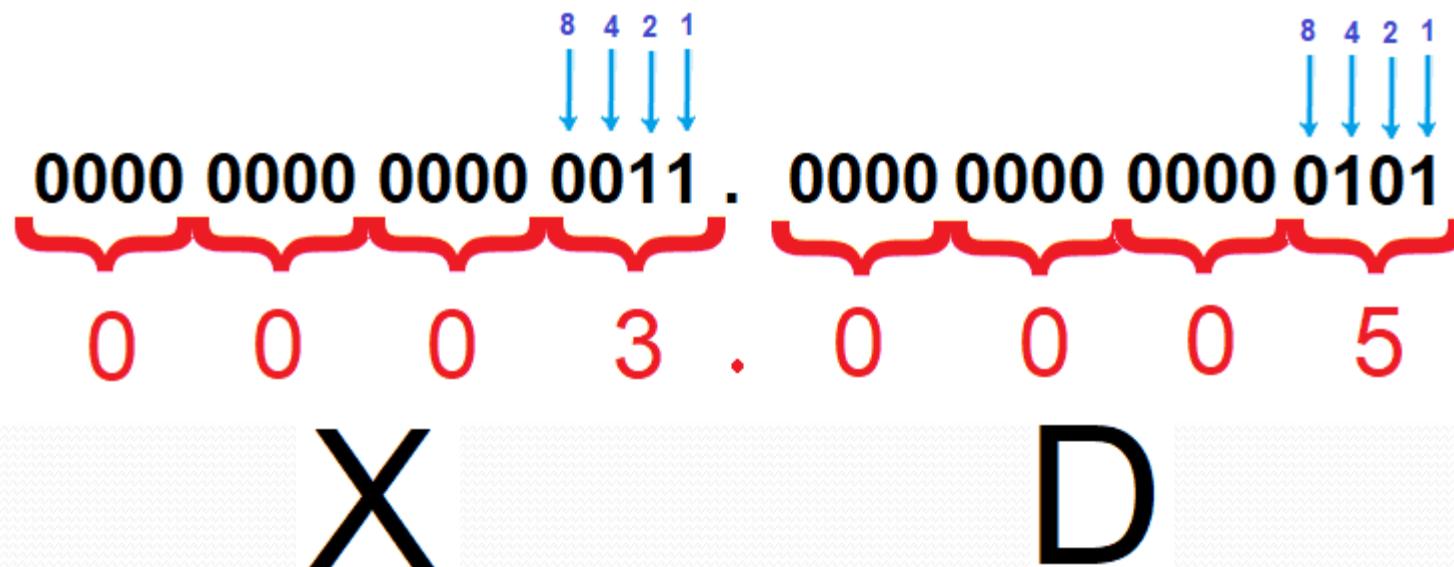
IDIV

$(D)/(X) \rightarrow X; r \rightarrow D$

- Modo de direccionamiento Inherente
- **Opcode:** \$02 **Ciclo:**41 **Byte:**1
- Divide el contenido del registro acumulador D entre el contenido del registro X y deja el resultado en el contenido del registro X y D.
- Actualiza bandera C



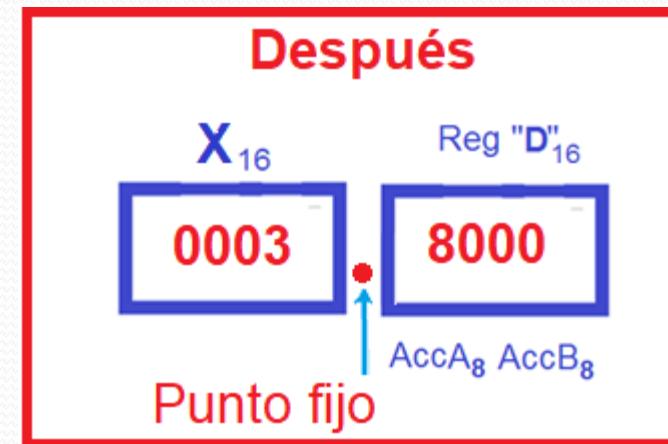
IDIV devuelve el resultado como dos cifras de 16 bits.



FDIV

$(D)/(X) \rightarrow X; r \rightarrow D$

- Modo de direccionamiento Inherente
- **Opcode:** \$03 **Ciclo:**41 **Byte:**1
- Divide el contenido del registro acumulador D entre el contenido del registro X y deja el resultado en el contenido del registro X y D.
- Actualiza bandera C



FDIV devuelve el resultado como una cifra de 32 bits.

