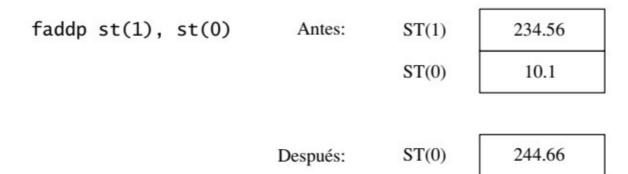
#### Intrucciones FPU

## Artiméticas

FCHS	Cambiar signo	
FADD	Sumar el origen al destino	
FSUB	Restar el origen del destino	
FSUBR	Restar el destino del origen	
FMUL	Multiplicar el origen por el destino	
FDIV	Dividir el destino entre el origen	
FDIVR	Dividir el origen entre el destino	

## Sufijo P

- Las instrucciones aritméticas aceptan el sufijo P, lo que hace que la instrucción al ejecutarse haga un pop sobre el valor de ST(0) después de realizar la operación
- MASM soporta el siguiente formato
  - FADDP ST(i),ST(0); el resultado queda en el registro ST(i)



## Prefijo FI

 Al igual que el sufijo P, el prefijo FI se puede agregar a todas las operaciones aritméticas y convierte el operando de origen al formato de punto flotante y precisión doble extendida, antes de realizar la operación con e ST(0)

## FCHS y FABS

- FCHS (cambiar signo)
  - invierte el signo del valor de punto flotante en ST(0).
- FABS (valor absoluto)
  - borra el signo del número en ST(0) para crear su valor absoluto

## Otras operaciones relacionadas con aritmética

FSQRT (raiz cuadrada),
FSCALE (escala un número),
FPREM/FPREM1 (encuentra residuo parcial),
FRNDINT (redondea a entero),
FXTRACT (extrae exponente y mantisa),

## Operaciones trascendentales

```
FPTAN (tangente parcial).
FPATAN (arcotangente parcial),
FSIN (seno)
FCOS (coseno),
FSINCOS (seno y coseno)
F2XM1 (2X - 1).
FYL2X (Y log<sub>2</sub> X)
FYL2XP1 (Y log_2 (X + 1))
```

## Comparaciónes FCMP

Instrucción	Descripción		
FCOM	Compara ST(0) con ST(1)		
FCOM m32pf	Compara ST(0) con <i>m32pf</i>		
FCOM m64pf	Compara ST(0) con m64pf		
FCOM ST(i)	Compara ST(0) con ST(i)		

#### **FCOMP**

La instrucción FCOMP lleva a cabo las mismas operaciones con los mismos tipos de operandos, y termina sacando a ST(0) de la pila.

La instrucción FCOMPP es igual que FCOMP, sólo que saca de la pila una vez más a ST(0).

## Códigos de condición

Condición	C3 (bandera Cero)	C2 (bandera Paridad)	C0 (bandera Acarreo)	Salto condicional a utilizar
ST(0) > SRC	0	0	0	JA, JNBE
ST(0) < SRC	0	0	1	JB, JNAE
ST(0) = SRC	1	0	0	JE, JZ
Sin orden <sup>a</sup>	1	1	1	(Ninguno)

<sup>&</sup>lt;sup>a</sup> Si se genera una excepción de operando aritmético inválido (debido a operandos inválidos) y la excepción está enmascarada, C3, C2 y C0 se establecen de acuerdo con la fila marcada *Sin orden*.

# Como saltar después de una comparación

- Para lograrlo se requiere:
  - 1) Usar la instrucción FNSTSW para mover la palabra de estado de la FPU hacia AX.
  - 2) Usar la instrucción SAHF para copiar AH al registro FLAGS.

#### **FCOMI**

La familia P6 de Intel incorporó la instrucción FCOMI

- Compara los valores de punto flotante y activa las banderas Cero, Paridad y Acarreo de manera directa
- Sintaxis:
  - FCOMI ST(0),ST(i)

La instrucción FCOMI no acepta operandos de memoria

## Comparación de igualdad

- No se debe comparar la igualdad entre valores de punto flotante debido a los errores de redondeo que ocurren durante los cálculos.
- Forma correcta:
  - Tomar el valor absoluto de su diferencia, | x y |, y
  - compararlo con un valor pequeño definido por el usuario, llamado épsilon

## Ejemplo

```
.data
epsilon REAL8 1.0E-12
val2 REAL8 0.0
                                   ; valor a comparar
val3 REAL8 1.001E-13
                                   ; se considera igual a val2
.code
; if( val2 == val3 ), mostrar "Los valores son iguales".
     f1d
           epsilon
     fld
         val2
    fsub
          val3
     fabs
     fcomi ST(0),ST(1)
     ja
            saltar
    mWrite <"Los valores son iguales",0dh,0ah>
saltar:
```

### Valores BCD y Enteros

- Las instrucciones que me permiten cargar valores enteros y en BCD a la pila de la FPU son:
  - FILD y FBLD