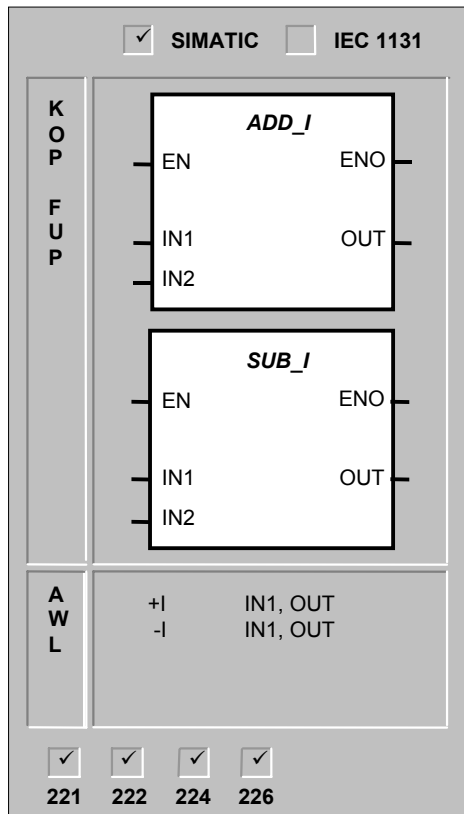


Sumar y Restar enteros de 16 bits fija

Operaciones aritméticas en coma fija

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VW, IW, QW, MW, SW, SMW, T, C, AC, LW, AIW, constante, *VD, *AC, *LD	INT
OUT	VW, IW, QW, MW, SW, SMW, T, C, LW, AC, *VD, *AC, *LD	INT

[Acceder a la memoria de la CPU](#)
[ENO](#)
[Errores](#)
[Operaciones soportadas por las CPUs S7-200](#)
[Nemotécnica SIMATIC/Internacional](#)



Las operaciones **Sumar enteros de 16 bits** y **Restar enteros de 16 bits** suman/restan dos enteros de 16 bits, arrojando un resultado de 16 bits (OUT).

En KOP y FUP: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

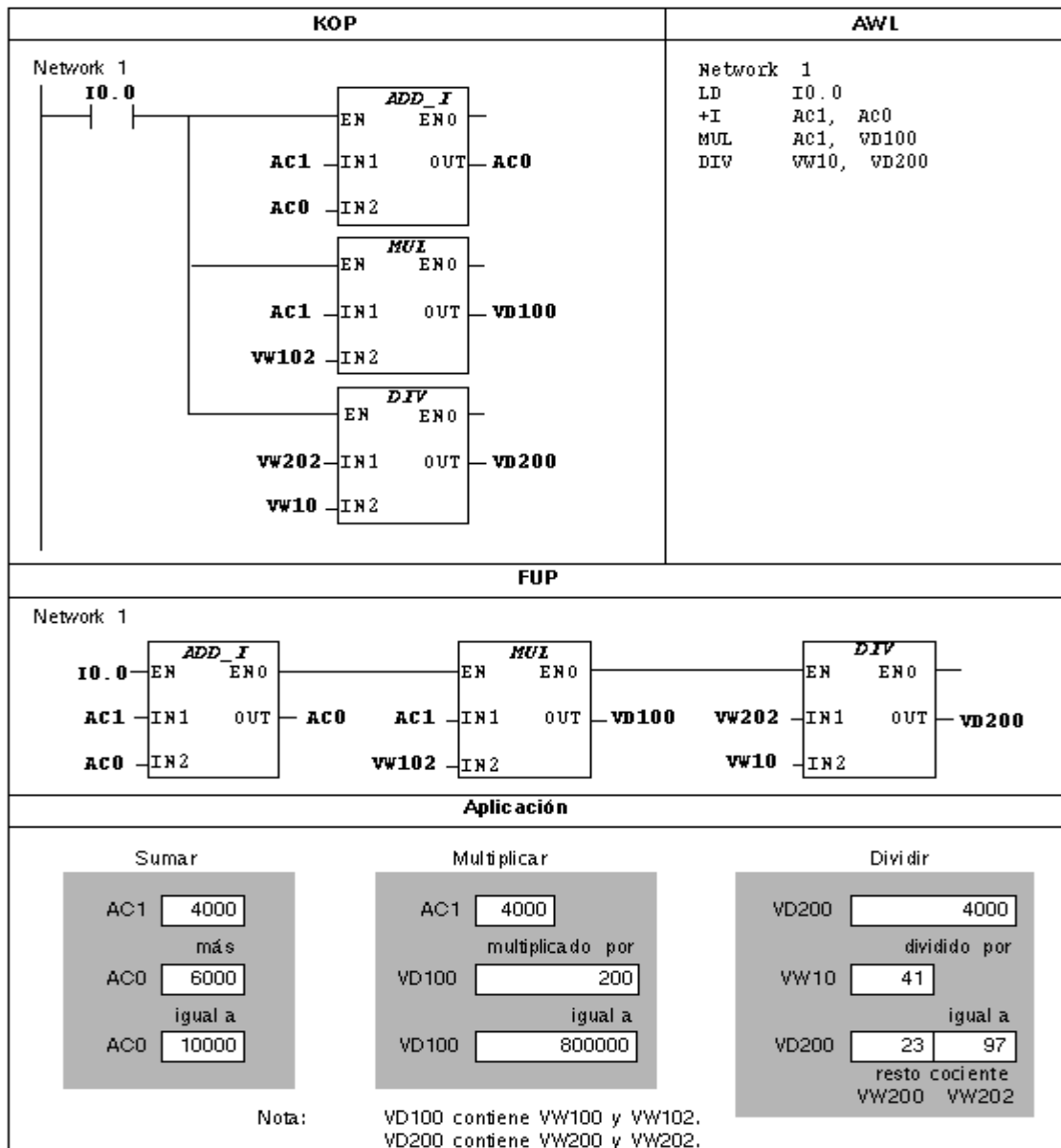
En AWL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

Condiciones de error que ponen ENO a 0:

0006 (direccionamiento indirecto), SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución)

Estas operaciones afectan a las siguientes marcas especiales:
SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Ejemplo

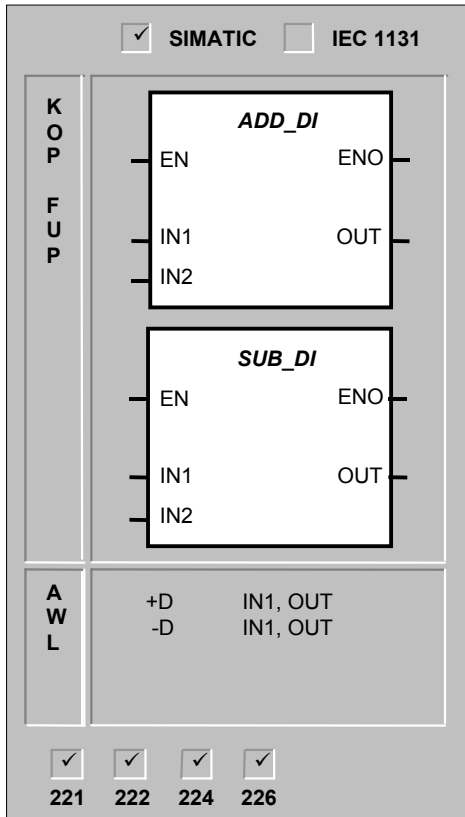


Sumar y Restar enteros de 32 bits fija

Operaciones aritméticas en coma fija

Entradas/salidas	Operandos	Tipos de datos
IN1, IN2	VD, ID, QD, MD, SMD, SD, LD, AC, HC, constante, *VD, *AC, *LD	DINT
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	DINT

[Acceder a la memoria de la CPU](#)
 [ENO](#)
 [Errores](#)
 [Operaciones soportadas por las CPUs S7-200](#)
 [Nemotécnica SIMATIC/Internacional](#)



Las operaciones **Sumar enteros de 32 bits** y **Restar enteros de 32 bits** suman/restan dos enteros de 32 bits, arrojando un resultado de 32 bits (OUT).

En KOP y FUP: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

En AWL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales: SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

Multiplicar y Dividir enteros de 16 bits a enteros de 32 bits

Operaciones aritméticas en coma fija



Entradas/salidas

IN1, IN2

OUT

[Acceder a la memoria de la CPU](#)

Operandos

VW, IW, QW, MW, SW, SMW, T, C, LW, AC, AIW, constante, *VD, *AC, *LD

VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *LD, *AC

[ENO](#)

[Errores](#)

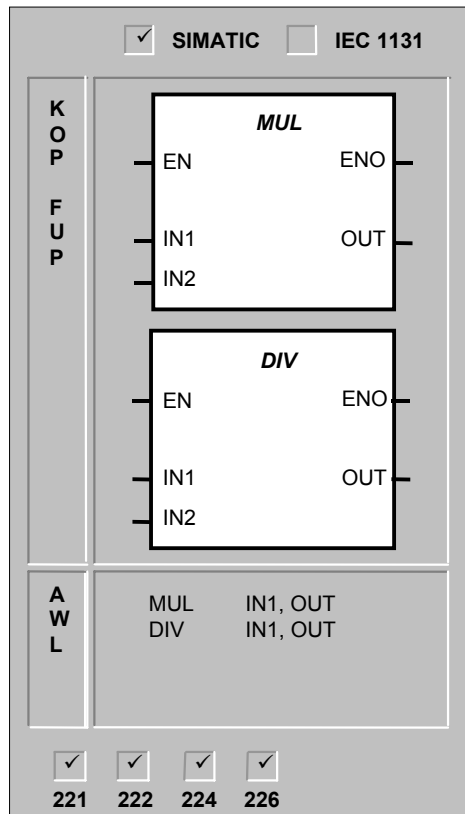
[Operaciones soportadas por las CPUs S7-200](#)

Tipos de datos

INT

DINT

[Nemotécnica SIMATIC/Internacional](#)



La operación **Multiplicar enteros de 16 bits a enteros de 32 bits** multiplica dos números enteros de 16 bits, arrojando un producto de 32 bits.

La operación **Dividir enteros de 16 bits a enteros de 32 bits** divide dos números enteros de 16 bits, arrojando un resultado de 32 bits compuesto de un cociente de 16 bits (los menos significativos) y un resto de 16 bits (los más significativos).

En la operación AWL de multiplicación, la palabra menos significativa (16 bits) del OUT de 32 bits se utiliza como uno de los factores.

En la operación AWL de división, la palabra menos significativa (16 bits) del OUT de 32 bits se utiliza como dividendo.

En KOP y FUP: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

En AWL: $IN1 * IN2 = OUT$
 $OUT / IN1 = OUT$

Condiciones de error que ponen ENO a 0:

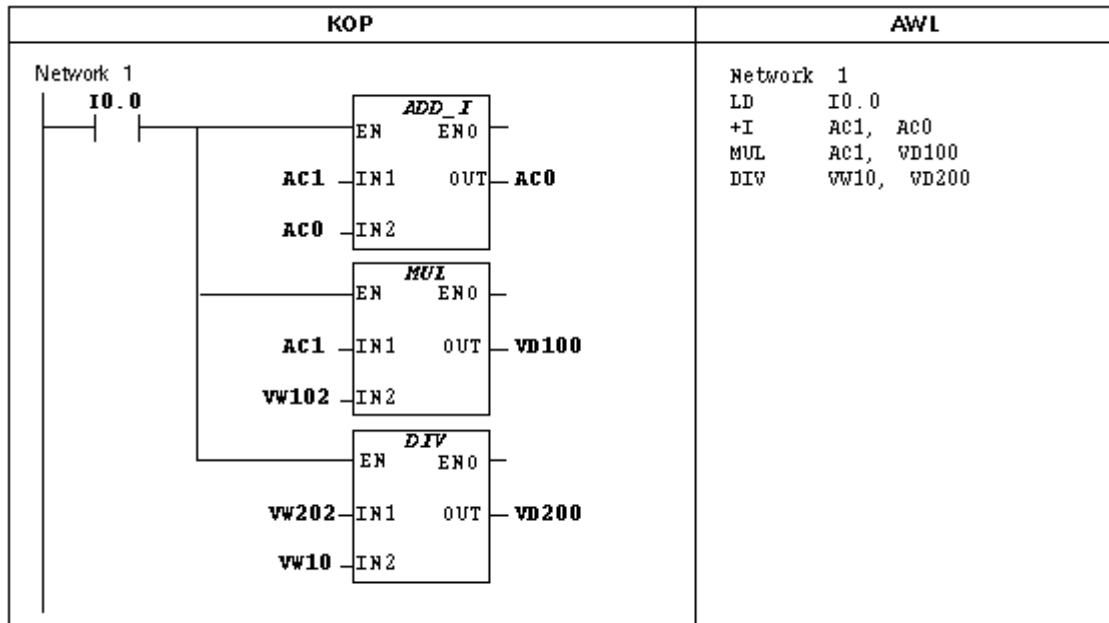
SM1.1 (desbordamiento), SM1.3 (división por cero), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:

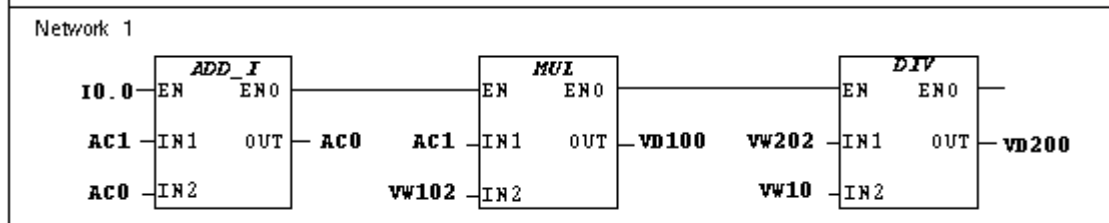
SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo); SM1.3 (división por cero)

Si se activa SM1.3 (división por cero) durante una operación de división, permanecerán inalterados los demás bits aritméticos de estado, así como los operandos de entrada originales. En otro caso, todos los bits aritméticos de estado asistidos contendrán el estado válido al finalizar la operación aritmética.

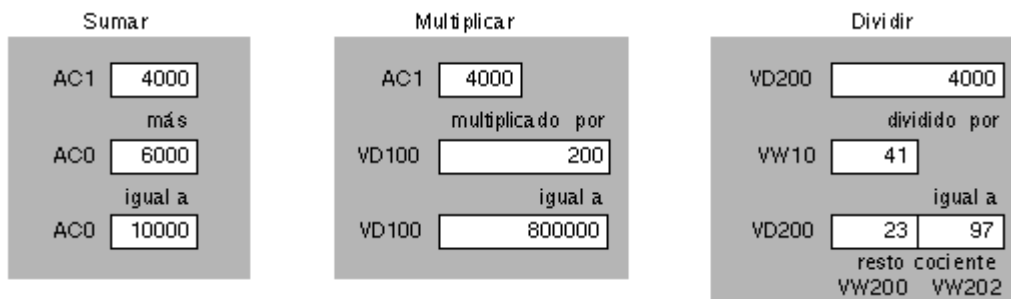
Ejemplo



FUP



Aplicación



Nota: VD100 contiene VW100 y VW102.
VD200 contiene VW200 y VW202.

Incrementar y Decrementar byte Operaciones aritméticas en coma fija

Entradas/salidas

IN
OUT

Operandos

VB, IB, QB, MB, SB, SMB, LB, AC, constante, *VD, *AC, *LD
VB, IB, QB, MB, SB, SMB, LB, AC, *VD, *AC, *LD

Tipos de datos

BYTE
BYTE

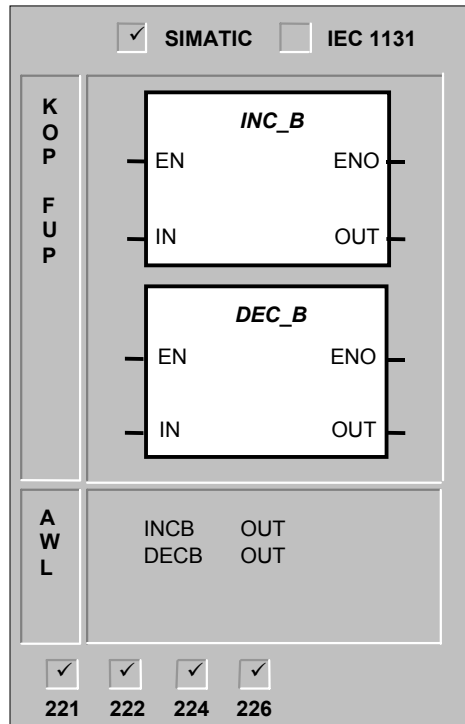
[Acceder a la memoria de la CPU](#)

[ENO](#)

[Errores](#)

[Operaciones soportadas por las CPUs S7-200](#)

[Nemotécnica SIMATIC/Internacional](#)



Las operaciones **Incrementar byte** y **Decrementar byte** suman/restan 1 al byte de entrada (IN) y depositan el resultado en la variable indicada por OUT.

Las operaciones Incrementar byte y Decrementar byte no llevan signo.

En KOP y FUP: $IN + 1 = OUT$
 $IN - 1 = OUT$

En AWL: $OUT + 1 = OUT$
 $OUT - 1 = OUT$

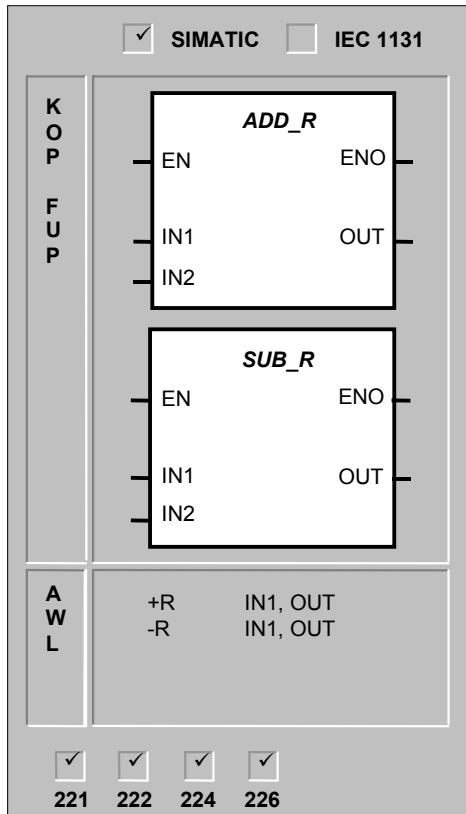
Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
SM1.0 (cero); SM1.1 (desbordamiento)

Sumar y Restar reales Operaciones aritméticas en coma flotante

Entradas/salidas	Operandos	Tipos de datos	
IN1, IN2	VD, ID, QD, MD, SD, SMD, AC, LD, constante, *VD, *AC, *LD	REAL	
OUT	VD, ID, QD, MD, SD, SMD, AC, LD, *VD, *AC, *LD	REAL	
Acceder a la memoria de la CPU	Errores	Operaciones soportadas por las CPUs S7-200	Nemotécnica SIMATIC/Internacional



Las operaciones **Sumar reales** y **Restar reales** suman/restan dos números reales de 32 bits, dando como resultado un número real de 32 bits (OUT).

En KOP y FUP: $IN1 + IN2 = OUT$
 $IN1 - IN2 = OUT$

En AWL: $IN1 + OUT = OUT$
 $OUT - IN1 = OUT$

Condiciones de error que ponen ENO a 0:

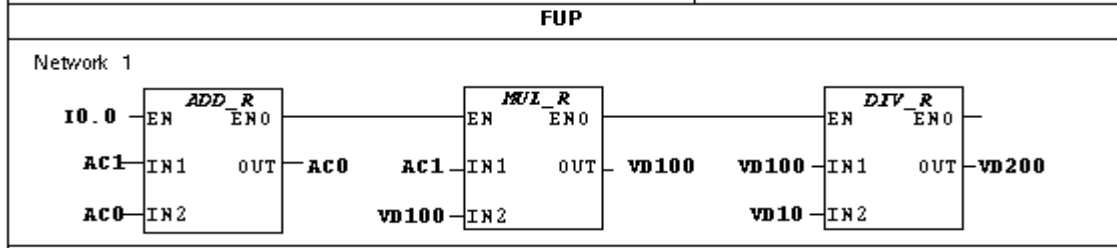
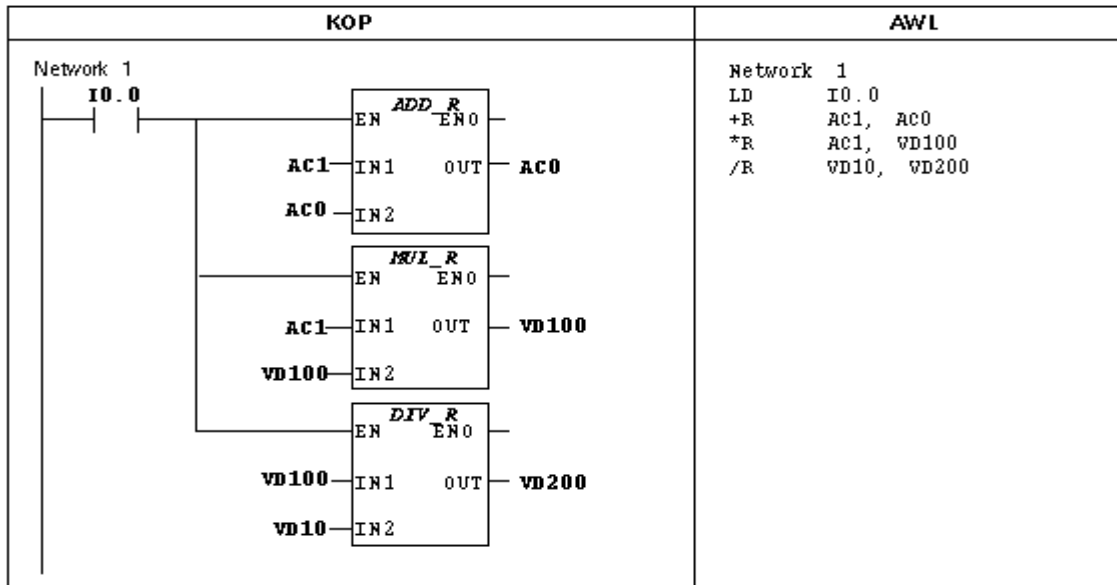
SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:
 SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

SM1.1 se utiliza para indicar errores de desbordamiento y valores no válidos. Si se activa SM1.1, el estado de SM1.0 y de SM1.2 no será válido y no se alterarán los operandos de entrada originales. Si SM1.1 y SM1.2 no se activan durante una operación de división, la operación aritmética habrá finalizado con un resultado válido, y tanto SM1.0 como SM1.2 contendrán un estado válido.

Nota: Los números reales (o números en coma flotante) se representan en el formato descrito en la norma ANSI/IEEE 754-1985 (precisión sencilla). Para obtener más información al respecto, consulte dicha norma.

Ejemplo

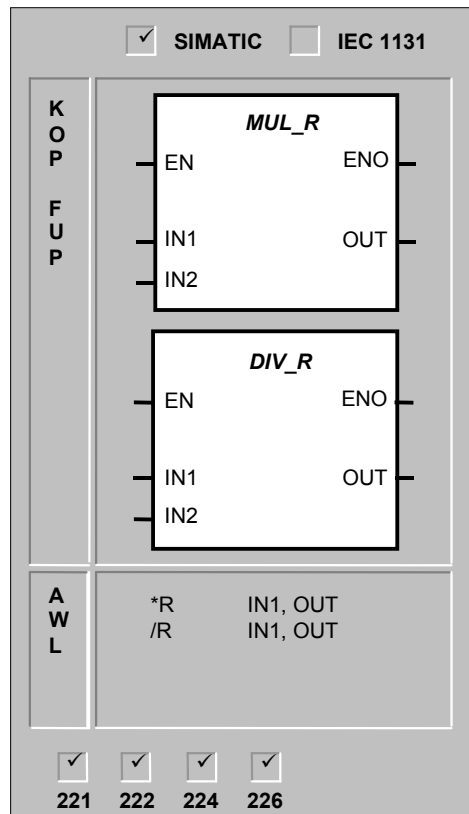


Aplicación

<p>Sumar</p> <table style="width: 100%; border-collapse: collapse;"> <tr><td>AC1</td><td><input type="text" value="4000.0"/></td></tr> <tr><td colspan="2" style="text-align: center;">más</td></tr> <tr><td>AC0</td><td><input type="text" value="6000.0"/></td></tr> <tr><td colspan="2" style="text-align: center;">igual a</td></tr> <tr><td>AC0</td><td><input type="text" value="10000.0"/></td></tr> </table>	AC1	<input type="text" value="4000.0"/>	más		AC0	<input type="text" value="6000.0"/>	igual a		AC0	<input type="text" value="10000.0"/>	<p>Multiplicar</p> <table style="width: 100%; border-collapse: collapse;"> <tr><td>AC1</td><td><input type="text" value="400.00"/></td></tr> <tr><td colspan="2" style="text-align: center;">multiplicado por</td></tr> <tr><td>VD100</td><td><input type="text" value="200.0"/></td></tr> <tr><td colspan="2" style="text-align: center;">igual a</td></tr> <tr><td>VD100</td><td><input type="text" value="800000.0"/></td></tr> </table>	AC1	<input type="text" value="400.00"/>	multiplicado por		VD100	<input type="text" value="200.0"/>	igual a		VD100	<input type="text" value="800000.0"/>	<p>Dividir</p> <table style="width: 100%; border-collapse: collapse;"> <tr><td>VD200</td><td><input type="text" value="4000.0"/></td></tr> <tr><td colspan="2" style="text-align: center;">dividido por</td></tr> <tr><td>VD10</td><td><input type="text" value="41.0"/></td></tr> <tr><td colspan="2" style="text-align: center;">igual a</td></tr> <tr><td>VD200</td><td><input type="text" value="97.5609"/></td></tr> </table>	VD200	<input type="text" value="4000.0"/>	dividido por		VD10	<input type="text" value="41.0"/>	igual a		VD200	<input type="text" value="97.5609"/>
AC1	<input type="text" value="4000.0"/>																															
más																																
AC0	<input type="text" value="6000.0"/>																															
igual a																																
AC0	<input type="text" value="10000.0"/>																															
AC1	<input type="text" value="400.00"/>																															
multiplicado por																																
VD100	<input type="text" value="200.0"/>																															
igual a																																
VD100	<input type="text" value="800000.0"/>																															
VD200	<input type="text" value="4000.0"/>																															
dividido por																																
VD10	<input type="text" value="41.0"/>																															
igual a																																
VD200	<input type="text" value="97.5609"/>																															

Multiplicar y Dividir reales Operaciones aritméticas en coma flotante

Entradas/salidas	Operandos	Tipos de datos	
IN1, IN2	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL	
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL	
Acceder a la memoria de la CPU	ENO Errores	Operaciones soportadas por las CPUs S7-200	Nemotécnica SIMATIC/Internacional



La operación **Multiplicar reales** multiplica dos números reales de 32 bits, dando como resultado un número real de 32 bits (OUT).

La operación **Dividir reales** divide dos números reales de 32 bits, dando como resultado un cociente de número real de 32 bits.

En KOP y FUP: $IN1 * IN2 = OUT$
 $IN1 / IN2 = OUT$

En AWL: $IN1 * IN2 = OUT$
 $OUT / IN1 = OUT$

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM1.3 (división por cero), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:

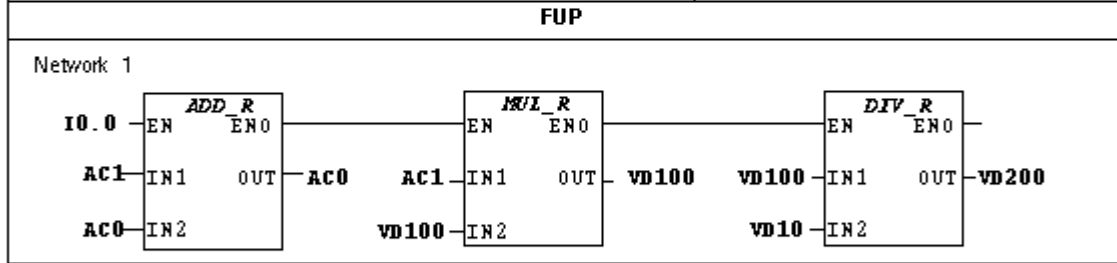
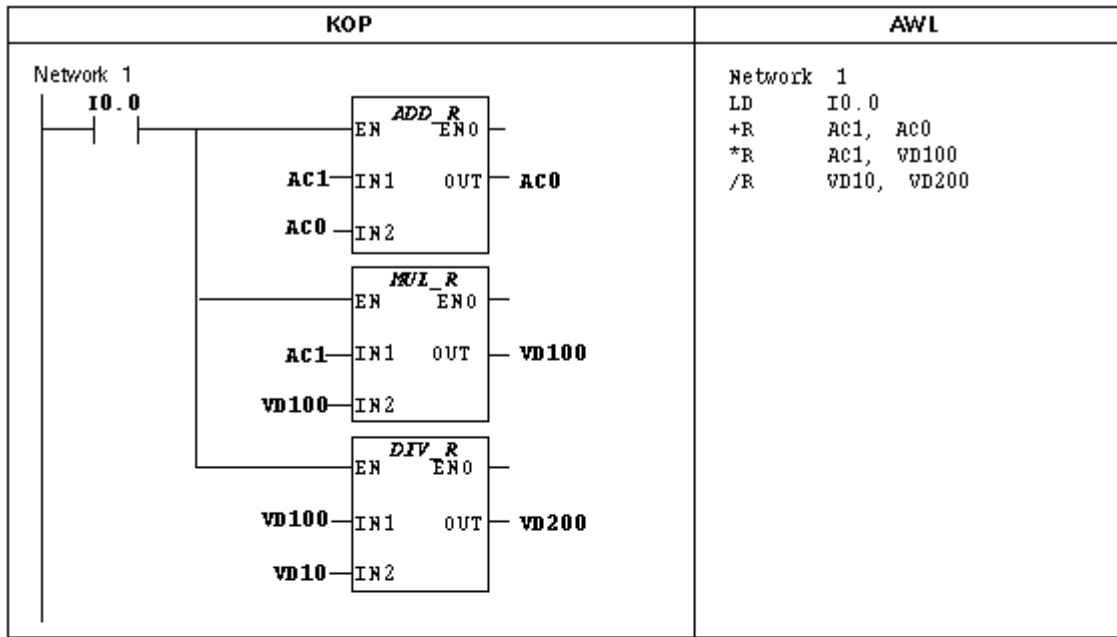
SM1.0 (cero); SM1.1 (desbordamiento o valor no válido generado durante la operación o parámetro de entrada no válido); SM1.2 (negativo); SM1.3 (división por cero)

Si se activa SM1.3 durante una operación de división, permanecerán inalterados los demás bits aritméticos de estado, así como los operandos de entrada originales. SM1.1 se utiliza para indicar errores de desbordamiento y valores no válidos. Si se activa SM1.1, el estado de SM1.0 y de SM1.2 no será válido y no se alterarán los operandos de entrada originales. Si SM1.1 y SM1.3 no se activan durante una operación de división, la operación aritmética habrá finalizado con un resultado válido, y tanto SM1.0 como SM1.2 contendrán un estado válido.

Nota

Los números reales (o números en coma flotante) se representan en el formato descrito en la norma ANSI/IEEE 754-1985 (precisión sencilla). Para obtener más información al respecto, consulte dicha norma.

Ejemplo



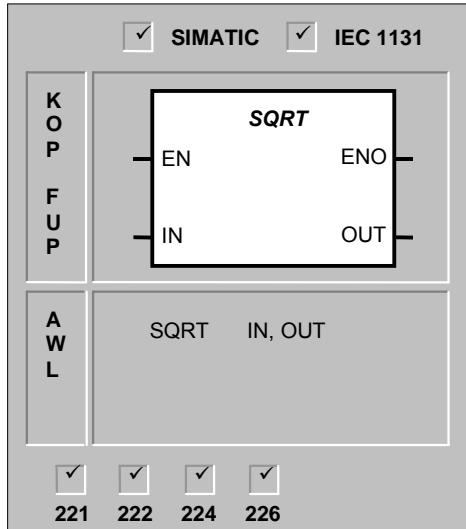
Aplicación

<p>Sumar</p> <p>AC1 <input type="text" value="4000.0"/></p> <p>más</p> <p>AC0 <input type="text" value="6000.0"/></p> <p>igual a</p> <p>AC0 <input type="text" value="10000.0"/></p>	<p>Multiplicar</p> <p>AC1 <input type="text" value="400.00"/></p> <p>multiplicado por</p> <p>VD100 <input type="text" value="200.0"/></p> <p>igual a</p> <p>VD100 <input type="text" value="800000.0"/></p>	<p>Dividir</p> <p>VD200 <input type="text" value="4000.0"/></p> <p>dividido por</p> <p>VD10 <input type="text" value="41.0"/></p> <p>igual a</p> <p>VD200 <input type="text" value="97.5609"/></p>
---	--	---

Raíz cuadrada Operaciones aritméticas en coma flotante (SIMATIC)

Funciones numéricas (IEC)

Entradas/salidas	Operandos	Tipos de datos
IN	VD, ID, QD, MD, SMD, SD, LD, AC, constante, *VD, *AC, *LD	REAL
OUT	VD, ID, QD, MD, SMD, SD, LD, AC, *VD, *AC, *LD	REAL
Acceder a la memoria de la CPU	ENO Errores Operaciones soportadas por las CPUs S7-200	Nemotécnica SIMATIC/Internacional



La operación **Raíz cuadrada** extrae la raíz cuadrada de un número real de 32 bits (IN), dando como resultado un número real de 32 bits (OUT), como muestra la ecuación:

$$\sqrt{IN} = OUT$$

Condiciones de error que ponen ENO a 0:

SM1.1 (desbordamiento), SM4.3 (tiempo de ejecución), 0006 (direccionamiento indirecto)

Estas operaciones afectan a las siguientes marcas especiales:

SM1.0 (cero); SM1.1 (desbordamiento); SM1.2 (negativo)

SM1.1 se utiliza para indicar errores de desbordamiento y valores no válidos. Si se activa SM1.1, el estado de SM1.0 y de SM1.2 no será válido y no se alterarán los operandos de entrada originales. Si SM1.1 y SM1.2 no se activan durante una operación de división, la operación aritmética habrá finalizado con un resultado válido, y tanto SM1.0 como SM1.2 contendrán un estado válido.