

## 4. RECHENFUNKTIONEN

Mit den Rechenfunktionen werden zwei digitale Werte addiert oder subtrahiert.

### 4.1 ADDITION VON DUALZAHLEN

Sollen Dualzahlen addiert werden, so gilt folgende Rechenregel:

$$\begin{aligned}0 + 0 &= 0 \\0 + 1 &= 1 \\1 + 0 &= 1 \\1 + 1 &= 0\end{aligned}$$

Hierbei wird eine 1 auf die nächst höhere Stelle übernommen

$$1 + 1 + 1 = 1$$

Hierbei wird eine 1 auf die nächst höhere Stelle übernommen

Beispiel: Es soll die Dualzahl 1101100 mit der Dualzahl 1011010 addiert werden.

01101100	Probe:	108
+ 01011010		+ 90
-----		-----
1111	Übertrag	198
-----		
11000110		
=====		

**Übung:**

Es sollen die Dualzahlen 01101011 und 00111001 addiert werden.

1101 0011 0011 0110 _____ _____ =====	Übertrag	Probe:    ---  +    --- _____  =====
---	----------	---

**Übung:**

Es sollen die Dualzahlen 10101110 und 0111 0100 addiert werden.

Übung:

1010 1110 0111 0100 _____ _____ =====	Übertrag	Probe:    ---  +    --- _____  =====
---	----------	---

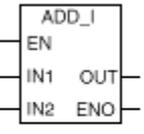
**Übung:**

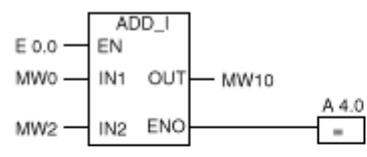
Es sollen die Dualzahlen 11010011 und 00110110 addiert werden.

1101 0011 0011 0110 _____ _____ =====	Übertrag	Probe:    ---  +    --- _____  =====
---	----------	---

## Ganze Zahlen addieren (16 Bit)

Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen addieren (16 Bit)*. Diese Operation addiert die Eingänge IN1 und IN2. Das Ergebnis kann an Ausgang OUT abgefragt werden. Liegt das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (16 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	INT	E, A, M, D, L oder Konstante	Erster Summand
	IN2	INT	E, A, M, D, L oder Konstante	Zweiter Summand
	OUT	INT	E, A, M, D, L	Additionsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



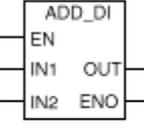
The diagram shows a normally open contact labeled 'E 0.0' connected to the EN input of the 'ADD\_I' function block. The IN1 input is connected to 'MW0' and the IN2 input to 'MW2'. The OUT output is connected to 'MW10'. The ENO output is connected to a coil labeled 'A 4.0'.

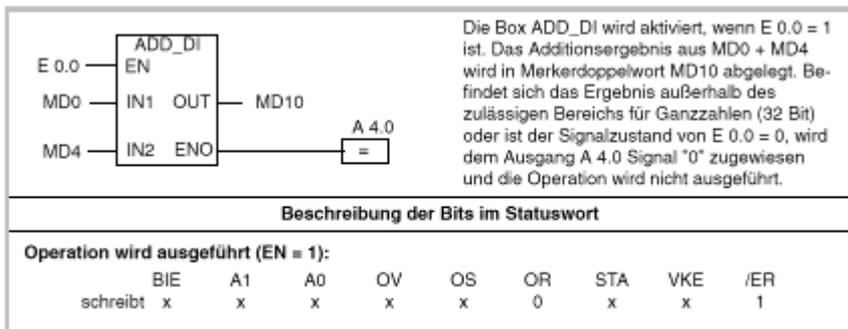
Die Box ADD\_I wird aktiviert, wenn E 0.0 = 1 ist. Das Additionsergebnis aus MW0 + MW2 wird in Merkerwort MW10 abgelegt. Befindet sich das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (16 Bit) oder ist der Signalzustand von E 0.0 = 0, wird dem Ausgang A 4.0 Signal "0" zugewiesen und die Operation wird nicht ausgeführt.

Beschreibung der Bits im Statuswort									
<b>Operation wird ausgeführt (EN = 1):</b>									
	BIE	A1	A0	OV	OS	OR	STA	VKE	/ER
schreibt	x	x	x	x	x	0	x	x	1

## Ganze Zahlen addieren (32 Bit)

Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen addieren (32 Bit)*. Diese Operation addiert die Eingänge IN1 und IN2. Das Ergebnis kann an Ausgang OUT abgefragt werden. Liegt das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (32 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	DINT	E, A, M, D, L oder Konstante	Erster Summand
	IN2	DINT	E, A, M, D, L oder Konstante	Zweiter Summand
	OUT	DINT	E, A, M, D, L	Additionsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



## 4.2 SUBTRAKTION VON DUALZAHLEN

Sollen Dualzahlen subtrahiert werden, so gilt folgende Rechenregel:

$0 - 0 = 0$	
$0 - 1 = 1$	Hierbei wird eine -1 auf die nächst höhere Stelle übernommen
$1 - 0 = 1$	
$1 - 1 = 0$	
$0 - 1 - 1 = 0$	Hierbei wird eine -1 auf die nächst höhere Stelle übernommen
$1 - 1 - 1 = 1$	Hierbei wird eine -1 auf die nächst höhere Stelle übernommen

Beispiel: Es soll die Dualzahl 10010110 mit der Dualzahl 111100 subtrahiert werden.

$$\begin{array}{r}
 1001\ 0110 \\
 - 0011\ 1100 \\
 \hline
 1111\ \text{Übertrag} \\
 \hline
 0101\ 1010 \\
 \hline
 \text{=====}
 \end{array}$$

$$\begin{array}{r}
 \text{Probe: } 150 \\
 - 60 \\
 \hline
 90
 \end{array}$$

**Übung:**

Es sollen die Dualzahlen 11111001 und 0011 0110 subtrahiert werden.

1111 1001	Probe:	---
0011 0110		---
_____	Übertrag	- ---
_____		_____
=====		=====

**Übung:**

Es sollen die Dualzahlen 1000 1010 und 0101 0111 subtrahiert werden.

Übung:

1000 1010	Probe:	---
0101 0111		---
_____	Übertrag	- ---
_____		_____
=====		=====

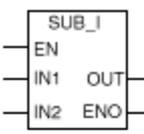
**Übung:**

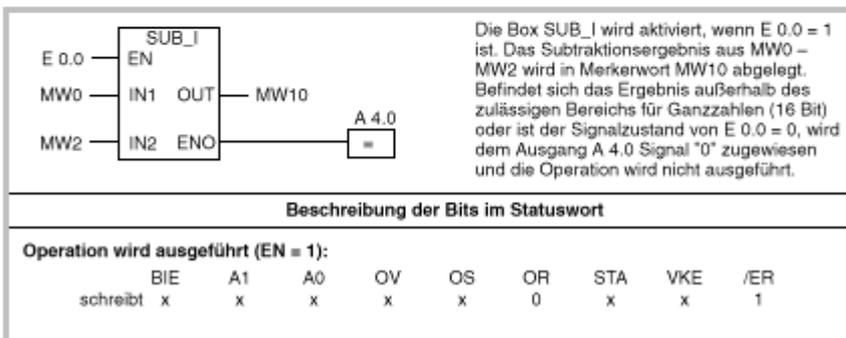
Es sollen die Dualzahlen 1111 0011 und 0101 1110 subtrahiert werden.

1111 0011	Probe:	---
0101 1110		---
_____	Übertrag	- ---
_____		_____
=====		=====

## Ganze Zahlen subtrahieren (16 Bit)

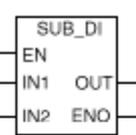
Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen subtrahieren (16 Bit)*. Diese Operation subtrahiert Eingang IN2 von IN1. Das Ergebnis kann an Ausgang OUT abgefragt werden. Liegt das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (16 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

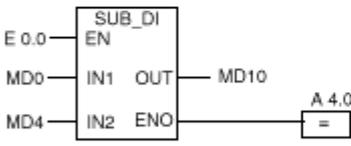
FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	INT	E, A, M, D, L oder Konstante	Minuend
	IN2	INT	E, A, M, D, L oder Konstante	Subtrahend
	OUT	INT	E, A, M, D, L	Subtraktionsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



## Ganze Zahlen subtrahieren (32 Bit)

Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen subtrahieren (32 Bit)*. Diese Operation subtrahiert Eingang IN2 von IN1. Das Ergebnis kann an Ausgang OUT abgefragt werden. Liegt das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (32 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	DINT	E, A, M, D, L oder Konstante	Minuend
	IN2	DINT	E, A, M, D, L oder Konstante	Subtrahend
	OUT	DINT	E, A, M, D, L	Subtraktionsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



The diagram shows a SUB\_DI box with EN connected to E 0.0, IN1 to MD0, and IN2 to MD4. The OUT output is connected to MD10. The ENO output is connected to a coil labeled A 4.0.

Die Box SUB\_DI wird aktiviert, wenn E 0.0 = 1 ist. Das Subtraktionsergebnis aus MD0 – MD4 wird in Merkerdoppelwort MD10 abgelegt. Befindet sich das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (32 Bit) oder ist der Signalzustand von E 0.0 = 0, wird dem Ausgang A 4.0 Signal "0" zugewiesen und die Operation wird nicht ausgeführt.

---

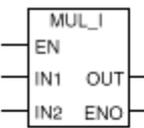
**Beschreibung der Bits im Statuswort**

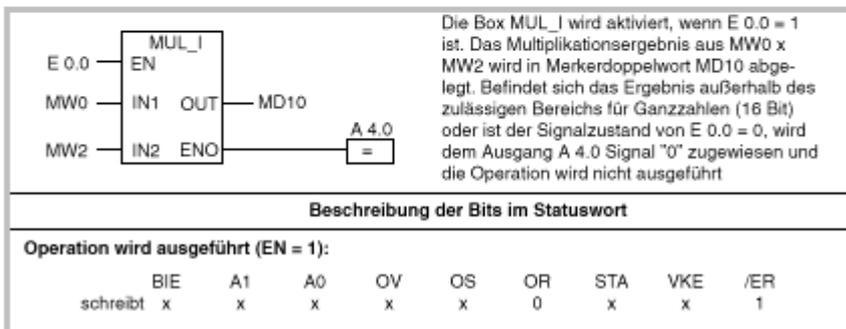
**Operation wird ausgeführt (EN = 1):**

	BIE	A1	A0	OV	OS	OR	STA	VKE	/ER
schreibt	x	x	x	x	x	0	x	x	1

### 4.3 GANZE ZAHLEN MULTIPLIZIEREN (16 BIT)

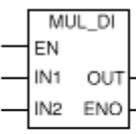
Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen multiplizieren (16 Bit)*. Diese Operation multipliziert Eingang IN1 mit IN2. Das Ergebnis kann an Ausgang OUT abgefragt werden. Liegt das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (16 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

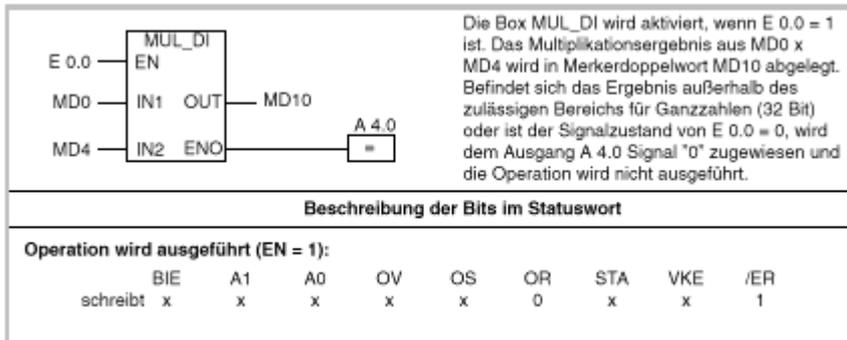
FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	INT	E, A, M, D, L oder Konstante	Multiplikand
	IN2	INT	E, A, M, D, L oder Konstante	Multiplikator
	OUT	DINT	E, A, M, D, L	Multiplikationsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



## Ganze Zahlen multiplizieren (32 Bit)

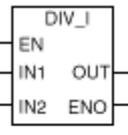
Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen multiplizieren (32 Bit)*. Diese Operation multipliziert Eingang IN1 mit IN2. Das Ergebnis kann an Ausgang OUT abgefragt werden. Liegt das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (32 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

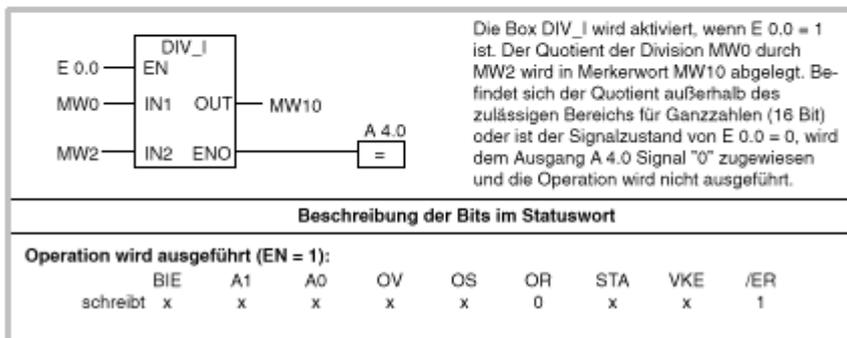
FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	DINT	E, A, M, D, L oder Konstante	Multiplikand
	IN2	DINT	E, A, M, D, L oder Konstante	Multiplikator
	OUT	DINT	E, A, M, D, L	Multiplikationsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



#### 4.4 GANZE ZAHLEN DIVIDIEREN (16 BIT)

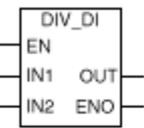
Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen dividieren (16 Bit)*. Diese Operation dividiert Eingang IN1 durch IN2. Der Quotient dieser Division (ganzzahliger Anteil) kann an Ausgang OUT abgefragt werden. Der Divisionsrest kann nicht abgefragt werden. Liegt der Quotient außerhalb des zulässigen Bereichs für Ganzzahlen (16 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

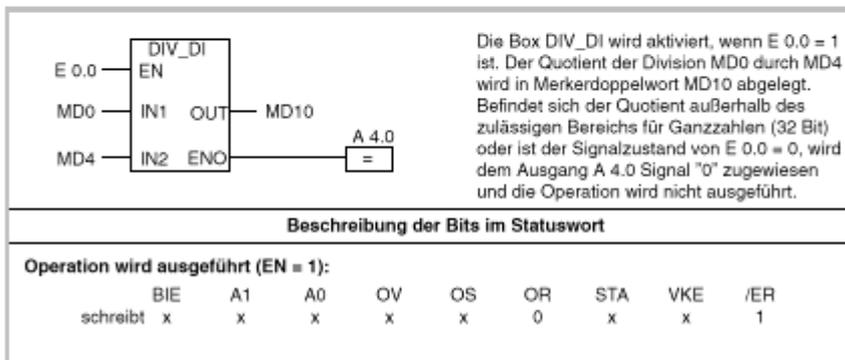
FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	INT	E, A, M, D, L oder Konstante	Dividend
	IN2	INT	E, A, M, D, L oder Konstante	Divisor
	OUT	INT	E, A, M, D, L	Divisionsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



## Ganze Zahlen dividieren (32 Bit)

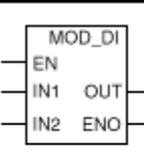
Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Ganze Zahlen dividieren (32 Bit)*. Diese Operation dividiert Eingang IN1 durch IN2. Der Quotient dieser Division (ganzzahliger Anteil) kann an Ausgang OUT abgefragt werden. Die Operation *Ganze Zahlen dividieren (32 Bit)* legt den Quotienten als einfachen 32-Bit-Wert im DINT-Format ab und erzeugt keinen Divisionsrest. Liegt der Quotient außerhalb des zulässigen Bereichs für Ganzzahlen (32 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

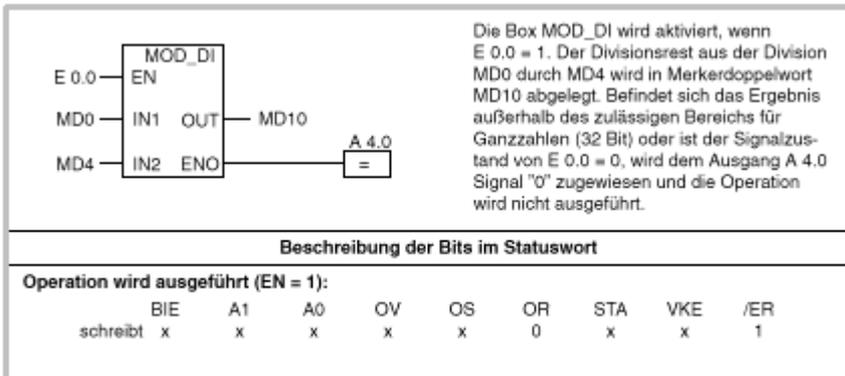
FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	DINT	E, A, M, D, L oder Konstante	Dividend
	IN2	DINT	E, A, M, D, L oder Konstante	Divisor
	OUT	DINT	E, A, M, D, L	Divisionsergebnis
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



## Divisionsrest gewinnen (32 Bit)

Ein Signalzustand von "1" am Freigabeeingang (EN) aktiviert die Operation *Divisionsrest gewinnen (32 Bit)*. Diese Operation dividiert Eingang IN1 durch IN2. Der Divisionsrest kann an Ausgang OUT abgefragt werden. Liegt das Ergebnis außerhalb des zulässigen Bereichs für Ganzzahlen (32 Bit), haben das OV-Bit und OS-Bit den Wert "1" und ENO den Wert "0".

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, D, L, T, Z	Freigabeeingang
	IN1	DINT	E, A, M, D, L oder Konstante	Dividend
	IN2	DINT	E, A, M, D, L oder Konstante	Divisor
	OUT	DINT	E, A, M, D, L	Divisionsrest
	ENO	BOOL	E, A, M, D, L	Freigabeausgang



#### 4.5 SCHIEBEOPERATIONEN

Mit den Schiebeoperationen können Sie den Inhalt von Eingang IN bitweise nach links oder rechts schieben. Ein Schieben um  $n$  Bits nach links multipliziert den Inhalt von Eingang IN mit  $2^n$ ; ein Schieben um  $n$  Bits nach rechts dividiert den Inhalt von Eingang IN durch  $2^n$ . Wenn Sie also beispielsweise das binäre Äquivalent des Dezimalwerts 3 um 3 Bits nachlinks schieben, so ergibt sich das binäre Äquivalent des Dezimalwerts 24.

Schieben Sie das binäre Äquivalent des Dezimalwerts 16 um 2 Bits nach rechts, so ergibt sich das binäre Äquivalent des Dezimalwerts 4. Die Zahl, die Sie für den Eingangsparameter angeben, zeigt an, um wie viele Bits geschoben werden soll. Die Stellen, die durch die Schiebeoperation frei werden, werden entweder mit Nullen oder mit dem Signalzustand des Vorzeichenbits ("0" steht für positiv, "1" steht für negativ) aufgefüllt. Das zuletzt geschobene Bit wird in das Bit A1 des Statuswort geladen. Die Bits A0 und OV werden auf "0" zurückgesetzt. Mit den Sprungoperationen können Sie das Bit A1 im Statuswort auswerten.

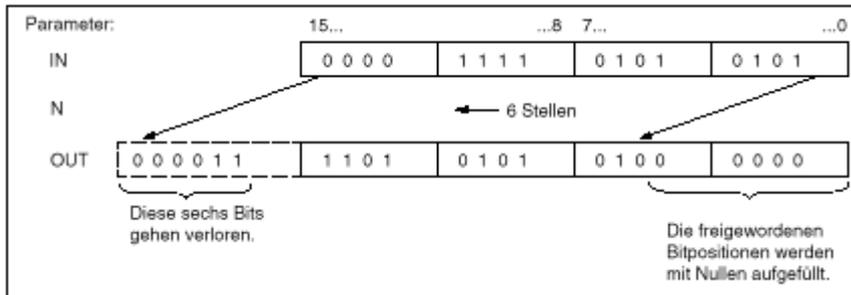
Folgende Schiebeoperationen stehen Ihnen zur Verfügung:

- \_ 16 Bit links schieben, 32 Bit links schieben
- \_ 16 Bit rechts schieben, 32 Bit rechts schieben
- \_ Ganzzahl (16 Bit) rechts schieben, Ganzzahl (32 Bit) rechts schieben

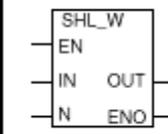
Die Operation *16 Bit links schieben* wird durch den Signalzustand "1" am Freigabeeingang (EN) aktiviert und schiebt die Bits 0 bis 15 von Eingang IN bitweise nach links.

Eingang N gibt an, um wie viele Bits geschoben wird. Ist N größer als 16, schreibt der Befehl in Ausgang OUT eine 0 und setzt die Bits A0 und OV des Statusworts auf "0". Die rechts frei werdenden Bitpositionen werden mit Nullen aufgefüllt. Das Ergebnis der Schiebeoperation kann an Ausgang OUT abgefragt werden.

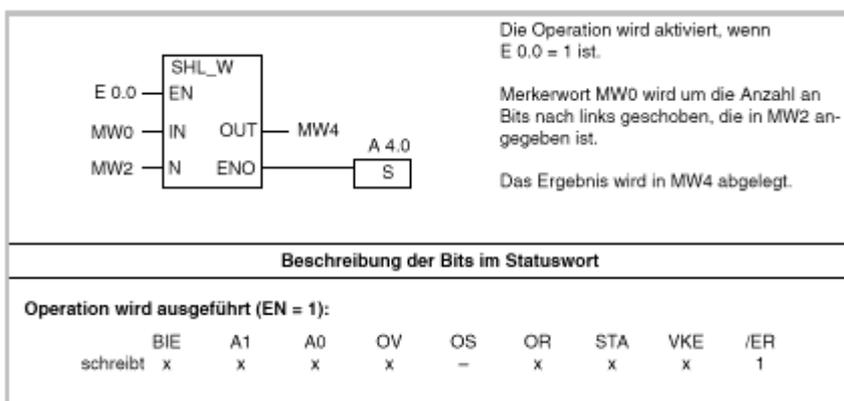
Die ausgelöste Operation setzt bei N ungleich Null das A0- und OV-Bit des Statusworts auf "0" zurück. ENO hat den gleichen Signalzustand wie EN.



Schieben der Bits von Eingang IN um sechs Bitpositionen nach links

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, L, D, T, Z	Freigabeeingang
	IN	WORD	E, A, M, L, D	Wert, der geschoben wird
	N	WORD	E, A, M, L, D	Anzahl der Bitpositionen, um die geschoben wird
	OUT	WORD	E, A, M, L, D	Ergebnis der Schiebeoperation
	ENO	BOOL	E, A, M, L, D	Freigabeausgang

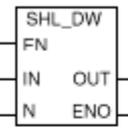
Box 16 Bit links schieben und Parameter

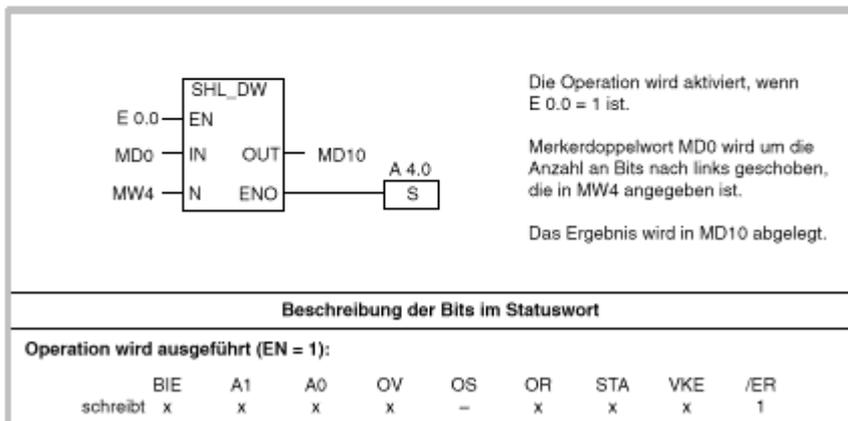


### 32 Bit links schieben

Die Operation *32 Bit links schieben* wird durch den Signalzustand "1" am Freigabeeingang (EN) aktiviert und schiebt die Bits 0 bis 31 von Eingang IN bitweise nach links. Eingang N gibt an, um wie viele Bits geschoben wird. Ist N größer als 32, schreibt der Befehl eine 0 in Ausgang OUT und setzt die Bits A0 und OV auf "0". Die rechts frei werdenden Bitpositionen werden mit Nullen aufgefüllt. Das Ergebnis der Schiebeoperation können Sie an Ausgang OUT abfragen.

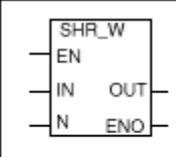
Die ausgelöste Operation setzt bei N ungleich Null das A0- und OV-Bit des Statusworts immer auf "0" zurück. ENO hat den gleichen Signalzustand wie EN.

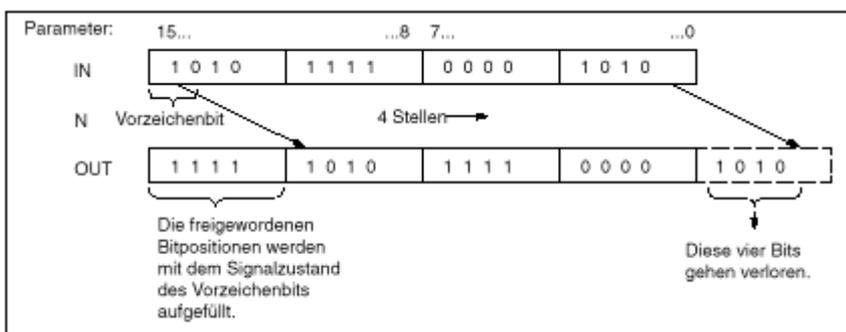
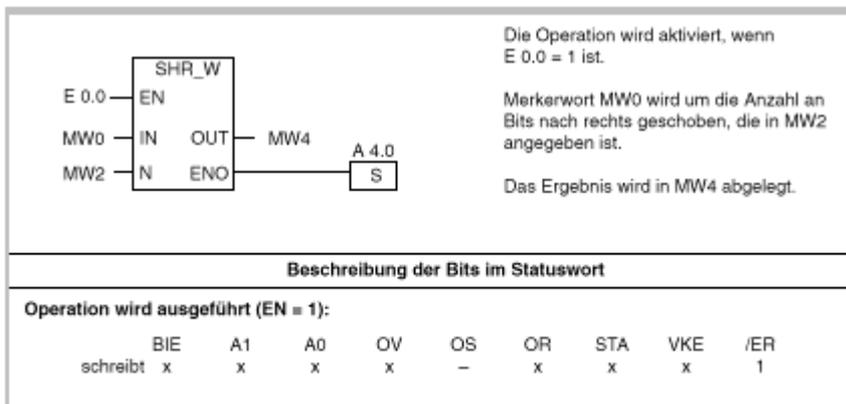
FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, L, D, T, Z	Freigabeeingang
	IN	DWORD	E, A, M, L, D	Wert, der geschoben wird
	N	WORD	E, A, M, L, D	Anzahl der Bitpositionen, um die geschoben wird
	OUT	DWORD	E, A, M, L, D	Ergebnis der Schiebeoperation
	ENO	BOOL	E, A, M, L, D	Freigabeausgang



## 16 Bit rechts schieben

Die Operation *16 Bit rechts schieben* wird durch den Signalzustand "1" am Freigabeeingang (EN) aktiviert und schiebt die Bits 0 bis 15 von Eingang IN bitweise nach rechts. Die Bits 16 bis 31 werden nicht beeinflusst. Eingang N gibt an, um wie viele Bits geschoben wird. Ist N größer als 16, schreibt der Befehl eine 0 in Ausgang OUT und setzt die Bits A0 und OV auf "0". Die links frei werdenden Bitpositionen werden mit Nullen aufgefüllt. Das Ergebnis der Schiebeoperation können Sie an Ausgang OUT abfragen. Die ausgelöste Operation setzt bei N ungleich Null das A0- und OV-Bit des Statusworts immer auf "0" zurück. ENO hat den gleichen Signalzustand wie EN.

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, L, D, T, Z	Freigabeeingang
	IN	WORD	E, A, M, L, D	Wert, der geschoben wird
	N	WORD	E, A, M, L, D	Anzahl der Bitpositionen, um die geschoben wird
	OUT	WORD	E, A, M, L, D	Ergebnis der Schiebeoperation
	ENO	BOOL	E, A, M, L, D	Freigabeausgang

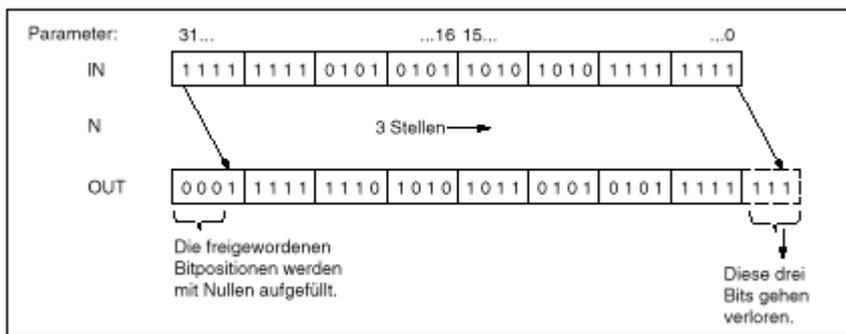


Schieben von vier Bits nach rechts

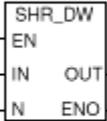
### 32 Bit rechts schieben

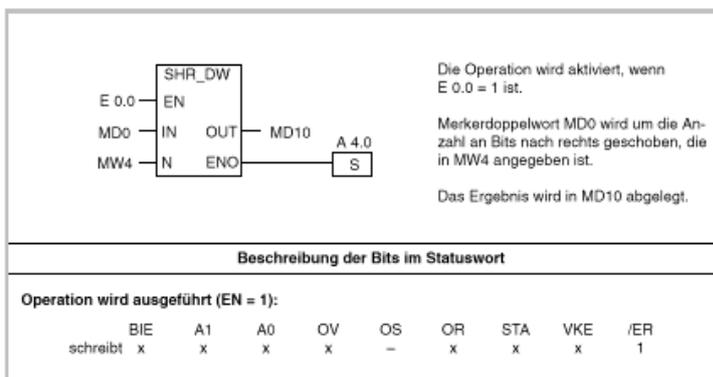
Die Operation *32 Bit rechts schieben* wird durch den Signalzustand "1" am Freigabeeingang (EN) aktiviert und schiebt die Bits 0 bis 31 von Eingang IN bitweise nach rechts. Eingang N gibt an, um wie viele Bits geschoben wird. Ist N größer als 32, schreibt der Befehl eine 0 in Ausgang OUT und setzt die Bits A0 und OV auf "0". Die links frei werdenden Bitpositionen werden mit Nullen aufgefüllt. Das Ergebnis der Schiebeoperation können Sie an Ausgang OUT abfragen.

Die ausgelöste Operation setzt bei N ungleich Null das A0- und OV-Bit des Statusworts auf "0" zurück. ENO hat den gleichen Signalzustand wie EN.



Schieben der Bits von Eingang IN um drei Bitpositionen nach rechts

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, L, D, T, Z	Freigabeeingang
	IN	DWORD	E, A, M, L, D	Wert, der geschoben wird
	N	WORD	E, A, M, L, D	Anzahl der Bitpositionen, um die geschoben wird
	OUT	DWORD	E, A, M, L, D	Ergebnis der Schiebeoperation
	ENO	BOOL	E, A, M, L, D	Freigabeausgang



## 4.6 ROTIEROPERATIONEN

Mit den Rotieroperationen können Sie den gesamten Inhalt von Eingang IN bitweise nach rechts oder links rotieren. Die frei gewordenen Stellen werden mit den Signalzuständen der Bits aufgefüllt, die aus dem Eingang IN geschoben wurden.

Die Zahl, die Sie für den Eingangsparameter N angeben, zeigt an, um wie viele Bits rotiert werden soll. Je nach der gewählten Operation wird über das Bit A1 rotiert und das Bit A0 im Statuswort wird auf "0" zurückgesetzt.

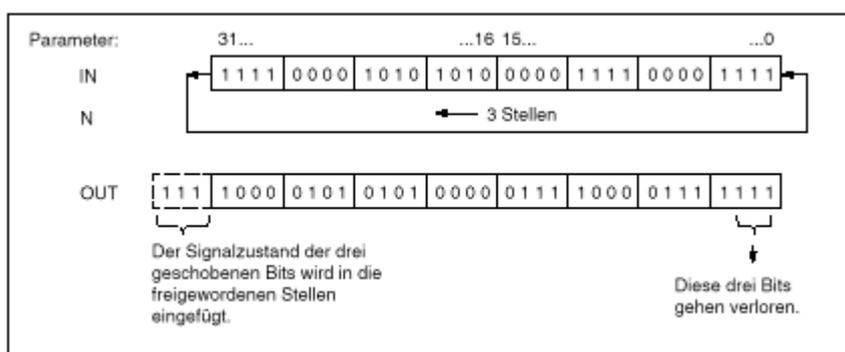
Folgende Rotieroperationen stehen Ihnen zur Verfügung:

- \_ 32 Bit links rotieren
- \_ 32 Bit rechts rotieren

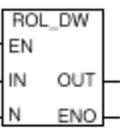
### **32 Bit links rotieren**

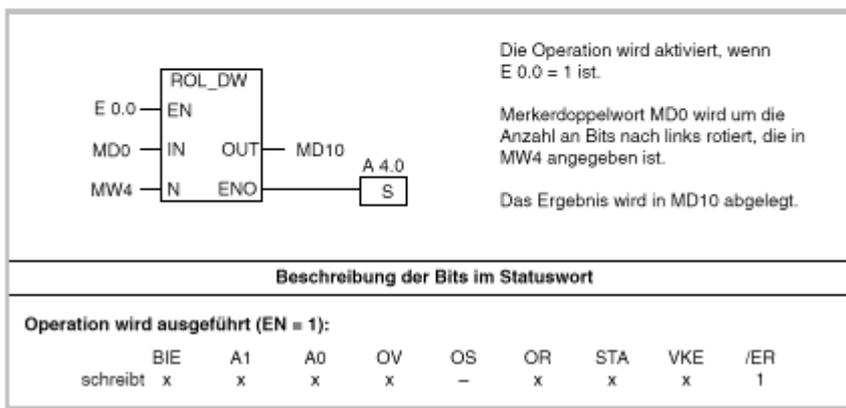
Die Operation *32 Bit links rotieren* wird durch den Signalzustand "1" am Freigabeeingang (EN) aktiviert und rotiert den gesamten Inhalt von Eingang IN bitweise nach links. Eingang N gibt an, um wie viele Bits rotiert wird. Ist N größer als 32, wird das Doppelwort mit  $((N-1) \text{ Modul } 32) + 1$  rotiert. Die rechts frei werdenden Bitpositionen werden mit den Signalzuständen der rotierten Bits aufgefüllt. Das Ergebnis der Rotieroperation können Sie an Ausgang OUT abfragen.

Die ausgelöste Operation setzt bei N ungleich Null das A0- und OV-Bit des Statusworts auf "0" zurück. ENO hat den gleichen Signalzustand wie EN.



## Box 32 Bit links rotieren und Parameter

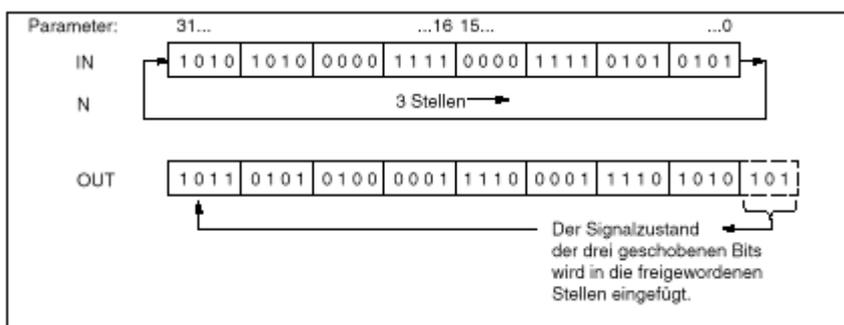
FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, L, D, T, Z	Freigabeingang
	IN	DWORD	E, A, M, L, D	Wert, der rotiert wird
	N	WORD	E, A, M, L, D	Anzahl der Bitpositionen, um die rotiert wird
	OUT	DWORD	E, A, M, L, D	Ergebnis der Rotieroperation
	ENO	BOOL	E, A, M, L, D	Freigabeausgang



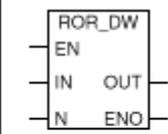
## 32 Bit rechts rotieren

Die Operation *32 Bit rechts rotieren* wird durch den Signalzustand "1" am Freigabeingang (EN) aktiviert und rotiert den gesamten Inhalt von Eingang IN bitweise nach rechts. Eingang N gibt an, um wie viele Bits rotiert wird. Der Wert von N kann zwischen 0 und 31 liegen. Ist N größer als 32, wird das Doppelwort mit  $((N-1) \text{ Modul } 32) + 1$  rotiert. Die links frei werdenden Bitpositionen werden mit den Signalzuständen der rotierten Bits aufgefüllt. Das Ergebnis der Rotieroperation können Sie an Ausgang OUT abfragen.

Die ausgelöste Operation setzt bei N ungleich Null das A0- und OV-Bit des Statusworts auf "0" zurück. ENO hat den gleichen Signalzustand wie EN.



## Box 32 Bit rechts rotieren und Parameter

FUP-Box	Parameter	Datentyp	Speicherbereich	Beschreibung
	EN	BOOL	E, A, M, L, D, T, Z	Freigabeingang
	IN	DWORD	E, A, M, L, D	Wert, der rotiert wird
	N	WORD	E, A, M, L, D	Anzahl der Bitpositionen, um die rotiert wird
	OUT	DWORD	E, A, M, L, D	Ergebnis der Rotieroperation
	ENO	BOOL	E, A, M, L, D	Freigabeausgang

