

FMTool und Funktionsmodul

Technische Dokumentation

Beschreibung der Funktionsblöcke

Funktionsblock-Bibliothek (FBL) Version 2.60

EIB System

Funktionsmodul und FMTool

Hinweise

Die in diesem Unterlagen enthaltenen Angaben, Daten, Werte usw. können ohne vorherige Ankündigung geändert werden. Ebenso sind die Abbildungen unverbindlich.

Ohne ausdrückliche schriftliche Erlaubnis von Gira, Giersiepen GmbH & Co. KG darf kein Teil dieser Unterlagen für irgendwelche Zwecke vervielfältigt oder übertragen werden, unabhängig davon, auf welche Art und Weise und mit welchen Mitteln, elektronisch oder mechanisch, dies geschieht.

©2000 Gira, Giersiepen GmbH & Co. KG
Alle Rechte vorbehalten.

Alle im Handbuch verwendeten Produktbezeichnungen sind eingetragene Warenzeichen der jeweiligen Firmen.

Gira
Giersiepen GmbH & Co. KG
Dahlienstrasse
D-42477 Radevormwald

Telefon 02195 / 602-0
Telefax 02195 / 602-118
Internet [http: // www.gira.de](http://www.gira.de)

Inhaltsverzeichnis

RÜCKFÜHRUNGEN VON SIGNALEN	7
INFORMATION ZU FMTOOL VERSION 1.50 UND ÄLTER.....	8
INFORMATIONEN ZU FBL-VERSIONEN 1.51 UND ÄLTER	9
BESCHREIBUNG DER FUNKTIONSBLOCKE FUNKTIONSBLOCK-BIBLIOTHEK (FBL) VERSION 2.60.....	12
AAS.....	12
AASZ.....	13
ADD	14
ALARM	15
DIV	17
DMFF	19
DT1	21
EAS.....	23
EASM.....	24
EASZ.....	25
EXP	26
FKTG	27
FLANKE	28
GIBD.....	29
GIBT.....	30
GLEICH.....	31
INCDEC (NEU).....	32
INTE	35
INV	37
KLEIN.....	38
KLICK.....	39
LOG	42
MINMAX (NEU).....	43
MUL	44
ODER.....	45
OSZI.....	46
PACBM.....	47
PACBW	48
PACC	49
PACD	50
PACD4	51
PACM.....	52
PACP	53
PACS	54
PACSV	55
PACT	56
PACVS	57
PACW	58
PACWV	59
PI.....	61
PT1	63
PULSG (NEU).....	65
RSFF.....	68
SETD	69
SETT.....	70
SMFF	71
SPEICH.....	72
SUB.....	74
SUHR.....	75
TEILC.....	76

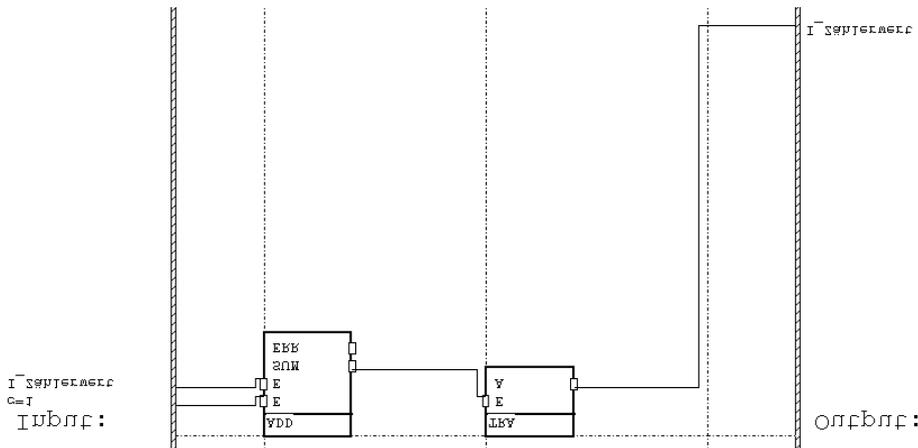
TEILD	77
TEILD4	78
TEILM	79
TEILMB	80
TEILP	81
TEILS	82
TEILT	83
TEILVW	84
TEILW	85
TEILWB	86
TFF	87
TRA	88
UND	89
VERZ	90
WECHS	90
XOR	92
ZÄHLER	93
ZUER	94
ZUINIT	96
ZUSET	97
ZWI (NEU)	98
ZWIZ (NEU)	99

Rückführungen von Signalen

Bei der Funktionsblockbibliothek (FBL) Version 1.53 gibt es Blöcke, die bei der Berechnung den Ausgang als Zwischenspeicher verwenden. Dies kann zu unerwarteten Ergebnissen kommen, falls die Ausgangssignale auf die Eingänge zurückgeführt werden.

Korrektur:

Zwischen den entsprechenden Funktionsblock und das Ausgangssignal ist ein TRA-Block zu schalten.



Liste der betroffenen Blöcke

Die folgende Liste enthält alle Funktionsblöcke, die Probleme bei rückgeführten Signalen machen. Es ist jeweils angegeben, welches die kritischen Ausgänge sind und welche Datentypen betroffen sind.

Funktionsblock	Datentyp	Ausgang
DIV	Value	QUOT
DT1		A
DMFF		A
INTE	alle	A
MUL	alle	A
PI		A
PT1		A
RSFF		A
SMFF		A
TFF		A

Information zu FMTool Version 1.50 und älter

Initialisierung der Eingangs-Kontrollsignale

Die FMTool-Versionen 1.50 und älter enthielten im Dialog 'Signal erzeugen' (theoretisch) die Möglichkeit, Eingangs-Kontrollsignale zu initialisieren.

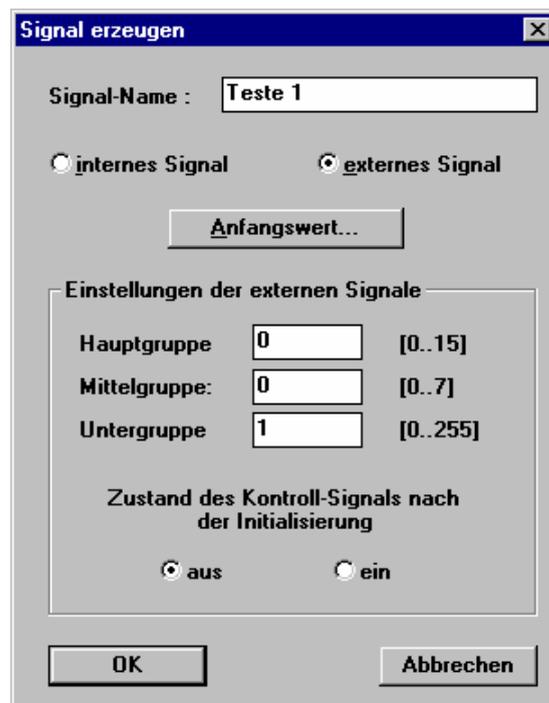
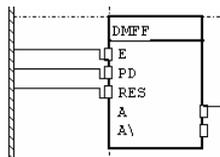
Die Eingangs-Kontrollsignale dürfen nicht initialisiert werden, d.h., die Einstellung 'Zustand des Kontroll-Signals nach der Initialisierung' muss immer auf 'aus' sein.

Ist die Einstellung auf 'ein' gesetzt, empfängt das Funktionsmodul dieses Signal nicht. Es generiert eine Fehlermeldung (ID 12; Zu viele Telegramme empfangen) falls ein Telegramm mit dieser Gruppenadresse auf den Bus gesendet wird.

Dies gilt für alle Versionen des Funktionsmodul-Betriebssystem (OS 2.20 und alle älteren Versionen (tiefere Versionsnummern)).

Beispiel:

Input:
C_Tastendruck
c=200
c=logical low



Informationen zu FBL-Versionen 1.51 und älter

Version 1.05 und 1.51— Funktionsblöcke EASM und TEILMB

Die Blöcke EASM und TEILMB (ab der FBL-Version 1.05 neu dazugekommen) sind in den genannten Versionen mit Fehlern behaftet. Diese sind ab der aktuellen Version 1.53 behoben.

Der Block EASM kann gemäss Beschreibung und Datentypendefinition im Dialog 'Funktionsblock auswählen' alle möglichen Datentypen annehmen. Der Ausgang wird jedoch durch das System selbst immer auf Datentyp 'Word' gesetzt, unabhängig vom gewählten Typ.

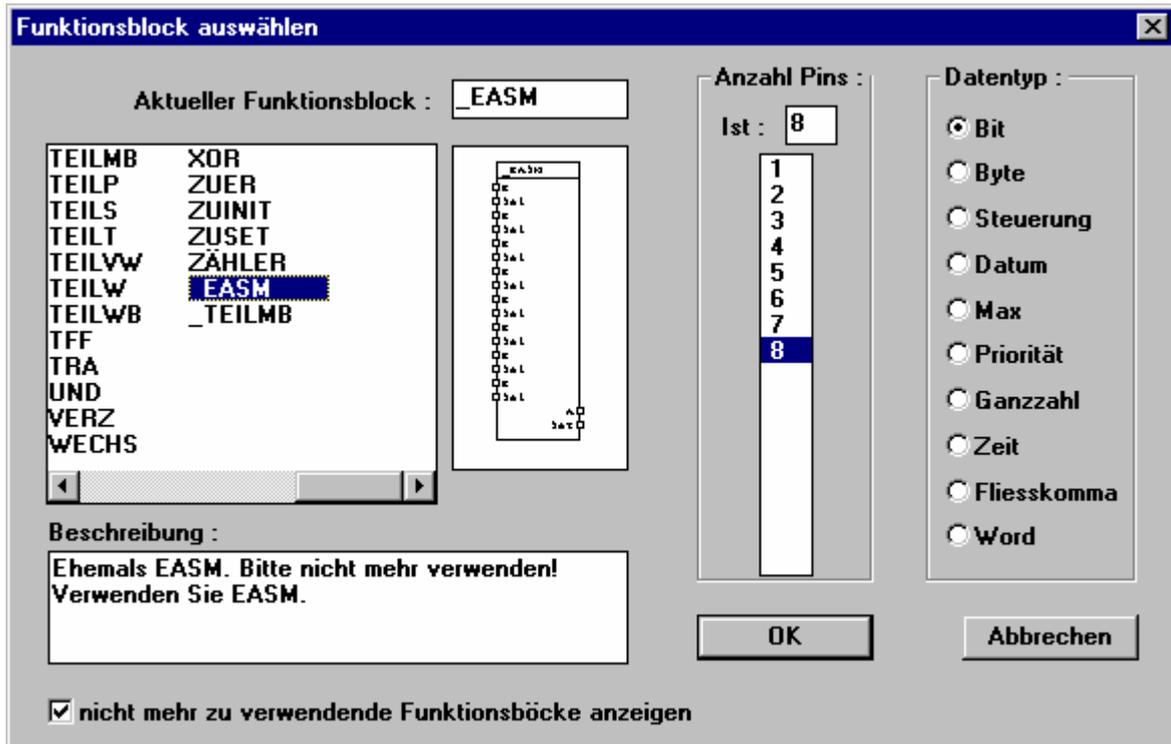
Der Block TEILMB wandelt Daten vom Typ 'Max' in 'Byte'. Der Ausgang ist jedoch fälschlicherweise vom 'Word'.

Verschiedene Anwender haben diese Fehler bei er Anwendung in Projekten dadurch 'korrigiert', dass sie den Blöcken entsprechende Funktionsblöcke zum Wandeln von Datentypen nachgeschaltet haben, z.B. bei TEILMB den Block TEILWB.

Damit diese Projekte mit der vorgängig beschriebenen Korrekturmassnahme auch nach einem Wechsel (Upgrade) auf die aktuelle FBL Version 1.53 weiterhin funktionsfähig bleiben, sind die mit den Fehlern behafteten Blöcken auch in der Version 1.53 enthalten, zusätzlich zu den korrigierten.

Die alten, bei neuen Projekten oder nach allfälligen Projektanpassungen möglichst nicht mehr zu verwendenden Blöcke mit den Fehlern sind unter den Namen _EASM bzw. _TEILMB enthalten (siehe nachfolgendes Bild).

Bei Projekten, die mit den Versionen 1.05 oder 1.51 erstellt wurden und bei denen die Blöcke EASM bzw. TEILMB verwendet wurden, erscheinen diese Blöcke nach dem Upgrade des Projektes auf FBL 1.53 automatisch als _EASM bzw. _TEIL MB. Die Funktion der Blöcke bleibt jedoch unverändert (mit den bekannten Fehlern).



Version 1.04 und älter — Benutzung der Ausgangs-Kontrollsignale

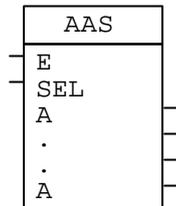
Wird die Funktionsblockbibliothek (FBL) mit der Version 1.04 oder älter verwendet, dürfen nicht alle Funktionsblockausgänge an ein Ausgangs-Kontrollsignal angeschlossen werden. Die folgende Liste enthält alle Funktionsblöcke mit den betroffenen Ausgängen, bei welchen Funktionsstörungen auftreten können. Bei allen nicht erwähnten Ausgängen treten keine Probleme auf.

Block	Ausgang	Was geschieht wenn der Ausgang an ein Kontrollsignal angeschlossen wird
AAS	A	Ausgänge die nicht durch den SEL-Eingang angewählt sind, werden durch das Anschliessen des Kontrollsignals automatisch immer auf 0 gesetzt. Der ausgewählte Ausgang hat immer den Wert des Einganges.
AASZ	A	Der Ausgang der nicht durch den SEL-Eingang angewählt ist, wird durch das Anschliessen des Kontrollsignals automatisch immer auf 0 gesetzt. Der ausgewählte Ausgang hat immer den Wert des Einganges.
ALARM	A	Wenn A mit einem Kontrollsignal verbunden ist, funktioniert das zyklische wiederholen des Pulses am Ausgang nicht (Eingang ZYKL auf 1).
DMFF	A	Wenn A mit einem Kontrollsignal verbunden ist, ist der Ausgang nur einen Taskzyklus lang auf 1. Der Ausgang A\ ist dann auch nur einen Taskzyklus lang auf 0. Ist der Eingang PD auf 1 gesetzt, treten keine Funktionsstörungen auf.
OSZI	A	Wenn A mit einem Kontrollsignal verbunden ist, ist der Ausgang nur einen Taskzyklus lang auf 1. Die Zykluszeit ändert sich aber nicht. Der Ausgang A\ ist dann auch nur einen Taskzyklus lang auf 0. Ist der Eingang PW auf 1 gesetzt, treten keine Funktionsstörungen auf.
RSFF	A	Wenn A mit einem Kontrollsignal verbunden ist, ist der Ausgang nur einen Taskzyklus lang auf 1. Der Ausgang A\ ist dann auch nur einen Taskzyklus lang auf 0.
SMFF	A	Wenn A mit einem Kontrollsignal verbunden ist, ist der Ausgang nur einen Taskzyklus lang auf 1. Der Ausgang A\ ist dann auch nur einen Taskzyklus lang auf 0. Ist der Eingang PD auf 1 gesetzt, treten keine Funktionsstörungen auf.
TFF	A	Wenn A mit einem Kontrollsignal verbunden ist, ist der Ausgang nur einen Taskzyklus lang auf 1. Der Ausgang A\ ist dann auch nur einen Taskzyklus lang auf 0.

Beschreibung der Funktionsblöcke Funktionsblock-Bibliothek (FBL) Version 2.60

AAS

Ausgangsauswahlschalter

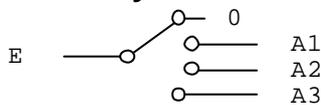


Funktion

Der AAS-Block schaltet den Eingang auf den selektierten Ausgang. Der ausgewählte Ausgang erhält also den Wert des Einganges. Die anderen Ausgänge verändern ihren Wert nicht. Die Anzahl der Ausgänge ist variabel zwischen 1 und 16. Der erste Ausgang ist die Nummer 1, der zweite die Nummer 2 usw. Hat SEL keinen gültigen Wert, so verändert sich kein Ausgang.

Nach einem Reset am Funktionsmodul haben alle nicht selektierten Ausgänge den Wert 0.

Schaltsymbol



Datentypen

Alle Datentypen können verwendet werden.

Ein-/Ausgänge

E Eingangssignal
SEL Selektion, wählt den Ausgang (Word)
A Ausgangssignale

Tips

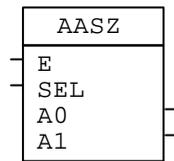
- Die Ausgänge dieses Blockes können als Signalwertspeicher verwendet werden. Wird der Ausgang angewählt, kann sein Wert verändert werden, sonst behält er seinen Wert.

Siehe auch Block:

AASZ, EAS, EASZ, EASM

AASZ

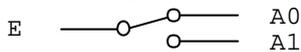
Ausgangsauswahlschalter zweifach



Funktion

Der AASZ-Block schaltet den Eingang auf einen der zwei Ausgänge. Der Selektions-Eingang ist vom Datentyp Bit. Ist SEL 0, wird der Ausgang A0 ausgewählt, sonst A1. Der nicht ausgewählte Ausgang ändert seinen Wert nicht. Nach einem Reset am Funktionsmodul hat der nicht selektierte Ausgang den Wert 0.

Schaltsymbol



Datentypen

Alle Datentypen können verwendet werden.

Ein-/Ausgänge

E Eingangssignal
SEL Selektion, wählt den Ausgang (Bit)
A0 Ausgang 0
A1 Ausgang 1

Tips

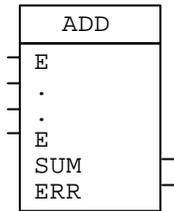
- Die Ausgänge dieses Blockes können als Signalwertspeicher verwendet werden. Wird der Ausgang angewählt kann sein Wert verändert werden, sonst behält er seinen Wert.
- Ein Signal kann unterbrochen werden, damit es seinen Wert nicht mehr ändert. So kann zum Beispiel verhindert werden, dass ein Telegramm gesendet wird.

Siehe auch Block:

AAS, EASZ, EAS, EASM

ADD

Addierer



Funktion

Der ADD-Block addiert mehrere Eingangswerte zusammen. Die Anzahl der Eingänge ist variabel zwischen 2 und 16. Geht die Summe oder ein Zwischenresultat über den Bereich des Datentyps hinaus, ist das Überlaufbit 1.

Datentypen

Word-, Ganzzahl- (Sint-) oder Fließkommawerte (Value-Werte) können addiert werden.

Ein-/Ausgänge

E Summande
SUM Summe
ERR Überlauf (Bit)

Formeln

Blockfunktion:

$$\text{SUM} = E1 + E2 + \dots + En$$

Tips

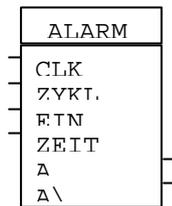
- Sollen grössere ganzzahlige Zahlen als 65535 addiert werden (z.B. EIS11), kann der ADD-Block mit Datentyp Word verwendet werden. Bei diesem Block entspricht das Überlaufbit dem Carry-Flag. Es kann also als Bit 16 zu dem höherwertigen Word dazuaddiert werden. Der A-Ausgang ist dann das niederwertige Word.

Siehe auch Block:

SUB, MUL, DIV

ALARM

Alarm



Funktion

Der ALARM-Block ist wie ein Wecker. Bei einer steigenden Flanke am CLK-Eingang wird ein Zähler auf den Wert gesetzt, welcher am ZEIT-Eingang vorhanden ist. Gleichzeitig beginnt der Zähler rückwärts zu zählen. Hat der Zähler Null erreicht, wird der Ausgang während einem Taskzyklus auf 1 gesetzt (kurzer Puls).

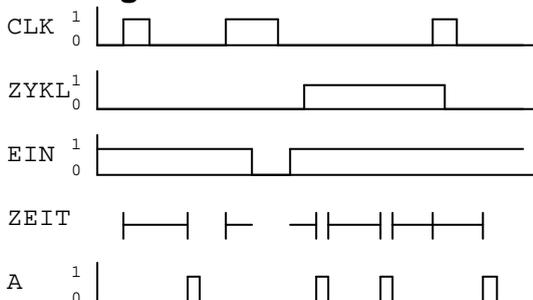
Ist der ZYKL-Eingang nach dem Puls 1, wird der Zähler erneut gestartet. Es entsteht so ein zyklischer Ausgang, wobei die Zykluszeit jedesmal vom ZEIT-Eingang abhängig ist.

Der Zählerstand wird eingefroren, wenn der EIN-Eingang 0 ist.

Wird eine zweite steigende Flanke auf den CLK-Eingang gegeben, bevor der Zähler Null wird, wird der Zähler wieder auf den Wert am ZEIT-Eingang gesetzt (retriggerbar).

Ist der CLK-Eingang auf den Wert 1 initialisiert, wird dies nach einem Reset des Funktionsmoduls als steigende Flanke interpretiert und der ALARM-Block wird gestartet.

Zeitdiagramm



Ein-/Ausgänge

CLK flankengesteuerter Eingang (Bit)

ZYKL zyklisches Ausgangssignal (Bit)

EIN Ein-/Ausschalten (Bit)

ZEIT Zeit in Taskzyklen (Word)

A nichtinvertierter Ausgang (Bit)

A\ invertierter Ausgang (Bit)

Formeln

Der Eingangswert für ZEIT kann folgendermassen berechnet werden:

$$\text{ZEIT} = \frac{\text{Alarmzeit [s]}}{\text{Taskzykluszeit [s]}}$$

Tips

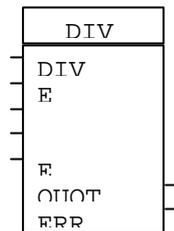
- Wird der A-Ausgang auf ein Ausgangskontrollsignal gegeben, kann der ALARM-Block dazu verwendet werden, Telegramme zyklisch zu senden.
- Er kann auch zum Verzögern von Pulsen verwendet werden.

Siehe auch Block:

OSZI

DIV

Dividierer



Funktion

Der DIV-Block dividiert eine Zahl (Dividend) durch eine oder mehrere Zahlen (Divisoren). Die Anzahl der Divisoren ist variabel zwischen 1 und 16.

Geht das Resultat oder ein Zwischenresultat über den Bereich des Datentypen hinaus, oder wird durch Null dividiert, ist der Ausgang undefiniert und das Überlaufbit ist 1.

Wird mit den Datentypen Word oder Sint dividiert, ist das Resultat immer auf die nächste ganze Zahl abgerundet.

Datentypen

Word-, Ganzzahl- (Sint-) und Fließkomma-Werte (Value-Werte) können voneinander dividiert werden.

Ein-/Ausgänge

DIV Dividend
E Divisoren
QUOT Quotient (Resultat)
ERR Divisionsfehler (Bit)

Formeln

Blockfunktion:

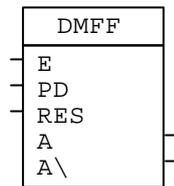
$$\text{QUOT} = (((\text{DIV} / \text{E1}) / \text{E2}) / \dots) / \text{En}$$

Siehe auch Block:

MUL, ADD, SUB

DMFF

Dynamisches, monostabiles Flip-Flop, retriggerbar



Funktion

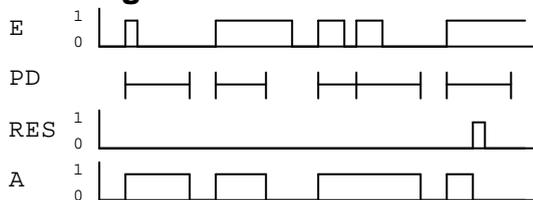
Der DMFF-Block ist ein monostabiles Flip-Flop. Bei einer steigenden Flanke am Eingang wird am Ausgang ein Puls generiert. Die Dauer des Pulses wird am PD-Eingang eingestellt. Gibt es während des Pulses erneut eine steigende Flanke am Eingang, wird die Pulsdauer neu gesetzt und der Puls wird verlängert (retriggerbar).

Der Wert am PD-Eingang hat nur während der steigenden Flanke am E-Eingang einen Einfluss auf die Pulsdauer. Ist PD Null bleibt auch der Ausgang Null.

Der RES-Eingang setzt das Ausgangssignal jederzeit auf Null.

Ist der E-Eingang auf den Wert 1 initialisiert, wird dies nach einem Reset des Funktionsmoduls als steigende Flanke interpretiert und der Puls am Ausgang wird gestartet.

Zeitdiagramm



Ein-/Ausgänge

- E flankengesteuerter Eingang (Bit)
- PD Pulsdauer in Taskzyklen (Word)
- RES Reset
- A nichtinvertiertes Ausgangssignal (Bit)
- A\ invertiertes Ausgangssignal (Bit)

Formeln

Der Eingangswert für PD kann folgendermassen berechnet werden:

$$PD = \frac{\text{Pulsdauer [s]}}{\text{Taskzykluszeit [s]}}$$

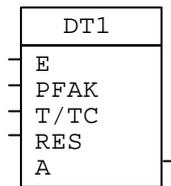
Tips

- Der DMFF-Block kann z. B. das Licht in einem Treppenhaus steuern (Treppenhausautomat).

Siehe auch Block:
SMFF

DT1

DT1-Glied (Hochpass ersten Grades)



Funktion

Dies ist ein DT1-Glied für Regelaufgaben. Es basiert auf dem Näherungsalgorithmus von Gauss. Reset setzt den Ausgang auf Null.

Damit eine vernünftige Rechengenauigkeit erreicht wird, soll die Taskzykluszeit mindestens 10 mal kleiner sein als die Periodendauer der kritischen Grenzfrequenz des Regelsystems.

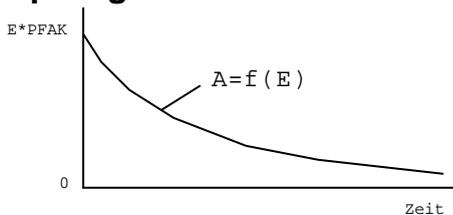
Der Ausgangswert ist auf den Wertebereich des Datentypen Value begrenzt. Ist T/TC negativ, oder ist PFAK negativ, so verändert sich der Ausgangswert nicht.

Die Auflösung des Ausgangssignals ist beschränkt. Ist die theoretische Änderung des Ausgangssignals pro Taskzyklus kleiner als 0.000'313, ändert sich der Ausgangswert nicht. Der Wert am T/TC-Eingang darf also nicht zu hoch gewählt werden.

Die Werte am PFAK- und am T/TC-Eingang dürfen zu jedem beliebigen Zeitpunkt geändert werden.

Ist der E-Eingang auf einen Wert ungleich 0 initialisiert, wird dies nach einem Reset des Funktionsmoduls als Signaländerung interpretiert und am Ausgang entsteht der entsprechende Sprung. Andernfalls bleibt der Ausgang 0.

Sprungantwort



Ein-/Ausgänge

- E Eingangssignal (Fließkomma; Value)
- PFAK Proportionalfaktor (Fließkomma; Value)
- T/TC Zeitkonstante (Fließkomma; Value)
- RES Reset (Bit)
- A Ausgangssignal (Fließkomma; Value)

Formeln

Blockfunktion:

$$A = ((PFAK * (E - Ealt) + Aalt) * (T/TC)) / (1 + (T/TC))$$

alt bedeutet: Wert aus dem letzten Taskzyklus.

Der Eingangswert für T/TC kann folgendermassen berechnet werden:

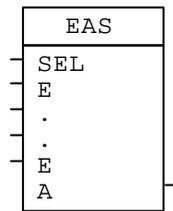
$$T / TC = \frac{\text{Zeitkonstante [s]}}{\text{Taskzykluszeit [s]}}$$

Siehe auch Block:

INTE, PI, PT1

EAS

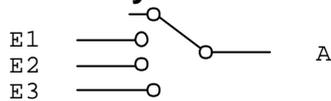
Eingangsauswahlschalter



Funktion

Der EAS-Block schaltet den selektierten Eingang auf den Ausgang. Der Ausgang erhält also den Wert des ausgewählten Eingangs. Die Anzahl der Eingänge ist variabel zwischen 1 und 16. Der erste Eingang ist die Nummer 1, der zweite die Nummer 2 usw. Hat SEL keinen gültigen Wert, so bleibt der Ausgang unverändert.

Schaltensymbol



Datentypen

Alle Datentypen können verwendet werden.

Ein-/Ausgänge

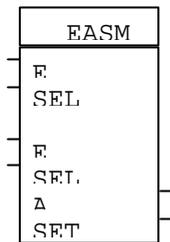
- SEL Selektion, wählt den Eingang (Word)
- E Eingangssignale
- A Ausgangssignal

Siehe auch Block:

EASZ, EASM, AAS, AASZ

EASM

Eingangsauswahlschalter mehrfach



Funktion

Der EASM-Block schaltet den selektierten Eingang auf den Ausgang. Dieser erhält somit den Wert des ausgewählten Eingangs, d.h., des Eingangs, dessen SEL-Bit gesetzt ist. Das Ausgangssignal SET (Bit) wird gesetzt, wenn (mindestens) eines der Selektionsbits aktiv ist.

Ist mehr als ein Selektionsbit aktiv, wird der Wert des untersten selektierten Eingangs auf den Ausgang geschaltet.

Die Anzahl der Eingänge ist variabel zwischen 1 und 8.

Datentypen

Alle Datentypen können verwendet werden.

Ein-/Ausgänge

E Eingangssignale

SEL Selektion, wählt den Eingang (Bit)

A Ausgangssignal

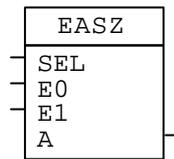
SET Ausgang (Bit) = Oder-Verknüpfung aller SEL (Bit)

Siehe auch Block:

EAS, EASZ, AAS, AASZ

EASZ

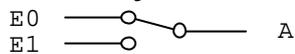
Eingangsauswahlschalter zweifach



Funktion

Der EASZ-Block schaltet den selektierten Eingang auf den Ausgang. Der Ausgang erhält also den Wert des ausgewählten Einganges. Ist SEL 0, wird der Eingang E0 ausgewählt, sonst E1.

Schaltsymbol



Datentypen

Alle Datentypen können verwendet werden.

Ein-/Ausgänge

SEL Selektion, wählt den Eingang (Bit)
E0 Eingangssignal 0
E1 Eingangssignal 1
A Ausgangssignal

Tips

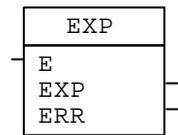
- Der EASZ-Block kann zur Realisierung von IF THEN ELSE verwendet werden.
Beispiel: IF (SEL == 1) THEN A = E1 ELSE A = E0.

Siehe auch Block:

EAS, EASM, AAS, AASZ

EXP

Exponentialfunktion



Funktion

Des EXP-Block berechnet die Exponentialfunktion mit der Basis e (e = 2.71828). Überschreitet das Resultat den Wertebereich des Datentypen Value, ist der Ausgangswert unbestimmt und das Überlaufbit wird gesetzt.

Ein-/Ausgänge

E Eingangssignal (Fließkomma; Value)
EXP Ausgangssignal (Fließkomma; Value)
ERR Überlauf (Bit)

Formeln

Blockfunktion:

$$EXP = e^E$$

Rechengenauigkeit

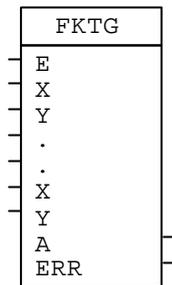
Die Auflösung des Datentypen Value ist beschränkt auf 0.000'313. Darum kann der Ausgangswert des EXP-Blockes bei kleinen Zahlen am Eingang, hohe relative Fehler aufweisen.

Siehe auch Block:

LOG

FKTG

Funktionsgeber



Funktion

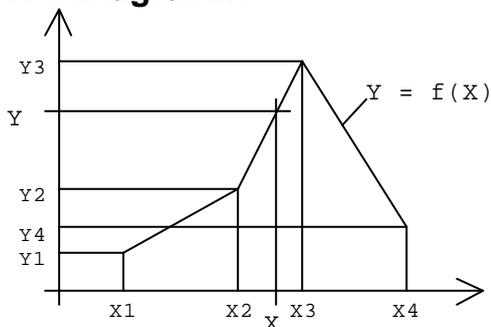
Der FKTG-Block bildet beliebige Funktionen nach. Die Funktion $Y=f(X)$ wird anhand von 2 bis 8 X-Y-Koordinaten definiert. Die Bereiche zwischen diesen Punkten werden durch eine Gerade verbunden (linear approximiert).

Ist der Eingangswert ausserhalb der definierten Funktion, ist das Ausgangssignal unbestimmt und das Fehlerbit wird gesetzt.

Die X-Koordinaten müssen in aufsteigender Reihenfolge sortiert sein, sonst kann ein falsches Ausgangssignal resultieren. D.h. die Bedingung $X_1 \leq X_2 \leq \dots \leq X_n$ muss erfüllt sein.

Haben zwei aufeinanderfolgende Koordinaten den gleichen X-Wert, so wird für diesen X-Wert der Y-Wert der ersten Koordinate auf den Ausgang gegeben. (z.B. ist $X_1=X_2$, so wird für diesen X-Wert Y_1 auf den Ausgang gegeben.)

X-Y-Diagramm

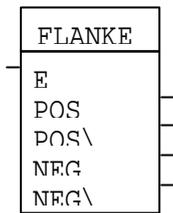


Ein-/Ausgänge

- E Eingangssignal (Fließkomma; Value)
- X X-Koordinate (Fließkomma; Value)
- Y Y-Koordinate (Fließkomma; Value)
- A Ausgangssignal (Fließkomma; Value)
- ERR Fehler (Bit)

FLANKE

Flankendetektor



Funktion

Der FLANKE-Block erkennt, ob es sich bei einem binären Eingangssignal um eine steigende oder um eine fallende Flanke handelt, d.h., ob der Wechsel am Eingang von 0 auf 1 oder umgekehrt erfolgt. Der entsprechende Ausgang wird danach für genau einen Taskzyklus gesetzt.

Ein-/Ausgänge

- E Eingangssignal
- POS nichtinvertierter Ausgang für steigende Flanke (Bit)
- POS\ invertierter Ausgang für steigende Flanke (Bit)
- NEG nichtinvertierter Ausgang für fallende Flanke (Bit)
- NEG\ invertierter Ausgang für fallende Flanke (Bit)

EIB System

Funktionsmodul und FMTool

GIBD

Datumgeber



Funktion

Der GIBD-Block gibt das Systemdatum des Funktionsmoduls. Das Datum enthält: Tag, Monat und Jahr. Das Jahr hat die Werte 00 bis 99.

Ein-/Ausgänge

DAT Datum (Date)

Einschränkungen

- Werden mehrere Tasks verwendet, soll dieser Block nur im schnellsten Task verwendet werden. Das Datum kann aber mit einem internen Signal auch in den anderen Tasks verwendet werden.

Siehe auch Block:

GIBT, SETT

GIBT

Zeitgeber



Funktion

Der GIBT-Block gibt die Systemzeit des Funktionsmoduls. Die Zeit enthält den Wochentag (Montag, Dienstag, usw.), die Stunden, Minuten und Sekunden.

Ein-/Ausgänge

ZEIT Zeit (Time)

Einschränkungen

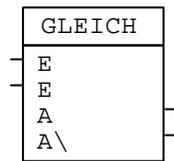
- Werden mehrere Tasks verwendet, soll dieser Block nur im schnellsten Task verwendet werden. Die Systemzeit kann aber mit einem internen Signal auch in den anderen Tasks verwendet werden.

Siehe auch Block:

SETT

GLEICH

Vergleicher



Funktion

Der GLEICH-Block vergleicht die beiden Eingänge. Sind sie gleich, ist der A-Ausgang 1.

Datentypen

Alle Datentypen

Ein-/Ausgänge

- E zu vergleichende Eingänge
- A nichtinvertierter Ausgang (Bit)
- A\ invertierter Ausgang (Bit)

Tips

- Der GLEICH-Block kann auch als XOR-Funktion verwendet werden, wenn der Datentyp Bit gewählt wird (siehe Beispiel, Wahrheitstabelle).

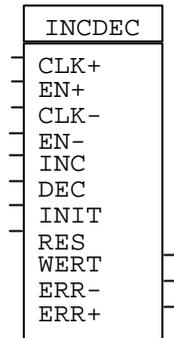
Wahrheitstabelle

E	E	A	A\
0	0	1	0
0	1	0	1
1	0	0	1
1	1	1	0

Siehe auch Block:

KLEIN

INCDEC (Neu)



Funktion

Der INCDEC-Block erhöht (addiert) den aktuellen Speicherwert um den Wert (INC), wenn der Eingang (CLK+) eine steigende Flanke (0 auf 1) erkennt. Der INCDEC-Block verkleinert (subtrahiert) den aktuellen Speicherwert um den Wert (DEC), wenn der Eingang (CLK-) eine steigende Flanke (0 auf 1) erkennt. Der Speicherwert setzt sich aus dem Ausgangswert (WERT) und den dazu addierten oder subtrahierten Werten zusammen. Der Speicherwert verändert sich mit jeder Inkrementierung oder Dekrementierung.

Die Eingänge (EN+ oder EN-) dienen zur Freigabe der entsprechenden Clockeingänge.

Ist der Minimalwert bei der Addition oder Subtraktion unterschritten wird der Ausgang (ERR-) gesetzt und der Minimalwert an den Ausgang (WERT) kopiert. Ist der Maximalwert bei der Addition oder Subtraktion überschritten wird der Ausgang (ERR+) gesetzt und der Maximalwert an den Ausgang (WERT) kopiert. Wird der Reset (RES) gesetzt, so kopiert der Block den Startwert (INIT) an den Ausgang (WERT). In diesem Fall werden keine Rechenoperationen ausgeführt. Nach einem Reset des Funktionsmodules wird der festgelegte Anfangswert des Signales, welches mit dem Ausgang (WERT) verbunden ist, als Startwert zum inkrementieren oder dekrementieren verwendet.

Hinweis

Die Addition und die Subtraktion können im gleichen Zyklus berechnet werden, die Addition wird immer zuerst ausgeführt.

Die Minimal- und Maximalwerte werden durch den selektierten Datentyp bestimmt, z.B. (Byte 0-255).

Nur wenn an den Eingängen (EN+ oder EN-) eine logische "1" anliegt, wird die steigende Flanke an den entsprechenden Clockeingängen übernommen und verarbeitet.

Datentypen

Byte, Sint, Value, Word

EIB System

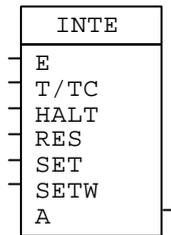
Funktionsmodul und FMTool

Ein-/Ausgänge

CLK+	Clock increment (Bit)
EN+	Enable increment (Bit)
CLK-	Clock decrement (Bit)
EN-	Enable decrement (Bit)
INC	Erhöhungswert
DEC	Verkleinerungswert
INIT	Initial-oder Startwert
RES	Reset (Bit)
WERT	Wertausgang
ERR-	unterer Grenzwert (Bit)
ERR+	oberer Grenzwert (Bit)

INTE

Integrator



Funktion

Der INTE-Block ist ein Integrator-Glied mit einer Integrationskonstante T/TC. Die Integration basiert auf dem Näherungsalgorithmus von Gauss.

Mit dem HALT-Eingang kann die Integration angehalten werden, d.h. der Ausgangswert verändert sich nicht mehr. RES setzt den Ausgang auf Null.

Wenn SET 1 ist, wird der Ausgang auf den Wert am SETW-Eingang gesetzt.

RES hat die höhere Priorität als SET und SET hat die höhere Priorität als HALT.

Erreicht der Ausgang den Wertebereich des Datentypen, so wird dieser Maximal- bzw. Minimalwert beibehalten bis die Integrationsrichtung wechselt.

Ist T/TC negativ oder Null, wird der Ausgang nicht verändert.

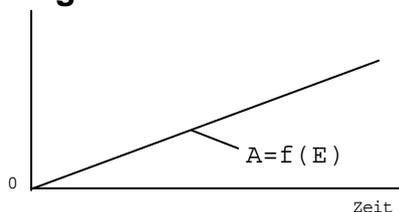
Damit eine vernünftige Rechengenauigkeit erreicht wird, soll die Taskzykluszeit mindestens 10 mal kleiner sein als die Periodendauer der kritischen Grenzfrequenz des Regelsystems.

Der Wert am T/TC-Eingang darf zu jedem beliebigen Zeitpunkt geändert werden.

Die Auflösung des Ausgangssignals ist beschränkt. Ist die theoretische Änderung des Ausgangssignals pro Taskzyklus kleiner als der kleinste darstellbare Wert, ändert sich der Ausgangswert nicht. Der Wert am T/TC-Eingang darf also nicht zu hoch gewählt werden. Die Auflösung beim Datentypen Value ist $0.000'313$, bei Word und Sint 1.

Nach einem Reset am Funktionsmodul startet der Integrationswert bei 0.

Sprungantwort



Datentypen

Word, Ganzzahl (Sint), Fließkomma (Value)

Ein-/Ausgänge

E	Eingangssignal
T/TC	Zeitkonstante
HALT	Integration anhalten (Bit)
RES	Reset (Bit)
SET	Ausgangssignal setzen (Bit)
SETW	Wert der gesetzt wird
A	Ausgangssignal

Formeln

Blockfunktion:

$$A = A_{\text{alt}} + E/(T/TC)$$

A_{alt} bedeutet: Ausgangswert aus dem letzten Taskzyklus.

Der Eingangswert für T/TC kann folgendermassen berechnet werden:

$$T / TC = \frac{\text{Zeitkonstante [s]}}{\text{Taskzykluszeit [s]}}$$

Tips

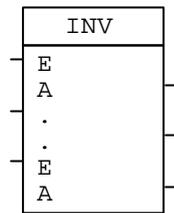
- Sind die Integrationsschritte pro Taskzyklus zu klein (weil der Wert am T/TC-Eingang zu gross ist), kann die Taskzykluszeit künstlich vergrössert werden indem der HALT-Eingang nur alle paar Taskzyklen einmal auf 0 gesetzt und sonst auf 1 belassen wird. Dies kann mit dem OSZI-Block realisiert werden, wobei die 'künstliche' Taskzykluszeit der Zykluszeit des Oszillators entspricht.

Siehe auch Block:

DT1, PI, PT1, ZÄHLER

INV

Logischer Inverter



Funktion

Der INV-Block invertiert ein binäres Signal. Die Anzahl der Ein- bzw. Ausgänge ist variabel zwischen 1 und 8.

Wahrheitstabelle

E	A
0	1
1	0

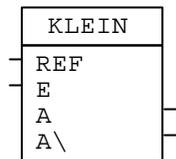
Ein-/Ausgänge

E Eingangssignal (Bit)

A Ausgangssignal (Bit)

KLEIN

Kleiner als (vergleichen)



Funktion

Der KLEIN-Block vergleicht den Eingang E mit einer Referenz. Ist E kleiner als REF, ist der nichtinvertierte Ausgang 1.

Verwendung

Die untenstehende Tabelle zeigt, wie die Signale für die verschiedenen Vergleiche an den KLEIN-Block angeschlossen werden müssen, zur Verwendung als KLEIN- bzw. als 'Grösser'-Block.

Vergleich	Signal A	Signal B	Signal C
$C = A < B$	E	REF	A
$C = A \leq B$	REF	E	A\
$C = A > B$	REF	E	A
$C = A \geq B$	E	REF	A\

Datentypen

Byte, Datum (Date), Ganzzahl (Sint), Zeit (Time), Fließkomma (Value), Word

Ein-/Ausgänge

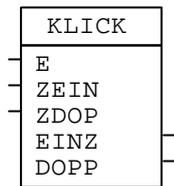
REF Referenzwert
 E zu vergleichendes Signal
 A nichtinvertierter Ausgang (Bit)
 A\ invertierter Ausgang (Bit)

Siehe auch Block:

GLEICH

KLICK

Erkennen von Klick und Doppelklick



Funktion

Der KLICK-Block erkennt, ob auf einem binären Eingangssignal ein einzelner Puls, oder zwei aufeinanderfolgende Pulse auftreten (einzelner Klick, oder Doppelklick).

Ein Puls darf nur einen Taskzyklus lang dauern. Meistens wird der E-Eingang mit einem Eingangskontrollsignal verbunden, auf welches eine Taste sendet.

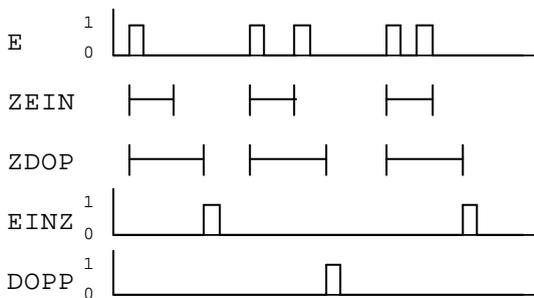
Wenn während der ganzen Zeit ZDOP nur ein Puls auftritt, wird der EINZ-Ausgang einen Taskzyklus lang auf 1 gesetzt. Tritt *nach* der Zeit ZEIN ein zweiter Puls auf, wird der DOPP-Ausgang einen Taskzyklus lang auf 1 gesetzt.

Treten zwei Pulse *während* der Zeit ZEIN auf, wird dies als *einen* Puls gewertet.

Dieser Mechanismus kann zum entprellen verwendet werden. Meistens wird aber ZEIN auf 0 gesetzt.

Die Werte am ZEIN- und ZDOP-Eingang werden jeweils beim ersten Puls übernommen. Danach hat eine Änderung dieser Werte, bis die Zeit am ZDOP-Eingang abgelaufen ist, keinen Einfluss auf die Funktion des KLICK-Blockes.

Zeitdiagramm



Ein-/Ausgänge

E Eingang (Bit)

ZEIN Zeit in Taskzyklen, in der zwei Pulse als einzelner Klick gelten (Word)

ZDOP Zeit in Taskzyklen, nach der die Ausgänge gesetzt werden (Word)

EINZ Ausgang: Zeigt einen einzelnen Klick an (Bit)

DOPP Ausgang: Zeigt einen Doppelklick an (Bit)

Formeln

Der Eingangswert für ZEIN und ZDOP kann folgendermassen berechnet werden:

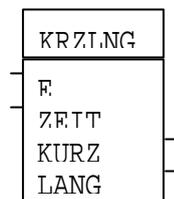
$$\text{ZEIN} = \frac{\text{Zeit [s]}}{\text{Taskzykluszeit [s]}}$$
$$\text{ZDOP} = \frac{\text{Zeit [s]}}{\text{Taskzykluszeit [s]}}$$

EIB System

Funktionsmodul und FMTool

KRZLNG

Erkennen von kurzen und langen Signalen



Funktion

Der KRZLNG-Block erkennt, ob ein Eingangssignal kürzer oder länger als eine spezifische Zeit gesetzt ist. Damit können z.B. kurze oder lange Signale von Tastensensoren unterschieden werden.

Liegt das Eingangssignal E länger an als ZEIT, wird der Ausgang LANG für genau einen Taskzyklus gesetzt; liegt es kürzer an als ZEIT, wird KURZ gesetzt.

Ein-/Ausgänge

E Eingang (Bit)

ZEIT Zeit (Word)

KURZ Ausgang: Zeigt ein kurzes Eingangssignal an (Bit)

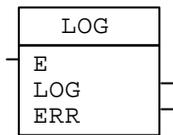
LANG Ausgang: Zeigt ein langes Eingangssignal an (Bit)

Formeln

$$ZEIT = \frac{Zeit[s]}{Taskzykluszeit[s]}$$

LOG

Natürlicher Logarithmus



Funktion

Der LOG-Block berechnet den natürlichen Logarithmus mit der Basis e (e = 2.71828).

Ist das Eingangssignal negativ, ist der Ausgangswert unbestimmt und das Fehlerbit wird gesetzt.

Ein-/Ausgänge

E Eingangssignal (Fließkomma; Value)
LOG Ausgangssignal (Fließkomma; Value)
ERR Fehler

Formeln

Blockfunktion:

$$\text{LOG} = \ln(E)$$

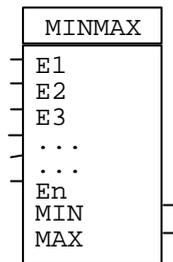
Rechengenauigkeit

Die Auflösung des Datentypen Value ist beschränkt auf 0.000'313. Darum kann der Ausgangswert des LOG-Blockes bei kleinen Zahlen am Eingang, hohe relative Fehler aufweisen.

Siehe auch Block:

EXP

MINMAX (Neu)



Funktion

Der MINMAX-Block sucht den kleinsten und den grössten anliegenden Wert an allen Eingängen (E1 bis En) und kopiert den Kleinsten an den Ausgang (MIN) und den Grössten an den Ausgang (MAX).

Der kleinste Wert wird an Ausgang (MIN) gelegt.

Der grösste Wert wird an Ausgang (MAX) gelegt.

Datentypen

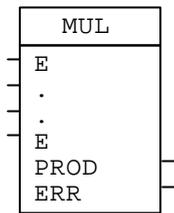
Byte, Date, Sint, Time, Value, Word

Ein-/Ausgänge

E1	Werteingang
E2	Werteingang
E3	Werteingang
En	Werteingang
MIN	Minimalwert
MAX	Maximalwert

MUL

Multiplizierer



Funktion

Der MUL-Block multipliziert mehrere Faktoren miteinander. Die Anzahl der Eingänge ist variabel zwischen 2 und 16. Geht das Produkt oder ein Zwischenresultat über den Bereich des Datentypes hinaus, ist der Ausgang undefiniert und das Überlaufbit wird gesetzt.

Datentypen

Ganzzahl (Sint), Fließkomma (Value), Word

Ein-/Ausgänge

E Faktoren
PROD Produkt
ERR Überlauf (Bit)

Formeln

Blockfunktion:

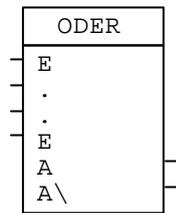
$$\text{PROD} = E1 \cdot E2 \cdot \dots \cdot E_n$$

Siehe auch Block:

DIV, ADD, SUB

ODER

Logisches ODER



Funktion

Der ODER-Block macht eine logische ODER-Verknüpfung mit mehreren Eingängen. Ist mindestens ein Eingang 1, ist der A-Ausgang 1. Die Anzahl der Eingänge ist variabel zwischen 2 und 16.

Wahrheitstabelle

E	E	A	A\
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

Ein-/Ausgänge

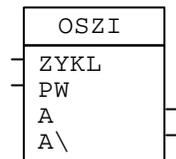
- E Eingangssignale (Bit)
- A nichtinvertierter Ausgang (Bit)
- A\ invertierter Ausgang (Bit)

Siehe auch Block:

UND

OSZI

Oszillator



Funktion

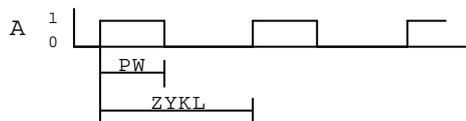
Der OSZI-Block generiert periodische Rechtecksignale. Die Zykluszeit und die Pulsweite sind programmierbar.

Eine Änderung des Eingangswertes am ZYKL- oder am PW-Eingang, wirkt sich erst zu Beginn der nächsten Periode aus (bei steigender Flanke am A-Ausgang).

Nach einem Reset des Funktionsmoduls startet der OSZI-Block sofort mit einer steigenden Flanke am A-Ausgang.

Ist der PW-Eingang 0, bleibt der A-Ausgang auf 0. Ist der Wert am PW-Eingang grösser als der Wert am ZYKL-Eingang, oder ist der ZYKL-Eingang 0, so bleibt der A-Ausgang 1.

Zeitdiagramm



Ein-/Ausgänge

ZYKL Zykluszeit in Taskzyklen (Word)

PW Pulsweite in Taskzyklen (Word)

A nichtinvertierter Ausgang (Bit)

A\ invertierter Ausgang (Bit)

Formeln

Der Eingangswert für ZYKL und PW kann folgendermassen berechnet werden:

$$ZYKL = \frac{\text{Zykluszeit [s]}}{\text{Taskzykluszeit [s]}}$$

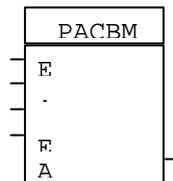
$$PW = \frac{\text{Pulsweite [s]}}{\text{Taskzykluszeit [s]}}$$

Siehe auch Block:

ALARM

PACBM

Packe Byte in Max



Funktion

Der PACBM-Block packt mehrere Bytes in ein Signal vom Datentyp Max. Die Anzahl der Eingänge ist variabel zwischen 1 und 14.

Ausgangsformat

Vorsicht: Das Ausgangssignal vom Typ MAX ist nur für den Teil gültig, der einen entsprechenden Eingang hat, d.h., wenn z.B. als Eingänge nur 4 Bytes verwendet werden, sind die Positionen 5...14 beim MAX-Ausgangssignal nicht definiert.



Ein-/Ausgänge

E Eingänge (Byte)

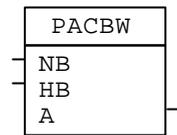
A Ausgang (Max)

Siehe auch Block:

TEILMB, PACM, TEILM

PACBW

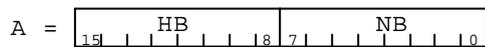
Packe Byte in Word



Funktion

Der PACB-Block packt ein niederwertiges und ein höherwertiges Byte in ein Word.

Ausgangsformat



Ein-/Ausgänge

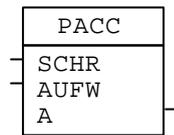
NB niederwertiges Byte
HB höherwertiges Byte
A Ausgang (Word)

Siehe auch Block:

TEILWB

PACC

Packe Control (Steuerung)



Funktion

Der PACC-Block packt ein Richtungsbit und eine Schrittweite in ein Signal vom Datentyp Steuerung (Control). Ist die Schrittweite grösser als 7, ist der Ausgang unbestimmt (die Schrittweite wird mit 3 Bit abgebildet).

Ein-/Ausgänge

SCHR Schrittweite (Byte)

- 0 = stop
- 1 = 100% auf/ab dimmen
- 2 = 50% auf/ab dimmen
- 3 = 25% auf/ab dimmen
- 4 = 13% auf/ab dimmen
- 5 = 6.3% auf/ab dimmen
- 6 = 3.1% auf/ab dimmen
- 7 = 1,6% auf/ab dimmen

AUFW Aufwärts (Richtung) (Bit)

- 0 = abwärts dimmen
- 1 = aufwärts dimmen

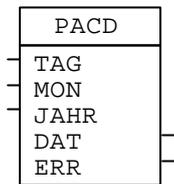
A Ausgang (Steuerung ; Control)

Siehe auch Block:

TEILC

PACD

Packe Datum (Date)



Funktion

Der PACD-Block packt drei Signale (Tag, Monat und Jahr) in ein Signal vom Datentyp Date. Hat Tag, Monat oder Jahr einen ungültigen Wertebereich, ist der Ausgang Null und das Fehlerbit ist gesetzt.

Ein-/Ausgänge

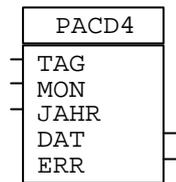
TAG Tag (Byte). Wertebereich: 1 .. 31
MON Monat (Byte). Wertebereich: 1 .. 12
JAHR Jahr (Byte). Wertebereich: 0 .. 99
DAT Datum (Date)
ERR Fehler (Bit)

Siehe auch Block:

PACD4, TEILD4, TEILD

PACD4

Packe Datum (Jahr 4-stellig)



Funktion

Der PACD-Block packt drei Signale (Tag, Monat und Jahr) in ein Signal vom Datentyp Date. Hat Tag, Monat oder Jahr einen ungültigen Wertebereich, ist der Ausgang Null und das Fehlerbit ist gesetzt.

Ein-/Ausgänge

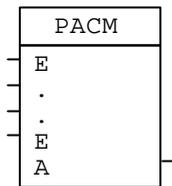
TAG Tag (Byte). Wertebereich: 1 .. 31
MON Monat (Byte). Wertebereich: 1 .. 12
JAHR Jahr (Byte). Wertebereich: 1990 .. 2089
DAT Datum (Date)
ERR Fehler (Bit)

Siehe auch Block:

PACD, TEILD, TEILD4

PACM

Packe Word in Max



Funktion

Der PACM-Block packt mehrere Words in ein Signal vom Datentyp Max. Die Anzahl der Eingänge ist variabel zwischen 1 und 7.

Ausgangsformat

Vorsicht: Das niederwertige Byte ist jeweils vor dem höherwertigen Byte im Datentyp Max abgelegt, d.h., das niederwertige Byte von Eingang 1 ist das erste Byte am Ausgang.



Ein-/Ausgänge

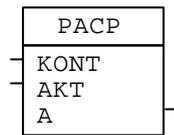
E Eingänge (Word)
A Ausgang (Max)

Siehe auch Block:

TEILM, PACBM, TEILMB

PACP

Packe Priorität (Pcontrol)



Funktion

Der PACP-Block packt ein Kontrollbit und ein Aktionsbit in ein Signal vom Datentyp Priorität (Pcontrol).

Wahrheitstabelle

KONT	AKT	Funktion
0	0	Kontrolle ausgeschaltet, Aktionsbit nicht aktiv
0	1	Kontrolle ausgeschaltet, Aktionsbit nicht aktiv
1	0	Kontrolle eingeschaltet, Aktion aus
1	1	Kontrolle eingeschaltet, Aktion ein

Ein-/Ausgänge

KONT Kontrollbit, Bit 1 auf dem Bus (Bit)

AKT Aktionsbit, Bit 0 auf dem Bus (Bit)

A Ausgang (Priorität; Pcontrol)

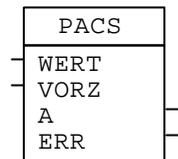
Siehe auch Block:

TEILP

(Vorsicht: Beim TEILP-Block ist die Reihenfolge der KONT- und AKT-Ausgänge umgekehrt.)

PACS

Packe Sint (Ganzzahl)



Funktion

Der PACS-Block packt einen Wert vom Datentyp Word und ein Vorzeichenbit in ein Signal vom Datentyp Ganzzahl (Sint). Eine 0 bedeutet positives Vorzeichen, eine 1 bedeutet negatives Vorzeichen.

Ist WERT zu gross für den Datentyp Sint, ist der Ausgangswert unbestimmt und das Überlaufbit ist gesetzt. Der Wertebereich für Ganzzahl (Sint) reicht von -32768 bis +32767.

Ein-/Ausgänge

WERT positiver Wert (Word)
VORZ Vorzeichen (Bit)
0 = positiv
1 = negativ
A Ausgang (Ganzzahl; Sint)
ERR Überlauf (Bit)

Siehe auch Block:

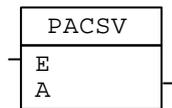
TEILS

EIB System

Funktionsmodul und FMTool

PACSV

Packe Sint (Ganzzahl) in Value (Fließkomma)



Funktion

Der PACSV-Block packt ein Signal vom Datentyp Ganzzahl (Sint) in ein Signal vom Datentyp (Fließkomma (Value)).

Ein-/Ausgänge

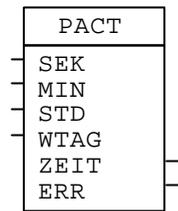
- E Eingang (Ganzzahl; Sint)
- A Ausgang (Fließkomma; Value)

Siehe auch Block:

PACVS

PACT

Packe Time (Zeit)



Funktion

Der PACT-Block packt vier Signale (Sekunden, Minuten, Stunden und Wochentag) in ein Signal vom Datentyp Zeit (Time). Haben Sekunden, Minuten, Stunden oder der Wochentag einen ungültigen Wertebereich, ist der Ausgang Null und das Fehlerbit wird gesetzt.

Ein-/Ausgänge

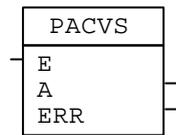
SEK Sekunden (Byte). Wertebereich: 0 .. 59
MIN Minuten (Byte). Wertebereich: 0 .. 59
STD Stunden (Byte). Wertebereich: 0 .. 23
WTAG Wochentag (Byte)
 0 = kein Tag (neutral)
 1 = Montag
 7 = Sonntag
ZEIT Zeit (Time)
ERR Fehler (Bit)

Siehe auch Block:

TEILT

PACVS

Packe Value (Fließkomma) in Sint (Ganzzahl)



Funktion

Der PACVS-Block packt ein Signal vom Datentyp Fließkomma (Value) in ein Signal vom Datentyp Ganzzahl (Sint). Es wird immer abgerundet.

Ist das Eingangssignal zu gross für den Datentyp Ganzzahl (Sint), ist der Ausgangswert unbestimmt und das Überlaufbit ist gesetzt. Der Wertebereich für Ganzzahl (Sint) reicht von -32768 bis $+32767$.

Ein-/Ausgänge

E Eingang (Fließkomma; Value). Wertebereich: $-32768.0 .. +32767.9$

A Ausgang (Ganzzahl; Sint)

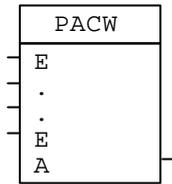
ERR Überlauf (Bit)

Siehe auch Block:

PACSV

PACW

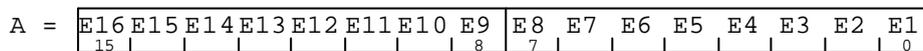
Packe Word



Funktion

Der PACW-Block packt mehrere Bits in ein Word. Die Anzahl der Eingänge ist variabel zwischen 1 und 16. Eingang Nummer 1 ist das niederwertigste Bit. Werden weniger als 16 Eingänge verwendet, sind die höherwertigen Bits 0.

Ausgangsformat



Ein-/Ausgänge

E Eingänge (Bit)
A Ausgang (Word)

Siehe auch Block:

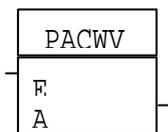
TEILW

EIB System

Funktionsmodul und FMTool

PACWV

Packe Word in Value (Fließkomma)



Funktion

Der PACWV-Block packt ein Signal vom Datentyp Word in ein Signal vom Datentyp Fließkomma (Value).

Ein-/Ausgänge

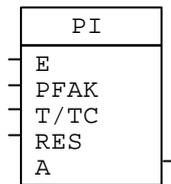
E Eingang (Word)
A Ausgang (Fließkomma; Value)

Siehe auch Block:

TEILVW

PI

PI-Glied



Funktion

Dies ist ein PI-Glied für Regelaufgaben. Es besteht aus einem Proportionalteil und einem Integratoranteil. Mit Reset wird der Integratoranteil auf Null gesetzt.

Damit eine vernünftige Rechengenauigkeit erreicht wird, soll die Taskzykluszeit mindestens 10 mal kleiner sein als die Periodendauer der kritischen Grenzfrequenz des Regelsystems.

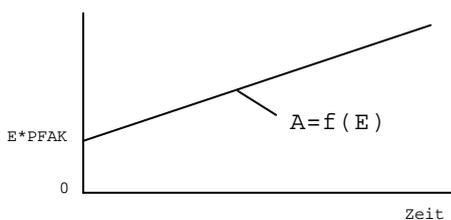
Der Ausgangswert ist auf den Wertebereich des Datentypen Value begrenzt. Ist T/TC negativ oder Null, verändert sich der Integratorwert nicht. Ist der Proportionalfaktor negativ, so wird er auf Null gesetzt.

Die Auflösung des Ausgangssignals ist beschränkt. Ist die theoretische Änderung des Ausgangssignals pro Taskzyklus kleiner als 0.000'313, ändert sich der Ausgangswert nicht. Der Wert am T/TC-Eingang darf also nicht zu hoch gewählt werden.

Die Werte am PFAK- und am T/TC-Eingang dürfen zu jedem beliebigen Zeitpunkt geändert werden.

Nach einem Reset am Funktionsmodul startet der Integrationswert bei 0.

Sprungantwort



Ein-/Ausgänge

- E Eingangssignal (Fließkomma; Value)
- P_{FAK} Proportionalfaktor (Fließkomma; Value)
- T/TC Zeitkonstante (Fließkomma; Value)
- RES Reset (Bit)
- Y Integrationswert (Fließkomma; Value)
- A Ausgangssignal (Fließkomma; Value)

Formeln

Blockfunktion:

$$Y = Y_{\text{alt}} + E/(T/TC)$$

Y_{alt} bedeutet: Integrationswert aus dem letzten Taskzyklus.

$$A = Y + E * P_{\text{FAK}}$$

Der Eingangswert für T/TC kann folgendermassen berechnet werden:

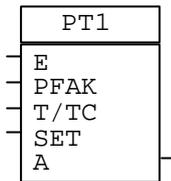
$$T / TC = \frac{\text{Zeitkonstante [s]}}{\text{Taskzykluszeit [s]}}$$

Siehe auch Block:

DT1, INTE, PT1

PT1

PT1-Glied



Funktion

Dies ist ein PT1-Glied für Regelaufgaben. Es basiert auf dem Näherungsalgorithmus von Gauss. Mit SET wird der Ausgang auf den Eingangswert gesetzt.

Damit eine vernünftige Rechengenauigkeit erreicht wird, soll die Taskzykluszeit mindestens 10 mal kleiner sein als die Periodendauer der kritischen Grenzfrequenz des Regelsystems.

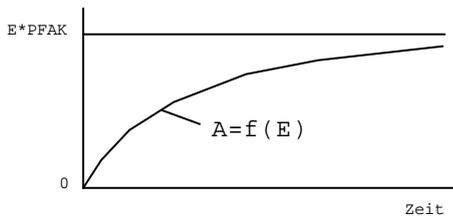
Der Ausgangswert ist auf den Wertebereich des Datentypen Value begrenzt. Ist T/TC negativ, oder ist PFAK negativ, verändert sich der Ausgangswert nicht.

Die Auflösung des Ausgangssignals ist beschränkt. Ist die theoretische Änderung des Ausgangssignals pro Taskzyklus kleiner als 0.000'313, ändert sich der Ausgangswert nicht. Der Wert am T/TC-Eingang darf also nicht zu hoch gewählt werden.

Die Werte am PFAK- und am T/TC-Eingang dürfen zu jedem beliebigen Zeitpunkt geändert werden.

Nach einem Reset am Funktionsmodul startet die Ausgangskurve bei 0.

Sprungantwort



Ein-/Ausgänge

- E Eingangssignal (Fließkomma; Value)
- PFAK Proportionalfaktor (Fließkomma; Value)
- T/TC Zeitkonstante (Fließkomma; Value)
- SET Set (Bit)
- A Ausgangssignal (Fließkomma; Value)

Formeln

Blockfunktion:

$$A = (\text{PFAK} * E + A_{\text{alt}} * (T/TC)) / (1 + (T/TC))$$

A_{alt} bedeutet: Ausgangswert aus dem letzten Taskzyklus.

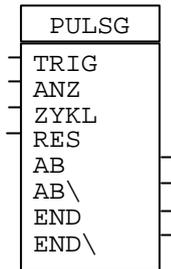
Der Eingangswert für T/TC kann folgendermassen berechnet werden:

$$T / TC = \frac{\text{Zeitkonstante [s]}}{\text{Taskzykluszeit [s]}}$$

Siehe auch Block:

DT1, INTE, PI

PULSG (Neu)



Funktion

Der PULSG-Block sendet eine bestimmte Anzahl (ANZ) Pulse an den Ausgang (AB). Die Länge der Pulse werden durch die Taskzykluszeit (z.B.100ms) bestimmt. Der zeitliche Abstand der Pulse wird durch den Eingang (ZYKL) Taskzykluszeit festgelegt.

Der Start einer Pulsreihe wird durch eine steigende Flanke (0 auf 1) am Eingang (TRIG) ausgelöst. Nach Abschluss einer Pulsreihe wird der Ausgang (END) für eine Tasklänge gesetzt.

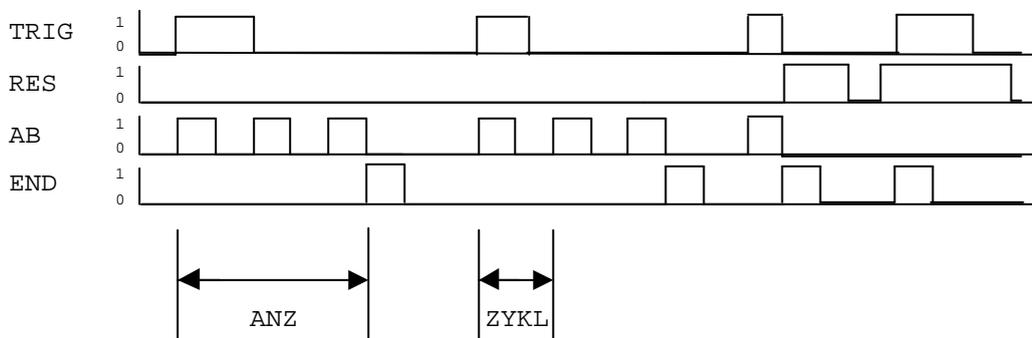
Die Pulsreihe kann mit dem Eingang (TRIG) jederzeit neu gestartet werden (retriggerbar). Dabei werden die Eingänge (ANZ und ZYKL) neu gelesen. Der Ausgang (END) wird in diesem Fall nicht gesetzt.

Im weiteren kann die Pulsreihe mit dem Eingang (RES) gestoppt werden. Der Ausgang (END) wird in diesem Fall sofort gesetzt.

Anwendungsbeispiel

Bei einer Storenzentralsteuerung benötigen wir den Aufwippbefehl für die Lamellen, der Wippbefehl erfordert zB. 3 Impulse um die Lamellenstellung einzuregulieren. Durch das Signal am Eingang (TRIG) generiert der Block die Pulse und kopiert sie an den Ausgang (AB) der mit dem Jalusieaktor verbunden ist, am Ende der Pulsreihe wird das Signal (END) gesendet.

Zeitdiagramm



Datentyp
Word

EIB System

Funktionsmodul und FMTool

Ein-/Ausgänge

TRIG	Trigger (Bit)
ANZ	Anzahl Impulse
ZYKL	Zykluszeit
RES	Reset (Bit)
AB	Ausgang (Bit)
AB\	Ausgang invertiert (Bit)
END	Endsignal (Bit)
END\	Endsignal invertiert (Bit)

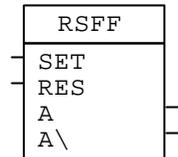
Formeln

Der Eingangswert für ZYKL kann folgendermassen berechnet werden:

$$ZYKL = \frac{\text{Zykluszeit [s]}}{\text{Taskzykluszeit [s]}}$$

RSFF

RS-Flip-Flop



Funktion

Der RSFF-Block ist ein Set-Reset-Flip-Flop. Ist der Set-Eingang 1, wird der A-Ausgang auf 1 gesetzt. Ist der Reset-Eingang 1, wird der A-Ausgang auf 0 gesetzt. Der Reset-Eingang hat die höhere Priorität als der Set-Eingang.

Nach einem Reset am Funktionsmodul ist der A-Ausgang 0.

Wahrheitstabelle

SET	RES	A	A\'
0	0	A	A\'
0	1	0	1
1	0	1	0
1	1	0	1

Ein-/Ausgänge

SET Set-Eingang (Bit)

RES Reset-Eingang (Bit)

A nichtinvertierter Ausgang (Bit)

A\' invertierter Ausgang (Bit)

Siehe auch Block:

TFF

SETD

Setze Datum



Funktion

Der SETD-Block setzt das Systemdatum des Funktionsmoduls auf das gegebene Datum, wenn der SET-Eingang 1 ist.

Das neue Datum ist erst im nächsten Taskzyklus gesetzt.

Der SETD-Block prüft die Gültigkeit des Datums nicht. Wird ein ungültiges Datum gesetzt, ist das Systemdatum unbestimmt.

Ein-/Ausgänge

DAT Datum (Date)

SET Setzen (Bit)

Siehe auch Block:

SETT, GIBD, PACD, TEILD

SETT

Setze Zeit



Funktion

Der SETT-Block setzt die Systemzeit des Funktionsmoduls auf die gegebene Zeit, wenn der SET-Eingang 1 ist.

Die neue Systemzeit ist erst im nächsten Taskzyklus gesetzt.

Der SETT-Block prüft die Gültigkeit der Zeit nicht. Wird eine ungültige Zeit gesetzt, ist die Systemzeit unbestimmt.

Der Wochentag kann unabhängig vom Systemdatum gesetzt werden. Es besteht also kein Zusammenhang zwischen dem Wochentag und dem Datum. Wird der Wochentag auf Noday gesetzt, verändert sich der Wochentag in der Systemuhr nicht.

Ein-/Ausgänge

ZEIT Zeit (Time)

SET Setzen (Bit)

Siehe auch Block:

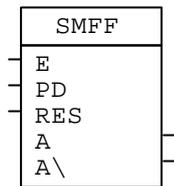
SETD, GIBT, PACT, TEILT

EIB System

Funktionsmodul und FMTool

SMFF

Statisches, Monostabiles Flip-Flop



Funktion

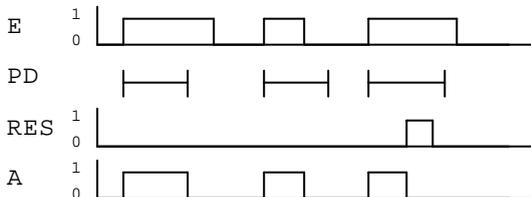
Der SMFF-Block ist ein monostabiles Flip-Flop. Er verkürzt einen Impuls am Eingang auf eine einstellbare Pulsweite.

Der RES-Eingang setzt das Ausgangssignal jederzeit auf Null.

Die Dauer des Pulses wird am PD-Eingang eingestellt. Während der A-Ausgang 1 ist, hat eine Änderung des PD-Einganges keinen Einfluss auf die Pulsdauer. Ist PD 0, bleibt auch der Ausgang 0.

Ist der E-Eingang auf den Wert 1 initialisiert, wird dies nach einem Reset des Funktionsmoduls als steigende Flanke interpretiert und der Puls am Ausgang wird gestartet.

Zeitdiagramm



Ein-/Ausgänge

- E Eingangssignal (Bit)
- PD Pulsdauer (Word)
- RES Reset (Bit)
- A nichtinvertierter Ausgang (Bit)
- A\ invertierter Ausgang (Bit)

Formeln

Der Eingangswert für PD kann folgendermassen berechnet werden:

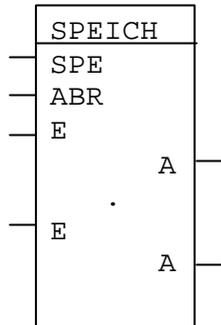
$$PD = \frac{\text{Pulsdauer [s]}}{\text{Taskzykluszeit [s]}}$$

Siehe auch Block:

DMFF

SPEICH

Speicher



Funktion

Der Funktionsblock SPEICH kann Werte speichern und wieder abrufen. Dabei werden die Werte im batteriegepufferten Speicher abgelegt und bleiben auch bei Stromausfall während 24 Stunden (Gangreserve des Funktionsmoduls) erhalten.

Wenn die beiden Steuereingänge "SPE" und "ABR" inaktiv (logisch 0) sind, arbeitet der SPEICH-Block wie ein Funktionsblock TRA. Die Eingangswerte werden direkt an die Ausgänge geschrieben.

Wenn der "SPE" Steuereingang auf logisch 1 gesetzt wird, wird der Eingangswert in den Speicher kopiert. Beim Setzen des Steuereinganges "ABR" auf logisch 1 wird der Wert des Speichers am Ausgang ausgegeben.

Sind beide Steuereingänge auf logisch 1 gesetzt, so wird der Eingangswert durch den Speicher geschleust und an den Ausgang geschrieben. Dadurch arbeitet der SPEICH - Block wiederum ähnlich wie der TRA-Block, speichert jedoch zugleich die Eingangswerte.

Achtung!

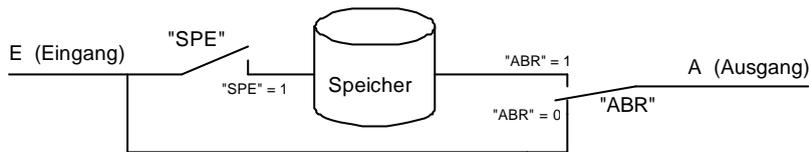
Wird bei der Inbetriebnahme des Funktionsmoduls bzw. nach einem Reset der Steuereingang "SPE" auf logisch 1 gesetzt, so wird dadurch der Wert des Speichers beim erstmaligen Abarbeiten des anliegenden Eingangssignals überschrieben, und der zuvor gespeicherte Wert geht verloren.

*Soll ein gespeicherter Wert auch nach einem Reset noch vorhanden sein, darf der Steuereingang "SPE" (durch das Programm) nach einem Reset **nicht** auf logisch 1 gesetzt werden.*

EIB System

Funktionsmodul und FMTool

Funktionsschema



Datentypen

Alle Datentypen können verwendet werden.

Siehe auch Block:

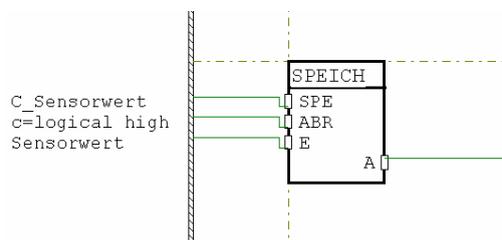
TRA

Anwendungsbeispiel: Gültige Sensorwerte über Reset hinaus

In vielen Anwendungen werden Werte von Sensoren für Vergleichs- oder Regeleroperationen verwendet. Die Sensoren übermitteln ihre Werte bei der Änderung einer entsprechenden physikalischen Größe an das Funktionsmodul.

Von Sensoren z.B. Helligkeitssensoren, welche aufgrund langsamer Helligkeitswechsel ihre Werte nur selten übermitteln, erhält das Funktionsmodul bei einem Reset oder einem kurzen Stromunterbruch lange Zeit keine gültigen Werte. Für eine Steuerung oder Regelung bedeutet ein solcher Zwischenfall die Handlungsunfähigkeit. Ein komplettes Steuersystem kann dadurch über längere Zeit einen Ausfall erleiden.

Mit dem Speicherbaustein kann dieser Ausfall einfach vermieden werden. Dazu wird der "Speicher"-Eingang mit dem Kontrollsignal des entsprechenden Sensorsignales verbunden. Der Eingang "Abrufen" wird mittels einer Konstanten auf logisch 1 gesetzt und das eigentliche Sensorsignal mit dem Eingang verbunden. Dadurch ist am Ausgang des Speicherbausteines immer der letztgültige Sensorwert vorhanden.

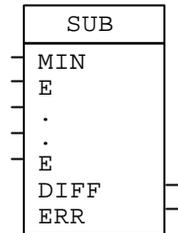


Vorsicht:

Bei der erstmaligen Inbetriebnahme muss sichergestellt sein, dass der Sensor einen gültigen Anfangswert übermittelt.

SUB

Subtrahierer



Funktion

Der SUB-Block subtrahiert ein oder mehrere Subtrahende von einem Minuend. Die Anzahl der Subtrahenden ist variabel zwischen 1 und 16.

Geht das Resultat oder ein Zwischenresultat über den Bereich des Datentypen hinaus, ist der Ausgang undefiniert und das Überlaufbit wird gesetzt.

Datentypen

Sint, Value, Word

Ein-/Ausgänge

MIN Minuend
E Subtrahenden
DIFF Differenz
ERR Überlauf (Bit)

Formeln

Blockfunktion:

$$\text{DIFF} = (((\text{MIN} - \text{E1}) - \text{E2}) - \dots) - \text{En}$$

Siehe auch Block:

ADD, MUL, DIV

SUHR

Schaltuhr



Funktion

Die Schaltuhr schaltet den Ausgang bei programmierbaren Zeiten ein und aus. Am ZEIT-Eingang muss die aktuelle Zeit vorhanden sein. Es können bis zu acht Einschaltbereiche angegeben werden. Ist die aktuelle Zeit innerhalb eines Einschaltbereiches, ist der A-Ausgang 1.

Wie die Einschaltbereiche definiert werden, zeigt untenstehendes Beispiel.

Enthält die aktuelle Zeit keinen Tag (noday), werden die Einschaltbereiche, welche einen Wochentag enthalten, nicht ausgewertet.

Die Werte an den ZEIN- und ZAUS-Eingängen können zu jedem beliebigen Zeitpunkt geändert werden.

Beispiel

ZEIN, ZAUS	Bereich	Beispiel
ZEIN < ZAUS ZEIN und ZAUS haben noday	jeden Tag von ZEIN bis ZAUS eingeschaltet	ZEIN = no, 11:50:00 ZAUS = no, 13:10:00 Mittags eingeschaltet
ZEIN > ZAUS ZEIN und ZAUS haben noday	jede Nacht von ZEIN bis ZAUS eingeschaltet	ZEIN = no, 18:30:00 ZAUS = no, 05:00:00 in der Nacht eingeschaltet
ZEIN und ZAUS haben einen Wochentag	einmal pro Woche, von ZEIN bis ZAUS eingeschaltet	ZEIN = Fri, 23:50:00 ZAUS = Mon, 02:00:00 am Wochendende eingeschaltet

Ein-/Ausgänge

- ZEIT Zeit-Eingang (Zeit; Time)
- ZEIN um diese Zeit wird eingeschaltet (Zeit; Time)
- ZAUS um diese Zeit wird ausgeschaltet (Zeit; Time)
- A nichtinvertierter Ausgang (Bit)
- A\ invertierter Ausgang (Bit)

TEILC

Teile Control (Steuerung)



Funktion

Der TEILC-Block teilt ein Signal vom Datentyp Steuerung (Control) in eine Schrittweite und ein Richtungsbit.

Ein-/Ausgänge

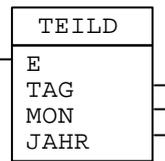
E	Eingang (Steuerung; Control)
SCHR	Schrittweite (Byte) 0 = stop 1 = 100% auf/ab dimmen 2 = 50% auf/ab dimmen 3 = 25% auf/ab dimmen 4 = 13% auf/ab dimmen 5 = 6.3% auf/ab dimmen 6 = 3.1% auf/ab dimmen 7 = 1,6% auf/ab dimmen
AUFW	Aufwärts (Richtung) (Bit) 0 = abwärts dimmen 1 = aufwärts dimmen

Siehe auch Block:

PACC

TEILD

Teile Datum (Date)



Funktion

Der TEILD-Block teilt ein Signal vom Datentyp Datum (Date) in Tag, Monat und Jahr.

Ein-/Ausgänge

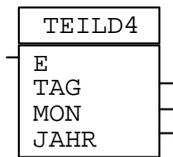
E Eingang (Date)
TAG Tag (Byte)
MON Monat (Byte)
JAHR Jahr (Byte). Wertebereich: 0 .. 99

Siehe auch Block:

TEILD4, PACD4, PACD

TEILD4

Teile Datum (Jahr 4-stellig)



Funktion

Der TEILD-Block teilt ein Signal vom Datentyp Datum (Date) in Tag, Monat und Jahr.

Ein-/Ausgänge

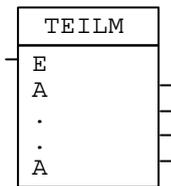
E Eingang (Datum; Date)
TAG Tag (Byte)
MON Monat (Byte)
JAHR Jahr (Byte). Wertebereich: 1990 .. 2089

Siehe auch Block:

PACD, PACD4, TEILD

TEILM

Teile Max in Word

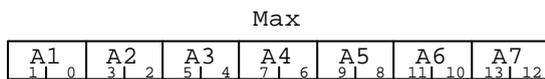


Funktion

Der TEILM-Block teilt ein Signal vom Datentyp Max in Words. Die Anzahl Ausgänge ist variabel zwischen 1 und 7.

Aufteilung in Words

Vorsicht: Das erste Byte vom Eingang wird im niederwertigen Byte von Ausgang 1 abgelegt.



Ein-/Ausgänge

E Eingang (Max)

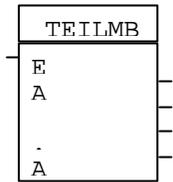
A Ausgänge (Word)

Siehe auch Block:

PACM, TEILMB, PACBM

TEILMB

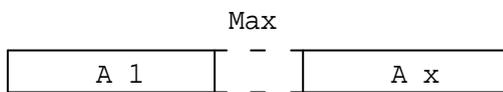
Teile Max in Byte



Funktion

Der TEILMB-Block teilt ein Signal vom Datentyp Max in Bytes. Die Anzahl Ausgänge ist variabel zwischen 1 und 14.

Aufteilung in Bytes



Ein-/Ausgänge

E Eingang (Max)
A Ausgänge (Byte)

Siehe auch Block:

PACM, TEILM, PACM

TEILP

Teile Priorität (Pcontrol)



Funktion

Der TEILP-Block teilt ein Signal vom Datentyp Priorität (Pcontrol) in ein Aktionsbit und ein Kontrollbit.

Wahrheitstabelle

KONT	AKT	Funktion
0	0	Kontrolle ausgeschaltet, Aktionsbit nicht aktiv
0	1	Kontrolle ausgeschaltet, Aktionsbit nicht aktiv
1	0	Kontrolle eingeschaltet, Aktion aus
1	1	Kontrolle eingeschaltet, Aktion ein

Ein-/Ausgänge

E Eingang (Pcontrol)

AKT Aktionsbit, Bit 0 vom Eingang (Bit)

KONT Kontrollbit, Bit 1 vom Eingang (Bit)

Siehe auch Block:

PACP

(*Vorsicht:* Beim PACP-Block ist die Reihenfolge der KONT- und AKT-Ausgänge umgekehrt.)

TEILS

Teile Sint (Ganzzahl)



Funktion

Der TEILS-Block teilt ein Signal vom Datentyp Ganzzahl (Sint) in einen Absolutwert vom Datentyp Word und in ein Vorzeichen.

Ein-/Ausgänge

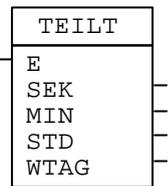
E Eingang (Ganzzahl; Sint)
WERT Absolutwert (Word)
VORZ Vorzeichen (Bit)
0 = positiv
1 = negativ

Siehe auch Block:

PACS

TEILT

Teile Time (Zeit)



Funktion

Der TEILT-Block teilt ein Signal vom Datentyp Zeit (Time) in Sekunden, Minuten, Stunden und Wochentag.

Ein-/Ausgänge

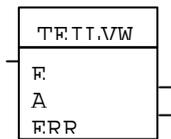
E	Eingang (Zeit; Time)
SEK	Sekunden (Byte)
MIN	Minuten (Byte)
STD	Stunden (Byte)
WTAG	Wochentag (Byte)
	0 = kein Tag (neutral)
	1 = Montag
	7 = Sonntag

Siehe auch Block:

PACT

TEILVW

Teile Value (Fließkomma) in Word



Funktion

Der TEILVW-Block wandelt ein Signal vom Datentyp Fließkomma (Value) in ein Signal vom Datentyp Word. Sollte der Ausgang ausserhalb des Wertbereichs zu liegen kommen, wird das Fehlerbit ERR gesetzt und der Ausgang wird Null.

Ein-/Ausgänge

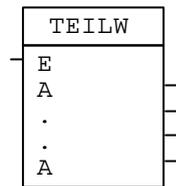
E Eingang (Fließkomma; Value)
A Ausgang (Word)
ERR Fehler (Bit)

Siehe auch Block:

PACWV

TEILW

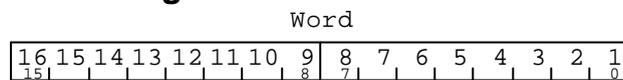
Teile Word



Funktion

Der TEILW-Block teilt ein Signal vom Datentyp Word in mehrere Bits. Die Anzahl der Ausgänge ist variabel zwischen 1 und 16. Das erste Bit ist das niederwertigste Bit.

Aufteilung in Bits



Ein-/Ausgänge

E Eingang (Word)

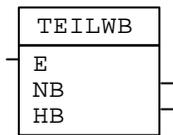
A Ausgänge (Bit)

Siehe auch Block:

PACW

TEILWB

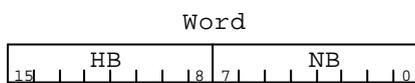
Teile Word in Byte



Funktion

Der TEILWB-Block teilt ein Signal vom Datentyp Word in ein niederwertiges und ein höherwertiges Byte.

Aufteilung in Bytes



Ein-/Ausgänge

E Eingang (Word)
NB niederwertiges Byte
HB höherwertiges Byte

Siehe auch Block:

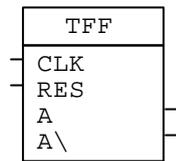
PACBW

EIB System

Funktionsmodul und FMTool

TFF

Toggle-Flip-Flop (Umschalter)



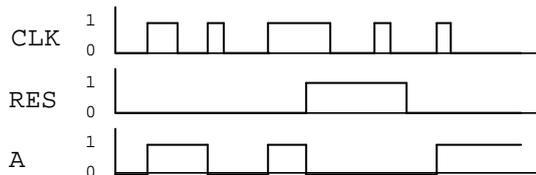
Funktion

Der TFF-Block ist ein Toggle-Flip-Flop. Bei einer steigenden Flanke am CLK-Eingang wechseln die Ausgänge ihren Wert von 0 auf 1 oder von 1 auf 0. Reset setzt den A-Ausgang immer auf 0.

Nach einem Reset am Funktionsmodul ist der A-Ausgang 0.

Ist der CLK-Eingang auf den Wert 1 initialisiert, wird dies nach einem Reset des Funktionsmoduls als steigende Flanke interpretiert und der A-Ausgang wird 1.

Zeitdiagramm



Ein-/Ausgänge

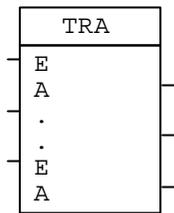
CLK flankengesteuerter Eingang (Bit)
RES Reset (Bit)
A nichtinvertierter Ausgang (Bit)
A\ invertierter Ausgang (Bit)

Siehe auch Block:

RSFF

TRA

Transfer



Funktion

Der TRA-Block transferiert (kopiert) den Wert des Eingangssignals an den Ausgang. Es können bis zu 8 Signale kopiert werden.

Datentypen

Alle Datentypen

Ein-/Ausgänge

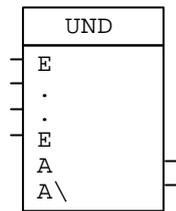
E Eingänge
A Ausgänge

Tips

- Der TRA-Block wird verwendet um ein Signal auf mehrere Signale zu kopieren. Zum Beispiel wenn ein Blockausgang intern verwendet wird (lokales Signal) und gleichzeitig auf ein Ausgangssignal geführt werden muss.

UND

Logisches UND



Funktion

Der UND-Block macht eine logische UND-Verknüpfung mit mehreren Eingängen. Sind alle Eingänge 1, ist der A-Ausgang 1. Die Anzahl der Eingänge ist variabel zwischen 2 und 16.

Wahrheitstabelle

E	E	A	A\'
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

Ein-/Ausgänge

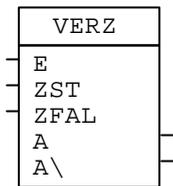
- E Eingänge (Bit)
- A nichtinvertierter Ausgang (Bit)
- A\' invertierter Ausgang (Bit)

Siehe auch Block:

ODER

VERZ

Verzögerer



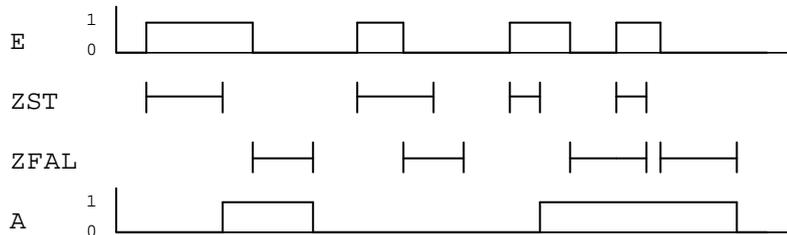
Funktion

Der VERZ-Block verzögert die steigende und die fallende Flanke vom Eingangssignal (Ein- und Ausschaltverzögerung). Die Verzögerungszeiten für das Ansteigen und Abfallen sind einzeln einstellbar. Beide Zeiten können auch auf 0 gesetzt werden.

Werden die Werte am ZST- oder ZFAL-Eingang während der Verzögerung des Ausgangssignales verändert, hat dies keinen Einfluss auf die laufende Verzögerung.

Ist der E-Eingang auf den Wert 1 initialisiert, wird dies nach einem Reset des Funktionsmodules als steigende Flanke interpretiert und der A-Ausgang wird nach der eingestellten Zeit auf 1 gesetzt.

Zeitdiagramm



Ein-/Ausgänge

- E Eingangssignal (Bit)
- ZST Verzögerungszeit der steigenden Flanke (Word)
- ZFAL Verzögerungszeit der fallenden Flanke (Word)
- A nichtinvertierter Ausgang (Bit)
- A\ invertierter Ausgang (Bit)

Formeln

Der Eingangswert für ZST und ZFAL kann folgendermassen berechnet werden:

$$ZST = \frac{\text{Verzögerungszeit [s]}}{\text{Taskzykluszeit [s]}}$$

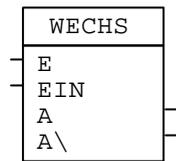
$$ZFAL = \frac{\text{Verzögerungszeit [s]}}{\text{Taskzykluszeit [s]}}$$

WECHS

EIB System

Funktionsmodul und FMTool

Wechsel eines Signals

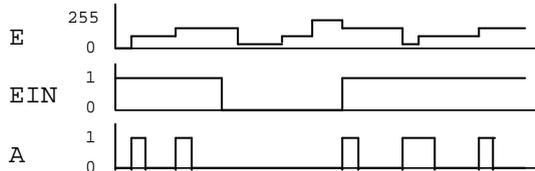


Funktion

Der WECHS-Block stellt Änderungen am Eingang fest und setzt in diesem Fall den Ausgang einen Taskzyklus lang auf 1 (kurzer Puls). Mit dem EIN-Eingang kann die Funktion ein- und ausgeschaltet werden.

Ist der E-Eingang auf einen anderen Wert als 0 initialisiert, wird dies nach einem Reset des Funktionsmoduls als Änderung interpretiert und am Ausgang wird ein Puls erzeugt.

Zeitdiagramm



Datentypen

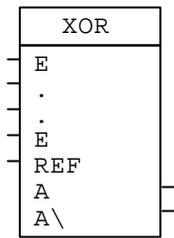
Alle Datentypen

Ein-/Ausgänge

- E Eingangssignal
- EIN Ein-/Ausschalten (Bit)
- A nichtinvertierter Ausgang (Bit)
- A\ invertierter Ausgang (Bit)

XOR

Logisches XOR



Funktion

Der XOR-Block addiert alle Bits an den Eingängen die 1 sind. Ist die Summe dieser Bits nicht 0 und nicht grösser als der Wert am REF-Eingang, ist der A-Ausgang 1. Die Anzahl der Eingänge kann von 2 bis 16 eingestellt werden.

Beispiel

Beispiel mit drei E-Eingängen

E	E	E	REF	A
0	0	0	2	0
1	0	0	2	1
0	1	0	2	1
0	1	1	2	1
1	1	1	2	0
1	0	0	1	1
1	1	0	1	0

Ein-/Ausgänge

E Eingänge (Bit)
 REF Referenzwert (Word)
 A nichtinvertierter Ausgang (Bit)
 A\ invertierter Ausgang (Bit)

Tips

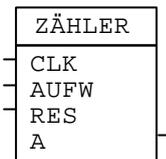
- Ein einfaches XOR-Tor mit zwei Eingängen kann auch mit dem GLEICH-Block realisiert werden (siehe GLEICH-Block).

Siehe auch Block:

GLEICH

ZÄHLER

Zähler



Funktion

Der ZÄHLER-Block zählt die steigenden Flanken am CLK-Eingang. Mit dem AUFW-Eingang kann die Zählrichtung (aufwärts/abwärts) gewählt werden. Reset setzt den Ausgangswert auf 0.

Der Zähler zählt im Kreis, d.h. beim Erreichen des Maximalwertes 32767 wird bei -32768 fortgefahren. In Abwärtszählrichtung umgekehrt.

Nach einem Reset am Funktionsmodul ist der Ausgangswert 0.

Ist der CLK-Eingang auf den Wert 1 initialisiert, wird dies nach einem Reset des Funktionsmoduls als steigende Flanke interpretiert und der Ausgang wird um 1 erhöht.

Wahrheitstabelle

CLK	AUFW	RES	A
↑	1	0	A+1
↑	0	0	A-1
X	X	1	0

↑ steigende Flanke

X kann den Wert 0 oder 1 haben

Datentypen

Der Ausgang kann vom Typ Ganzzahl (Sint) oder Word sein.

Ein-/Ausgänge

CLK flankengesteuerter Eingang (Bit)

AUFW Zählrichtung (Bit)

0 = abwärts

1 = aufwärts

RES Reset

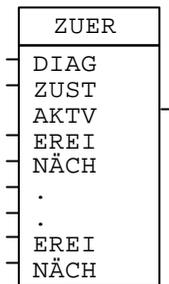
A Ausgang (Ganzzahl; Sint)

Siehe auch Block:

INTE

ZUER

Zustands-Ereignis-Block



Funktion

Mit dem ZUER-Block können Zustands-Ereignisdiagramme programmiert werden. Jeder ZUER-Block, sowie ein ZUINIT-Block und ZUSET-Blöcke, werden über ein gemeinsames Signal am DIAG-Eingang miteinander verbunden.

Ein ZUER-Block entspricht einem bestimmten Zustand im Diagramm. Die Zustandsnummer wird am ZUST-Eingang angegeben. Ist der jeweilige Zustand aktiv, ist der AKTV-Ausgang 1.

In jedem Zustand können Ereignisse auftreten. Diese können am ZUER-Block an den EREI-Eingang angeschlossen werden. Ist nun der betreffende Zustand aktiv und ein Ereignis ist 1, wird in den neuen Zustand umgeschaltet, welcher am NÄCH-Eingang angegeben ist. Treffen gleichzeitig mehrere Ereignisse ein, so wird das erste (das oberste am ZUER-Block) ausgewertet. Der neue Zustand ist immer erst im nachfolgenden Taskzyklus aktiv. D.h. der Zustandswechsel erfolgt nicht sofort.

Die Anzahl der EREI- und NÄCH-Eingänge ist variabel zwischen 1 und 8. Reicht die Anzahl dieser Eingänge nicht aus, können mehrere ZUER-Blöcke mit der gleichen Zustandsnummer eingesetzt werden.

Ein Zustands-Ereignisdiagramm kann bis zu 65'536 verschiedene Zustände haben. Die Anzahl ZUER-Blöcke in einem Diagramm ist durch die Kapazität des Funktionsmoduls beschränkt.

Wichtig

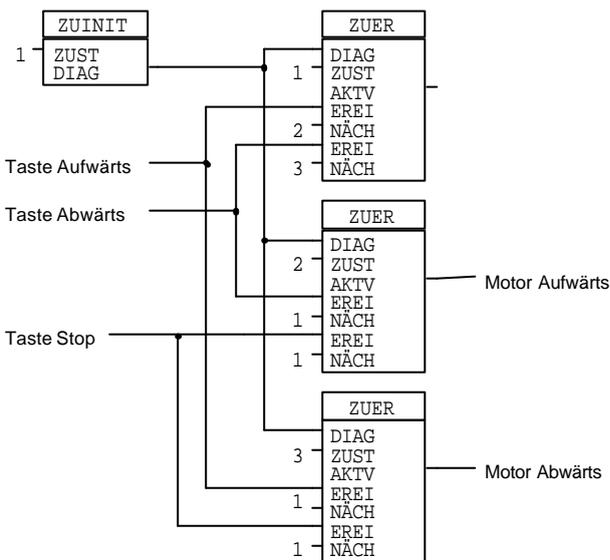
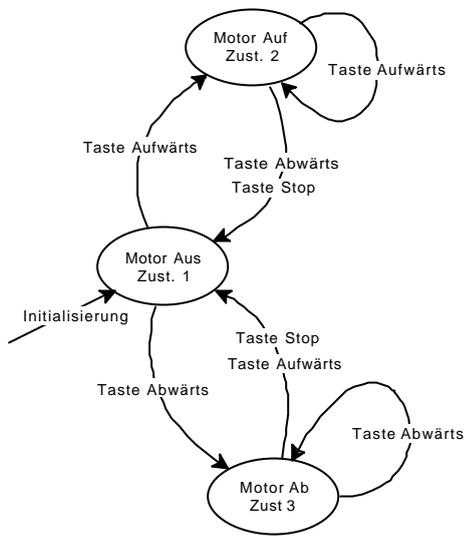
Alle ZUER- und ZUSET-Blöcke vom gleichen Zustands-Ereignisdiagramm müssen sich im gleichen Task befinden.

EIB System

Funktionsmodul und FMTool

Beispiel

Das Zustands-Ereignisdiagramm und die Realisierung mit Funktionsblöcken:



Ein-/Ausgänge

- DIAG Diagrammsignal (Max)
- ZUST Zustandsnummer (Word)
- AKTV Aktiv (Bit)
- EREI Ereignis-Eingänge (Bit)
- NÄCH nächster Zustand nach Ereignis (Word)

Siehe auch Block:

ZUINIT, ZUSET

ZUINIT

Zustand initialisieren



Funktion

Der ZUINIT-Block initialisiert ein Zustands-Ereignisdiagramm. Der Start-Zustand kann am ZUST-Eingang angegeben werden. Jedes Zustands-Ereignisdiagramm muss mit einem ZUINIT-Block verbunden sein.

Wichtig

Dieser Block muss sich im Inittask befinden.

Ein-/Ausgänge

ZUST Initialisierungs-Zustand (Word)

DIAG Diagrammsignal (Max)

Siehe auch Block:

ZUER, ZUSET

ZUSET

Zustand setzen



Funktion

Der ZUSET-Block kann ein Zustands-Ereignisdiagramm zu einem beliebigen Zeitpunkt auf einen bestimmten Zustand setzen. Dazu muss der SET-Eingang auf 1 gesetzt und am ZUST-Eingang der neue Zustand angelegt werden. Der neue Zustand ist erst im nächsten Taskzyklus aktiv.

Am AKT-Ausgang liegt immer die Nummer des aktiven Zustandes, und am LETZ-Ausgang liegt immer die Zustandsnummer vor dem letzten Wechsel an.

Wichtig

Alle ZUER- und ZUSET-Blöcke vom gleichen Zustands-Ereignisdiagramm müssen sich im gleichen Task befinden.

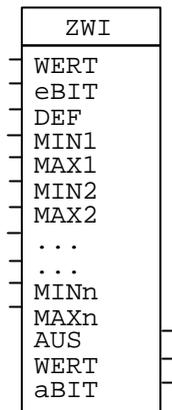
Ein-/Ausgänge

DIAG Diagrammsignal (Max)
SET Setzen eines Zustandes (Bit)
ZUST zu setzender Zustand (Word)
AKT aktueller Zustand (Word)
LETZ letzter Zustand (Word)

Siehe auch Block:

ZUINIT, ZUER

ZWI (Neu)



Funktion

Der ZWI-Block prüft den Eingangswert (WERT), vergleicht den Wert mit den festgelegten Betriebsbereichen (MINn) und (MAXn). Liegt er zwischen einem festgelegten Bereich, kopiert er den Eingangswert (WERT) an den Ausgang (WERT).

Gleichzeitig wird das Signal an (AUS) gelöscht.

Liegt der Eingangswert (WERT) nicht zwischen einer der festgelegten Limiten übernimmt er den Defaultwert (DEF) und kopiert ihn an den Ausgang (WERT).

Gleichzeitig wird das Signal an (AUS) gesetzt.

Wenn der Eingangswert in einem Betriebsbereich liegt, wird der Eingangsstatus (eBIT) an den Ausgangsstatus (aBIT) kopiert. Andernfalls ist der Ausgang (aBIT) "0".

Datentypen

Byte, Date, Sint, Time, Value, Word

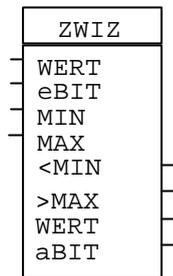
Ein-/Ausgänge

WERT	Werteingang
eBIT	Eingangsstatus (Bit)
DEF	Devaultwert
MINn	Minimalwert
MAXn	Maximalwert
AUS	Ausserhalb (Bit)
WERT	Wertausgang
aBIT	Ausgangsstatus (Bit)

Siehe auch Block:

ZWIZ (Neu)

ZWIZ (Neu)



Funktion

Der ZWIZ-Block prüft, ob der Eingangswert (WERT) im Betriebsbereich (MIN) und (MAX) liegt. Wenn der Eingangswert die Betriebslimite nicht unter-/überschreitet, wird er an den Ausgang (WERT) ausgegeben. Andernfalls wird die entsprechende Limite inklusive Grenzwert (MIN) oder (MAX) an den Ausgang gelegt.

Bei Unterschreitung des Minimums (MIN) wird der Ausgang (<MIN) gesetzt.

Bei Überschreitung des Maximums (MAX) wird der Ausgang (>MAX) gesetzt.

Wenn der Eingangswert im Betriebsbereich liegt, wird der Eingangsstatus (eBIT) an den Ausgangsstatus (aBIT) kopiert. Andernfalls ist der Ausgang (aBIT) "0".

Datentypen

Byte, Date, Sint, Time, Value, Word

Ein-/Ausgänge

WERT	Werteingang
eBIT	Eingangsstatus (Bit)
MIN	Minimalwert
MAX	Maximalwert
<MIN	Minsignal (Bit)
>MAX	Maxsignal (Bit)
WERT	Wertausgang
aBIT	Ausgangsstatus (Bit)

Siehe auch Block:

ZWI (Neu)

