

CAN-CBM-DIO8- Counter

Zählermodul

Hardware-Handbuch

Der Inhalt dieses Handbuches wurde mit größter Sorgfalt erarbeitet und geprüft. **esd** übernimmt jedoch keine Verantwortung für Schäden, die aus Fehlern in der Dokumentation resultieren könnten. Insbesondere Beschreibungen und technische Daten sind keine zugesicherten Eigenschaften im rechtlichen Sinne.

esd hat das Recht, Änderungen am beschriebenen Produkt oder an der Dokumentation ohne vorherige Ankündigung vorzunehmen, wenn sie aus Gründen der Zuverlässigkeit oder Qualitätssicherung vorgenommen werden oder dem technischen Fortschritt dienen.

Sämtliche Rechte an der Dokumentation liegen bei **esd**. Die Weitergabe an Dritte und Vervielfältigung jeder Art, auch auszugsweise, sind nur mit schriftlicher Genehmigung durch **esd** gestattet.

esd electronic system design gmbh

Vahrenwalder Str. 205
30165 Hannover

Tel.: 0511/372 98-0
FAX : 0511/372 98-68
E-Mail: info@esd.electronics.com
Internet: www.esd-electronics.com

Dokument-Datei:	I:\texte\Doku\MANUALS\CAN\Cbm\Counter\CNT-06h.ma6
Datum der Druckvorlagenherstellung:	09.03.2000
Handbuch-Bestellnummer:	C.2830.22

Änderungen in den Kapiteln

Die hier aufgeführten Änderungen im Dokument betreffen sowohl Änderungen in der *Hardware* als auch reine Änderungen in der *Beschreibung* der Sachverhalte.

Kapitel	Änderungen gegenüber Vorversion
-	1. Ausgabe
-	

Weitere technische Änderungen vorbehalten.

Inhalt	Seite
1. Übersicht	3
1.1 Vorwort	3
2. Funktionsweise des Frequenzzählers	5
2.1 Eingangssignale	5
2.1.1 Ladeimpuls 1	5
2.1.2 Ladeimpuls 2	5
2.1.3 Zähltakt	5
2.1.4 Zählrichtung	5
2.2 Vergleicher und Ausgangssignale	6
3. Parametrierung des Zählers	7
Impulszählerstand	7
Impulszähler-Minimum	7
Impulszähler-Maximum	7
Impulszähler-Lade-Wert 1	7
Impulszähler-Lade-Wert 2	7
Impulszähler-Vergleichswert	7
Impulszähler-Konfiguration	7
Impulszähler-Ergebnis-Flags	8
Impulszähler-Lade-Eingang 1	8
Impulszähler-Lade-Eingang 2	8
Impulszähler-Vergleicher-Ausgang 'kleiner'	8
Impulszähler-Vergleicher-Ausgang 'gleich'	8
Impulszähler-Vergleicher-Ausgang 'größer'	8
Bitmaske	9
4. Betrieb des Zählers	11
4.1 Senden eines Prozeßdaten-Telegramms mit Triggerung durch den Impulszähler	12
5. Anhang: Code-Stellen des Zählermoduls	15

1. Übersicht

1.1 Vorwort

Dieser Zusatz zum Systemhandbuch CAN-CBM-DIO8 enthält **ergänzende** Informationen zum CAN-CBM-DIO8-Counter-Modul. Es beschreibt die Funktionalität und Einsatzbereiche des implementierten Impuls- und Frequenzzählers.

Allgemeingültige Informationen zu den Modulen finden Sie im Systemhandbuch CAN-CBM-DIO8.

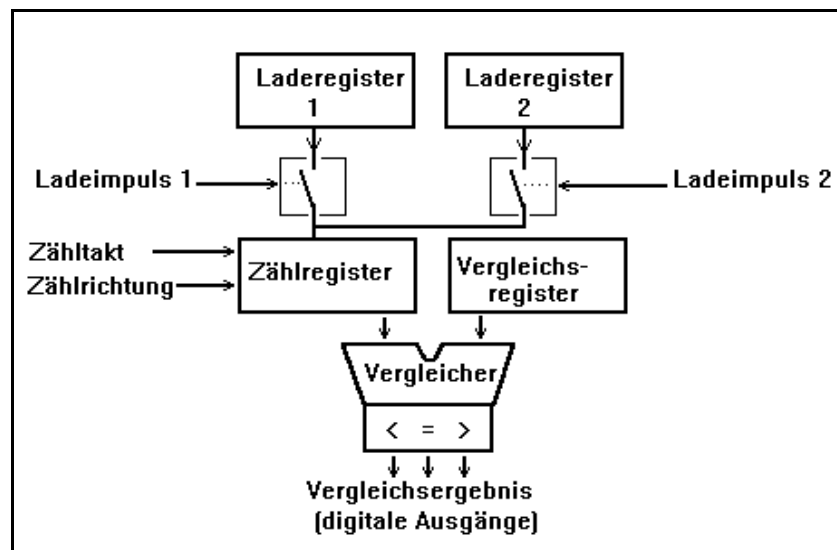


Abb 1.1: Eingangssignale

2. Funktionsweise des Frequenzzählers

Für besondere Anwendungen enthält das Modul CAN-CBM-DIO8-Counter einen per Software implementierten Aufwärts/Abwärts-Zähler, der für einfache Zähl- und Positionieraufgaben verwendet werden kann.

2.1 Eingangssignale

Als Eingangssignale des Zählers werden digitale Eingänge verwendet, die zum Teil konfigurierbar sind. Dies sind:

2.1.1 Ladeimpuls 1

Dieser parametrierbare Eingang dient zum schnellen Kopieren des Inhalts von Laderegister 1 in das Zählregister.

Als Eingangssignal kann der digitale Eingang Nr. 6 oder Nr. 7 verwendet werden.

Das Signal ist zustandsgesteuert, d.h. solange an der entsprechenden Eingangsleitung HIGH-Potential anliegt wird der Wert aus dem Laderegister 'immer wieder' in das Zählregister übertragen (d.h. das Zählregister kann nicht zählen).

2.1.2 Ladeimpuls 2

Für Ladeimpuls 2 gilt entsprechendes wie für Ladeimpuls 1.

Auch hier kann als Quelle der digitale Eingang Nr. 6 oder Nr. 7 verwendet werden.

Mit den beiden Lade-Eingängen können z.B. relative Weg-Meß-Systeme durch absolute 'Wegmarken' (Lichtschranken, Endlagenschalter o.ä.) synchronisiert werden.

2.1.3 Zähltakt

Der Zähltakt ist fest auf den digitalen Eingang Nr. 1 gelegt (weil nur dieser Eingang intern einen CPU-Interrupt auslöst). Wenn der Zähler per Konfiguration aktiv ist, sollte die Eingangsfrequenz maximal 1 kHz betragen, um die CPU durch den zusätzlichen Aufwand nicht zu überfordern. Sollte eine höhere Zählfrequenz nötig sein, richten Sie sich bitte an den Hersteller ! Bei reduzierten Anforderungen an die 'logische Verarbeitung' sind erheblich höhere Zählakte möglich.

2.1.4 Zählrichtung

Bei jeder low->high -Flanke am Zähltakt-Eingang wird der Zustand der Zählrichtung abgefragt, um zu entscheiden ob der Zähler erhöht oder erniedrigt wird. Bei LOW-Pegel (0 V) an diesem Signal wird der Zählerstand erhöht, bei HIGH-Pegel (z.B. 24 V) wird der Zähler um 1 erniedrigt.

Damit lassen sich z.B. Inkrementalgeber abfragen, solange die Frequenz an Eingang 1 unter 1 kHz liegt (unter bestimmten Bedingungen bis 4 kHz).

2.2 Vergleich und Ausgangssignale

Drei Ausgangssignale können vom Vergleich des Impulzzählers auf flexible Art angesteuert werden.

Der aktuelle Zählerstand kann vom Modul ständig mit einem einstellbaren Wert verglichen werden. Bestimmte Ausgangsleitungen können dann in Abhängigkeit vom Vergleichsergebnis ('kleiner, größer, gleich') angesteuert werden.

Der Vergleich liefert intern drei Ergebnisse:

Zählerstand < Vergleichswert
Zählerstand = Vergleichswert
Zählerstand > Vergleichswert

Jeder dieser drei Zustände **kann** zum Ansteuern einer definierbaren Ausgangsleitung dienen. Als Ausgangsleitungen können die Ausgänge Nr. 3, 4 oder 5 verwendet werden. Sie sollten bei der Definition der Datenrichtungen (C10) auch als Ausgang definiert werden, falls sie für den Zählerbetrieb als Ausgang genutzt werden bzw. wenn Änderungen der Komparatorausgänge zum Senden des PDOs führen sollen (siehe Kapitel 4.1).

Die Aktion, die beim Zutreffen oder Nicht-Zutreffen eines Vergleichs ausgeführt werden soll, ist für jeden der drei Vergleich-Ausgänge konfigurierbar:

Ausgang	Vergleich	Aktionscode
Einschalten wenn Vergleich zutrifft	'TRUE'	(0x80)
Ausschalten wenn Vergleich zutrifft	'TRUE'	(0x40)
Einschalten wenn Vergleich nicht zutrifft	'FALSE'	(0x20)
Ausschalten wenn Vergleich nicht zutrifft	'FALSE'	(0x10)

Die in Klammern angegebenen Werte sind die 'Aktionscodes' für die Konfiguration, siehe Kapitel 'Parametrierung des Zählers'.

Die vier Aktionen sind zum Teil kombinierbar.

Ein Ausgang kann durch den Zählervergleich eingeschaltet werden, wenn der Vergleich 'TRUE' liefert. Lieferte der Vergleich 'FALSE' kann der Ausgang abgeschaltet werden.

Der Aktionscode in diesem Beispiel lautet dann: $(0x80) + (0x10) = (0x90)$

Zusätzlich wird noch die Nummer des verwendeten digitalen Ausganges benötigt (3, 4 oder 5).

Durch die flexiblen Definitionsmöglichkeiten kann z.B. der Zähler des E/A-Moduls einen Ausgang einschalten, während das Ausschalten dieses Ausganges per CAN-Bus-Befehl erfolgt. Zeitkritische Aufgaben lassen sich so besser realisieren, als wenn der Vergleich im übergeordneten Steuerungssystem erfolgt.

3. Parametrierung des Zählers

Die Parametrierung des Impulszählers, z.B. Auswahl der verwendeten Signale, erfolgt mit Codestelle C0256. Bis auf den Zählerstand werden alle Einstellungen von C0256 im EEPROM nichtflüchtig gespeichert.

Zum Zugriff auf die einzelnen Komponenten des Zählers dienen die ‘Subindizes’ (CANopen-Terminologie).

Variable	Code-Stelle	Sub-Index	Daten-format	Zugriffs-art	Werte-bereich	Bemerkungen
Impulszählerstand	0256	1	16 Bit	r,w	-32767 ... +32767	Diese Variable enthält den aktuellen Stand des Impulszählers. Sie kann per PDO-Mapping zum Lesen in den Prozeßdatenkanal eingeblendet werden. Einfacher ist allerdings der Zugriff per Service-Daten-Objekt.
Impulszähler-Minimum	0256	2	16 Bit	r,w	-32767 ... +32767	Mit dieser Variablen wird der kleinste zulässige Stand des Impulszählers definiert. Der Impulszählerstand wird auf diesen Wert begrenzt. Es erfolgt niemals ein Überlauf vom Impulszähler-Minimum zum Impulszähler-Maximum. Externe Zählimpulse, die den Zählerstand unter den hiermit definierten Stand bringen würden, werden quasi ignoriert.
Impulszähler-Maximum	0256	3	16 Bit	r,w	-32767 ... +32767	Wie C0256.2, hiermit wird allerdings der höchste zulässige Zählerstand definiert.
Impulszähler-Lade-Wert 1	0256	4	16 Bit	r,w	-32767 ... +32767	Dient zum Zugriff auf Impulszähler-Laderegister 1. Der Inhalt dieser Variablen wird vom Counter-Modul in das Zählregister kopiert, wenn der entsprechend konfigurierte Lade-Eingang 1 ‘aktiven’ Pegel führt.
Impulszähler-Lade-Wert 2	0256	5	16 Bit	r,w	-32767 ... +32767	Wie C0256.1, allerdings für Laderegister 2.
Impulszähler-Vergleichswert	0256	6	16 Bit	r,w	-32767 ... +32767	Diese Variable enthält den Wert, mit dem der Inhalt des Zählregisters ‘ständig’ verglichen wird. Siehe Kapitel 2.2.
Impulszähler-Konfiguration	0256	7	8 Bit	r,w	0x00 ... 0xFF (Bitvektor)	Bedeutung der Bits : Bit 0 (0x01) : gesetzt = Zählermodul aktiviert gelöscht = Zählermodul passiv

Parametrierung des Zählers

Variable	Code-Stelle	Sub-Index	Daten-format	Zugriffs-art	Werte-bereich	Bemerkungen
Impulszähler-Ergebnis-Flags	0256	8	8 Bit	r,w	0x00 ... 0xFF (Bitvektor)	
Impulszähler-Lade-Eingang 1	0256	10	8 Bit	r,w	0... (keine Steuerung) 6... (Kopieren wenn High-Pegel an digitalem Eingang Nr. 6) 7... (Kopieren wenn High-Pegel an digitalem Eingang Nr. 7)	Diese Variable enthält die Definition, bei welchem Ereignis der Impulszähler-Lade-Wert 1 in das Zählerregister kopiert werden soll. Aus Rechenzeit-Gründen kommen dafür nur wenige Quellen in Frage.
Impulszähler-Lade-Eingang 2	0256	11	8 Bit	r,w	0... (keine Steuerung) 6... (Kopieren wenn High-Pegel an digitalem Eingang Nr. 6) 7... (Kopieren wenn High-Pegel an digitalem Eingang Nr. 7)	Diese Variable enthält die Definition, bei welchem Ereignis der Impulszähler-Lade-Wert 2 in das Zählerregister kopiert werden soll. Aus Rechenzeit-Gründen kommen dafür nur wenige Quellen in Frage. Es sollte vermieden werden, Lade-Eingang 1 und Lade-Eingang 2 den gleichen Eingangsklemmen zuzuordnen.
Impulszähler-Vergleicher-Ausgang 'kleiner'	0256	12	8 Bit	r,w	0x00 ... 0xFF (Bitvektor)	Mit dieser Variablen wird die Aktion und Ausgabekanal definiert, die das Counter-Modul ausführen soll wenn der Impulszählerstand kleiner als der Impulszähler-Vergleichswert ist. Erklärung der Bitmaske siehe Kapitel 3.1.1.
Impulszähler-Vergleicher-Ausgang 'gleich'	0256	13	8 Bit	r,w	0x00 ... 0xFF (Bitvektor)	Mit dieser Variablen wird die Aktion definiert, die das Counter-Modul ausführen soll wenn der Impulszählerstand mit dem Impulszähler-Vergleichswert identisch ist. Erklärung der Bitmaske siehe Kapitel 3.1.1.
Impulszähler-Vergleicher-Ausgang 'größer'	0256	14	8 Bit	r,w	0x00 ... 0xFF (Bitvektor)	Mit dieser Variablen wird die Aktion definiert, die das Counter-Modul ausführen soll wenn der Impulszählerstand größer als der Impulszähler-Vergleichswert ist. Erklärung der Bitmaske siehe Kapitel 3.1.1.

Bitmaske

Der Bitvektor enthält die Beschreibung der Aktion in den oberen 4 Bits:

Aktions-Bits für Impulszähler-Vergleicher-Ausgänge:

- 0x80: SETZEN des angegebenen Ausgangs wenn Bedingung erfüllt
- 0x40: RÜCKSETZEN des angegebenen Ausgangs wenn Bedingung erfüllt
- 0x20: SETZEN des angegebenen Ausgangs wenn Bedingung NICHT erfüllt
- 0x10: RÜCKSETZEN des angegebenen Ausgangs wenn Bedingung NICHT erfüllt

Diese Bits lassen sich nur zum Teil kombinieren. So ist es z.B. möglich, daß das Modul Ausgang 5 setzt, wenn die Bedingung (z.B.: 'Zählerstand kleiner als Vergleichswert') erfüllt ist und diesen Ausgang wieder rücksetzt, wenn die Bedingung nicht mehr erfüllt ist. In diesem Fall lautet die komplette Bitmaske $0x80 + 0x10 + 5 = 0x95$.

Nicht kombinierbar sind z.B. die Aktions-Bits $0x80+0x40$ (Widerspruch) oder $0x80+0x20$ (sinnlos). Siehe auch Kapitel 2.2.

Mit den unteren 4 Bits des Bitvektors wird der digitale Ausgang definiert, den der Vergleicher je nach Vergleichsergebnis modifizieren kann:

Digitale Ausgabeleitungen für Impulszähler-Vergleicher-Ausgänge:

- 0x03: digitaler Ausgang Nr. 3
- 0x04: digitaler Ausgang Nr. 4
- 0x05: digitaler Ausgang Nr. 5

Aus Rechenzeit-Gründen kommen nur diese drei Ausgabemöglichkeiten in Frage. Es werden auch maximal drei Ausgabeleitungen benötigt, wenn alle drei Vergleichsergebnisse ('kleiner', 'gleich', 'größer') eigene Aktionen auslösen sollen.

Wenn die Änderung eines Impulszähler-Vergleicher-Ausganges zum Senden eines Prozeßdatenkanals verwendet werden soll, verfahren Sie wie in Kapitel 3.1 beschrieben.

4. Betrieb des Zählers

- Parametrieren Sie den Zähler per SDO-Zugriff wie in Kapitel 2 beschrieben. Definieren Sie -wenn nötig- Aktionen, die beim Erreichen bestimmter Zählerstände vom Modul 'autonom' durchgeführt werden sollen (siehe Kapitel 'Parametrierung des Zählers'/'Impulszähler-Vergleicher-Ausgang 'Kleiner'')
- Aktivieren Sie den Zähler, indem Sie das Aktivierungsbit setzen
- (C256.7, Bit 0 setzen)
- Lesen Sie den Zählerstand per SDO aus C256.1 aus
oder
- Blenden Sie -falls erforderlich- das Zählregister in Ihre Prozeßdaten ein (siehe PDO-Mapping, leider reichlich kompliziert...)
- Wenn nötig, aktivieren Sie die Prozeßdatenübertragung indem Sie das Modul per NMT in den Zustand 'operational' schalten.

In den folgenden Unterkapiteln haben wir einige Anwendungen als Beispiel für den Einsatz des Impulszählers aufgeführt.

4.1 Senden eines Prozeßdaten-Telegramms mit Triggerung durch den Impulszähler

Dieses Kapitel beschreibt, wie Sie das Modul konfigurieren müssen, wenn sie das Senden eines Prozeßdaten-Telegramms durch den Impulszähler 'triggern' wollen.

Dazu müssen Sie zunächst den Vergleich der Impulszähler so konfigurieren, daß der Vergleich einen oder mehrere digitale Ausgänge des Counter-Moduls ansteuern kann. Dies wird in Kapitel 2.2 beschrieben. Zum Beispiel setzen Sie C256.12 auf den Wert 0x93, damit der digitale Ausgang Nr.3 entsprechend des Vergleichs-Ergebnisses 'Zählerstand < Vergleichswert' angesteuert wird. (Dazu sollten Sie I/O 3 mit C10 auch als Ausgang definieren).

Da jeder digitaler Ausgang auch als digitaler Eingang 'zurückgelesen' wird, können sie den Vergleich-Ausgang als 'Trigger' zum Senden eines Prozeßdatentelegramms machen, indem Sie einfach die 'Interrupt-Maske' der digitalen Eingänge entsprechend setzen.

Zum Beispiel setzen wir C12 auf den Wert 0x1C (hexadezimal). Damit werden die Klemmen 'I/O3', 'I/O4' und 'I/O5' als 'Interrupteingänge' definiert¹. Diese drei Klemmen können vom Impulszähler-Vergleich angesteuert werden. 'I/O3' gibt in unserem Beispiel das Vergleichsergebnis 'Zählerstand < Vergleichswert' aus (das wurde im letzten Absatz so konfiguriert). Die beiden niederwertigen Bits in C12 lassen wir absichtlich auf '0', weil die Klemmen 'I/O1' und 'I/O2' die Zähler-Eingänge (Takt und Richtung) sind, die sich im Betrieb ständig ändern werden und zu einer extremen Buslast führen könnten.

Damit die Prozeßdaten 'ereignisgesteuert' gesendet werden können, muß außerdem noch der 'PDO Transmission Type' (Teil aus 'Transmit PDO Communication Parameter', Objekt 1800H, Subindex 2, siehe CiA Draft Standard 301, Version 3.0) auf '255' (dezimal) gesetzt werden. Dies bedeutet: 'Die Aussendung des PDOs wird durch ein im Device Profile (hier: DS401) definiertes Ereignis getriggert'.

Anschließend können die so eingestellten Parameter dauerhaft im EEPROM abgespeichert werden (falls nicht ohnehin 'autonom' gespeichert wird, siehe Systemhandbuch). Im Zweifelsfall schreiben Sie die Zeichenkette 'save' in Objekt 1010H, Subindex 1 um die Daten dauerhaft zu sichern.

¹ Die Bezeichnung 'Interrupteingang' stammt aus der CANopen-Terminologie. Ein als 'Interrupteingang' definierter digitaler Eingang kann bei Änderung des Pegels das Senden eines Prozeßdaten-Telegramms (PDO) bewirken. Im CAN-CBM-DIO8 diente dazu zunächst Codestelle C12, später kam aus Kompatibilitätsgründen mit CiA Draft Standard Proposal 401 (kurz: DS401) noch das Objekt 6006H dazu. C12 enthält 'intern' die gleichen Daten wie Objekt 6006H. Die Polarität der Flanke spielt dabei keine Rolle: ein Low->High - Übergang löst ebenso wie ein High->Low - Übergang einen 'Interrupt' aus.

Die so konfigurierte ‘impulszählergesteuerte PDO-Sendung’ funktioniert folgendermaßen:

1. Beim Unterschreiten des in C256.6 definierten Vergleichswertes setzt der Vergleich der digitalen Ausgang ‘I/O3’.
2. Die Routine zum Abtasten der digitalen Eingänge erkennt die Flanke an ‘I/O3’ und setzt das Sende-Flag für die Prozeßdaten, weil ‘I/O3’ als ‘Interrupteingang’ definiert wurde.
3. Die Prozeßdaten-Sende-Routine stellt ein aktuelles Prozeßdaten-Telegramm zusammen, in dem möglicherweise auch der ‘aktuelle’ Inhalt des Zählregisters enthalten ist. Durch die Verzögerung beim ‘Rücklesen’ der digitalen Eingänge entsteht eine Verzögerung im Millisekunden-Bereich, die bei schnellem Zähltakt dazu führen kann, daß der im PDO eingetragene Zählerstand schon einige Schritte ‘weitergelaufen’ ist.

5. Anhang: Code-Stellen des Zählermoduls

Code-nr.	Inhalt	Datentyp, [Einheit]	Zugriff r = read w = write	Einstell- / Anzeigemöglichkeiten	Werkeinstellung	Bemerkungen, Verweise
256.1	Impulszählerstand	16 Bit	r,w	-32767 ... +32767, max. 1 kHz Zählrate (an dig. Input 1)		in PDO1 einblendbar
256.2	Impulszähler- Minimum	16 Bit	r,w	-32767 ... +32767		
256.3	Impulszähler- Maximum	16 Bit	r, w	-32767 ... +32767		
256.4	Impulszähler- Lade-Wert 1	16 Bit	r,w	-32767 ... +32767		
256.5	Impulszähler- Lade-Wert 2	16 Bit	r,w	-32767 ... +32767		
256.6	Impulszähler- Vergleichswert	16 Bit	r,w	-32767 ... +32767		
256.7	Impulszähler- Konfiguration	8 Bit	r,w	0x00 .. 0xFF (Bitvektor)	0	
256.8	Impulszähler- Ergebnis-Flags	8 Bit	r,w	0x00 .. 0xFF (Bitvektor)	-	
256. 10	Impulszähler- Lade-Eingang 1	8 Bit	r, w	0 (keine Steuerung) 3..8 (Klemmen-Nr.)	0	Signal-Konfig für Lade-Wert 1
256. 11	Impulszähler- Lade-Eingang 2	8 Bit	r, w	0 (keine Steuerung) 3..8 (Klemmen-Nr.)	0	Signal-Konfig. für Lade-Wert 2
256. 12	Impulszähler- Ausg. 'kleiner'	8 Bit	r, w	0 (kein Ausgang) 3..8 (Klemmen-Nr.)	0	Signal-Konfig. 'Zähler<Vergl.'
256. 13	Impulszähler- Ausg. 'gleich'	8 Bit	r, w	0 (kein Ausgang) 3..8 (Klemmen-Nr.)	0	Signal-Konfig. 'Zähler=Vergl.'
256. 14	Impulszähler- Ausg. 'größer'	8 Bit	r, w	0 (kein Ausgang) 3..8 (Klemmen-Nr.)	0	Signal-Konfig. 'Zähler>Vergl.'