

ADwin-CO2

Zähleroption

für ***ADwin***-Meßwerterfassungskarten

Version 1.4

März 1998

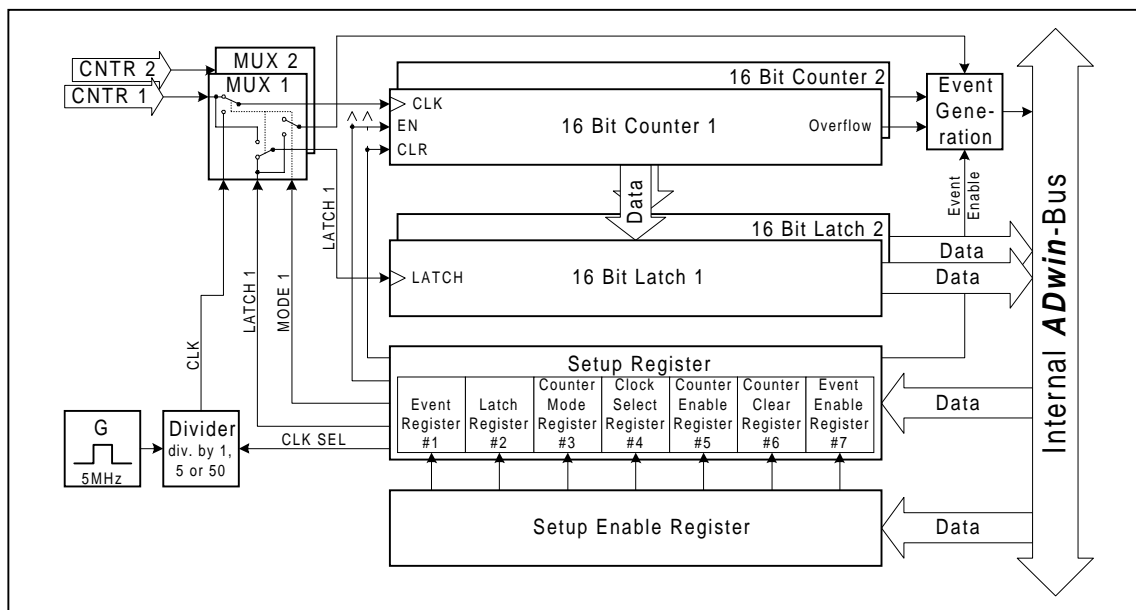
Inhaltsverzeichnis

| | |
|--|---|
| 1. ADwin -Karte mit der Zähleroption ADwin-CO2 | 3 |
| 1.1 Betriebsarten | 3 |
| 1.1.1 Impulszähler | 3 |
| 1.1.2 Periodenbreitenmessung..... | 4 |
| 1.2 Steckerbelegungen | 4 |
| 1.3 Technische Daten der Zählereingänge | 4 |
| 1.4 Steuerung der Zähler | 5 |
| 2. ADbasic -Beispielprogramme | 7 |
| 2.1 Die ADwin -Karte mit ADwin-CO2 -Zähleroption in der Betriebsart Impulszähler | 7 |
| 2.1.1 Programmcode..... | 7 |
| 2.2 Die ADwin -Karte mit ADwin-CO2 -Zähleroption in der Betriebsart Periodenbreitenzähler | 8 |
| 2.2.1 Programmcode..... | 8 |

1. ADwin-Karte mit der Zähleroption ADwin-CO2

Die **ADwin**-Karte mit der Zähleroption **ADwin-CO2** enthält zwei 16-Bit-Zähler mit je einem Zwischenregister (Latch). Mit Hilfe der Zwischenregister können die Zähler auch während des Zählens ausgelesen werden. Die Zählereingänge liegen auf der Sub-D-Buchse an der Kartenrückseite und sind mit 5 Volt TTL-Signalen anzusteuern.

Die beiden Zähler sind bis auf die Registeradressen und Steuerbits völlig identisch. Sie können wahlweise als Impulszähler oder zur Periodenbreitenmessung völlig unabhängig voneinander (auch in verschiedenen Betriebsarten) benutzt werden; d. h. jeder Zähler kann für sich gestartet, gestoppt, gelöscht und ausgelesen werden.



Hinweis: Zugunsten der Übersichtlichkeit sind viele Verbindungsleitungen (MUX 2 und 16 Bit Latch 2) von Zähler 2 nicht dargestellt.

1.1 Betriebsarten

1.1.1 Impulszähler

In der Betriebsart Impulszähler wird jede ansteigende Flanke am Zählereingang gezählt. Mit einem Latch-Befehl wird der Zählerstand in das Latch übernommen. Der Zählerstand kann danach aus dem Latch ausgelesen werden. Jeder Zähler besitzt ein eigenes Latch, so daß beide Zählerstände exakt gleichzeitig erfaßt werden können. Wenn der Zähler überläuft, kann ein Event für den Transputer erzeugt werden.

1.1.2 Periodenbreitenmessung

In der Betriebsart Periodenbreitenmessung wird bei jeder ansteigenden Flanke auf dem Zählereingang der aktuelle Zählerstand in das Latch übernommen. Der Zähler wird dabei intern mit einer Taktfrequenz von wahlweise 5 MHz, 1 MHz oder 100 kHz getaktet. Bei jedem Latch-Impuls kann ein Event für den Transputer erzeugt werden. Bei jedem Zählerüberlauf kann ebenfalls ein Event erzeugt werden.

1.2 Steckerbelegungen

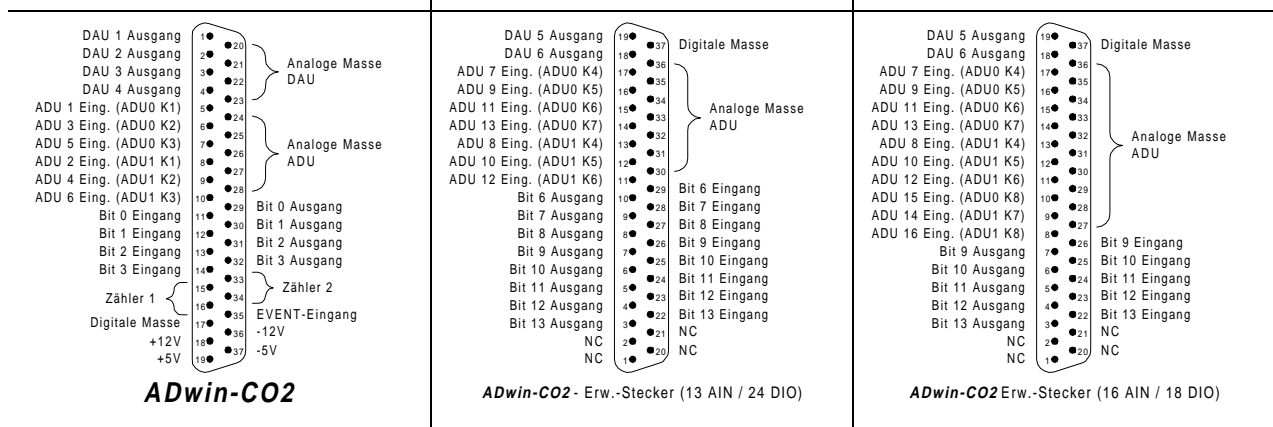
Wie bei der Standard-**ADwin**-Karte erhalten Sie auch mit der **ADwin**-Karte mit der Zähleroption **ADwin-CO2** zwei Erweiterungsstecker. Mit diesen Erweiterungssteckern steht Ihnen eine größere Anzahl analoger Eingänge (AIN) und digitaler I/Os (DIO) zur Verfügung.

Durch die geänderte Ausnutzung der ASICs gegenüber der Standardversion der **ADwin**-Karte ergeben sich folgende Abweichungen an den Steckerbelegungen:

Digital I/O 4 und 5 wurden durch die Zählereingänge 1 und 2 ersetzt.

Digital I/O 14 und 15 entfallen. Mit diesem Erweiterungsstecker stehen Ihnen insgesamt 13 analoge Eingänge und 24 digitale IOs zur Verfügung.

Digital I/O 14 und 15 entfallen. Mit diesem Erweiterungsstecker stehen Ihnen insgesamt 16 analoge Eingänge und 18 digitale IOs zur Verfügung.



1.3 Technische Daten der Zählereingänge

| | |
|---------------------|--------------|
| Typ: | TTL |
| Eingangswiderstand: | 5,6 kΩ |
| V _{IH} : | 2,4 V min. |
| V _{IL} : | 0.8 V max. |
| Spannungsbereich: | -0,3 bis 7 V |
| I _{IH} : | 1 mA max. |
| I _{IL} : | 0,2 mA max. |

1.4 Steuerung der Zähler

Der Zähler kann mit Hilfe von sieben internen Registern konfiguriert werden. Diese Register liegen alle auf der gleichen I/O-Adresse 0xC0 (**CO_REG**). Das gewünschte Register ist vor dem Zugriff mit dem Setup-Enable-Register mit der Adresse 0xD0 (**CO_MODE**) auszuwählen. Die Zählerstände können aus dem Datenregister mit der Adresse 0xB0 (**CO_RD**) ausgelesen werden.

Register #1 ist das Event-Register.

Hier kann nach einem Event ausgelesen werden, wodurch der Event ausgelöst wurde. Events können nur ausgelöst werden, wenn der Verursacher vorher im Event-Enable-Register (Register #7) enabled wurde. Das Event-Register muß einmal im Init-Block Ihres **ADbasic**-Programms zurückgesetzt werden. Anschließend wird es beim Auslesen automatisch zurückgesetzt.

Ein Event auf dem externen Eventeingang wird im Event-Register nicht angezeigt.

| | |
|---------|--|
| Bit 0=1 | Zählerstand von Zähler 1 in das Latch übernommen |
| Bit 1=1 | Zählerstand von Zähler 2 in das Latch übernommen |
| Bit 2=1 | Overflow Zähler 1 |
| Bit 3=1 | Overflow Zähler 2 |

Hinweis: Der Overflow-Event wird schon beim vorletzten Bit (0x8000) ausgelöst.

Hinweis: Falls mehrere Ursachen eintraten, bevor das Event-Register ausgelesen wurde, können auch mehrere Bits gleichzeitig gesetzt sein.

Register #2 ist das Latch-Register.

Durch Schreiben in Register #2 wird der aktuelle Zählerstand in das Latch übernommen.

| | |
|---------------------|---|
| Bit 0=1 | Zähler 1 wird übernommen |
| Bit 1=1 | Zähler 2 wird übernommen |
| Bit 0=1 und Bit 1=1 | Beide Zähler werden gleichzeitig übernommen |

Register #3 ist das Counter-Mode-Register.

Mit dem Register #3 kann man die Betriebsart (Impuls- oder Periodenbreitenmessung) wählen.

| | | |
|-------|--------------------------|---|
| Bit 0 | Betriebsart für Zähler 1 | (0 = Impulsmessung, 1 = Periodenbreitenmessung) |
| Bit 1 | Betriebsart für Zähler 2 | (0 = Impulsmessung, 1 = Periodenbreitenmessung) |

Register #4 ist das Clock-Select-Register.

Mit dem Register #4 wird die Taktfrequenz für die Periodenbreitenmessung festgelegt.

0 = 5 MHz

1 = 1 MHz

2 = 100 kHz

Register #5 ist das Counter-Enable-Register.

Mit dem Register #5 werden die Zähler gestartet und gestoppt.

| | |
|-------|---|
| Bit 0 | Enable Zähler 1 (0 = gestoppt, 1 = gestartet) |
| Bit 1 | Enable Zähler 2 (0 = gestoppt, 1 = gestartet) |

Register #6 ist das Counter-Clear-Register.

Durch Schreiben in Register #6 werden die Zähler gelöscht.

| | |
|-------------------------|---|
| Bit 0 = 1 | Zähler 1 wird gelöscht |
| Bit 1 = 1 | Zähler 2 wird gelöscht |
| Bit 0 = 1 und Bit 1 = 1 | Beide Zähler werden gleichzeitig gelöscht |

Register #7 ist das Event-Enable-Register.

Hier können Sie festlegen welche Ereignisse einen Event auslösen sollen.

| | |
|-------|---|
| Bit 0 | Latch-Befehl für Zähler 1 (1 = enable, 0 = disable) |
| Bit 1 | Latch-Befehl für Zähler 2 (1 = enable, 0 = disable) |
| Bit 2 | Overflow Zähler 1 (1 = enable, 0 = disable) |
| Bit 3 | Overflow Zähler 2 (1 = enable, 0 = disable) |
| Bit 4 | externer Event Eingang (1 = enable, 0 = disable) |

Um die Latches auslesen zu können, muß zuerst im Setup-Enable-Register (Adresse 0xD0) die Nummer des gewünschten Zählers (1 oder 2) angegeben werden. Danach kann der Zählerstand aus dem Datenregister (Adresse 0xB0) gelesen werden.

2. ADbasic-Beispielprogramme

2.1 Die ADwin-Karte mit ADwin-CO2-Zähleroption in der Betriebsart Impulszähler

Das beschriebene Programm (co2_test1.bas) befindet sich auf der Diskette mit der Bezeichnung „Beispielprogramme“. Mit dem Programm co2_test1.bas können Sie alle 100 Millisekunden die beiden Zähler einer **ADwin**-Karte mit **ADwin-CO2**-Zähleroption auslesen und die Zählerstände und die Frequenz in Parameter schreiben.

2.1.1 Programmcode

```
REM PAR_1 = Zählerstand von Zähler 1
REM PAR_2 = Zählerstand von Zähler 2
REM PAR_3 = Frequenz von Zähler 1
REM PAR_4 = Frequenz von Zähler 2

DIM CO_REG, CO_MODE, CO_RD as integer
DIM f1, f2, co1a, co2a, ovl as integer

INIT:
REM Registeradressen definieren
CO_REG = 0C0h      ' Registeradresse für I/O-Register
CO_MODE = 0D0h     ' Registeradresse für das Setup-Enable-Register
CO_RD   = 0B0h     ' Registeradresse für das Datenregister

POKE(CO_MODE,1)    ' Event Register auswählen und damit zurücksetzen
ovl = PEEK(CO_REG)

POKE(CO_MODE,7)    ' Event-Enable-Register auswählen
POKE(CO_REG,0)     ' Events deaktivieren

POKE(CO_MODE,3)    ' Counter-Mode-Register auswählen
POKE(CO_REG,0)     ' Betriebsart Zähler einstellen

POKE(CO_MODE,6)    ' Counter-Clear-Register auswählen
POKE(CO_REG,3)     ' beide Zähler löschen

POKE(CO_MODE,5)    ' Counter-Enable-Register auswählen
POKE(CO_REG,3)     ' beide Zähler starten

co1a = 0           ' alten Zählerstand löschen
co2a = 0           ' alten Zählerstand löschen

EVENT:
POKE(CO_MODE,2)    ' Latch-Register auswählen
POKE(CO_REG,3)     ' beide Zählerstände latchen

POKE(CO_MODE,1)    ' Zähler 1 lesen selektieren
PAR_1 = PEEK(CO_RD)

POKE(CO_MODE,2)    ' Zähler 2 lesen selektieren
PAR_2 = PEEK(CO_RD)
```

```
f1 = (PAR_1-co1a)           ' Frequenz 1 berechnen
f2 = (PAR_2-co2a)           ' Frequenz 2 berechnen
IF (f1<0) THEN f1 = f1 + 10000h ' falls der Zaehler gerade
IF (f2<0) THEN f2 = f2 + 10000h ' uebergelaufen ist
PAR_3 = f1 * 10
PAR_4 = f2 * 10
PAR_1 = co1a
PAR_2 = co2a
```

2.2 Die ADwin-Karte mit ADwin-CO2-Zähleroption in der Betriebsart Periodenbreitenzähler

Das beschriebene Programm (co2_test2.bas) befindet sich auf der Diskette mit der Bezeichnung „Beispielprogramme“. Mit dem Programm co2_test2.bas werden bei jedem Event die beiden Zähler einer **ADwin**-Karte mit **ADwin-CO2**-Zähleroption ausgelesen. Falls das Event von einer steigenden Flanke am Zählereingang erzeugt wurde, dann wird der Zählerstand ausgelesen und die Differenz zum letzten Auslesen berechnet. Bei einem Zählerüberlauf wird der High-Word-Zähler incrementiert.

2.2.1 Programmcode

```
REM PAR_1  = Zähler 1
REM PAR_2  = Zähler 2
REM PAR_3  = Pulsabstand 1 in Mikrosekunden
REM PAR_4  = Pulsabstand 2 in Mikrosekunden

DIM CO_REG, CO_MODE, CO_RD as integer
DIM col_H, co2_H, co1a, co2a, ovl, ih as integer

INIT:
CO_REG = 0C0h           ' Registeradressen definieren
CO_MODE = 0D0H
CO_RD = 0B0H

POKE(CO_MODE,7)         ' Überlaufbit enablen
POKE(CO_REG,0001111b)
POKE(CO_MODE,3)         ' Betriebsart Periodenbreitenmessung einstellen
POKE(CO_REG,3)
POKE(CO_MODE,4)         ' Taktfrequenz einstellen
POKE(CO_REG,0)          ' 1 = 1 MHz ( 0 = 5 MHz , 2 = 100 kHz)
POKE(CO_MODE,6)         ' die Zähler löschen
POKE(CO_REG,3)
POKE(CO_MODE,5)         ' beide Zähler starten
POKE(CO_REG,3)
POKE(CO_MODE,1)         ' Überlaufregister zurücksetzen
ovl = PEEK(CO_REG)
col_H = 0               ' High-Word-Zähler 1 löschen
co2_H = 0               ' High-Word-Zähler 2 löschen
co1a = 0                ' alten Zählerstand löschen
co2a = 0
```



```
EVENT:
POKE(CO_MODE,1)           ' Eventregister auslesen
ovl = PEEK(CO_REG)
DO
IF ((ovl AND 1) = 1) THEN   ' Falls Flanke an Eingang 1
    POKE(CO_MODE,1)        ' Zähler 1 lesen selektieren
    PAR_1 = (PEEK(CO_RD)AND 07ffffh) + col_h    ' Zähler 1 lesen
    PAR_3 = (PAR_1-col_a)    ' Pulsbreite berechnen
    col_a = PAR_1
ENDIF
IF ((ovl AND 2) = 2) THEN   ' Falls Flanke an Eingang 2
    POKE(CO_MODE,2)        ' Zähler 2 lesen selektieren
    PAR_2 = (PEEK(CO_RD)AND 07ffffh) + co2_h    ' Zähler 2 lesen
    PAR_4 = (PAR_2-co2_a)    ' Pulsbreite berechnen
    co2_a = PAR_2
ENDIF

IF ((ovl AND 4) = 4) THEN
    col_h = col_h + 32768
ENDIF

IF ((ovl AND 8) = 8) THEN
    co2_h = co2_h + 32768
ENDIF

POKE(CO_MODE,1)
ovl = PEEK(CO_REG)
UNTIL (ovl = 0)    ' falls inzwischen noch ein Event kam wiederholen
```