

# Virtuell Encoder

## Betriebsanleitung @P5190

Ausgabe-/Rev.-Datum:	14.06.2007
Dokument-Revision:	00
Firmware-Version:	V1
Dateiname:	@P5190-TRS-V-BA-D-0000-00.doc
Artikel.-Nr.:	TRS-DOC-000
Verfasser:	Hilz

TR-Systemtechnik GmbH  
Eglishalde 16  
D-78647 Trossingen

Tel. 07425 / 228-0  
Fax 07425 / 228-34

## Impressum

TR-Systemtechnik GmbH  
D-78647 Trossingen  
Eglisshalde 16  
Tel.: (++49) 07425/228-0  
Fax: (++49) 07425/228-34  
[info@tr-systemtechnik.de](mailto:info@tr-systemtechnik.de)  
<http://www.tr-systemtechnik.de/>

© Copyright 2003 TR-Systemtechnik

## Änderungsvorbehalt

Änderungen der in diesem Dokument enthaltenen Informationen, die aus unserem stetigen Bestreben zur Verbesserung unserer Produkte resultieren, behalten wir uns jederzeit vor.

## Druck

Dieses Handbuch wurde mit MS-WORD für Windows auf einem Personal-Computer erstellt. Der Text wurde in Arial gedruckt.

## Schreibweisen

Kursive oder fette Schreibweise steht für den Titel eines Dokuments oder wird zur Hervorhebung benutzt.

Courier-Schrift zeigt Text an, der auf dem Bildschirm / Display sichtbar ist und Menü auswählen von Software.

" < > " weist auf Tasten der Tastatur Ihres Computers hin (wie etwa <RETURN>).

## Hinweis



Meldungen, die nach dem Symbol "Hinweis" erscheinen, markieren wichtige Merkmale des verwendeten Produkts.

## Hinweise zu Urheberrechten (Copyright ©)

MS-WORD ist ein eingetragenes Warenzeichen der Microsoft AG.

## Literatur

## Änderungs-Index

### Hinweis

Auf dem Deckblatt dieses Dokumentes ist der aktuelle Revisionsstand mit dem dazugehörigen Datum vermerkt.

Zeichnungen, die sich im Anhang befinden können, sind mit einem eigenen Änderungs-Index versehen.

Dokumenterstellung:

14.06.2007

Rev	Änderung	Datum

## Inhaltsverzeichnis

Impressum .....	2
Änderungs-Index .....	3
Inhaltsverzeichnis .....	4
<b>1. Allgemeines zum Modul .....</b>	<b>5</b>
<b>2. Definitionen .....</b>	<b>5</b>
<b>3. Betriebsarten .....</b>	<b>6</b>
<b>4. Parameter- / Statusübertragung .....</b>	<b>7</b>
<b>5. Master-Mode .....</b>	<b>8</b>
5.1. Datenübertragung .....	8
5.1.1 Data_Control .....	8
5.2. Ist-Position.....	9
5.2.1 Presetwert setzen .....	9
5.3. Steuerung Multiplexed Data .....	9
5.3.1 Ist-Frequenz (Multiplexed Data).....	10
5.3.2 Maximal-Frequenz (Multiplexed Data).....	10
5.3.3 Beschleunigungs-Frequenz (Multiplexed Data).....	11
5.3.4 Verzögerungs-Frequenz (Multiplexed Data).....	11
5.3.5 Berechnung der Ausgabefrequenz .....	13
5.3.6 Stopp-Position (Multiplexed Data) .....	14
<b>6. SLAVE-Mode.....</b>	<b>15</b>
6.1. Presetwert laden .....	15
<b>A @ctiveIO-Systembus Kommunikation.....</b>	<b>16</b>
A.1 Setzen des Virtual-Encoder (über Systembus).....	17
A.2 Auswerten der Ist-Position .....	18
A.2.1 Presetwert .....	18
A.3 Stopp-Position.....	18

## 1. Allgemeines zum Modul

Das von TR-Systemtechnik **@P5190** Modul ist ein **Virtual-Encoder**. Das Modul kann, je nach Betriebsart, Inkrement-Impulse ausgeben (Master) oder einlesen (Slave). In der Betriebsart "Master" sind verschiedene Einstellungen (z.B.: Ausgabe-, Beschleunigungs- und Verzögerungs-Frequenz) für die Ausgabe der Inkrement-Impulse einstellbar. Zum Modul können folgende Funktionen eingestellt werden:

- Vordefinieren einer **Stopp-Position**
- Laden eines Wertes für die **Ist-Position**

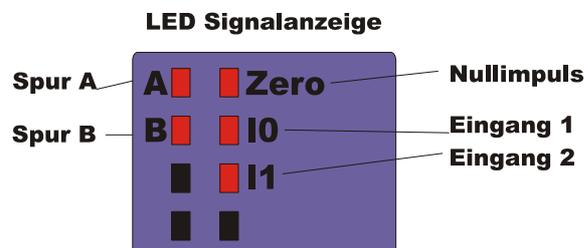
Als zusätzliche Anwendungsmöglichkeit kann das Modul **@P5191 Virtual-Repeater** zu dem Virtual-Encoder eingesetzt werden. Dieses Modul stellt einen Signalverteiler in Verbindung zum Virtual-Encoder dar. Das heißt, es wird direkt an den Virtual-Encoder angesteckt und gibt die Daten dieses weiter. Hierbei ist es unabhängig welche Betriebsart im Virtual-Encoder eingestellt wird. Ist der Virtual-Encoder im Masterbetrieb eingestellt, so werden die generierten Impulse ausgegeben. Im Slavebetrieb werden die eingelesenen Impulse eines Inkremental-Encoders gezählt und weitergeleitet.



Abbildung 1: @P5190 Virtual-Encoder

### Anzeigen

Im Sichtfenster auf der Front des Moduls befinden sich LEDs, die folgende Signale anzeigen:



## 2. Definitionen

Hexadezimalzahlen werden mit **0x** dargestellt. (0xFF hex = 255 dezimal).

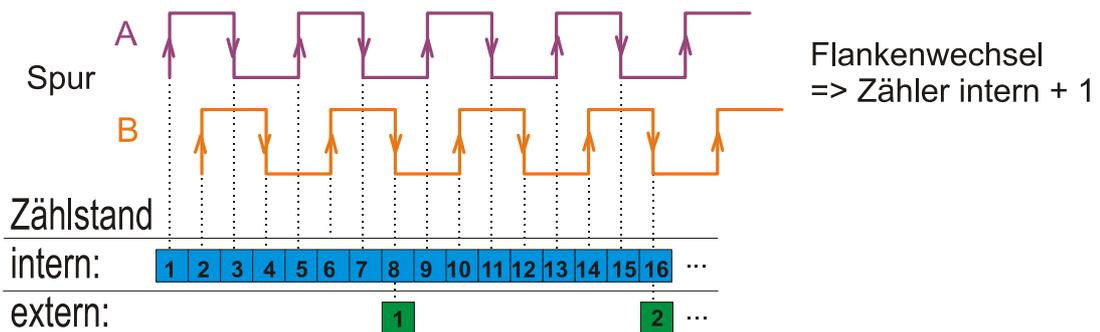
Der Virtual-Encoder belegt **48 Bit** also **6 Byte** oder **3 Word** im Prozessabbild. Zur Kommunikation mit dem Virtual-Encoder sind **Wort0 / Wort1** und **Wort2** definiert. Je nach Anwendung **Parameter- / Statusübertragung** oder **Datenübertragung** beinhalten diese Worte unterschiedliche Funktionen.

### 3. Betriebsarten

Das Modul kann in 2 verschiedenen Betriebsarten betrieben werden. Der Virtual-Encoder befindet sich Standardmäßig immer im Slave-Mode.

**Funktionsprinzip:**

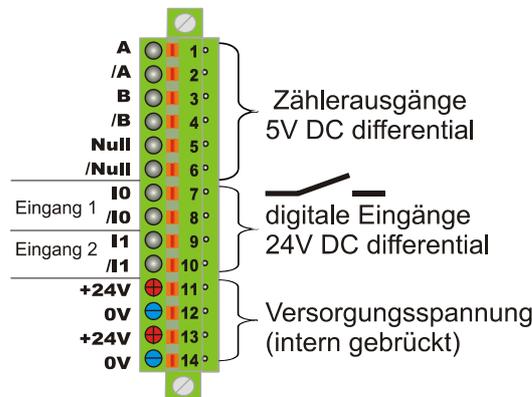
Je nach Betriebsart werden Inkrement-Impulse mit 4-fach Auswertung eingelesen/generiert. Bei jedem Flankenwechsel wird der interne Zähler (18Bit Zähler), je nach Drehrichtung, um den Wert 1 inkrementiert bzw. dekrementiert. Der tatsächlich angezeigte Zählerwert beträgt = interner Zählerwert / 8. Das heißt, die Bits 0 bis 2 werden abgeschnitten, nur die Bits 3 bis 18 des internen Zählers werden für die Anzeige verwendet. Somit ergibt sich das nur jede achte Flanke die Anzeige verändert. Der externe Zählstand wird als **16Bit** Inkrementalzähler angezeigt.



#### 1. Master-Mode

Virtual-Encoder generiert Inkremental-Impulse, welche mit den gewünschten Frequenzen eingestellt werden können.

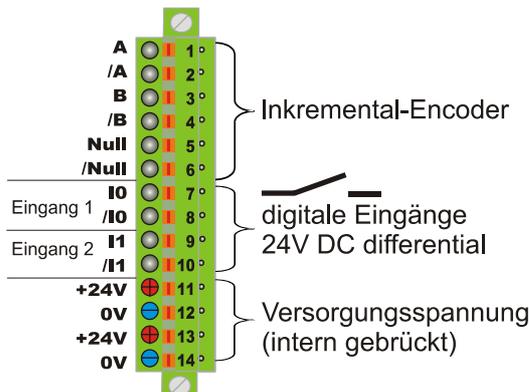
**Anschluß:**



#### 2. Slave-Mode

Virtual-Encoder arbeitet als **16Bit Inkrementalzähler**.

**Anschluß:**



## 4. Parameter- / Statusübertragung

	Adresse Receive-Page (Offset <b>0x0100</b> )	Adresse Send-Page (Offset <b>0x0000</b> )
Wort0:	<b>Param_Control</b>	<b>Param_Control</b>
Wort1:	keine Funktion	keine Funktion
Wort2:	keine Funktion	keine Funktion

Tabelle 1: Receive- / Sendpage bei Parameter / Statusübertragung

### Param Control:

Das Kontrollwort **Data\_Control** hat eine Größe von 16 Bit. Mit diesem Datenwort (Wort2) werden alle Vorgaben für den Virtual-Encoder übertragen.

16-Bit Kontrollwort

Bit	Funktion
<b>0</b>	Write   <b>Write Enable</b> sämtliche Werte der Sendepage werden nur übernommen wenn <b>Bit0 = 1</b>
	Read   Firmware-Version <b>Bit0</b>
<b>1</b>	Read   Firmware-Version <b>Bit1</b>
<b>2</b>	Read   Firmware-Version <b>Bit2</b>
<b>3</b>	Read   Firmware-Version <b>Bit3</b>
<b>4</b>	n.u.
<b>5</b>	n.u.
<b>6</b>	n.u.
<b>7</b>	n.u.
<b>8</b>	n.u.
<b>9</b>	n.u.
<b>10</b>	n.u.
<b>11</b>	Write   Betriebsart ( <b>Vorsicht geändert</b> ) <b>MASTER</b> → <b>Bit11 = 1</b> <b>SLAVE</b> (Inkrementalzähler) → <b>Bit11 = 0</b>
<b>12</b>	n.u.
<b>13</b>	n.u.
<b>14</b>	n.u.
<b>15</b>	n.u.

Tabelle 2: Parameter der Parameter- / Statusübertragung

## 5. Master-Mode

### 5.1. Datenübertragung

#### 5.1.1 Data\_Control

Das Kontrollwort **Data\_Control** hat eine Größe von 16 Bit. Mit diesem Kontrollwort (Wort2) werden alle Vorgaben für den Virtual-Encoder übertragen. Die Bits **8, 9** und **10** steuern den **Multiplexed Data** Bereich (siehe Kapitel 5.3), welcher Einstellungen zum Virtual-Encoder im Master-Betrieb beinhaltet.

	Adresse Receive-Page (Offset <b>0x0100</b> )	Adresse Send-Page (Offset <b>0x0000</b> )
Wort0:	<b>0x0100</b> Ist-Position	<b>0x0000</b> <i>berichtigt: laden Ist-Position</i>
Wort1:	<b>0x0102</b> Multiplexed Data	<b>0x0002</b> Multiplexed Data
Wort2:	<b>0x0104</b> Data_Control	<b>0x0004</b> Data_Control

Tabelle 3: Adressen für Datenübertragung Master-Mode

Die untere Tabelle beschreibt die Funktionen im **Data\_Control** (Wort2).

Bit	Funktion
<b>0</b>	Hardware Eingang <b>I0</b>
<b>1</b>	Hardware Eingang <b>I1</b>
<b>2</b>	Hardware Eingang <b>Nullpuls</b> von <b>Inkrementalzähler</b>
<b>3</b>	n.u.
<b>4</b>	n.u.
<b>5</b>	n.u.
<b>6</b>	n.u.
<b>7</b>	n.u.
<b>8</b>	<b>MuxCtrl 0</b> (siehe Kapitel 5.3)
<b>9</b>	<b>MuxCtrl 1</b> (siehe Kapitel 5.3)
<b>10</b>	<b>MuxCtrl 2</b> (siehe Kapitel 5.3)
<b>11</b>	Preset_CNT (Inkrementalzähler laden)
<b>12</b>	<b>RUN</b>
<b>13</b>	<b>STOP</b>
<b>14</b>	Zählfolge => rückwärts <b>Bit14 = 1</b> => vorwärts <b>Bit 14 = 0</b>
<b>15</b>	<b>Write_Enable</b> sämtliche Werte der Sendepage werden nur übernommen wenn <b>Bit15 = 1</b>

Tabelle 4: Funktionen im **Data\_Control**

Um die Funktionen / Einstellungen zu setzen, muss im Schreibtelegramm das Schreibbit "Write\_Enable" und das entsprechende Bit für die Funktion gesetzt werden.

**Beispiel:** Soll der Virtual\_Encoder im Master-Mode starten, so muss das Schreibbit **Bit15** + das **Bit12** für "RUN" Virtual-Encoder auf **1** gesetzt werden (Wert: **0x9000**). Somit muss der Wert **0x9000** im **Data\_Control** an das Modul übertragen werden.

## 5.2. Ist-Position

16-Bit Wert des Inkrementalzählers Wort0 (Entspricht dem tatsächlichen Inkrementalzahlstand / 8).  
**Ist-Position lesen:**

**Empfangspage:**

Wort0: Ist-Position	Wort1: Multiplexed Data	Wort2: Data_Control
0xXXXX	0xXXXX	0x0000

Erklärung:

- Der Lesezugriff auf das Wort0 liefert die aktuelle Ist- Position.

### 5.2.1 Presetwert setzen

Ein vordefinierter Positionswert kann durch durch das Bit11 (Preset\_CNT) geladen werden. Bleibt das Bit11 bei mehreren Übertragungen anstehen, so wird der Zähler nur bei der ersten Übertragung geladen. Für erneutes Laden muss es zuerst auf Null gesetzt und anschließend wieder auf Eins. Wird bei einem Lesezugriff auf der Empfangspage Data\_Control Bit11 eine 1 gelesen, so ist der Inkrementalzähler erfolgreich geladen worden.

**Sendepage:**

Wort0: Ist-Position	Wort1: Wert Multiplexed Data	Wort2: Data_Control
0x1234	0xXXXX	0x8800

Erklärung:

- Wort2 Data\_Control
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 11 gesetzt => Presetwert laden
- Wort0 Ist-Position
  - > Ist-Position auf den Wert 1234hex = 4660dezimal laden

**Empfangspage:**

Wort0: Ist-Position	Wort1: Multiplexed Data	Wort2: Data_Control
0x1234	0xXXXX	0x8800

Erklärung:

- Wort2 liefert als Rückantwort Schreibfreigabe und Preset\_CNT gesetzt
- Wort0 liefert den eingestellten Presetwert für die Ist- Position => 1234hex

## 5.3. Steuerung Multiplexed Data

Mit den 3-Bit Steuerbits (Bit 8 / 9 / 10) des Data\_Control werden die Vorgabefunktionen zum Virtual-Encoder gesteuert. Je Nachdem welche Einstellung gesetzt werden soll, muss das entsprechende Bit im Data\_Control gesetzt werden. Die Funktionen sind:

Funktion	Bit Zuweisung			Wert im Data_Control (Hex) (Write enable)
	MuxCtrl2 Bit10	MuxCtrl1 Bit9	MuxCtrl0 Bit8	
Ist-Frequenz	0	0	0	0x8000
Maximal-Frequenz	0	0	1	0x8100
Beschleunigungs-Frequenz	0	1	0	0x8200
Verzögerungs-Frequenz	0	1	1	0x8300
Stopp-Position	1	0	0	0x8400

Tabelle 5: Beschreibung Steuerung Multiplexed Data

Die folgende Abbildung zeigt eine Übersicht zur Verdeutlichung der Steuerung der Multiplexed Data.

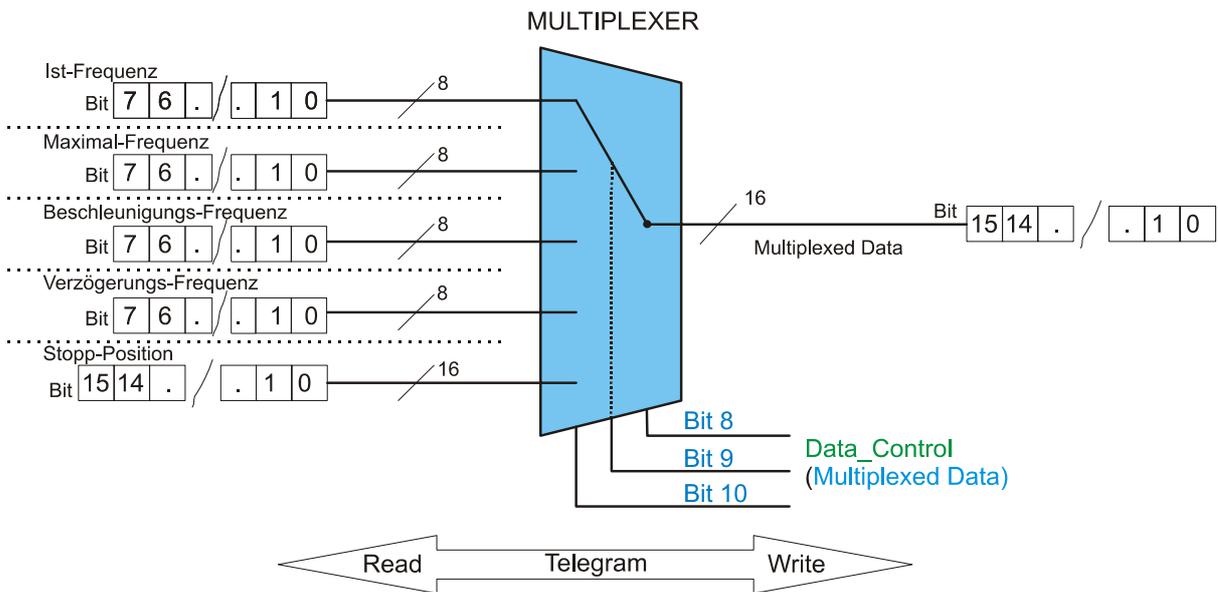


Abbildung 2: Übersicht Steuerung Multiplexed Data

Die Beschreibung der Funktionen folgt in den nachfolgenden Kapiteln.

### 5.3.1 Ist-Frequenz (Multiplexed Data)

Aktuelle Frequenz des Inkrementalsignals (nur im Master-Mode und Status "RUN"). Mit dieser Funktion kann die aktuelle Ist-Frequenz des Virtual-Encoders gelesen werden.

#### Sendepage:

Wort0: Ist-Position	Wort1: Wert Multiplexed Data	Wort2: Data_Control
0xXXXX	0xXXXX	0x9000

#### Erklärung:

- Wort0 Data\_Control
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 12 gesetzt => RUN

#### Empfangspage:

Wort0: Ist-Position	Wort1: Multiplexed Data	Wort2: Data_Control
0xXXXX	0x0010	0x9000

#### Erklärung:

- Wort0 liefert als Rückantwort Schreibfreigabe und RUN gesetzt
- Wort1 liefert die eingestellte Ist-Frequenz (hier 16kHz)

### 5.3.2 Maximal-Frequenz (Multiplexed Data)

Mit dieser Funktion wird die maximale Ausgabefrequenz der Inkrementalspuren eingestellt. Der Wert kann während des Betriebs geändert werden. Beschleunigung und Verzögerung findet dann mit den entsprechenden Werten der Beschleunigungs- / Verzögerungs-Frequenz statt. Es ist ein Wertebereich von 1-511 (dezimal) zulässig.

#### Sendepage:

Wort0: Ist-Position	Wort1: Wert Multiplexed Data	Wort2: Data_Control
0xXXXX	0x0006	0x8100

Erklärung:

- Wort0 **Data\_Control**
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 8 gesetzt => Steuerbit für Funktion Maximal-Frequenz
- Wort1 **Multiplexed Data**
  - > Wert 6hex = 6dezimal für die Maximalfrequenz

**Empfangspage:**

Wort0: Ist-Position	Wort1: <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0xXXXX</b>	<b>0x0006</b>	<b>0x8100</b>

Erklärung:

- Wort0 liefert als Rückantwort Schreibfreigabe und Steuerbit 8 gesetzt
- Wort1 liefert die eingestellte Maximal-Frequenz in Hex

### 5.3.3 Beschleunigungs-Frequenz (**Multiplexed Data**)

Diese Funktion gibt den Wert für die Beschleunigung der Frequenz an. Mit diesem Wert wird die aktuelle Ist-Frequenz im 8ms Takt inkrementiert. Es ist ein Wertebereich von **1-511** (dezimal) zulässig.

**Sendepage:**

Wort0: Ist-Position	Wort1: Wert <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0xXXXX</b>	<b>0x0001</b>	<b>0x8200</b>

Erklärung:

- Wort0 **Data\_Control**
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 9 gesetzt => Steuerbit für Funktion Beschleunigungs-Frequenz
- Wort1 **Multiplexed Data**
  - > Wert 1hex = 1dezimal für die Maximalfrequenz

**Empfangspage:**

Wort0: Ist-Position	Wort1: <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0xXXXX</b>	<b>0x0001</b>	<b>0x8200</b>

Erklärung:

- Wort0 liefert als Rückantwort Schreibfreigabe und Steuerbit 9 gesetzt
- Wort1 liefert die eingestellte Beschleunigungs-Frequenz in Hex

### 5.3.4 Verzögerungs-Frequenz (**Multiplexed Data**)

Diese Funktion gibt den Wert für die Verzögerung der Frequenz an. Mit diesem Wert wird die Ist-Frequenz im 8ms Takt dekrementiert. Es ist ein Wertebereich von **1-511** (dezimal) zulässig.

**Sendepage:**

Wort0: Ist-Position	Wort1: Wert <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0xXXXX</b>	<b>0x0002</b>	<b>0x8300</b>

Erklärung:

- Wort0 **Data\_Control**
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 8 und 9 gesetzt => Steuerbit für Funktion Verzögerungs-Frequenz
- Wort1 **Multiplexed Data**
  - > Wert 2hex = 2dezimal für die Maximalfrequenz

**Empfangspage:**

Wort0: Ist-Position	Wort1: Multiplexed Data	Wort2: Data_Control
0xXXXX	0x0002	0x8300

**Erklärung:**

- Wort0 liefert als Rückantwort Schreibfreigabe und Steuerbit 8 und 9 gesetzt
- Wort1 liefert die eingestellte Verzögerungs-Frequenz in Hex

Das Diagramm zeigt den Zusammenhang zwischen den einstellbaren Funktionen.

Die eingestellten Werte sind:

- Beschleunigungs-Frequenz => 0,65kHz
- Maximal-Frequenz => 3,88kHz
- Verzögerungs-Frequenz => 1,29kHz

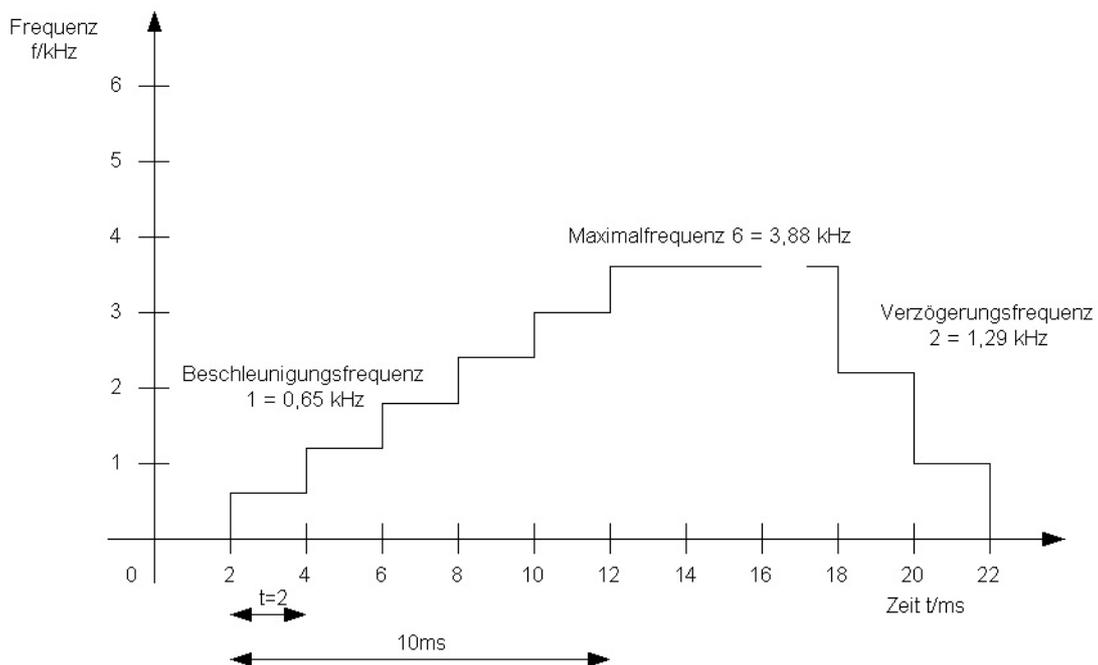


Abbildung 3: Zeitdarstellung Frequenzgang Ausgabesignal

## 5.3.5 Berechnung der Ausgabefrequenz

Die Ausgabefrequenz wird nach folgender Formel berechnet:

Gesucht:

**O\_FRQ** = Ausgabefrequenz in kHz

Gegeben:

**Osz** = Oszillatorfrequenz (20 000 000 Hz)

**V** = Programmierter Wert für "Maximalfrequenz"

Berechnung:

**h = 7742 ÷ V** Vom Ergebnis **h** Integerwert verwenden (Nachkommastellen abschneiden)

$$O\_FRQ = \left( \frac{Osz}{1000} \right) \div (h \times 4)$$

**Beispiel:**

Es wird die Maximalfrequenz **V = 153** dezimal programmiert, dies führt zu folgender Ausgabefrequenz O\_FRQ:

$$h = 7742 \div 153 = \underline{50,6} \quad \text{Integer -> } h = 50$$

$$O\_FRQ = \left( \frac{20000000}{1000} \right) \div (h * 4)$$

$$O\_FRQ = 20000 \div (50 * 4) = \underline{\underline{100kHz}}$$

**Wichtig:** Für den korrekten Betrieb des Virtual-Encoders müssen immer Frequenzparameter >0 programmiert werden!!!

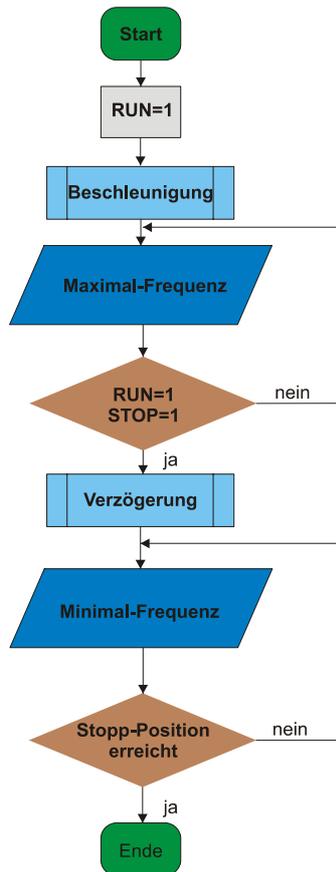
Es muss folgendes immer erfüllt sein:

Maximalfrequenz	>	"0"
Verzögerungsfrequenz	>	"0"
Beschleunigungsfrequenz	>	"0"

### 5.3.6 Stopp-Position (Multiplexed Data)

Mit dieser Funktion lässt sich eine Stopp-Position vordefinieren. Der Vorgabewert hat eine Größe von **16-Bit**. Als zusätzliche Funktion ist der Wert **Minimal-Frequenz** definiert. Die Minimal-Frequenz hat den Wert der programmierten Verzögerungs-Frequenz (Minimal-Frequenz = Verzögerungs-Frequenz). Beim Betriebsstatus **RUN=1** und **STOP=1** wird solange die Ist-Frequenz dekrementiert bis die Minimalfrequenz, die dem programmierten Wert der Verzögerungs-Frequenz entspricht, erreicht ist.

Folgendes Flussdiagramm soll den Programmablauf mit vordefinierter Stopp-Position verdeutlichen:



- 1) Wert für Stopp-Position setzen
- 2) Virtual-Encoder in Status RUN setzen (Bit 15 **Write\_Enable** und Bit12 **RUN**)
- 3) Der Virtual-Encoder beschleunigt mit dem Wert der Beschleunigungs-Frequenz bis die Maximal-Frequenz erreicht wird und gibt die Inkrementalimpulse mit der Maximal-Frequenz aus
- 4) Bevor die Stopp-Position erreicht ist muss zusätzlich der Status **STOP** (Bit13) gesetzt werden
- 5) Der Virtual-Encoder verzögert mit dem Wert der Verzögerungs-Frequenz bis zur Minimal-Frequenz
- 6) Sobald die Stopp-Position erreicht ist, stoppt der Virtual-Encoder die Ausgabe der Inkrementimpulse

Tabelle 6: Flussdiagramm Stopp-Position

**Sendepage:**

Wort0: Ist-Position	Wort1: Wert <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0xXXXX</b>	<b>0x9434</b>	<b>0x8400</b>

Erklärung:

- Wort0 **Data\_Control**
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 10 gesetzt => Steuerbit für Funktion Stopp-Position
- Wort1 **Multiplexed Data**
  - > Wert 9434hex für die Stopp-Position

**Empfangspage:**

Wort0: Ist-Position	Wort1: <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0xXXXX</b>	<b>0x9434</b>	<b>0x8400</b>

Erklärung:

- Wort0 liefert als Rückantwort Schreibfreigabe und Steuerbit 10 gesetzt
- Wort1 liefert die eingestellte Stopp-Position in Hex

## 6. SLAVE-Mode

Im **Slave-Mode** arbeitet der Virtual-Encoder als **16Bit Inkrementalzähler** (siehe Kapitel 3). Die externen Zählerwerte werden im Wort0 abgebildet. Das Wort2 enthält Statusinformationen zum Modul. Im Slave-Mode lässt sich ein vordefinierter Wert (Preset\_CNT) laden.

	Adresse Receive-Page (Offset <b>0x0100</b> )	Adresse Send-Page (Offset <b>0x0000</b> )
Wort0:	<b>0x0100</b> Ist-Position	<b>0x0000</b> <i>berichtigt: laden Ist-Position</i>
Wort1:	<b>0x0102</b> n.u.	<b>0x0002</b> n.u.
Wort2:	<b>0x0104</b> <b>Data_Control</b>	<b>0x0004</b> <b>Data_Control</b>

Tabelle 7: Übersicht Adressbelegung Slave-Mode

Die untere Tabelle beschreibt die Funktionen im **Data\_Control** (Wort2) für den SLAVE-Mode.

Bit	Funktion
<b>0</b>	Hardware Eingang <b>I0</b>
<b>1</b>	Hardware Eingang <b>I1</b>
<b>2</b>	Hardware Eingang <b>Nullpuls</b> von <b>Inkrementalzähler</b>
<b>3</b>	n.u.
<b>4</b>	n.u.
<b>5</b>	n.u.
<b>6</b>	n.u.
<b>7</b>	n.u.
<b>8</b>	n.u.
<b>9</b>	n.u.
<b>10</b>	n.u.
<b>11</b>	Preset_CNT (Inkrementalzähler laden)
<b>12</b>	n.u.
<b>13</b>	n.u.
<b>14</b>	n.u.
<b>15</b>	<b>Write_Enable</b> sämtliche Werte der Sendepage werden nur übernommen wenn <b>Bit15 = 1</b>

Tabelle 8: **Data\_Control** für SLAVE-Mode

### 6.1. Presetwert laden

#### Sendepage:

Wort0: Ist-Position	Wort1: Wert <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0x1234</b>	<b>0xXXXX</b>	<b>0x8800</b>

#### Erklärung:

- Wort2 Data\_Control
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 11 gesetzt => Presetwert laden
- Wort0 Ist-Position
  - > Ist-Position auf den Wert 1234hex = 4660dezimal laden

#### Empfangspage:

Wort0: Ist-Position	Wort1: <b>Multiplexed Data</b>	Wort2: <b>Data_Control</b>
<b>0x1234</b>	<b>0xXXXX</b>	<b>0x8800</b>

#### Erklärung:

- Wort2 liefert als Rückantwort Schreibfreigabe und Preset\_CNT gesetzt
- Wort0 liefert den eingestellten Presetwert für die Ist- Position => 1234hex

## A @ctiveIO-Systembus Kommunikation

Der @ctiveIO-Systembus wird über eine Sendepage und eine Empfangspage angesprochen. Die Sendepage enthält Daten welche zu einem Modul geschickt werden, die Empfangspage enthält Daten welche von einem Modul kommen.

Damit eine schnelle asynchrone Datenübertragung möglich ist, gibt es 2 Speicherbausteine. Der eine enthält die komplette Empfangs- / Sendepage für den Systembus und der andere für die PC-Schnittstelle. Eine Aktualisierung der Moduldaten erfolgt immer durch einen **ram\_switch** (Vertauschung der Schnittstellen Systembus und PC). Erst dann wird eine Datenübertragung angetriggert.

Damit beide Sendepages in beiden Speicherbausteinen gleichzeitig aktualisiert werden, gibt es den **Dual\_RAM\_Sendepage** Adressbereich. Dieser darf aber nur benutzt werden, wenn keine Übertragung auf dem Systembus läuft.

Die Ansteuerung des Systembus erfolgt in dem Adressbereich:

Basisadresse (C-Syntax):

```
#define MEMADR 0xD000000 // Basisadresse für Systembus-Kommunikation Memory-Bereich
```

Reale Hardware Basisadresse: **0xD0000**

	Adressoffset C-Prog (durch 16-Bit Typdeklaration Offset 1 Bit rechtsschieben)	Reale Hardware Adressoffset <b>Wichtig nur 16-Bit Zugriff erlaubt!</b>
Sendepage	0x0000 – 0x007F (0-127)	0x0000 – 0x00FF (0-255)
Empfangspage	0x0080 – 0x00FF (128-255)	0x0100 – 0x01FF (256-511)
Dual_RAM_Sendepage	0x0100 – 0x017F (256-383)	0x0200 – 0x02FF (512-767)
Steueregister	0x0200 – 0x027F (512 – 639)	0x0400 – 0x04FF (1024- 1279)

Wenn das Modul Virtual-Encoder der erste Busteilnehmer ist, dann kann man über folgendes Adress-Offset darauf zugreifen (16-Bit Wort-Zugriffe):

	Word Offset	
Send Page:	0x0000	Send Word 0
	0x0001	Send Word 1
	0x0002	Send Word 2
Receive Page:	0x0080	Receive Word 0
	0x0081	Receive Word 1
	0x0082	Receive Word 2
Dual RAM Send Page:	0x0100	Send Word 0
	0x0101	Send Word 1
	0x0102	Send Word 2

Genau diese Adressbereiche sind mit Daten oder Parametern belegt.

Es wird erst durch die Funktion *Status* zwischen Daten und Parametern unterschieden.

Parameter/Status übertragen:

- Sendepage mit den gewünschten Parametern beschreiben
- *Status*-Funktion aufrufen (Steuerregister)
- Ram\_switch ausführen (Steuerregister)

Um eine gültige Rückantwort auf eine Parameter- oder Daten- Übertragung zu bekommen sind mindestens drei ram\_switch notwendig (davor aber immer Funktion *Status* aufrufen). Vor dem Aufruf von ram\_switch sollte sichergestellt sein, dass eine eventuell vorherige Übertragung abgeschlossen

ist. Entweder Statusregister Bit M\_WARTEN=1 abfragen oder sicherstellen das mindestens eine Zeit von AnzahlBytes\*0,5 µs seit dem letzten ram\_switch verstrichen ist.

AnzahlBytes gibt die Anzahl der zu übertragenden Bytes auf dem Systembus an. Der Virtual-Encoder belegt 48Bit also 6 Byte. Ist der Virtual-Encoder der einzige Busteilnehmer muss mindestens 6\*0,5µs = 3 µs verstrichen sein, seit dem letzten ram\_switch.

Daten übertragen:

- 7) Sendepage mit gewünschten Daten beschreiben
- 8) Ram\_switch ausführen

## A.1 Setzen des Virtual-Encoder (über Systembus)

### Master-Mode

Sendepage

Adresse (Word Offset)	Data	
<b>0x0002</b>	<b>0x0801</b>	// Send Page Param_Control Betriebsart MASTER + Write_Enable
<b>0x0202</b>	<b>0x0080</b>	// set Status Bit7 in Set only Register
<b>0x0202</b>	<b>0x0008</b>	// set ram_switch Bit3 in Set only Register

Erklärung:

- Telegramm1
  - > Bit 1 gesetzt => Schreibfreigabe aktiv
  - > Bit 11 gesetzt => Betriebsart MASTER
- Telegramm2
  - > Bit 7 gesetzt, Schreibfreigabe nur Registerkommunikation
- Telegramm3
  - > Bit 3 gesetzt, ram\_switch Datenübernahme auf dem Systembus

### Slave-Mode

Wenn Notwendig kann später in den SLAVE-Mode gewechselt werden.

Sendepage

Adresse (Word Offset)	Data	
<b>0x0000</b>	<b>0x0001</b>	// Send Page Param_Control Betriebsart MASTER + Write_Enable
<b>0x0202</b>	<b>0x0080</b>	// set Status Bit7 in Set only Register
<b>0x0202</b>	<b>0x0008</b>	// set ram_switch Bit3 in Set only Register

Erklärung:

- Telegramm1
  - > Bit 1 gesetzt => Schreibfreigabe aktiv
  - > Bit 11 nicht gesetzt => Betriebsart SLAVE
- Telegramm2
  - > Bit 7 gesetzt, Schreibfreigabe nur Registerkommunikation
- Telegramm3
  - > Bit 3 gesetzt, ram\_switch Datenübernahme auf dem Systembus

## A.2 Auswerten der Ist-Position

### Beispiel

Receive-page

---

<b>ram_switch</b>	// Receivepage aktualisieren
Adresse (Word Offset)	
<b>0x0080</b>	// Wort 0 Receivepage Ist Position

---

### A.2.1 Presetwert

#### Ist-Position auf Wert **0x1F45** laden

Sendepage

---

Adresse (Word Offset)	Data	
<b>0x0000</b>	<b>0x1F45</b>	// Wert 1F45hex in Wort0 Dual_RAM_Send_PAGE
<b>0x0002</b>	<b>0x8800</b>	// Wort2 Dual_RAM_Send_PAGE Data_Control (Bit "Write enable" und Bit "Preset_CNT" auf 1 setzen)
<b>0x0202</b>	<b>0x0008</b>	// set ram_switch Bit3 in Set only Register

---

Erklärung:

- Telegramm1
  - > Wert für die vordefinierte Position
- Telegramm2
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 11 gesetzt => Presetwert laden
- Telegramm3
  - > Bit 3 gesetzt, ram\_switch Datenübernahme auf dem Systembus

## A.3 Stopp-Position

### Stopp-Position auf den Wert **0x9434** setzen:

Sendepage

---

Adresse (Word Offset)	Data	
<b>0x0001</b>	<b>0x9434</b>	// Wort 1 DUAL_RAM_Send_PAGE Multiplexed_Data
<b>0x0002</b>	<b>0x8400</b>	// Wort 2 DUAL_RAM_Send_PAGE Data_Control
<b>0x0202</b>	<b>0x0008</b>	// set ram_switch Bit3 in Set only Register

---

Erklärung:

- Telegramm1
  - > Wert 9434hex für die Stopp-Position
- Telegramm2
  - > Bit 15 gesetzt => Schreibfreigabe aktiv
  - > Bit 10 gesetzt => Steuerbit für Funktion Stopp-Position
- Telegramm3
  - > Bit 3 gesetzt, ram\_switch Datenübernahme auf dem Systembus