



JÄGER

Computergesteuerte
Messtechnik GmbH

ADwin-Pro II

System- und Hardware-Beschreibung



Hier finden Sie immer einen Ansprechpartner für Ihre Fragen:

Hotline: (0 62 51) 9 63 20
Fax: (0 62 51) 5 68 19
E-Mail: info@ADwin.de
Internet: www.ADwin.de



Jäger Computergesteuerte
Messtechnik GmbH
Rheinstraße 2-4
D-64653 Lorsch

Inhaltsverzeichnis

1 Inhaltsverzeichnis	III
Typografische Konventionen	IV
1 Das ADwin-Pro II -System	1
2 Installation des ADwin-Pro II -Systems	2
3 Betriebliche Umgebung	3
4 Gehäuse für das ADwin-Pro II -System	4
4.1 ADwin-Pro II	5
4.2 ADwin-Pro II-DC	6
4.3 ADwin-Pro II-BM	7
4.4 ADwin-Pro II-light	8
4.5 ADwin-Pro II-light-DC	9
4.6 ADwin-Pro II-mini	10
5 ADwin-Pro II -Module	11
5.1 Adressen der ADwin-Pro II -Module	11
5.2 Prozessormodule	13
5.3 Pro II: Multi-IO-Module	17
5.4 Pro II: Analoge Eingangsmodule	37
5.5 Pro II: Analoge Ausgangsmodule	69
5.6 Pro II: Digital-IO-Module	74
5.7 Pro II: Zusatz- und Schnittstellenmodule	95
6 Kalibrierung	122
6.1 Allgemeine Hinweise	122
6.2 Berechnungsgrundlagen	123
6.3 Kalibrieren	124
7 Zubehör	126
7.1 LEMO-Kabelsätze für ADwin-Pro -Systeme	126
7.2 LEMO-Adaptersätze	126
7.3 Bezugsadressen	126
Anhang	A-1
A.1 RoHS Konformitätserklärung	A-1
A.2 Liste der Module	A-1

Typografische Konventionen



Das „Achtung“-Zeichen steht bei Informationen, die auf Folgeschäden durch Fehlbedienung an der Hard- oder Software, am Messaufbau oder an Personen hinweisen.



Einen „Hinweis“ finden Sie bei

- Informationen, die für einen fehlerfreien Betrieb unbedingt beachtet werden müssen.
- Tipps und Ratschlägen für einen effizienten Betrieb.



Das Zeichen „Information“ verweist auf weiterführende Informationen in dieser Dokumentation oder andere Quellen wie Handbücher, Datenblätter, Literatur etc.

`<C:\ADwin\...>`

Dateinamen und -verzeichnisse sind in spitzen Klammern und im Schrifttyp Courier New angegeben.

`Programtext`

Programmanweisungen und Benutzer-Eingaben sind durch den Schrifttyp Courier New gekennzeichnet.

`Var_1`

Elemente eines Quelltextes wie Befehle, Variablen, Kommentar und sonstiger Text werden im Schrifttyp Courier New und farbig dargestellt (wie im Editor der Entwicklungsumgebung *ADbasic*).

In einem Datenwort (hier: 16 Bit) werden die Bits wie folgt nummeriert:

Bit-Nr.	15	14	13	...	1	0
Wert des Bits	2^{15}	2^{14}	2^{13}	...	$2^1=2$	$2^0=1$
Bezeichnung	MSB	-	-	-	-	LSB

1 Das ADwin-Pro II-System

Das *ADwin-Pro II*-System ist ein modular erweiterbares Prozessrechner-System. Je nach Anforderung können die verschiedenen Gehäuseformen mit bisherigen *ADwin-Pro*- und neuen *ADwin-Pro II*-Modulen bestückt werden.

Bei der Entwicklung des *ADwin-Pro II*-Systems wurde großer Wert auf die EMV-Verträglichkeit gelegt. Das *ADwin-Pro II*-System hat mit allen lieferbaren Ein- / Ausgabemodulen das CE-Zeichen und kann deshalb bei Bedarf auch nachträglich beliebig umkonfiguriert werden.

Jedes *ADwin-Pro II*-System benötigt ein Prozessormodul. Das Prozessormodul kommuniziert über Ethernet mit dem PC oder Notebook.

Um den vielfältigen Anforderungen bei Mess- und Steuerungsaufgaben zu entsprechen, kann das System mit folgenden Modulen ausgerüstet werden:

- analoge Eingabemodule und analoge Ausgabemodule
- digitale Eingabemodule und digitale Ausgabemodule
- Zähler
- Filter, Trennverstärker
- Verstärker für Thermoelemente und PT100-Widerstände
- serielle Kommunikations-Schnittstellen: CAN, LIN, RSxxx, Feldbus
- Speicher-/Lese-Modul für PCMCIA-Datenträger

Alle Module haben eine Revisionsbezeichnung auf der Frontseite, z. B. Rev. A2, Rev. B3, Rev. C3. Früher gelieferte Module sind nicht gekennzeichnet; sie besitzen den Revisionsstand „Rev. A“. *ADwin-Pro II*-Module tragen die Revisionsbezeichnung Rev. E1 oder höher.

Unterschiedliche Revisions-Buchstaben bedeuten unterschiedliche Moduleigenschaften und sind separat dokumentiert.

Der Revisionsbezeichnung angehängt ist eine untergeordnete Zählnummer, die für interne Zwecke der Firma Jäger Computergesteuerte Messtechnik GmbH verwendet wird.

Ausrüstung mit Modulen

Revisionsbezeichnung

2 Installation des ADwin-Pro II-Systems

Halten Sie bitte unbedingt die folgende Reihenfolge ein:

1. Beginnen Sie mit dem Handbuch „ADwin-Installation“:
 - Software und Schnittstellen-Treiber von der ADwin-CD installieren.
 - Datenverbindung vom PC zum ADwin-System in Betrieb nehmen und auf korrekte Funktion prüfen.
Die Anschlüsse für die Stromversorgung sind in Kapitel 4 “Gehäuse für das ADwin-Pro II-System“ beschrieben.
 - Beachten Sie die Hinweise in Kapitel 3 “Betriebliche Umgebung“.
2. Moduladressen einstellen siehe Kapitel 5.1 auf Seite 11.
3. Machen Sie die ersten Schritte mit dem ADbasic-Tutorial.
4. Programmieren in ADbasic:

Das Handbuch ADbasic beschreibt die Echtzeit-Entwicklungsumgebung, den Aufbau eines ADbasic-Programms und gibt Hinweise für Optimierungen.

Die ADbasic-Befehle finden Sie in der Online-Hilfe der Entwicklungsumgebung oder in diesen Dokumenten:

- Handbuch ADbasic: Grundlegende Befehle für Berechnungen, Programmstruktur und Prozesssteuerung.
- Handbuch ADwin-Pro II Systembeschreibung: Befehle und Hinweise zum Ansprechen der Pro-Module.

Beachten Sie auch die Hinweise zu den Modulen in diesem Handbuch.

Bitte beachten Sie folgende Hinweise

Damit Ihr ADwin-System sicher arbeitet, halten Sie sich an die Informationen dieser und weiterführender Dokumentationen, auf die hier verwiesen wird.

Der Hersteller des in dieser Dokumentation beschriebenen Systems geht davon aus, dass an dem Gerät nur qualifiziertes Personal arbeitet.

*Qualifiziertes Personal sind Personen, die aufgrund ihrer Ausbildung, Erfahrung und Unterweisung sowie ihrer Kenntnisse über einschlägige Normen, Bestimmungen, Unfallverhütungsvorschriften und Betriebsverhältnisse von dem für die Sicherheit der Anlage Verantwortlichen be-rechtigt worden sind, die jeweils erforderlichen Tätigkeiten auszuführen und die dabei mögliche Gefahren erkennen und vermeiden können.
(Definition für Fachkräfte nach VDE 105 und IEC 60364).*

Diese Produktdokumentation und Unterlagen, auf die verwiesen wird, müssen stets verfügbar sein und konsequent beachtet werden. Für Schäden, die durch Missachtung der Informationen in dieser bzw. der weiterführenden Dokumenten entstehen, übernimmt die Firma Jäger Computergesteuerte Messtechnik GmbH, Lorsch, keine Haftung.

Diese Dokumentation ist einschließlich aller Abbildungen urheberrechtlich geschützt. Reproduktion, Übersetzung sowie elektronische und fotografische Archivierung und Veränderung bedürfen der schriftlichen Genehmigung der Firma Jäger Computergesteuerte Messtechnik GmbH, Lorsch.

Fremdprodukte werden ohne Vermerk auf mögliche Patentrechte genannt, deren Existenz nicht auszuschließen ist.

Hotline-Adresse siehe vordere Umschlagseite, innen.

Einschränkung der Anwendergruppe

Verfügbarkeit der Unterlagen



Rechtliche Grundlagen

Änderungen vorbehalten.

3 Betriebliche Umgebung

Das *ADwin-Pro II*-Gerät muss geerdet werden, um

- einen Massebezugspunkt für die Elektronik herzustellen und
- Störungsenergie auf die Erde ableiten zu können.

Verbinden Sie dazu die GND-Klemme / Buchse über ein kurzes impedanzarmes Masseband mit dem zentralen Erdungspunkt Ihrer Anlage. Die GND-Klemme / Buchse ist im Gerät mit der Masse und dem Gehäuse verbunden.

Beim Ethernet-Kabel sind die Datenleitungen galvanisch entkoppelt, die Massepotenziale sind jedoch gekoppelt, weil die Schirmung des Ethernet-Steckers (RJ-45) mit GND verbunden ist.

Ausgleichsströme, die über das Gehäuse oder die Schirmung abfließen, beeinflussen das Messsignal.

Wenn Sie Ausgleichsströme vermindern wollen, müssen Sie darauf achten, dass die Wirkung des Schirmes erhalten bleibt, indem Sie geeignete Maßnahmen zur Ableitung von Störungen treffen, wie z.B. das Auflegen des Schirms kurz vor dem Eintritt in den Schaltschrank. Je häufiger Sie die Schirmung auf dem Weg zur Maschine erden, desto besser ist die Schirmwirkung.

Verwenden Sie für die **Signalleitungen** möglichst Kabel mit beidseitig aufgelegtem Schirm. Auch hier sollte das Ableiten von Störungen über das Gehäuse mit der Verwendung von Schirmklemmen reduziert werden.

Betreiben Sie das Gerät nur mit der passenden Netzspannung. Für den Betrieb mit einem externem Netzteil gelten die Angaben des Herstellers. Betreiben Sie das Gerät nur im geschlossenen Zustand, schließen Sie Lücken zwischen den eingebauten Modulen mit Abdeckplatten.

ADwin-Pro II ist für den Betrieb in trockenen Räumen konzipiert und muss daher vor Feuchtigkeit und Kondenswasser geschützt werden. Das Gehäuse kann in Schaltschränken eingebaut oder mobil betrieben werden (z.B. im Kfz). Am Einbauort sollen eine Umgebungstemperatur von +5°C ... +50°C und eine relative Luftfeuchte von 0 ... 80% (nicht kondensierend) vorhanden sein.

Die Gehäusetemperatur (Oberflächentemperatur) darf auch unter extremen betrieblichen Bedingungen, z.B. im Schaltschrank oder bei direkter Sonneneinstrahlung, +60°C nicht überschreiten. Es besteht sonst die Gefahr, dass Schäden am Gerät entstehen oder nicht definierte Daten (Werte) ausgegeben werden, die unter ungünstigen Umständen zu Schäden in ihrer Anlage führen können.

Beachten Sie insbesondere beim Schaltschrank-Einbau:

- Das *ADwin-Pro II*-Gerät soll nicht über starken Wärmequellen stehen wie z.B. Leistungstransformatoren.
- Die Be- und Entlüftung im Schaltschrank bis zum und vom *ADwin-Pro II*-Gerät muss gewährleistet sein. Insbesondere müssen die Lüftungsschlitze des Geräts frei bleiben, so dass die vom Gerät erzeugte Wärme vollständig abgeführt wird.

Erdung



Galvanische Kopplung

Ausgleichströme ausschließen



Netzspannung

Umgebungsklima

Gehäusetemperatur



4 Gehäuse für das ADwin-Pro II-System

Die Gehäusevarianten für das ADwin-Pro II-System unterscheiden sich durch die Anzahl der Steckplätze und die Art der Stromversorgung:

Gehäuse	Anzahl Steckplätze	Stromversorgung	
ADwin-Pro II	16	100V...240V	AC
ADwin-Pro II-DC	16	10V...35V	DC
ADwin-Pro II-BM	15	100V...240V	AC
ADwin-Pro II-light	7	100V...240V	AC
ADwin-Pro II-light-DC	7	10V...35V	DC
ADwin-Pro II-mini	5	10V...36V	DC

Die Anzahl der Steckplätze gilt für Pro II-Module. Wenn Pro I-Module verwendet werden – auch gemischt mit Pro II-Modulen – passen weniger Module in das Gehäuse.

Für die Abmessungen des Einschubbereichs (inklusive Netzteileinschub) gelten folgende Maßeinheiten:

$$1 \text{ TE} = 1/5 \text{ inch} = 5,08 \text{ mm}$$

$$1 \text{ HE} = 5/3 \text{ inch} = 42,3 \text{ mm}$$

Die Einsteckmodule haben meistens eine Breite von 5 TE = 1 inch.

Modul einstecken



So stecken Sie ein Modul in das Gehäuse ein:

- Schalten Sie das Gerät aus! Ein Modul kann beschädigt werden, wenn Sie es bei eingeschalteter Stromversorgung einstecken oder herausziehen.
- Entfernen Sie ein oder mehr Abdeckbleche an der gewünschten Position, so dass am linken Rand die Führungsschienen zu sehen sind: je eine oben und eine unten.
 - Achten Sie auf die Farbe der Schienen. Es gibt für Pro I- und Pro II-Module verschiedene, leicht versetzte Führungsschienen:
Weiße Schienen: Pro II-Module.
Schwarze Schienen: Pro I-Module.
 - Das Prozessormodul hat einen festen Steckplatz, andere Positionen sind nicht möglich.
- Führen Sie die Platine mit dem Stecker voran oben und unten sorgfältig in die Führungsschienen ein. Bei korrekter Positionierung lässt sich das Modul nicht schräg stellen.
- Schieben Sie das Modul ganz nach hinten. Am Ende spüren Sie einen leichten Widerstand, wenn Sie den Modulstecker in die Buchse der Rückwand einschieben.

Das Deckblech des Moduls sollte ganz am Gehäuse anliegen.

- Drehen Sie die Schrauben oben und unten am Deckblech fest.
- Schließen Sie eventuelle Lücken zwischen den eingebauten Modulen mit den Abdeckplatten. Es gibt Abdeckplatten mit 2, 3 oder 5 TE Breite.

4.1 ADwin-Pro II

Das Standard-Gehäuse für *ADwin-Pro II*-Systeme. Die Rückwand (Backplane) des Gehäuses verbindet das Prozessormodul mit den anderen Modulen über einen internen Bus.

Die Gerätesicherung befindet sich an in einem Einschub im Netzteil, oberhalb der Buchse für den Netzstecker (Gehäuserückseite).

Anzahl Steckplätze	16
Außenabmessungen (L x B x H)	336mm x 447,5mm x 146mm
Einschubbereich (B x H)	84 TE x 3 HE
Netzteil	min. 70W, 100V...240VAC bei 50/60Hz Schaltnetzteil
Sicherung	5A, träge

Abb. 1 – Gehäuse *ADwin-Pro II*: Spezifikation

Auf der Gehäuserückseite befindet sich über dem Netzstecker ein Aufkleber mit der Revisionsbezeichnung des Gehäuses:

Revision	Ausgabe	Änderung zur Vorgänger-Version
E1	Jun. 2005	<i>ADwin-Pro II</i> : Neue Gehäusekonstruktion und neue Backplane mit Pro I- und Pro II-Bus.

Im Pro II-Gehäuse können sowohl Pro I-Module als auch Pro II-Module eingesteckt werden. Die Gehäuserückwand enthält sowohl den Pro I-Bus als auch den Pro II-Bus; das Prozessormodul arbeitet parallel mit beiden Bussen.

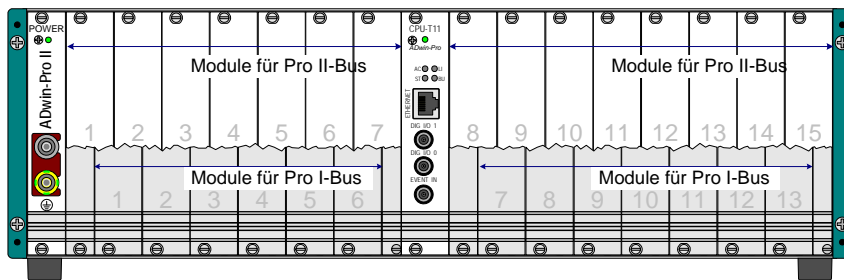


Abb. 2 – Gehäuse *ADwin-Pro II*

Bitte beachten Sie, dass Module für Pro I-Bus (im Bild grau) und Pro II-Bus leicht versetzte Einsteckpositionen haben. Die Positionen sind an der Farbe der Schienen leicht zu unterscheiden:

- Weiße Schienen: Module für Pro II-Bus.
- Schwarze Schienen: Module für Pro I-Bus.

Ausgangsmodule Pro-AOut-x mit Rev. A können aus technischen Gründen nicht verwendet werden.

Das Prozessormodul wird immer in der Mittelposition eingesteckt (weiße Schienen).

Zwischen dem Prozessormodul und Modulen für Pro I-Bus bleibt ein halber Steckplatz frei (Abdeckbleche liegen bei), während Module für Pro II-Bus ohne Abstand neben das Prozessormodul passen.

16 Steckplätze

Gehäuse Pro II



Pro II-DC
16 Steckplätze


4.2 ADwin-Pro II-DC

Das Gehäuse *ADwin-Pro II-DC* entspricht vollständig dem Standardgehäuse *ADwin-Pro II*, ist aber mit einem Gleichstromnetzteil ausgerüstet.

Wenn zur Spannungsversorgung ein strombegrenzendes Netzteil verwendet wird, sollte dies in der Lage sein, beim Einschalten ein Mehrfaches des Ruhestroms zur Verfügung zu stellen, um einen einwandfreien Betrieb zu gewährleisten.

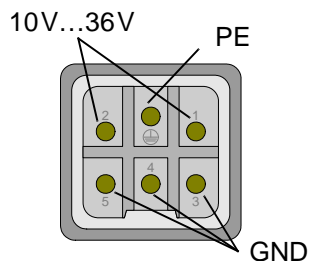


Abb. 3 – Gehäuse *ADwin-Pro II-DC*:
Detailansicht der Steckerbelegung

Anzahl Steckplätze	16
Außenabmessungen (L x B x H)	336mm x 447,5mm x 146mm
Einschubbereich (B x H)	84 TE x 3 HE
Netzteil	min. 80W, DC-DC-Wandler 10V...35V
Sicherung	25A, 58V

Abb. 4 – Gehäuse *ADwin-Pro-DC*: Spezifikation

Auf der Gehäuserückseite befindet sich über dem Netzstecker ein Aufkleber mit der Revisionsbezeichnung des Gehäuses:

Revision	Ausgabe	Änderung zur Vorgänger-Version
E1	Jun. 2005	<i>ADwin-Pro II</i> : Neue Gehäusekonstruktion und neue Backplane mit Pro I- und Pro II-Bus. Neuer Stromversorgungsstecker.

4.3 ADwin-Pro II-BM

Das Gehäuse entspricht vollständig dem Standardgehäuse, aber die Module werden auf der Rückseite eingesteckt (BM = back mounted).

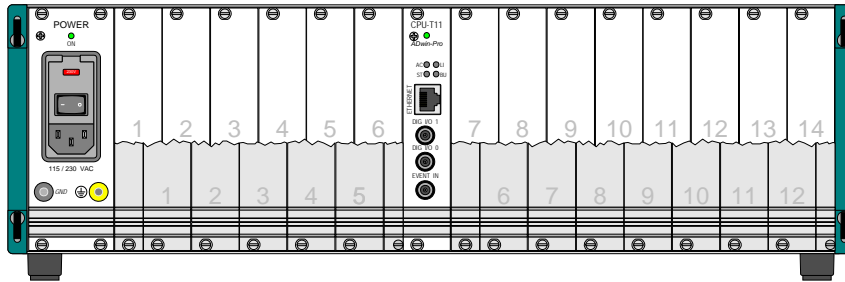


Abb. 5 – Gehäuse ADwin-Pro II-BM (Rückseite)

Die Gerätesicherung befindet sich in einem Einschub im Netzteil, oberhalb der Buchse für den Netzstecker.

Anzahl Steckplätze	16
Außenabmessungen (L x B x H)	336mm x 447,5mm x 146mm
Einschubbereich (B x H)	84 TE x 3 HE
Netzteil	min. 70W, 100...240VAC bei 50/60Hz Schaltnetzteil
Sicherung	5A, träge

Abb. 6 – Gehäuse ADwin-Pro II-BM: Spezifikation

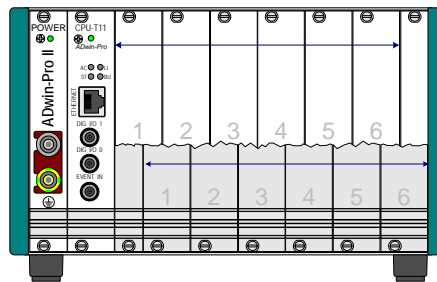
Auf der Gehäuserückseite befindet sich über dem Netzstecker ein Aufkleber mit der Revisionsbezeichnung des Gehäuses:

Revision	Ausgabe	Änderung zur Vorgänger-Version
E1	Jun. 2005	ADwin-Pro II: Neue Gehäusekonstruktion und neue Backplane mit Pro I- und Pro II-Bus.

Pro II „backmounted“ mit 15 Steckplätzen

Pro II-light mit
7 Steckplätzen

4.4 ADwin-Pro II-light



Gehäuse ADwin-Pro II-light

Die Rückwand (Backplane) des Gehäuses verbindet das Prozessormodul mit den Pro-Modulen.

Anzahl Steckplätze	7
Außenabmessungen (L x B x H)	336mm x 234mm x 146mm
Einschubbereich (B x H)	42 TE x 3 HE
Netzteil	min. 40W, 100...240VAC bei 50/60Hz Schaltnetzteil
Sicherung	2A, träge

Abb. 7 – Gehäuse ADwin-Pro II-light. Spezifikation

Auf der Gehäuserückseite befindet sich über dem Netzstecker ein Aufkleber mit der Revisionsbezeichnung des Gehäuses:

Revision	Ausgabe	Änderung zur Vorgänger-Version
E1	Jun. 2005	ADwin-Pro II: Neue Gehäusekonstruktion, schmaler Netzteileinschub und neue Backplane mit Pro I- und Pro II-Bus.

4.5 ADwin-Pro II-light-DC

Das Gehäuse *ADwin-Pro II-light-DC* entspricht vollständig dem Gehäuse *ADwin-Pro II-light*, ist aber mit einem Gleichstromnetzteil ausgerüstet.

Wenn zur Spannungsversorgung ein strombegrenzendes Netzteil verwendet wird, sollte es in der Lage sein, beim Einschalten ein Mehrfaches des Ruhestroms zur Verfügung zu stellen, um einen einwandfreien Betrieb zu gewährleisten.

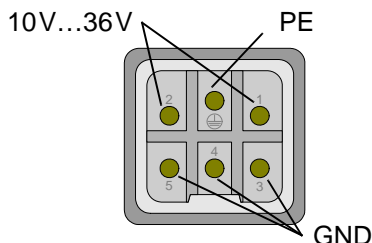


Abb. 8 – Gehäuse *ADwin-Pro II-light-DC*: Anschlussbuchse für die Stromversorgung

Anzahl Steckplätze	7
Außenabmessungen (L x B x H)	336mm x 234mm x 146mm
Einschubbereich (B x H)	42 TE x 3 HE
Netzteil	min. 80W, DC-DC-Wandler 10V...35V
Sicherung	15A, 58V

Abb. 9 – Gehäuse *ADwin-Pro II-light-DC*: Spezifikation

Auf der Gehäuserückseite befindet sich über dem Netzstecker ein Aufkleber mit der Revisionsbezeichnung des Gehäuses:

Revision	Ausgabe	Änderung zur Vorgänger-Version
E1	Okt. 2006	Erst-Version, nur für Pro II.

Das Gehäuse wird mit einer Anschlussbuchse für die Stromversorgung ausgeliefert. Die Buchse kann bei Bedarf wie folgt nachbestellt werden:

Bezugsadresse: Regional-Electronic-Distribution Handelsgesellschaft mbH
Postfach 1250, 63084 Rodgau

Bezeichnungen: Steckergehäuse Harting, Serie HA.3.XX.X,
Best.nr. HA.3.STO.1.11

Buchseneinsatz Harting, 5+PE, 400V, 16A,
Best.nr. HE.Q.5.BU.C

4 Stk. Buchsenkontaktstift 2.5mm², vergoldet,
Best.nr. HE-HA.C.BU.2,5.AU

**Pro II-light-DC
mit 7 Steckplätzen**



Pro II-mini mit 5 Steckplätzen

4.6 ADwin-Pro II-mini

Das kleinste Gehäuse für ADwin-Pro II-Systeme hat 5 Steckplätze und benötigt ein externes Netzteil. Der Anschluss für das Netzteil ist auf der Rückseite des Gehäuses.

Anzahl Steckplätze	5
Außenabmessungen (L x B x H)	253mm x 147,3mm x 146mm
Einschubbereich (B x H)	20 TE x 3 HE
Sicherung	10A, 58V
Externes Netzteil	externes Netzteil erforderlich: 10V...36V DC, 50W

Abb. 10 – Gehäuse ADwin-Pro II-mini: Spezifikation

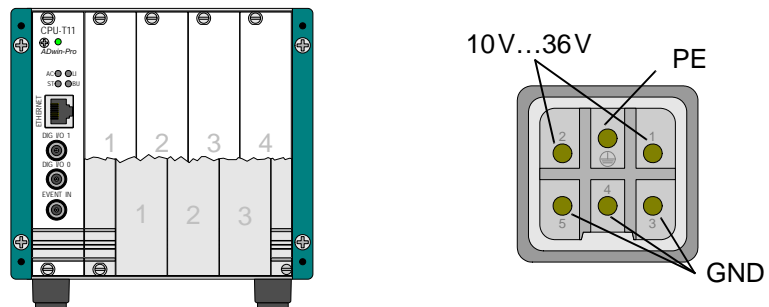


Abb. 11 – Gehäuse ADwin-Pro II-mini und Anschlussbuchse für die Stromversorgung

Auf der Gehäuserückseite befindet sich ein Aufkleber mit der Revisionsbezeichnung des Gehäuses:

Revision	Ausgabe	Änderung zur Vorgänger-Version
E1	Dez. 2005	ADwin-Pro II: Neue Gehäusekonstruktion und neue Backplane mit Pro I- und Pro II-Bus. Variable Eingangsspannung.

5 ADwin-Pro II-Module

Ein *ADwin-Pro II*-Modul belegt in der Regel einen Steckplatz (5 TE) in einem *ADwin-Pro II*-System, manche Module auch 2 Steckplätze.

Alle technischen Daten der Module beziehen sich auf das eingeschaltete Gerät.

Beachten Sie beim Einstecken eines Moduls in das Gehäuse die Beschreibung auf Seite 4.



5.1 Adressen der ADwin-Pro II-Module

Ein Pro-Modul (ausgenommen CPU-Module) wird in einem *ADbasic*-Programm über seine Moduladresse angesprochen. Die Moduladresse ist weitgehend frei wählbar.

Moduladresse wählen

Für das Wählen der Moduladresse gelten folgende Regeln:

- Eine Moduladresse muss innerhalb der Modulgruppe eindeutig sein.

Jedes Modul gehört zu einer Modulgruppe:

- Pro II-Module.
- Pro I-Module, Funktionsgruppe CPU: Prozessormodule.
- Pro I-Module, Funktionsgruppe ADC: analoge Eingangsmodule.
- Pro I-Module, Funktionsgruppe DAC: analoge Ausgangsmodule.
- Pro I-Module, Funktionsgruppe DIO: digitale Ein- / Ausgangs-, Relais- und Zählermodule.
- Pro I-Module, Funktionsgruppe EXT: Sondermodule aller Art.

- Die Moduladresse kann in folgenden Grenzen frei gewählt werden:

- Pro II-Module: 1 ... 15.
- Pro I-Module: 1 ... 255.

Es ist zwar möglich, Modulen aus verschiedenen Gruppen die gleiche Moduladresse zu geben. Um Verwechslungen zu vermeiden, empfehlen wir aber, eindeutige Adressen zu vergeben.



Moduladresse einstellen: Pro II-Module

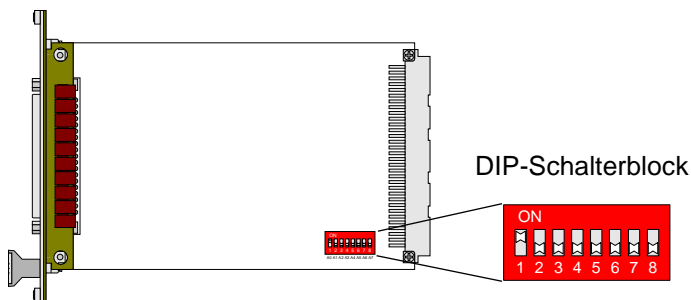
Bei Pro II-Modulen stellen Sie die Moduladresse mit dem Programm *ADpro* ein. Das Einstellen setzt gleichzeitig alle Pro II-Module in den Anfangszustand zurück. Gehen Sie vor wie folgt:

- Schalten Sie die Stromversorgung des *ADwin*-Systems aus und stecken die Pro II-Module in das Gehäuse ein; beachten Sie die Beschreibung auf Seite 4. Schalten Sie die Stromversorgung wieder ein.
- Anschließend starten Sie das Programm *ADpro* aus dem Windows Startmenü unter *Programs* ▶ *ADwin*. Unter dem Menüpunkt *Edit* ▶ *Set module addresses* können Sie die gewählten Moduladressen einstellen.

Moduladresse einstellen: Pro I-Module

Bei Pro I-Modulen stellen Sie die Moduladresse manuell an einem DIP-Schalterblock ein. Der Schalterblock befindet sich auf der Platine rechts unten.

Mit den 8 DIP-Schaltern ist eine Adresse zwischen 1 und 255 einstellbar (siehe Abb. 12). Wie oben beschrieben, muss jedes Modul einer Gruppe eine eindeutige Adresse haben.



Moduladresse	Einstellung der DIP-Schalter							
	1	2	3	4	5	6	7	8
1	1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0
3	1	1	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0
5	1	0	1	0	0	0	0	0
6	0	1	1	0	0	0	0	0
7	1	1	1	0	0	0	0	0
8	0	0	0	1	0	0	0	0
...	...							
254	0	1	1	1	1	1	1	1
255	1	1	1	1	1	1	1	1

Abb. 12 – Adressierung der ADwin-Pro I-Module mit den DIP-Schaltern



Beachten Sie bitte:

- Ein RSxxx-Modul mit 4 seriellen Schnittstellen belegt 2 Adressen (Gruppe EXT): die eingestellte Adresse und die nächsthöhere.
- Ein Feldbus-Modul belegt 32 Adressen (Gruppe EXT); die Adressverteilung ist wie folgt:

Eingestellte Moduladresse	Zusätzl. belegte Adressen	Einstellung der DIP-Schalter							
		1	2	3	4	5	6	7	8
1	160...191	1	0	0	0	0	0	0	0
2	192...223	0	1	0	0	0	0	0	0
3	224...255	1	1	0	0	0	0	0	0
4	128...159	0	0	1	0	0	0	0	0

5.2 Prozessormodule

Für jedes *ADwin-Pro II*-System wird ein Prozessormodul benötigt. Das Prozessormodul ist die zentrale Recheneinheit des Pro II-Systems und hat folgende Aufgaben:

- Kommunikation mit dem PC oder Laptop. Die Verbindung wird über Ethernet hergestellt.
- Kommunikation mit allen anderen Pro II-Modulen über den internen Bus.
- Kommunikation mit eventuell vorhandenen TiCo-Prozessoren auf Pro II-Modulen über den internen Bus.
- Aufnehmen und Ausführen der benutzerdefinierten Prozesse.

Auf dem Prozessormodul ist der Speicher für Daten und Programme untergebracht, unterteilt in den schnellen internen Speicher (SRAM) und den externen Speicher (DRAM).

Derzeit steht für *ADwin-Pro II* das Prozessormodul Pro-CPU-T11 zur Verfügung.



5.2.1 Pro-CPU-T11

Das Prozessormodul kann nur in einem Pro II-Gehäuse eingesetzt werden und arbeitet sowohl mit Pro I- als auch mit Pro II-Modulen.

Die Ausgangsmodule Pro II-AOut-x arbeiten erst ab Rev. B mit dem T11.

Geeignet für Pro II-System	Pro II
Prozessor	ADSP TS101S
Taktrate	300MHz
Rechenauflösung Float-Werte	40 Bit
Datenleitung	Ethernet
Interner Speicher	768KiB
Externer Speicher	256MiB
TTL-Signaleingänge	Event In, mit 4,7 kΩ Pull-down-Widerstand Dig I/O 0, mit 4,7 kΩ Pull-up-Widerstand Dig I/O 1, mit 4,7 kΩ Pull-up-Widerstand

Abb. 13 – Pro-CPU-T11: Spezifikation

Das Prozessormodul hat im Gehäuse eine feste Position. Beachten Sie zum Einstecken des Moduls auch die Hinweise auf Seite 4: Modul einstecken.

Der interne Speicher des Prozessors ist aufgeteilt in Programmspeicher (PM), Datenspeicher (DM) und frei verwendbaren Zusatzspeicher (EM). Die Speicherbereiche haben je eine Größe von 256 KiB.

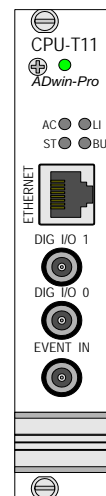


Abb. 14 – Pro-CPU-T11: Frontplatte

Das Prozessormodul signalisiert seinen Betriebszustand durch LEDs am Ethernet-Anschluss. Die Bedeutung der LEDs ist im Handbuch ADwin Installation, Kapitel 10.3 beschrieben.

Eingang Event In

Mit dem externen Trigger-Eingang (**Event In**) kann das Prozessormodul ein externes Signal (Trigger) als Event-Signal erkennen und einen Prozess auslösen, der sofort und vollständig abgearbeitet wird (siehe Handbuch *ADbasic*, Kapitel: Struktur des *ADbasic*-Programms).

Das Event-Signal muss 50ns lang anstehen, um erkannt zu werden.

Alternativ kann ein Event-Eingang eines anderen Moduls verwendet werden. Alle Event-Signale gelangen beim Prozessormodul auf die gleiche Signalleitung wie der Eingang **Event In**.

Bestehende *ADbasic*-Programme können – soweit sie mit Pro I-Modulen arbeiten – mit wenigen, aber notwendigen Änderungen weiter verwendet werden:

- Für den Prozessor T11 muss die Include-Datei `<ADwinPro II_All.inc>` neu eingebunden werden. Gleichzeitig sollten alle anderen Include-Dateien für Pro II-Module aus dem Programm gelöscht werden.
- Die Zeiteinheit des **Processdelay** (Zykluszeit) beträgt $3,3\text{ ns}$ für hoch-priore wie für nieder-priore Prozesse.
Alle entsprechenden Werte und Berechnungen müssen angepasst werden. Das größtmögliche **Processdelay** entspricht $7,1\text{ s}$; für größere Zykluszeiten muss eine Hilfsvariable verwendet werden.
- Die Zeiteinheit von $3,3\text{ ns}$ gilt auch für den internen Zähler, d.h. Abfragen des Zählers mit **Read_Timer** müssen ebenfalls angepasst werden.
Beachten Sie bitte: Das Zeitverhalten von Prozessen im Zusammenhang mit Ein-/Ausgabebefehlen ist komplexer geworden (s. u.). Mit **Read_Timer** bestimmte Zeitdifferenzen spiegeln jetzt nur noch einen Teilaspekt des gesamten Zeitverhaltens wider.
- Der Befehl **SLEEP** muss durch einen der folgenden neuen Befehle ersetzt werden:
 - **CPU_SLEEP** lässt den Prozessor warten. Der Befehl **SLEEP** hatte bei den Prozessoren T9 und T10 die gleiche Funktion.
 - **P1_SLEEP** lässt den Pro I-Bus warten, z. B. um Ein-/Ausgabebefehle aufeinander abzustimmen. Der Befehl lässt auch Pro II-Bus und externen Speicher warten.
 - **P2_SLEEP** lässt den Pro II-Bus warten, z. B. um Ein-/Ausgabebefehle aufeinander abzustimmen.

Die neuen Befehle haben eine Zeiteinheit von 10 ns (bei **SLEEP**: 100 ns).

Welcher Befehl ist der richtige? In der Regel wird **SLEEP** verwendet, um die Wartezeit eines Ein-/Ausgabebefehls zu überbrücken, z. B. das Einschwingen des Multiplexers bei **SET_MUX**. In diesem Fall ist für bisherige Module (Pro I-Bus) **P1_SLEEP** der passende Befehl, für Pro II-Module ist es **P2_SLEEP**.

Beachten Sie die Hinweise im *ADbasic*-Handbuch, wie Sie die Wartezeit genau einstellen (Kapitel 5.2.4).

Warum gibt es neue Befehle? Der Prozessor T11 unterscheidet zwischen Prozessorbefehlen einerseits und Ein-/Ausgabebefehlen andererseits. Die Prozessorarchitektur erlaubt oft eine quasi-parallele Bearbeitung¹ der beiden Befehlsgruppen und damit eine deutlich schnellere Bearbeitung der *ADbasic*-Prozesse. Das heißt gleichzeitig, dass die Befehlsgruppen (weitgehend) zeitlich unabhängig bearbeitet werden. Um das Zeitverhalten durch Warten zu beeinflussen, sind deshalb den Gruppen zugeordnete Befehle notwendig. Die Unterscheidung nach Bussen ergibt sich, weil für das Warten bei den Ein-/Ausgabebefehlen der jeweilige Bus angehalten wird.

1. Die Prozessorstruktur unterscheidet sich in diesem Punkt von T9 und T10: Dort wurden beide Befehlsgruppen sequentiell bearbeitet. Ein Anhalten des Prozessors mit **SLEEP** beeinflusste deshalb auch folgende Ein-/Ausgabebefehle.

Software-Änderung beim Umstieg von T9 / T10

PROCESSDELAY

READ_TIMER

SLEEP



Software

5.2.2 Pro II-Boot

Mit Pro II-Boot, auch Bootloader genannt, steht Ihnen eine Erweiterung zur Verfügung, mit der nach dem Einschalten

- das *ADwin-Pro II*-System gebootet wird.
- bis zu 10 Prozesse geladen werden können.
- Prozess 10 automatisch gestartet wird (falls vorhanden).
- Daten gespeichert werden können.

Pro II-Boot ist eine Bestelloption und kann nicht nachgerüstet werden.

Der Bootloader wird mit dem Programm *ADethflash* (im Windows Startmenü unter *Programs\ADwin*) programmiert. Hinweise zur Bedienung sind im Programm enthalten.

Mit der Installation von *ADbasic* und der *ADwin*-Treiber von der CD-ROM sind bereits die für die Bootloader-Option nötigen Dateien/Programme auf die Festplatte kopiert worden.

Wenn Sie den Bootloader benutzen, darf eine Anwendung, die Sie z. B. zur Visualisierung der Messdaten geschrieben haben, das *ADwin*-System nicht neu booten.



5.2.3 Modul-Überwachung mit Watchdog

Sie können Ihr Prozessormodul mit einem Watchdog überwachen. Der Watchdog ist ein Zähler, der regelmäßig über den Programmcode zurückgesetzt werden muss. Geschieht das nicht, erzeugt der Watchdog ein Reset-Signal (siehe auch *ADwin-Pro II* Systembeschreibung „Programmierung in *ADbasic*“). Dieses Reset-Signal setzt die digitalen und analogen Ausgänge auf diejenigen Werte, die der Konfiguration nach dem Einschalten entsprechen, im Normalfall digital 0 bzw. 0 Volt.

Hinweise im Zusammenhang mit Pro II-Flash-Boot:

- Bitte achten Sie darauf, dass der Watchdog spätestens alle 1,6s zurückgesetzt werden muss. Ein längerer Zeitraum zwischen zwei Impulsen wird ansonsten als Fehler interpretiert.
- Der Watchdog kann auch mit dem Bootloader Pro II-Flash-Boot verwendet werden, sorgt dann aber nicht für das automatische Laden und Starten der Software.
- Testen Sie Ihre Programme immer mit ausgeschaltetem Watchdog. Aktivieren Sie den Watchdog erst, wenn Ihre Programme zuverlässig arbeiten!



5.3 Pro II: Multi-IO-Module

Modulname	Rev.	Typ	Kanäle	Kenndaten	Seite
MIO-4	E	TTL-Ein-/Ausgang	8	U_{Ein} 5V TTL	18
		Analogeingänge	16 s.e. 8 diff.	1 ADC, 18 Bit, $\pm 10\text{V}$ Wandlungszeit max. $2\mu\text{s}$ Abtastrate max. 500 kSample/s	
		Analogausgänge	4	4 DAC, 16 Bit, $\pm 10\text{V}$ Einschwingzeit $9\mu\text{s}$	
		TiCo-Prozessor	–	128 KiByte interner Speicher 4 MiByte externes SRAM	
MIO-4-ET1	E	TTL-Ein-/Ausgang	8	U_{Ein} 5V TTL	25
		Analogeingänge	16 s.e. 8 diff.	1 ADC, 18 Bit, $\pm 10\text{V}$ Wandlungszeit max. $2\mu\text{s}$ Abtastrate max. 500 kSample/s	
		Analogausgänge	4	4 DAC, 16 Bit, $\pm 10\text{V}$ Einschwingzeit $9\mu\text{s}$	
		Transistor-Ausgänge	4	$5\ldots 30 V_{\text{DC}}$ (extern), 200mA	
		Optokoppler-Eingänge	4 s.e.	einstellbar 5V, 12V, 24V	
		Zählerblock	1	Universal, 32 Bit, 5V diff.	
		SSI-Decoder	1	max. 2MHz	
		EtherCAT-Schnittstelle	1	Slave-Schnittstelle	
		TiCo-Prozessor	–	128 KiByte interner Speicher 4 MiByte externes SRAM	

5.3.1 Pro II-MIO-4 Rev. E

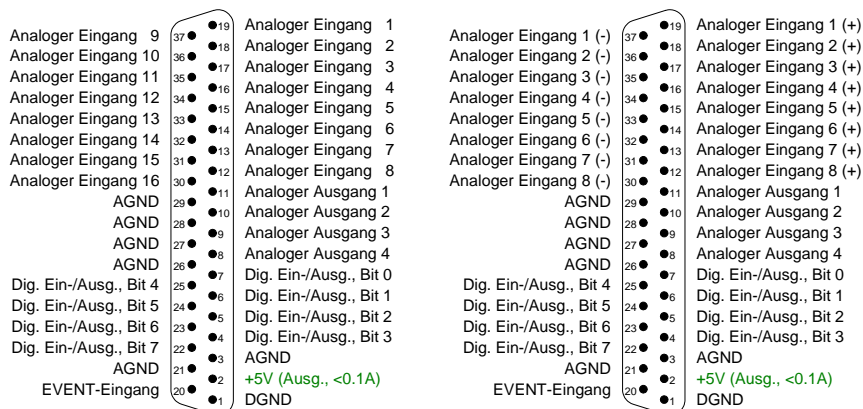
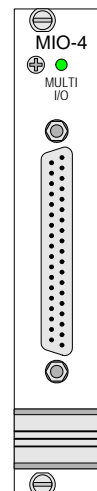
Das Modul Pro II-MIO-4 Rev. E ist mit folgender Hardware ausgerüstet:

- 16 analoge Eingänge (16 single-ended oder 8 differentiell), über Multiplexer mit 18 Bit ADC, Abtastrate bis 500kHz
- 4 analoge Ausgänge mit 16 Bit DAC.
- 8 digitale Ein- und Ausgangskanäle mit TTL-Pegeln
- 1 Event-Eingang
- *TiCo*-Prozessor mit 128 KiByte internem Speicher und 4MiB externem SRAM-Speicher

Der *TiCo*-Prozessor hat Zugriff auf alle Ein- und Ausgänge des Moduls. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCo-Basic*.

Wenn der *TiCo*-Bootloader programmiert ist, kann das Modul eigenständig und unabhängig vom CPU-Modul des *ADwin-Pro II*-Systems arbeiten.

Die erweiterte Modulvariante Pro II-MIO-4-ET1 Rev. E ist ab Seite 25 beschrieben.



Analogeingänge single-ended

Analogeingänge differentiell

Abb. 15 – Pro II-MIO-4 Rev. E: Pinbelegung

Die Funktionen des Moduls sind in folgenden Abschnitten beschrieben:

- Analoge Eingänge
- Analoge Ausgänge
- Digitale Ein-/Ausgänge
- *TiCo*-Prozessor
- Technische Daten
- Programmierung

Analoge Eingänge

Das Modul Pro II-MIO-4 Rev. E hat 16 single ended-Eingänge oder 8 differentiell Eingänge (über Software einstellbar). Nach dem Einschalten ist das Modul auf 8 differentiell Eingänge eingestellt.

Die Eingänge sind auf eine 37-polige D-Sub-Buchse geführt; Pinbelegung siehe Abb. 15.

Die Eingänge sind über einen Multiplexer mit dem ADC verbunden. Der ADC hat eine Auflösung von 18 Bit und kann mit einer Abtastrate von bis zu 500kSamples/s arbeiten.

Das Modul besitzt einen Eingangsspannungsbereich von $\pm 10V$ und eine programmierbare Verstärkung von 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Das Modul beinhaltet eine Ablaufsteuerung, die Messwerte an mehreren oder allen Eingangskanälen nacheinander einlesen kann.

Das Modul kann jeden Eingangskanal daraufhin überwachen, ob ein oberer oder unterer Grenzwert – die Grenzwerte sind für jeden Eingangskanal separat einstellbar – über- oder unterschritten wurde.

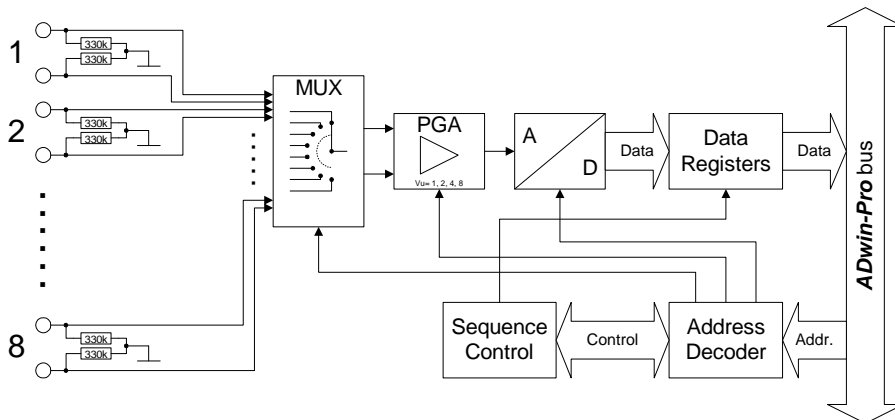


Abb. 16 – Pro II-MIO-4 Rev. E: Blockschaltbild der Analogeingänge

Analoge Ausgänge

Das Modul Pro II-MIO-4 Rev. E hat 4 analoge Ausgänge mit einem DAC zu 16 Bit. Jeder Ausgang hat einen festen Tiefpass 1. Ordnung, um Störungen zu unterdrücken ($f_g = 10MHz$).

Der Ausgangsspannungsbereich der DAC ist fest auf $\pm 10V$ bipolar eingestellt und lässt sich nicht verändern. Der Abgleich des Offset erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Die Ausgänge sind auf die 37-polige D-Sub-Buchse geführt; Pinbelegung siehe Abb. 15.

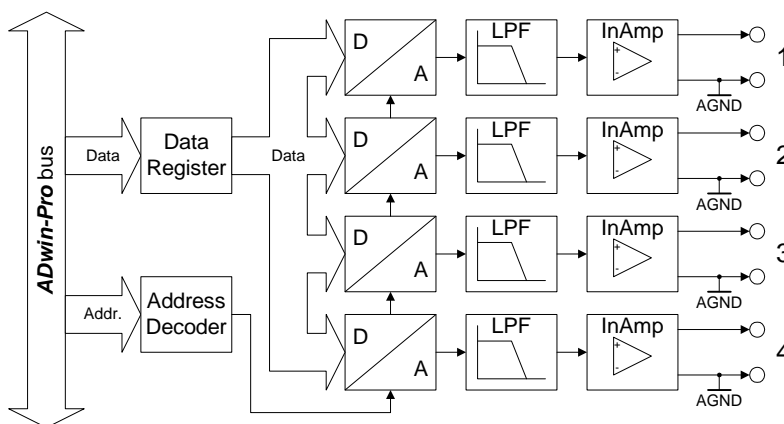


Abb. 17 – Pro II-MIO-4 Rev. E: Blockschaltbild der Analogausgänge

Digitale Ein-/Ausgänge

Das Modul Pro II-MIO-4 Rev. E stellt 8 programmierbare Ein- und Ausgangskanäle mit TTL-Pegeln bereit. Die Kanäle können in Blöcken zu jeweils 4 mit

ADbasic-Befehlen als Ein- oder Ausgänge konfiguriert werden. Nach dem Einschalten sind alle Kanäle als Eingänge konfiguriert.

Über den Trigger-Eingang **EVENT** kann ein Signal (Trigger) einen Prozess auslösen, der dann sofort und vollständig abgearbeitet wird (siehe *ADbasic*-Handbuch).

Die Digitalkanäle sind auf die 37-polige D-Sub-Buchse geführt; Pinbelegung siehe Abb. 15.

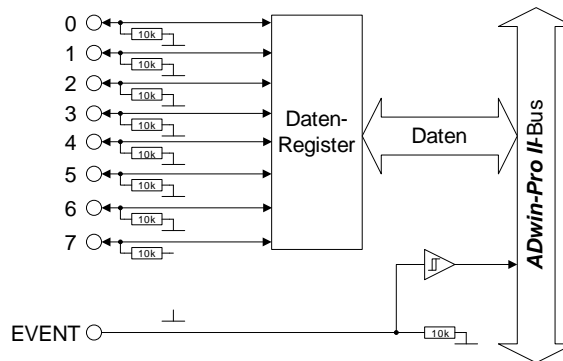


Abb. 18 – Pro II-MIO-4 Rev. E: Blockschaltbild

TiCo-Prozessor

Das Modul besitzt einen frei programmierbaren *TiCo*-Prozessor mit 128KiByte internem Speicher und 4MiByte externem SRAM-Speicher. Der interne Speicher dient je zur Hälfte als Daten- und als Programmspeicher. Den *TiCo*-Prozessor programmieren Sie in *TiCoBasic*.

Der *TiCo*-Prozessor hat – wie auch die *ADwin*-CPU – Zugriff auf alle Ein- und Ausgangskanäle, analog wie digital. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

Wenn Sie ein *TiCoBasic*-Programm im *TiCo*-Bootloader ablegen, wird das Programm beim Einschalten der Stromversorgung in den *TiCo*-Prozessor geladen und gestartet. Auf diese Weise kann das Modul eigenständig und unabhängig vom CPU-Modul des *ADwin-Pro II*-Systems arbeiten.

Technische Daten

Analoge Eingänge	
Eingangskanäle	16 single ended / 8 differentiell, Multiplexer
Auflösung	18 Bit
Wandlungszeit	max. 2µs
Abtastrate	max. 500ksps
Multiplexer Einschwingzeit	5µs
Messbereich	±10V
Verstärkung	1, 2, 4, 8 per Software einstellbar
Genauigkeit INL	typisch ±4 LSB
Genauigkeit DNL	max. ±1 LSB
Eingangswiderstand	330kΩ, ±2%
Spannungsfestigkeit	±20V
Offsetfehler	abgleichbar
Offsetdrift	±30ppm/°C
Analoge Ausgänge	
Ausgangskanäle	4
Auflösung	16 Bit
Einschwingzeit	9µs (auf 0,01% FSR)
Ausgangsspannung	±10V
Max. Ausgangsstrom	±5mA pro Kanal für optimale Funktion
Genauigkeit INL	±2 LSB typisch
Genauigkeit DNL	±1 LSB typisch
Offsetfehler	abgleichbar
Verstärkungsfehler	abgleichbar
Digitale Ein-/Ausgänge	
Digitale Eingänge	8 Kanäle mit TTL-Logik, konfigurierbar in Gruppen zu 4 Kanälen
Pull-Down-Widerstand	10kΩ
V _{IH}	min. 2V
V _{IL}	max. 0,8V
I _{IH}	max. 1µA
I _{IL}	max. 0,01mA
Spannungsbereich	-0,5V ... +5,5V
Ausgangsstrom	max. ±24mA pro Kanal, max. ±50mA je Block (4 Kanäle) über V _{CC} oder GND
Event-Eingang	TTL-Logik
Power-Up-Status	Alle digitalen Kanäle als Eingänge
Allgemein	
Speichergröße (TiCo)	128KiByte intern, 4MiByte externes SRAM
Steckerverbindung	37-polige D-Sub-Buchse

Abb. 19 – Pro II-MIO-4 Rev. E: Spezifikation

Programmierung in ADbasic

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Analoge Eingänge	
Eingänge einstellen auf single-ended oder differentiell	<code>P2_SE_Diff</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADC</code> , <code>P2_ADC24</code> <code>P2_Set_Mux</code> , <code>P2_Start_Conv</code> <code>P2_Wait_EOC</code> <code>P2_Read_ADC</code> , <code>P2_Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv24</code>
Ablaufsteuerung anwenden	<code>P2_Seq_Init</code> , <code>P2_Seq_Start</code> <code>P2_Seq_Read</code> , <code>P2_Seq_Read24</code> <code>P2_SEQ_Read_Packed</code> <code>P2_Seq_Wait</code>
Grenzwerte überwachen	<code>P2_ADC_Read_Limit</code> <code>P2_ADC_Set_Limit</code>
Analoge Ausgänge	
Ausgabe durchführen	<code>P2_DAC</code> , <code>P2_DAC4</code> , <code>P2_DAC4_Packed</code>
Ausgabe schrittweise durchführen	<code>P2_Write_DAC</code> , <code>P2_Write_DAC4</code> <code>P2_Write_DAC4_Packed</code> <code>P2_Write_DAC32</code> <code>P2_Start_DAC</code>
Digitale Ein-/Ausgänge	
Ein- und Ausgänge konfigurieren	<code>P2_MIO_DigProg</code>
Eingangssignale abfragen	<code>P2_MIO_Digin_Long</code>
Latch-Register nutzen	<code>P2_MIO_Dig_Latch</code> <code>P2_MIO_Dig_Read_Latch</code> <code>P2_MIO_Dig_Write_Latch</code>
Ausgangssignale setzen und rücklesen	<code>P2_MIO_Digout</code> <code>P2_MIO_Digout_Long</code> <code>P2_MIO_Get_Digout_Long</code>
Allgemein	
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Abläufe synchronisieren	<code>P2_Sync_All</code> , <code>P2_Sync_Enable</code> <code>P2_Sync_Stat</code>
Interrupts und Event-Eingang einstellen	<code>P2_eVENT_eENABLE</code> , <code>P2_eVENT_rEAD</code> <code>P2_eVENT_cONFIG</code>

Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe erläutert.

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Die Include-Datei `MIO_TiCo.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Analoge Eingänge	
Eingänge einstellen auf single-ended oder differentiell	<code>SE_Diff</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>ADC, ADC24</code> <code>Set_Mux, Start_Conv, Wait_EOC</code> <code>Read_ADC, Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>Read_ADC_SConv</code> <code>Read_ADC_SConv24</code>
Ablaufsteuerung anwenden	<code>Seq_Init, Seq_Start</code> <code>Seq_Read, Seq_Read24</code> <code>Seq_Wait</code>
Grenzwerte überwachen	<code>ADC_Read_Limit</code> <code>ADC_Set_Limit</code>
Analoge Ausgänge	
Ausgabe durchführen	<code>DAC</code>
Ausgabe schrittweise durchführen	<code>Write_DAC, Write_DAC32</code> <code>Start_DAC</code>
Digitale Ein-/Ausgänge	
Ein- und Ausgänge konfigurieren	<code>MIO_DigProg</code>
Eingangssignale abfragen	<code>MIO_Digin_Long</code>
Latch-Register nutzen	<code>MIO_Dig_Latch</code> <code>MIO_Dig_Read_Latch</code> <code>MIO_Dig_Write_Latch</code>
Ausgangssignale setzen und rücklesen	<code>MIO_Digout</code> <code>MIO_Digout_Long</code> <code>MIO_Get_Digout_Long</code>
Allgemein	
LEDs einstellen	<code>Check_LED, Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>Event_Enable, Event_Read</code> <code>Event_Config, Trigger_Event</code>

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei `ADwinPro_All.inc` definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch über globale Variablen	<code>P2_TDrv_Init</code> <code>P2_GetData_Long, P2_Get_Par,</code> <code>P2_Get_Par_Block</code> <code>P2_SetData_Long, P2_Set_Par,</code> <code>P2_Set_Par_Block</code>
Datenaustausch über globale Ringbuffer	<code>P2_Get_TiCo_RingBuffer,</code> <code>P2_Set_TiCo_RingBuffer</code> <code>P2_RingBuffer_Empty</code> <code>P2_RingBuffer_Full</code>

Programmierung in TiCoBasic

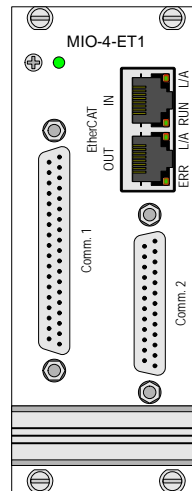
Programmierung TiCo-Zugriff

Bereich	Befehle
TiCo-Prozessor steuern	P2_TiCo_Reset, P2_TiCo_Start, P2_TiCo_Stop P2_Get_TiCo_Bootloader_ Status P2_Get_TiCo_Status, P2_Workload
TiCo-Prozesse steuern	P2_Process_Status P2_TiCo_Get_Processdelay P2_TiCo_Set_Processdelay P2_TiCo_Start_Process P2_TiCo_Stop_Process
TiCo-Programme übertragen	P2_TiCo_Flash, P2_TiCo_Load

5.3.2 Pro II-MIO-4-ET1 Rev. E

Das Modul Pro II-MIO-4-ET1 Rev. E ist mit folgender Hardware ausgerüstet:

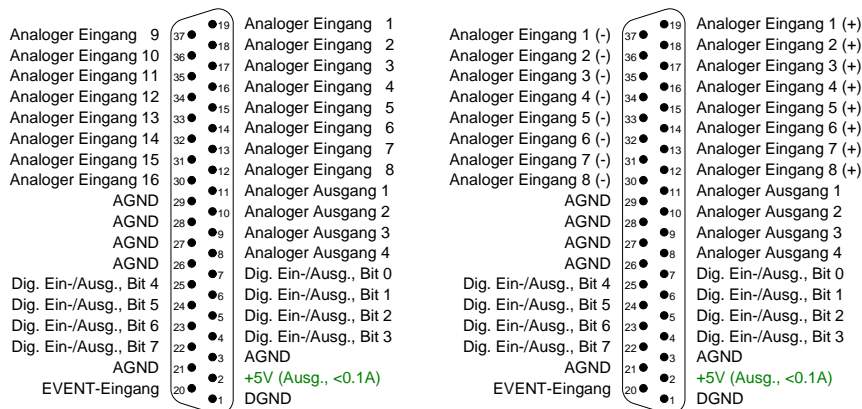
- 16 analoge Eingänge (16 single-ended oder 8 differentiell), über Multiplexer mit 18 Bit ADC, Abtastrate bis 500kHz
- 4 analoge Ausgänge mit 16 Bit DAC.
- 8 digitale Ein- und Ausgangskanäle mit TTL-Pegeln
- 1 Event-Eingang
- 4 Transistor-Ausgänge (TRA)
- 4 optisch isolierte Eingänge (OPT)
- 1 Zählerblock mit zwei 32 Bit-Zählern:
 - Ein Vor-/ Rückwärtszähler mit Takt/Richtungs-Auswertung oder mit Vierflankenauswertung zum Anschluss von Encodern.
 - Ein Zähler zur Periodendauer- und Tastverhältnismessung.
- 1 SSI-Decoder zum Anschluss eines Inkremental-Encoders
- 1 Ethercat-Schnittstelle (Slave)
- *TiCo*-Prozessor mit 128 KiByte internem Speicher und 4MiB externem SRAM-Speicher



Der *TiCo*-Prozessor hat Zugriff auf alle Ein- und Ausgänge des Moduls. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

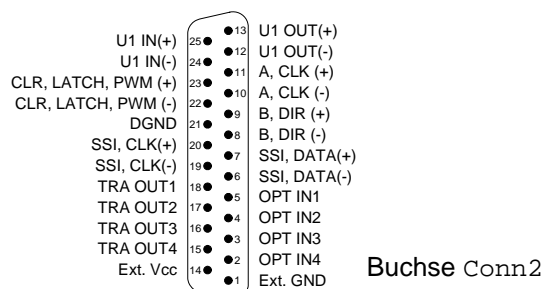
Wenn der *TiCo*-Bootloader programmiert ist, kann das Modul eigenständig und unabhängig vom CPU-Modul des *ADwin-Pro II*-Systems arbeiten.

Die einfachere Modulvariante Pro II-MIO-4 Rev. E ist ab Seite 18 beschrieben.



Buchse Conn1, single-ended

Buchse Conn1, differentiell



Buchse Conn2

Abb. 20 – Pro II-MIO-4-ET1 Rev. E: Pinbelegungen

Die Funktionen des Moduls sind in folgenden Abschnitten beschrieben:

- Analoge Eingänge
- Analoge Ausgänge
- Digitale Ein-/Ausgänge
- Transistor-Ausgänge
- Optokoppler-Eingänge
- SSI-Decoder
- Zählerblock
- EtherCAT-Schnittstelle
- TiCo-Prozessor
- Technische Daten
- Programmierung

Analoge Eingänge

Das Modul Pro II-MIO-4-ET1 Rev. E hat 16 single ended-Eingänge oder 8 differentielle Eingänge (über Software einstellbar). Nach dem Einschalten ist das Modul auf 8 differentielle Eingänge eingestellt.

Die Eingänge sind auf eine 37-polige D-Sub-Buchse Conn1 geführt; Pinbelegung siehe Abb. 20.

Die Eingänge sind über einen Multiplexer mit dem ADC verbunden. Der ADC hat eine Auflösung von 18 Bit und kann mit einer Abtastrate von bis zu 500kSamples/s arbeiten.

Das Modul besitzt einen Eingangs-Spannungsbereich von $\pm 10V$ und eine programmierbare Verstärkung von 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Das Modul beinhaltet eine Ablaufsteuerung, die Messwerte an mehreren oder allen Eingangskanälen nacheinander einlesen kann.

Das Modul kann jeden Eingangskanal daraufhin überwachen, ob ein oberer oder unterer Grenzwert – die Grenzwerte sind für jeden Eingangskanal separat einstellbar – über- oder unterschritten wurde.

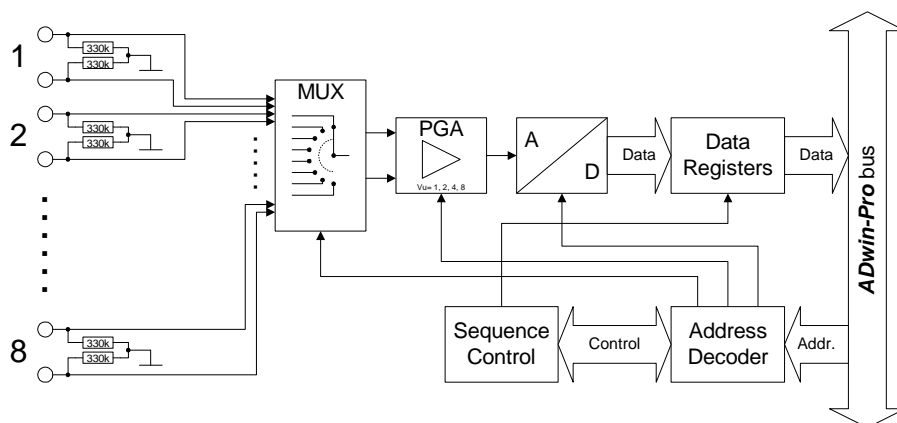


Abb. 21 – Pro II-MIO-4-ET1 Rev. E: Blockschaltbild Analogeingänge

Analoge Ausgänge

Das Modul Pro II-MIO-4-ET1 Rev. E hat 4 analoge Ausgänge mit einem DAC zu 16 Bit. Jeder Ausgang hat einen festen Tiefpass 1. Ordnung, um Störungen zu unterdrücken ($f_g = 10MHz$).

Der Ausgangs-Spannungsbereich der DAC ist fest auf $\pm 10V$ bipolar eingestellt und lässt sich nicht verändern. Der Abgleich des Offset erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Die Ausgänge sind auf die 37-polige D-Sub-Buchse Conn1 geführt; Pinbelegung siehe Abb. 20.

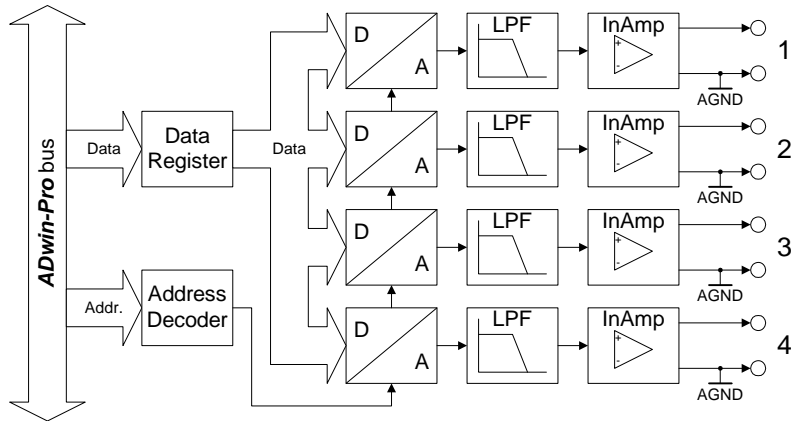


Abb. 22 – Pro II-MIO-4-ET1 Rev. E: Blockschaltbild Analogausgänge

Digitale Ein-/Ausgänge

Das Modul Pro II-MIO-4-ET1 Rev. E stellt 8 programmierbare Ein- und Ausgangskanäle mit TTL-Pegeln bereit. Die Kanäle können in Blöcken zu jeweils 4 mit *ADbasic*-Befehlen als Ein- oder Ausgänge konfiguriert werden. Nach dem Einschalten sind alle Kanäle als Eingänge konfiguriert.

Über den Trigger-Eingang EVENT kann ein Signal (Trigger) einen Prozess auslösen, der dann sofort und vollständig abgearbeitet wird (siehe *ADbasic*-Handbuch).

Die Digitalkanäle sind auf die 37-polige D-Sub-Buchse Conn1 geführt; Pinbelegung siehe Abb. 20.

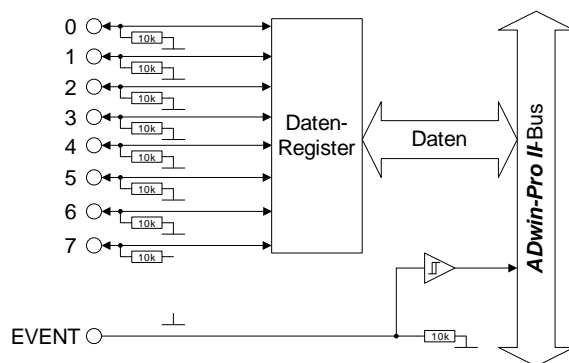


Abb. 23 – Pro II-MIO-4-ET1 Rev. E: Blockschaltbild Digitalkanäle

Transistor-Ausgänge

Das Modul Pro II-MIO-4-ET1 Rev. E stellt 4 galvanisch getrennte Transistor-Schaltausgänge bereit. Die Ausgänge schalten nach V_{CC} .

Die Schaltspannung V_{CC} muss durch eine externe Spannungsversorgung zugeführt werden.

Die Kanäle wie auch der Event-Eingang sind optisch vom System-Stromkreis isoliert.

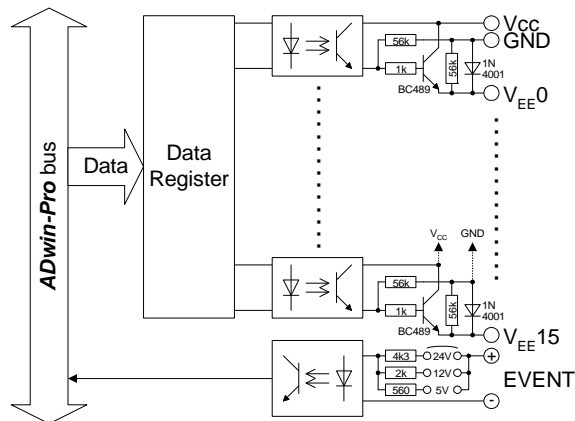


Abb. 24 – Pro II-MIO-4-ET1 Rev. E: Blockschaltbild Transistor-Ausgänge

Die Ausgänge sind auf die 25-polige D-Sub-Buchse Conn2 geführt; Pinbelegung siehe Abb. 20.

Optokoppler-Eingänge

Das Modul Pro II-MIO-4-ET1 Rev. E stellt 4 Kanäle mit optisch isolierten digitalen Eingängen bereit. Die Eingangs-Spannungsbereiche sind für jeden Eingang separat über Jumper einstellbar (5V, 12V, 24V). Die Voreinstellung ist 24V. Die Schaltzeit von nur 100ns erlaubt das Einlesen von schnellen digitalen Signalen.

Das Modul kann automatisch die Flanken an Eingangskanälen überwachen. Bei einer Änderung wird der aktuelle Pegelstand gemeinsam mit einem Zeitstempel in einem FIFO zwischengespeichert. Die FIFO-Daten können ausgelesen und weiter verarbeitet werden.

Jeder Kanal ist vom Systemstromkreis und von den anderen Eingängen optisch isoliert, wie auch der Event-Eingang.

Die Eingänge sind auf die 25-polige D-Sub-Buchse Conn2 geführt; Pinbelegung siehe Abb. 20.

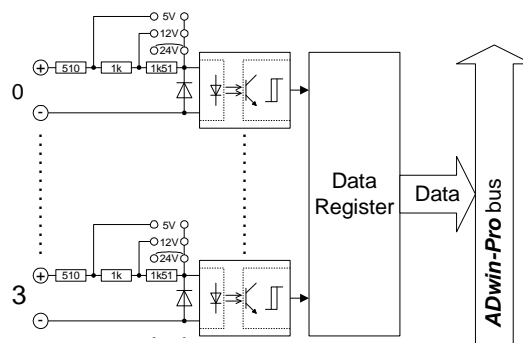


Abb. 25 – Pro II-MIO-4-ET1 Rev. E: Blockschaltbild Optokoppler-Eingänge

SSI-Decoder

An den SSI-Decoder kann ein Inkremental-Encoder mit SSI-Schnittstelle angeschlossen werden. Die Signale sind differentiell und haben RS422/485-Pegel.

Programmierbar sind die Taktraten über einen Vor-Teiler (von ca. 40kHz bis 1MHz) ebenso wie die Auflösung des Decoders (bis 32 Bit).

Die Decoder-Eingänge sind auf die 25-polige D-Sub-Buchse Conn2 geführt; Pinbelegung siehe Abb. 20.

Zählerblock

Das Modul Pro II-MIO-4-ET1 Rev. E stellt einen konfigurierbaren Zählerblock zur Verfügung. Der Zählerblock enthält zwei 32 Bit-Zähler: Einerseits einen Vor-/ Rückwärtszähler mit Takt/Richtungs-Auswertung oder Vierflankenauswertung zum Anschluss von Encodern. Zum anderen einen Zähler zur Periodendauer- und Tastverhältnismessung. Beide Zähler können parallel genutzt werden.

Die Zählereingänge arbeiten differentiell.

Der Vor-/ Rückwärtszähler kann in 2 Betriebsarten arbeiten:

- Takt/Richtungs-Auswertung (CLK- und DIR-Signale)

Eine negative Flanke am CLK-Eingang löst einen Zählimpuls am 32 Bit-Zähler aus. Das DIR-Signal bestimmt die Zählrichtung des Zählers, TTL high bedeutet Hochzählen, TTL low bedeutet Herunterzählen.

Sie können den Zählerstand programmgesteuert ins Latch übernehmen oder den Zähler durch ein externes CLR-/LATCH-Signal beeinflussen.

Das CLR-/LATCH-Signal kann je nach Programmierung ein Löschen (CLR) des Zählerstands oder die Übernahme des Zählerstands ins Latch (LATCH) bewirken. Diese Funktion wird erst wirksam, wenn sie durch den Befehl **P2_CNT_CLEAR_ENABLE** oder **P2_CNT_LATCH_ENABLE** freigegeben ist.

Das Löschen oder Latchen des Zählers erfolgt bei einer steigenden Flanke am Eingang CLR/LATCH. Beim Latchen lässt sich aus der Differenz von zwei gelesenen Latch-Werten die Frequenz der Messung ermitteln, denn die Differenz gibt die Anzahl der Impulse zwischen den beiden Lesevorgängen an.

- Vierflankenauswertung (A- und B-Signale)

Die Vierflankenauswertung wandelt die (möglichst um 90° phasenverschobenen) Signale eines angeschlossenen Inkremental-Encoders an A- und B-Eingang in ein CLK- und DIR-Signal um. Hierzu sind die Eingänge in *ADbasic* entsprechend zu programmieren (siehe „*ADwin-Pro* Systembeschreibung, Programmierung in *ADbasic*“).

Da jede Flanke des A- und B-Signales einen Zählimpuls erzeugt, wird die Auflösung um den Faktor 4 vergrößert. Besitzt der Encoder ein Referenz-Signal, so kann dies (nach Freigabe des CLR- bzw. LATCH-Einganges) zum Löschen oder Latchen des Zählers genutzt werden. Das Löschen des Zählers erfolgt, wenn die Signale A, B und CLR auf logisch „1“ stehen (über Software umstellbar: Löschen, wenn nur das CLR-Signal auf logisch „1“ steht).

Der PWM-Zähler des Zählerblocks wertet die Signale an den PWM-Eingängen aus. Mit Software-Befehlen können folgende Daten direkt ausgelesen werden:

- Frequenz und Tastverhältnis
- Eintastzeit und Austastzeit

Die Zähler-Eingänge sind auf die 25-polige D-Sub-Buchse Conn2 geführt; Pinbelegung siehe Abb. 20.

EtherCAT-Schnittstelle

Das Modul stellt einen Feldbusknoten mit der Funktionalität eines EtherCAT-Slave zur Verfügung. Alle Einstellungen werden per Software vorgenommen.

Nach dem Einschalten müssen Sie den Feldbusknoten in *ADbasic* initialisieren. Mit der Initialisierung wird die Größe der Ein- und Ausgangsbereiche festgelegt.

Vor-/Rückwärtszähler

PWM-Zähler

Es gibt je einen Bereich für eingehende und für ausgehende Daten; jeder Bereich hat eine maximale Größe von 254 Byte. Die Begriffe „Eingang“ und „Ausgang“ sind aus Sicht des Feldbus-Controllers zu sehen.

Sie legen für beide Bereiche separat fest, wie viele Datenbereiche in jedem Bereich vorhanden sind und wie groß die Datenbereiche sind.

Die Schnittstelle hat je eine Buchse vom Typ RJ45 für den Dateneingang (IN) und für den Datenausgang (OUT). An jeder Buchse ist oben eine LED „Link / Activity“ (L/A), die den Betriebszustand des Knotens im EtherCAT-Bus anzeigt. Die beiden weiteren LEDs zeigen den Status der EtherCAT-Zustandsmaschine (RUN) und das Auftreten von Kommunikationsfehlern (ERR) an.

LED	Status	Bedeutung
Link / Acti- vity	Aus	Nicht online (oder keine Stromversorgung).
	An	Feldbusknoten online, kein Datenaustausch.
	flackernd	Feldbusknoten online, mit Datenaustausch.
RUN	Aus	Status INIT: Die Schnittstelle wird initialisiert (oder keine Stromversorgung).
	blinkt grün	Status PRE-OP: Schnittstelle hat Kontakt zum Bus-Master.
	leuchtet einmal grün	Status SAFE-OP: Schnittstelle kann Daten vom Bus lesen, aber nicht senden.
	leuchtet grün	Status OP: Schnittstelle ist vollständig eingerichtet, Ein- und Ausgänge sind aktiv.
	leuchtet rot	Status EXCEPTION: Ausnahmesituation.
ERR	Aus	Kommunikation arbeitet ohne Fehler (oder keine Stromversorgung).
	blinkt rot	Fehler bei der Konfiguration.
	leuchtet einmal rot	Lokaler Fehler in der Schnittstelle; der EtherCAT-Status wurde geändert.
	leuchtet doppelt rot	Fehler durch Zeitüberschreitung (timeout).
	leuchtet rot	Kritischer Kommunikationsfehler.

Abb. 26 – Pro II-MIO-4-ET1 Rev. E: Bedeutung der EtherCAT-LED

Wenn beide LEDs RUN und ERR rot leuchten, ist ein gravierender Fehler in der Schnittstelle aufgetreten. Melden Sie sich dann bitte beim Support von Jäger Messtechnik; die Adresse finden Sie auf der vorderen Umschlagseite des Handbuchs, innen.

Der Zugriff auf den ADwin-EtherCAT-Slave ist nur in *TiCoBasic* möglich. Sie konfigurieren den Slave mit dem Befehl **MIO_ECATT_Init**. Anschließend projektieren Sie den EtherCAT-Bus mit einem – zum Bus-Master passenden – Konfigurations-Tool, beispielsweise mit dem Programm „TwinCAT System Manager“ der Firma Beckhoff als EtherCAT-Master. Sie müssen dazu die Beschreibungsdatei `ADwin-EtherCAT.xml` des EtherCAT-Moduls von `C:\ADwin\Feldbus\EtherCAT\` in das Verzeichnis des Konfigurations-Tools kopieren.

TiCo-Prozessor

Das Modul besitzt einen frei programmierbaren *TiCo*-Prozessor mit 128KiByte internem Speicher und 4MiByte externem SRAM-Speicher. Der interne Speicher dient je zur Hälfte als Daten- und als Programmspeicher. Den *TiCo*-Prozessor programmieren Sie in *TiCoBasic*.

Der *TiCo*-Prozessor hat – wie auch die *ADwin*-CPU – Zugriff auf alle Ein- und Ausgangskanäle, analog wie digital. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

Wenn Sie ein *TiCoBasic*-Programm im *TiCo*-Bootloader ablegen, wird das Programm beim Einschalten der Stromversorgung in den *TiCo*-Prozessor geladen und gestartet. Auf diese Weise kann das Modul eigenständig und unabhängig vom CPU-Modul des *ADwin-Pro II*-Systems arbeiten.

Technische Daten

Analoge Eingänge	
Eingangskanäle	16 single ended / 8 differentiell, Multiplexer
Auflösung	18 Bit
Wandlungszeit	max. 2µs
Abtastrate	max. 500kps
Multiplexer Einschwingzeit	5µs
Messbereich	±10V
Verstärkung	1, 2, 4, 8 per Software einstellbar
Genauigkeit INL	typisch ±4 LSB
Genauigkeit DNL	max. ±1 LSB
Eingangswiderstand	330kΩ, ±2%
Spannungsfestigkeit	±20V
Offsetfehler	abgleichbar
Offsetdrift	±30ppm/°C
Analoge Ausgänge	
Ausgangskanäle	4
Auflösung	16 Bit
Einschwingzeit	9µs (auf 0,01% FSR)
Ausgangsspannung	±10V
Max. Ausgangsstrom	±5mA pro Kanal für optimale Funktion
Genauigkeit INL	±2 LSB typisch
Genauigkeit DNL	±1 LSB typisch
Offsetfehler	abgleichbar
Verstärkungsfehler	abgleichbar
Digitale Ein-/Ausgänge	
Digitale Eingänge	8 Kanäle mit TTL-Logik, konfigurierbar in Gruppen zu 4 Kanälen
Pull-Down-Widerstand	10kΩ
V _{IH}	min. 2V
V _{IL}	max. 0,8V
I _{IH}	max. 1µA
I _{IL}	max. 0,01mA
Spannungsbereich	-0,5V ... +5,5V
Ausgangsstrom	max. ±24mA pro Kanal, max. ±50mA je Block (4 Kanäle) über V _{CC} oder GND

Abb. 27 – Pro II-MIO-4-ET1 Rev. E: Spezifikation

Event-Eingang	TTL-Logik
Power-Up-Status	Alle digitalen Kanäle als Eingänge
Transistor-Ausgänge	
Ausgangskanäle	4
Schaltspannung V_{CC}	5...30V DC durch externe Spannungsversorgung
Schaltstrom	200mA max. pro Kanal
Spannungsabfall	0,5V
Schaltzeit	2,5µs
Isolation	42V Kanal zu Kanal / Kanal zu Masse gemeinsame Masse für alle OPT- und TRA-Kanäle
Optokoppler-Eingänge	
Eingangskanäle	4
Eingangsstrom	typ. 3,5mA / max. 7,5mA
Eingangs-Spannungsbereich (über Jumper wählbar)	0...5V / 0...12V / 0...24V
Schaltschwelle für 0-low	0...0,8V / 0...1,6V / 0...3,2V
Schaltschwelle für 1-high	4,5...5V / 10...12V / 20...24V
Spannungsfestigkeit	-5V ... 8V / -5V ... 16V / -5V ... 30V /
Schaltzeit	100ns
Isolation	42V Kanal zu Kanal / Kanal zu Masse gemeinsame Masse für alle OPT- und TRA-Kanäle
Zähler	
Anzahl	1 Universalzähler
Zählerbreite	32 Bit
Referenztakt	50MHz
Taktfrequenz Vierflankenauswertung	12,5MHz max. (bei 90° Phasenverschiebung der Signale)
Taktfrequenz Vor- / Rückwärtszähler	15MHz max.
Referenzfrequenz PWM-Analyse	100MHz
Eingangspegel	RS422/485 kompatibel (5V differentiell, 120 Ω Bus-Abschlusswiderstand)
Isolation	keine
SSI-Decoder	
Anzahl	1
Taktfrequenz SSI-Decoder (CLK)	2MHz max.
Ethercat-Schnittstelle	
Anzahl	1
Allgemein	
Speichergröße (TiCo)	128KiByte intern, 4MiByte externes SRAM

Abb. 27 – Pro II-MIO-4-ET1 Rev. E: Spezifikation

Steckerverbindungen	37-polige D-Sub-Buchse Conn1 25-polige D-Sub-Buchse Conn2 2 Buchsen Typ RJ45 für EtherCAT-Dateneingang (IN) und -Datenausgang (OUT)
---------------------	---

Abb. 27 – Pro II-MIO-4-ET1 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Analoge Eingänge	
Eingänge einstellen auf single-ended oder differentiell	<code>P2_SE_Diff</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADC</code> , <code>P2_ADC24</code> <code>P2_Set_Mux</code> , <code>P2_Start_Conv</code> <code>P2_Wait_EOC</code> <code>P2_Read_ADC</code> , <code>P2_Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv24</code>
Ablaufsteuerung anwenden	<code>P2_Seq_Init</code> , <code>P2_Seq_Start</code> <code>P2_Seq_Read</code> , <code>P2_Seq_Read24</code> <code>P2_SEQ_Read_Packed</code> <code>P2_Seq_Wait</code>
Grenzwerte überwachen	<code>P2_ADC_Read_Limit</code> <code>P2_ADC_Set_Limit</code>
Analoge Ausgänge	
Ausgabe durchführen	<code>P2_DAC</code> , <code>P2_DAC4</code> , <code>P2_DAC4_Packed</code>
Ausgabe schrittweise durchführen	<code>P2_Write_DAC</code> , <code>P2_Write_DAC4</code> <code>P2_Write_DAC4_Packed</code> <code>P2_Write_DAC32</code> <code>P2_Start_DAC</code>
Digitale Ein-/Ausgänge	
Ein- und Ausgänge konfigurieren	<code>P2_MIO_DigProg</code>
Eingangssignale abfragen	<code>P2_MIO_Digin_Long</code>
Latch-Register nutzen	<code>P2_MIO_Dig_Latch</code> <code>P2_MIO_Dig_Read_Latch</code> <code>P2_MIO_Dig_Write_Latch</code>
Ausgangssignale setzen und rücklesen	<code>P2_MIO_Digout</code> <code>P2_MIO_Digout_Long</code> <code>P2_MIO_Get_Digout_Long</code>
Transistor-Ausgänge	
Ausgangssignale setzen und rücklesen	<code>P2_Digout</code> , <code>P2_Digout_Long</code> <code>P2_Digout_Bits</code> <code>P2_Get_Digout_Long</code>
Latch-Register nutzen	<code>P2_Dig_Latch</code> , <code>P2_Sync_All</code> <code>P2_Dig_Write_Latch</code>
Optokoppler-Eingänge	

Programmierung in ADbasic

Bereich	Befehle
Eingangssignale abfragen	P2_Digin_Edge P2_Digin_Long
Latch-Register nutzen	P2_Dig_Latch, P2_Sync_All P2_Dig_Read_Latch
Flanken an Eingangskanälen überwachen	P2_Digin_FIFO_Enable P2_Digin_FIFO_Read P2_Digin_FIFO_Read_Timer P2_Digin_FIFO_Clear P2_Digin_FIFO_Full
Zählerblock	
Zähler konfigurieren	P2_Cnt_Enable, P2_Cnt_Mode
Zähler ansteuern	P2_Cnt_Clear, P2_Cnt_Get_Status P2_Cnt_Latch P2_Cnt_Read, P2_Cnt_Read4 P2_Cnt_Read_Latch P2_Cnt_Read_Latch4
PWM-Zähler ansteuern	P2_Cnt_PW_Enable, P2_Cnt_PW_Latch P2_Cnt_Get_PW, P2_Cnt_Get_PW_HL
SSI-Decoder	
SSI-Decoder ansteuern	P2_SSI_Mode, P2_SSI_Set_Bits P2_SSI_Set_Clock P2_SSI_Set_Delay, P2_SSI_Read P2_SSI_Read2 P2_SSI_Start, P2_SSI_Status
EtherCat-Schnittstelle	
Schnittstelle ansteuern	nur in <i>TiCoBasic</i>
Allgemein	
LEDs einstellen	P2_Check_LED, P2_Set_LED
Abläufe synchronisieren	P2_Sync_All, P2_Sync_Enable P2_Sync_Stat
Interrupts und Event-Eingang einstellen	P2_eVENT_eENABLE, P2_eVENT_rEAD P2_eVENT_cONFIG

Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe erläutert.

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Die Include-Datei `MIO_TiCo.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Analoge Eingänge	
Eingänge einstellen auf single-ended oder differentiell	SE_Diff
Einzelmessung durchführen – vollständig oder schrittweise	ADC, ADC24 Set_Mux, Start_Conv, Wait_EOC Read_ADC, Read_ADC24
Messwert lesen und neue Messung starten	Read_ADC_SConv Read_ADC_SConv24
Ablaufsteuerung anwenden	Seq_Init, Seq_Start Seq_Read, Seq_Read24 Seq_Wait
Grenzwerte überwachen	ADC_Read_Limit ADC_Set_Limit
Analoge Ausgänge	
Ausgabe durchführen	DAC
Ausgabe schrittweise durchführen	Write_DAC, Write_DAC32 Start_DAC
Digitale Ein-/Ausgänge	
Ein- und Ausgänge konfigurieren	MIO_DigProg
Eingangssignale abfragen	MIO_Digin_Long
Latch-Register nutzen	MIO_Dig_Latch MIO_Dig_Read_Latch MIO_Dig_Write_Latch
Ausgangssignale setzen und rücklesen	MIO_Digout MIO_Digout_Long MIO_Get_Digout_Long
Transistor-Ausgänge	
Ausgangssignale setzen und rücklesen	MIO_Digout, MIO_Digout_Long MIO_Get_Digout_Long
Latch-Register nutzen	MIO_Dig_Latch MIO_Dig_Write_Latch
Optokoppler-Eingänge	
Eingangssignale abfragen	MIO_Digin_Long
Latch-Register nutzen	MIO_Dig_Latch MIO_Dig_Read_Latch
Zählerblock	
Zähler konfigurieren	Cnt_Enable, Cnt_Mode
Zähler ansteuern	Cnt_Clear, Cnt_Get_Status Cnt_Latch, Cnt_Read Cnt_Read_Int_Register Cnt_Sync_Latch, Cnt_Read_Latch
PWM-Zähler ansteuern	Cnt_PW_Enable, Cnt_PW_Latch Cnt_Get_PW_HL
SSI-Decoder	

Programmierung in TiCoBasic

Programmierung TiCo-Zugriff

Bereich	Befehle
SSI-Decoder ansteuern	<code>SSI_Mode</code> , <code>SSI_Set_Bits</code> <code>SSI_Set_Clock</code> <code>SSI_Set_Delay</code> , <code>SSI_Read</code> <code>SSI_Start</code> , <code>SSI_Status</code>

EtherCat-Schnittstelle

Schnittstelle ansteuern	<code>MIO_ECAT_Init</code> <code>MIO_ECAT_Get_Version</code> <code>MIO_ECAT_Check_State_Transition</code> <code>MIO_ECAT_Read_Data_16L</code> <code>MIO_ECAT_Write_Data_16L</code>
-------------------------	--

Allgemein

LEDs einstellen	<code>Check_LED</code> , <code>Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>eVENT_eENABLE</code> , <code>eVENT_rEAD</code> <code>eVENT_cONFIG</code> , <code>Trigger_Event</code>

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei `ADwinPro_All.inc` definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch über globale Variablen	<code>P2_TDrv_Init</code> <code>P2_GetData_Long</code> , <code>P2_Get_Par</code> , <code>P2_Get_Par_Block</code> <code>P2_SetData_Long</code> , <code>P2_Set_Par</code> , <code>P2_Set_Par_Block</code>
Datenaustausch über globale Ringbuffer	<code>P2_Get_TiCo_RingBuffer</code> , <code>P2_Set_TiCo_RingBuffer</code> <code>P2_RingBuffer_Empty</code> <code>P2_RingBuffer_Full</code>
<i>TiCo</i> -Prozessor steuern	<code>P2_TiCo_Reset</code> , <code>P2_TiCo_Start</code> , <code>P2_TiCo_Stop</code> <code>P2_Get_TiCo_Bootloader_Status</code> <code>P2_Get_TiCo_Status</code> , <code>P2_Workload</code>
<i>TiCo</i> -Prozesse steuern	<code>P2_Process_Status</code> <code>P2_TiCo_Get_Processdelay</code> <code>P2_TiCo_Set_Processdelay</code> <code>P2_TiCo_Start_Process</code> <code>P2_TiCo_Stop_Process</code>
<i>TiCo</i> -Programme übertragen	<code>P2_TiCo_Flash</code> , <code>P2_TiCo_Load</code>

5.4 Pro II: Analoge Eingangsmodule

Dieser Abschnitt beschreibt analoge Eingangsmodule für *ADwin-Pro II*.

Sie finden analoge Eingangsmodule für *ADwin-Pro I* im Handbuch „ADwin-Pro-Hardware ab Seite 17.

Modulname	Aln 8/18-L2	Aln 8/18-L2- TiCo	Aln 32/18-D	Aln 32/18-D- TiCo	Aln 8/18-8B	Aln 16/18-8B
Revision	E	E	E	E	E	E
Anzahl ADC	1	1	1	1	1	1
Auflösung [Bit]	18	18	18	18	18	18
max. Wandlungs- zeit [µs]	2	2	2	2	2	2
max. Abtastrate [ksample/s]	500	500	500	500	500	500
Kanäle diff.	8	8	16	16	8	–
Kanäle s.e.	–	–	32	32	–	–
Kanäle 8B	–	–	–	–	8	16
Messbereich	±10V					
Verstärkung	1, 2, 4, 8					
Kalibrierung	per Software					
TiCo-Prozessor	–	1	–	1	–	–
Seite	38	38	41	41	44	46

Modulname	Aln F-4/14	Aln F-8/14	Aln F-4/16	Aln F-8/16	Aln F-4/18	Aln F-8/18
Revision	E	E	E	E	E	E
Anzahl ADC	4	8	4	8	4	8
Auflösung [Bit]	14	14	16	16	18	18
max. Wandlungs- zeit [µs]	0,02	0,02	0,25	0,25	2	2
max. Abtastrate [ksample/s]	50000	4×50000 8×25000	4000	4000	500	500
Kanäle diff.	4	8	4	8	4	8
Messbereich	±10V					
Verstärkung	1		1, 2, 4, 8		1	
Kalibrierung	per Software					
Seite	48	52	56	60	64	66

Hinweis zur Eingangsverschaltung

Offene Eingänge können zu Fehlern führen, vor allem in einer nicht störungs-freien Umgebung. Sie vermeiden offene Eingänge folgendermaßen:

- Trennen Sie nicht benutzte Eingänge von offenen Leitungen.
- Legen Sie nicht benutzte Eingänge auf einen definierten Pegel (z.B. GND). Der Anschluss sollte möglichst nah an Stecker oder Buchse des Moduls liegen.



5.4.1 Pro II-Aln-8/18-L2 Rev. E

Analoges Eingangsmodul Pro II-Aln-8/18-L2 Rev. E mit einem 18 Bit-ADC, 8 differentiellen Eingängen und einem programmierbaren Verstärker (PGA). Das Modul kann mit Verstärkern, Pro-TC und Pro-PT-Modulen kombiniert werden.

Die Variante Pro II-Aln-8/18-L2-TiCo Rev. E besitzt zusätzlich einen frei programmierbaren *TiCo*-Prozessor, der Zugriff auf alle Eingänge des Moduls hat. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

Die Eingänge verfügen über geschirmte LEMO-Buchsen (2-polig, CAMAC Europannorm, siehe Abb. 30).

Das Modul Pro II-Aln-8/18-L2 Rev. E hat einen Eingangs-Spannungsbereich von $\pm 10V$ und per Software programmierbare Verstärkung von 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Das Modul beinhaltet zusätzlich eine Ablaufsteuerung, die Messwerte an mehreren oder allen Eingangskanälen nacheinander einlesen kann.

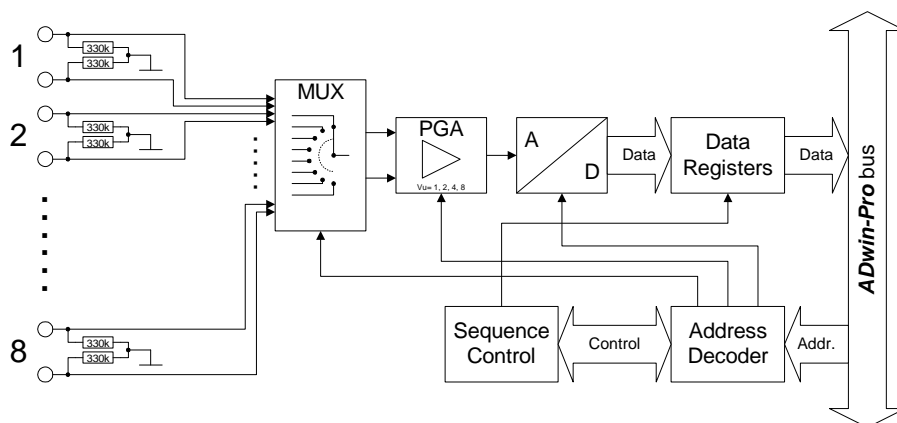
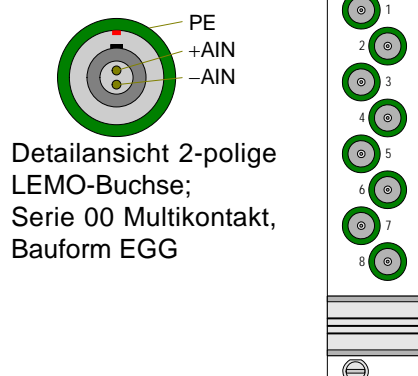


Abb. 28 – Pro II-Aln-8/18-L2 Rev. E: Blockschaltbild

Eingangskanäle	8 differentiell über Multiplexer	
Auflösung	18 Bit	
Wandlungszeit	max. 2 μs	
Abtastrate	max. 500ksps	
Multiplexer Einschwingzeit	2,5 μs	
Messbereich	$\pm 10V$	
Verstärkung	1, 2, 4, 8 per Software einstellbar	
Genauigkeit	INL	typisch ± 4 LSB
	DNL	max. ± 1 LSB
Eingangswiderstand	330k Ω , $\pm 2\%$	
Spannungsfestigkeit	$\pm 35V$	
Offsetfehler	abgleichbar	
Offsetdrift	± 30 ppm/ $^{\circ}C$	
Steckverbindung	8 LEMO-Buchsen, 2-polig	

Abb. 29 – Pro II-Aln-8/18-L2 Rev. E: Spezifikation



Detailsicht 2-polige LEMO-Buchse; Serie 00 Multikontakt, Bauform EGG

Abb. 30 – Pro II-AIn-8/18-L2 Rev. E: Frontplatte

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADC</code> , <code>P2_ADC24</code> <code>P2_Set_Mux</code> , <code>P2_Start_Conv</code> <code>P2_Wait_EOC</code> <code>P2_Read_ADC</code> , <code>P2_Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv24</code>
Ablaufsteuerung anwenden	<code>P2_Seq_Init</code> , <code>P2_Seq_Start</code> <code>P2_Seq_Read</code> , <code>P2_Seq_Read24</code> <code>P2_Seq_Read24_Packed</code> <code>P2_Seq_Wait</code>
Grenzwerte überwachen	<code>P2_ADC_Read_Limit</code> <code>P2_ADC_Set_Limit</code>
Wandlung synchronisieren	<code>P2_Sync_All</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> , <code>P2_Event_Read</code> <code>P2_Event_Config</code> , <code>P2_Event2_Config</code>

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Die Include-Datei `AInTiCo.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Einzelmessung durchführen – vollständig oder schrittweise	<code>ADC</code> , <code>ADC24</code> <code>Set_Mux</code> , <code>Start_Conv</code> , <code>Wait_EOC</code> <code>Read_ADC</code> , <code>Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>Read_ADC_SConv</code> <code>Read_ADC_SConv24</code>
LEDs einstellen	<code>Check_LED</code> , <code>Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>Event_Enable</code> , <code>Event_Read</code> <code>Event_Config</code> , <code>Trigger_Event</code>

Programmierung in ADbasic

Programmierung in TiCoBasic

Programmierung TiCo-Zugriff

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei `ADwinPro_All.inc` definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch mit dem <i>TiCo</i> -Prozessor über globale Variablen	P2_TDrv_Init P2_GetData_Long, P2_Get_Par, P2_Get_Par_Block P2_SetData_Long, P2_Set_Par, P2_Set_Par_Block P2_Get_TiCo_RingBuffer, P2_Set_TiCo_RingBuffer P2_RingBuffer_Empty P2_RingBuffer_Full
<i>TiCo</i> -Prozessor steuern	P2_TiCo_Reset, P2_TiCo_Start, P2_TiCo_Stop P2_Get_TiCo_Bootloader_Status P2_Get_TiCo_Status, P2_Workload
<i>TiCo</i> -Prozesse steuern	P2_Process_Status P2_TiCo_Get_Processdelay P2_TiCo_Set_Processdelay P2_TiCo_Start_Process P2_TiCo_Stop_Process
<i>TiCo</i> -Programme übertragen	P2_TiCo_Flash, P2_TiCo_Load

5.4.2 Pro II-AIn-32/18-D Rev. E

Analoges Eingangsmodul Pro II-AIn-32/18-D Rev. E mit einem 18-Bit ADC, 32 Eingängen und einem programmierbaren Verstärker (PGA). Das Modul kann mit Verstärkern, Pro-TC und Pro-PT-Modulen kombiniert werden.

Die Variante Pro II-AIn-32/18-D-TiCo Rev. E besitzt zusätzlich einen frei programmierbaren *TiCo*-Prozessor, der Zugriff auf alle Eingänge des Moduls hat. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

Das Modul hat 32 single ended-Eingänge oder 16 differentielle Eingänge (über Software wählbar). Die Eingänge sind auf eine 37-polige D-Sub-Buchse geführt; Pinbelegung siehe Abb. 33 und 34.

Nach dem Einschalten ist das Modul auf 16 differentielle Eingänge eingestellt.

Das Modul Pro II-AIn-32/18-D Rev. E besitzt einen Eingangs-Spannungsbereich von $\pm 10V$ und eine programmierbare Verstärkung von 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Ab Werk sind die Eingänge mit der Masse des Pro-Geräts verbunden. Alternativ kann ein Massepotenzial an einem der AGND-Pins angeschlossen werden. Die Masseverbindung zum Pro-Gerät sollte dann aufgetrennt werden, indem der DIP-Schalter auf die Position **GND LIFT** gestellt wird (Abb. 35).

Betreiben Sie das Modul in keinem Fall ohne Masseverbindung.

Das Modul beinhaltet zusätzlich eine Ablaufsteuerung, die Messwerte an mehreren oder allen Eingangskanälen nacheinander einlesen kann.

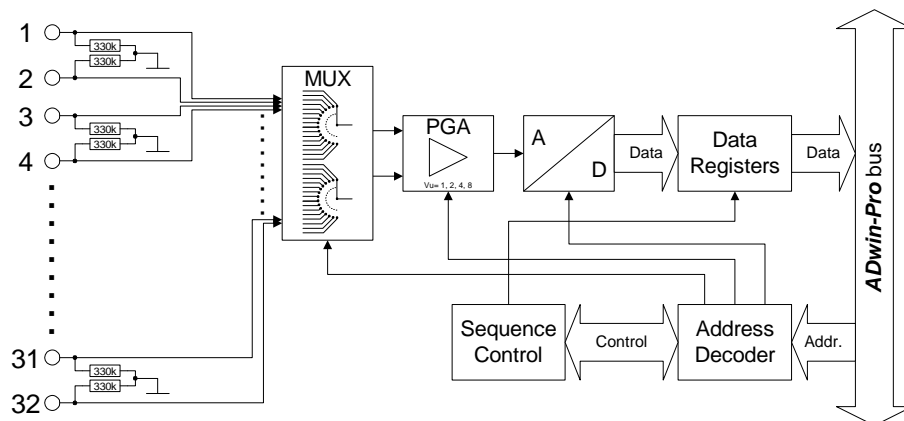


Abb. 31 – Pro II-AIn-32/18-D Rev. E: Blockschaltbild

Eingangskanäle:	32 single ended / 16 differentiell, Multiplexer	
Auflösung:	18 Bit	
Wandlungszeit:	max. 2µs	
Abtastrate:	max. 500ksps	
Multiplexer Einschwingzeit:	2,5µs	
Messbereich:	±10V	
Verstärkung:	1, 2, 4, 8 per Software einstellbar	
Genauigkeit	INL	typisch ±4 LSB
	DNL	max. ±1 LSB
Eingangswiderstand:	330kΩ, ±2%	
Spannungsfestigkeit:	±35V	

Abb. 32 – Pro II-AIn-32/18-D Rev. E: Spezifikation

Offsetfehler:	abgleichbar
Offsetdrift:	$\pm 30 \text{ ppm}/^\circ\text{C}$
Steckverbindung:	37-polige D-Sub-Buchse

Abb. 32 – Pro II-AIn-32/18-D Rev. E: Spezifikation

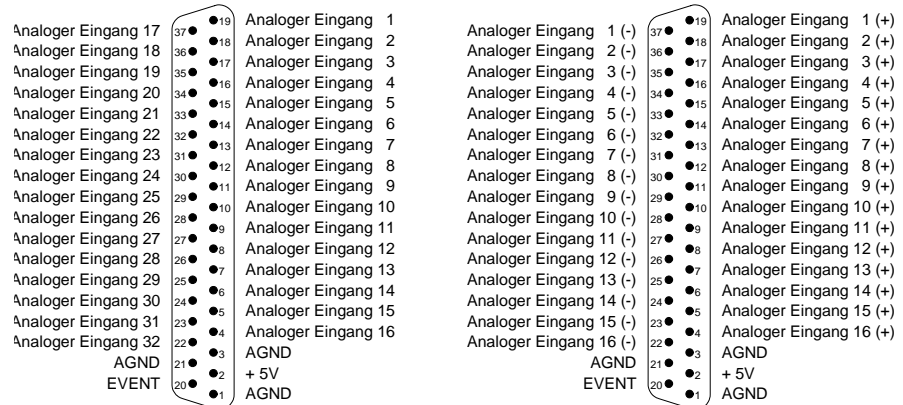


Abb. 33 – Pro II-AIn-32/18-D Rev. E: Pinbelegung single ended

Abb. 34 – Pro II-AIn-32/18-D Rev. E: Pinbelegung differentiell

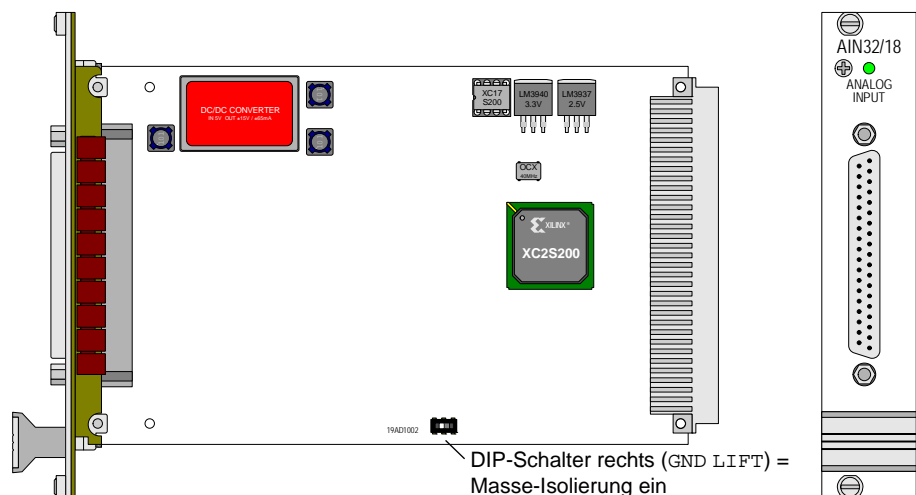


Abb. 35 – Pro II-AIn-32/18-D Rev. E: Platine und Frontplatte

Programmierung in ADbasic

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Einzelmessung durchführen – vollständig oder schrittweise	P2_ADC, P2_ADC24 P2_Set_Mux, P2_Start_Conv P2_Wait_EOC P2_Read_ADC, P2_Read_ADC24
Messwert lesen und neue Messung starten	P2_Read_ADC_SConv P2_Read_ADC_SConv24
Ablaufsteuerung anwenden	P2_Seq_Init, P2_Seq_Start P2_Seq_Read, P2_Seq_Read24 P2_Seq_Read24_Packed P2_Seq_Wait

Bereich	Befehle
Grenzwerte überwachen	<code>P2_ADC_Read_Limit</code> <code>P2_ADC_Set_Limit</code>
Wandlung synchronisieren	<code>P2_Sync_All</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> , <code>P2_Event_Read</code> <code>P2_Event_Config</code> , <code>P2_Event2_Config</code>

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Die Include-Datei `AInTiCo.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Einzelmessung durchführen – vollständig oder schrittweise	<code>SE_Diff</code> <code>ADC</code> , <code>ADC24</code> <code>Set_Mux</code> , <code>Start_Conv</code> <code>Wait_EOC</code> <code>Read_ADC</code> , <code>Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>Read_ADC_SConv</code> <code>Read_ADC_SConv24</code>
LEDs einstellen	<code>Check_LED</code> , <code>Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>Event_Enable</code> , <code>Event_Read</code> <code>Event_Config</code> , <code>Trigger_Event</code>

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei `ADwinPro_All.inc` definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch mit dem <i>TiCo</i> -Prozessor über globale Variablen	<code>P2_TDrv_Init</code> <code>P2_GetData_Long</code> , <code>P2_Get_Par</code> , <code>P2_Get_Par_Block</code> <code>P2_SetData_Long</code> , <code>P2_Set_Par</code> , <code>P2_Set_Par_Block</code> <code>P2_Get_TiCo_RingBuffer</code> , <code>P2_Set_TiCo_RingBuffer</code> <code>P2_RingBuffer_Empty</code> <code>P2_RingBuffer_Full</code>
<i>TiCo</i> -Prozessor steuern	<code>P2_TiCo_Reset</code> , <code>P2_TiCo_Start</code> , <code>P2_TiCo_Stop</code> <code>P2_Get_TiCo_Bootloader_Status</code> <code>P2_Get_TiCo_Status</code> , <code>P2_Workload</code>
<i>TiCo</i> -Prozesse steuern	<code>P2_Process_Status</code> <code>P2_TiCo_Get_Processdelay</code> <code>P2_TiCo_Set_Processdelay</code> <code>P2_TiCo_Start_Process</code> <code>P2_TiCo_Stop_Process</code>
<i>TiCo</i> -Programme übertragen	<code>P2_TiCo_Flash</code> , <code>P2_TiCo_Load</code>

Programmierung in TiCoBasic

Programmierung TiCo-Zugriff

5.4.3 Pro II-AIn-8/18-8B Rev. E

Analoges Eingangsmodul Pro II-AIn-8/18-8B Rev. E mit einem 18-Bit ADC, 16 analogen Eingängen und einem programmierbaren Verstärker (PGA). Das Modul basiert auf Pro II-AIn-32/18-D Rev. E und hat eine Zusatzplatine mit 8 Steckplätzen für 8B-Module.

Von den 16 Moduleingängen sind 8 Eingänge für 8B-Module und 8 differentielle Eingänge. Die Eingänge stehen auf zwei 37-poligen D-Sub-Buchsen zur Verfügung; Pinbelegung siehe Abb. 37.

Der Eingangs-Spannungsbereich nach den 8B-Modulen beträgt $\pm 10V$. Die Verstärkung ist programmierbar als 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Ab Werk sind die Eingänge mit der Masse des Pro-Gehäuses verbunden. Alternativ kann an den differentiellen Eingängen ein Massepotenzial an einem der AGND-Pins angeschlossen werden. Die Masseverbindung zum Pro-Gehäuse sollte dann aufgetrennt werden, indem der DIP-Schalter auf die Position **GND LIFT** gestellt wird (siehe Abb. 35, Pro II-AIn-32/18-D Rev. E).

Betreiben Sie das Modul in keinem Fall ohne Masseverbindung.

Das Modul beinhaltet zusätzlich eine Ablaufsteuerung, die Messwerte an mehreren oder allen Eingangskanälen nacheinander einlesen kann.

Eingangskanäle:	16 über Multiplexer: 8 für MB8-Module, 8 differentiell	
Auflösung:	18 Bit	
Wandlungszeit:	max. 2µs	
Abtastrate:	max. 500ksps	
Multiplexer Einschwingzeit:	2,5µs	
Messbereich:	±10V	
Verstärkung:	1, 2, 4, 8 per Software einstellbar	
Genauigkeit	INL	typisch ±4 LSB
	DNL	max. ±1 LSB
Eingangswiderstand:	330kΩ, ±2%	
Spannungsfestigkeit:	±35V	
Offsetfehler:	abgleichbar	
Offsetdrift:	±30ppm/°C	
Steckverbindung:	37-polige D-Sub-Buchse	

Abb. 36 – Pro II-AIn-8/18-8B Rev. E: Spezifikation

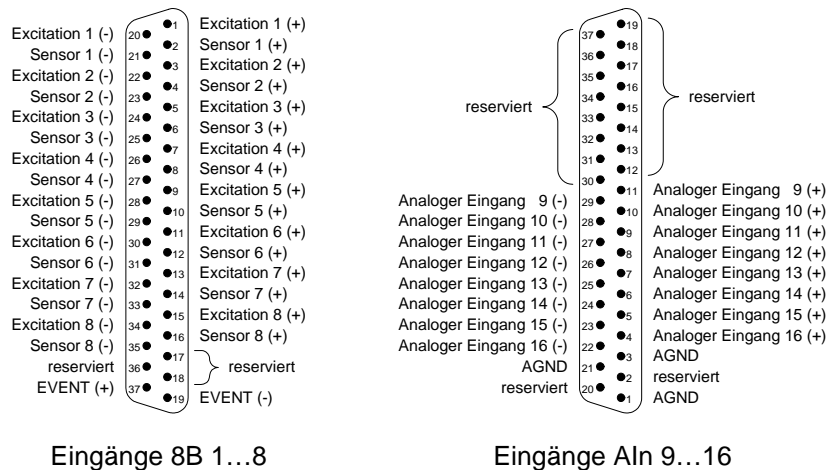


Abb. 37 – Pro II-AIn-8/18-8B Rev. E: Pinbelegungen

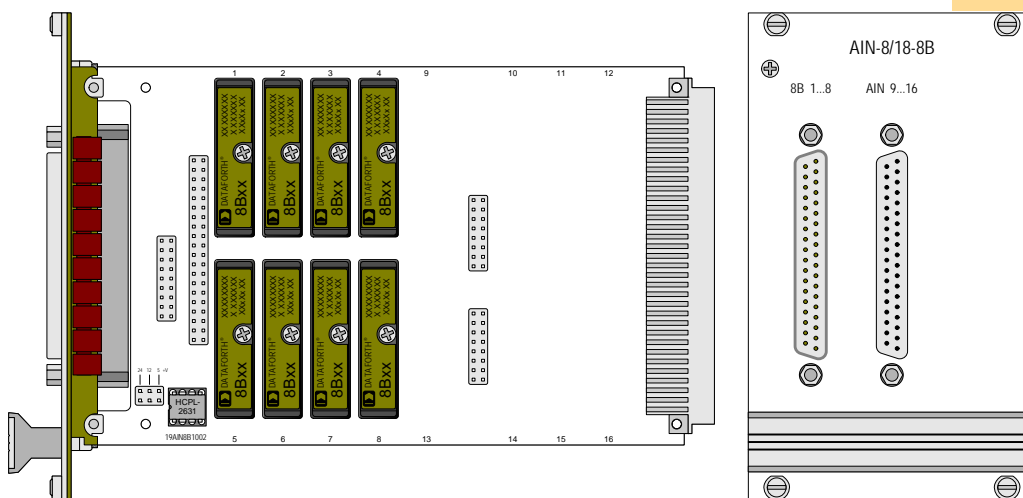


Abb. 38 – Pro II-AIn-8/18-8B Rev. E: Platine und Frontplatte

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADC</code> , <code>P2_ADC24</code> <code>P2_Set_Mux</code> , <code>P2_Start_Conv</code> <code>P2_Wait_EOC</code> <code>P2_Read_ADC</code> , <code>P2_Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv24</code>
Ablaufsteuerung anwenden	<code>P2_Seq_Init</code> , <code>P2_Seq_Start</code> <code>P2_Seq_Read</code> , <code>P2_Seq_Read24</code> <code>P2_Seq_Read24_Packed</code> <code>P2_Seq_Wait</code>
Grenzwerte überwachen	<code>P2_ADC_Read_Limit</code> <code>P2_ADC_Set_Limit</code>
Wandlung synchronisieren	<code>P2_Sync_All</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> , <code>P2_Event_Read</code> <code>P2_Event_Config</code> , <code>P2_Event2_Config</code>

Programmierung

5.4.4 Pro II-Aln-16/18-8B Rev. E

Analoges Eingangsmodul Pro II-Aln-16/18-8B Rev. E mit einem 18-Bit ADC, 16 analogen Eingängen und einem programmierbaren Verstärker (PGA). Das Modul basiert auf Pro II-Aln-32/18-D Rev. E und hat eine Zusatzplatine mit 16 Steckplätzen für 8B-Module.

Das Modul hat 16 Eingänge (über 8B-Module) auf zwei 37-poligen D-Sub-Buchsen; Pinbelegung siehe Abb. 40.

Der Eingangs-Spannungsbereich nach den 8B-Modulen beträgt $\pm 10V$. Die Verstärkung ist programmierbar als 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Ab Werk sind die Eingänge mit der Masse des Pro-Gehäuses verbunden. Alternativ kann ein Massepotenzial an einem der AGND-Pins angeschlossen werden. Die Masseverbindung zum Pro-Gehäuse sollte dann aufgetrennt werden, indem der DIP-Schalter auf die Position GND LIFT gestellt wird (siehe Abb. 35, Pro II-Aln-32/18-D Rev. E).

Betreiben Sie das Modul in keinem Fall ohne Masseverbindung.

Das Modul beinhaltet zusätzlich eine Ablaufsteuerung, die Messwerte an mehreren oder allen Eingangskanälen nacheinander einlesen kann.

Eingangskanäle:	16 über Multiplexer
Auflösung:	18 Bit
Wandlungszeit:	max. 2 μs
Abtastrate:	max. 500ksps
Multiplexer Einschwingzeit:	2,5 μs
Messbereich:	$\pm 10V$
Verstärkung:	1, 2, 4, 8 per Software einstellbar
Genauigkeit	INL
	DNL
	typisch ± 4 LSB
	max. ± 1 LSB
Eingangswiderstand:	330k Ω , $\pm 2\%$
Spannungsfestigkeit:	$\pm 35V$
Offsetfehler:	abgleichbar
Offsetdrift:	± 30 ppm/ $^{\circ}C$
Steckverbindung:	37-polige D-Sub-Buchse

Abb. 39 – Pro II-Aln-16/18-8B Rev. E: Spezifikation

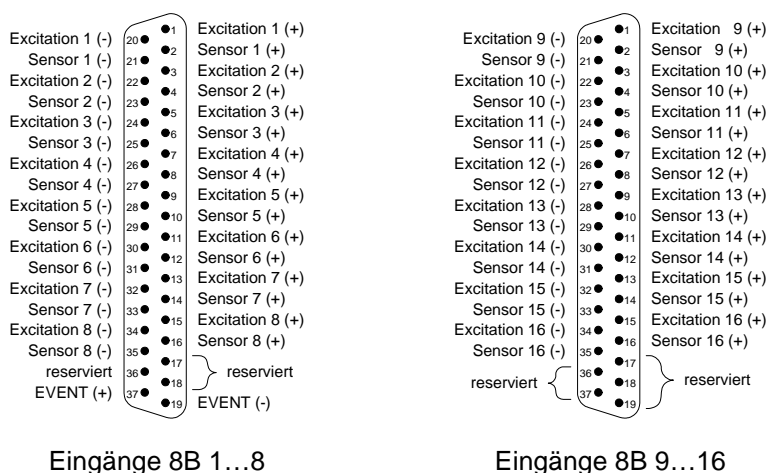


Abb. 40 – Pro II-Aln-16/18-8B Rev. E: Pinbelegungen

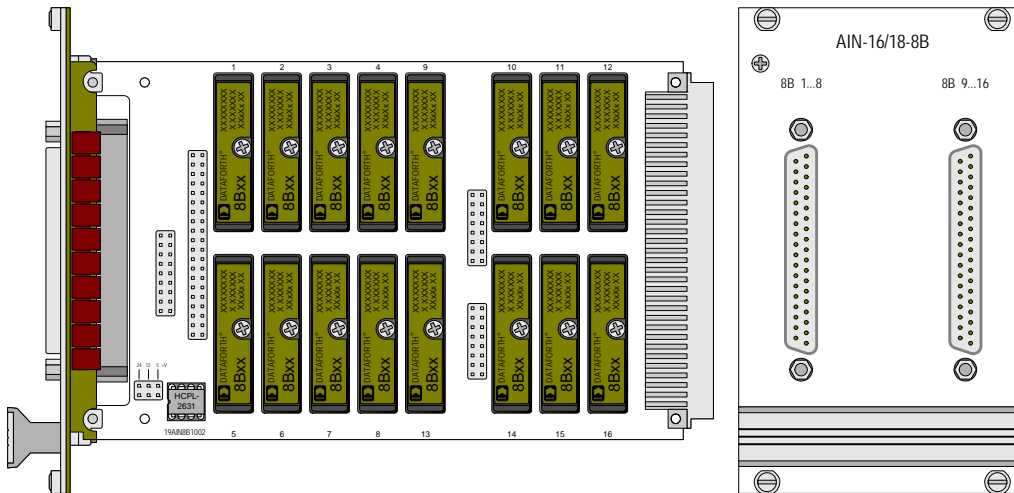


Abb. 41 – Pro II-AIn-16/18-8B Rev. E: Platine und Frontplatte

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADC</code> , <code>P2_ADC24</code> <code>P2_Set_Mux</code> , <code>P2_Start_Conv</code> <code>P2_Wait_EOC</code> <code>P2_Read_ADC</code> , <code>P2_Read_ADC24</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv24</code>
Ablaufsteuerung anwenden	<code>P2_Seq_Init</code> , <code>P2_Seq_Start</code> <code>P2_Seq_Read</code> , <code>P2_Seq_Read24</code> <code>P2_Seq_Read24_Packed</code> <code>P2_Seq_Wait</code>
Grenzwerte überwachen	<code>P2_ADC_Read_Limit</code> <code>P2_ADC_Set_Limit</code>
Wandlung synchronisieren	<code>P2_Sync_All</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> , <code>P2_Event_Read</code> <code>P2_Event_Config</code> , <code>P2_Event2_Config</code>

Programmierung

5.4.5 Pro II-Aln-F-4/14 Rev. E

Analoges Eingangsmodul Pro II-Aln-F-4/14 Rev. E mit 4 Fast-ADC zu 14 Bit und 4 differentiellen Eingängen.

Für die Eingänge sind folgende Steckverbindungen verfügbar:

- Pro II-Aln-F-4/14-L2: geschirmte LEMO-Buchsen 2-polig, CAMAC Europannorm.
- Pro II-Aln-F-4/14-D: D-Sub-Buchse 37-polig.
- Pro II-Aln-F-4/14-B: BNC-Buchsen.

Die Wandler des Moduls arbeiten eigenständig mit einer festen Abtastrate von 50 MHz. Der moduleigene Speicher erlaubt, die große Menge anfallender Daten – insbesondere bei Burst-Messungen – zwischenspeichern.

Das Modul verfügt über mehrere Betriebsarten für die Wandlung von analogen Signalen:

- Einzelmessung: Jede Messung wird im *ADbasic*-Programm einzeln abgefragt und weiter verarbeitet; die Wandlung führt das Modul selbstständig durch.
- Einfache Burst-Messreihe: Das *ADbasic*-Programm startet eine vollständige Messreihe, also eine definierte Anzahl von Einzelmessungen.
- Kontinuierliche Burst-Messreihe: Das *ADbasic*-Programm startet eine Messreihe, die kontinuierlich Einzelmessungen ausführt, bis die Messreihe gestoppt wird. Die Daten werden in einem Ringspeicher abgelegt.

Das Modul Pro II-Aln-F-4/14 Rev. E hat einen Eingangs-Spannungsbereich von bipolar $\pm 10V$. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Das Modul kann optional anstelle einfacher Messwerte den gleitenden Mittelwert aus 2...32 Messwerten zurückgeben.

Burst-Messreihen

Das Modul führt Burst-Messreihen unabhängig vom Prozessormodul des *ADwin*-Systems aus. Die Messwerte – Anzahl und Messfrequenz sind vorab im Programm zu definieren – werden im Burst-Speicher auf dem Modul abgelegt. Das Prozessormodul liest nur noch die gespeicherten Messwerte (auch während der laufenden Messreihe) und verarbeitet sie.

Bei einer kontinuierlichen Burst-Messreihe muss die Messfrequenz mit der Auslesegeschwindigkeit abgestimmt werden. Hierbei sind verschiedene Aspekte von Bedeutung:

- Das Auslesen der Messwerte geschieht in Blöcken. Je größer der Datenblock ist, um so schneller ist der Lesevorgang im Mittel.

Beachten Sie: Das blockweise Lesen kann andere, auch hochpriorie Prozesse verzögern. Je größer der gelesene Block ist, umso eher kann eine Verzögerung entstehen.

- Der Zeitversatz zwischen kontinuierlicher Wandlung und blockweisem Lesen erfordert einen Datenpuffer. Hierzu muss bei der Initialisierung der Burst-Messung ein genügend großer Speicherbereich reserviert werden (**P2_BURST_INIT**, Parameter *samples*).

Event-Eingänge

Bei der Modulversion Pro II-Aln-F-4/14-D Rev. E (mit D-Sub-Buchse) kann eine Burst-Messreihe über externe Event-Signale gesteuert werden, so dass für jedes (resultierende) Event-Signal ein Messwert gespeichert wird.

Optional kann ein Kanal der Burst-Messreihe als Zeitkanal genutzt werden, in dem bei jedem Event-Signal der Zählerstand des moduleigenen Timers gespeichert wird.

Das Modul hat 3 differentielle Event-Eingänge: EVENT/A, B, ENABLE, deren Signale das Modul zum resultierenden Event-Signal verarbeitet. Diese Vorverarbeitung der Signale ist konfigurierbar mit **P2_Event2_Config**.

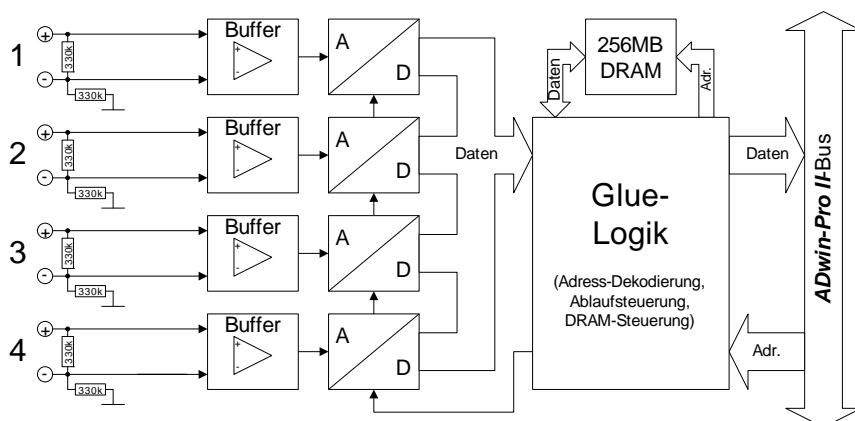


Abb. 42 – Pro II-AIn-F-4/14 Rev. E: Blockschaltbild

Eingangskanäle	4 differentiell	
Auflösung	14 Bit	
Konvertierungszeit	0,02µs (je ADC)	
Abtastrate	50 000ksps (je ADC)	
Eingangs-Bandbreite	0 ... 4MHz	
Speichergroße	256MiB oder 2 ²⁷ = 134 217 728 Messwerte insgesamt	
Messbereich	±10V mit max. Offset 3,5 V	
Genauigkeit	INL	typisch ±1,2 LSB, max. ±5 LSB
	DNL	typisch ±0,5 LSB, max. ±1 LSB
Eingangswiderstand	330kΩ, ±2%	
Spannungsfestigkeit:	±35V	
Offsetfehler	abgleichbar	
Offsetdrift	±30ppm/°C vom Endwert	
Event-Eingang (nur D-Sub-Buchse)	3 differentiell; RS422/485 kompatibel (5V differentiell, 120 Ω Bus-Abschlusswiderstand) max. Signalfrequenz 16MHz	
Steckerverbindung	4 LEMO-Buchsen optional: 37-polige D-Sub-Buchse optional: 4 BNC-Buchsen.	

Abb. 43 – Pro II-AIn-F-4/14 Rev. E: Spezifikation

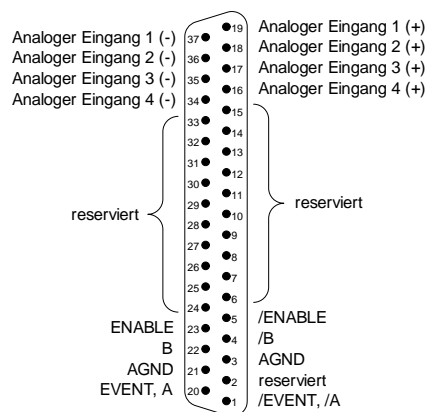


Abb. 44 – Pro-AIn-F-4/14-D Rev. E: Pinbelegung differenziell

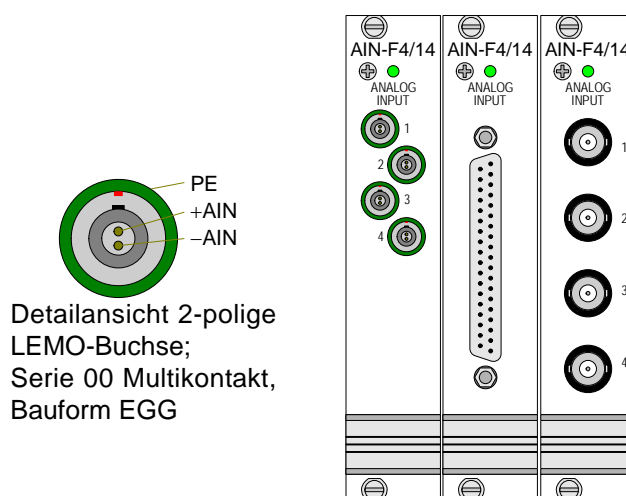


Abb. 45 – Pro II-AIn-F-4/14 Rev. E: Frontplatten

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ablaufsteuerung konfigurieren	<code>P2_ADCF_Mode</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADCF</code> <code>P2_Start_ConvF</code> , <code>P2_Wait_EOCF</code> <code>P2_Read_ADCF</code>
Mehrere Messwerte lesen	<code>P2_Read_ADCF32</code> , <code>P2_Read_ADCF4</code> <code>P2_Read_ADCF4_Packed</code>
Burst-Messung durchführen	<code>P2_Burst_Init</code> <code>P2_Burst_Start</code> , <code>P2_Burst_Stop</code> <code>P2_Burst_Status</code> , <code>P2_Burst_Reset</code> <code>P2_Burst_Read</code> <code>P2_Burst_Read_Index</code> <code>P2_Burst_Read_Unpacked1</code> <code>P2_Burst_Read_Unpacked2</code> <code>P2_Burst_Read_Unpacked4</code> <code>P2_Burst_CRead_Unpacked1</code> <code>P2_Burst_CRead_Unpacked2</code> <code>P2_Burst_CRead_Unpacked4</code>

Bereich	Befehle
Wandlung synchronisieren	P2_Sync_All, P2_Sync_Enable P2_Sync_Stat
Grenzwerte überwachen	P2_ADCF_Read_Limit P2_ADCF_Set_Limit
LEDs einstellen	P2_Check_LED, P2_Set_LED
Interrupts und Event-Eingang einstellen	P2_Event_Enable, P2_Event_Read P2_Event_Config, P2_Event2_Config

5.4.6 Pro II-Aln-F-8/14 Rev. E

Analoges Eingangsmodul Pro II-Aln-F-8/14 Rev. E mit 8 Fast-ADC zu 14 Bit und 8 differentiellen Eingängen.

Für die Eingänge sind folgende Steckverbindungen verfügbar:

- Pro II-Aln-F-8/14-L2: geschirmte LEMO-Buchsen 2-polig, CAMAC Europannorm.
- Pro II-Aln-F-8/14-D: D-Sub-Buchse 37-polig.
- Pro II-Aln-F-8/14-B: BNC-Buchsen.

Die Wandler des Moduls arbeiten eigenständig mit einer festen Abtastrate von 50MHz. Die große Menge anfallender Daten – insbesondere bei Burst-Messungen – wird im moduleigenen Speicher zwischengespeichert. Wenn auf allen 8 Kanälen Messwerte gewandelt werden, ist die Abtastrate wegen Erreichens der maximalen Speicherzugriffsrate auf 25MHz je Kanal begrenzt.

Das Modul verfügt über mehrere Betriebsarten für die Wandlung von analogen Signalen:

- Einzelmessung: Das Modul führt die Wandlung selbstständig (mit fester Abtastrate) durch. Das *ADbasic*-Programm fragt bei Bedarf den aktuellsten Messwert ab und verarbeitet ihn weiter.
- Einfache Burst-Messreihe: Das *ADbasic*-Programm startet eine vollständige Messreihe, also eine definierte Anzahl von Einzelmessungen.
- Kontinuierliche Burst-Messreihe: Das *ADbasic*-Programm startet eine Messreihe, die kontinuierlich Einzelmessungen ausführt, bis die Messreihe gestoppt wird. Die Daten werden in einem Ringspeicher abgelegt.

Das Modul Pro II-Aln-F-8/14 Rev. E hat einen Eingangs-Spannungsbereich von bipolar $\pm 10V$. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Das Modul kann optional anstelle einfacher Messwerte den gleitenden Mittelwert aus 2...32 Messwerten zurückgeben.

Burst-Messreihen

Das Modul führt Burst-Messreihen unabhängig vom Prozessormodul des *ADwin*-Systems aus. Die Messwerte – Anzahl und Messfrequenz sind vorab im Programm zu definieren – werden im Burst-Speicher auf dem Modul abgelegt. Das Prozessormodul liest nur noch die gespeicherten Messwerte (auch während der laufenden Messreihe) und verarbeitet sie.

Bei einer kontinuierlichen Burst-Messreihe muss die Messfrequenz mit der Auslesegeschwindigkeit abgestimmt werden. Hierbei sind verschiedene Aspekte von Bedeutung:

- Das Auslesen der Messwerte geschieht in Blöcken. Je größer der Datenblock ist, um so schneller ist der Lesevorgang im Mittel.

Beachten Sie: Das blockweise Lesen kann andere, auch hochpriorie Prozesse verzögern. Je größer der gelesene Block ist, umso eher kann eine Verzögerung entstehen.

- Der Zeitversatz zwischen kontinuierlicher Wandlung und blockweisem Lesen erfordert einen Datenpuffer. Hierzu muss bei der Initialisierung der Burst-Messung ein genügend großer Speicherbereich reserviert werden (**P2_BURST_INIT**, Parameter *samples*).

Event-Eingänge

Bei der Modulversion Pro II-Aln-F-8/14-D Rev. E (mit D-Sub-Buchse) kann eine Burst-Messreihe über externe Event-Signale gesteuert werden, so dass für jedes (resultierende) Event-Signal ein Messwert gespeichert wird.

Optional kann ein Kanal der Burst-Messreihe als Zeitkanal genutzt werden, in dem bei jedem Event-Signal der Zählerstand des moduleigenen Timers gespeichert wird.

Das Modul hat 3 differentielle Event-Eingänge: EVENT/A, B, ENABLE, deren Signale das Modul zum resultierenden Event-Signal verarbeitet. Diese Vorverarbeitung der Signale ist konfigurierbar mit **P2_Event2_Config**.

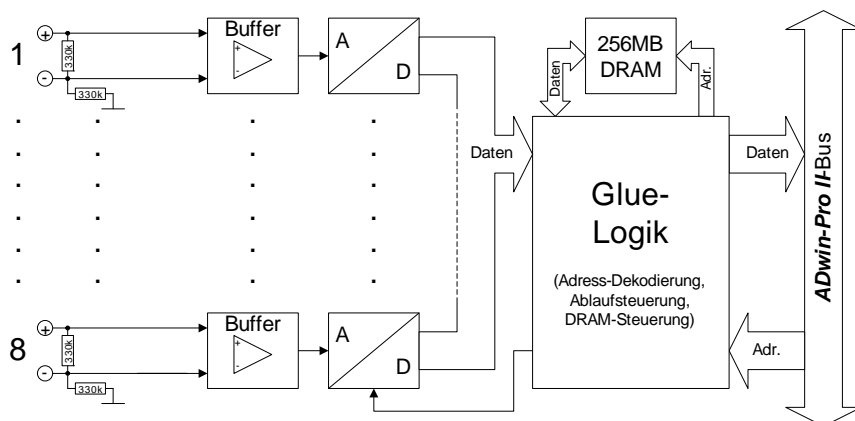


Abb. 46 – Pro II-AIn-F-8/14 Rev. E: Blockschaltbild

Eingangskanäle	8 differentiell
Auflösung	14 Bit
Konvertierungszeit	0,02µs (je ADC)
Abtastrate	4×50000 ksps (je ADC) oder 8×25000 ksps (je ADC)
Eingangs-Bandbreite	0 ... 4MHz
Speichergroße	256MiB oder $2^{27} = 134217728$ Messwerte insgesamt
Messbereich	±10V mit max. Offset 3,5 V
Genauigkeit	INL typisch ±1,2 LSB, max. ±5 LSB
DNL	typisch ±0,5 LSB, max. ±1 LSB
Eingangswiderstand	330kΩ, ±2%
Spannungsfestigkeit:	±35V
Offsetfehler	abgleichbar
Offsetdrift	±30ppm/°C vom Endwert
Event-Eingang (nur D-Sub-Buchse)	3 differentiell; RS422/485 kompatibel (5V differentiell, 120 Ω Bus-Abschlusswiderstand) max. Signalfrequenz 16MHz
Steckerverbindung	8 LEMO-Buchsen, 2-polig optional: 37-polige D-Sub-Buchse optional: 8 BNC-Buchsen.

Abb. 47 – Pro II-AIn-F-8/14 Rev. E: Spezifikation

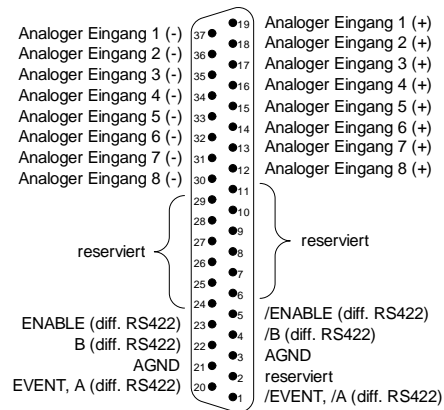


Abb. 48 – Pro II-AIn-F-8/14 Rev. E: Pinbelegung D-Sub-Buchse, differentiell

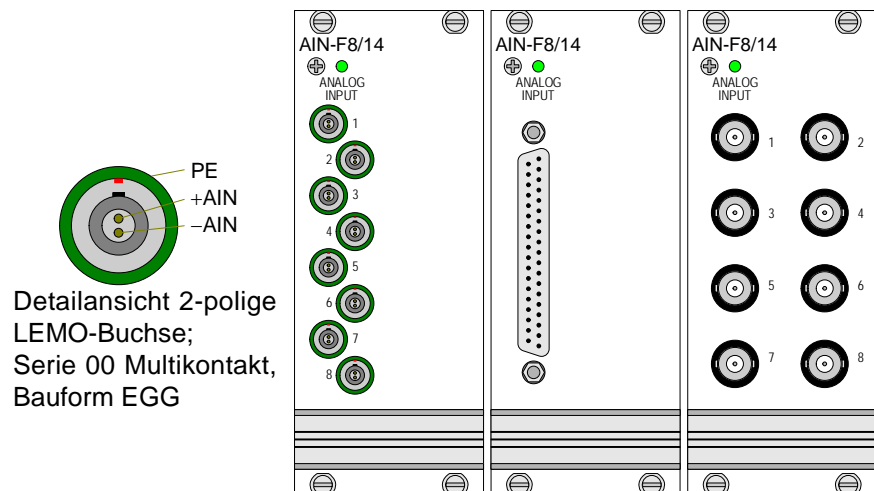


Abb. 49 – Pro II-AIn-F-8/14 Rev. E: Frontplatten

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ablaufsteuerung konfigurieren	<code>P2_ADCF_Mode</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADCF</code> <code>P2_Start_ConvF, P2_Wait_EOCF</code> <code>P2_Read_ADCF</code>
Mehrere Messwerte lesen	<code>P2_Read_ADCF32</code> <code>P2_Read_ADCF4, P2_Read_ADCF8</code> <code>P2_Read_ADCF4_Packed</code> <code>P2_Read_ADCF8_Packed</code>
Burst-Messung durchführen	<code>P2_Burst_Init</code> <code>P2_Burst_Start, P2_Burst_Stop</code> <code>P2_Burst_Status, P2_Burst_Reset</code> <code>P2_Burst_Read</code> <code>P2_Burst_Read_Index</code> <code>P2_Burst_Read_Unpacked1/2/4/8</code> <code>P2_Burst_CRead_Unpacked1/2/4/8</code>
Wandlung synchronisieren	<code>P2_Sync_All, P2_Sync_Enable</code> <code>P2_Sync_Stat</code>

Bereich	Befehle
Grenzwerte überwachen	P2_ADCF_Read_Limit P2_ADCF_Set_Limit
LEDs einstellen	P2_Check_LED, P2_Set_LED
Interrupts und Event-Eingang einstellen	P2_Event_Enable, P2_Event_Read P2_Event_Config, P2_Event2_Config

5.4.7 Pro II-Aln-F-4/16 Rev. E

Analoges Eingangsmodul Pro II-Aln-F-4/16 Rev. E mit 4 Fast-ADC zu 16 Bit und 4 differentiellen Eingängen.

Für die Eingänge sind folgende Steckverbindungen verfügbar:

- Pro II-Aln-F-4/16-L2: geschirmte LEMO-Buchsen 2-polig, CAMAC Europannorm.
- Pro II-Aln-F-4/16-D: D-Sub-Buchse 37-polig.
- Pro II-Aln-F-4/16-B: BNC-Buchsen.

Die Wandler des Moduls arbeiten mit einer Abtastrate von bis zu 4MHz. Der moduleigene Speicher erlaubt, die anfallenden Daten – insbesondere bei Burst-Messungen – zwischenspeichern.

Das Modul verfügt über mehrere Betriebsarten für die Wandlung von analogen Signalen:

- Einzelmessung: Jede Messung wird im *ADbasic*-Programm einzeln gestartet, abgefragt und weiter verarbeitet.
- Einfache Burst-Messreihe: Das *ADbasic*-Programm startet eine vollständige Messreihe, also eine definierte Anzahl von Einzelmessungen.
- Kontinuierliche Burst-Messreihe: Das *ADbasic*-Programm startet eine Messreihe, die kontinuierlich Einzelmessungen ausführt, bis die Messreihe gestoppt wird. Die Daten werden in einem Ringspeicher abgelegt.

Das Modul Pro II-Aln-F-4/16 Rev. E hat einen Eingangs-Spannungsbereich von $\pm 10V$ und per Software programmierbare Verstärkung von 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Das Modul kann optional anstelle einfacher Messwerte den Mittelwert aus 2...32 Messwerten zurückgeben. In diesem Fall wird die eingestellte Anzahl an Messwerten gewandelt und daraus der Mittelwert berechnet.

Burst-Messreihen

Das Modul führt Burst-Messreihen unabhängig vom Prozessormodul des *ADwin*-Systems aus. Die Messwerte – Anzahl und Messfrequenz legen Sie vorab im Programm fest – werden im Burst-Speicher auf dem Modul abgelegt. Das Prozessormodul liest nur noch die gespeicherten Messwerte (auch während der laufenden Messreihe) und verarbeitet sie.

Bei einer kontinuierlichen Burst-Messreihe müssen Sie die Messfrequenz mit der Auslesegeschwindigkeit abstimmen. Hierbei sind verschiedene Aspekte von Bedeutung:

- Das Auslesen der Messwerte geschieht in Blöcken. Je größer der Datenblock ist, um so schneller ist der Lesevorgang im Mittel.

Beachten Sie: Das blockweise Lesen kann andere, auch hochpriorisierte Prozesse verzögern. Je größer der gelesene Block ist, umso eher kann eine Verzögerung entstehen.

- Der Zeitversatz zwischen kontinuierlicher Wandlung und blockweisem Lesen erfordert einen Datenpuffer. Hierzu müssen Sie bei der Initialisierung der Burst-Messung einen genügend großen Speicherbereich reservieren (**P2_BURST_INIT**, Parameter *samples*).

Event-Eingänge

Bei der Modulversion Pro II-Aln-F-4/16-D Rev. E (mit D-Sub-Buchse) kann eine Burst-Messreihe über externe Event-Signale gesteuert werden, so dass für jedes (resultierende) Event-Signal ein Messwert gespeichert wird.

Optional kann ein Kanal der Burst-Messreihe als Zeitkanal genutzt werden, in dem bei jedem Event-Signal der Zählerstand des moduleigenen Timers gespeichert wird.

Das Modul hat 3 differentielle Event-Eingänge: EVENT/A, B, ENABLE, deren Signale das Modul zum resultierenden Event-Signal verarbeitet. Diese Vorverarbeitung der Signale ist konfigurierbar mit **P2_Event2_Config**.

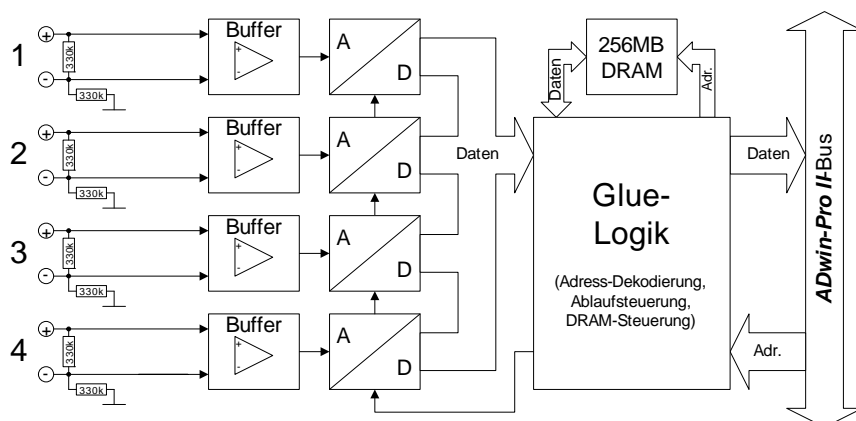


Abb. 50 – Pro II-AIn-F-4/16 Rev. E: Blockschaltbild

Eingangskanäle	4 differentiell	
Auflösung	16 Bit	
Konvertierungszeit	0,25µs (je ADC)	
Eingangs-Bandbreite	0 ... 600kHz	
Speichergröße	256MiB oder 2 ²⁷ = 134217728 Messwerte insgesamt	
Messbereich	±10V mit max. Offset 3,5V	
Verstärkung:	1, 2, 4, 8 per Software einstellbar	
Genauigkeit	INL	typisch ±1,2 LSB, max. ±5 LSB
	DNL	typisch ±0,5 LSB, max. ±1 LSB
Eingangswiderstand	330kΩ, ±2%	
Spannungsfestigkeit:	±20V	
Offsetfehler	abgleichbar	
Offsetdrift	±30ppm/°C vom Endwert	
Event-Eingang (nur D-Sub-Buchse)	3 differentiell; RS422/485 kompatibel (5V differentiell, 120 Ω Bus-Abschlusswiderstand)	
Steckerverbindung	4 LEMO-Buchsen, 2-polig optional: 37-polige D-Sub-Buchse optional: 4 BNC-Buchsen.	

Abb. 51 – Pro II-AIn-F-4/16 Rev. E: Spezifikation

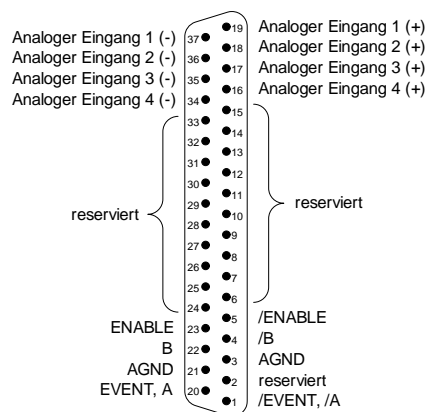


Abb. 52 – Pro-AIn-F-4/16-D Rev. E: Pinbelegung differenziell

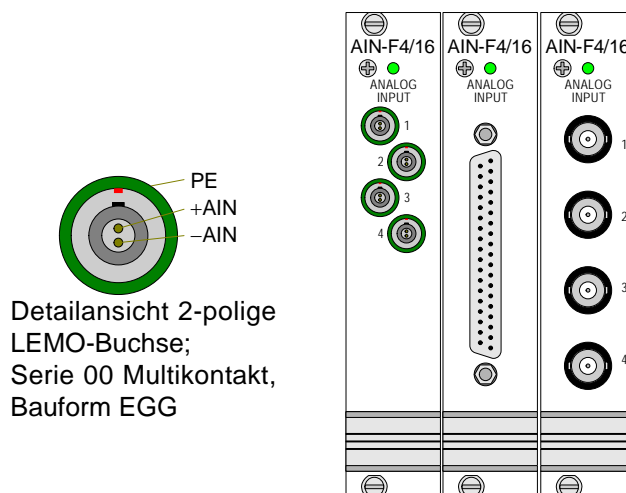


Abb. 53 – Pro II-AIn-F-4/16 Rev. E: Frontplatten

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ablaufsteuerung konfigurieren	<code>P2_ADCF_Mode</code> , <code>P2_Set_Average_Filter</code> , <code>P2_Set_Gain</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADCF</code> <code>P2_Start_ConvF</code> , <code>P2_Wait_EOCF</code> <code>P2_Read_ADCF</code>
Mehrere Messwerte lesen	<code>P2_Read_ADCF32</code> , <code>P2_Read_ADCF4</code> <code>P2_Read_ADCF4_Packed</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv32</code>
Burst-Messung durchführen	<code>P2_Burst_Init</code> <code>P2_Burst_Start</code> , <code>P2_Burst_Stop</code> <code>P2_Burst_Status</code> , <code>P2_Burst_Reset</code> <code>P2_Burst_Read</code> <code>P2_Burst_Read_Index</code> <code>P2_Burst_Read_Unpacked1/2/4</code> <code>P2_Burst_CRead_Unpacked1/2/4</code>

Bereich	Befehle
Wandlung synchronisieren	P2_Sync_All, P2_Sync_Enable P2_Sync_Mode, P2_Sync_Stat
Grenzwerte überwachen	P2_ADCF_Read_Limit P2_ADCF_Set_Limit P2_ADCF_Reset_Min_Max, P2_ADCF_Read_Min_Max4
LEDs einstellen	P2_Check_LED, P2_Set_LED
Interrupts und Event-Eingang einstellen	P2_Event_Enable, P2_Event_Read P2_Event_Config, P2_Event2_Config

5.4.8 Pro II-Aln-F-8/16 Rev. E

Analoges Eingangsmodul Pro II-Aln-F-8/16 Rev. E mit 8 Fast-ADC zu 16 Bit und 8 differentiellen Eingängen.

Für die Eingänge sind folgende Steckverbindungen verfügbar:

- Pro II-Aln-F-8/16-L2: geschirmte LEMO-Buchsen 2-polig, CAMAC Europannorm.
- Pro II-Aln-F-8/16-D: D-Sub-Buchse 37-polig.
- Pro II-Aln-F-8/16-B: BNC-Buchsen.

Die Wandler des Moduls arbeiten mit einer Abtastrate von bis zu 4MHz. Der moduleigene Speicher erlaubt, die anfallenden Daten – insbesondere bei Burst-Messungen – zwischenspeichern.

Das Modul verfügt über mehrere Betriebsarten für die Wandlung von analogen Signalen:

- Einzelmessung: Jede Messung wird im *ADbasic*-Programm einzeln gestartet, abgefragt und weiter verarbeitet.
- Einfache Burst-Messreihe: Das *ADbasic*-Programm startet eine vollständige Messreihe, also eine definierte Anzahl von Einzelmessungen.
- Kontinuierliche Burst-Messreihe: Das *ADbasic*-Programm startet eine Messreihe, die kontinuierlich Einzelmessungen ausführt, bis die Messreihe gestoppt wird. Die Daten werden in einem Ringspeicher abgelegt.

Das Modul Pro II-Aln-F-8/16 Rev. E hat einen Eingangs-Spannungsbereich von $\pm 10V$ und per Software programmierbare Verstärkung von 1, 2, 4 oder 8. Der Abgleich der Verstärkung und des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Das Modul kann optional anstelle einfacher Messwerte den Mittelwert aus 2...32 Messwerten zurückgeben. In diesem Fall wird die eingestellte Anzahl an Messwerten gewandelt und daraus der Mittelwert berechnet.

Burst-Messreihen

Das Modul führt Burst-Messreihen unabhängig vom Prozessormodul des *ADwin*-Systems aus. Die Messwerte – Anzahl und Messfrequenz sind vorab im Programm zu definieren – werden im Burst-Speicher auf dem Modul abgelegt. Das Prozessormodul liest nur noch die gespeicherten Messwerte (auch während der laufenden Messreihe) und verarbeitet sie.

Bei einer kontinuierlichen Burst-Messreihe muss die Messfrequenz mit der Auslesegeschwindigkeit abgestimmt werden. Hierbei sind verschiedene Aspekte von Bedeutung:

- Das Auslesen der Messwerte geschieht in Blöcken. Je größer der Datenblock ist, um so schneller ist der Lesevorgang im Mittel.

Beachten Sie: Das blockweise Lesen kann andere, auch hochpriorisierte Prozesse verzögern. Je größer der gelesene Block ist, umso eher kann eine Verzögerung entstehen.

- Der Zeitversatz zwischen kontinuierlicher Wandlung und blockweisem Lesen erfordert einen Datenpuffer. Hierzu muss bei der Initialisierung der Burst-Messung ein genügend großer Speicherbereich reserviert werden (**P2_BURST_INIT**, Parameter *samples*).

Event-Eingänge

Bei der Modulversion Pro II-Aln-F-8/16-D Rev. E (mit D-Sub-Buchse) kann eine Burst-Messreihe über externe Event-Signale gesteuert werden, so dass für jedes (resultierende) Event-Signal ein Messwert gespeichert wird.

Optional kann ein Kanal der Burst-Messreihe als Zeitkanal genutzt werden, in dem bei jedem Event-Signal der Zählerstand des moduleigenen Timers gespeichert wird.

Das Modul hat 3 differentielle Event-Eingänge: EVENT/A, B, ENABLE, deren Signale das Modul zum resultierenden Event-Signal verarbeitet. Diese Vorverarbeitung der Signale ist konfigurierbar mit **P2_Event2_Config**.

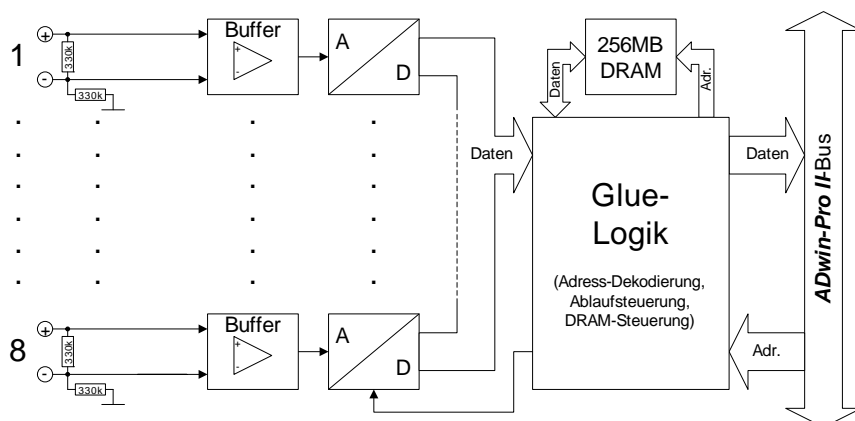


Abb. 54 – Pro II-AIn-F-8/16 Rev. E: Blockschaltbild

Eingangskanäle	8 differentiell
Auflösung	16 Bit
Konvertierungszeit	0,25µs (je ADC)
Eingangs-Bandbreite	0 ... 600kHz
Speichergröße	256MiB oder $2^{27} = 134217728$ Messwerte insgesamt
Messbereich	±10V mit max. Offset 3,5V
Verstärkung:	1, 2, 4, 8 per Software einstellbar
Genauigkeit	INL typisch ±1,2 LSB, max. ±5 LSB
	DNL typisch ±0,5 LSB, max. ±1 LSB
Eingangswiderstand	330kΩ, ±2%
Spannungsfestigkeit:	±20V
Offsetfehler	abgleichbar
Offsetdrift	±30ppm/°C vom Endwert
Event-Eingang (nur D-Sub-Buchse)	3 differentiell; RS422/485 kompatibel (5V differentiell, 120 Ω Bus-Abschlusswiderstand)
Steckerverbindung	8 LEMO-Buchsen, 2-polig optional: 37-polige D-Sub-Buchse optional: 8BNC-Buchsen.

Abb. 55 – Pro II-AIn-F-8/16 Rev. E: Spezifikation

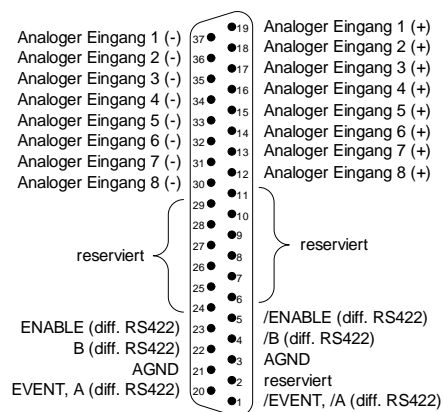


Abb. 56 – Pro II-AIn-F-8/16 Rev. E: Pinbelegung D-Sub-Buchse, differentiell

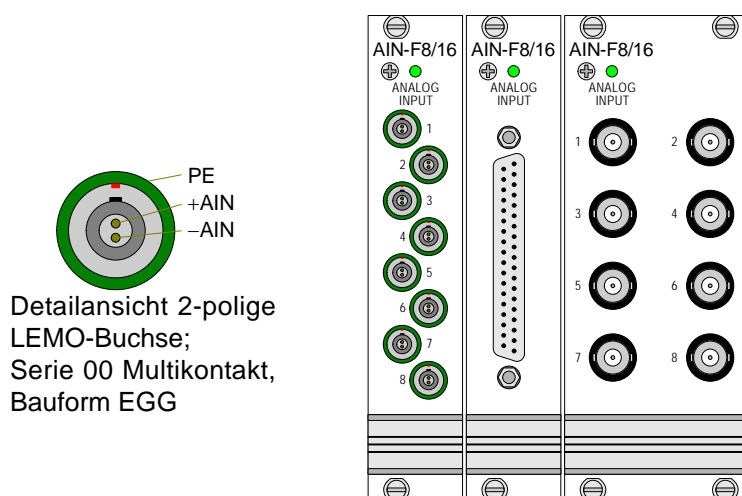


Abb. 57 – Pro II-AIn-F-8/16 Rev. E: Frontplatten

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ablaufsteuerung konfigurieren	<code>P2_ADCF_Mode</code> , <code>P2_Set_Average_Filter</code> , <code>P2_Set_Gain</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADCF</code> <code>P2_Start_ConvF</code> , <code>P2_Wait_EOCF</code> <code>P2_Read_ADCF</code>
Mehrere Messwerte lesen	<code>P2_Read_ADCF32</code> <code>P2_Read_ADCF4</code> , <code>P2_Read_ADCF8</code> <code>P2_Read_ADCF4_Packed</code> <code>P2_Read_ADCF8_Packed</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv32</code>

Bereich	Befehle
Burst-Messung durchführen	P2_Burst_Init P2_Burst_Start, P2_Burst_Stop P2_Burst_Status, P2_Burst_Reset P2_Burst_Read P2_Burst_Read_Index P2_Burst_Read_Unpacked1/2/4/8 P2_Burst_CRead_Unpacked1/2/4/8
Wandlung synchronisieren	P2_Sync_All, P2_Sync_Enable P2_Sync_Mode, P2_Sync_Stat
Grenzwerte überwachen	P2_ADCF_Read_Limit P2_ADCF_Set_Limit P2_ADCF_Reset_Min_Max, P2_ADCF_Read_Min_Max4, P2_ADCF_Read_Min_Max8
LEDs einstellen	P2_Check_LED, P2_Set_LED
Interrupts und Event-Eingang einstellen	P2_Event_Enable, P2_Event_Read P2_Event_Config, P2_Event2_Config

5.4.9 Pro II-Aln-F-4/18 Rev. E

Analoges Eingangsmodul Pro II-Aln-F-4/18 Rev. E mit 4 Fast-ADC zu 18 Bit und 4 differentiellen Eingängen. Die Eingänge sind voneinander und von anderen Modulen galvanisch getrennt.

Für die Eingänge sind folgende Steckverbindungen verfügbar:

- Pro II-Aln-F-4/18-L2: geschirmte LEMO-Buchsen 2-polig, CAMAC Europeanorm.
- Pro II-Aln-F-4/18-D: D-Sub-Buchse 37-polig.

Das Modul Pro II-Aln-F-4/18 Rev. E hat einen Eingangs-Spannungsbereich von bipolar $\pm 10V$. Der Abgleich des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

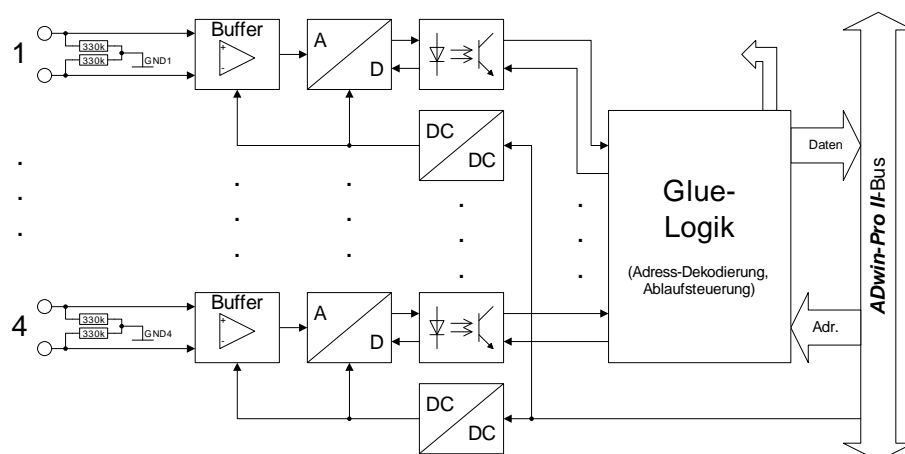


Abb. 58 – Pro II-Aln-F-4/18 Rev. E: Blockschaltbild

Eingangskanäle	4 differentiell, galvanisch getrennt
Auflösung	18 Bit
Wandlungszeit	max. 2 μ s (je ADC)
Abtastrate	max. 500ksps (je ADC)
Eingangsbandbreite	0 ... 0,6MHz
Messbereich	$\pm 10V$
Genauigkeit	INL max. ± 4 LSB
	DNL max. ± 3 LSB
Eingangswiderstand	330k Ω , $\pm 2\%$
Spannungsfestigkeit:	$\pm 35V$
Offsetfehler	abgleichbar
Offsetdrift	± 30 ppm/ $^{\circ}C$ vom Endwert
Steckerverbindung	4 LEMO-Buchsen optional: 37-polige D-Sub-Buchse

Abb. 59 – Pro II-Aln-F-4/18 Rev. E: Spezifikation

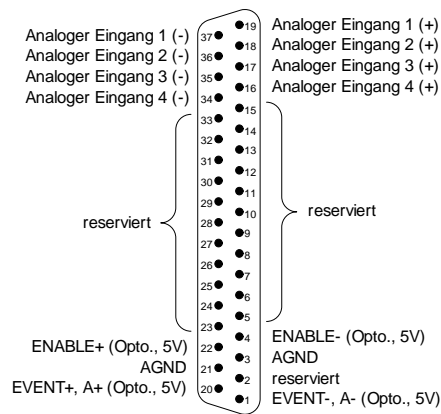


Abb. 60 – Pro II-AIn-F-4/18-D Rev. E: Pinbelegung differenziell

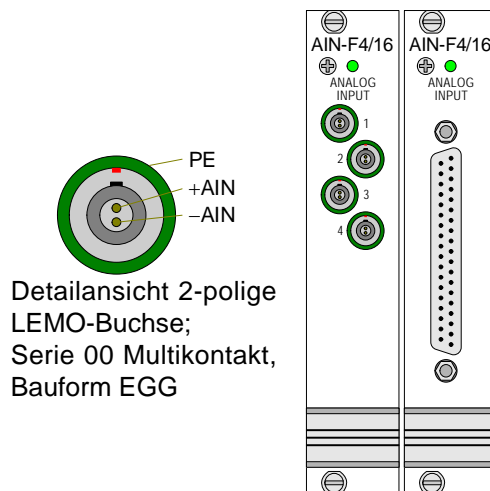


Abb. 61 – Pro II-AIn-F-4/18 Rev. E: Frontplatten

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ablaufsteuerung konfigurieren	<code>P2_ADCF_Mode</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADCF</code> , <code>P2_ADCF24</code> <code>P2_Start_ConvF</code> , <code>P2_Wait_EOCF</code> <code>P2_Read_ADCF</code> , <code>P2_Read_ADCF24</code>
Mehrere Messwerte lesen	<code>P2_Read_ADCF32</code> <code>P2_Read_ADCF4</code> , <code>P2_Read_ADCF4_24B</code> <code>P2_Read_ADCF4_Packed</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv24</code> <code>P2_Read_ADC_SConv32</code>
Wandlung synchronisieren	<code>P2_Sync_All</code> , <code>P2_Sync_Enable</code> <code>P2_Sync_Stat</code>
Grenzwerte überwachen	<code>P2_ADCF_Read_Limit</code> <code>P2_ADCF_Set_Limit</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> , <code>P2_Event_Read</code> <code>P2_Event_Config</code> , <code>P2_Event2_Config</code>

Programmierung

5.4.10 Pro II-Aln-F-8/18 Rev. E

Analoges Eingangsmodul Pro II-Aln-F-8/18 Rev. E mit 8 Fast-ADC zu 18 Bit und 8 differentiellen Eingängen. Die Eingänge sind voneinander und von anderen Modulen galvanisch getrennt.

Für die Eingänge sind folgende Steckverbindungen verfügbar:

- Pro II-Aln-F-8/18-L2: geschirmte LEMO-Buchsen 2-polig, CAMAC Europeanorm.
- Pro II-Aln-F-8/18-D: D-Sub-Buchse 37-polig.

Das Modul Pro II-Aln-F-8/18 Rev. E hat einen Eingangs-Spannungsbereich von bipolar $\pm 10V$. Der Abgleich des Offsets erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

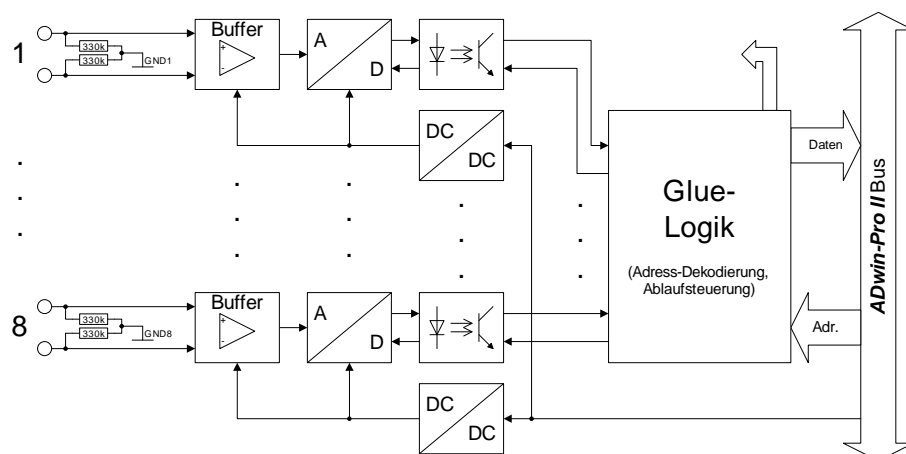


Abb. 62 – Pro II-Aln-F-8/18 Rev. E: Blockschaltbild

Eingangskanäle	8 differentiell, galvanisch getrennt
Auflösung	18 Bit
Wandlungszeit	max. 2µs (je ADC)
Abtastrate	max. 500ksps (je ADC)
Eingangsbandbreite	0 ... 0,6MHz
Messbereich	$\pm 10V$
Genauigkeit	INL max. ± 4 LSB
	DNL max. ± 3 LSB
Eingangswiderstand	330kΩ, $\pm 2\%$
Spannungsfestigkeit:	$\pm 35V$
Offsetfehler	abgleichbar
Offsetdrift	± 30 ppm/°C vom Endwert
Steckerverbindung	8 LEMO-Buchsen, 2-polig optional: 37-polige D-Sub-Buchse

Abb. 63 – Pro II-Aln-F-8/18 Rev. E: Spezifikation

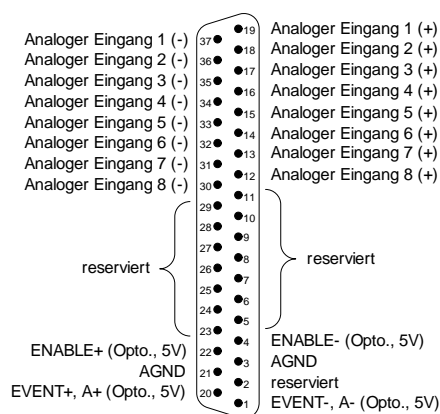


Abb. 64 – Pro II-AIn-F-8/18-D Rev. E: Pinbelegung differenziell

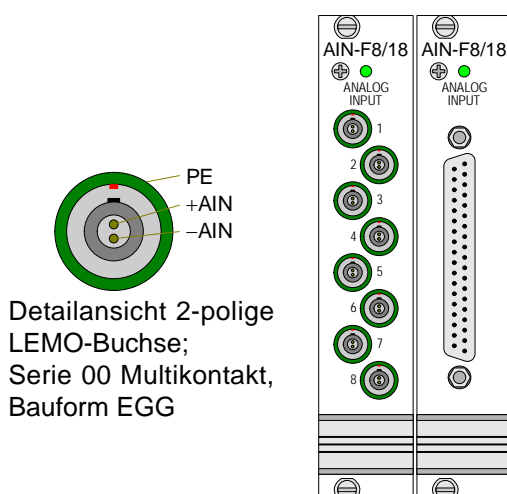


Abb. 65 – Pro II-AIn-F-8/18 Rev. E: Frontplatten

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ablaufsteuerung konfigurieren	<code>P2_ADCF_Mode</code>
Einzelmessung durchführen – vollständig oder schrittweise	<code>P2_ADCF</code> , <code>P2_ADCF24</code> <code>P2_Start_ConvF</code> , <code>P2_Wait_EOCF</code> <code>P2_Read_ADCF</code> , <code>P2_Read_ADCF24</code>
Mehrere Messwerte lesen	<code>P2_Read_ADCF32</code> <code>P2_Read_ADCF4</code> , <code>P2_Read_ADCF4_24B</code> <code>P2_Read_ADCF4_Packed</code> <code>P2_Read_ADCF8</code> , <code>P2_Read_ADCF8_24B</code> <code>P2_Read_ADCF8_Packed</code>
Messwert lesen und neue Messung starten	<code>P2_Read_ADC_SConv</code> <code>P2_Read_ADC_SConv24</code> <code>P2_Read_ADC_SConv32</code>
Wandlung synchronisieren	<code>P2_Sync_All</code> , <code>P2_Sync_Enable</code> <code>P2_Sync_Stat</code>
Grenzwerte überwachen	<code>P2_ADCF_Read_Limit</code> <code>P2_ADCF_Set_Limit</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>

Programmierung

Bereich	Befehle
Interrupts und Event-Eingang einstellen	P2_Event_Enable , P2_Event_Read P2_Event_Config , P2_Event2_Config

5.5 Pro II: Analoge Ausgangsmodule

Dieser Abschnitt beschreibt analoge Ausgangsmodule für **ADwin-Pro II**. Sie finden analoge Ausgangsmodule für **ADwin-Pro I** im Handbuch „ADwin-Pro-Hardware ab Seite 60.

Modulname	AOut 4/16	AOut 8/16
Revision	E	E
Anzahl DAC	4	8
Auflösung [Bit]	16	16
max. Einschwingzeit [μ s]	< 3	< 3
Kanäle sng. end.	4	8
Ausgangsspannung	± 10 V	± 10 V
Kalibrierung ^a	SW	SW
Seite	70	72

a. SW: per Software

5.5.1 Pro II-AOut-4/16 Rev. E

Das analoge Ausgangsmodul Pro II-AOut-4/16 Rev. E hat 4 DAC zu 16 Bit mit festem Tiefpass 1. Ordnung, um Störungen zu unterdrücken ($f_g = 10\text{MHz}$).

Der Ausgangs-Spannungsbereich der DAC ist fest auf $\pm 10\text{V}$ bipolar eingestellt und lässt sich nicht verändern. Der Abgleich der Verstärkung und des Offset erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Für die Ausgänge sind folgende Steckverbindungen verfügbar:

- Pro II-AOut-4/16-L: geschirmte LEMO-Buchsen, CAMAC Europeanorm.
- Pro II-AOut-4/16-D: D-Sub-Buchse 37-polig.

Module mit D-Sub-Buchse haben einen Event-Eingang; ein anliegendes Signal kann als Triggersignal an das Prozessormodul geleitet werden.

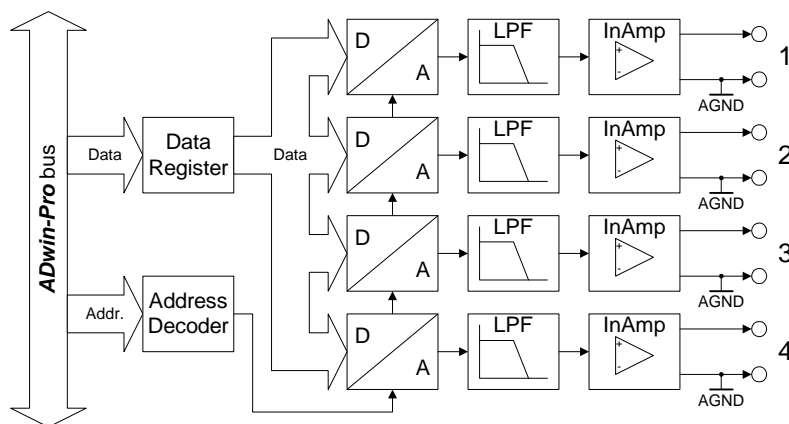


Abb. 66 – Pro II-AOut-4/16 Rev. E: Blockschaltbild

Ausgangskanäle	4 single ended
Auflösung	16 Bit
Einschwingzeit auf 0,01% FSR	$< 3\mu\text{s}$
Ausgangsspannung	$\pm 10\text{V}$
Maximaler Ausgangsstrom	$\pm 5\text{mA}$ pro Kanal für optimale Funktion $\pm 35\text{mA}$ technisch möglich, kurzschlussfest
Genauigkeit	INL ± 2 LSB typisch
	DNL ± 1 LSB typisch
Offsetfehler	abgleichbar
Verstärkungsfehler	abgleichbar
Offsetdrift	$\pm 10 \mu\text{V}/^\circ\text{C}$
Steckerverbindung	4 LEMO-Buchsen optional: 37-polige D-Sub-Buchse

Abb. 67 – Pro II-AOut-4/16 Rev. E: Spezifikation

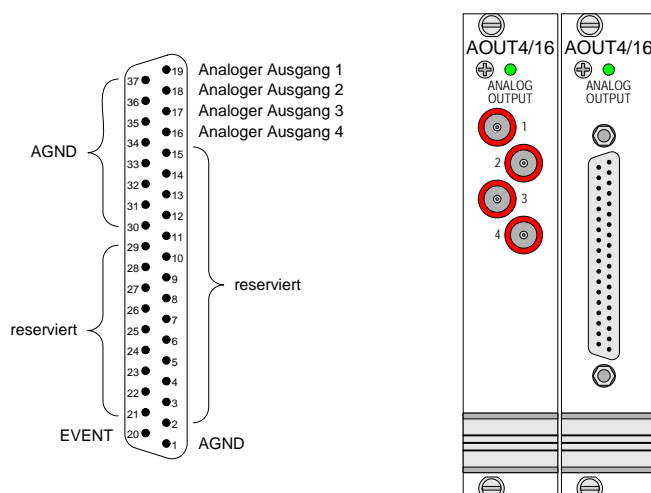


Abb. 68 – Pro II-AOut-4/16 Rev. E: Pinbelegung D-Sub und Frontplatten

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ausgabe durchführen	<code>P2_DAC</code> , <code>P2_DAC4</code> , <code>P2_DAC4_Packed</code>
Ausgabe schrittweise durchführen	<code>P2_Write_DAC</code> , <code>P2_Write_DAC4</code> <code>P2_Write_DAC4_Packed</code> <code>P2_Write_DAC32</code> <code>P2_Start_DAC</code>
Wandlung synchronisieren	<code>P2_Sync_All</code> , <code>P2_Sync_Enable</code> <code>P2_Sync_Stat</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_eVENT_eENABLE</code> , <code>P2_eVENT_rEAD</code> <code>P2_eVENT_CONFIG</code>

Programmierung

5.5.2 Pro II-AOut-8/16 Rev. E

Das analoge Ausgangsmodul Pro II-AOut-8/16 Rev. E hat 8 DAC zu 16 Bit mit festem Tiefpass 1. Ordnung, um Störungen zu unterdrücken ($f_g = 10\text{MHz}$).

Der Ausgangs-Spannungsbereich der DAC ist fest auf $\pm 10\text{V}$ bipolar eingestellt und lässt sich nicht verändern. Der Abgleich der Verstärkung und des Offset erfolgt per Software (siehe Kapitel 6 "Kalibrierung").

Für die Ausgänge sind folgende Steckverbindungen verfügbar:

- Pro II-AOut-8/16-L: geschirmte LEMO-Buchsen, CAMAC Europeanorm.
- Pro II-AOut-8/16-D: D-Sub-Buchse 37-polig.

Module mit D-Sub-Buchse haben einen Event-Eingang; ein anliegendes Signal kann als Triggersignal an das Prozessormodul geleitet werden.

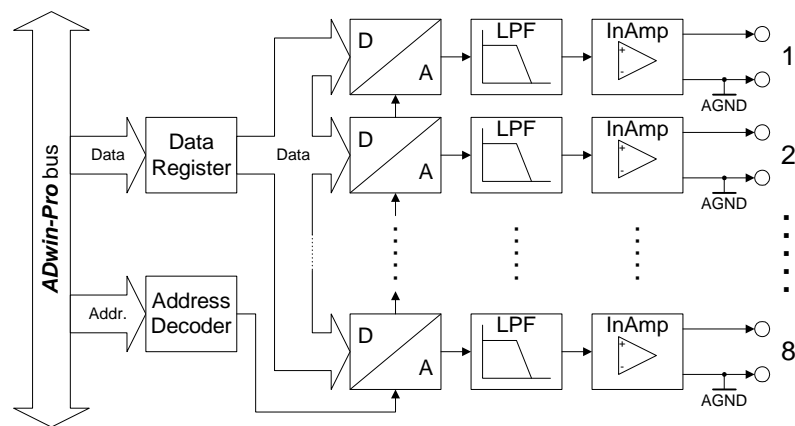


Abb. 69 – Pro II-AOut-8/16 Rev. E: Blockschaltbild

Ausgangskanäle	8 single ended
Auflösung	16 Bit
Einschwingzeit auf 0,01% FSR	< 3 μs
Ausgangsspannung	$\pm 10\text{V}$
Maximaler Ausgangsstrom	$\pm 5\text{mA}$ pro Kanal für optimale Funktion $\pm 35\text{mA}$ technisch möglich, kurzschlussfest
Genauigkeit	INL ± 2 LSB typisch
	DNL ± 1 LSB typisch
Offsetfehler	abgleichbar
Verstärkungsfehler	abgleichbar
Offsetdrift	$\pm 10 \mu\text{V}/^\circ\text{C}$
Steckerverbindung	8 LEMO-Buchsen optional: 37-polige D-Sub-Buchse

Abb. 70 – Pro II-AOut-8/16 Rev. E: Spezifikation

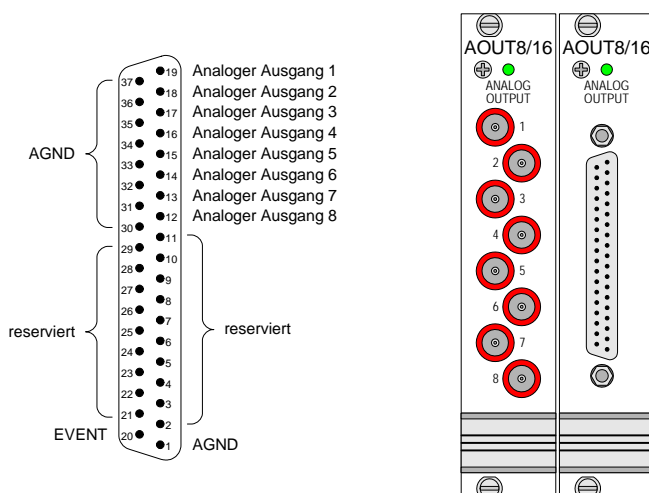


Abb. 71 – Pro II-AOut-8/16 Rev. E: Pinbelegung D-Sub und Frontplatten

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ausgabe durchführen	<code>P2_DAC</code> , <code>P2_DAC4</code> , <code>P2_DAC4_Packed</code> <code>P2_DAC8</code> , <code>P2_DAC8_Packed</code>
Ausgabe schrittweise durchführen	<code>P2_Write_DAC</code> , <code>P2_Write_DAC4</code> <code>P2_Write_DAC4_Packed</code> <code>P2_Write_DAC8</code> <code>P2_Write_DAC8_Packed</code> <code>P2_Write_DAC32</code> <code>P2_Start_DAC</code>
Wandlung synchronisieren	<code>P2_Sync_All</code> , <code>P2_Sync_Enable</code> <code>P2_Sync_Stat</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_eEVENT_eENABLE</code> , <code>P2_eEVENT_rEAD</code> <code>P2_eEVENT_cONFIG</code>

Programmierung

5.6 Pro II: Digital-IO-Module

Digital-IO-Module

Modulname	Rev.	Typ	Kanäle	U _{Ein} [V]		High Pegel [mA]	Isola- tion [V]	Seite
DIO-32	E	TTL-Ein-/Ausgang	32	5	TTL	–	–	75
DIO-32-TiCo	E	TTL-Ein-/Ausgang, mit TiCo-Prozessor	32	5	TTL	–	–	77
OPT-16	E	Optokoppler-Eingang	16	5, 12, 24	DC	–	42	80
OPT-32	E	Optokoppler-Eingang	32	5 / 12 / 24	DC	–	42	82
REL-16	E	Relais-Ausgang	16	max. 30	AC / DC	500	42	84
TRA-16	E	Transistor-Ausgang	16	5...30	DC	200	42	86
PWM-16	E	PWM-Ausgangssignal	16	5	TTL	–	–	88
PWM-16-I	E	PWM-Ausgangssignal	16	5...30	DC	100	42	88

Zähler-Module

Modulname	Rev.	Kanäle	Zähler			Eingangsspg. U _{Ein}		Isolation [V]	Seite
			Anzahl	Typ	Auflösg. [Bit]	[V]	Typ		
CNT-T	E	4	1	U	32	5	TTL	–	90
CNT-I	E	4	1	U	32	5, 12, 24	DC	42	90
CNT-D	E	4 + 2 SSI	1	U	32	5 diff.	RS422/ RS485	–	90

Der Zählertyp „Universalzähler“ (U) beinhaltet Vor-/Rückwärtszähler, Vierflanken-
auswertung und PWM-Zähler.

5.6.1 Pro II-DIO-32 Rev. E

Das digitale Ein-/Ausgangsmodul Pro II-DIO-32 Rev. E stellt 32 programmierbare Ein- und Ausgangskanäle mit TTL-Pegeln bereit. Die Kanäle können in Blöcken zu jeweils 8 Bit mit *ADbasic*-Befehlen als Ein- oder Ausgänge konfiguriert werden. Nach dem Einschalten sind alle Kanäle als Eingänge konfiguriert.

Das Modul kann mit einer Frequenz von 100MHz die Flanken an Eingangskanälen überwachen. Bei einer Änderung wird der aktuelle Pegelstand gemeinsam mit einem Zeitstempel in einem FIFO zwischengespeichert; es können bis zu 511 solcher Wertepaare (Pegelstand und Zeitstempel) gespeichert werden. Die FIFO-Daten können ausgelesen und weiter verarbeitet werden.

Außerdem kann abgefragt werden, an welchen Eingangskanälen eine positive oder negative Flanke aufgetreten ist.

Die Variante Pro II-DIO-32-TiCo Rev. E besitzt zusätzlich einen frei programmierbaren *TiCo*-Prozessor und kann selbstständig Pegel zu bestimmten Zeitpunkten auf Digitalausgängen ausgeben. Weiteres siehe Seite 77.

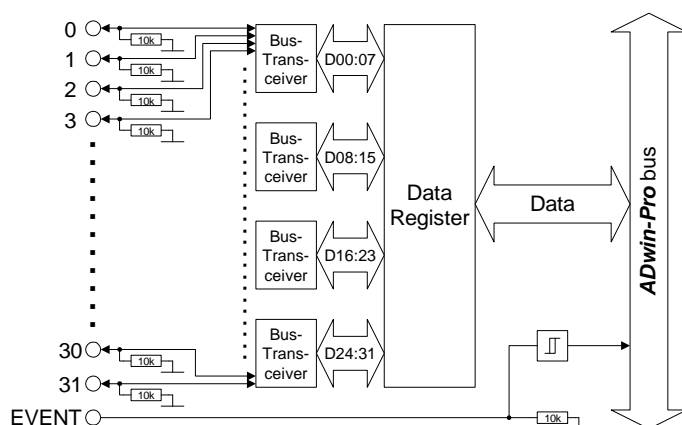


Abb. 72 – Pro II-DIO-32 Rev. E: Blockschaltbild

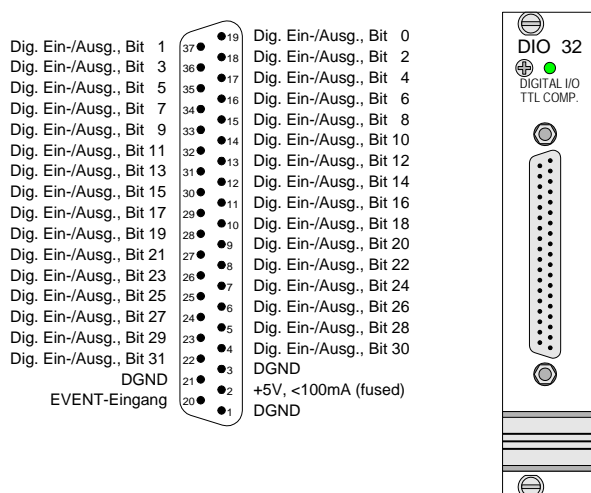


Abb. 73 – Pro II-DIO-32 Rev. E: Frontplatte und Pinbelegung

Ein-/Ausgangskanäle	32; in Blöcken zu 8 Bit als Ein-/Ausgang mittels Software einstellbar
Digitale Eingänge	TTL-Logik
Pull-Down-Widerstand	10kΩ

Abb. 74 – Pro II-DIO-32 Rev. E: Spezifikation

V _{IH}	min. 2V
V _{IL}	max. 0,8V
I _{IH}	max. 1 µA
I _{IL}	max. 0,01 mA
Spannungsbereich	-0,5V ... +5,5V
Ausgangsstrom	max. ±35mA pro Kanal, max. ±70mA je Block (8 Kanäle) über VCC oder GND
Event-Eingang	TTL-Logik
Power-Up-Status	Alle Kanäle als Eingänge
Steckerverbindung	37-polige D-Sub-Buchse

Abb. 74 – Pro II-DIO-32 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ein- und Ausgänge konfigurieren	<code>P2_DigProg</code>
Eingangssignale abfragen	<code>P2_Digin_Long</code>
Latch-Register nutzen	<code>P2_Dig_Latch</code> <code>P2_Dig_Read_Latch</code> <code>P2_Dig_Write_Latch</code> <code>P2_Sync_All</code>
Flanken an Eingangskanälen überwachen	<code>P2_Digin_FIFO_Enable</code> <code>P2_Digin_FIFO_Read</code> <code>P2_Digin_FIFO_Read_Timer</code> <code>P2_Digin_FIFO_Clear</code> <code>P2_Digin_FIFO_Full</code>
Flankenstatus abfragen	<code>P2_Digin_Edge</code>
Ausgangssignale setzen und rücklesen	<code>P2_Digout</code> , <code>P2_Digout_Bits</code> <code>P2_Digout_Long</code> <code>P2_Get_Digout_Long</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> , <code>P2_Event_Config</code> <code>P2_Event_Read</code>

5.6.2 Pro II-DIO-32-TiCo Rev. E

Das digitale Ein-/Ausgangsmodul Pro II-DIO-32-TiCo Rev. E stellt 32 programmierbare Ein- und Ausgangskanäle mit TTL-Pegeln bereit. Die Kanäle können in Blöcken zu jeweils 8 Bit mit *ADbasic*-Befehlen als Ein- oder Ausgänge konfiguriert werden. Nach dem Einschalten sind alle Kanäle als Eingänge konfiguriert.

Das Modul kann mit einer Frequenz von 100MHz die Flanken an Eingangskanälen überwachen. Bei einer Änderung wird der aktuelle Pegelstand gemeinsam mit einem Zeitstempel in einem FIFO zwischengespeichert; es können bis zu 511 solcher Wertepaare (Pegelstand und Zeitstempel) gespeichert werden. Die FIFO-Daten können ausgelesen und weiter verarbeitet werden.

Außerdem kann abgefragt werden, an welchen Eingangskanälen eine positive oder negative Flanke aufgetreten ist.

Im Unterschied zur Variante Pro II-DIO-32 Rev. E besitzt das Modul zusätzlich einen frei programmierbaren *TiCo*-Prozessor und 256MiByte Speicher. Der *TiCo*-Prozessor hat Zugriff auf alle digitalten Ein- und Ausgangskanäle. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

Ab Revision E 03 kann das Modul selbstständig Pegel zu bestimmten Zeitpunkten auf Digitalausgängen ausgeben. Ein FIFO dient als Zwischenspeicher für die vom Benutzer festgelegten Pegel und Zeitpunkte.

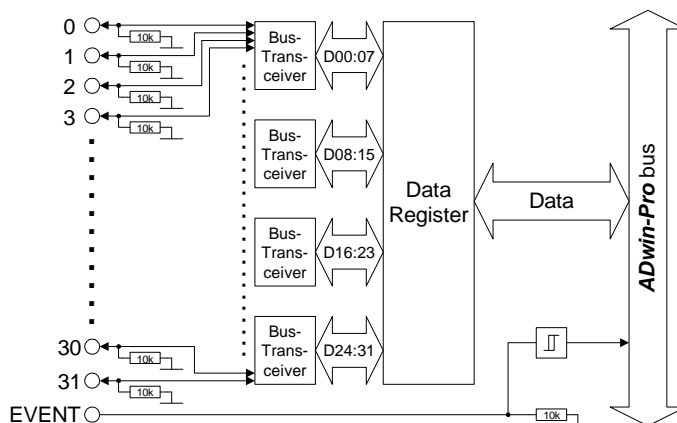


Abb. 75 – Pro II-DIO-32-TiCo Rev. E: Blockschaltbild

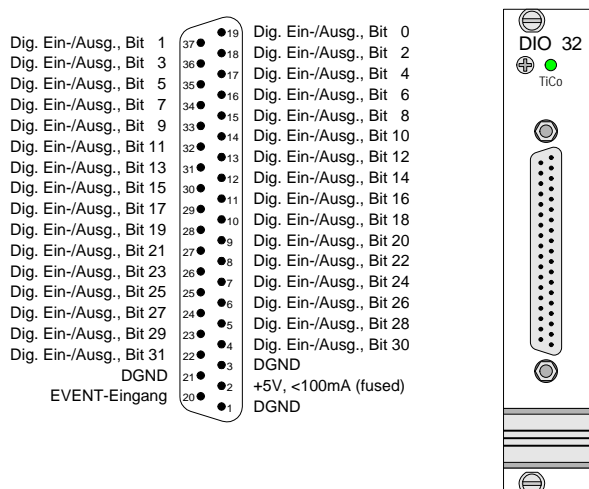


Abb. 76 – Pro II-DIO-32-TiCo Rev. E: Frontplatte und Pinbelegung

Ein-/Ausgangskanäle	32; in Blöcken zu 8 Bit als Ein-/Ausgang mittels Software einstellbar
Digitale Eingänge	TTL-Logik
Pull-Down-Widerstand	10kΩ
V _{IH}	min. 2V
V _{IL}	max. 0,8V
I _{IH}	max. 1μA
I _{IL}	max. 0,01 mA
Spannungsbereich	-0,5V ... +5,5V
Ausgangsstrom	max. ±35mA pro Kanal, max. ±70mA je Block (8 Kanäle) über VCC oder GND
Event-Eingang	TTL-Logik
Power-Up-Status	Alle Kanäle als Eingänge
Speichergröße	56KiByte intern, 256MiByte extern
Steckerverbindung	37-polige D-Sub-Buchse

Abb. 77 – Pro II-DIO-32-TiCo Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ein- und Ausgänge konfigurieren	<code>P2_DigProg</code>
Eingangssignale abfragen	<code>P2_Digin_Long</code>
Latch-Register nutzen	<code>P2_Dig_Latch</code> <code>P2_Dig_Read_Latch</code> <code>P2_Dig_Write_Latch</code> <code>P2_Sync_All</code>
Flanken an Eingangskanälen überwachen	<code>P2_Digin_FIFO_Enable</code> <code>P2_Digin_FIFO_Read</code> <code>P2_Digin_FIFO_Read_Timer</code> <code>P2_Digin_FIFO_Clear</code> <code>P2_Digin_FIFO_Full</code>
Flankenstatus abfragen	<code>P2_Digin_Edge</code>
Ausgangssignale setzen und rücklesen	<code>P2_Digout</code> , <code>P2_Digout_Bits</code> <code>P2_Digout_Long</code> <code>P2_Get_Digout_Long</code>
Ausgangssignale automatisch setzen	<code>P2_Digout_FIFO_Clear</code> <code>P2_Digout_FIFO_Empty</code> <code>P2_Digout_FIFO_Enable</code> <code>P2_Digout_FIFO_Read_Timer</code> <code>P2_Digout_FIFO_Start</code> <code>P2_Digout_FIFO_Write</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> , <code>P2_Event_Config</code> <code>P2_Event_Read</code>

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Programmierung in
TiCoBasic

Die Include-Datei `DIO32TiCo.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ein- und Ausgänge konfigurieren	DigProg
Eingangssignale abfragen	Digin_Long
Flanken an Eingangskanälen überwachen	Digin_FIFO_Enable Digin_FIFO_Read Digin_FIFO_Read_Timer Digin_FIFO_Clear Digin_FIFO_Full
Flankenstatus abfragen	Digin_Edge
Ausgangssignale setzen und rücklesen	Digout, Digout_Bits Digout_Set, Digout_Reset Digout_Long Get_Digout_Long
Ausgangssignale automatisch setzen	Digout_FIFO_Clear Digout_FIFO_Empty Digout_FIFO_Enable Digout_FIFO_Read_Timer Digout_FIFO_Start Digout_FIFO_Write
LEDs einstellen	Check_LED, Set_LED
Interrupts und Event-Eingang einstellen	Event_Enable, Trigger_Event Event_Config

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei `ADwinPro_All.inc` definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch mit dem <i>TiCo</i> -Prozessor über globale Variablen	P2_TDrv_Init P2_GetData_Long, P2_Get_Par, P2_Get_Par_Block P2_SetData_Long, P2_Set_Par, P2_Set_Par_Block P2_Get_TiCo_RingBuffer, P2_Set_TiCo_RingBuffer P2_RingBuffer_Empty P2_RingBuffer_Full
<i>TiCo</i> -Prozessor steuern	P2_TiCo_Reset, P2_TiCo_Start, P2_TiCo_Stop P2_Get_TiCo_Bootloader_ Status P2_Get_TiCo_Status, P2_Workload
<i>TiCo</i> -Prozesse steuern	P2_Process_Status P2_TiCo_Get_Processdelay P2_TiCo_Set_Processdelay P2_TiCo_Start_Process P2_TiCo_Stop_Process
<i>TiCo</i> -Programme übertragen	P2_TiCo_Flash, P2_TiCo_Load

Programmierung TiCo-Zugriff

5.6.3 Pro II-OPT-16 Rev. E

Das Eingangsmodul Pro II-OPT-16 Rev. E stellt 16 Kanäle mit optisch isolierten digitalen Eingängen bereit. Die Eingangs-Spannungsbereiche sind für jeden Eingang separat über Jumper einstellbar (5V, 12V, 24V). Die Voreinstellung ist 24V. Die Schaltzeit von nur 100ns erlaubt das Einlesen von schnellen digitalen Signalen.

Das Modul kann automatisch die Flanken an Eingangskanälen überwachen. Bei einer Änderung wird der aktuelle Pegelstand gemeinsam mit einem Zeitstempel in einem FIFO zwischengespeichert. Die FIFO-Daten können ausgelesen und weiter verarbeitet werden.

Jeder Kanal ist vom Systemstromkreis und von den anderen Eingängen optisch isoliert, wie auch der Event-Eingang.

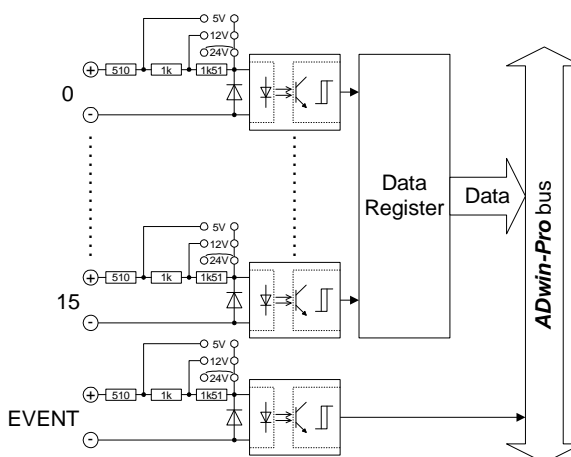


Abb. 78 – Pro II-OPT-16 Rev. E: Blockschaltbild

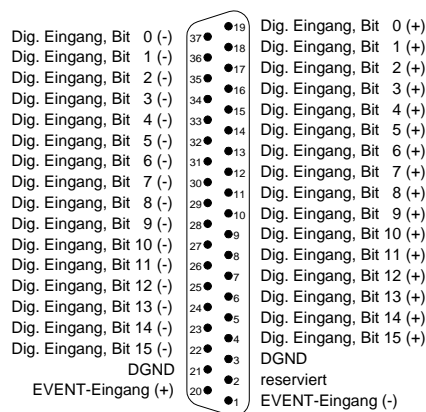


Abb. 79 – Pro II-OPT-16 Rev. E: Pinbelegung

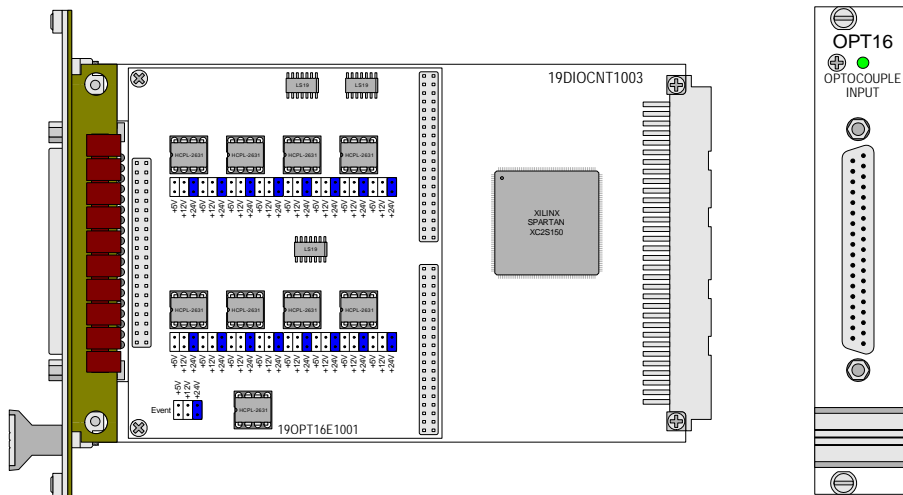


Abb. 80 – Pro II-OPT-16 Rev. E: Platine und Frontplatte

Eingangskanäle	16		
Event-Eingänge	1		
Eingangsstrom	typ. 3,5mA / max. 7,5mA		
Eingangs-Spannungsbereich (über Jumper wählbar)	0...5V	0...12V	0...24V
Schaltsschwelle für 0-low	0...0,8V	0...1,6V	0...3,2V
Schaltsschwelle für 1-high	4,5...5V	10...12V	20...24V
Spannungsfestigkeit	-5V ... 8V	-5V ... 16V	-5V ... 30V
Schaltzeit	100ns		
Isolation	42V Kanal zu Kanal / Kanal zu Masse		
Steckerverbindung	37-polige D-Sub-Buchse		

Abb. 81 – Pro II-OPT-16 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Eingangssignale abfragen	<code>P2_Digin_Edge</code> <code>P2_Digin_Long</code>
Latch-Register nutzen	<code>P2_Dig_Latch</code> , <code>P2_Sync_All</code> <code>P2_Dig_Read_Latch</code>
Flanken an Eingangskanälen überwachen	<code>P2_Digin_FIFO_Enable</code> <code>P2_Digin_FIFO_Read</code> <code>P2_Digin_FIFO_Read_Fast</code> <code>P2_Digin_FIFO_Read_Timer</code> <code>P2_Digin_FIFO_Clear</code> <code>P2_Digin_FIFO_Full</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> <code>P2_Event_Config</code> <code>P2_Event_Read</code>

5.6.4 Pro II-OPT-32 Rev. E

Das Eingangsmodul Pro II-OPT-32 Rev. E stellt 32 Kanäle mit optisch isolierten digitalen Eingängen bereit.

Der Eingangs-Spannungsbereich ist fest eingestellt (Bestelloptionen: 5V, 12V, 24V) und gilt auch für den Event-Eingang. Die Schaltzeit von nur 100ns erlaubt das Einlesen von schnellen digitalen Signalen.

Das Modul kann automatisch die Flanken an Eingangskanälen überwachen. Bei einer Änderung wird der aktuelle Pegelstand gemeinsam mit einem Zeitstempel in einem FIFO zwischengespeichert. Die FIFO-Daten können ausgelesen und weiter verarbeitet werden.

Jeder Kanal ist vom Systemstromkreis und von den anderen Eingängen optisch isoliert, wie auch der Event-Eingang.

Die Kanäle sind als single-ended ausgeführt und haben ein gemeinsames Massepotenzial (ext. GND), das auf der D-SUB-Buchse bereitgestellt werden muss.

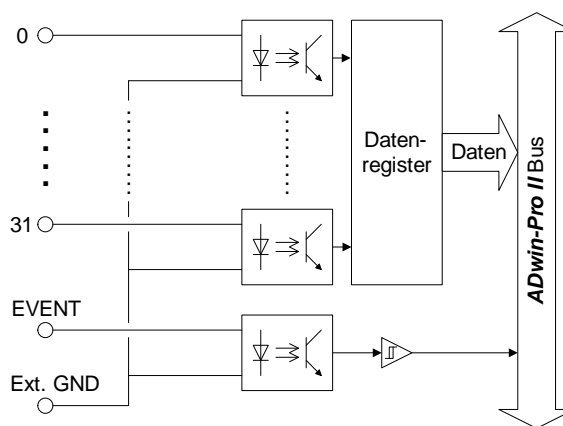


Abb. 82 – Pro II-OPT-32 Rev. E: Blockschaltbild

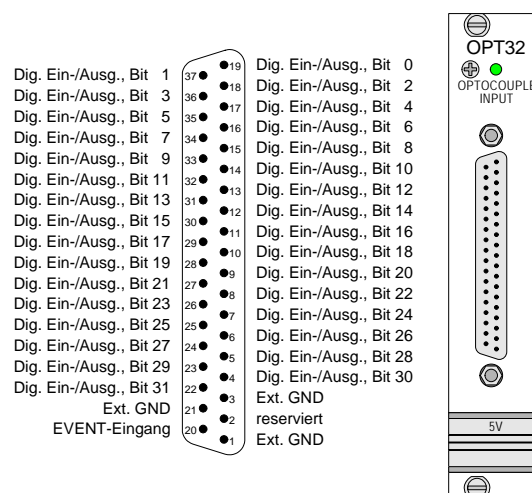


Abb. 83 – Pro II-OPT-32 Rev. E: Pinbelegung und Frontplatte

Eingangskanäle	32 single-ended		
Event-Eingänge	1		
Eingangsstrom	typ. 3,5mA / max. 7,5mA		
Eingangs-Spannungsbereich (Bestelloption)	0...5V	0...12V	0...24V
Schaltswelle für 0-low	0...0,8V	0...1,6V	0...3,2V
Schaltswelle für 1-high	4,5...5V	10...12V	20...24V
Spannungsfestigkeit	-5V ... 8V	-5V ... 16V	-5V ... 30V
Schaltzeit	100ns		
Isolation	42V Kanal zu Kanal / Kanal zu Masse		
Steckerverbindung	37-polige D-Sub-Buchse		

Abb. 84 – Pro II-OPT-32 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert.

Die Datei `ADwinPRO_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Eingangssignale abfragen	<code>P2_Digin_Edge</code> <code>P2_Digin_Long</code>
Latch-Register nutzen	<code>P2_Dig_Latch</code> , <code>P2_Sync_All</code> <code>P2_Dig_Read_Latch</code>
Flanken an Eingangskanälen überwachen	<code>P2_Digin_FIFO_Enable</code> <code>P2_Digin_FIFO_Read</code> <code>P2_Digin_FIFO_Read_Fast</code> <code>P2_Digin_FIFO_Read_Timer</code> <code>P2_Digin_FIFO_Clear</code> <code>P2_Digin_FIFO_Full</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> <code>P2_Event_Config</code> <code>P2_Event_Read</code>

Die Befehle sind im Handbuch Pro-Software und in der Online-Hilfe erläutert.

5.6.5 Pro II-REL-16 Rev. E

Das Ausgangsmodul Pro II-REL-16 Rev. E stellt 16 isolierte Relaisausgänge bereit. Jeder Kanal ist vom System und den anderen Kanälen getrennt. Der Event-Eingang ist optisch vom Systemstromkreis isoliert.

Das Modul ist mit Schließern bestückt. Optional ist das Modul auch mit Öffnern erhältlich.

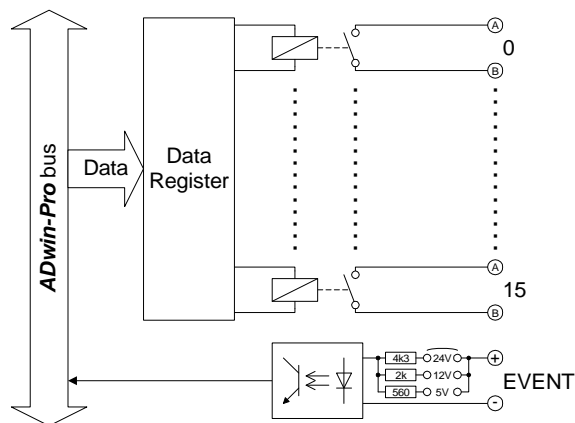


Abb. 85 – Pro II-REL-16 Rev. E: Blockschaltbild

Relais 0 A	37	●	19	Relais 0 B
Relais 1 A	36	●	18	Relais 1 B
Relais 2 A	35	●	17	Relais 2 B
Relais 3 A	34	●	16	Relais 3 B
Relais 4 A	33	●	15	Relais 4 B
Relais 5 A	32	●	14	Relais 5 B
Relais 6 A	31	●	13	Relais 6 B
Relais 7 A	30	●	12	Relais 7 B
Relais 8 A	29	●	11	Relais 8 B
Relais 9 A	28	●	10	Relais 9 B
Relais 10 A	27	●	9	Relais 10 B
Relais 11 A	26	●	8	Relais 11 B
Relais 12 A	25	●	7	Relais 12 B
Relais 13 A	24	●	6	Relais 13 B
Relais 14 A	23	●	5	Relais 14 B
Relais 15 A	22	●	4	Relais 15 B
DGND	21	●	3	DGND
EVENT-Eingang (+)	20	●	2	reserviert
		●	1	EVENT-Eingang (-)

Abb. 86 – Pro II-REL-16 Rev. E: Pinbelegung

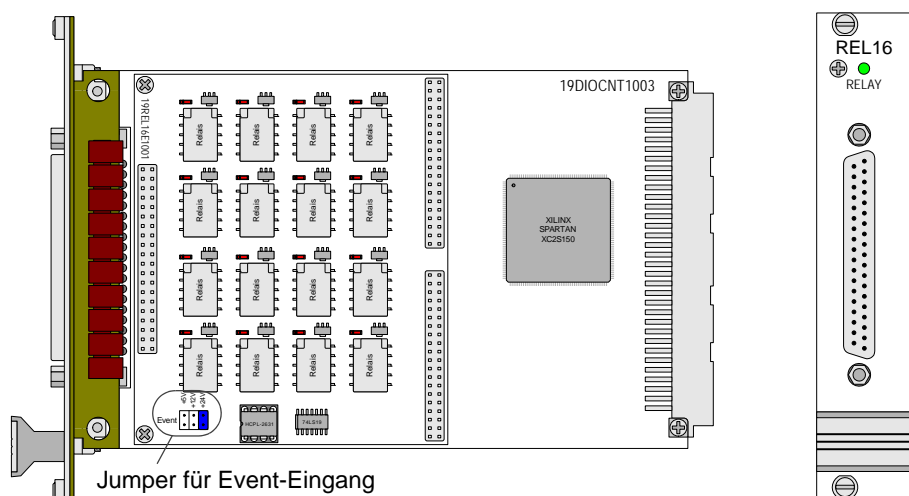


Abb. 87 – Pro II-REL-16 Rev. E: Platine und Frontplatte

Ausgangskanäle	16
Kontaktspannung	30V AC/DC Maximum
Kontaktstrom	max. 500mA pro Kanal
Kontaktausführung	1 Schließer pro Kanal, (optional: Öffner)
Ansprechzeit	4ms
Abfallzeit	3ms
Prellzeit	2ms
Event-Eingänge	1
Isolation	42V Kanal zu Kanal/ Kanal zu Masse
Event-Eingangsspannung	5V, 12V, 24V (über Jumper wählbar)
Power-Up-Status	low (mit Schließern: offen / mit Öffnern: geschlossen)
Steckerverbindung	37-polige D-Sub-Buchse

Abb. 88 – Pro II-REL-16 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ausgangssignale setzen und rücklesen	<code>P2_Digout</code> , <code>P2_Digout_Long</code> <code>P2_Digout_Bits</code> <code>P2_Get_Digout_Long</code>
Latch-Register nutzen	<code>P2_Dig_Latch</code> , <code>P2_Sync_All</code> <code>P2_Dig_Write_Latch</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> <code>P2_Event_Config</code> <code>P2_Event_Read</code>

5.6.6 Pro II-TRA-16 Rev. E

Das Ausgangsmodul Pro II-TRA-16 Rev. E stellt 16 galvanisch getrennte Transistor-Schaltausgänge bereit. Es gibt 2 Modulvarianten:

- Pro II-TRA-16 Rev. E: Die Ausgänge schalten nach V_{CC} .
- Pro II-TRA-16-G Rev. E: Die Ausgänge schalten nach Masse.

Die Schaltspannung V_{CC} muss durch eine externe Spannungsversorgung zugeführt werden.

Die Kanäle wie auch der Event-Eingang sind optisch vom System-Stromkreis isoliert.

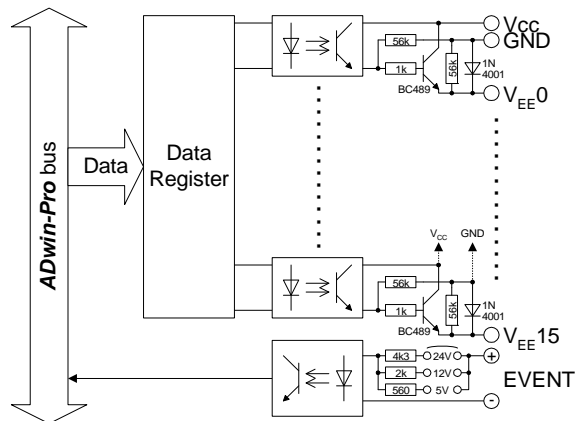


Abb. 89 – Pro II-TRA-16 Rev. E: Blockschaltbild

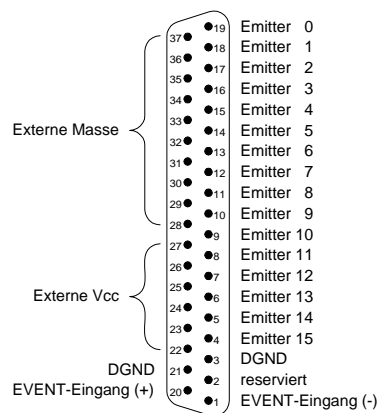


Abb. 90 – Pro II-TRA-16 Rev. E: Pinbelegung

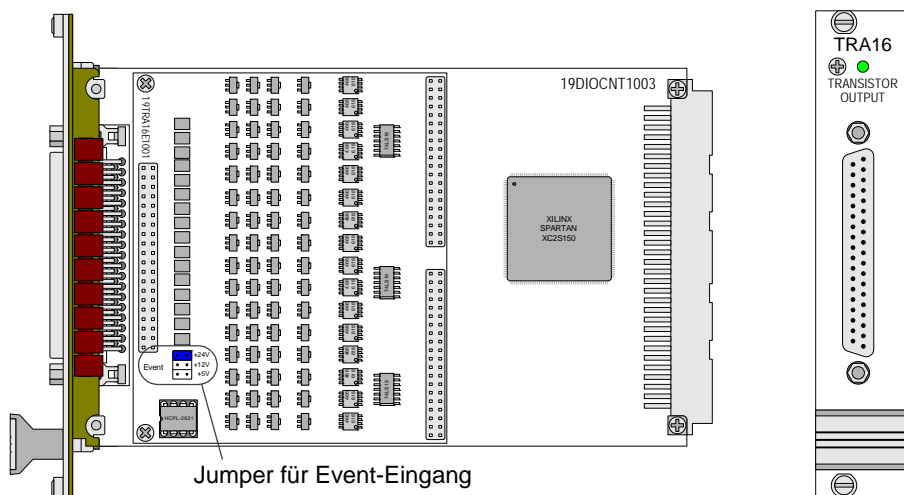


Abb. 91 – Pro II-TRA-16 Rev. E: Platine und Frontplatte

Ausgangskanäle	16
Schaltspannung V_{CC}	5...30V DC durch externe Spannungsversorgung
Schaltstrom	Pro II-TRA-16: 200mA max. pro Kanal Pro II-TRA-16-G: 100mA max. pro Kanal
Spannungsabfall	0,5V
Schaltzeit	2,5µs
Event-Eingang	1
Isolation	42V Kanal zu Kanal / Kanal zu Masse
Event-Eingangsspannung	5V, 12V, 24V (über Jumper wählbar)
Power-Up-Status	low (GND extern)
Steckerverbindung	37-polige D-Sub-Buchse

Abb. 92 – Pro II-TRA-16 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Ausgangssignale setzen und rücklesen	P2_Digout , P2_Digout_Long P2_Digout_Bits P2_Get_Digout_Long
Latch-Register nutzen	P2_Dig_Latch , P2_Sync_All P2_Dig_Write_Latch
LEDs einstellen	P2_Check_LED , P2_Set_LED
Interrupts und Event-Eingang einstellen	P2_Event_Enable P2_Event_Config P2_Event_Read

5.6.7 Pro II-PWM-16 Rev. E

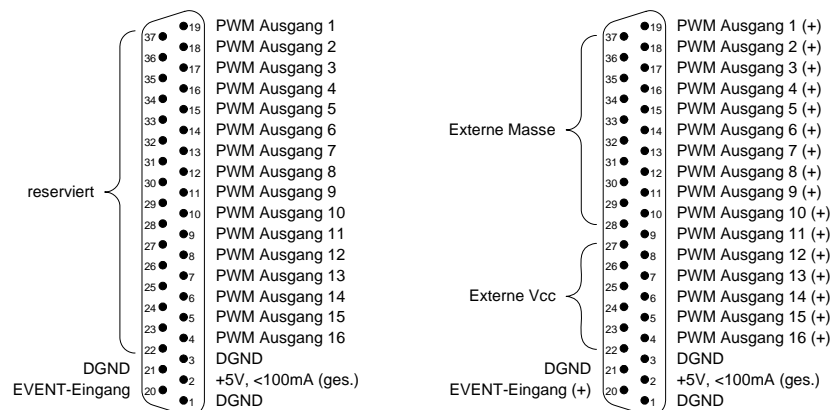
Das Modul Pro II-PWM-16 Rev. E gibt auf 16 Ausgängen pulsweitenmodulierte Signale (PWM-Signale) aus. Die (PWM-)Signale sind getrennt voneinander über Software konfigurierbar, d.h. sie können getrennt voneinander eingestellt werden.

Es gibt das Modul auch in der Variante Pro II-PWM-16-I Rev. E. Dort sind die Ausgänge gegen den Systemstromkreis und die anderen Ausgänge optisch isoliert. Auch der Event-Eingang ist gegen den Systemstromkreis isoliert.

Die Ausgabesignale werden mit einem Referenztakt von 100MHz getaktet.

Die niedrigste Ausgangsfrequenz beträgt ca. 0,6 Hz.

Die höchste Ausgangsfrequenz, bei der das Tastverhältnis noch in 1%-Schritten einstellbar ist, beträgt 1000kHz.



Pro II-PWM-16 Rev. E

Pro II-PWM-16-I Rev. E

Abb. 93 – Pro II-PWM-16 Rev. E: Pinbelegung

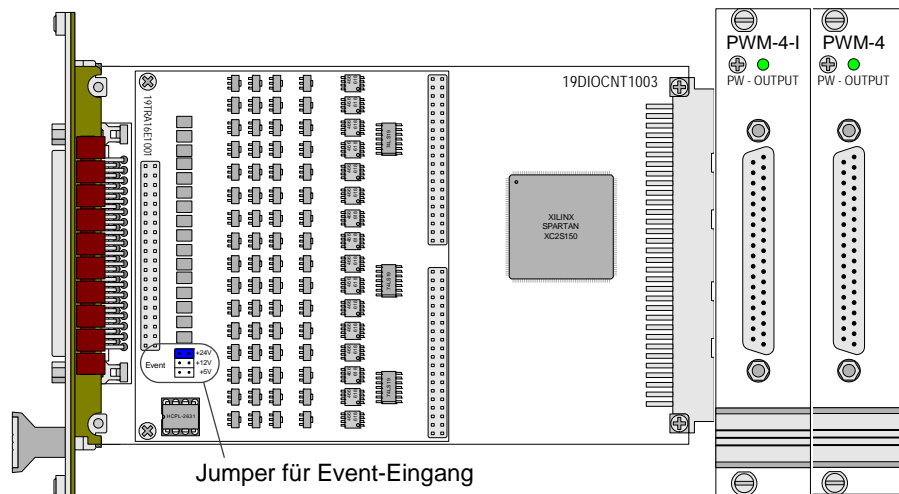


Abb. 94 – Pro II-PWM-16-I Rev. E: Platine und Frontplatte

Ausgangskanäle	16 PWM-Kanäle
Ausgänge	TTL
Zähler-/Registerbreite	32 Bit
Referenztakt	100MHz
Event-Eingang	1 Eingang, positiv TTL

Abb. 95 – Pro II-PWM-16 Rev. E: Spezifikation

Steckerverbindung	37-polige D-Sub-Buchse
Pro II-PWM-16 Rev. E	
V _{OH}	2,4V min.
V _{OL}	0,8V max.
Ausgangsstrom	5mA pro Kanal max.
Isolation	Nein
Pro II-PWM-16-I Rev. E	
Ausgangs-Spannung	5...30V DC durch externe Spannungsversorgung
Ausgangsstrom	100mA max. pro Kanal
Spannungsabfall	0,5V max.
Schaltzeit	ohne Last: 3µs mit Last: 1...2µs
Event-Eingangsspannung	5V, 12V, 24V (über Jumper wählbar)
Isolation	42V Kanal zu Kanal / Kanal zu Masse

Abb. 95 – Pro II-PWM-16 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Modul initialisieren	<code>P2_PWM_Init</code> <code>P2_PWM_Standby_Value</code> <code>P2_PWM_Enable</code> <code>P2_PWM_Reset</code>
PWM-Signale ausgeben und rücklesen	<code>P2_PWM_Write_Latch</code> <code>P2_PWM_Latch, P2_Sync_All</code> <code>P2_PWM_Get_Status</code>
LEDs einstellen	<code>P2_Check_LED, P2_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_Event_Enable</code> <code>P2_Event_Config</code> <code>P2_Event_Read</code>

5.6.8 Pro II-CNT-x Rev. E

Das Modul Pro II-CNT-x Rev. E stellt 4 konfigurierbare Zählerblöcke zur Verfügung. Ein Zählerblock enthält zwei 32 Bit-Zähler: Einerseits einen Vor-/Rückwärtszähler mit Takt/Richtungs-Auswertung oder Vierflankenauswertung zum Anschluss von Encodern. Zum anderen einen Zähler zur Periodendauer- und Tastverhältnismessung. Beide Zähler eines Blocks können parallel genutzt werden.

Es gibt folgende Modulversionen:

- *Pro II-CNT-T*: Zählereingänge mit TTL-Logik. Das Modul besitzt einen *TiCo*-Prozessor.
- *Pro II-CNT-D*: Zählereingänge differentiell; das Modul enthält zusätzlich 2 SSI-Decoder und einen *TiCo*-Prozessor.
- *Pro II-CNT-I*: Zählereingänge sind optisch isoliert. Das Modul besitzt einen *TiCo*-Prozessor.

Die Zähler-Eingänge sind gegen den Systemstromkreis und gegeneinander optisch isoliert. Auch der Event-Eingang ist gegen den Systemstromkreis isoliert.

Der Eingangs-Spannungsbereich der Zählereingänge und des Event-Eingangs kann mit Hilfe von Jumpers auf 0...5V, 0...12V oder 0...24V eingestellt werden. Die Voreinstellung ist 0...24V.

Der *TiCo*-Prozessor ist frei programmierbar und hat Zugriff auf alle Zählerfunktionen. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

Vor-/Rückwärtszähler

Der Vor-/ Rückwärtszähler eines Blocks kann in 2 Betriebsarten arbeiten:

- Takt/Richtungs-Auswertung (CLK- und DIR-Signale)

Eine negative Flanke am CLK-Eingang löst einen Zählimpuls am 32 Bit-Zähler aus. Das DIR-Signal bestimmt die Zählrichtung des Zählers, TTL high bedeutet Hochzählen, TTL low bedeutet Herunterzählen.

Sie können die Signale an den Eingängen CLK und DIR per Software (Befehl **P2_Cnt_Mode**) invertieren und damit sowohl die auslösende Flanke als auch die Zählrichtung ändern.

Sie können den Zählerstand programmgesteuert ins Latch übernehmen oder den Zähler durch ein externes CLR-/LATCH-Signal beeinflussen.

Das CLR-/LATCH-Signal kann je nach Programmierung ein Löschen (CLR) des Zählerstands oder die Übernahme des Zählerstands ins Latch (LATCH) bewirken. Diese Funktion wird erst wirksam, wenn sie durch den Befehl **P2_CNT_CLEAR_ENABLE** oder **P2_CNT_LATCH_ENABLE** freigegeben ist.

Das Löschen oder Latchen des Zählers erfolgt bei einer steigenden Flanke am Eingang CLR/LATCH. Beim Latchen lässt sich aus der Differenz von zwei gelesenen Latch-Werten die Frequenz der Messung ermitteln, denn die Differenz gibt die Anzahl der Impulse zwischen den beiden Lesevorgängen an.

- Vierflankenauswertung (A- und B-Signale)

Die Vierflankenauswertung wandelt die (möglichst um 90° phasenverschobenen) Signale eines angeschlossenen Inkremental-Encoders an A- und B-Eingang in ein CLK- und DIR-Signal um. Hierzu sind die Eingänge in *ADbasic* entsprechend zu programmieren (siehe „ADwin-Pro Systembeschreibung, Programmierung in *ADbasic*“).

Da jede Flanke des A- und B-Signales einen Zählimpuls erzeugt, wird die Auflösung um den Faktor 4 vergrößert. Besitzt der Encoder ein Referenz-Signal, so kann dies (nach Freigabe des CLR- bzw. LATCH-Einganges) zum Löschen oder Latchen des Zählers genutzt werden. Das Löschen des Zählers erfolgt, wenn die Signale A, B und CLR auf logisch „1“ stehen (über Software umstellbar: Löschen, wenn nur das CLR-Signal auf logisch „1“ steht).

Der PWM-Zähler des Zählerblocks wertet die Signale an den PWM-Eingängen aus. Mit Software-Befehlen können folgende Daten direkt ausgelesen werden:

- Frequenz und Tastverhältnis
- Eintastzeit und Austastzeit

Dieser Eingang kann, sofern er freigegeben wurde, einen extern getriggerten ADbasic-Prozess starten.

PWM-Zähler

EVENT-Eingang

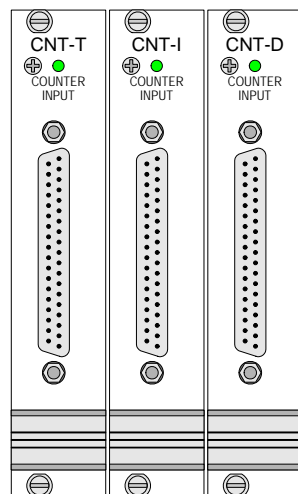


Abb. 96 – Pro II-CNT-x Rev. E: Frontplatten

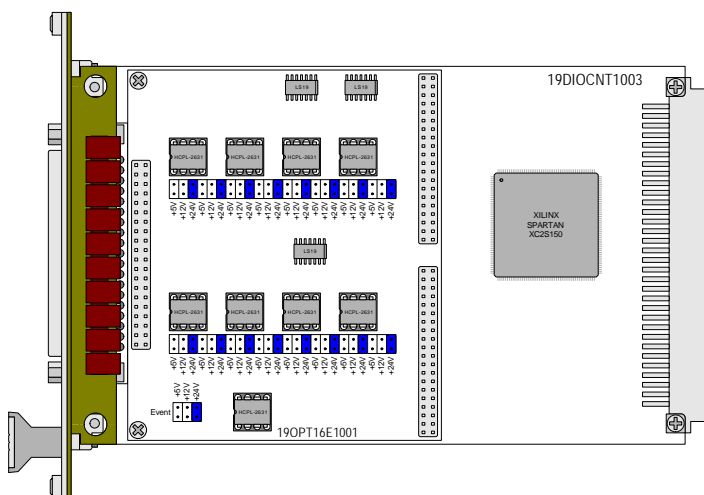
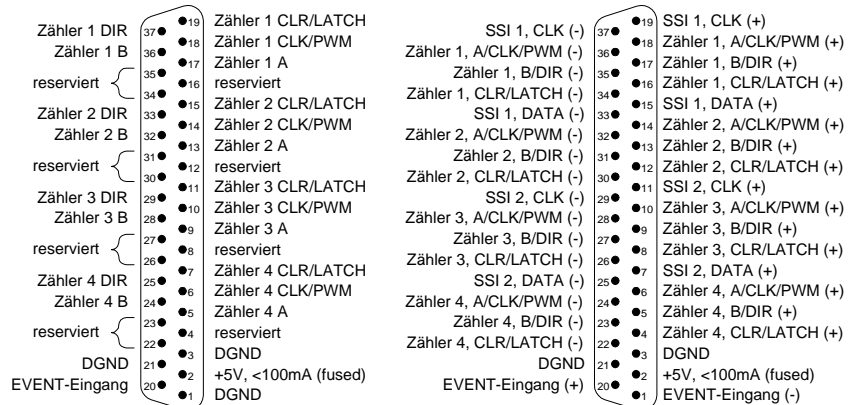
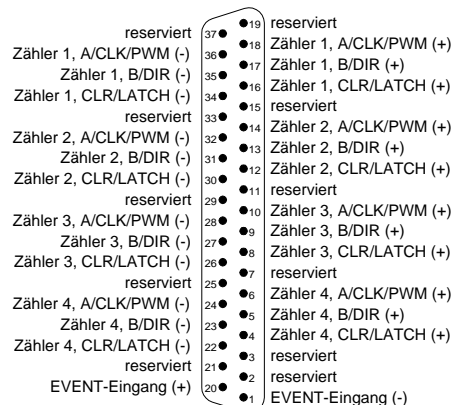


Abb. 97 – Pro II-CNT-I Rev. E: Platine mit Jumpers



Pinbelegung Pro II-CNT-T

Pinbelegung Pro II-CNT-D



Pinbelegung Pro II-CNT-I

Zähler	4 Universalzähler CNT-D: zusätzl. 2 SSI-Decoder
Zählerbreite	32 Bit
Eventeingang	1
Referenztakt	50MHz
Taktfrequenz Vierflankenauswertung	12,5MHz max. (bei 90° Phasenverschiebung der Signale)
Taktfrequenz Vor- / Rückwärtszähler	CNT-T: 25MHz max. CNT-I und CNT-D: 15MHz max.
Referenzfrequenz PWM-Analyse	100MHz
Steckerverbindung	37-pol. D-Sub-Buchse
Strombedarf des Moduls	CNT-T ca. 150mA CNT-I ca. 200mA CNT-D ca. 200mA

Abb. 98 – Pro II-CNT-x Rev. E: Allg. Spezifikation

Pro II-CNT-T			
Ein-/Ausgangspegel	TTL-Logik		
Event-Eingang	TTL-Logik		
Isolation	keine		
Pro II-CNT-D			
Ein-/Ausgangspegel	RS422/485 kompatibel (5V differentiell, 120 Ω Bus-Abschlusswiderstand)		
Event-Eingang	1 differentiell (single ended-Betrieb möglich)		
Taktfrequenz SSI-Decoder (CLK)	2MHz max.		
Isolation	keine		
Pro II-CNT-I			
Eingangsstrom	typ. 3,5mA / max. 7,5mA		
Eingangs-Spannungsbereich (über Jumper wählbar)	0...5V	0 ... 12V	0...24V
sichere Schaltschwelle ¹ für 0 (low)	0...0,8V	0...1,6V	0...3,2V
sichere Schaltschwelle ¹ für 1 (high)	4,5...5V	10...12V	20...24V
Spannungsfestigkeit	8V	16V	30V
negative Spannung	-5V für alle Bereiche		
Schaltzeit	100ns		
Isolation	42V Kanal zu Kanal / Kanal zu Masse		

Abb. 99 – Spezifikationen der Modulvarianten

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Zähler konfigurieren	<code>P2_Cnt_Enable</code> , <code>P2_Cnt_Mode</code>
Zähler ansteuern	<code>P2_Cnt_Clear</code> , <code>P2_Cnt_Get_Status</code> <code>P2_Cnt_Latch</code> <code>P2_Cnt_Read</code> , <code>P2_Cnt_Read4</code> <code>P2_Cnt_Read_Latch</code> <code>P2_Cnt_Read_Latch4</code>
PWM-Zähler ansteuern	<code>P2_Cnt_PW_Enable</code> , <code>P2_Cnt_PW_Latch</code> <code>P2_Cnt_Get_PW</code> , <code>P2_Cnt_Get_PW_HL</code>
SSI-Decoder ansteuern (nur CNT-D)	<code>P2_SSI_Mode</code> , <code>P2_SSI_Set_Bits</code> <code>P2_SSI_Set_Clock</code> <code>P2_SSI_Set_Delay</code> , <code>P2_SSI_Read</code> <code>P2_SSI_Read2</code> <code>P2_SSI_Start</code> , <code>P2_SSI_Status</code>
Event-Eingang ansteuern	<code>P2_Event_Enable</code> , <code>P2_Event_Config</code> <code>P2_Event_Read</code>
LED ansteuern	<code>P2_Check_LED</code> , <code>P2_Set_LED</code>
Synchronisieren	<code>P2_Sync_All</code>

1. Innerhalb der angegebenen Spannungsbereiche wird ein low-/high-Signal sicher erkannt. Der Schaltvorgang kann jedoch bereits außerhalb dieser Spannungsbereiche erfolgen.

Programmierung

Programmierung in TiCoBasic

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Die Include-Datei `Cnt_TiCo.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Zähler konfigurieren	<code>Cnt_Enable</code> , <code>Cnt_Mode</code>
Zähler ansteuern	<code>Cnt_Clear</code> , <code>Cnt_Get_Status</code> <code>Cnt_Latch</code> , <code>Cnt_Sync_Latch</code> <code>Cnt_Read</code> , <code>Cnt_Read_Latch</code> <code>Cnt_Read_Int_Register</code>
PWM-Zähler ansteuern	<code>Cnt_PW_Enable</code> , <code>Cnt_PW_Latch</code> <code>Cnt_Get_PW_HL</code>
SSI-Decoder ansteuern (nur CNT-D)	<code>SSI_Mode</code> , <code>SSI_Set_Bits</code> <code>SSI_Set_Clock</code> <code>SSI_Set_Delay</code> , <code>SSI_Read</code> <code>SSI_Start</code> , <code>SSI_Status</code>
Event-Eingang ansteuern	<code>Event_Enable</code> , <code>Event_Config</code> <code>Trigger_Event</code>
LED ansteuern	<code>Check_LED</code> , <code>Set_LED</code>

Programmierung TiCo- Zugriff

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei `ADwinPro_All.inc` definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch mit dem <i>TiCo</i> - Prozessor über globale Variab- len	<code>P2_TDrv_Init</code> <code>P2_GetData_Long</code> , <code>P2_Get_Par</code> , <code>P2_Get_Par_Block</code> <code>P2_SetData_Long</code> , <code>P2_Set_Par</code> , <code>P2_Set_Par_Block</code> <code>P2_Get_TiCo_RingBuffer</code> , <code>P2_Set_TiCo_RingBuffer</code> <code>P2_RingBuffer_Empty</code> <code>P2_RingBuffer_Full</code>
<i>TiCo</i> -Prozessor steuern	<code>P2_TiCo_Reset</code> , <code>P2_TiCo_Start</code> , <code>P2_TiCo_Stop</code> <code>P2_Get_TiCo_Bootloader_</code> <code>Status</code> <code>P2_Get_TiCo_Status</code> , <code>P2_Workload</code>
<i>TiCo</i> -Prozesse steuern	<code>P2_Process_Status</code> <code>P2_TiCo_Get_Processdelay</code> <code>P2_TiCo_Set_Processdelay</code> <code>P2_TiCo_Start_Process</code> <code>P2_TiCo_Stop_Process</code>
<i>TiCo</i> -Programme übertragen	<code>P2_TiCo_Flash</code> , <code>P2_TiCo_Load</code>

5.7 Pro II: Zusatz- und Schnittstellenmodule

Modul	RTD-8	TC-8-ISO
Revision	E	E
Funktion	Pt-/Ni-Temperaturmessung	Thermoelement-Schnittstelle
Messtechnik	2-Leiter, 3-Leiter oder 4-Leiter	–
Temperaturbereich	-200°C...+700°C	B: 250°C ... 1820°C E: -200°C...1000°C J: -210°C...1250°C K: -200°C ... 1372°C N: -200°C...1300°C R: -50°C ... 1768°C S: -50°C ... 1768°C T: -270°C ... 400°C
Genauigkeit	±0,05...0,1K	±1K
Kanäle	8	8

Seite 96 100

Modul	CAN-2	CANL-2
Revision	E	E
Funktion	CAN-Schnittstelle	
CAN-Version	High speed	Low speed
Schnittstellen	2	

Seite 102

Modul	RSxxx-2	RSxxx-4
Revision	E	E
Funktion	RSxxx-Schnittstelle	
RSxxx-Version	RS232 RS485	RS232 RS485
Schnittstellen	2	4
Übertragungsrate [kBaud]	0,035... 2304	0,035... 2304

Seite 108

Modul	LIN2	Profi-SL	EtherCAT-SL	FlexRay-2
Revision	E	E	E	E
Funktion	Schnittstelle LIN-Bus	Schnittstelle Profibus	Schnittstelle EtherCAT	Schnittstelle FlexRay
Schnittstellen	2	1	1	2

Seite 112 114 117 120

5.7.1 Pro II-RTD-8 Rev. E

Das Modul Pro II-RTD-8 Rev. E hat 8 Eingänge zum Anschluss von Platin-Temperaturfühlern vom Typ Pt 100, Pt 500, Pt 1000 oder Ni 100. Die Eingänge sind über einen Multiplexer an einen ADC angebunden. Der mögliche Messbereich ist $-200^{\circ}\text{C} \dots +700^{\circ}\text{C}$, je nach eingesetztem Temperaturfühler (siehe Datenblatt des Herstellers).

Es gibt 2 Modulvarianten:

- Pro II-RTD-8 Rev. E: Moduleingänge auf der 37-poligen D-Sub-Buchse.
- Pro II-RTD-8-L Rev. E: Moduleingänge auf geschirmten Lemo-Buchsen. Pinbelegung siehe Abb. 101.

Messungen können für jeden Kanal separat in 2-, 3- oder 4-Leitertechnik durchgeführt werden; Eingangsbeschaltung siehe Abb. 100. Die Messmethode und der Fühlertyp werden per Software eingestellt.

Die Messmethoden und die Verkabelung zwischen Sensor und Modul sind ab Seite 97 beschrieben.

Messungen werden entsprechend den Einstellungen automatisch durch eine Ablaufsteuerung durchgeführt, so dass die ADwin CPU die Messwerte nur noch abholen muss.

Die Eingangssignale durchlaufen einen Tiefpassfilter 2. Ordnung mit 25kHz. Zusätzlich können Sie eine einzelne Frequenz aus den Digitalsignalen mit dem Befehl **P2_RTD_Channel_Config** herausfiltern; hierbei wird der Messwert als Mittelwert aus mehreren, in definierten Zeitabständen erfassten Messungen berechnet.

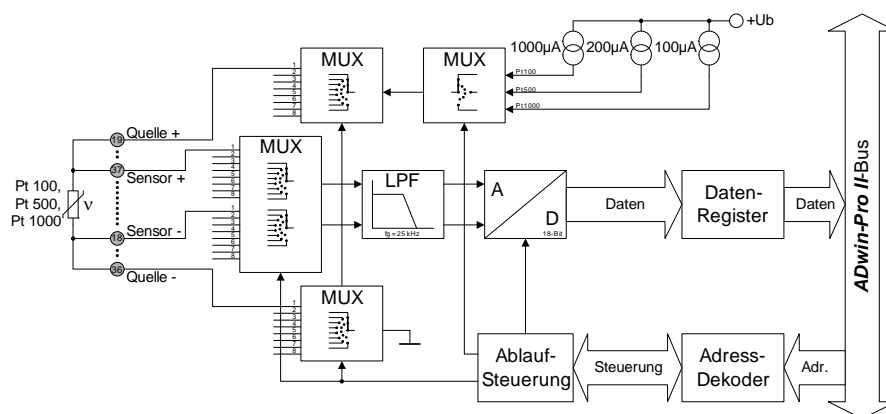


Abb. 100 – Pro II-RTD-8 Rev. E: Blockschaltbild

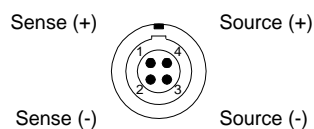


Abb. 101 – Pro II-RTD-8-L Rev. E: Pinbelegung LEMO-Stecker

Eingänge	8
Messtechnik	2-Leiter, 3-Leiter oder 4-Leiter
Multiplexer Einschwingzeit	100µs
Eingangsfiler	25 kHz (2. Ordnung)
max. Messbereich	$-200^{\circ}\text{C} \dots +700^{\circ}\text{C}$

Abb. 102 – Pro II-RTD-8 Rev. E: Spezifikation

Genauigkeit	PT100: $\pm 0,05K$ PT500: $\pm 0,1K$ PT1000: $\pm 0,1K$ Ni100: $\pm 0,05K$
Auflösung	0,015K
Temperaturdrift	$\pm 10\text{ppm/K}$
$I_1 = I_2$	PT100: 1 mA PT500: 0,2 mA PT1000: 0,1 mA Ni100: 1 mA
Steckerverbindung	37-polige D-Sub-Buchse oder geschirmte Lemo-Buchsen

Abb. 102 – Pro II-RTD-8 Rev. E: Spezifikation

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Modul konfigurieren	P2_RTD_Config P2_RTD_Channel_Config
Temperatureingänge ansprechen	P2_RTD_Start P2_RTD_Status P2_RTD_Read P2_RTD_Read8
Messwert umrechnen	P2_RTD_Convert
LEDs einstellen	P2_Check_LED, P2_Set_LED

Messmethoden

Sie haben per Software 3 Messmethoden zur Auswahl: 2-Leiter-Messung, 3-Leiter-Messung oder 4-Leiter-Messung.

– 2-Leiter-Messung

Achten Sie auf eine sehr kurze und niederohmige Verbindung zwischen dem Pt-Sensor und dem Moduleingang, weil der Spannungsabfall über die Messleitungen additiv in die gemessene Spannung eingeht.

Aus diesem Grunde ist diese Messmethode für präzise Messungen generell nicht zu empfehlen.

Für eine 2-Leiter Messung am Kanal n verbinden Sie den Sensor mit den Eingängen *Quelle +* und *Quelle -*. Für Kanal 1 wären dies die Pins 19 und 36 (siehe Abb. 103 bzw.) auf der D-Sub-Buchse.

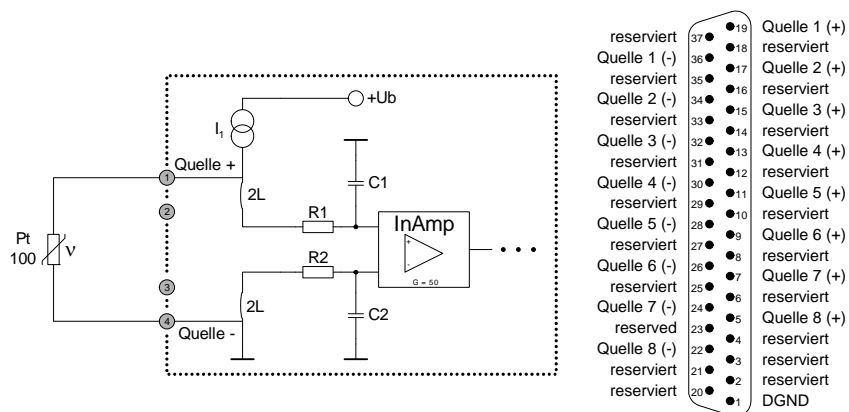


Abb. 103 – Pro II-RTD-8 Rev. E:
Schaltung und Pinbelegung bei 2-Leiter-Messung

– 3-Leiter-Messung

Um die Nachteile der 2-Leiter-Messung zu umgehen, wird hier der Spannungsabfall auf den Messleitungen mittels der zweiten Stromquelle I_2 kompensiert.

Um den Messfehler so gering wie möglich zu halten, sollte der Widerstandswert der 3 Messleitungen vom Pt-Sensor zum Moduleingang identisch sein.

Bei der 3-Leiter-Messung werden immer 2 Messungen durchgeführt, um einen Messwert zu erhalten. Daher braucht eine 3-Leiter-Messung doppelt so viel Zeit wie die 2-Leiter- oder 4-Leiter-Messung.

Für eine 3-Leiter Messung am Kanal n verbinden Sie den Sensor mit den Eingängen *Quelle +*, *Sensor -* und *Quelle -*. Für Kanal 1 wären dies die Pins 18, 19 und 36 (siehe Abb. 104) auf der D-Sub-Buchse.

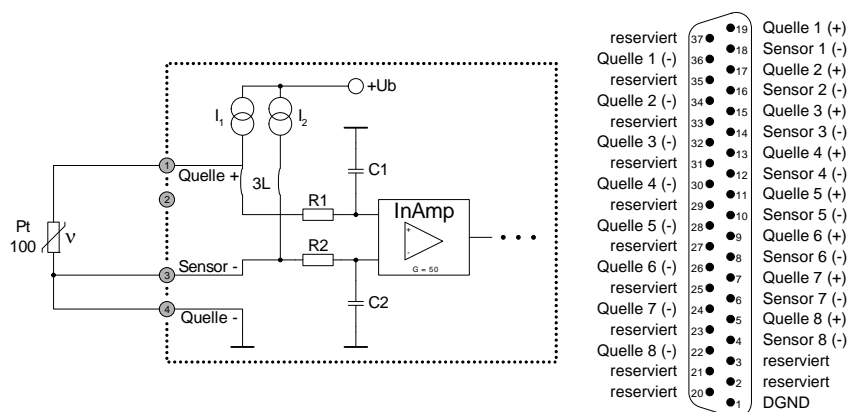


Abb. 104 – Pro II-RTD-8 Rev. E:
Schaltung und Pinbelegung bei 3-Leiter-Messung

– 4-Leiter-Messung

Der Spannungsabfall über den Pt-Sensor wird direkt am Platinelement durch die beiden „Sensor“-Eingänge hochohmig abgegriffen. Die Widerstände der Messleitungen gehen hier nicht mehr in die Messung ein und bedürfen damit auch keiner Kompensation.

Für eine 4-Leiter Messung am Kanal n verbinden Sie den Sensor mit den Eingängen *Quelle +*, *Sensor +*, *Sensor -* und *Quelle -*. Für Kanal 1 wären dies die Pins 18, 19, 36 und 37 (siehe Abb. 105) auf der D-Sub-Buchse.

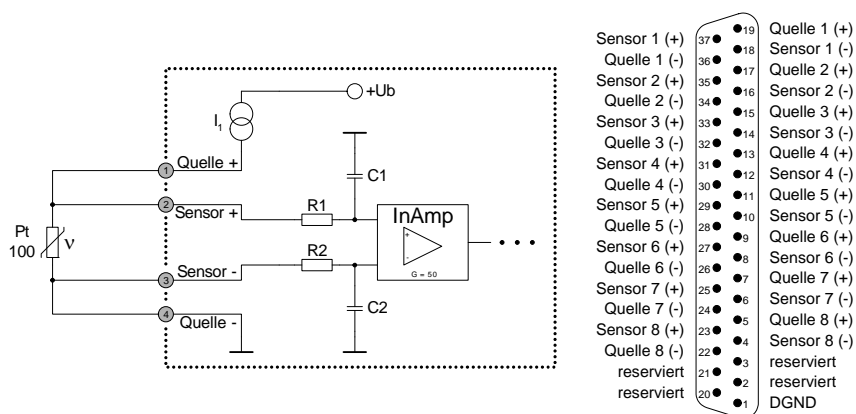


Abb. 105 – Pro II-RTD-8 Rev. E:
Schaltung und Pinbelegung bei 4-Leiter-Messung

5.7.2 Pro II-TC-8 ISO Rev. E

Das Modul Pro II-TC-8 ISO Rev. E hat 8 Eingänge für Thermoelemente und kann mit den Thermoelement-Typen B, E, J, K, N, R, S und T betrieben werden. Über Software-Befehle können Sie die Thermospannung (mit oder ohne Kaltstellenkompensation) oder die Temperatur jedes Kanals separat abfragen.

Jeder Eingang ist mit einem separaten ADC ausgerüstet. Das Modul hat eine gemeinsame Kaltstellenkompensation für alle Temperatureingänge.

Über die Position der Steckbrücken (Jumper, siehe Abb. 108 links) wird für jeden Kanal getrennt die Kanaltrennung eingestellt:

- Position rechts: Die Potentiale eines Eingangs sind voneinander getrennt (Voreinstellung).
- Position links: Der negative Eingang ist mit Erde verbunden.

Die Signalerfassung an den ADC erfolgt mit einer schrittweise einstellbaren Abtastrate. Sobald per Software ein Messwert abgefragt wird, wird aus dem zuletzt erfassten Messsignal die Thermospannung oder den Temperaturwert in Grad Celsius oder in Grad Fahrenheit berechnet. Die Berechnung basiert auf der Grundwertreihe der Norm IEC 584-1. Die Thermospannung ohne Kaltstellenkompensation kann auch zurückgegeben werden.

Die Kalibrierung des Moduls erfolgt beim Hersteller. Senden Sie das Modul hierzu an die Lieferanschrift auf der Rückseite der Titelseite.

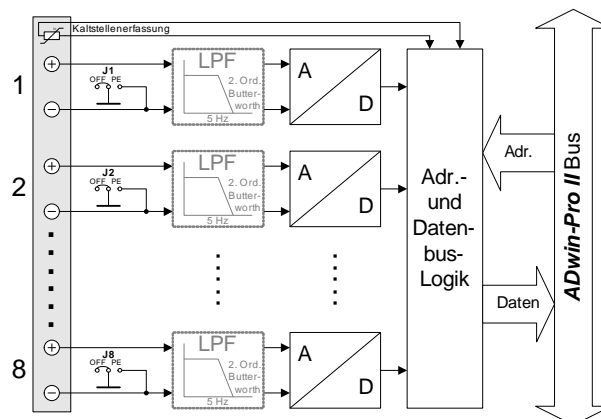


Abb. 106 – Pro II-TC-8 ISO Rev. E: Blockschaltbild

Eingangskanäle	8
Abtastrate	7Hz ... 3500Hz
Thermoelement-Typen, Messbereiche und Genauigkeit	B: 250°C ... 1820°C; $\pm 5^\circ\text{C}$ E: -200°C...1000°C; $\pm 1^\circ\text{C}$ J: -210°C...1250°C; $\pm 1^\circ\text{C}$ K: -200°C ... 1372°C; $\pm 1^\circ\text{C}$ N: -200°C...1300°C; $\pm 2^\circ\text{C}$ R: -50°C ... 1768°C; $\pm 3^\circ\text{C}$ S: -50°C ... 1768°C; $\pm 3^\circ\text{C}$ T: -270°C ... 400°C; $\pm 1^\circ\text{C}$
Auflösung	0,1°C
Eingangswiderstand	10M Ω
Spannungsfestigkeit	$\pm 20\text{V}$
Offsetdrift	$\pm 30\text{ppm}/^\circ\text{C}$ vom Endwert

Abb. 107 – Pro II-TC-8 ISO Rev. E: Spezifikation

Steckerverbindung	Omega Subminiature Connector, Typ SMP
-------------------	--

Abb. 107 – Pro II-TC-8 ISO Rev. E: Spezifikation

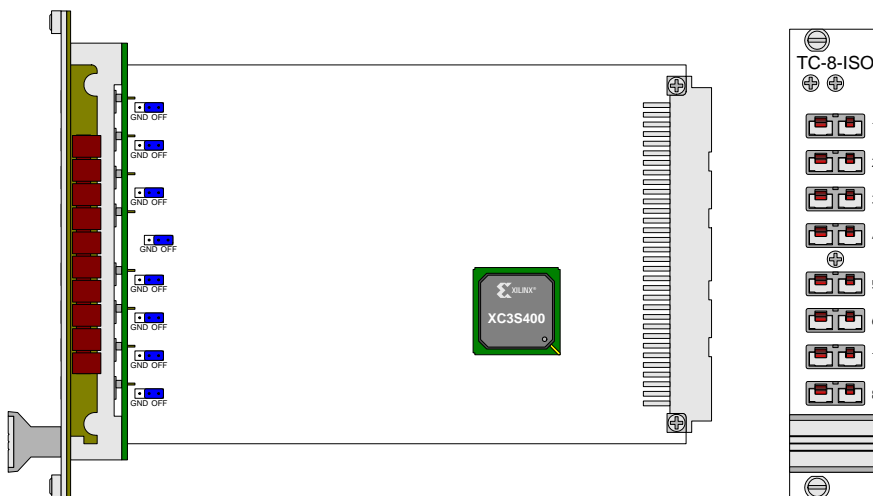


Abb. 108 – Pro II-TC-8 ISO Rev. E: Platine und Frontplatte

Mit folgenden Befehlen wird das Modul programmiert:

Abtastrate ein- stellen	P2_TC_Set_Rate
Eingangswerte in Latches kopieren	P2_TC_Latch, P2_Sync_All
Messwerte lesen	P2_TC_Read_Latch P2_TC_Read_Latch4, P2_TC_Read_Latch8

Programmierung

5.7.3 Pro II-CAN-2 Rev. E

Das Modul Pro II-CAN-2 besitzt 2 CAN-Schnittstellen, die als „high speed“ oder als „low speed“-Variante ausgeführt sind. Die Bezeichnungen der Modulvarianten lauten:

- Pro II-CAN-2 Rev. E: CAN-Schnittstelle high speed
- Pro II-CAN-2-LS Rev. E: CAN-Schnittstelle low speed

Jedes Modul besitzt einen frei programmierbaren *TiCo*-Prozessor, der vollen Zugriff die CAN-Schnittstellen hat. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCo-Basic*.

Die Beschreibung des CAN-Moduls ist in folgende Abschnitte gegliedert:

- CAN-Controller
- Hardware-Aufbau
- Nachrichten verwalten
- Busfrequenz einstellen
- Interrupt freigeben / Event auslösen
- Modul-Revisionen
- Programmierung

CAN-Controller

Die CAN-Schnittstelle ist mit dem CAN-Controller AN82527 von Intel® bestückt und arbeitet nach der Spezifikation „CAN 2.0 part A+B“ sowie ISO 11898. Sie programmieren die Schnittstelle mit *ADbasic*-Befehlen, die direkt auf die Register des Controllers zugreifen.

Über den CAN-Bus verschickte Nachrichten sind Datentelegramme mit bis zu 8 Bytes, die durch sogenannte „Identifizier“ gekennzeichnet sind. Der CAN-Controller unterstützt Identifizier mit 11 Bit und 29 Bit Länge. Die eigentliche Kommunikation, d.h. die Verwaltung der Bus-Nachrichten, erfolgt über 15 „Message-Objekte“.



Zur Konfiguration und Statusanzeige des CAN-Controllers dienen die in ihm enthaltenen Register. Hier werden Busgeschwindigkeit, Interrupt handling usw. eingestellt (siehe separate Dokumentation „82527 - Serial Communications Controller, Architectural Overview“ von Intel®).

Der CAN-Bus (high speed) ist auf Frequenzen bis 1 MHz einstellbar und wird standardmäßig mit 1 MHz betrieben; bei CAN low speed beträgt die max. Frequenz 125kHz. Der CAN-Bus ist durch Optokoppler vom *ADwin*-System galvanisch getrennt.

Der Eingang einer Nachricht kann einen Interrupt auslösen, der sofort einen Event am Prozessor erzeugt. Dadurch kann eine sofortige Bearbeitung der Nachrichten gewährleistet werden.

Hardware-Aufbau

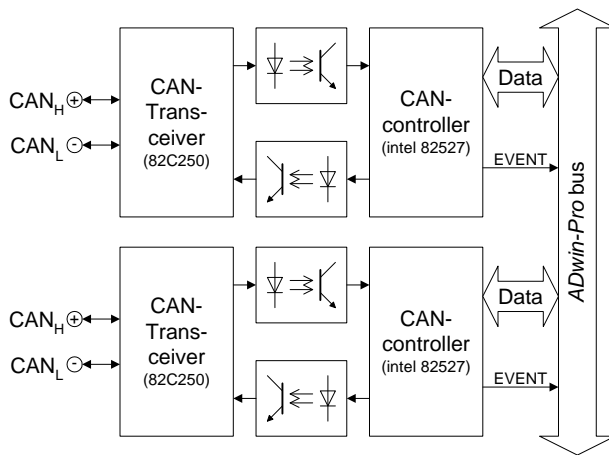


Abb. 109 – Pro II-CAN-2 Rev. E: Blockschaltbild

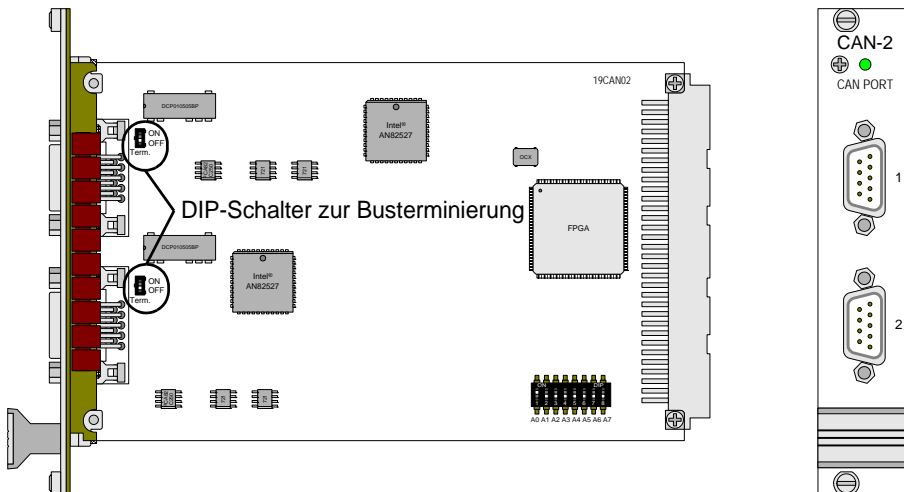


Abb. 110 – Pro II-CAN-2 Rev. E: Platine und Frontplatte

Die Anschlüsse der CAN-Bus Schnittstelle stehen auf einem 9-poligen D-Sub-Verbinder zur Verfügung; die Pin-Belegung ist unten dargestellt.

CAN-2	CAN-2-LS
<div style="display: flex; justify-content: space-around;"> <div> <p>GND 6 ● 1</p> <p>CAN(+) 7 ● 2</p> <p>reserviert 8 ● 3</p> <p>reserviert 9 ● 4</p> <p>reserviert 5 ● 5</p> </div> <div> <p>reserviert 1 ● 1</p> <p>CAN(-) 2 ● 2</p> <p>GND 3 ● 3</p> <p>reserviert 4 ● 4</p> <p>reserviert 5 ● 5</p> </div> </div>	<div style="display: flex; justify-content: space-around;"> <div> <p>GND 6 ● 1</p> <p>CAN(+) 7 ● 2</p> <p>reserviert 8 ● 3</p> <p>+12V (Eingang) 9 ● 4</p> <p>reserviert 5 ● 5</p> </div> <div> <p>reserviert 1 ● 1</p> <p>CAN(-) 2 ● 2</p> <p>GND 3 ● 3</p> <p>reserviert 4 ● 4</p> <p>reserviert 5 ● 5</p> </div> </div>

Abb. 111 – Pro II-CAN-2: Pinbelegungen (Stecker)

Die „low speed“-Variante Pro II-CAN-2-LS benötigt eine externe Spannungsversorgung mit 12V Gleichstrom. Die Spannung muss für jeden Controller separat eingespeist werden.

Wenn das CAN-Modul das physikalische Ende eines CAN-Bus vom Typ „High speed“ bildet, muss es mit einem Abschlusswiderstand 120Ω terminiert werden (also nur am ersten oder letzten CAN-Knoten). An CAN-Knoten, die sich nicht an einem physikalischen Ende der Kette befinden, darf nicht terminiert werden.

Wenn die Terminierung erforderlich ist, legen Sie den entsprechenden DIP-Schalter (siehe Abb. 110) nach oben um.

**Spannungsversorgung
(nur Low speed)**

**Bus-Terminierung
(nur High speed)**

	Nachrichten verwalten
Identifizier	Der CAN-Controller unterscheidet über den Bus verschickte Nachrichten durch „Identifizier“, das sind Kennzahlen mit einer definierten Bitlänge. Aus der Bitlänge ergeben sich hier die möglichen Kennzahlen $0...2^{11}-1$ bzw. $0...2^{29}-1$.
Message-Objekte	<p>Jede Nachricht (zu sendende oder zu empfangende) speichert der Controller in einem von 15 „Message-Objekten“. Die Message-Objekte können jeweils entweder zum Senden oder zum Empfangen konfiguriert werden. Als Ausnahme kann das Message-Objekt 15 nur zum Empfangen genutzt werden. Nach der Initialisierung des CAN-Controllers sind sämtliche Message-Objekte nicht konfiguriert und beteiligen sich nicht am Busverkehr.</p> <p>Jedes Message-Objekt erhält einen Identifizier, der die Zuordnung einer Nachricht zu einem Message-Objekt ermöglicht.</p>
Nachricht übergeben	In <i>ADbasic</i> übergeben Sie eine Nachricht an ein Message-Objekt über das Feld <code>can_msg</code> , das 8 Datenbytes plus die Anzahl der Datenbytes aufnehmen kann (9 Elemente). Ebenso wird eine Nachricht beim Auslesen aus einem Message Objekt in das Feld <code>can_msg</code> übertragen.
Nachricht senden	<p>Das Versenden einer Nachricht läuft in folgenden Schritten ab:</p> <ul style="list-style-type: none">– Sie konfigurieren ein Message-Objekt zum Senden und definieren den Identifizier des Objekts (Befehl EN_TRANSMIT).– Sie speichern die Nachricht im Feld <code>can_msg</code>.– Sie senden die Nachricht (Befehl TRANSMIT). Die Nachricht im Feld <code>can_msg</code> wird an das Message-Objekt übergeben. Sobald der Bus frei ist, wird die Nachricht gesendet (mit dem Identifizier des Message-Objekts).
Nachricht empfangen	<p>Das Empfangen einer Nachricht läuft in folgenden Schritten ab:</p> <ul style="list-style-type: none">– Sie konfigurieren ein Message-Objekt für Empfang und definieren den Identifizier des Objekts (Befehl EN_RECEIVE).– Der Controller überwacht den CAN-Bus auf eingehende Nachrichten und speichert Nachrichten mit dem richtigen Identifizier in dem Message-Objekt.– Sie übertragen die Nachricht aus dem Message-Objekt in das Feld <code>can_msg</code> (Befehl READ_MSG) und lesen den zugehörigen Identifizier aus. <p>Eine eingehende Nachricht überschreibt die alten Daten in dem Message-Objekt, die dadurch unwiderruflich verloren sind. Achten Sie daher beim Programmieren darauf, dass die Daten schneller ausgelesen als empfangen werden. Ein Datenverlust wird durch ein Flag angezeigt.</p> <p>Bei dem Message Objekt 15 existiert ein zusätzlicher interner Zwischenspeicher, so dass dort 2 Nachrichten gespeichert werden können.</p>
Nachricht zuordnen	<p>Die Zuordnung einer eingehenden Nachricht zu einem Message-Objekt wird automatisch durch einen Vergleich ihrer Identifizier gesteuert. Die globale Maske (CAN-Register 6...7 bzw. 6...9) steuert diesen Vergleich:</p> <ul style="list-style-type: none">– Der Identifizier der Nachricht wird bitweise mit dem Identifizier des Message-Objekts verglichen. Wenn die relevanten Bits gleich sind, wird die Nachricht in das Message-Objekt übernommen. Nicht relevante Bits werden nicht verglichen, d.h. die Nachricht wird (sofern es von diesem Bit abhängt) in das Objekt übernommen.– Relevante Bits werden in der globalen Maske festgelegt, indem sie dort gesetzt werden.

Durch die globale Maske kann ein Message-Objekt für den Empfang von Nachrichten mit **verschiedenen Identifiern** (ID) genutzt werden. Das folgende Beispiel zeigt die Zuordnung der Nachrichten-ID 1...4 zu den Message-Objekt-ID 1...4, wenn alle Bits der globalen Maske gesetzt sind bis auf die beiden niederwertigsten (bei einem 11-Bit-Identifizier also **11111111100b**).

Nachrichten-ID	ID des Message-Objekts			
	1 ...001b	2 ...010b	3 ...011b	4 ...100b
1 (...001b)	x	x	x	0
2 (...010b)	x	x	x	0
3 (...011b)	x	x	x	0
4 (...100b)	0	0	0	x

x: Nachricht wird übernommen

0: Nachricht wird nicht übernommen

In diesem Beispiel entscheidet nur der Vergleich des Bits 2 über die Zuordnung, denn die Bits 3...10 der hier verglichenen Identifier sind identisch (= 0) und die Bits 0 und 1 werden nicht verglichen, weil sie in der globalen Maske auf Null gesetzt sind (= nicht relevant).

Busfrequenz einstellen

Die **CAN-Bus-Frequenz** hängt von der Konfiguration des Controllers ab.

Bei der Initialisierung mit **INIT_CAN** wird der Controller automatisch so konfiguriert, dass die CAN-Bus-Frequenz 1MHz beträgt. Soll der CAN-Bus mit einer anderen Frequenz betrieben werden, geschieht dies am einfachsten mit dem Befehl **SET_CAN_BAUDRATE**.

Bei CAN low speed muss die Busfrequenz auf Werte $\leq 125\text{ kBit/s}$ eingestellt werden.

In Sonderfällen kann es vorteilhaft sein, die Einstellungen anders zu wählen, als es mit **SET_CAN_BAUDRATE** möglich ist. Zu diesem Zweck müssen bestimmte Register mit dem Befehl **Poke** gesetzt werden. Der Registeraufbau ist in der Dokumentation des Controllers beschrieben.

Interrupt freigeben / Event auslösen

Sie können bei einem Message-Objekt freigeben, ob es beim Eingang einer Nachricht einen Interrupt auslöst. Der Interrupt-Ausgang des CAN-Controllers ist intern mit dem Event-Eingang des Prozessors verbunden. Dadurch kann der Prozessor sofort auf eingehende Nachrichten reagieren, ohne den Nachrichteneingang kontrollieren zu müssen (Polling).

Sie können die Interrupts mehrerer Message-Objekte freigeben. Welches Objekt den Interrupt ausgelöst hat, kann aus dem Interrupt-Register (**5Fh**) ersehen werden: Es enthält die Nummer des auslösenden Message-Objekts. Wird das Interrupt-Flag (new message flag) im Message-Objekt zurückgesetzt, wird das Interrupt-Register aktualisiert. Wenn kein Interrupt mehr ansteht, wird das Register auf „0“ gesetzt. Ist während der Bearbeitung des ersten Interrupts ein weiterer aufgetreten, so wird dessen Quelle nun im Interrupt-Register angezeigt. Ein weiterer Hardware-Interrupt erfolgt in diesem Fall nicht.

Modul-Revisionen

Die Unterschiede der Revisionsstände sind nachfolgend dargestellt:

Globale Maske

Busfrequenz für Sonderfälle

Programmierung in ADbasic

Revision	Ausgabe- datum	Änderungen zur Vorgänger-Version
E1		Erst-Version

Die Revisionsbezeichnung befindet sich auf der Frontseite.

Unterschiedliche Revisions-Buchstaben bedeuten unterschiedliche Modul-Eigenschaften und sind separat dokumentiert. Der Revisionsbezeichnung angehängt ist eine Zählnummer für interne Zwecke.

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
CAN-Controller initialisieren, Baudrate einstellen	<code>P2_Init_CAN</code> <code>P2_Set_CAN_Baudrate</code>
Message-Objekte übertragen	<code>P2_En_Receive</code> , <code>P2_Read_Msg</code> <code>P2_En_Transmit</code> , <code>P2_Transmit</code>
Register setzen und lesen	<code>P2_Set_CAN_Reg</code> , <code>P2_Get_CAN_Reg</code>
LEDs einstellen	<code>P2_Check_LED</code> , <code>P2_Set_LED</code> , <code>P2_CAN_Set_LED</code>
Interrupts und Event-Eingang einstellen	<code>P2_En_Interrupt</code> , <code>P2_Event_Enable</code> , <code>P2_Event_Config</code> , <code>P2_Event_Read</code>

Programmierung in TiCoBasic

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Die Include-Datei `CAN_TiCo.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
CAN-Controller initialisieren, Baudrate einstellen	<code>Init_CAN</code> <code>Set_CAN_Baudrate</code>
Message-Objekte übertragen	<code>En_Receive</code> , <code>Read_Msg</code> <code>Read_Msg_Con</code> , <code>CAN_Msg</code> <code>En_Transmit</code> , <code>Transmit</code> <code>Transmit_Status</code>
Register setzen und lesen	<code>Set_CAN_Reg</code> , <code>Get_CAN_Reg</code>
LEDs einstellen	<code>Check_LED</code> , <code>Set_LED</code> <code>CAN_Set_LED</code>
Interrupts einstellen	<code>En_Interrupt</code>

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei `ADwinPro_All.inc` definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch mit dem <i>TiCo</i> -Prozessor über globale Variablen	P2_TDrv_Init P2_GetData_Long, P2_Get_Par, P2_Get_Par_Block P2_SetData_Long, P2_Set_Par, P2_Set_Par_Block P2_Get_TiCo_RingBuffer, P2_Set_TiCo_RingBuffer P2_RingBuffer_Empty P2_RingBuffer_Full
<i>TiCo</i> -Prozessor steuern	P2_TiCo_Reset, P2_TiCo_Start, P2_TiCo_Stop P2_Get_TiCo_Bootloader_ Status P2_Get_TiCo_Status, P2_Workload
<i>TiCo</i> -Prozesse steuern	P2_Process_Status P2_TiCo_Get_Processdelay P2_TiCo_Set_Processdelay P2_TiCo_Start_Process P2_TiCo_Stop_Process
<i>TiCo</i> -Programme übertragen	P2_TiCo_Flash, P2_TiCo_Load

Programmierung *TiCo*-Zugriff

5.7.4 Pro II-RSxxx Rev. E

Das Modul Pro II-RSxxx besitzt 2 oder 4 Schnittstellen vom Typ RS-232 oder RS-485. Der Schnittstellentyp wird mit dem Befehl **P2_rs485_send** eingestellt.

Die Bezeichnungen der Modulvarianten lauten:

- Pro II-RSx-2 Rev. E: 2 Schnittstellen RS232/485
- Pro II-RSx-4 Rev. E: 4 Schnittstellen RS232/485

Alle Modulvarianten der RSxxx-Module sind mit dem Controller „Quad Universal Asynchronous Receiver/Transmitter“ (UART) vom Typ TL16C754 der Firma Texas Instruments® bestückt. Die Funktionalität und Programmierung der Schnittstellen beruhen auf diesem Controller.

Jedes Modul besitzt einen frei programmierbaren *TiCo*-Prozessor, der vollen Zugriff die RSxxx-Schnittstellen hat. Nähere Hinweise zur Anwendung und Programmierung des *TiCo*-Prozessors entnehmen Sie bitte dem Handbuch *TiCoBasic*.

Der physikalische Unterschied zwischen den Schnittstellen-Typen liegt in den Pegeln der Signale, die auf dem „Bus“ durch entsprechende Treiber-Bausteine bereitgestellt werden.

Die Beschreibung ist in folgende Abschnitte gegliedert:

- Hardware
- Einstellbare Schnittstellen-Parameter
- Modul-Revisionen
- Programmierung

Hardware

Nachfolgend sind Frontplatten und Pin-Belegungen der Module Pro II-RSxxx gezeigt. Die Pinbelegung wird mit dem Schnittstellentyp umgeschaltet.

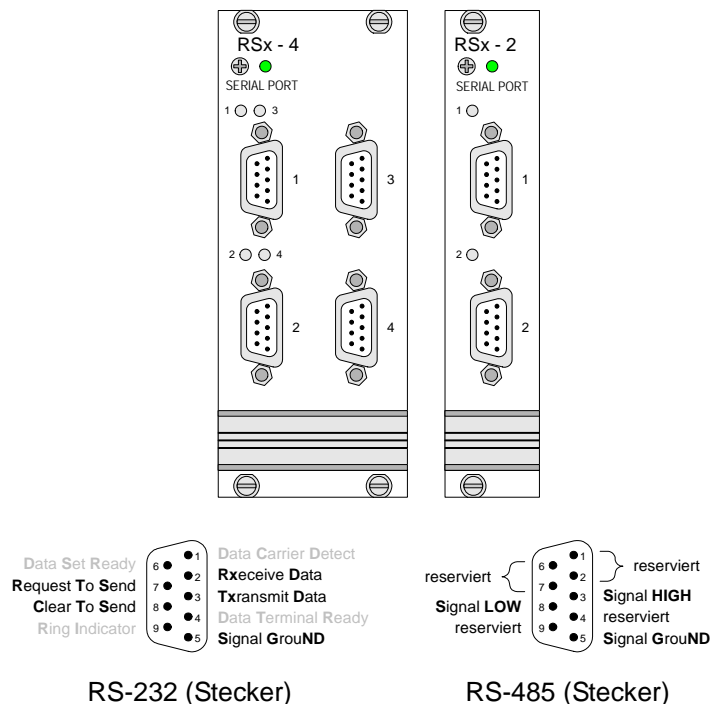


Abb. 112 – Pro II-RS-xxx: Frontplatten / Pinbelegungen

Einstellbare Schnittstellen-Parameter

Jede Schnittstelle verfügt über einen Eingangs- und einen Ausgangs-FIFO mit einer Länge von jeweils 64 Byte. Die Einstellung der Schnittstellen-Parameter wird mit Hilfe der Controller-Register vorgenommen, und zwar getrennt für jeden Kanal. Im Folgenden werden die Einstellmöglichkeiten dargestellt:

- Handshake: Die Schnittstelle kann in 3 Modi betrieben werden:
 1. Ohne Handshake
 2. Software-Handshake
 3. Hardware-Handshake (nur RS232).
Beim Hardware-Handshake müssen die Signale RTS und CTS angeschlossen sein.
- Parität: Um einen Fehler bei der Übertragung und damit fehlerhafte Daten erkennen zu können, kann ein Paritätsbit mit übertragen werden. Die Parität kann gerade oder ungerade sein, oder es kann auf das Paritätsbit verzichtet werden.
- Datenbits: Die Nutzdaten, die übertragen werden sollen, können aus 5...8 Bits bestehen.
- Stopp-Bits: Die Anzahl der Stopp-Bits kann auf 1, 1½ oder 2 eingestellt werden. Dabei ist die Anzahl der Stoppbits von der Anzahl der Datenbits abhängig:
 - 5 Datenbits: 1 oder 1½ Stoppbits.
 - 6...8 Datenbits: 1 oder 2 Stoppbits.
- Baudrate: Die physikalisch erreichbaren Werte liegen zwischen 35 Baud und 2,304MBAud; bei einer RS-232 Schnittstelle liegt die max. Baudrate laut Spezifikation bei 115,2kBAud.

Die einstellbaren Baudraten werden vom moduleigenen Taktgeber abgeleitet; der Grundtakt hat eine Frequenz von 2,304MHz. Davon ausgehend ist jede Baudrate möglich, die sich durch ganzzahlige Division dieses Grundtakts ergibt. Der Teiler kann Werte im Bereich von 1...0FFFFh annehmen. Die nachfolgende Tabelle zeigt einige gängige Baudraten und die zugehörigen Teiler.

Baudrate	Teiler		Baudrate	Teiler	
	dez.	hex.		dez.	hex.
2304000	1	0001h	19200	120	0078h
1152000	2	0002h	9600	240	00F0h
460800	5	0005h	4800	480	01E0h
230400	10	000Ah	2400	960	03C0h
115200	20	0014h	1200	1920	0780h
57600	40	0028h	600	3840	0F00h
38400	60	003Ch	300	7680	1E00h

Abb. 113 – Pro II-RS-xxx: Gängige Baudraten

Über eine RS485-Schnittstelle können – im Gegensatz zu RS232 – mehr als 2 Teilnehmer miteinander kommunizieren. So kann mit Hilfe von RS485-Schnittstellen ein Bus aufgebaut werden.

Daraus ergeben sich folgende Hinweise:

- Bei RS485 gibt es keinen Handshake, da ein Handshake immer nur zwischen zwei Teilnehmern funktioniert.
- Jeder Schnittstelle muss mit **RS485_SEND** mitgeteilt werden, ob sie auf den Bus schreiben soll oder nur Daten vom Bus übernehmen darf.

Handshake

Parität

Daten-Bits

Stopp-Bits

Baudrate

Besonderheiten RS485

Programmierung in ADbasic

Modul-Revisionen

Die Unterschiede der Revisionsstände sind nachfolgend dargestellt:

Revision	Ausgabe- datum	Änderungen zur Vorgänger-Version
E	11/2007	Erst-Version

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei *ADwinPro_All.inc* enthält Befehle für folgende Bereiche:

Bereich	Befehle
Initialisierung	P2_RS_INIT, P2_rs_reset
Empfangen und Senden von Daten	P2_read_fifo, P2_write_fifo
RS485-Kanal konfigurieren	P2_rs485_send
Schreib- / Lesezugriff auf Controller-Register	P2_GET_RS, P2_SET_RS
LED ansteuern	P2_Check_LED, P2_SET_LED P2_CAN_SET_LED

Programmierung in TiCoBasic

Das Modul kann mit *TiCoBasic*-Befehlen programmiert werden. Die Befehle sind in der Online-Hilfe *TiCoBasic* erläutert.

Die Include-Datei *RS_LIN_TiCo.inc* enthält Befehle für folgende Bereiche:

Bereich	Befehle
Initialisierung	RS_INIT, rs_reset
Empfangen und Senden von Daten	read_fifo, write_fifo Check_Shift_Reg
RS485-Kanal konfigurieren	rs485_send
Schreib- / Lesezugriff auf Controller-Register	GET_RS, SET_RS
LED ansteuern	Check_LED, SET_LED RS_Set_LED

Programmierung TiCo- Zugriff

Für den Zugriff auf den *TiCo*-Prozessor von der ADwin CPU sind die folgenden *ADbasic*-Befehle in der Datei *ADwinPro_All.inc* definiert. Die Befehle sind im Handbuch *TiCoBasic* und in der Online-Hilfe *ADbasic* erläutert.

Bereich	Befehle
Datenaustausch mit dem <i>TiCo</i> - Prozessor über globale Variab- len	P2_TDrv_Init P2_GetData_Long, P2_Get_Par, P2_Get_Par_Block P2_SetData_Long, P2_Set_Par, P2_Set_Par_Block P2_Get_TiCo_RingBuffer, P2_Set_TiCo_RingBuffer P2_RingBuffer_Empty P2_RingBuffer_Full
<i>TiCo</i> -Prozessor steuern	P2_TiCo_Reset, P2_TiCo_Start, P2_TiCo_Stop P2_Get_TiCo_Bootloader_ Status P2_Get_TiCo_Status, P2_Workload

Bereich	Befehle
TiCo-Prozesse steuern	P2_Process_Status
	P2_TiCo_Get_Processdelay
	P2_TiCo_Set_Processdelay
	P2_TiCo_Start_Process
	P2_TiCo_Stop_Process
TiCo-Programme übertragen	P2_TiCo_Flash, P2_TiCo_Load

5.7.5 Pro II-LIN-2 Rev. E

Das Modul Pro II-LIN-2 Rev. E besitzt 2 LIN-Schnittstellen, die unabhängig voneinander als LIN-Master oder LIN-Slave konfiguriert werden können.

LIN-Schnittstelle

Die LIN-Schnittstellen des Moduls sind nach der Spezifikation "LIN 2.1" (Local Interconnect Network) vom November 2006 implementiert. Sie programmieren die LIN-Schnittstelle mit *ADbasic*-Befehlen.

LIN ist ein serielles Kommunikationsprotokoll auf einem Eindrahtbus mit einer Übertragungsgeschwindigkeit bis zu 20 KiBit/s. Der Bus ist besonders kostengünstig und wird für die Vernetzung mechatronischer Knoten in Fahrzeugen eingesetzt.

Über den LIN-Bus verschickte Nachrichten sind Datenpakete mit bis zu 8 Bytes Nutzdaten, die durch sogenannte "Identifizier" gekennzeichnet sind. Die Verwaltung der Bus-Nachrichten auf dem LIN-Modul erfolgt über 64 "Message-Boxen".

Das Buskonzept sieht einen einzelnen Master-Knoten und mehrere Slave-Knoten vor. Der Master kontrolliert den gesamten Datenverkehr auf dem Bus: Bei jedem Datenpaket sendet der Master zunächst einen Header mit dem Identifizier des nächsten Datenpakets. Daraufhin reagieren diejenigen Knoten (das kann auch der Master selbst sein), die eine Messagebox mit diesem Identifizier verwalten, indem sie ein Datenpaket senden oder ein auf dem Bus anstehendes Datenpaket empfangen.

Hardware-Aufbau

Die Anschlüsse der LIN-Bus Schnittstelle stehen auf zwei 9-poligen D-Sub-Verbindern zur Verfügung; die Pin-Belegung ist unten dargestellt.

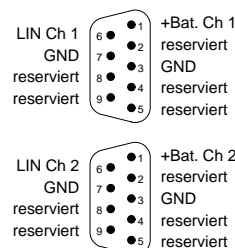


Abb. 114 – Pro II-LIN-2: Pinbelegungen

Mit LIN arbeiten

Die 2 LIN-Schnittstellen des Moduls können mit **P2_LIN_Init_Write** unabhängig voneinander als LIN-Master oder LIN-Slave konfiguriert werden; dies gilt auch für die Einstellung der Baudrate. Der Busabschluss wird je nach Konfiguration automatisch geschaltet.

Das Zeitverhalten des Masters (Netzwerkmanagement NMT) wird in *ADbasic* programmiert.

Beim Konfigurieren einer Messagebox mit **P2_LIN_Msg_Write** geben Sie den Identifizier und den Sendestatus (Senden, Empfangen) der Messagebox an. Einer LIN-Schnittstelle können beliebig viele Messageboxen zugeordnet werden, jedoch darf auf dem LIN-Bus jeder Identifizier nur einmal zum Senden und einmal zum Empfangen vergeben werden; anderenfalls entstehen Datenkollisionen.

Nach dem Konfigurieren einer Messagebox ist diese sofort auf dem LIN-Bus aktiv, d. h. es können Daten empfangen oder gesendet werden.

Wenn die LIN-Schnittstelle als LIN-Master konfiguriert ist, senden Sie mit **P2_LIN_Msg_Transmit** einen Header mit einem Identifier auf den LIN-Bus. Die mit diesem Identifier konfigurierten Messageboxen am LIN-Bus reagieren darauf automatisch.

Die Messagebox eines Master-Knotens verhält sich anders als die eines Slave-Knotens:

- **Master-Knoten, Senden:** Der Master sendet sowohl den Header (siehe **P2_LIN_Msg_Transmit**) als auch gleich anschließend das Datenpaket der Messagebox.
- **Master-Knoten, Empfangen:** Der Master sendet den Header (siehe **P2_LIN_Msg_Transmit**) auf den LIN-Bus und wartet auf die Antwort des passenden Slaves. Das empfangene Datenpaket wird in die Messagebox eingetragen.
- **Slave-Knoten, Senden:** Der Slave wartet, bis der Master den Header mit dem zur Messagebox passenden Identifier sendet. Erst dann sendet der Slave-Knoten das Datenpaket.
- **Slave-Knoten, Empfangen:** Der Slave wartet, bis der Master den Header mit dem zur Messagebox passenden Identifier sendet, empfängt anschließend das Datenpaket und trägt es in die Messagebox ein.

Modul-Revisionen

Die Unterschiede der Revisionsstände sind nachfolgend dargestellt:

Revision	Ausgabe- datum	Änderungen zur Vorgänger-Version
E1		Erst-Version

Die Revisionsbezeichnung befindet sich auf der Frontseite.

Unterschiedliche Revisions-Buchstaben bedeuten unterschiedliche Moduleigenschaften und sind separat dokumentiert. Der Revisionsbezeichnung angehängt ist eine Zählnummer für interne Zwecke.

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
LIN-Schnittstellen initialisieren und zurücksetzen	P2_LIN_Init P2_LIN_Init_Write P2_LIN_Init_Apply P2_LIN_Reset
Schnittstellenversion abfragen	P2_LIN_Get_Version
Daten lesen und senden	P2_LIN_Read_Dat
LIN-Header senden	P2_LIN_Msg_Write P2_LIN_Msg_Transmit
Modul-LED einstellen	P2_Check_LED P2_Set_LED P2_LIN_Set_LED

5.7.6 Pro II-Profi-SL Rev. E

Das Modul Pro II-Profi-SL Rev. E stellt einen Feldbusknoten mit der Funktionalität eines Profibus-Slave zur Verfügung. Alle Einstellungen werden per Software vorgenommen.

Funktionsbeschreibung

Nach dem Einschalten muss der Feldbusknoten initialisiert werden. Mit der Initialisierung werden die Stationsadresse (Slave-Knotenadresse) auf dem Profibus und die Größe der Ein- und Ausgangsbereiche festgelegt.

Es gibt je einen Bereich für eingehende und für ausgehende Daten; jeder Bereich hat eine maximale Größe von 76 Byte. Die Begriffe „Eingang“ und „Ausgang“ sind aus Sicht des Feldbus-Controllers zu sehen.

Sie legen für beide Bereiche separat fest, wieviele Datenbereiche in jedem Bereich vorhanden sind und wie groß die Datenbereiche sind.

Hardware

Die Pinbelegung der 9-poligen D-Sub-Buchse DP-V1 entspricht der DIN E 19245, Teil 3.

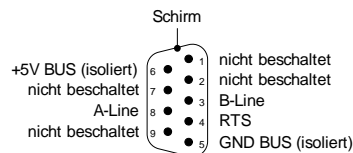


Abb. 115 – Pro II-Profi-SL Rev. E: Pinbelegung

Der Profibus muss am physikalischen Anfang und Ende der Busleitung mit einem Abschlusswiderstand abgeschlossen werden. Falls erforderlich, müssen Sie den Abschlusswiderstand selbst an den entsprechenden Datenleitungen des Feldbusknotens anbringen oder einen entsprechenden Steckverbinder verwenden.

Über und unter der D-Sub-Buchse befinden sich zwei LEDs, die den Betriebszustand des Knotens im Profibus anzeigen, nämlich Betriebsmodus (OP) und Status (ST).

LED	Status	Bedeutung
OP	aus	Nicht online oder keine Stromversorgung.
	grün	Feldbusknoten online, Datenaustausch.
	blinkt grün	Feldbusknoten online, Status Clear.
	blinkt rot, einfach	Fehler: Ein/Ausgangskonfiguration stimmt nicht mit der Masterkonfiguration überein.
	blinkt rot, doppelt	Fehler bei der Profibus-Konfiguration.
ST	aus	Nicht online oder keine Stromversorgung.
	grün	Initialisiert.
	blinkt grün	Initialisiert, Diagnosemeldung liegt an.
	rot	Ausnahmefehler.

Abb. 116 – Pro II-Profi-SL Rev. E: Bedeutung der LED

Profibus projektieren

Sie projektieren den Profibus mit einem – zum Bus-Master passenden – Konfigurations-Tool. Für das folgende Beispiel wurden ein Profibus-Master der Firma Hilscher und das zugehörige Programm SyCon verwendet.

Für andere Konfigurations-Tools gilt die folgende Ablaufbeschreibung entsprechend. Entnehmen Sie die genaue Vorgehensweise bei der Busprojektierung der Dokumentation Ihres Konfigurations-Tools.

- Kopieren oder importieren Sie die GSD-Datei `hmsb1811.gsd` (Gerätstammdatens-Datei) des Feldbusknotens von `C:\ADwin\Fieldbus\Profibus` in das Quellverzeichnis des Konfigurations-Tools.

Das Konfigurations-Tool lädt die benötigten Informationen über einen neuen Slave aus der zugehörigen GSD-Datei; der Dateiinhalt ist durch die EN 50170 festgelegt. Anschließend kann der Slave von jedem Master angesprochen werden.

- Fügen Sie im Konfigurations-Tool den Slave, also den Feldbusknoten zum Profibus hinzu, in dem Sie die GSD-Datei `hmsb1811.gsd` auswählen. Die Stationsadresse muss die gleiche sein wie bei der Initialisierung in *ADbasic* mit **P2_Init_Profibus**.

Danach könnte das Profibus-Layout wie folgt aussehen:



- Legen Sie die Anzahl und Größe der Datenbereiche für eingehende und für ausgehende Daten an. Verwenden Sie für Eingang und Ausgang im Speicher des Feldbusknotens jeweils nur eine Datenbereichsgröße.

Datenbereiche können in Größen von 1, 2, 4 oder 8 Byte angelegt werden (2 byte = 1 word).

Anzahl und Größe der Datenbereiche müssen die gleichen sein wie bei der Initialisierung in *ADbasic* mit **P2_Init_Profibus**.

Danach könnte die Konfiguration des Slave wie folgt aussehen:

Module	Inputs	Outputs	In/Out	Identifier
Input 1 byte	1 Byte			0x90
Input 1 word	1 Word			0xD0
Input 2 words	2 Word			0xD1
Input 4 words	4 Word			0xD3
Output 1 byte		1 Byte		0xA0
Output 1 words		1 Word		0xE0

Slot	Idx	Module	Symbol	Type	I Addr.	I Len.	Type	O Addr.	O Len.
1	1	Output 1	Module1				QB	0	1
2	1	Output 1	Module2				QB	1	1
3	1	Input 1	Module3	IB	0	1			
4	1	Input 1	Module4	IB	1	1			
5	1	Input 1	Module5	IB	2	1			

GSD-Datei kopieren

Slave einbinden

Slave konfigurieren

Modul-Revisionen

Die Unterschiede der Revisionsstände sind nachfolgend dargestellt:

Revision	Ausgabe- datum	Änderungen zur Vorgänger-Version
E1	Juli 2008	Erst-Version

Die Revisionsbezeichnung befindet sich auf der Frontseite.

Unterschiedliche Revisions-Buchstaben bedeuten unterschiedliche Modul-Eigenschaften und sind separat dokumentiert. Der Revisionsbezeichnung angehängt ist eine Zählnummer für interne Zwecke.

Programmieren in *ADbasic*

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Stationsadresse und Datenbereiche initialisieren	P2_Init_Profibus
Daten schreiben und lesen	P2_Run_Profibus

Die Initialisierung muss mit niedriger Priorität ablaufen, da sie einige Sekunden in Anspruch nimmt; bei hoher Priorität würde der PC nach einer bestimmten Zeit (time-out) die Kommunikation abbrechen. Aus dem gleichen Grund sollte auch das Schreiben und Lesen von Daten mit niedriger Priorität ablaufen.

Spezifikationen

Der Feldbusknoten entspricht dem europäischen Standard EN 50170 Volume 2. Dieser kann von der Profibus-Nutzerorganisation bezogen werden:

Profibus Nutzerorganisation e.V.
Haid-und-Neu-Str.7
76131 Karlsruhe
Tel.: +497219658590
Fax : +497219658589
Bestellnummer: 0.042

Die nachfolgende Tabelle zeigt die Betriebszustände, die der Feldbusknoten unterstützt und welches Verhalten er in den verschiedenen Zuständen zeigt.

Betriebs- Zustand	Verhalten
Operate	Der Profibusslave nimmt am zyklischen Datenverkehr teil. Eingangsdaten werden von einem Master über den Bus übernommen und Ausgangsdaten werden für den Master zum Abholen bereitgestellt.
Clear	Die Eingänge werden weiterhin aktualisiert und die Ausgänge werden auf Null gesetzt.
Stop	Der Slave nimmt nicht an der Buskommunikation teil.

Abb. 117 – Pro II-Profi-SL Rev. E: Betriebszustände

Betriebszustände des Feldbusknotens

5.7.7 Pro II-EtherCAT-SL Rev. E

Das Modul Pro II-EtherCAT-SL Rev. E stellt einen Feldbusknoten mit der Funktionalität eines EtherCAT-Slave zur Verfügung. Alle Einstellungen werden per Software vorgenommen.

Funktionsbeschreibung

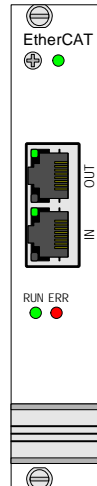
Nach dem Einschalten müssen Sie den Feldbusknoten in *ADbasic* initialisieren. Mit der Initialisierung wird die Größe der Ein- und Ausgangsbereiche festgelegt.

Es gibt je einen Bereich für eingehende und für ausgehende Daten; jeder Bereich hat eine Größe von 16 Long oder 64 Byte. Die Begriffe „Eingang“ und „Ausgang“ sind aus Sicht des Feldbus-Controllers zu sehen.

Hardware

Die Schnittstelle hat je eine Buchse vom Typ RJ45 für den Dateneingang (IN) und den Datenausgang (OUT). An jeder Buchse ist oben rechts eine LED „Link / Activity“, die den Betriebszustand des Knotens im EtherCAT-Bus anzeigt. Die beiden weiteren LEDs (jeweils unten an der Buchse) sind ohne Funktion.

Unterhalb der neben den Buchsen befinden sich LEDs, die den den Status der EtherCAT-Zustandsmaschine (RUN) und das Auftreten von Kommunikationsfehlern (ERR) anzeigen.



LED	Status	Bedeutung
Link / Activity	Aus	Nicht online (oder keine Stromversorgung).
	An	Feldbusknoten online, kein Datenaustausch.
	flackernd	Feldbusknoten online, mit Datenaustausch.
RUN	Aus	Status INIT: Die Schnittstelle wird initialisiert (oder keine Stromversorgung).
	blinkt grün	Status PRE-OP: Schnittstelle hat Kontakt zum Bus-Master.
	leuchtet einmal grün	Status SAFE-OP: Schnittstelle kann Daten vom Bus lesen, aber nicht senden.
	leuchtet grün	Status OP: Schnittstelle ist vollständig eingerichtet, Ein- und Ausgänge sind aktiv.
	leuchtet rot	Status EXCEPTION: Ausnahmesituation.
ERR	Aus	Kommunikation arbeitet ohne Fehler (oder keine Stromversorgung).
	blinkt rot	Fehler bei der Konfiguration.
	leuchtet einmal rot	Lokaler Fehler in der Schnittstelle; der EtherCAT-Status wurde geändert.
	leuchtet doppelt rot	Fehler durch Zeitüberschreitung (timeout).
	leuchtet rot	Kritischer Kommunikationsfehler.

Abb. 118 – Pro II-EtherCAT-SL Rev. E: Bedeutung der LED

Wenn beide LEDs RUN und ERR rot leuchten, ist ein gravierender Fehler in der Schnittstelle aufgetreten. Melden Sie sich dann bitte beim Support von Jäger Messtechnik; die Adresse finden Sie auf der vorderen Umschlagseite des Handbuchs, innen.

EtherCAT projektieren

Sie projektieren den EtherCAT mit einem – zum Bus-Master passenden – Konfigurations-Tool. Für das folgende Beispiel wurde das Programm „TwinCAT System Manager“ der Firma Beckhoff als EtherCAT-Master verwendet.

Für andere Konfigurations-Tools gilt die folgende Ablaufbeschreibung entsprechend. Entnehmen Sie die genaue Vorgehensweise bei der Busprojektierung der Dokumentation Ihres Konfigurations-Tools.

- Kopieren Sie die Beschreibungsdatei `ADwin-EtherCAT.xml` des Feldbusknotens von `C:\ADwin\Feldbus\EtherCAT` in das Quellverzeichnis des Konfigurations-Tools.

Beim Starten lädt das Konfigurations-Tool die benötigten Informationen über den neuen Slave aus der zugehörigen Beschreibungsdatei.

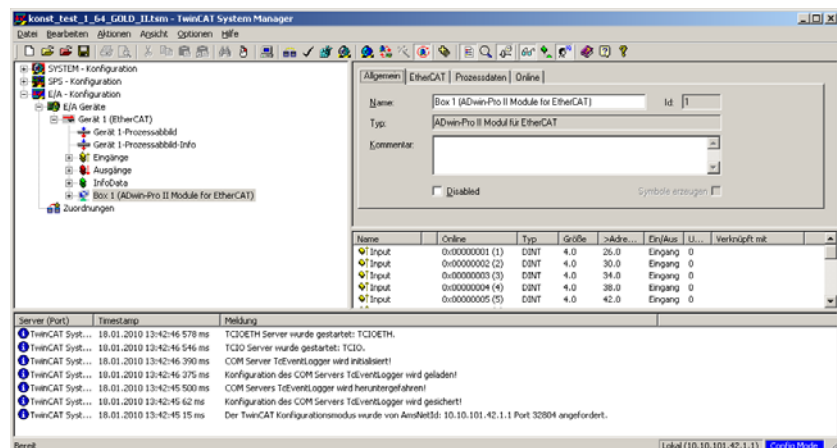
- Fügen Sie die *ADwin*-EtherCAT-Slave als Busteilnehmer zum EtherCAT-Bus hinzu.

Im TwinCAT System Manager markieren Sie dazu den EtherCAT-Master und wählen im Kontextmenü (rechte Maustaste) den Menüpunkt **Boxen scannen**.

Ihnen wird eine Liste aller Busteilnehmer angezeigt.

- Wählen Sie aus der Liste den *ADwin*-EtherCAT-Slave; damit ist der Slave als Busteilnehmer bestätigt.

Danach könnte das EtherCAT-Layout wie folgt aussehen:



- Konfigurieren Sie den *ADwin*-EtherCAT-Slave in einem *ADbasic*-Programm mit dem Befehl **P2_ECATCH_Init**.
- Lesen Sie die Konfiguration im Konfigurations-Tool aus.

Im TwinCAT System Manager markieren Sie dazu den *ADwin*-EtherCAT-Slave und klicken auf die Schaltfläche **Lade PDO Info** aus dem Gerät.

Die Konfiguration des Slave ist in jedem Fall wie folgt: 16 **DINT** (4 Byte) als Eingänge und 16 **DINT** (4 Byte) als Ausgänge.

Modul-Revisionen

Die Unterschiede der Revisionsstände sind nachfolgend dargestellt:

Revision	Ausgabe- datum	Änderungen zur Vorgänger-Version
E1	Dezember 2009	Erst-Version

Die Revisionsbezeichnung befindet sich auf der Frontseite.

Unterschiedliche Revisions-Buchstaben bedeuten unterschiedliche Moduleigenschaften und sind separat dokumentiert. Der Revisionsbezeichnung angehängt ist eine Zählnummer für interne Zwecke.

Programmieren in ADbasic

Das Modul wird komfortabel mit ADbasic-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe ADbasic erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
Datenbereiche initialisieren	<code>P2_ECAT_Init</code>
Version der Schnittstelle lesen	<code>P2_ECAT_Get_Version</code>
Betriebsstatus der Schnittstelle lesen	<code>P2_ECAT_Get_State</code>
Daten schreiben und lesen, Format 16 Bit Long	<code>P2_ECAT_Read_Data_16L</code> <code>P2_ECAT_Write_Data_16L</code>

Die Initialisierung muss mit niedriger Priorität ablaufen, da sie einige Sekunden in Anspruch nimmt; bei hoher Priorität würde der PC nach einer bestimmten Zeit (time-out) die Kommunikation abbrechen. Aus dem gleichen Grund sollte auch das Schreiben und Lesen von Daten mit niedriger Priorität ablaufen.

Spezifikationen

Der Feldbusknoten entspricht den internationalen Standards IEC 61158 (Protokolle und Dienste) und IEC 61784-2 (Kommunikationsprofile für die spezifischen Geräteklassen). Nähere Informationen erhalten Sie von der EtherCAT-Nutzerorganisation:

EtherCAT Technology Group
Ostendstraße 196
90482 Nürnberg
Tel.: +499115405620
Fax : +499115405629
<http://www.ethercat.org/>

Die nachfolgende Tabelle zeigt die Betriebszustände, die die EtherCAT-Schnittstelle unterstützt.

Betriebs- Zustand	Verhalten
Init	Der EtherCAT-Slave wird vom Bus-Master initialisiert.
Boot	Der EtherCAT-Slave befindet sich im Boot-Modus.
PreOp	Die Schnittstelle nimmt am Datenverkehr teil, Ein- und Ausgänge sind noch inaktiv.
SafeOp	Die Schnittstelle kann Daten empfangen, die Ausgänge sind noch inaktiv.
Op	Die Schnittstelle ist vollständig betriebsbereit; Ein- und Ausgänge sind aktiv.

Abb. 119 – Pro II-EtherCAT-SL Rev. E: Betriebszustände

Betriebszustände der EtherCAT-Schnittstelle

5.7.8 Pro II-FlexRay-2 Rev. E

Das Modul Pro II-FlexRay-2 Rev. E besitzt 2 FlexRay-Schnittstellen, wobei jede Schnittstelle einen kompletten FlexRay-Bus mit 2 Kanälen repräsentiert. Das Modul ist so konfigurierbar, dass es ohne weitere Busteilnehmer einen FlexRay-Bus starten kann. Dazu müssen beide FlexRay-Schnittstellen als Coldstarter-Knoten arbeiten und die Schnittstellen per DIP-Schalter zu einem FlexRay-Bus zusammengeschlossen werden.

Ebenfalls per DIP-Schalter kann für jede Schnittstelle kanalweise die Bus-Terminierung eingestellt werden.

Die Beschreibung des FlexRay-Moduls ist in folgende Abschnitte gegliedert:

- FlexRay-Controller
- Hardware-Aufbau
- Modul-Revisionen
- Programmierung

FlexRay-Controller

Das Modul Pro II-FlexRay-2 ist mit zwei FlexRay-Controllern MFR4310 von FreeScale® bestückt und arbeitet nach der „FlexRay Communications System Protocol Specification V2.1“. Sie programmieren die Schnittstelle mit *ADbasic*-Befehlen, die direkt auf die Register des Controllers zugreifen.

Zur Konfiguration und Statusanzeige der FlexRay-Controller dienen die in ihnen enthaltenen Register. Hier werden alle FlexRay-Parameter wie z.B. Busgeschwindigkeit, Bustiming etc. eingestellt.

Weitere Informationen finden Sie auf der Webseite <http://www.freescale.com> in folgenden Dokumentationen von FreeScale®:

- Engineering Bulletin EB683: MFR4310 and MFR4310 differences
- Data sheet MFR4300: MFR4300 FlexRay Communication Controller

Hardware-Aufbau

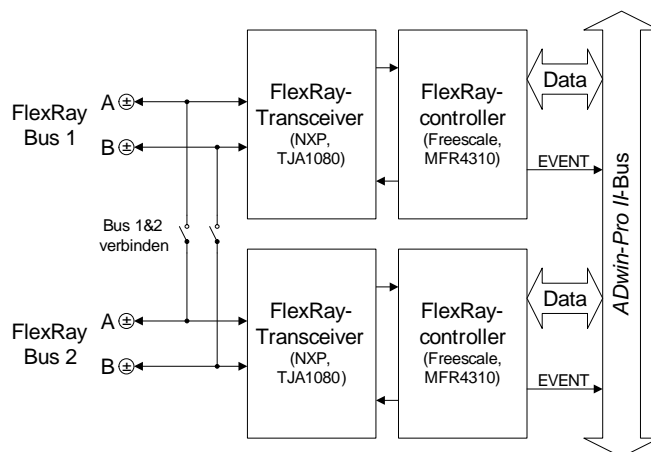


Abb. 120 – Pro II-FlexRay-2 Rev. E: Blockschaltbild

Die Anschlüsse der FlexRay-Schnittstelle stehen auf einem 9-poligen D-Sub-Verbinder zur Verfügung; die Pin-Belegung ist unten dargestellt.

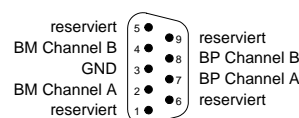


Abb. 121 – Pro II-FlexRay-2 Rev. E: Pinbelegung

Wenn eine Schnittstelle des FlexRay-Moduls das physikalische Ende eines FlexRay-Busses bildet, muss der Bus an an dieser Schnittstelle mit dem entsprechenden DIP-Schalter terminiert werden (siehe Abb. 122); die Kanäle A und B können separat terminiert werden. Da die Kanäle differentiell betrieben werden, müssen Sie zum Terminieren immer beide (!) DIP-Schalter eines Kanals nach rechts umlegen.

Wenn sich das FlexRay-Modul nicht an einem physikalischen Ende des Busses befinden, darf nicht terminiert werden.

Die beiden FlexRay-Schnittstellen können per DIP-Schalter zu einem startfähigen FlexRay-Cluster zusammengeschlossen werden. Für den Zusammenschluss der Kanäle A und B müssen Sie immer beide (!) DIP-Schalter eines Kanals nach oben umlegen (siehe Abb. 122).

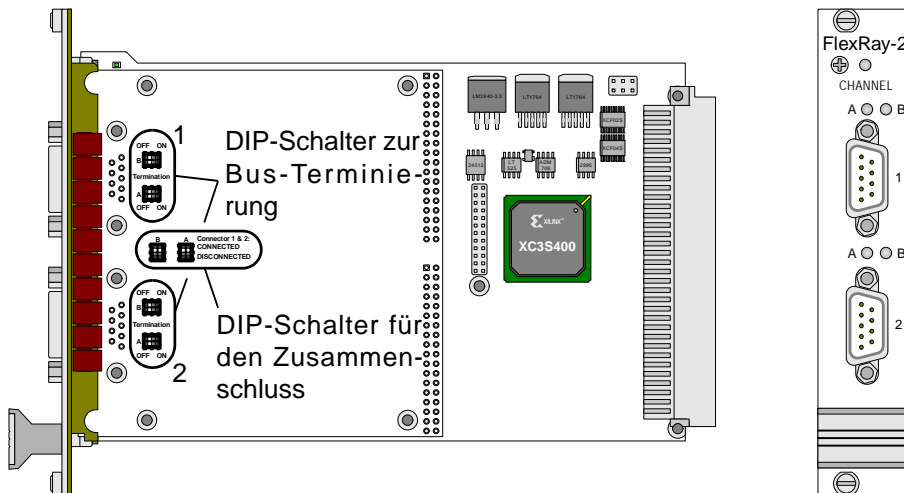


Abb. 122 – Pro II-FlexRay-2 Rev. E: Platine und Frontplatte

Modul-Revisionen

Die Unterschiede der Revisionsstände sind nachfolgend dargestellt:

Revision	Ausgabe- datum	Änderungen zur Vorgänger-Version
E1	08 / 2009	Erst-Version

Die Revisionsbezeichnung befindet sich auf der Frontseite.

Unterschiedliche Revisions-Buchstaben bedeuten unterschiedliche Modul-Eigenschaften und sind separat dokumentiert. Der Revisionsbezeichnung angehängt ist eine Zählnummer für interne Zwecke.

Programmierung

Das Modul wird komfortabel mit *ADbasic*-Befehlen programmiert. Die Befehle sind im Handbuch Pro II-Software und in der Online-Hilfe *ADbasic* erläutert.

Die Include-Datei `ADwinPro_All.inc` enthält Befehle für folgende Bereiche:

Bereich	Befehle
FlexRay-Schnittstelle initialisieren	<code>P2_FlexRay_Init</code>
FlexRay-Controller zurücksetzen	<code>P2_FlexRay_Reset</code>
Schnittstellenversion abfragen	<code>P2_FlexRay_Get_Version</code>
Register setzen und lesen	<code>P2_FlexRay_Read_Word</code> <code>P2_FlexRay_Write_Word</code>
Modul-LED einstellen	<code>P2_Check_LED, P2_Set_LED</code> <code>P2_FlexRay_Set_LED</code>

Einschränkung der
Anwendergruppe

Verfügbarkeit der
Unterlagen



Hilfsmittel

6 Kalibrierung

6.1 Allgemeine Hinweise

Die Digital/Analog- (DAC) und Analog/Digitalwandler (ADC) der **ADwin**-Systeme sind werkseitig kalibriert. Entsprechend der Vorschriften zur Einhaltung der Messgenauigkeit für Ihr Anwendungsgebiet sind die Geräte in regelmäßigen Abständen zu kalibrieren.

Der Hersteller des in dieser Dokumentation beschriebenen Systems geht davon aus, dass an dem Gerät nur qualifiziertes Personal arbeitet.

Qualifiziertes Personal sind Personen, die aufgrund ihrer Ausbildung, Erfahrung und Unterweisung sowie ihrer Kenntnisse über einschlägige Normen, Bestimmungen, Unfallverhütungsvorschriften und Betriebsverhältnisse von dem für die Sicherheit der Anlage Verantwortlichen berechtigt worden sind, die jeweils erforderlichen Tätigkeiten auszuführen und die dabei mögliche Gefahren erkennen und vermeiden können. (Definition für Fachkräfte nach VDE 105 und IEC 60364).

Diese Produktdokumentation und Unterlagen, auf die verwiesen wird, müssen stets verfügbar sein und konsequent beachtet werden. Für Schäden, die durch Missachtung der Informationen in dieser bzw. der weiterführenden Dokumentation entstehen, übernimmt die Firma **Jäger Computergesteuerte Messtechnik GmbH**, Lorsch, keine Haftung.

Zur Kalibrierung benötigen Sie folgende Hilfsmittel:

- Eine Referenzspannungsquelle mit einer Genauigkeit von
 - 10µV bei 18 Bit Wandlern
 - 30µV bei 16 Bit Wandlern
 - 100µV bei 14 Bit Wandlern
- Ein Digital-Multimeter mit einer Genauigkeit von
 - 10µV bei 18 Bit Wandlern
 - 30µV bei 16 Bit Wandlern
 - 100µV bei 14 Bit Wandlern
- Verbindungskabel von den Ein/Ausgängen zur Referenzspannungsquelle und zum Messgerät

Warnung: Gefahr des elektrischen Schlags.

ADwin-Pro II-Systeme verfügen über ein Netzteil, das bei geöffnetem Gerät Zugang zu hochspannungsführenden Leitungen bzw. Anschlüssen ermöglicht. Die Lüftungsschlitze lassen die Durchführung eines Abgleichbestecks mit einem Durchmesser von 2,5 mm zu.



Kalibrieren Sie nur bei geschlossenem Gerät!

Führen Sie keine stromleitenden Objekte durch die Lüftungsschlitze!

6.2 Berechnungsgrundlagen

Die **ADwin**-Systeme arbeiten bei den analogen Ein- und Ausgängen in der Standardeinstellung mit einem Spannungsbereich von -10V...+10V (bipolar $\pm 10V$).

Die 65536 (2^{16}) Digits sind den jeweiligen Spannungsbereichen der ADC und DAC so zugeordnet, dass der Wert für

- 0 (Null) Digit der maximalen negativen Spannung
- 65535 Digit der maximalen positiven Spannung entspricht.

Der Wert für 65536 Digit, genau 10 Volt, liegt damit gerade außerhalb des Messbereiches, womit sich für die 16 Bit AD- bzw. DA-Wandlung ein maximaler Spannungswert von 9,999695 Volt und für die 18 Bit AD-Wandlung von 9,999923706 Volt ergibt.

In der Einstellung bipolar $\pm 10V$ entsteht damit eine Nullpunktverschiebung, die im folgenden auch als Offset bezeichnet wird. Die Verschiebung beträgt $U_{OFF} = -10V$.

Die Quantisierungsstufe (U_{LSB}) gibt die Spannung des niederwertigsten Bit an (Least Significant Bit). In der Standardeinstellung entspricht

- eine Quantisierungsstufe bei einem 18 Bit-Wandler dem 2^{18} -ten Teil von 20V gleich 76,294 μV .
- eine Quantisierungsstufe bei einem 16 Bit-Wandler dem 2^{16} -ten Teil von 20V gleich 305,175 μV .
- eine Quantisierungsstufe bei einem 16 Bit-Wandler dem 2^{14} -ten Teil von 20V gleich 1220,7 μV .

Bei Eingangsmodulen mit einem programmierbaren Verstärker (PGA) können Sie die Eingangsspannung um die Faktoren 2, 4, und 8 verstärken. Damit verkleinert sich der Messbereich um den jeweiligen Verstärkungsfaktor k_V .

Beachten Sie bei Anwendungen mit $k_V > 1$, dass auch die Störsignale entsprechend mit verstärkt werden. Diese können Sie mit der Programmierung von digitalen Filtern im **ADbasic** vermindern.

Um bei Messungen mit einem 14 Bit-ADC und einem 16 Bit-ADC dieselbe Zuordnung der Bits zu erreichen, wird der gewandelte Wert beim 14 Bit-ADC linksbündig in einem Wort (16 Bit) zurückgeliefert, wobei die untersten 2 Bits stets 0 sind (siehe Abb. 123).

Ein 18 Bit-Messwert wird als 24 Bit-Wert zurückgegeben, d.h. der Wert wird um 6 Bits nach links verschoben, die Bits 0...5 sind immer Null.

Bit-Nr.	31...2 4	23...1 6	15...6	5...2	1...0
Inhalt	0	18-Bit Messwert in den Bits 6...23	0	0	
	0	0	16 Bit-Messwert in den Bits 0...15		
	0	0	14 Bit-Messwert in den Bits 2...15	0	
oberes Wort			unteres Wort		

Abb. 123 – Bit-Zuordnung bei verschiedenen Auflösungen

Die 16384 Digits des 14 Bit-ADC werden auf die 65535 Digits des 16 Bit-ADC abgebildet. Damit entsprechen 4 Digits des 16 Bit-ADC einem Digit des 14 Bit-ADC.

Für einen DAC gilt:

Spannungsbereich

Zuordnung von Digits zu Spannung

Nullpunktverschiebung

Least Significant Bit
 U_{LSB}

Verstärkung k_V

Zuordnung der Bits

DAC

ADC

Für einen ADC gilt:

$$U_{OUT} = \text{Digits} \cdot U_{LSB} + U_{OFF}$$

$$\text{Digits} = \frac{U_{OUT} - U_{OFF}}{U_{LSB}}$$

$$\text{Digits} = \frac{k_V \cdot U_{IN} - U_{OFF}}{U_{LSB}}$$

$$U_{IN} = \frac{\text{Digits} \cdot U_{LSB} + U_{OFF}}{k_V}$$

Toleranzbereiche

Geringe Abweichungen zu den rechnerischen Werten können innerhalb der Toleranzbereiche einzelner Bauteile liegen. Es gibt 2 charakteristische Abweichungsarten, die in diesem Handbuch (in LSB) angegeben sind:

INL

- Die integrale Nicht-Linearität (INL) beschreibt die maximale Abweichung von der Geraden über den gesamten Eingangsspannungsbereich.

DNL

- Die differentielle Nicht-Linearität (DNL) beschreibt die maximale Abweichung von der Breite einer Quantisierungsstufe.

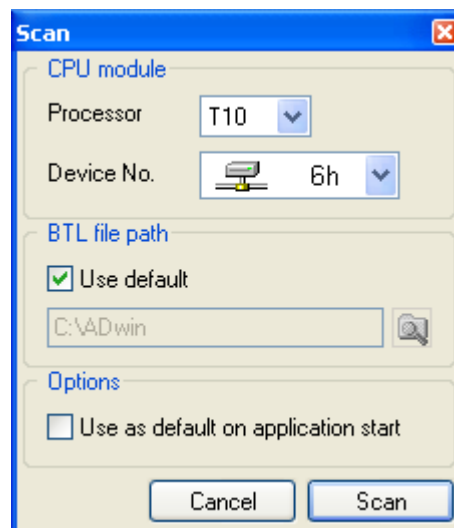
6.3 Kalibrieren

Legen Sie den Spannungsbereich des Moduls fest.

Die Kalibrierung muss bei Betriebstemperatur erfolgen. Bei einer Temperatur des Gerätes von ca. 20...25 Grad Celsius (Raumtemperatur) ist die Betriebstemperatur etwa 30 Minuten nach dem Einschalten erreicht.

Beachten Sie bitte die allgemeinen Hinweise in Kapitel 6.1.

Rufen Sie das Programm `ADpro.exe` aus dem Windows-Startmenü auf unter `Programme\ADwin`. Es erscheint das Einstellungsfenster `Scan`.



Wenn beim Programmstart eine Fehlermeldung auftritt, prüfen Sie bitte, ob auf Ihrem Rechner das Programmpaket `.NET Framework 2.0` installiert ist.

Beachten Sie: Mit dem nächsten Schritt stoppen Sie alle Prozesse und setzen alle Moduleinstellungen zurück!

Stellen Sie die Daten für das zu kalibrierende *ADwin*-System ein. Mit der Schaltfläche `Scan` wird eine Verbindung zum *ADwin*-System aufgebaut und



Vorbereitung der Hardware

Systeminformationen werden gelesen. Hierbei initialisiert das Programm `ADpro.exe` das *ADwin*-System, d.h. es beendet und löscht noch laufende Prozesse.

Wenn Ihr *ADwin*-System korrekt initialisiert ist, erscheint das Fenster `ADpro`.

Sie erhalten eine Fehlermeldung, wenn

- Ihr **ADwin**-System nicht erfolgreich gebootet wurde.
- die Angaben im Fenster `Scan` nicht zutreffen.

Überprüfen und – falls erforderlich – korrigieren Sie die Angaben im Fenster. Starten Sie Ihr System nun nochmals mit der Schaltfläche „Scan“.

Markieren Sie das zu kalibrierende Modul im Fenster `ADpro` und wählen Sie den Menüeintrag `Calibration` im Menü `Module`. Wenn der Menüeintrag `Calibration` fehlt, kann das gewählte Modul nicht kalibriert werden.

Schließen Sie das Messgerät und die Referenzspannungsquelle am Modul an.

Folgen Sie nun den Anweisungen, die im Eingabefenster auf englisch angezeigt werden. Beachten Sie bitte den Unterschied zwischen analogen Eingangs-Modulen mit und ohne Multiplexer:

- Analoge Eingangs-Module mit Multiplexer (`AIIn-...`): Die Kalibrierung des ADC erfolgt über den Eingangskanal 1.
- Analoge Eingangs-Module ohne Multiplexer (`AIIn-F-...`): Der jeweils angeschlossene Kanal wird im Fenster „Input channel“ angewählt.

Kalibrieren

7 Zubehör

7.1 LEMO-Kabelsätze für ADwin-Pro-Systeme

Pro-CS-1	4 x Kabel 200 mm (7,8 inch) und 4 x Kabel 400 mm (15,7 inch)
Pro-CS-2	4 x Kabel 400 mm (15,7 inch) und 4 x Kabel 800 mm (31,5 inch)
Pro-CS-3	4 x Kabel 1000 mm (39,4 inch) und 4 x Kabel 1500 mm (59 inch)
Pro-CS-4	4 x Kabel 5000 mm (196,8 inch)

Alle Kabel haben LEMO-Stecker an beiden Enden.

7.2 LEMO-Adaptersätze

Pro-As-1	4 Adapter: LEMO-Buchsen auf BNC-Stecker
Pro-As-3	4 LEMO-Buchsen T-Stück (1 x Stecker, 2 x Buchse)
Pro-As-4	4 Adapter: LEMO-Buchse - LEMO-Buchse
Pro-As-5	4 LEMO-Buchse mit 50 Ohm-Abschluss
Pro-As-6	4 Adapter: LEMO-Stecker - Kabel (Länge: 10cm) - BNC-Buchse

7.3 Bezugsadressen

7.3.1 LEMO-Stecker

Pro-Module sind mit folgenden LEMO-Steckverbindern ausgerüstet:

- Buchsen/Stecker der Serie 00 NIM-CAMAC, 1-polig
 - Kabelstecker: Bauform FFS (straight cable plug)
 - Einbaubuchse: Bauform ERN
- Buchsen/Stecker der Serie 00 Multikontakt, 2-polig
 - Kabelstecker: Bauform FGG
 - Einbaubuchse: Bauform EGG
- Modul Pt100: Buchsen/Stecker der Serie B
 - Kabelstecker: Bauform FGG
 - Einbaubuchse: Bauform EGG

Hersteller der LEMO-Steckverbinder:

LEMO GmbH
Hanns-Schwindt-Straße 6
Postfach 820529
D-81829 München

Tel.: +49 89 42770-3
Fax: +49 89 4202192
E-Mail: lemo@info.de
Internet: www.lemo.com

7.3.2 Stromversorgungs-Stecker Pro-Mini

Der Anschlussstecker für die externe Stromversorgung des Gehäuses Pro-Mini stammt von Phoenix Contact GmbH:

Combicon-Steckerteil, Raster 5,0mm, Typ MSTB 2,5/ 3-STF;
Artikelnr. 1786844 (Stand Dez. 2005)

Hersteller des Steckers:

Phoenix Contact GmbH & Co. KG
Flachsmarktstraße 8
D-32825 Blomberg

Tel.: +49 5235 300
Fax: +49 5235 341 200
E-Mail: info@phoenixcontact.com
Internet: www.phoenixcontact.de

Anhang

A.1 RoHS Konformitätserklärung

Die Richtlinie 2002/95/EG der Europäischen Union zur Beschränkung und Verwendung gefährlicher Stoffe in elektrischen und elektronischen Geräten (RoHS-Richtlinie) ist am 1. Juli 2006 in Kraft getreten.

Dabei handelt es sich um folgende Substanzen:

- Blei (Pb)
- Cadmium (Cd)
- Hexavalentes Chrom (Cr VI)
- Polybromierte Biphenyle (PBB)
- Polybromierte Diphenylether (PBDE)
- Quecksilber (Hg)

Die Produktlinie *ADwin-Pro II* erfüllt die Voraussetzungen der RoHS-Richtlinie in allen gelieferten Varianten.

A.2 Liste der Module

Pro-CPU-T11;	14
Pro II-Boot;	16
Pro II-MIO-4 Rev. E;	18
Pro II-MIO-4-ET1 Rev. E;	25
Pro II-Aln-8/18-L2 Rev. E;	38
Pro II-Aln-32/18-D Rev. E;	41
Pro II-Aln-8/18-8B Rev. E;	44
Pro II-Aln-16/18-8B Rev. E;	46
Pro II-Aln-F-4/14 Rev. E;	48
Pro II-Aln-F-8/14 Rev. E;	52
Pro II-Aln-F-4/16 Rev. E;	56
Pro II-Aln-F-8/16 Rev. E;	60
Pro II-Aln-F-4/18 Rev. E;	64
Pro II-Aln-F-8/18 Rev. E;	66
Pro II-AOut-4/16 Rev. E;	70
Pro II-AOut-8/16 Rev. E;	72
Pro II-DIO-32 Rev. E;	75
Pro II-DIO-32-TiCo Rev. E;	75
Pro II-DIO-32 Rev. E;	77
Pro II-DIO-32-TiCo Rev. E;	77
Pro II-OPT-16 Rev. E;	80
Pro II-REL-16 Rev. E;	84
Pro II-TRA-16 Rev. E;	86
Pro II-TRA-16-G Rev. E;	86
Pro II-PWM-16 Rev. E;	88
Pro II-PWM-16-I Rev. E;	88
Pro II-CNT-T Rev. E;	90
Pro II-CNT-D Rev. E;	90
Pro II-CNT-I Rev. E;	90
Pro II-RTD-8 Rev. E;	96
Pro II-TC-8 ISO Rev. E;	100

Pro II-CAN-2 Rev. E;	102
Pro II-RSxxx Rev. E;	108
Pro II-LIN-2 Rev. E;	112
Pro II-Profi-SL Rev. E;	114
Pro II-EtherCAT-SL Rev. E;	117
Pro II-FlexRay-2 Rev. E;	120

