

ADwin-Gold- USB / -ENET

Handbuch



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

Inhaltsverzeichnis

Typografische Konventionen	IV
1 Zu diesem Handbuch	1
2 Systembeschreibung	2
2.1 ADwin Systemkonzept	2
2.2 Das ADwin-Gold-System	4
3 Betriebliche Umgebung	6
4 Inbetriebnahme der Hardware	7
5 Ein- und Ausgänge	9
5.1 Analoge Ein- und Ausgänge	10
5.2 Digitale Ein- und Ausgänge	14
5.3 Zeitkritische Aufgaben	15
6 Kalibrierung	19
6.1 Allgemeine Hinweise	19
6.2 Kalibrierung durchführen	19
7 DA-Erweiterung	23
8 CO1-Zählererweiterung	24
8.1 Hardware	24
8.2 Software	26
8.3 Betriebsart Impuls-/Ereigniszähler	28
8.4 Betriebsart Impulsbreiten- und Periodendauer-Messung	30
8.5 Hardware-Adressen (CO1-Erweiterung)	32
9 CAN-Erweiterung	33
9.1 SSI-Decoder	34
9.2 CAN-Schnittstelle	36
9.3 RSxxx-Schnittstellen	40
10 ADwin-Gold-Boot	45
11 Zubehör	46
12 Software	46
Anhang	A-1
A.1 Technische Daten	A-1
A.2 Hardware-Adressen - Gesamtübersicht	A-5
A.3 Hardware-Revisionen	A-7
A.4 Abbildungsverzeichnis	A-8
A.5 Index	A-9

Typografische Konventionen



<C:\ADwin\ ...>

Programmtext

Var_1

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.

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

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

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 Zu diesem Handbuch

Dieses Handbuch enthält umfassende Informationen für den Betrieb Ihres **ADwin-Gold**-Systems. Es wird ergänzt durch

- das Handbuch „**ADwin** Treiber-Installation“, das die Schnittstellen-Installation zu allen **ADwin**-Systemen beschreibt.
Beginnen Sie hier die Installation Ihres Systems!
- die Beschreibung des Konfigurationsprogramms **ADconfig**, mit dem Sie die Kommunikation von der jeweiligen Schnittstelle (Interface) zur Ihrem **ADwin**-Gerät einrichten.
- das Handbuch **ADbasic**, das alle Befehle für den gleichnamigen Compiler enthält sowie das Funktionsprinzip von **ADwin**-Systemen näher erläutert.
- die Installations- und Befehlsbeschreibungen für die Treiber der gängigen Entwicklungsumgebungen.



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 ICE 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.

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.

Änderungen vorbehalten.

Hotline-Adresse siehe vordere Umschlagseite, innen.

Einschränkung der Anwendergruppe

Verfügbarkeit der Unterlagen



Rechtliche Grundlagen

2 Systembeschreibung

2.1 ADwin Systemkonzept

ADwin-Systeme garantieren den schnellen und zeitlich präzisen Ablauf von Messdatenerfassungs- und Automatisierungsaufgaben mit sehr schnellen Echtzeitanforderungen. Das bietet eine ideale Basis für Anwendungen wie:

- sehr schnelle digitale Regler
- sehr schnelle Steuerungen
- Datenerfassung mit sehr schneller Online-Analyse der Messdaten
- Überwachung komplexer Triggerbedingungen und vieles mehr

ADwin-Systeme sind optimiert für Abläufe mit **kurzen Prozesszykluszeiten** von einer Millisekunde bis zu wenigen Mikrosekunden.

Systemmerkmale

Das **ADwin**-System besitzt analoge und digitale Ein- und Ausgänge, einen schnellen Prozessor (32-Bit-Floating-Point Signalprozessor) und lokalen Speicher. Der Prozessor übernimmt die gesamte Echtzeitverarbeitung im System. Die Anwendungen **laufen eigenständig** und unabhängig vom PC und dessen Auslastung.

Prozessor

Der Prozessor des **ADwin**-Systems **verarbeitet jeden Messwert sofort**.

In einem Zyklus können die Zustände von Eingängen erfasst, diese mit beliebigen mathematischen Funktionen verarbeitet und auf dieses Ergebnis reagiert werden, und das sogar bei sehr kurzen Prozesszykluszeiten von wenigen Mikrosekunden. Es ergibt sich eine perfekte und logische Arbeitsteilung: auf dem PC läuft ein Programm zur Visualisierung von Daten, zur Eingabe und Bedienung der Abläufe mit Netzwerk- und Datenbankzugriffen, während gleichzeitig auf dem Prozessor des **ADwin**-Systems alle Aufgaben, die Echtzeit erfordern, abgearbeitet werden.

Echtzeitkern

Das Betriebssystem für den DSP des **ADwin**-Systems wurde auf das Erreichen kürzester Reaktionszeiten optimiert. Dieser Echtzeitkern verwaltet parallele Prozesse, die im **Multitasking-Verfahren** gleichzeitig ablaufen können. Prozesse mit niedriger Priorität werden in einem Zeitscheibenverfahren verwaltet. Prozesse mit hoher Priorität unterbrechen bei ihrer Anforderung alle niedrigpriorisierten Prozesse und werden sofort vollständig ausgeführt (präemptives Multitasking). Hochpriorisierte Prozesse werden zeitgesteuert oder von externen Events (Trigger) ausgelöst.

Zeitsteuerung

Für den präzisen Aufruf hochpriorisierter Prozesse sorgt der im System integrierte **Timer**. Er hat eine Auflösung von 25 Nanosekunden. Zu beachten ist die extrem kurze Reaktionszeit von nur 300 Nanosekunden beim Wechsel von einem niedrig- zu einem hochpriorisierten Prozess. Ein ständig laufender Kommunikationsprozess ermöglicht einen kontinuierlichen Datenaustausch zwischen dem **ADwin**-System und dem PC auch während laufenden Anwendungen. Dabei hat die Kommunikation keinen Einfluss auf die Echtzeitfähigkeit des **ADwin**-Systems, trotzdem können jederzeit Daten ausgetauscht werden.

ADbasic

Das Echtzeit-Entwicklungstool **ADbasic** ermöglicht die einfache und schnelle Erstellung von zeitkritischen Programmen für **ADwin**-Systeme. **ADbasic** ist eine **integrierte Entwicklungsumgebung** unter Windows mit Möglichkeiten zum Online-Debugging. Die gewohnte, leicht erlernbare BASIC-Befehlssyntax wurde um Funktionen für den direkten Zugriff auf Ein- und Ausgänge sowie zur Prozesssteuerung und zur Kommunikation mit dem PC erweitert.

Die Kommunikation zwischen ADwin-System und PC

Das **ADwin**-System ist mit dem PC über eine **USB- oder Ethernet-Schnittstelle** verbunden. Über diese Schnittstelle kann das **ADwin**-System nach dem Einschalten vom PC gebootet werden. Nach dem Booten erwartet das **ADwin**-Betriebssystem Kommandos vom PC, die es abarbeitet.

Es gibt zwei Arten von Kommandos: Zum einen Kommandos, die nur Daten vom PC an das **ADwin**-System schicken, wie z.B. „Prozess laden“, „Prozess starten“ oder „Parameter setzen“, zum anderen Kommandos, die von dem **ADwin**-System eine Antwort erwarten, wie z.B. „Variablen lesen“ oder „Datensätze lesen“. Beide Arten von Kommandos werden vom **ADwin**-System sofort bearbeitet beziehungsweise sofort und vollständig beantwortet. Das **ADwin**-System schickt nie unaufgefordert Daten an den PC. Die Datenübertragung an den PC ist immer nur die Antwort auf ein Kommando vom PC. Dadurch wird die Einbindung des **ADwin**-Systems in die unterschiedlichsten Programmiersprachen und messtechnischen Standardsoftwarepakete sehr erleichtert, denn diese müssen nur in der Lage sein, eine Funktion aufzurufen und den Rückgabewert zu verarbeiten.

Unter Windows 95/98/NT/ME/2000/XP stehen eine **DLL-** und eine **ActiveX-Schnittstelle** zur Verfügung. Darauf basierend gibt es Treiber für die folgenden **Entwicklungsumgebungen**:

.NET, Visual Basic, Visual-C, C/C++, C#, Delphi, VBA (Excel, Access, Word), TestPoint, LabVIEW / LabWINDOWS, Agilent VEE (HP-VEE), InTouch, *DIAdem*, MATLAB.

Die einfache, kommandoorientierte Kommunikation mit dem **ADwin**-System ermöglicht es, dass mehrere Windows Programme in Abstimmung miteinander gleichzeitig auf das gleiche **ADwin**-System zugreifen. Dies ist vor allem bei der Programmentwicklung und bei der Inbetriebnahme ein großer Vorteil.

Schnittstellen

Befehlsverarbeitung

Software Schnittstellen

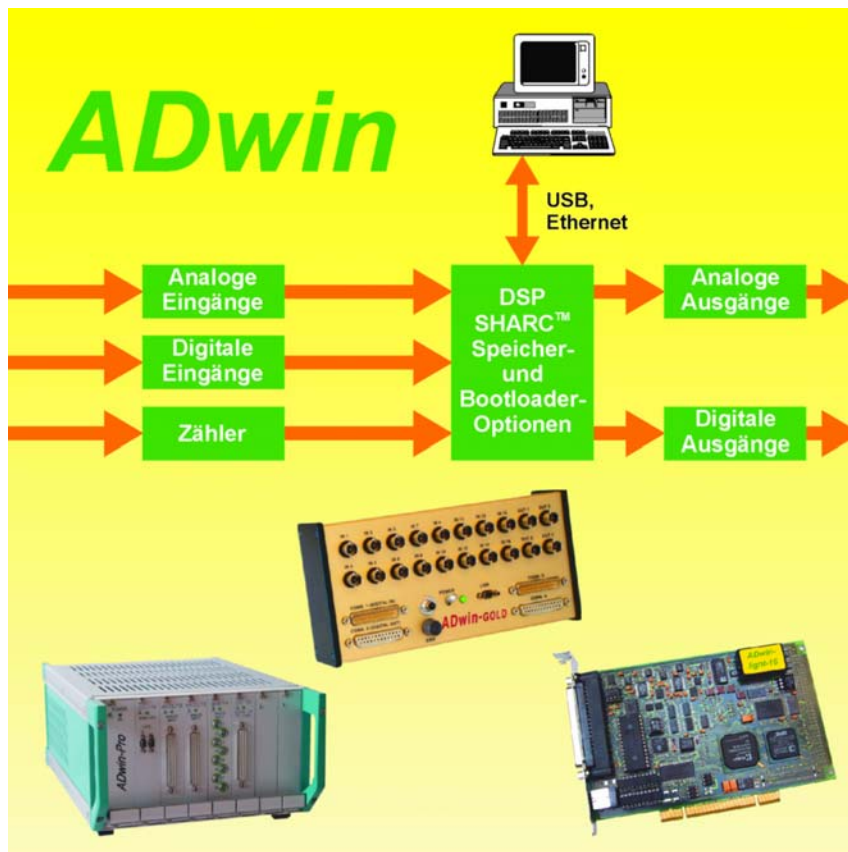


Abb. 1 – Konzept der **ADwin**-Systeme

Prozessor und Speicher

2.2 Das ADwin-Gold-System

Das **ADwin-Gold**-System besitzt den digitalen **32Bit-Signalprozessor T9** (SHARC ADSP 21062) von Analog Devices mit Floating-Point- und Integer-Verarbeitung. Er übernimmt die gesamte Messwerterfassung, Online-Verarbeitung und Signalausgabe und kann in Verbindung mit A/D-Wandlern jeden Messwert mit Abtastraten bis zu mehreren 100kHz sofort verarbeiten.

Der interne **Speicher mit 256kB** hat eine sehr kurze Zugriffszeit von 25ns und nimmt das komplette **ADwin**-Betriebssystem, die **ADbasic**-Prozesse und alle Variablen auf.

Für maximale Zugriffsgeschwindigkeiten liegen alle Ein- und Ausgänge direkt im Adressbereich des DSP. Zum Zwischenspeichern größerer Datenmengen benutzt der DSP einen externen Speicher von 16MB (DRAM; optional 64MB).

Analoge Eingänge

Das System hat **16 analoge Eingänge** mit BNC-Buchsen (alternativ: Sub-D-Buchsen), die in zwei Gruppen jeweils mit einem Multiplexer verbunden sind. Deren Ausgänge werden wahlweise mit einem 14Bit oder 16Bit Analog-Digital-Wandler (ADC) konvertiert (siehe Abb. 2 „Funktionsschema des ADwin-Gold“). Mit dem 14Bit-ADC kann sehr schnell, mit dem 16Bit-ADC sehr genau gemessen werden.

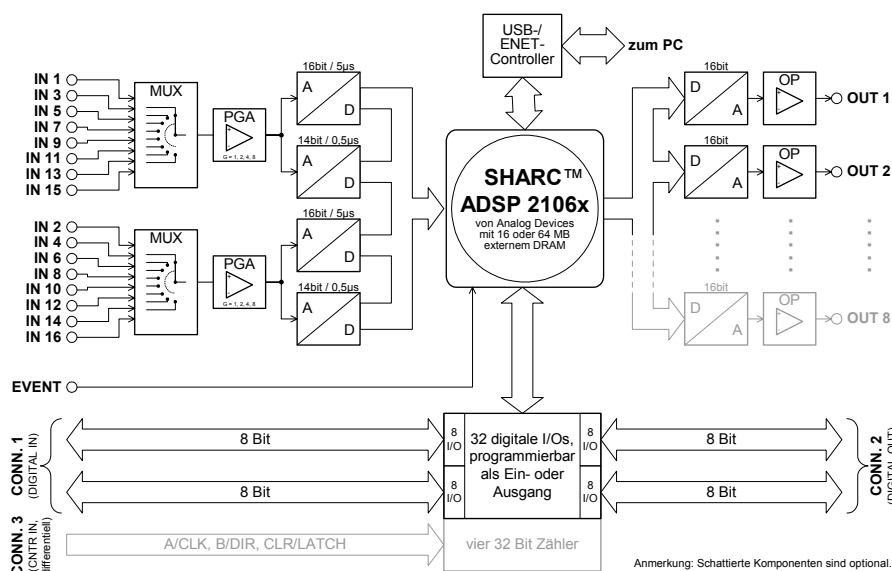


Abb. 2 – Funktionsschema des **ADwin-Gold**

Analoge Ausgänge

In der Standardversion verfügt das **ADwin-Gold** über **2 analoge Ausgänge** (optional 8) mit 16Bit Auflösung und einem Ausgangsspannungsbereich von -10V...+10V. Per Software können Sie die Ausgabe der Spannung aller DAC synchronisieren.

Digitale Ein- und Ausgänge

Auf zwei 25-poligen Sub-D-Anschlüssen stehen **32 digitale Ein- oder Ausgänge** zur Verfügung. Sie sind in Gruppen zu jeweils 8 als Ein- oder Ausgang frei programmierbar. Die Ein- bzw. Ausgänge sind TTL-kompatibel.

Trigger-Eingang (EVENT)

Das **ADwin-Gold**-System besitzt einen Trigger-Eingang (EVENT, siehe auch Kapitel 5.2 „Digitale Ein- und Ausgänge“). Hiermit können Prozesse durch ein Signal (Trigger) ausgelöst und sofort vollständig abgearbeitet werden. (siehe **ADbasic**-Handbuch, Kapitel Struktur des **ADbasic**-Programms).

Alle analogen Daten-Eingänge und -Ausgänge des Geräts sind differentiell.

Die Verbindung zwischen **ADwin-Gold**-System und PC wird (je nach Systemausführung) über die USB- oder das Ethernet-Schnittstelle hergestellt.

Der Standard-Lieferumfang des **ADwin-Gold**-Systems umfasst

- das **ADwin-Gold**-Gerät mit USB- oder Ethernet-Schnittstelle,
- ein USB-Kabel oder „Cross-over“ Ethernet-Kabel vom PC zum Gold-Gerät (Länge ca. 1,8 m),
- Power-Adapter: Ein dreipoliges, verpolungssicheres Stromversorgungs-Kabel an einem Slotblech mit Steckbuchse,
- Stromversorgungskabel vom Power-Adapter zum System,
- **ADwin**-CD,
- Handbuch „Treiber-Installation“,
- das vorliegende Hardware-Handbuch.

2.2.1 Optionen (nicht nachrüstbar)

Folgende Zusatzoptionen stehen zur Verfügung:

- **Gold-D**: Alle Ein- und Ausgänge liegen auf D-Sub-Buchsen/Steckern, darunter auch die analogen Eingänge (anstelle von BNC-Buchsen).
- **Gold-DA**: 6 zusätzliche analoge Ausgänge (differentiell), jeweils mit einem 16Bit DAC.
- **Gold-CO1**: 4 Stück 32Bit-Zähler, wahlweise zur Periodendauermessung, als Impulzzähler, als Vorwärts-/Rückwärtszähler mit Takt-/Richtung oder als Vier-Flanken-Auswertung für Inkremental-Encoder.
- **Gold-CAN**: 4 Decoder zum Anschluss von Inkremental-Encodern mit SSI-Schnittstelle, 2 CAN-Schnittstellen (beide high speed oder low speed) sowie 2 RSxxx-Schnittstellen (RS232, RS485). Diese Option ist nur in Kombination mit der Option Gold-D erhältlich.
- **GOLD-MEM-64**: Externer Speicher mit 64 MB anstatt 16 MB sowie interner CPU-Speicher mit 512 kB anstatt 256 kB.
- **Gold-Boot**: Flash-EPROM-Bootloader zum eigenständigen Betrieb ohne PC (**nur** in Verbindung mit **Gold-ENET**).

Sofern nicht explizit ausgeschlossen, sind alle Zusatzoptionen miteinander kombiniert lieferbar.

2.2.2 Zubehör

- **ADbasic**, Echtzeit-Entwicklungsumgebung für alle **ADwin**-Systeme.
- **ADwin-Gold-pow**: externes Netzteil (u.a. erforderlich für Notebook-Betrieb).
- **Gold-Mount**: Gehäuseumbau zur Hutschienen-Montage in einem Schaltschrank mit isolierten Clipsen.
- Einzelner Stromversorgungs-Stecker für ein selbst-konfektioniertes Stromversorgungs-Kabel.

Standard-Lieferumfang

Optionen

Zubehör

3 Betriebliche Umgebung

Die **ADwin-Gold**-Elektronik ist in einem geschlossenen Aluminiumgehäuse untergebracht, und das System darf nur in diesem Zustand betrieben werden. Mit entsprechendem Zubehör ist die Unterbringung in Schaltschränken oder der mobile Betrieb (z.B. im Kfz) möglich (siehe Kapitel 2.2.2 „Zubehör“).

Das **ADwin-Gold-Gerät muss geerdet werden**, um

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

Verbinden Sie dazu die GND-Buchse, die intern mit der Masse und dem Gehäuse verbunden ist, über ein kurzes impedanzarmes Masseband mit dem zentralen Erdungspunkt Ihrer Anlage.

Das Stromversorgungskabel stellt eine galvanische Verbindung zwischen dem PC und dem ADwin-Gold-Gerät her.

Die Liefervariante mit USB-Schnittstelle hat über diese eine galvanische Verbindung zum PC sowie ggf. über die Stromversorgung.

Bei der Liefervariante mit Ethernet-Schnittstelle 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** Kabel mit beidseitig aufgelegtem Schirm. Auch hier sollte das Ableiten von Störungen über das Gehäuse mit der Verwendung von Schirmklemmen reduziert werden.

Die Abschirmung von BNC-Kabeln wird üblicherweise als differentielle Masse verwendet und verliert dadurch an Schirmwirkung. Daher sind BNC-Kabel bei differentiellen Messungen Störeinflüssen ausgesetzt. Für die Signal- und Datenübertragung außerhalb des Schaltschranks ist eine Umsetzung auf Datenübertragungskabel erforderlich, die paarig verdreht (twisted pair) und kanalweise geschirmt sind.

Das **ADwin-Gold**-System wird extern mit einer Schutz-Kleinspannung von 10V bis 35V versorgt; intern wird es mit einer Spannung von +5V und ±15V gegen GND betrieben. Es stellt von dieser Seite keine Gefahr für Leib und Leben dar. Für den Betrieb mit einem externem Netzteil gelten die Angaben des Herstellers.

ADwin-Gold ist für den Betrieb in trockenen Räumen konzipiert. Am Einbauort sollen eine Umgebungstemperatur von +5°C ... +50°C und eine relative Luftfeuchte von 0 ... 80% (nicht kondensierend, siehe Anhang) 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.

Erdung



Galvanische Kopplung

Ausgleichsströme ausschließen



BNC-Kabel

Schutzkleinspannung

Umgebungs-klima

Gehäusetemperatur



4 Inbetriebnahme der Hardware

Schließen Sie bei der **Inbetriebnahme** keine Kabel an das **ADwin-Gold**-System an, bevor Sie nicht **folgende Schritte** durchgeführt haben:

- Führen Sie die Installation der Treiber und Stromversorgung am PC oder Notebook vollständig durch (siehe Handbuch: „**ADwin**-Treiber-Installation“).
- Verbinden Sie das System nur mit dem PC oder Notebook (s.u.).
- Lesen Sie das Kapitel 5 „Ein- und Ausgänge“ in diesem Handbuch.
- Beginnen Sie erst jetzt mit dem Anschluss von Ein- und Ausgängen.

Beachten Sie bitte, dass über Stromversorgungskabel, USB- und Ethernet-Leitungen zwischen **ADwin-Gold**-System und PC eine galvanische Verbindung besteht (siehe Kapitel 3, Abschnitt „Galvanische Kopplung“).

Achten Sie auf eine zuverlässige Spannungsversorgung.
Im Standardlieferungsumfang betrifft das den PC, ansonsten auch das externe Netzteil, bei Betrieb im Fahrzeug die Batteriespannung.

Die Stromversorgung des **ADwin-Gold** mit 12V (siehe Anhang, Technische Daten) erfolgt über den Einbaustecker links neben dem Power-Schalter bzw. über der GND-Buchse (siehe Abb. 3). Schließen Sie dort einen 3-poligen Subminiatur-Rundsteckverbinder an; die Pin-Belegung entnehmen Sie bitte der folgenden Zeichnung.

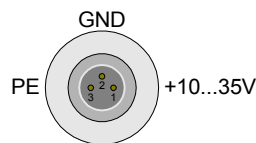


Abb. 3 – Stromversorgungsstecker (männlich)

Für die Benutzung eines separaten Netzteils benötigen Sie den o.g. Rundsteckverbinder. Sie erhalten diesen z.B. beim folgenden Hersteller unter der Bestell-Nummer 712 299-0406-00-03 (Serie 712):

Franz Binder GmbH + Co. elektrische Bauelemente KG
Rötelstrasse 27
D-74172 Neckarsulm
Tel.: 07132 / 325-0
www.binder-connector.de

Beim Betrieb mit einem Notebook muss die Stromversorgung durch ein separates Netzteil erfolgen (siehe Kapitel 2.2.2 auf Seite 5). Bitte beachten Sie, dass dieses ausreichend dimensioniert ist.

Achten Sie bei der Verwendung strombegrenzender Netzteile darauf, dass beim Einschalten der Strombedarf ein Mehrfaches des Betriebsstroms betragen kann. Genaue Angaben finden Sie bei den Technischen Daten (Anhang).

Bei Ausfall der Betriebsspannung gehen alle ungesicherten Daten verloren. Nicht definierte Daten (Werte) können unter ungünstigen Umständen zu Schäden in Ihrer Anlage führen.

Wenn Sie die **ADwin**-Treiber-Installation und die Einstellungen im **ADbasic**-Menü „Options\Compiler“ abgeschlossen haben, schließen Sie jetzt das USB- oder Ethernet-Datenübertragungs- und das Stromversorgungs-Kabel an. Starten Sie anschließend Ihren Rechner.

Um versehentliches Ausschalten zu verhindern, besitzt der Ein-/Ausschalt-Hebel eine Umschaltsperr. Ziehen Sie zum Einschalten den Hebel leicht her-



Sicherstellen der Spannungsversorgung

Stromversorgung



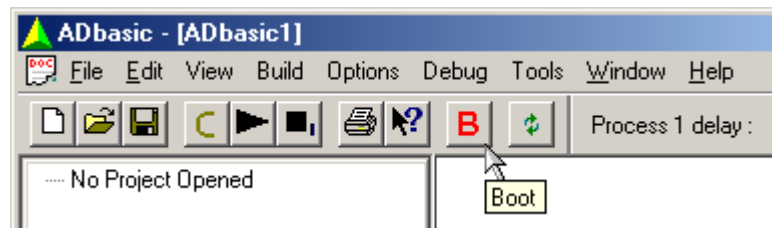
PC anschließen

Einschalten

Booten

aus (ca. 1,5mm) und legen ihn in Richtung „Power“ um. Damit ist das Gerät eingeschaltet und die LED leuchtet rot auf.

Starten Sie **ADbasic** und booten das **ADwin**-System durch Anklicken des Boot-Schaltfläche **B**.



Das Blinken der LED (jetzt grün) und die Anzeige in der Statuszeile: „ADwin is booted“ zeigen an, dass das Betriebssystem richtig geladen ist und **ADbasic** eine Verbindung zum **ADwin**-System herstellen kann (wenn nicht, überprüfen Sie zuerst die Anschlüsse).

Programme mit ADbasic

Die Programmierung von **ADwin**-Systemen ist im **ADbasic**-Handbuch ausführlich beschrieben.

Beginnen Sie mit Programmbeispielen aus dem **ADbasic**-Tutorial.

5 Ein- und Ausgänge

Alle Ein- und Ausgänge dürfen nur im Bereich der angegebenen Spezifikation betrieben werden (siehe Anhang A.1 Technische Daten). Im Zweifel wenden Sie sich bitte an den Hersteller des Gerätes, das Sie an das **ADwin-Gold**-System anschließen wollen.

Offene Eingänge können zu Fehlern führen – vor allem in einer nicht störungsfreien Umgebung. Zu Ihrer Sicherheit legen Sie nicht benutzte Eingänge möglichst nah an Stecker oder Buchse des **ADwin-Gold** auf einen definierten Pegel (z.B. GND). Schließen Sie keine Kabel mit offenem Ende an die Eingänge an; dies kann Störimpulse an den Eingängen verursachen.

Ausnahme hierzu ist der Event-Eingang, der bereits einen internen Pull-down-Widerstand (10 k Ω) besitzt.

Für eine schnelle und einfache Programmierung gibt es im Compiler **ADbasic** Standardbefehle, die **einfaches Messen bzw. Ausgeben von Daten** ermöglichen (siehe auch **ADbasic**-Handbuch). Verwenden Sie andere Befehle (z.B. direkter Registerzugriff) erst, wenn extrem zeitkritische oder besondere Aufgaben es erfordern (siehe auch Handbuch **ADbasic**).

Genauere Angaben zu den analogen sowie den digitalen Ein- und Ausgängen finden Sie in den folgenden Kapiteln.

Die Pin-Belegung der Anschlüsse CONN. 1 und CONN. 2 (Gold-D: DIO00-15 und DIO16-31) ist auf Seite 15 dargestellt.

Anschlüsse



Standardbefehle

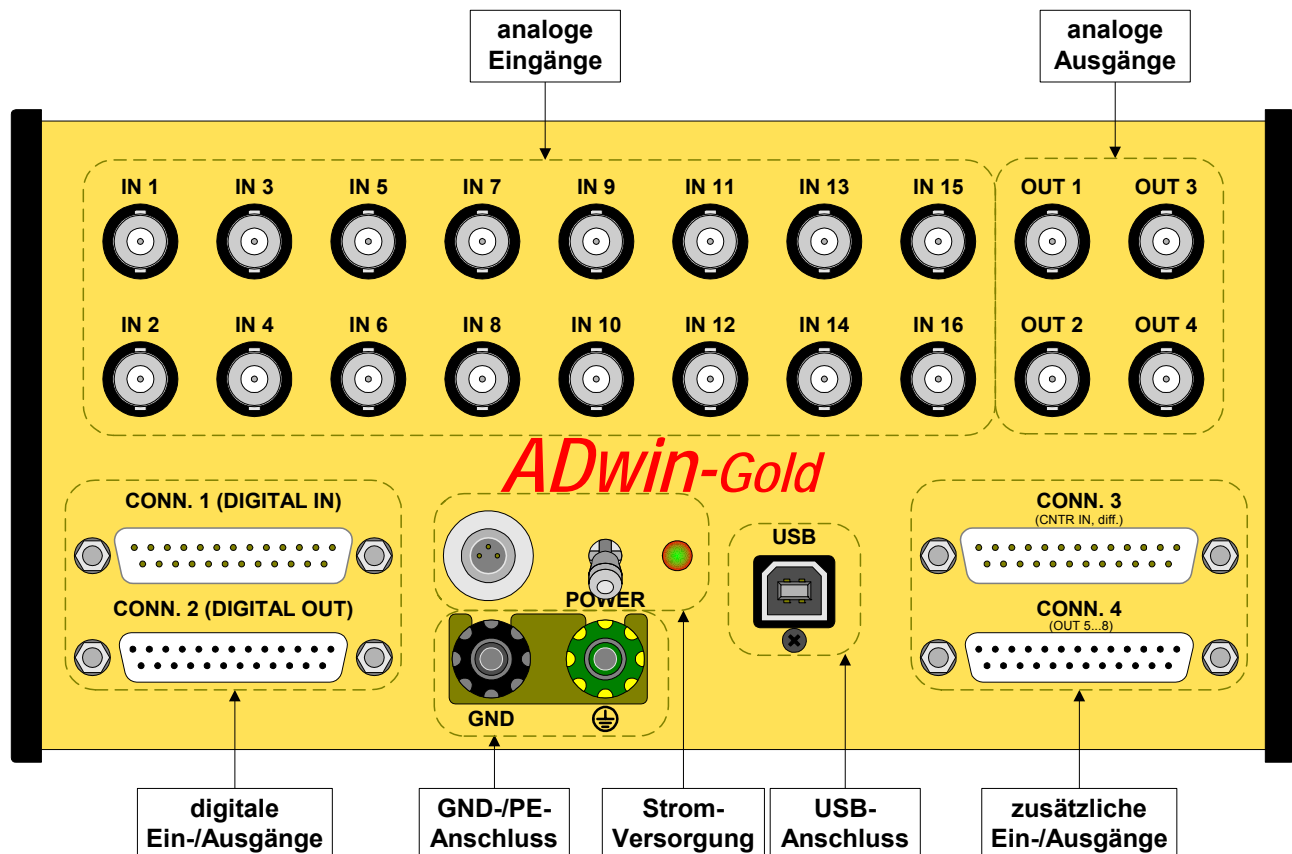
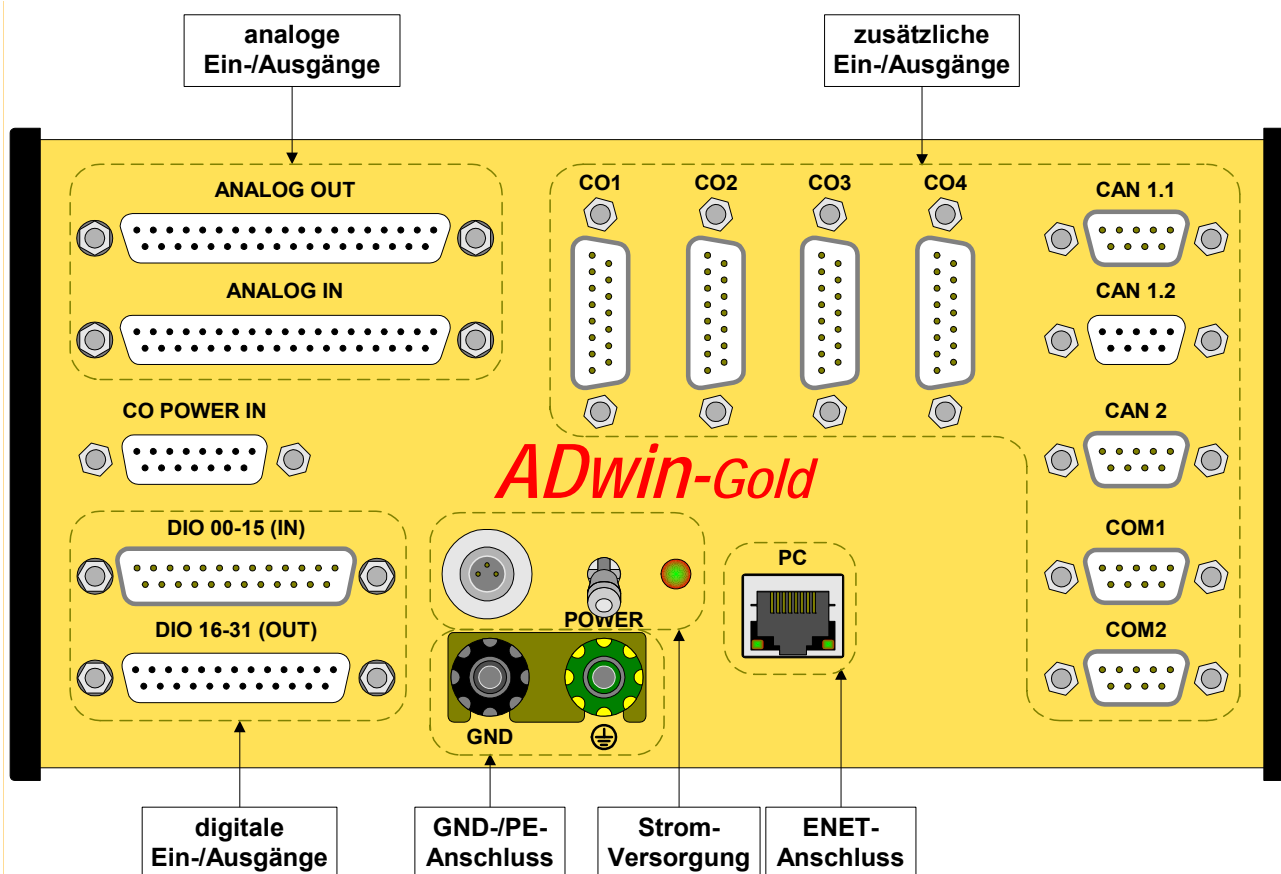


Abb. 4 – Übersichtsbild **ADwin-Gold** (USB-Version)

Abb. 5 – Übersichtsbild **ADwin-Gold-D** (ENET-Version)

5.1 Analoge Ein- und Ausgänge

Für störungsfreien Betrieb sind **isolierte BNC-Stecker** erforderlich. Es besteht ansonsten die Gefahr von Schäden durch elektrostatische Entladungen und Kurzschlüssen an den Eingängen. Das gilt vor allem bei Verwendung von nicht isolierten BNC-T-Stücken.

Das **ADwin-Gold**-Gerät **muss geerdet werden**, um Messungen störungsfrei durchführen zu können. Verbinden Sie dazu die GND-Buchse über ein impedanzarmes Masseband mit dem zentralen Erdungspunkt Ihrer Anlage.

Die Spannungsversorgung vom Power-Adapter am PC verbindet auch die Erdung des **ADwin-Gold**-Systems mit der Erdung des PC. Wenn Sie den PC und das System nicht am selben Ort betreiben, können **unterschiedliche Massepotenziale** am **ADwin-Gold** und am Messobjekt bzw. den Messleitungen Störungen verursachen. Vermeiden Sie solche Einflüsse, indem Sie ein externes Netzteil benutzen.

Neben der Beschreibung der Ein- und Ausgänge finden Sie nachfolgend Hinweise zur Umrechnung zwischen Digits und analogen Spannungswerten und zur Eingangsbeschaltung der analogen Eingänge.

Die Pin-Belegung der Analog-Kanäle für die Bestelloption Gold-D ist in Abb. 6 dargestellt.



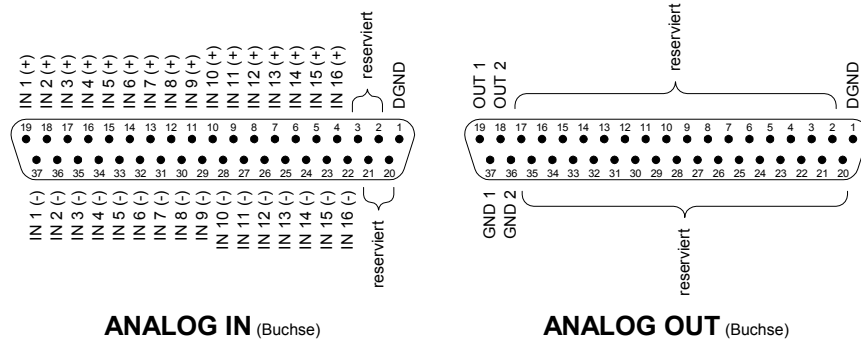


Abb. 6 – Pin-Belegung der Analogkanäle für Option Gold-D

5.1.1 Analoge Eingänge

Das System hat 16 analoge Eingänge IN1 ... IN16. Die Eingänge mit ungeraden Zahlen (1, 3, ... 15) sind dem Multiplexer 1, diejenigen mit geraden Zahlen (2, 4, ... 16) sind dem Multiplexer 2 zugeordnet. Der Ausgang jedes Multiplexers ist mit je einem 14 Bit-ADC und einem 16 Bit-ADC verbunden (siehe auch Funktionsschema des ADwin-Gold, Seite 4).

Die analogen Eingänge sind differentiell. Für jeden Messkanal sind je ein Plus- und ein Minuseingang vorhanden, zwischen denen die Spannungsdifferenz gemessen wird (jedoch nicht potenzialfrei). Für jeden Kanal müssen Plus- und Minuseingang angeschlossen werden.

Die Eingänge sind mit männlichen BNC-Buchsen bestückt, die in zwei Reihen angeordnet sind; bei Gold-D sind die Eingänge auf die Sub-D-Buchse ANALOG IN gelegt. Bei den BNC-Buchsen ist der Innenleiter der Plus-Eingang, der Außenleiter der Minus-Eingang.

Beachten Sie, dass zusätzlich zu Plus- und Minuseingang des Kanals immer eine Masseverbindung zwischen dem System (GND-Buchse) und der Signalquelle bestehen muss.

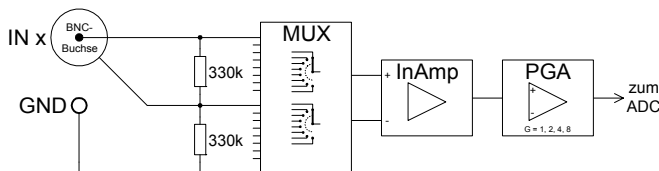


Abb. 7 – Eingangsbeschaltung eines analogen Eingangs

Sie können die Signale an den Multiplexer-Ausgängen wahlweise mit einem 14Bit oder 16Bit Analog-Digital-Wandler (ADC) konvertieren (siehe Abb. 2 „Funktionsschema des ADwin-Gold“). Sie messen mit

- dem 14Bit-ADC sehr schnell (max. 0,5µs, Auflösung 1,221mV)
- dem 16Bit-ADC sehr genau (max. 5µs, Auflösung 305µV).

Die Befehle `ADC ()` für den 16Bit-ADC und `ADC12 ()` für den 14Bit-ADC führen mit einem der ADC eine **komplette Messung** auf einem analogen Eingang durch. So berücksichtigen diese Befehle z.B. die Einschwingzeit des Multiplexers und stellen einwandfreie Messungen sicher (siehe auch **ADbasic**-Handbuch).

Multiplexer

Differentiell



16Bit- und 14Bit-Messung

ADC-Befehl





Achten Sie auf einen möglichst geringen Innenwiderstand Ihrer Spannungsquelle (für die Eingangssignale), denn dieser kann die Messgenauigkeit beeinflussen. Falls dies nicht möglich ist:

- Abhängig vom Ausgangswiderstand wird ein linearer Messfehler erzeugt.
Sie können dies ausgleichen, indem Sie den Messwert mit einem entsprechenden Faktor multiplizieren und dadurch sozusagen „nachkalibrieren“.
- Ab etwa $3\text{k}\Omega$ aufwärts verlängert sich zusätzlich die Einschwingzeit des Multiplexers.
Die in den Standard-Befehlen `ADC` und `ADC12` definierte Einschwingzeit ist dann zu kurz, so dass zu früh ungenaue Werte abgerufen werden. Verwenden Sie für diesen Fall die in Kapitel 5.3.1 beschriebenen Befehle.



5.1.2 Analoge Ausgänge

Das System hat 2 analoge Ausgänge (OUT1, OUT2) mit BNC-Buchsen; bei Gold-D liegen die Ausgänge auf der Sub-D-Buchse ANALOG OUT (siehe Abb. 6). Den Ausgängen ist je ein eigener Digital-Analog-Wandler (DAC) zugeordnet.

DAC-Befehl

Der Standardbefehl `DAC(Nummer, Wert)` prüft jeden Wert auf die Über- und Unterschreitung des 16Bit-Wertebereiches (0...65535). Liegt der Wert innerhalb dieses Bereiches, wird der angegebene Wert auf dem Ausgang *Nummer* ausgegeben. Liegt er außerhalb, wird der Maximal- bzw. Minimalwert ausgegeben (siehe auch **ADbasic**-Handbuch).



5.1.3 Berechnungsgrundlagen

Spannungsbereich

Das **ADwin-Gold**-System arbeitet bei den analogen Ein- und Ausgängen mit einem Spannungsbereich von -10V bis $+10\text{V}$ (= bipolar 10V).

Zuordnung von Digits zu Spannung

Die 65536 (2^{16}) Digits sind den jeweiligen Spannungsbereichen der ADC und DAC so zugeordnet, dass

- 0 (Null) Digit der maximalen negativen Spannung und
- 65535 Digit der maximalen positiven Spannung

entspricht.



Der Wert für 65536 Digit, genau 10 Volt, liegt gerade *außerhalb* des Messbereichs, womit sich für die 16Bit-Wandlung ein maximaler Spannungswert von 9,999695 Volt und für die 14Bit-Wandlung von 9,998779 Volt ergibt.

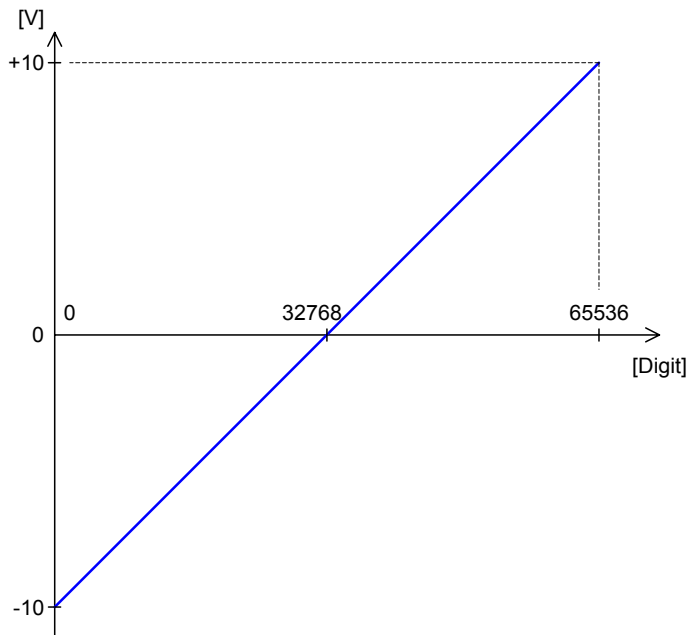


Abb. 8 – Nullpunktverschiebung bei Standardeinstellung bipolar 10 Volt

Die bipolare Einstellung führt zu einer Nullpunktverschiebung, die im folgenden auch als Offset U_{OFF} bezeichnet wird.

Beim Spannungsbereich $-10\text{V} \dots +10\text{V}$ gilt:

$$U_{\text{OFF}} = -10\text{V}$$

Das **ADwin-Gold-System** besitzt einen programmierbaren Verstärker (PGA), mit dem Sie die Eingangsspannung um die Faktoren 1, 2, 4, und 8 verstärken können. Gleichzeitig verkleinert sich damit der Messbereich um den jeweiligen Verstärkungsfaktor k_v (siehe Anhang „Technische Daten“).

Beachten Sie bei Anwendungen mit $k_v > 1$, dass auch die Störsignale entsprechend mit verstärkt werden.

Die Quantisierungsstufe U_{LSB} ist die kleinste digital darstellbare Spannungsdifferenz und ist gleich der Spannung des niederwertigsten Bit (Least Significant Bit, LSB). Sie ist für die beiden ADC unterschiedlich:

- 16Bit-ADC: $U_{\text{LSB}} = 20\text{V} / 2^{16} = 305,175\mu\text{V}$
- 14Bit-ADC: $U_{\text{LSB}} = 20\text{V} / 2^{14} = 1220,7\mu\text{V}$

Der gemessene 16Bit-Wert des ADC wird im unteren Wort der Speicherzelle zurückgeliefert. Dort muss sich auch ein auszugebender DAC-Wert befinden.

Bit-Nr.	31...16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
32 Bit-Speicher	0	16 Bit-Wert des 16Bit-ADC / DAC im unteren Wort															
	0	14Bit-Wert des 14Bit-ADC im unteren Wort														0	0

Tab. 1 – Ablage der ADC/DAC-Bits im Speicher

Um Messwerte des 14Bit ADC und des 16Bit ADC einfach vergleichen zu können, wird der gewandelte Wert beim 14Bit ADC linksbündig in das untere Wort der Speicherzelle geschrieben. Die untersten 2 Bits sind daher stets 0 (Null).

Die 16384 Digits des 14Bit ADC werden auf die 65536 Digits des 16Bit ADC abgebildet. Damit entsprechen je 4 Digits des 16Bit ADC einem Digit des 14Bit ADC.

Die folgenden Gleichungen gelten somit für beide ADC-Typen:

Nullpunktverschiebung
 U_{OFF}

Verstärkungsfaktor k_v

Quantisierungsstufe
 U_{LSB}

DAC

Umrechnung Digit ↔ Spannung

Für einen DAC gilt:

$$U_{OUT} = \text{Digits} \cdot U_{LSB} + U_{OFF}$$

$$\text{Digits} = \frac{U_{OUT} - U_{OFF}}{U_{LSB}}$$

ADC

Für einen ADC (14Bit und 16Bit) gilt:

$$\text{Digits} = \frac{k_v \cdot U_{IN} - U_{OFF}}{U_{LSB}}$$

$$U_{IN} = \frac{\text{Digits} \cdot U_{LSB} + U_{OFF}}{k_v}$$

Toleranzbereiche

Geringe Abweichungen zu den rechnerischen Werten können innerhalb der Toleranzbereiche einzelner Bauteile liegen. Es gibt zwei charakteristische Abweichungsarten, die in diesem Handbuch angegeben sind (in LSB):

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.

5.2 Digitale Ein- und Ausgänge

Digitale Ein- / Ausgänge

Auf zwei 25-poligen Sub-D-Buchsen (DIO 00...DIO 31) stehen 32 digitale Ein- oder Ausgänge zur Verfügung. Sie sind in Gruppen zu jeweils 8 als Ein- oder Ausgang programmierbar.



Die digitalen Eingänge sind TTL-kompatibel und gegen Überspannung nicht geschützt.

Beschalten Sie keine freien Anschlüsse, die als „reserviert“ gekennzeichnet sind. Diese sind Änderungen oder Erweiterungen vorbehalten; Nichtbeachten kann das System beschädigen.

Trigger-Eingang (EVENT)



Das **ADwin-Gold**-System besitzt einen externen Trigger-Eingang (EVENT). Ein externes Signal (Trigger) mit steigender Flanke an diesem Eingang kann Prozesse aufrufen, die sofort und vollständig abgearbeitet werden (siehe auch **ADbasic**-Handbuch, Kapitel: „Programmaufbau“).

Einschaltkonfiguration

Nach dem Einschalten des Gerätes sind alle 4 Anschlussgruppen als Eingang konfiguriert.

CONF_DIO(12)



Der Befehl `CONF_DIO(12)` konfiguriert DIO 15:00 als digitale Eingänge und DIO 31:16 als digitale Ausgänge (siehe Abb. 9).

Nur in dieser Konfiguration können Sie mit den Befehlen `DIGIN`, `DIGIN_WORD`, `DIGOUT_WORD`, `SET_DIGOUT`, `CLEAR_DIGOUT` in vollem Umfang auf die Ein- und Ausgänge zugreifen.

Über die Programmierung bei anderen Konfigurationen informiert Sie das folgende Kapitel 5.3 „Zeitkritische Aufgaben“ (siehe auch **ADbasic**-Handbuch und Tutorial).

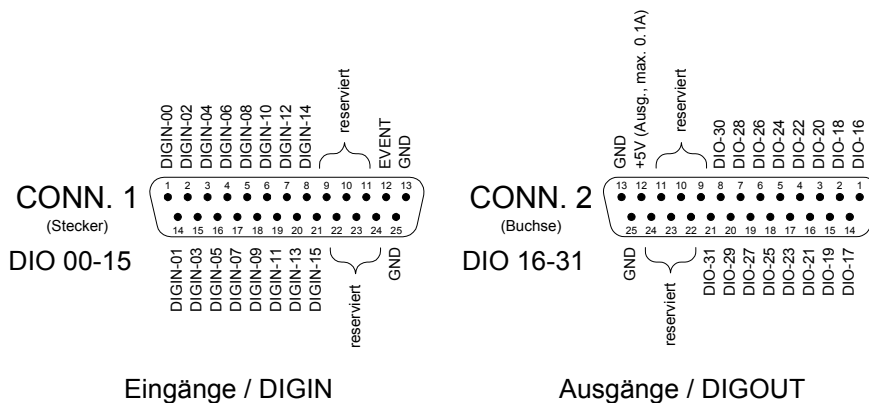


Abb. 9 – Pin-Belegung bei der Konfiguration mit `CONF_DIO(12)`

5.3 Zeitkritische Aufgaben

Für extrem zeitkritische Aufgaben können Sie Befehle einsetzen, mit denen Sie direkt auf die **Steuer- und Datenregister der ADC und DAC** zugreifen (siehe **ADbasic**-Handbuch). Diese Register liegen im Speicheradress-Bereich des ADSP (memory mapped). Die Befehle ermöglichen auch eine Optimierung der Programmstruktur (s.u.).

Im Gegensatz zu den Standardbefehlen `ADC()`, `ADC12()` und `DAC()` besitzen die Befehle für den Direktzugriff **keine Prüfroutinen**. Vor deren Benutzung sollten Sie sich deshalb genaue Kenntnisse über Zeitabläufe, Programmstrukturen und Funktionsabläufe in einem ADC aneignen.

5.3.1 Analoge Ein- und Ausgänge

Die Standardbefehle `ADC()` und `ADC12()` bestehen aus einer Sequenz von mehreren Befehlen (im folgenden dargestellt) und benötigen eine werkseitig festgelegte Zeit zur Ausführung. Die Ausführungszeit wird vor allem durch die Einschwingzeit des Multiplexers und die Wandlungszeit bestimmt.

```
SET_MUX( )
...
START_CONV( )
WAIT_EOC( )
READ_ADC( )
```

'Einschwingzeit abwarten
'Auf Wandlungsende warten
'bzw. READ_ADC12() bei ADC12()

Sie können die im Standardbefehl enthaltenen Wartezeiten durch Verwendung der Einzelbefehle für andere Zwecke nutzen (oder ggf. auch verlängern). Bei geschicktem Einsatz der Befehle können Sie dadurch schnellere Messvorgänge realisieren.

Es ist wichtig, den `START_CONV()` Befehl in ausreichendem Zeitabstand vom `SET_MUX()` Befehl zu setzen, um die Einschwingzeit des Multiplexers zu berücksichtigen (siehe auch **ADbasic**-Handbuch, Befehlsreferenz).

Nutzen Sie die entstehenden Wartezeiten, z.B. für Rechenoperationen, und sparen Sie somit Rechenzeit ein:



ADC() und ADC12()

Programmstruktur



- Einschwingzeit des Multiplexers: Diese beträgt beim maximalen Spannungssprung von 20 Volt für den 16Bit ADC höchstens 6,5µs und für den 14Bit ADC höchstens 2,5µs.
- Wandlungszeit der ADC: Sie beträgt beim 14Bit ADC 0,5µs und beim 16Bit ADC 5µs.

ADC**DAC****Direkter Registerzugriff**

Eine Messung kann sehr schnell ausgeführt werden, wenn Sie direkt auf die Steuer- und Datenregister der ADC zugreifen.

Ist bei den analogen Ausgängen sichergestellt, dass die Werte innerhalb der Bereichsgrenzen liegen, können Sie mit direktem Zugriff auf die Hardware-Register sehr schnell ein oder mehrere DAC-Register beschreiben und synchron die Ausgabe aktivieren (siehe **ADbasic**-Handbuch).

Die Hardware-Adressen für den direkten Zugriff auf die Steuer- und Datenregister sind auf den folgenden Seiten dargestellt.

Adresse [HEX]	Funktion	Bit													Kommentar
		31:16	15:10	9	8	7	6	5	4	3	2	1	0		
20400000	MUX 1 setzen: Kanäle 1, 3, 5, ..., 15	-	-	-	-	-	-	-	-	-	n	n	n	„nnn“ binär = 0...7 dezimal, gewählter Kanal = nnn + 1	
	MUX 2 setzen: Kanäle 2, 4, 6, ..., 16	-	-	-	-	-	-	n	n	n	-	-	-	„nnn“ binär = 0...7 dezimal, gewählter Kanal = 2 (nnn + 1)	
	Verstärkung PGA 1	-	-	-	-	g	g	-	-	-	-	-	-	„gg“ binär = 0...3 dezimal, gewählte Verstärkung = 2 ^{gg}	
	Verstärkung PGA 2	-	-		g	g	-	-	-	-	-	-	-	-	
20400010	Konvertierung starten: ADC 1 (16Bit)	-	-	-	-	-	-	-	-	-	1	-	s	s = 0 : Konvertierung starten s = 1 : kein Einfluss	
	Konvertierung starten: ADC 2 (16Bit)	-	-	-	-	-	-	-	-	-	1	s	-		
	Konvertierung starten: ADC 1 (14Bit)	-	-	-	-	-	-	-	-	s	1	-	-		
	Konvertierung starten: ADC 2 (14Bit)	-	-	-	-	-	-	-	s	-	1	-	-		
20400020	EOC-Status: ADC 1 (16Bit)	-	-	-	-	-	-	-	-	-	-	-	e	e = 0 : Konvertierung beendet e = 1 : Konvertierung läuft	
	EOC-Status: ADC 2 (16Bit)	-	-	-	-	-	-	-	-	-	-	e	-		
	EOC-Status: ADC 1 (14Bit)	-	-	-	-	-	-	-	-	e	-	-	-		
	EOC-Status: ADC 2 (14Bit)	-	-	-	-	-	-	-	-	e	-	-	-		
20400030	Register auslesen: ADC 1 (16Bit)	-	x	x	x	x	x	x	x	x	x	x	x	x : Ergebnis der Konvertierung	
20400040	Register auslesen: ADC 2 (16Bit)	-	x	x	x	x	x	x	x	x	x	x	x		
20400130	Register auslesen: ADC 1 (14Bit)	-	x	x	x	x	x	x	x	x	x	0	0		
20400140	Register auslesen: ADC 2 (14Bit)	-	x	x	x	x	x	x	x	x	x	0	0		
20400100	Register auslesen und Konvertierg. starten: ADC 1 (16Bit)	-	x	x	x	x	x	x	x	x	x	x	x		
20400110	Register auslesen und Konvertierg. starten: ADC 2 (16Bit)	-	x	x	x	x	x	x	x	x	x	x	x		
20400120	Register auslesen und Konvertierg. starten: ADC 1 (14Bit)	-	x	x	x	x	x	x	x	x	x	0	0		
204001D0	Register auslesen und Konvertierg. starten: ADC 2 (14Bit)	-	x	x	x	x	x	x	x	x	x	0	0		

Tab. 2 – Hardware-Adressen der Steuer- und Datenregister für die **ADC**

Adresse [HEX]	Funktion	Bit													Kommentar
		31:16	15:10	9	8	7	6	5	4	3	2	1	0		
20400010	Konvertierung starten: Alle DAC synchron	-	-	-	-	-	-	-	1	1	s	1	1	s = 0 : Konvertierung starten s = 1 : kein Einfluss	
20400050	Register nur beschreiben: DAC 1	-	x	x	x	x	x	x	x	x	x	x	x		
20400060	Register nur beschreiben: DAC 2	-	x	x	x	x	x	x	x	x	x	x	x		
20400070	Register nur beschreiben: DAC 3 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400080	Register nur beschreiben: DAC 4 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400090	Register nur beschreiben: DAC 5 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
204000A0	Register nur beschreiben: DAC 6 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400190	Register nur beschreiben: DAC 7 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
204001A0	Register nur beschreiben: DAC 8 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x	x : zu konvertierender Digital- wert	
20400200	Register beschreiben und Konvertierung sofort starten: DAC 1	-	x	x	x	x	x	x	x	x	x	x	x		
20400210	Register beschreiben und Konvertierung sofort starten: DAC 2	-	x	x	x	x	x	x	x	x	x	x	x		
20400220	Register beschreiben und Konvertierung sofort starten: DAC 3 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400230	Register beschreiben und Konvertierung sofort starten: DAC 4 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400240	Register beschreiben und Konvertierung sofort starten: DAC 5 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400250	Register beschreiben und Konvertierung sofort starten: DAC 6 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400260	Register beschreiben und Konvertierung sofort starten: DAC 7 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		
20400270	Register beschreiben und Konvertierung sofort starten: DAC 8 (<i>Gold-DA</i>)	-	x	x	x	x	x	x	x	x	x	x	x		

Tab. 3 – Hardware-Adressen der Steuer- und Datenregister für die **DAC**



5.3.2 Digitale Ein- und Ausgänge

Nach dem Einschalten des Gerätes sind alle 4 Anschlussgruppen als Eingang konfiguriert; dies entspricht dem Befehl `CONF_DIO(0)`. Die folgende Tabelle zeigt, wie Ein- und Ausgänge (IN, OUT) konfiguriert werden, wenn Sie den Wert der ersten Spalte als Befehlsargument verwenden.

CONF_DIO()	DIO31:24	DIO23:16	DIO15:08	DIO07:00
0	IN	IN	IN	IN
1	IN	IN	IN	OUT
2	IN	IN	OUT	IN
3	IN	IN	OUT	OUT
4	IN	OUT	IN	IN
5	IN	OUT	IN	OUT
6	IN	OUT	OUT	IN
7	IN	OUT	OUT	OUT
8	OUT	IN	IN	IN
9	OUT	IN	IN	OUT
19	OUT	IN	OUT	IN
11	OUT	IN	OUT	OUT
12	OUT	OUT	IN	IN
13	OUT	OUT	IN	OUT
14	OUT	OUT	OUT	IN
15	OUT	OUT	OUT	OUT
Anwendbare Befehle:	DIGOUT_WORD, CLEAR_DIGOUT, SET_DIGOUT		DIGIN_WORD, DIGIN	
Befehl ist anwendbar für den Anschluss DIO _n , bei	Einstellung „OUT“		Einstellung „IN“ Bei Einstellung „OUT“ wird der Register-Inhalt dieses Bytes zurückgelesen	

Tab. 4 – Übersicht der Konfigurationen mit CONF_DIO

Beachten Sie folgende Einschränkung:

Nur bei Konfiguration der Ein-/Ausgänge mit dem Befehl `CONF_DIO(12)` (entsprechende Pinbelegung siehe Seite 15) können Sie mit den folgenden Befehlen im vollem Umfang auf die Kanäle zugreifen

DIGOUT_WORD, SET_DIGOUT, CLEAR_DIGOUT, DIGIN_WORD, DIGIN.

Für jede andere Konfiguration müssen Sie das entsprechende Hardware-Register auslesen oder beschreiben (siehe Befehle PEEK und POKE im Handbuch **ADbasic**).

Adresse [HEX]	Funktion	Bit													Kommentar
		31:16	15:10	9	8	7	6	5	4	3	2	1	0		
204001E0	DIO07:00 konfigurieren	0	0	0	0	0	0	0	0	-	-	-	c	c = 0 : Eingänge c = 1 : Ausgänge	
	DIO15:08 konfigurieren	0	0	0	0	0	0	0	0	-	-	c	-		
	DIO23:16 konfigurieren	0	0	0	0	0	0	0	0	-	c	-	-		
	DIO31:24 konfigurieren	0	0	0	0	0	0	0	0	c	-	-	-		
204000B0	Eingangs-Register DIO15:00	-	x	x	x	x	x	x	x	x	x	x	x	x : eingelesener Digitalwert	
204001B0	Eingangs-Register DIO31:16	-	x	x	x	x	x	x	x	x	x	x	x		
204001C0	Ausgangs-Register DIO15:00	-	x	x	x	x	x	x	x	x	x	x	x	x : auszugebender Digitalwert	
204000C0	Ausgangs-Register DIO31:16	-	x	x	x	x	x	x	x	x	x	x	x		

Tab. 5 – Hardware-Adressen der digitalen Ein- und Ausgänge

6 Kalibrierung

6.1 Allgemeine Hinweise

Die 2 Digital/Analog-Wandler (DAC; optional 8) und die 4 Analog/Digital-Wandler (ADC) Ihres **ADwin-Gold**-Systems sind bei der Auslieferung werkseitig kalibriert. Entsprechend den Vorschriften zur Einhaltung der Messgenauigkeit für Ihr Anwendungsgebiet sind die Geräte in regelmäßigen Abständen zu kalibrieren.

Sie führen die Kalibrierung mit dem Programm <GoldCalib.exe> durch; der Pfad bei Standardinstallation ist <C:\ADwin\Tools\ADwin-Gold>.

Zur Kalibrierung benötigen Sie folgende Hilfsmittel:

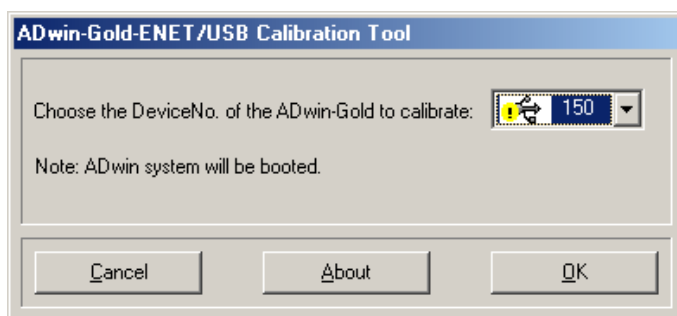
- Ein Digital-Multimeter (DMM) mit einer Messgenauigkeit von
 - 30µV bei 16Bit -Wandlern
 - 120µV bei 14Bit -Wandlern
- Eine stabile Referenz-Gleichspannungsquelle mit einer Einstellgenauigkeit von
 - 30µV bei 16Bit -Wandlern
 - 120µV bei 14Bit -Wandlern
- Verbindungskabel von den Ein/Ausgängen zur Referenzspannungsquelle und zum Messgerät (bevorzugt BNC-Kabel).

6.2 Kalibrierung durchführen

Verbinden Sie Ihr **ADwin-Gold**-Gerät mit dem PC und konfigurieren Sie es mit dem Programm <ADconfig.exe>.

Die Kalibrierung muss bei Betriebstemperatur des **ADwin-Gold** erfolgen. Bei einer Einschalttemperatur des Gerätes von ca. 20...25 Grad Celsius (Raumtemperatur) ist die Betriebstemperatur etwa 30 Minuten nach dem Einschalten erreicht.

Starten Sie das **Kalibrierprogramm** <GoldCalib.exe>. Es erscheint das Fenster „ADwin-Gold-ENET/USB Calibration Tool“.



Wählen Sie die Device-Nummer des zu kalibrierenden Geräts und bestätigen Sie durch Drücken von „OK“.

Sie erhalten eine Warnung, wenn Sie kein **ADwin-Gold** gewählt haben oder eines mit einer älteren Firmware-Version. Sie können die Warnung mit „Yes“ übergehen oder mit „No“ zum vorigen Fenster zurückkehren.

Es erscheint das **Übersichtsfenster**. In der Kopfzeile wird die von Ihnen gewählte Device-Nummer angezeigt.

Hilfsmittel



Schritt 1

Das obere Feld zeigt die aktuellen Messwerte an den Eingängen IN1 und IN2, jeweils gemessen mit dem 16Bit- und dem 14Bit-ADC.

Wählen Sie unteren Feld links den zu kalibrierenden DAC, rechts den zu kalibrierenden ADC. Der Messwert am gewählten ADC wird oben hervorgehoben. In den Eingabefeldern darunter sehen Sie die jeweils gültige Kalibriereinstellung für Offset und Gain der DAC und ADC; Sie können dort direkt Werte eingeben. Mit den Schaltflächen „Calibrate DAC“ oder „Calibrate ADC“ starten Sie die Kalibrierung des gewählten Wandler.

Im Eingabefeld „Test Output“ können Sie einen Spannungswert eingeben, der automatisch auf den oben gewählten Wandler/Ausgang gelegt wird.

Jede Ihrer Eingaben wird sofort in das **ADwin-Gold**-System übertragen. Wenn Sie das Programm mit „Exit“ verlassen, bleiben die neuen Einstellungen erhalten. Mit der Schaltfläche „Undo&Exit“ machen Sie alle Eingaben rückgängig und verlassen das Kalibrierprogramm (d.h. die ursprünglichen Einstellungen werden in das **ADwin-Gold**-System übertragen).

„Diagram“ zeigt Ihnen in einer Grafik die Genauigkeit der aktuellen Kalibriereinstellung. Ein Protokoll der eingestellten Werte drucken Sie mit der Schaltfläche „Print Calibration“.

Kalibrieren Sie die Wandler in beliebiger Reihenfolge (nur mit Referenz-Spannungsquelle). Die Kalibrierung eines Wandlers wird jeweils in 3 Stufen ausgeführt; Sie können zwischen den Fenstern der Stufen durch Vorwärts-/ Rückwärts-Schaltflächen hin- und herschalten.

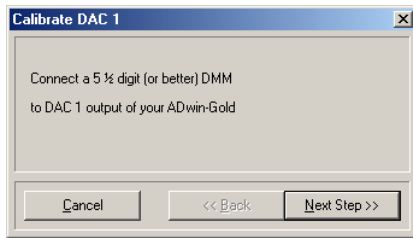
Ohne Referenz-Spannungsquelle ist die Kalibrierung möglich, aber ungenauer. Kalibrieren Sie dabei zuerst die DAC, dann erst die ADC.

Die 3 Stufen zum **Kalibrieren eines Wandlers** sind nachfolgend beschrieben, jeweils in der linken Spalte für einen DAC, in der rechten für einen ADC.

1. Externes Hilfsgerät (DMM / Spannungsquelle) anschließen:
Wählen Sie zur Kalibrierung eines Wandlers den gewünschten Wandler und dann die Schaltfläche „Calibrate ...“; es erscheint das erste Fenster:



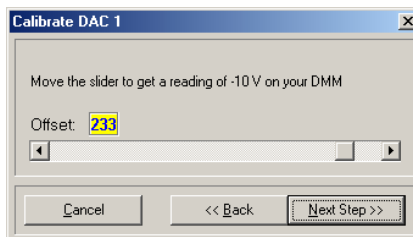
Schritt 2



Schließen Sie ein DMM am gewählten Ausgang an.

Beachten Sie bitte Abb. 4 „Übersichtsbild ADwin-Gold (USB-Version)“. Wählen Sie „Next Step >>“.

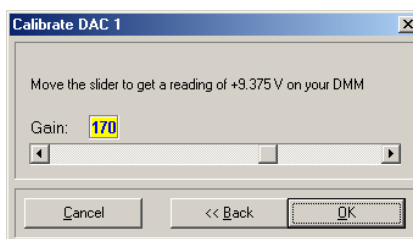
2. Offset einstellen



Verstellen Sie am Rollbalken den Offset-Wert so, dass Ihr Digital-Multimeter -10V anzeigt.

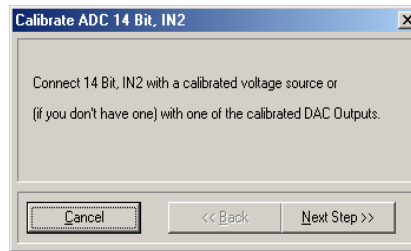
Wählen Sie „Next Step >>“.

3. Gain einstellen

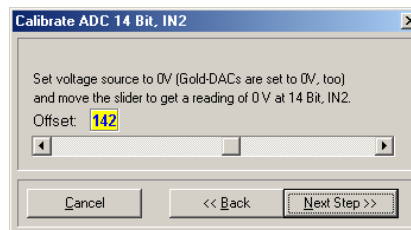


Verstellen Sie am Rollbalken den Offset-Wert so, dass Ihr Digital-Multimeter 9,375V anzeigt.

Die Kalibrierung für diesen Wandler ist beendet. Wählen Sie „OK“. Wiederholen Sie Schritt 2 ggf. für die anderen Wandler.

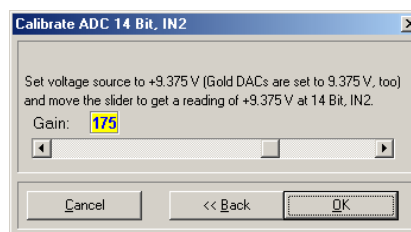


Schließen Sie die Spannungsquelle (oder einen kalibrierten DAC-Ausgang) am gewählten Eingang an.



Stellen Sie an der Spannungsquelle den Sollwert 0V ein. Die Einstellung des ADC auf diesen Wert erfolgt automatisch.

Stellen Sie am Rollbalken den Offset-Wert so ein, dass der Sollwert am ADC im Übersichtsfenster angezeigt wird.



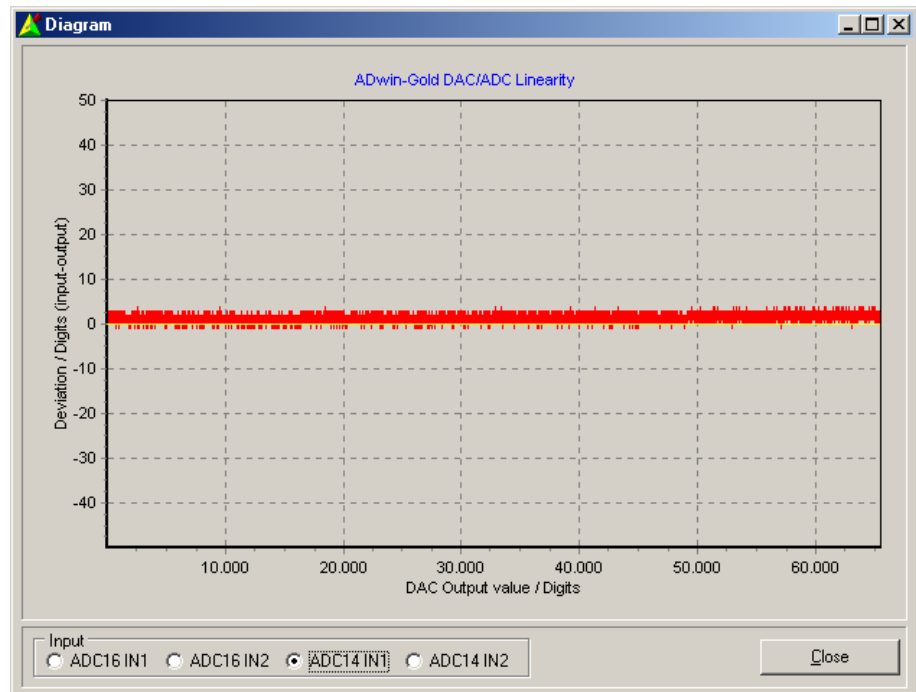
Stellen Sie an der Spannungsquelle den Sollwert 9,375V ein. Die Einstellung des ADC auf diesen Wert erfolgt automatisch.

Stellen Sie am Rollbalken den Offset-Wert so ein, dass der Sollwert am ADC im Übersichtsfenster angezeigt wird.



Schritt 3

Die **Genauigkeit** der eingestellten Kalibrierung können Sie anhand eines Diagramms prüfen (Schaltfläche „Diagram“ im Übersichtsfenster). Verbinden Sie zunächst 2 beliebige Ausgänge mit den Eingängen IN1 und IN2. Wählen Sie im Diagramm einen der Eingänge und den zugehörigen Wandler aus.



Das Programm gibt die Werte 0...65535 Digits auf alle DAC aus, vergleicht sie mit den Messwerten am gewählten Eingang und stellt die Abweichung in einer Kurve dar.

Die Abweichung sollte kleiner als 5 Digits sein.

Mit Close kehren Sie zum Übersichtsfenster zurück.

Schritt 4

Mit der Schaltfläche „Print Calibration“ können Sie ein **Protokoll** der eingestellten Kalibrier-Daten ausdrucken.

In dem geöffneten Fenster können Sie verschiedene Informationen eingeben, die in Ihrem Ausdruck erscheinen (für eine spätere Zuordnung des Protokolls). Mit der Schaltfläche „Print“ starten Sie den Druckvorgang; das Programm kehrt automatisch zum Übersichtsfenster zurück.

Im Protokoll-Ausdruck sind die Kalibriereinstellungen aller Ein- und Ausgänge für Gain und Offset sowie das Druckdatum enthalten.

Schritt 5

Die Kalibrierung ist beendet.

7 DA-Erweiterung

Mit der DA-Erweiterung erhalten Sie **6 zusätzliche analoge Ausgänge** mit einer Auflösung von 16 Bit (und je einem DAC).

In der Standard-Version führen 2 dieser Ausgänge von DAC 3 und DAC 4 auf die BNC-Buchsen OUT 3 und OUT 4. Die weiteren 4 Ausgänge führen von DAC 5...DAC 8 auf die Pins 1...4 und 14...17 der 25-poligen Sub-D-Buchse CONN. 4 (siehe Abbildung).

Bei der Bestelloption Gold-D liegen alle zusätzlichen Ausgänge auf Pins der Sub-D-Buchse ANALOG OUT.

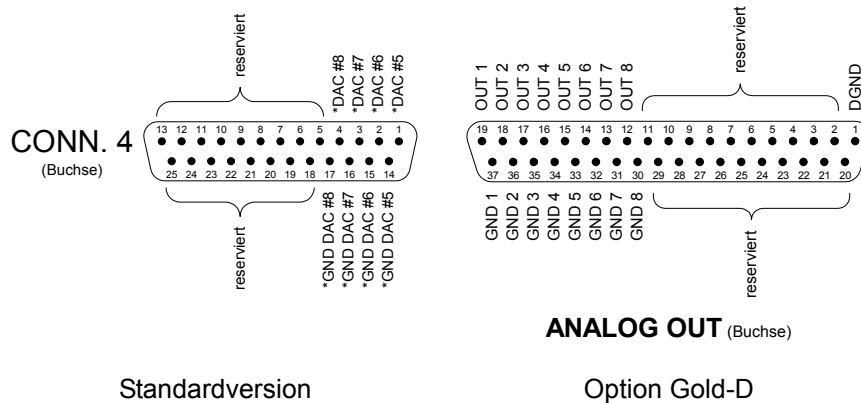


Abb. 10 – Pin-Belegung der DA-Erweiterung

Sie programmieren und kalibrieren die zusätzlichen DAC wie bei den DAC 1 und DAC 2 (siehe Kapitel 5.1, Kapitel 6 und das **ADbasic**-Handbuch).

Anschlüsse

Programmierung und Kalibrierung

8 CO1-Zählererweiterung

Die technischen Daten der CO1-Zählererweiterung sind im Anhang A.1 beschrieben.

8.1 Hardware

Die Zählererweiterung **CO1** (Bestelloption **Gold-CO1**) stellt 4 Stück 32 Bit Vor- / Rückwärtszähler mit Vierflankenauswertung zur Verfügung. Sie können die Zähler per Software sowohl einzeln als auch gemeinsam konfigurieren und auslesen (die Grafik zeigt den Aufbau eines einzelnen Zählers).

Die Zähler können intern oder extern getaktet werden und werden über zugeordnete Latches ausgelesen. Alle Zähler haben je ein Latch A und ein Latch B. Der Zählerstand kann mit Programmierbefehlen oder (bei entsprechender Einstellung) bei einem externen Signal an CLR/LATCH gelöscht oder in ein Latch übertragen werden.

Zähler

Latch A und B

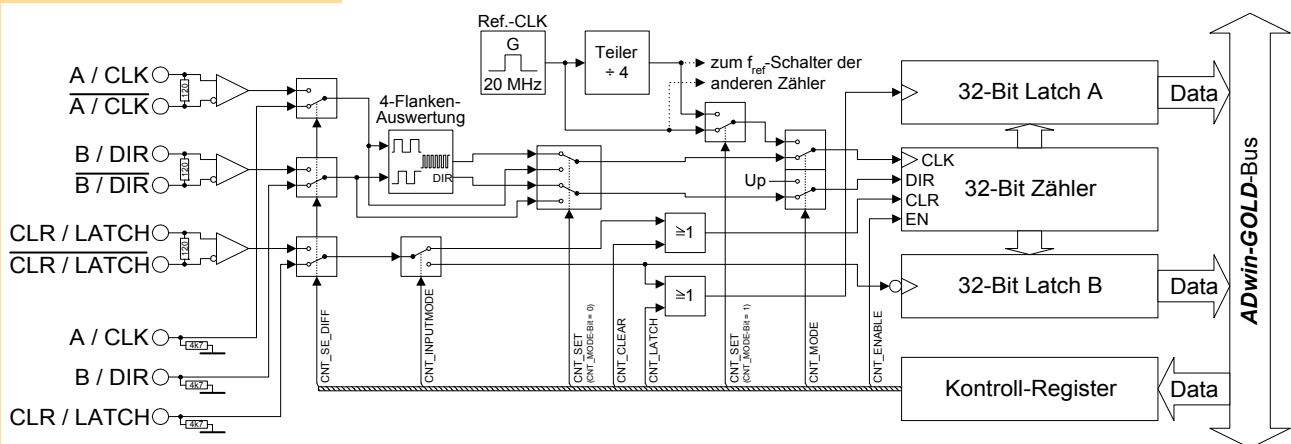


Abb. 11 – Schema der *Gold-CO1*-Zählererweiterung

Es gibt die Betriebsarten Ereigniszählung (externer Takt) und Pulsbreitenmessung (interner Takt); siehe auch Kapitel 8.3 / 8.4:

a) **Ereigniszählung:** Das In-/Dekrementieren des Zählers wird durch externe Rechtecksignale an den Eingängen A/CLK und B/DIR ausgelöst. Eine steigende Flanke an CLR/LATCH bewirkt, dass entweder der Zähler auf Null gesetzt (CLR) oder der Zählerstand ins Latch geschrieben wird (LATCH).

Es gibt die Modi:

1. **Takt und Richtung:** Eine steigende Flanke an CLK in- oder dekrementiert den Zählerstand um eins. Das Signal an DIR bestimmt die Zählrichtung (0 = Dekrementieren; 1 = Inkrementieren).
2. **Vierflankenauswertung:** Jede Flanke der (um 90 Grad) versetzten Signale an A/CLK und an B/DIR löst ein In-/Dekrementieren des Zählers aus. Die Zählrichtung ergibt sich aus der Reihenfolge der steigenden/fallenden Flanken dieser Signale. Dieser Modus wird besonders für Inkrementalgeber (Winkel-Encoder) eingesetzt.

b) **Pulsbreitenmessung:** Das In-/Dekrementieren des Zählers wird von einem internen Referenztaktgeber ausgelöst; es kann eine Signalfrequenz von 5MHz oder 20MHz verwendet werden. Ausgewertet wird das an CLR/LATCH anliegende Rechtecksignal: Mit jeder positiven Flanke wird der Zählerstand in Latch A geschrieben, mit jeder negativen in Latch B.

Externer Takteingang

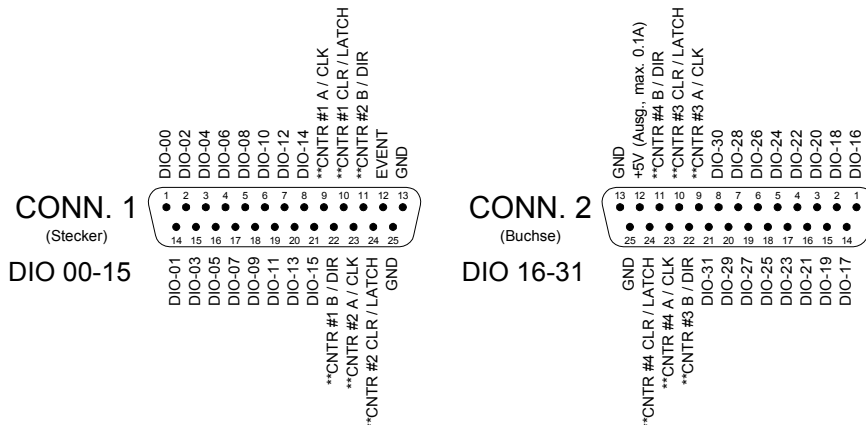
Interner Takteingang

Sie können damit berechnen:

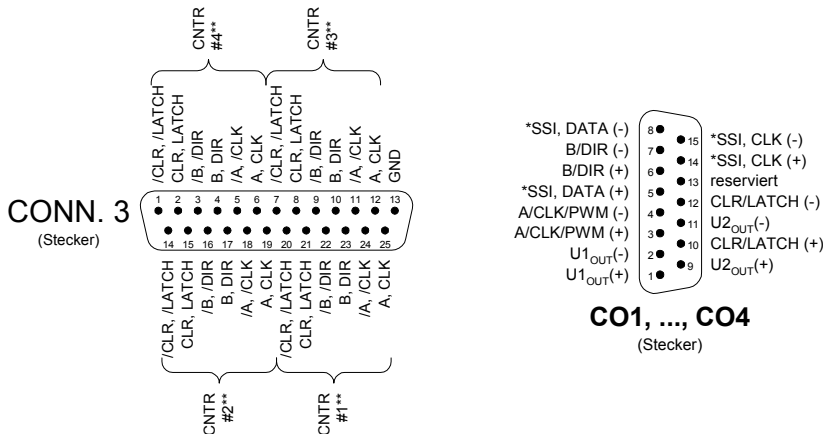
1. die **Periodendauer** des Eingangssignals an CLR/LATCH aus den Werten in Latch A oder Latch B.
2. die **Impulsbreite** und **Pausenzeit** aus den Werten in Latch A und Latch B.

Die Zähler werden mit **ADbasic**-Befehlen über ein Kontrollregister gesteuert (Befehle siehe Tabelle in Kapitel 8.2).

An den differentiellen Eingängen A/CLK, B/DIR und CLR/LATCH sind TTL-ähnliche Signale erforderlich. Einzelheiten und Grenzwerte finden Sie bei den Technischen Daten im Anhang.



Zählereingänge bei Betriebsart TTL für Gold / Gold-D
(single-ended; Betriebsart bei Rev. B2 nicht vorhanden)



nur Version Gold

nur Version Gold-D

Zählereingänge bei Betriebsart differentiell

Abb. 12 – Pin-Belegung der CO1-Erweiterung

Es ist in jedem Fall erforderlich, dass Sie die Zählereingänge mit dem Befehl `CNT_SE_DIFF` für die gewünschte Betriebsart einstellen. Dies geschieht paarweise, d. h. die Zähler 1 und 2 gemeinsam sowie die Zähler 3 und 4 gemeinsam (siehe auch **ADbasic**-Handbuch).

Bei Rev. B2 können Sie nur differentielle Eingänge einstellen, ab Rev. B3 alternativ auch TTL-Eingänge (single ended).

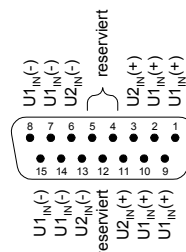
Obwohl alle Eingänge der CO1-Erweiterung einen Pull-down-Widerstand besitzen, können offene Eingänge (bei dem zugehörigen Zähler) vor allem in einer nicht störungsfreien Umgebung zu Fehlern führen. Wenn Sie einen Eingang eines Zählers nicht benutzen, legen Sie sicherheitshalber beide Leitun-



gen des (differentiellen) Eingangs auf ein definiertes Potenzial: Legen Sie den Pluseingang auf +5V und den Minuseingang auf GND.

Bei der Lieferoption Gold-D können Sie über den Anschluss **CO Power in** eine Spannung einspeisen, die an den Steckern CO1...CO4 zur Verfügung steht, z.B. für externe Inkrementalgeber.

Beachten Sie bitte: Die Minus-Eingänge $U_{1in}(-)$ sind intern über eine gemeinsame Leitung galvanisch mit GND verbunden; auch die Minus-Eingänge $U_{2in}(-)$ haben eine solche Verbindung.



CO POWER IN
(Buchse)

Abb. 13 – Pin-Belegung Zähler-Spannungsversorgung (Gold-D)

8.2 Software

Die für den Zugriff auf die Zähler benötigten Funktionen befinden sich in der Include-Datei:

<ADWGCNT . INC>

Binden Sie diese Include-Datei zum Beginn eines Programms ein, damit Sie die Befehle aus der nachfolgenden Tabelle benutzen können. Die Befehle sind im **ADbasic**-Handbuch beschrieben.

Befehl	Funktion
CNT_CLEAR () *	Zähler löschen
CNT_ENABLE ()	Zähler sperren oder freigeben (achten Sie auf bereits laufende Zähler)
CNT_GETSTATUS (#)	Statusregister auslesen (# = Zähler-Nr. 1...4)
CNT_INPUTMODE ()	CLR/LATCH-Eingang auf CLR- oder LATCH-Modus stellen
CNT_LATCH () *	Zählerstand in Latch A übernehmen
CNT_MODE ()	Externen Takteingang oder internen Referenztakt verwenden
CNT_SE_DIFF ()	Differentielle oder TTL-Eingänge einstellen (paarweise)
CNT_SET ()	In Kombination mit CNT_MODE () : Zähler-Modus oder Höhe des internen Referenztakts einstellen
CNT_READ (#)	Zählerstand in Latch A übernehmen und auslesen (# = Zähler-Nr. 1...4)
CNT_READLATCH (#)	Latch A (getriggert durch pos. Flanke) auslesen (# = Zähler-Nr. 1...4)
CNT_READFLATCH (#)	Latch B (getriggert durch neg. Flanke) auslesen (# = Zähler-Nr. 1...4)

* diese Funktionen werden nach der Durchführung wieder zurückgesetzt. Alle anderen Funktionen werden nur durch die entgegengesetzte Funktion aufgehoben.

Abb. 14 – Befehle der Gold-CO1-Zählererweiterung

Mit den Befehlen in der Tabellenmatrix beeinflussen Sie immer **alle** Zähler (ausgenommen CNT_READ...). Achten Sie deshalb darauf, immer alle Bits korrekt zu setzen oder zu löschen. Sie können dadurch jeden Zähler einzeln oder beliebig viele Zähler gemeinsam beeinflussen.

Konfigurieren Sie die Zähler bitte in dieser Reihenfolge:

1. Gewünschten Zähler sperren (CNT_ENABLE)
2. Betriebsart einstellen (CNT_MODE, CNT_SET, CNT_INPUTMODE, CNT_SE_DIFF)
3. Zähler löschen (CNT_CLEAR)
4. Zähler freigeben (CNT_ENABLE)

Für die Verarbeitung der Werte im **ADbasic**-Programm übertragen Sie die Werte ggf. ins Latch-Register und lesen sie dort aus.

Beachten Sie die Abhängigkeit des Befehls CNT_SET vom Befehl CNT_MODE.

Wenn Sie einen bestimmten Zähler sperren oder freigeben möchten, müssen Sie auch die schon laufenden Zähler freigeben (= Bits setzen). Wenn Sie (unbeabsichtigt) die Bits dieser Zähler nicht setzen, werden diese gesperrt.

8.2.1 Auswerten des Zählerinhalts

Die Binärzähler der CO1-Erweiterung erzeugen 32 Bit-Werte, die **ADbasic** nach dem Modell des unten stehenden Zahlenkreises als Zahlen mit Vorzeichen interpretiert: Das höchste Bit (MSB) stellt das Vorzeichen dar; die größte positive Zahl ($2^{31}-1$) schließt an die höchste negative Zahl (-2^{31}) an und die kleinste positive (0) an die kleinste negative Zahl (-1).

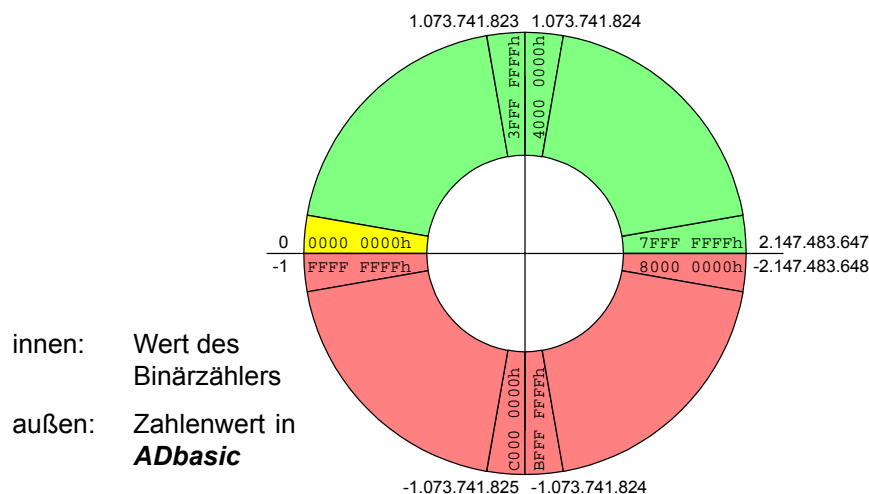


Abb. 15 – Zahlenkreis als Interpretation von Zählerwerten

Beachten Sie deswegen bei der Programmierung die nachstehenden Regeln:

- a) Verarbeiten Sie den gelesenen 32 Bit-Werts nur mit Variablen vom Typ **INTEGER** oder **LONG**. **ADbasic** behält dann intern das gelesene Bitmuster unverändert bei und berücksichtigt automatisch den Übergang zwischen positivem und negativem Zahlenbereich. Damit gilt:
- b) Die Zählrichtung (vor- oder rückwärts) ergibt sich zuverlässig nur aus dem **Vorzeichen der Differenz**: [neuer Zählerstand] minus [alter Zählerstand] und nicht aus dem Vergleich der Zählerstände.

Berücksichtigen Sie bei der Programmierung, dass ein „Überlauf“ zwischen dem Auslesen von zwei Zählerständen - d.h. der aktuelle Zählerstand „über-rundet“ den zuletzt gelesenen - nicht erfasst wird. Ein solcher Überlauf tritt bei einer Eingangsfrequenz von 20MHz nach etwas mehr als 3½ Minuten ein, bei 5MHz nach über 14 Minuten.

Sie finden mehrere Beispielprozesse zur CO1-Erweiterung im Verzeichnis <C:\ADwin\ADbasic3\samples_ADwin_Gold> (Standardinstallation).

Befehlsfolge

Zahlenkreis

Zählrichtung

„Überlauf“

Beispielprozesse

Löschen

Latches

8.3 Betriebsart Impuls-/Ereigniszähler

Externe Rechtecksignale an den Eingängen A/CLK und B/DIR takten in dieser Betriebsart den jeweiligen Zähler. Mit `CNT_SET` aktivieren Sie den Modus zur Ermittlung von Taktfrequenz und Richtung oder die Vierflankenbewertung.

Der Eingang CLR/LATCH kann benutzt werden, um (jeweils bei einem dort anliegenden High-Signal)

- den Zähler zu löschen (CLR)
- den Zählerstand ins Latch-Register A zu übernehmen (LATCH).

8.3.1 Takt und Richtung

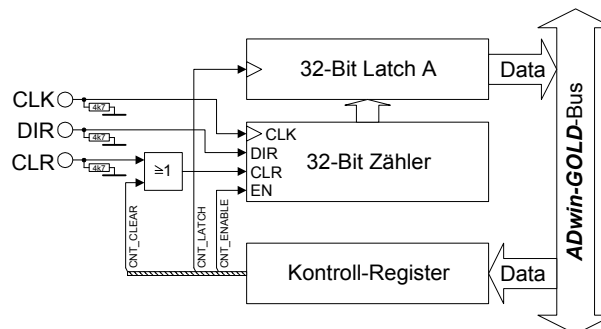
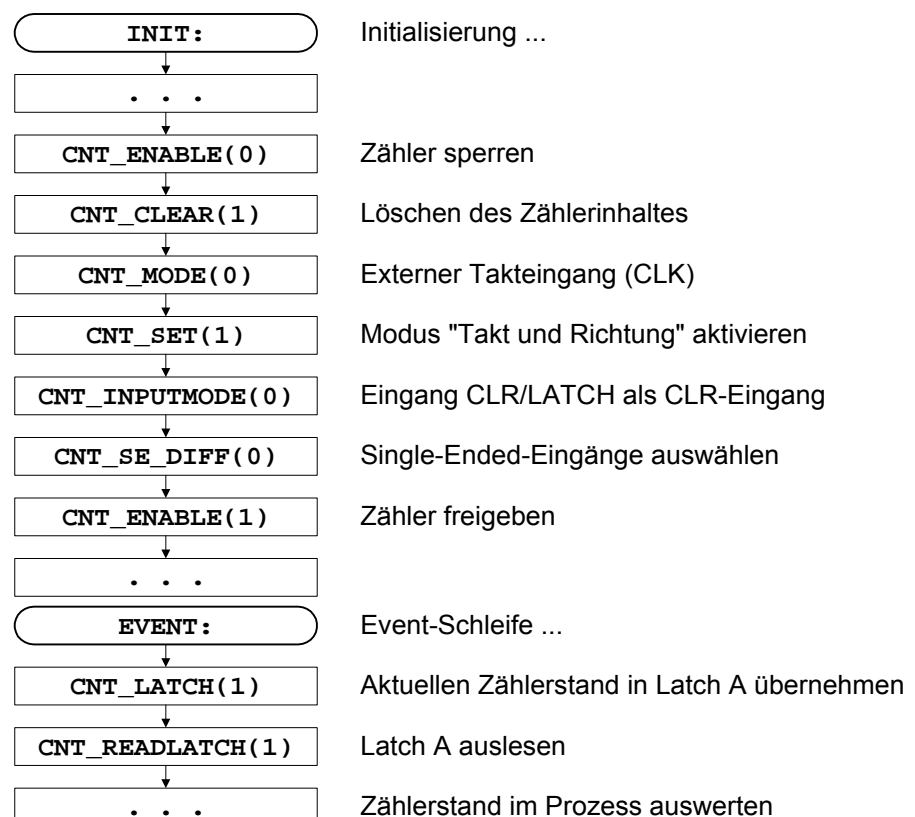


Abb. 16 – Schema CO1-Erweiterung im Modus „Takt und Richtung“

Jede positive Flanke eines Rechtecksignals auf dem CLK-Eingang (Clock) wird bis zu einer maximalen Frequenz von 20MHz gezählt. Die Richtung ergibt sich aus einem High- (vorwärts) bzw. Low-Signal (rückwärts) auf dem DIR-Eingang (Direction); dieses Signal kann sowohl statisch sein, für eine feste Zählrichtung, oder auch dynamisch, für wechselnde Zählrichtungen.

Programmierbeispiel



8.3.2 Vier-Flanken-Auswertung

Dieser Modus ermittelt Takt und Zählrichtung aus zwei Rechteck-Signalen, die an den Eingängen A und B um 90 Grad versetzt anliegen. Die Zählrichtung ergibt sich aus der zeitlichen Abfolge mit der die steigenden und fallenden Flanken der beiden Signale eintreffen.

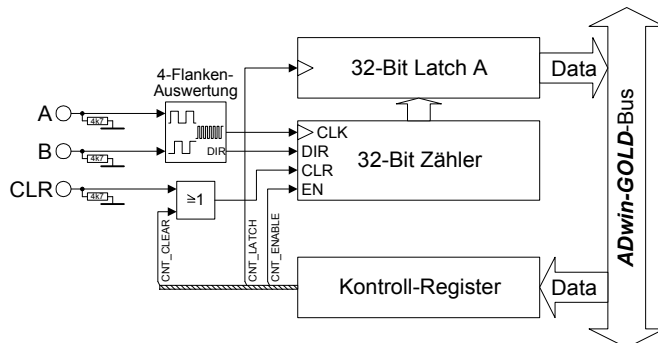
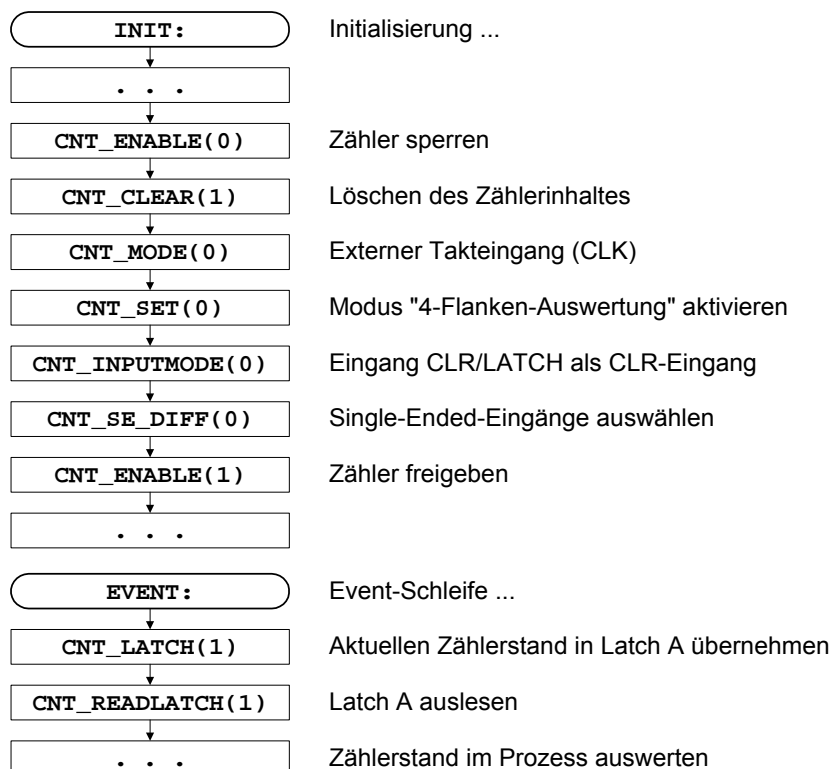


Abb. 17 – Schema CO1-Erweiterung im Modus „4-Flanken-Auswertung“

Berücksichtigen Sie bitte:

- Der Zähler registriert bei einem Zyklus des A/B-Signals 4 Flanken.
- Die maximale Zählfrequenz beträgt 20MHz. Gemeinsam mit den 4 Flanken je Zyklus ergibt sich daraus eine maximale Eingangsfrequenz von 5MHz.
- Der Abstand zwischen einer Flanke an A und einer Flanke an B darf 50ns nicht unterschreiten. Impulsbreiten oder Pausenzeiten kürzer als 100ns werden nicht gezählt.
- Eine Änderung der Phasenverschiebung hat Einfluss auf die maximale Eingangsfrequenz wegen der Mindestabstände der Flanken. Bei einem Abweichen von 90 Grad sinkt die maximale Eingangsfrequenz von 5MHz beispielsweise bei 45 Grad auf 2,5MHz.



Programmierbeispiel

Referenztaktgeber



8.4 Betriebsart Impulsbreiten- und Periodendauer-Messung

In dieser Betriebsart taktet ein interner Referenztaktgeber den Zähler mit einer Signalfrequenz von 20 MHz oder (nach einem Teiler) 5 MHz. Alle Zähler besitzen einen Umschalter für die Signalfrequenz. Es können die Periodendauer oder die Impuls-/Pausenzeit eines Rechtecksignals am Eingang LATCH gemessen werden.

In diesem Modus müssen Sie bei hohen Frequenzen berücksichtigen, dass Ihr GLOBALDELAY kleiner bleibt als eine Signalperiode, um jeden Zyklus zu erfassen.

8.4.1 Periodendauer-Messung

Die Messung der Periodendauer können alle 4 Zähler durchführen.

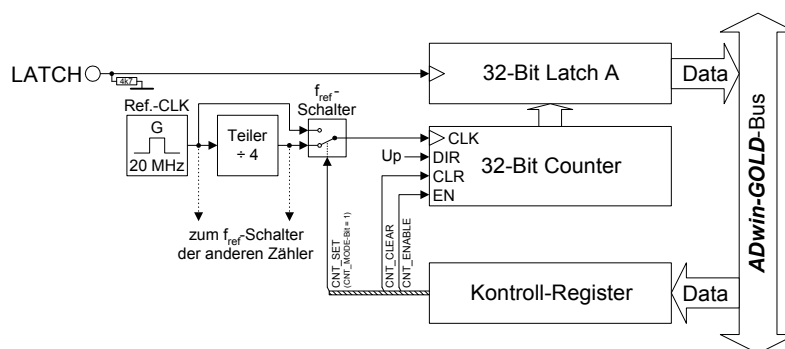
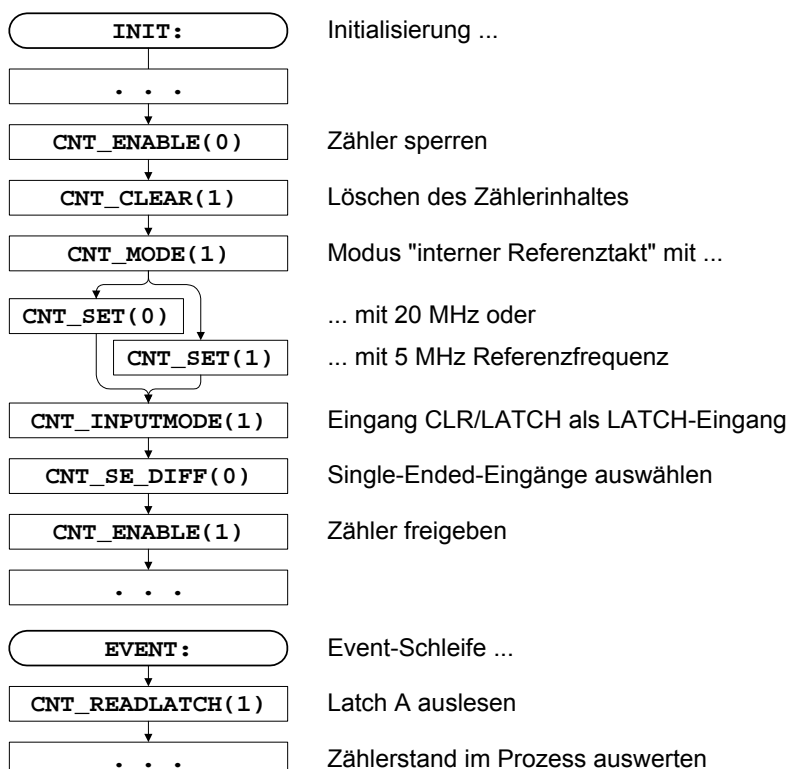


Abb. 18 – Schema CO1-Erweiterung im Modus „Periodendauermessung“

In diesem Modus wird bei jeder positiven Flanke am Eingang LATCH der Zählerstand in Latch A geschrieben, wobei der jeweils vorherige Stand überschrieben wird. Die Periodendauer ergibt sich aus dem Produkt von Zählerstands-differenz mal Periodendauer des Referenztaktes.

Programmierbeispiel



8.4.2 Messung von Impuls- und Pausenzeit

Die Messung von Impulsbreite und Pausenzeit können alle 4 Zähler durchführen.

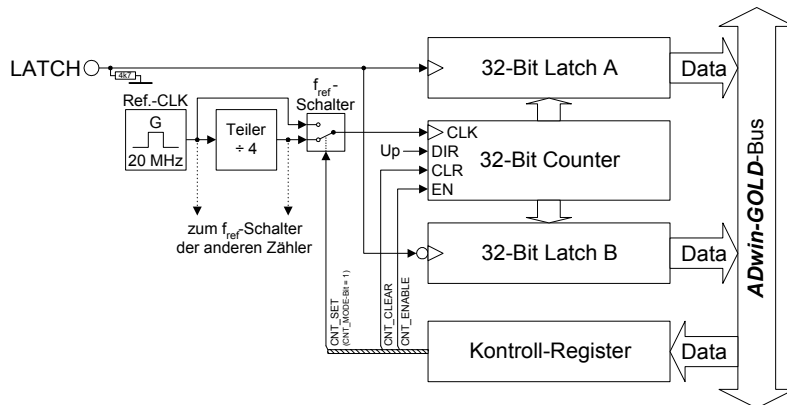
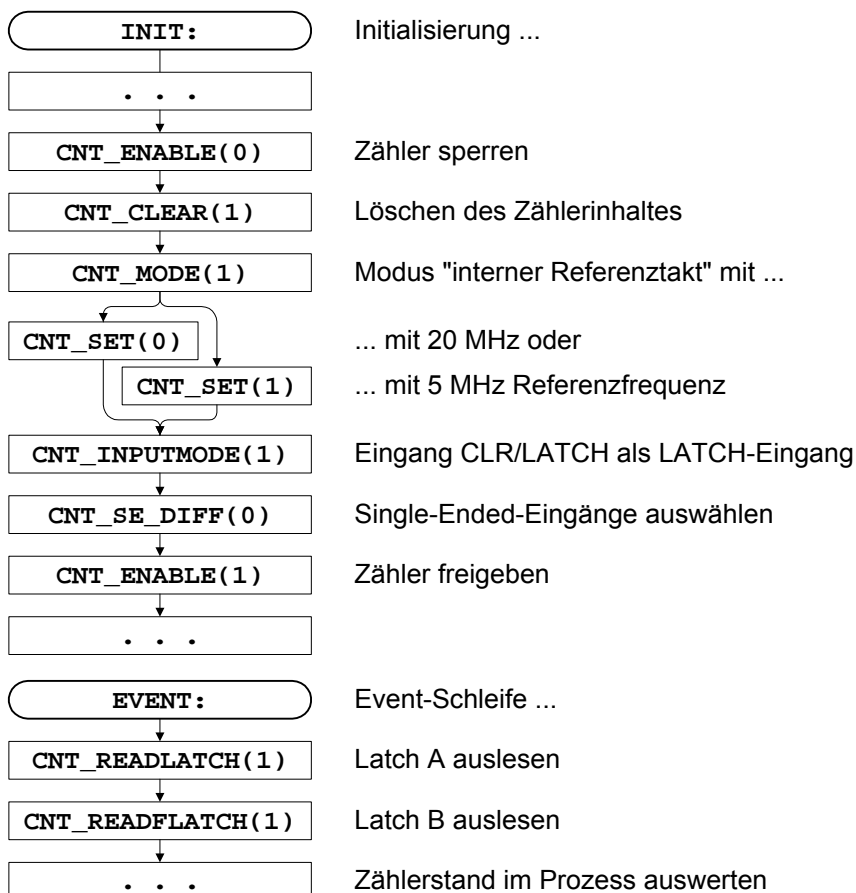


Abb. 19 – Schema CO1-Erweiterung im Modus „Impuls-/Pausenzeitmessung“

Die Zähler besitzen je ein Latch A für positive Flanken und ein Latch B für negative Flanken. Aus der Zählerstands- und Latch-Differenz können Impulsbreite und Pausenzeit unabhängig voneinander ermittelt werden.



Programmierbeispiel

8.5 Hardware-Adressen (CO1-Erweiterung)

Ein Prozess kann sehr schnell abgearbeitet werden, wenn Sie direkt auf die Steuer- und Datenregister zugreifen (siehe Kapitel 5.3 sowie **ADbasic**-Handbuch). Die Hardware-Adressen der CO1-Erweiterung können Sie folgender Tabelle entnehmen (vgl. Befehlstabelle in Kapitel 8.2).

Adresse [HEX]	Funktion	Bit					Kommentar
		31:04	3	2	1	0	
20400204	Latch A auslesen: Zähler 1	x	x	x	x	x	x : Inhalt des Latches
20400208	Latch B auslesen: Zähler 1	x	x	x	x	x	
20400214	Latch A auslesen: Zähler 2	x	x	x	x	x	
20400218	Latch B auslesen: Zähler 2	x	x	x	x	x	
20400224	Latch A auslesen: Zähler 3	x	x	x	x	x	
20400238	Latch B auslesen: Zähler 3	x	x	x	x	x	
20400234	Latch A auslesen: Zähler 4	x	x	x	x	x	
20400238	Latch B auslesen: Zähler 4	x	x	x	x	x	
20400300	Zähler freigeben / sperren	-	x	x	x	x	x = 0 : Zähler sperren x = 1 : Zähler freigeben
20400304	Umschalten zwischen Zähler-Eingängen für single-ended oder differentiellen Betrieb (nur paarweise)	-	-	-	y	x	x: Eingänge für Zähler 1+2 y: Eingänge für Zähler 3+4 x,y = 0: single-ended x,y = 1: differentiell
20400310	Zähler löschen	-	x	x	x	x	x = 0 : Kein Einfluss x = 1 : Zähler löschen
20400320	Zähler latches	-	x	x	x	x	x = 0 : Kein Einfluss x = 1 : Zähler latches
20400330	Eingang: CLR oder LATCH	-	x	x	x	x	x = 0 : CLR-Eingang x = 1 : LATCH-Eingang
20400340	Impuls-/Ereigniszähler oder Impuls-/Pausenzeitmessung	-	x	x	x	x	x = 0 : Ext. Takteingang x = 1 : Int. Ref.-Takt (20/5MHz)
20400350	4-Fl.-Auswertung / CLK+DIR oder 20 / 5MHz Ref.-Takt	-	x	x	x	x	CNT_MODE = 0: x = 0 : 4-Fl.; x = 1 : CLK+DIR CNT_MODE = 1: x = 0 : 20MHz; x = 1 : 5MHz
20400370	Zähler: Fehlerregister ^a	div. Bits					Fehler-Bits, s. CNT_GETSTATUS

a. Dieses Register müssen Sie von Hand wieder zurücksetzen!

Abb. 20 – Hardware-Adressen der CO1-Zählererweiterung

9 CAN-Erweiterung

Die Erweiterung *Gold-CAN* beinhaltet mehrere zusätzliche Schnittstellen, die unabhängig voneinander konfiguriert und betrieben werden können:

- 4 SSI-Decoder (Seite 34)

Die Decoder eignen sich zum Anschluss von Inkremental-Encodern mit SSI-Schnittstelle. Alle Eingänge sind differentiell und für RS422/485-Pegel (5V) ausgelegt.

Die Decoder-Eingänge liegen auf den Steckern CO1...CO4, die ggf. auch Eingänge der Erweiterung CO1 enthalten.

- 2 CAN-Schnittstellen (Seite 36)

Die beiden Schnittstellen eignen sich je nach Bestelloption entweder für „High-speed“ oder für „Low-speed“. Ein Umschalten im Betrieb ist nicht möglich.

Die Eingänge der Schnittstelle CAN 1 liegen auf den Steckern CAN 1.1 und CAN 1.2, die der Schnittstelle CAN 2 auf dem Stecker CAN 2.

- 2 RSxxx-Schnittstellen (Seite 40)

Beide Schnittstellen können unabhängig voneinander per Software auf RS232 oder auf RS485 eingestellt und betrieben werden.

Die Schnittstellen-Eingänge liegen auf den Steckern COM1 und COM2.

Die Erweiterung Gold-CAN ist nur gemeinsam mit der Bestelloption Gold-D erhältlich.

9.1 SSI-Decoder

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

Die Decoder können entweder (auf Anforderung) einen einzelnen Wert auslesen oder aber kontinuierlich den aktuellen Wert bereit stellen.

Die Anschlüsse der 4 Decoder stehen auf den Steckern CO1...CO4 (15-polig, Sub-D) zur Verfügung, und zwar auf den Pins 5, 8, 14 und 15 (siehe Abb. 21). Falls das Gerät mit der Erweiterung CO1 ausgerüstet ist, sind die übrigen Pins als Zähleranschlüsse geschaltet.

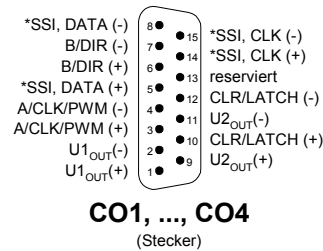
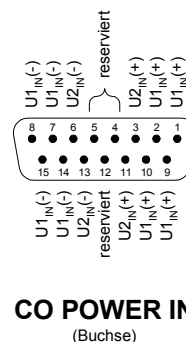


Abb. 21 – Pin-Belegung SSI-Decoder

Über den Anschluss CO Power in kann eine Spannung eingespeist werden, die an den Steckern CO1...CO4 zur Verfügung steht, z.B. für externe Inkrementalgeber.

Beachten Sie bitte: Die Minus-Eingänge U1_{in} (-) sind intern über eine gemeinsame Leitung galvanisch mit GND verbunden; auch die Minus-Eingänge U2_{in} (-) haben eine solche Verbindung.



Eigenschaften einstellen

Folgende Eigenschaften der Decoder sind per Software einstellbar:

- Taktrate: Über einen Vor-Teiler sind Taktraten von ca. 40kHz bis 1MHz möglich mit **SSI_SET_CLOCK**.
- Auflösung: Einstellbar bis 32 Bit mit **SSI_SET_BITS**.

Eine Umsetzung von Gray- in Binär-Code erfolgt durch eine zu programmierende Routine im **ADbasic**-Prozess (siehe unten).

```
REM PAR_1 = zu wandelnder Gray-Wert
REM PAR_2 = Flag für einen neuen Gray-Wert
REM PAR_9 = Ergebnis der Gray-zu-Binär-Wandlung

DIM m, n AS LONG

EVENT:
IF (PAR_2=1) THEN           'Start der Wandlung
    m=0                     'Variable initialisieren
    PAR_9=0                 ' _"-
    FOR n=1 TO 32           'Alle 32 Bits durchgehen
        m=(SHIFT_RIGHT(PAR_1,(32-n)) AND 1) XOR m
        PAR_9=(SHIFT_LEFT(m,(32-n))) OR PAR_9
    NEXT n
    PAR_2=0                 'Nächste Wandlung ermöglichen
ENDIF
```

Abb. 22 – Listing: Konvertierung von Gray- in Binär-Code

Die Funktionalität der Decoder wird mit **ADbasic**-Befehlen komfortabel programmiert:

Bereich	Befehle
Initialisierung	SSI_MODE SSI_SET_BITS SSI_SET_CLOCK
Empfangen von Daten	SSI_READ SSI_START SSI_STATUS

Die Befehle sind in der Include-Datei <ADWGCAN.INC> enthalten und werden im Handbuch **ADbasic** und der Online-Hilfe erläutert.



Beispiel:
Umsetzung von
Gray-Code

Programmierung

9.2 CAN-Schnittstelle

Die CAN-Schnittstellen 1 und 2 können voneinander unabhängig betrieben werden. Sie eignen sich je nach Bestelloption entweder für „High-speed“ oder für „Low-speed“. Eine Umschaltung im Betrieb ist nicht möglich.

9.2.1 Hardware-Beschreibung

Die Anschlüsse der Schnittstellen 1 und 2 stehen auf 9-poligen Sub-D-Verbindern zur Verfügung:

- Schnittstelle 1: Stecker CAN 1.1 und Buchse CAN 1.2. Die Pins der beiden Steckverbinder sind intern miteinander verbunden.
- Schnittstelle 2: Stecker CAN 2.

Die Pinbelegung für CAN „High speed“ und „Low speed“ ist unterschiedlich.

CAN-Bus	Anschlüsse CAN 1.1, CAN 2 (Stecker)	Anschluss CAN 1.2 (Buchse)
High speed	<p>Diagram showing the pinout for High speed CAN 1.1/2 (stecker). The 9-pin connector has the following assignments: Pin 1: reserviert, Pin 2: CAN(-), Pin 3: CAN-GND, Pin 4: reserviert, Pin 5: CAN(+), Pin 6: CAN-GND, Pin 7: reserviert, Pin 8: reserviert, Pin 9: reserviert.</p>	<p>Diagram showing the pinout for High speed CAN 1.2 (buchse). The 9-pin connector has the following assignments: Pin 1: reserviert, Pin 2: CAN(-), Pin 3: CAN-GND, Pin 4: reserviert, Pin 5: CAN(+), Pin 6: CAN-GND, Pin 7: reserviert, Pin 8: reserviert, Pin 9: reserviert.</p>
Low speed	<p>Diagram showing the pinout for Low speed CAN 1.1/2 (stecker). The 9-pin connector has the following assignments: Pin 1: reserviert, Pin 2: CAN(-), Pin 3: CAN-GND, Pin 4: reserviert, Pin 5: CAN(+), Pin 6: CAN-GND, Pin 7: reserviert, Pin 8: 12V (Eingang), Pin 9: reserviert.</p>	<p>Diagram showing the pinout for Low speed CAN 1.2 (buchse). The 9-pin connector has the following assignments: Pin 1: reserviert, Pin 2: CAN(-), Pin 3: CAN-GND, Pin 4: reserviert, Pin 5: CAN(+), Pin 6: CAN-GND, Pin 7: reserviert, Pin 8: 12V (Eingang), Pin 9: reserviert.</p>

Abb. 23 – CAN: Pinbelegungen

Beide Schnittstellen haben ein eigenes Potential CAN-GND, die sowohl voneinander galvanisch getrennt sind als auch vom Masse-Potential (GND) des Gehäuses.

Bei der „low speed“-Variante ist eine externe Spannungsversorgung mit 12V Gleichstrom erforderlich, um den CAN-Controller zu betreiben. Die Spannung muss für jede Schnittstelle separat eingespeist werden.

Wenn die CAN-Schnittstelle 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 für eine der beiden (oder beide) Schnittstellen die Terminierung erforderlich ist, müssen Sie die Pins CAN(+) und CAN(-) durch einen Widerstand von 120Ω verbinden.

9.2.2 Beschreibung der CAN-Schnittstelle

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

Spannungsversorgung
(nur Low speed)

Bus-Terminierung
(nur High speed)

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 255 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.

Verwaltung von Nachrichten

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 $0 \dots 2^{11}-1$ bzw. $0 \dots 2^{29}-1$.

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.

Jedes Message-Objekt erhält einen Identifizier, der die Zuordnung einer Nachricht zu einem Message-Objekt ermöglicht.

In **ADbasic** übergeben Sie eine Nachricht an ein Message-Objekt über das Feld `can_msg`, 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 `can_msg` übertragen.

Das Versenden einer Nachricht läuft in folgenden Schritten ab:

- Sie konfigurieren ein Message-Objekt zum Senden und definieren den Identifizier des Objekts (Befehl **EN_TRANSMIT**).
- Sie speichern die Nachricht im Feld `can_msg`.
- Sie senden die Nachricht (Befehl **TRANSMIT**). Die Nachricht im Feld `can_msg` wird an das Message-Objekt übergeben. Sobald der Bus frei ist, wird die Nachricht gesendet (mit dem Identifizier des Message-Objekts).

Das Empfangen einer Nachricht läuft in folgenden Schritten ab:

- 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 `can_msg` (Befehl **READ_MSG**) und lesen den zugehörigen Identifizier aus.



Identifizier

Message-Objekte

Nachricht übergeben

Nachricht senden

Nachricht empfangen

Nachricht zuordnen

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.

Bei dem Message Objekt 15 existiert ein zusätzlicher interner Zwischenspeicher, so dass dort 2 Nachrichten gespeichert werden können.

Die Zuordnung einer eingehenden Nachricht zu einem Message-Objekt wird automatisch durch einen Vergleich ihrer Identifier gesteuert. Die globale Maske (CAN-Register 6...7 bzw. 6...9) steuert diesen Vergleich:

- Der Identifier der Nachricht wird bitweise mit dem Identifier 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.

Globale Maske

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-Identifier 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, müssen die Werte im „Bit Timing Register 0“ (BTR0, Adresse 3Fh) und „Bit Timing Register 1“ (BTR1, Adresse 4Fh) geändert werden. Für eine große Auswahl an Busfrequenzen 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 nachfolgend beschrieben.

Busfrequenz für Sonderfälle

Bit Timing Register 0
(BTR0)

Bit Timing Register 1
(BTR1)

Bits	7...6	5...0	7	6...4	3...0
Sub-Reg.	SJW	BRP	SPL	TSEG2	TSEG1

Die folgende Tabelle zeigt zulässige Werte und die Bedeutung der Bereiche:

Bereich	zulässige Werte	Bedeutung
SJW	0 ... 3	Maximale Pulsdehnung bei der Bus-Synchronisation
BRP	0 ... 63	Vor-Teiler
SPL	0 ... 1	Sampling Mode
TSEG1	2 ... 15	Zeitsegmente vor der Abtastung
TSEG2	1 ... 7	Zeitsegmente nach der Abtastung

Die Bereiche SJW und SPL sind standardmäßig 0 und sollten nur bei Bedarf geändert werden. Der Sample-Punkt (bestimmt durch TSEG1 und TSEG2) sollte so gewählt werden, dass er zwischen 50% und 80% der Gesamt-Bitlänge liegt.

Die CAN-Bus-Frequenz berechnet sich folgendermaßen:

$$f_{\text{CAN}} = \frac{8\text{MHz}}{(\text{BRP} + 1)(\text{TSEG1} + \text{TSEG2} + 3)}$$

Die folgende Tabelle zeigt die Einstellungen für die gängigsten Baudraten.

Baudrate [kBit/s]	125	250	500	1000
BRP	3	1	0	0
TSEG1	6	6	6	2
TSEG2	7	7	7	3
BTR0	03h	01h	00h	00h
BTR1	76h	76h	76h	32h
Sample-Punkt [%]	54	54	54	60

Abb. 24 – CAN: Gängige Baudraten einstellen

Der Zugriff auf die beiden Timing-Register BTR0 und BTR1 ist nur möglich, wenn der Zugriff zuvor freigegeben wird. Dies geschieht mit dem „CCE-Bit“ im „Control Register“. Das Bit muss anschließend wieder zurückgesetzt werden.

Interrupt / Event

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) gesehen 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.

Programmierung

Die Schnittstelle wird mit **ADbasic**-Befehlen komfortabel programmiert:

Bereich	Befehle
Initialisierung	INIT_CAN EN_INTERRUPT SET_CAN_BAUDRATE
Empfangen und Senden von Daten	CAN_MSG EN_RECEIVE EN_TRANSMIT READ_MSG TRANSMIT
Schreib- / Lesezugriff auf Controller-Register	SET_CAN_REG GET_CAN_REG

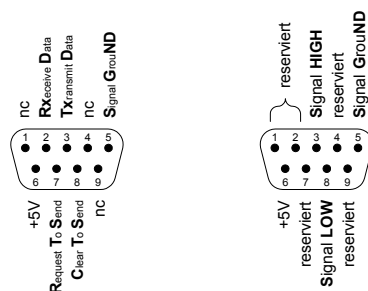
Die Befehle sind in der Include-Datei <ADWGCAN.INC> enthalten und werden im Handbuch **ADbasic** und der Online-Hilfe erläutert.

9.3 RSxxx-Schnittstellen

Jede der 2 RSxxx-Schnittstellen ist 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.

Beide Schnittstellen können unabhängig voneinander mit dem Protokoll RS232 oder RS485 betrieben werden. Der physikalische Unterschied der Protokolle liegt in den Pegeln der Signale, die auf dem „Bus“ durch entsprechende Treiber-Bausteine bereit gestellt werden.

Pinbelegung



Bus-Terminierung (nur RS485)

Wenn eine RS485-Schnittstelle das physikalische Bus-Ende bildet, muss es mit einem Abschlusswiderstand terminiert werden (also nur der erste oder letzte RS485-Teilnehmer). Bei RS485-Teilnehmern, die sich nicht an einem physikalischen Ende der Kette befinden, darf nicht terminiert werden.

Für die Terminierung steht – falls die gewählte Schaltungsvariante dies erfordert – am Pin 6 eine Spannung von +5V zur Verfügung.

9.3.1 Einstellung der Schnittstellen-Parameter

Jede Schnittstelle verfügt über einen Eingangs- und einen Ausgangs-FIFO mit einer Länge von jeweils 64 Byte.

Die Schnittstellen-Parameter werden mit Hilfe der Controller-Register eingestellt, und zwar getrennt für jede Schnittstelle. Im folgenden werden die Einstellmöglichkeiten dargestellt.

- Handshake: Die Schnittstelle kann in 4 Modi betrieben werden:
 1. RS232 ohne Handshake
 2. RS232 mit Software-Handshake (Xon/Xoff)
 3. RS232 Hardware-Handshake (RTS/CTS). Hierbei müssen die Signale RTS und CTS angeschlossen sein.
 4. RS485
- 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. 25 – 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:

- Es gibt keinen Handshake, da ein Handshake immer nur zwischen 2 Teilnehmern funktioniert.
- Jeder Schnittstelle muss vor dem Betrieb mitgeteilt werden, ob sie auf den Bus schreiben soll oder nur Daten vom Bus übernehmen darf (**RS485_SEND**).

Handshake

Parität

Daten-Bits

Stopp-Bits

Baudrate

Besonderheiten RS485

9.3.2 Programmierung

Die Funktionalität und Programmierung der Schnittstelle beruhen auf dem eingebauten Schnittstellen-Controller. Der Controller wird mit **ADbasic**-Befehlen komfortabel programmiert:

Bereich	Befehle
Initialisierung	RS_INIT, RS_RESET
Empfangen und Senden von Daten	RS485_SEND, READ_FIFO, WRITE_FIFO
Schreib- / Lesezugriff auf Controller-Register	GET_RS, SET_RS

Die Befehle sind in der Include-Datei <ADWGCAN.INC> enthalten und werden im Handbuch **ADbasic** und der Online-Hilfe erläutert.

Beispielprogramme

RS232

Das nachfolgende Programm zeigt die Initialisierung der seriellen RS232-Schnittstelle im Abschnitt **INIT**: und das zyklische Lesen und Schreiben von Daten im Abschnitt **EVENT**:. Der Prozess ist zeitgesteuert.

```
REM Das Programm initialisiert die seriellen
REM Schnittstellen im Abschnitt Init:
REM Im Abschnitt Event: werden Daten zwischen den
REM Schnittstellen 1 & 2 des RS-Moduls ausgetauscht.
REM Mit Hilfe dieses Programms können die Schnittstellen
REM untereinander getestet werden. Dazu müssen Sie die
REM Schnittstellen vor dem Programmstart miteinander
REM verbinden.
```

```
#INCLUDE adwgcان.inc
DIM DATA_1[1000] AS LONG 'Sendedaten
DIM DATA_2[1000] AS LONG 'Empfangsdaten
DIM lauf AS LONG          'Laufvariable

INIT:
  FOR lauf = 1 TO 1000 'Initialisierung der Sendedaten
    DATA_1[lauf] = lauf AND 0FFh
  NEXT lauf
  REM Initialisierung der Schnittstellen:
  REM 9600 Baud, Kein Paritätsbit, 8 Datenbits,
  REM 2 Stoppbits, RS232 ohne Handshake
  RS_INIT(1,9600,0,8,1,0)
  RS_INIT(2,9600,0,8,1,0)
  PAR_1 = 1
  PAR_4 = 1

EVENT:
  REM Einen Datensatz lesen und schreiben
  IF (PAR_1 <= 1000) THEN 'Daten senden
    PAR_2 = WRITE_FIFO(1,DATA_1[PAR_1])
    IF (PAR_2 = 0) THEN INC PAR_1
  ENDIF

  PAR_3 = READ_FIFO(2) 'Daten lesen
  IF (PAR_3 <> -1) THEN
    DATA_2[PAR_4] = PAR_3
    INC PAR_4
  ENDIF
  IF (PAR_4 > 1000) THEN END 'Alle Daten sind übertragen
```



RS485

In diesem Beispiel wird eine RS485-Schnittstelle als passiver Teilnehmer verwendet, der alle Daten liest, die an seinem Eingang anliegen. Wenn ein bestimmter Wert (55) empfangen wird, wird die Schnittstelle aktiv und sendet dann ihrerseits fortlaufend den Wert 44.

REM Schnittstelle 2 liest so lange alle Daten vom Bus, bis
REM sie den Wert 55 empfängt. Danach wird die Schnittstelle
REM aktiv und sendet den Wert 44.

```
#INCLUDE adwgcan.inc
DIM ret_val, val AS LONG
```

INIT:

```
RS_RESET()
REM Initialisierung der Schnittstellen:
REM 38400 Baud, Kein Paritätsbit, 8 Datenbits,
REM 1 Stoppbit, RS485 Software-Handshake
RS_INIT(1,38400,0,8,0,3)
RS_INIT(2,38400,0,8,0,3)
RS485_SEND(1,1)      'Schnittstelle 1 senden
RS485_SEND(2,0)      'Schnittstelle 2 empfangen
```

EVENT:

```
val = READ_FIFO(2)    'Daten aus Schnittstelle 2 lesen

IF (val = 55) THEN
  RS485_SEND(2,1)      'Schnittstelle 2 senden
  ret_val = WRITE_FIFO(2,44) 'Daten schreiben
ENDIF
```

10 ADwin-Gold-Boot

Diese Option ist nur bei **ADwin-Gold-ENET** möglich.

ADwin-Gold-Boot startet eine zuvor programmierte Anwendung automatisch nach dem Einschalten. Damit ist nach dem Einrichten der Anwendung ein Betrieb ohne PC möglich.

Folgende Schritte führt **ADwin-Gold-Boot** nach dem Einschalten aus:

- Laden des Betriebssystems
- Laden der mit dem **ADbasic**-Compiler kompilierten Prozesse (max. 10).
- Automatisches Starten des Prozesses Nr. 10. Hier müssen Sie auch das Starten weiterer Prozesse programmieren.

Wenn Sie nicht mit der Bootloader-Option arbeiten wollen:

- Booten Sie das System nach dem Einschalten, und die gespeicherten Prozesse werden deaktiviert.
- Nach dem Ausschalten und erneuten Einschalten ist die Bootloader-Option wieder aktiv.

Durch Beschreiben des Flash-EEPROM ohne Prozesse und nur mit der Datei `<ADwin9.btl>` wird das System nach dem erneuten Einschalten nur noch gebootet, aber ein Prozess kann nicht ausgeführt werden.

Wenn Sie **ADwin**-Software für Entwicklungsumgebungen von der beigefügten **ADwin**-CD installieren, wird das Programm für die Bootloader-Option (ADethflash) automatisch kopiert. Die Version der CD sollte 3.002735 oder höher sein.

Benutzen Sie für ein **ADwin-Gold**-System mit einem Ethernet-Interface das Programm `<ADethflash.exe>`.

Bei Standardinstallation finden Sie das Programm in dem Verzeichnis

`<C:\ADwin\Tools\Ethernet Interface\...>`.

Hinweise zum Bootloader mit Ethernet-Interface finden Sie im Handbuch Treiber-Installation.

In Verbindung mit dem Ethernet-Interface mit Bootloader können Sie bis zu 2000 Long- oder Float-Werte zu 32Bit mittels **ADbasic**-Prozess im Flash-EEPROM-Speicher ablegen und auslesen. Eine nähere Beschreibung hierzu können Sie im Programm `<ADethflash.exe>` aufrufen mit der Schaltfläche „Info about eeprom support“.

Bootloader deaktivieren

Hilfe zur Ethernet-Schnittstelle

2000 Werte zur freien Verfügung

ADwin-Gold-pow

11 Zubehör

Für das **ADwin-Gold**-System ist folgendes Zubehör lieferbar:

- **ADwin-Gold-pow**: externes 12V-Netzteil
- diverse Längen der Spannungsversorgungs- und USB- bzw. Ethernet-Kabel
- Kabel-Stecker für eine externe Spannungsversorgung
- Montagesätze für Schaltschränke

ADwin-Gold-pow stellt auf der Sekundärseite 12 Volt bei einer maximalen Dauerbelastung von 2 Ampere zur Verfügung. Das Netzteil ist für maximale Erweiterung und Auslastung ausgelegt.

Achten Sie auf ausreichende Schirmung des USB- und Ethernet-Kabels, um Störungen auf den Datenleitungen zu vermeiden. Störungen müssen vor dem Gehäuse über die Masse abgeleitet werden (siehe auch Kapitel 3 „Betriebliche Umgebung“).

12 Software

Sie programmieren das **ADwin-Gold**-System inklusive aller Erweiterungen mit einfachen **ADbasic**-Befehlen. Diese Befehle sind im **ADbasic**-Handbuch beschrieben.

Im Anhang des Handbuchs finden Sie eine Übersicht, welche Befehle für **ADwin-Gold** und die Erweiterungen verfügbar sind.

Anhang

A.1 Technische Daten

Sämtliche technischen Daten beziehen sich auf ein eingeschaltetes **ADwin-Gold**-System.

Allgemeine Daten / Grenzwerte						
	Symbol	Konditionen	min.	typ.	max.	Einheit
Versorgungs-Spannung						
Spannung	U _b		10	12	35	V
Ruhestrom, USB-Schnittstelle	I _{idle}	U _b =10V		1,1		A
		U _b =12V		0,9		
		U _b =35V		0,3		
		U _b =12V; Gold-DA		1,4		
		U _b =12V; Gold-CO1		0,9		
Einschaltstrombedarf, USB-Schnittstelle	I _{power-on}	U _b =12V	1,7			A
		U _b =12V; Gold-DA	2,9			
		U _b =12V; Gold-CO1	1,7			
Ruhestrom, Ethernet-Schnittstelle	I _{idle}	U _b =10V		1,3		A
		U _b =12V		1,1		
		U _b =35V		0,4		
		U _b =12V; Gold-DA		1,5		
		U _b =12V; Gold-CO1		1,1		
Einschaltstrombedarf, Ethernet-Schnittstelle	I _{power-on}	U _b =12V	2,1			A
		U _b =12V; Gold-DA	3,1			
		U _b =12V; Gold-CO1	2,1			
Zulässiger Betriebsbereich						
Temperatur	T _{Gehäuse}		+5		+60	°C
rel. Feuchte	F _{rel}	nicht kondensierend	0		80	%
Lagerung						
Temperatur	T		-20		+70	°C
Steckverbinder						
Sub-D-Verbinder	Metrisches ISO-Gewinde; UNC-Gewinde als Bestelloption erhältlich					
Abmessungen						
Breite x Höhe x Tiefe	B x H x T	Gold-USB, Gold-ENET	214 × 67 × 109			mm
		+ CO1-Erweiterung	Höhe: +30			
Nettogewicht						
Gewicht	m _{Netto}	Gold-USB, Gold-ENET	1.320			g
		mit CO1-Erweiterung	1.760			
		Clipse ^a	32			

a. Zubehör zur Hutschienenmontage: *Gold-Mount*

Digitale Ein- / Ausgänge						
Parameter	Symbol	Konditionen	min.	typ.	max.	Einheit
I/O-Leitungen						
Anzahl	DIO00:DIO31	32 (in Gruppen zu 8 als Ein-oder Ausgang programmierbar)				
	EVENT	ext. Trigger-Eingang (positive TTL-Logik)				
Eingänge						
max. Eingangsspanng.		$V_{CC} = 5V$	-0,5		+5,5	V
Logik-Eingangsspannung	V_{IH} (High)	$V_{CC} = 5V$	2,4			
	V_{IL} (Low)	$V_{CC} = 5V$			0,8	
Logik-Eingangsstrom	I_I	$V_{CC} = 5V$		±0,01	±2	µA
Ausgänge						
Logik-Ausgangsspannung	V_{OH} (High)	$I_{OH} = -6mA$	3,84	4,3		V
	V_{OL} (Low)	$I_{OL} = +6mA$		0,17	0,33	
Logik-Ausgangsstrom	I_O	je DIO-Leitung			±35	mA
	I_{TOTAL}	alle DIGIN bzw. alle DIGOUT über V_{CC} / GND			±70	
EVENT-Eingang						
Flankenerkennung, pos.	V_{T+} (Low)	$V_{CC} = 5V$	1,65	1,9	2,15	V
Schalthysterese	$V_{T+} - V_{T-}$		0,4	0,9		
Eingangsstrom	I_{IH}	$V_I = 2,7V$			20	µA
	I_{IL}	$V_I = 0,4V$			-50	

Analoge Ein- / Ausgänge						
Parameter	Symbol	Konditionen	min.	typ.	max.	Einheit
Eingänge						
Anzahl	2 × 8 über Multiplexer, differentiell					
Eingangswiderstand	R _i		323,4	330	336,6	kΩ
Spannungsfestigkeit	U _{in max.}	ON & OFF			±35	V
Multiplexer- Einschwingzeit	t _{MUX}	1 LSB 14Bit		2,5		µs
		1 LSB 16Bit		6,5		µs
ADC 14Bit						
Konvertierungszeit	t _{conv}				0,5	µs
Messbereich	U _{in}	F _v =1	-10		+9,999695	V
		F _v =2	-5		+4,999847	
		F _v =4	-2,5		+2,499924	
		F _v =8	-1,25		+1,249962	
Diff. Gleichtaktspanng.					±2,5	LSB
Integrale Nichtlinearität	INL			±1	±3	
Different. Nichtlinearität	DNL			±0,25	±0,5	

Analoge Ein- / Ausgänge						
Parameter	Symbol	Konditionen	min.	typ.	max.	Einheit
Offset	Drift			±2		ppm/K
	Fehler	abgleichbar				
Gain	Drift			±5		ppm/K
	Fehler	abgleichbar				
ADC 16Bit						
Konvertierungszeit	t _{conv}				5	µs
Messbereich	U _{in}	F _V =1	-10		+9,999695	V
		F _V =2	-5		+4,999847	
		F _V =4	-2,5		+2,499924	
		F _V =8	-1,25		+1,249962	
Diff. Gleichtaktspanng.					±2,5	
Integrale Nichtlinearität	INL			±1	±3	LSB
Different. Nichtlinearität	DNL			±0,25	±0,5	
Offset	Drift			±2		ppm/K
	Fehler	abgleichbar				
Gain	Drift			±5		ppm/K
	Fehler	abgleichbar				
Ausgänge: DAC 16 Bit						
Anzahl	2 (mit DA-Erweiterung: 8)					
Ausgangsspannung	U _{out}		-10		+9,999695	V
Einschwingzeit	t _{settle}	2V-Sprung		3		µs
		FSR ^a (20V)		10		
Zulässiger Strom					±25	mA
Integrale Nichtlinearität	INL				±2	LSB
Different. Nichtlinearität	DNL				±1	
Offset	Drift			±1		ppm/K
	Fehler	abgleichbar				
Gain	Drift			±3		ppm/K
	Fehler	abgleichbar				

a. Full Scale Range

Prozessor						
Parameter	Symbol	Konditionen	min.	typ.	max.	Einheit
Typ	ADSP21062 (SHARC™)					
Hersteller	Analog Devices					
Taktfrequenz	f_{CLK}			40		MHz
Register-Breite				32		Bit
Interner Speicher	SRAM	für Programm		128	256 ^a	kByte
		für Daten		128	256 ^a	
Externer Speicher	SDRAM			16	64 ^a	MByte

a. kombinierte Speichererweiterung G-MEM-64-512

CO1-Erweiterung						
Parameter	Symbol	Konditionen	min.	typ.	max.	Einheit
Zähler						
Anzahl	4 Zähler (CNTR1 ... CNTR4)					
Eingänge	Je Zähler 3 differentielle Eingänge (A/CLK, B/DIR, CLR/LATCH); Zähler paarweise programmierbar mit differentiellen oder single-ended Eingängen.					
Zählerbreite				32		Bit
Zählfrequenz	f_{CLK}	Eingang CLK		20		MHz
		Eingang A/B		5		
Latch-Breite	LATCH			32		Bit
Referenz-Quarzoszillator						
Referenzfrequenz	f_{ref}			20		MHz
Vorteiler durch 4	$f_{\text{ref}} / 4$			5		
Genauigkeit und Drift					100	ppm
Zählereingänge differentiell ^a						
Differentielle Eingangs-Schwellenspannung	V_{TH}	$-10\text{V} \leq V_{\text{CM}} \leq 13,2\text{V}$	-200		+200	mV
Schalthysterese	ΔV_{TH}	$-10\text{V} \leq V_{\text{CM}} \leq 13,2\text{V}$		40		mV
Bereich der Gleichtaktspannung	V_{CM}		-10		+13,2	V
Anstieg-/Abfallgeschw. der differentiellen Spg.			0,33			V/ μs
Zulässige differentielle Eingangsspannung		für jeden Eingang			$\pm 3,9$	V
Zählereingänge single ended ^b (mit Schmitt-Trigger)						
Flankenerkennung, pos.	$V_{\text{T+}} (\text{Low})$	$V_{\text{CC}} = 5\text{V}$	1,65	1,9	2,15	V
Flankenerkennung, neg.	$V_{\text{T-}} (\text{Low})$		0,75	1,0	1,25	
Schalthysterese	$V_{\text{T+}} - V_{\text{T-}}$		0,4	0,9		
Eingangsstrom	I_{H}	$V_{\text{I}} = 2,7\text{V}$			20	μA
	I_{L}	$V_{\text{I}} = 0,4\text{V}$			-50	

a. siehe auch Datenblatt MAX3098 von MAXIM

b. siehe auch Datenblatt 74LS19 von Texas Instruments

A.2 Hardware-Adressen - Gesamtübersicht

Adresse [HEX]	Funktion	Bit Nr.												Kommentar	Register verfügbar im Modul			
		31:16	15:10	9	8	7	6	5	4	3	2	1	0		Gold	Gold- DA	Gold- CO1	
20400000	MUX 1 setzen: IN1, IN3, ..., IN15	-	-	-	-	-	-	-	-	-	n	n	n	„nnn“ binär = 0...7 dezimal, gewählter Kanal = nnn + 1	x	x	x	
	MUX 2 setzen: IN2, IN4, ..., IN16	-	-	-	-	-	-	n	n	n	-	-	-	„nnn“ binär = 0...7 dezimal, gewählter Kanal = 2×(nnn + 1)	x	x	x	
	Verstärkung PGA 1	-	-	-	-	g	g	-	-	-	-	-	-	„gg“ binär = 0...3 dezimal, gewählter Kanal = 2 ^{gg}	x	x	x	
	Verstärkung PGA 2	-	-	g	g	-	-	-	-	-	-	-	-	-		x	x	x
20400010	Konvertierung starten: ADC 1, 16Bit	-	-	-	-	-	-	-	1	1	1	1	s	s = 0 : Konvertierung starten s = 1 : kein Einfluss	x	x	x	
	Konvertierung starten: ADC 2, 16Bit	-	-	-	-	-	-	-	1	1	1	s	1					
	Konvertierung starten: alle DAC syn- chron	-	-	-	-	-	-	-	1	1	s	1	1					
	Konvertierung starten: ADC 1, 14Bit	-	-	-	-	-	-	-	1	s	1	1	1					
	Konvertierung starten: ADC 2, 14Bit	-	-	-	-	-	-	-	s	1	1	1	1					
	Konvertierungs-Status: ADC 1, 16Bit	-	-	-	-	-	-	-	-	-	-	-	e					e = 0 : Konvertierung beendet e = 1 : Konvertierung läuft
Konvertierungs-Status: ADC 2, 16Bit	-	-	-	-	-	-	-	-	-	-	-	e	-					
Konvertierungs-Status: ADC 1, 14Bit	-	-	-	-	-	-	-	-	e	-	-	-	-					
Konvertierungs-Status: ADC 2, 14Bit	-	-	-	-	-	-	-	e	-	-	-	-	-					
20400030	Register auslesen: ADC 1, 16Bit	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
20400040	Register auslesen: ADC 2, 16Bit	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
20400050	Register nur beschreiben: DAC 1	-	x	x	x	x	x	x	x	x	x	x	x	x	x : zu wandelnder Digitalwert	x	x	x
20400060	Register nur beschreiben: DAC 2	-	x	x	x	x	x	x	x	x	x	x	x	x		x	x	x
20400070	Register nur beschreiben: DAC 3	-	x	x	x	x	x	x	x	x	x	x	x	x		-	x	-
20400080	Register nur beschreiben: DAC 4	-	x	x	x	x	x	x	x	x	x	x	x	x		-	x	-
20400090	Register nur beschreiben: DAC 5	-	x	x	x	x	x	x	x	x	x	x	x	x		-	x	-
204000A0	Register nur beschreiben: DAC 6	-	x	x	x	x	x	x	x	x	x	x	x	x		-	x	-
204000B0	Eingangs-Register DIO00:DIO15	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
204000C0	Ausgangs-Register DIO16:DIO31	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	
20400100	Register auslesen und Konvertie- rung starten: ADC 1, 16Bit	-	x	x	x	x	x	x	x	x	x	x	x	x	x : Ergebnis der Konvertierung	x	x	x
20400110	Register auslesen und Konvertie- rung starten: ADC 2, 16Bit	-	x	x	x	x	x	x	x	x	x	x	x	x		x	x	x
20400120	Register auslesen und Konvertie- rung starten: ADC 1, 14Bit	-	x	x	x	x	x	x	x	x	x	0	0	x		x	x	x
20400130	Register auslesen: ADC 1, 14Bit	-	x	x	x	x	x	x	x	x	x	0	0	x		x	x	x
20400140	Register auslesen: ADC 2, 14Bit	-	x	x	x	x	x	x	x	x	x	0	0	x		x	x	x
20400190	Register nur beschreiben: DAC 7	-	x	x	x	x	x	x	x	x	x	x	x	x	x : zu wandelnder Digitalwert	-	x	-
204001A0	Register nur beschreiben: DAC 8	-	x	x	x	x	x	x	x	x	x	x	x	x		-	x	-
204001B0	Eingangs-Register DIO16:DIO31	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
204001C0	Ausgangs-Register DIO00:DIO15	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
204001D0	Register auslesen und Konvertie- rung starten: ADC 2, 14Bit	-	x	x	x	x	x	x	x	x	x	0	0	x	x	x	x	x
204001E0	DIO00:DIO07 konfigurieren	0	0	0	0	0	0	0	0	-	-	-	c	c = 0 : Eingänge c = 1 : Ausgänge	x	x	x	
	DIO08:DIO15 konfigurieren	0	0	0	0	0	0	0	0	-	-	c	-		x	x	x	
	DIO16:DIO23 konfigurieren	0	0	0	0	0	0	0	0	-	c	-	-		x	x	x	
	DIO24:DIO31 konfigurieren	0	0	0	0	0	0	0	0	c	-	-	-		x	x	x	
20400200	Register beschreiben und sofort Konvertierung starten: DAC 1	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
20400204	Inhalt Latch A, Zähler 1	x	x	x	x	x	x	x	x	x	x	x	x	x	x : Inhalt des Latches	x	x	x
20400208	Inhalt Latch B, Zähler 1	x	x	x	x	x	x	x	x	x	x	x	x	x		-	-	x
20400210	Register beschreiben und sofort Konvertierung starten: DAC 2	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
20400214	Inhalt Latch A, Zähler 2	x	x	x	x	x	x	x	x	x	x	x	x	x	x : Inhalt des Latches	x	x	x
20400218	Inhalt Latch B, Zähler 2	x	x	x	x	x	x	x	x	x	x	x	x	x		-	-	x
20400220	Register beschreiben und sofort Konvertierung starten: DAC 3	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
20400224	Inhalt Latch A, Zähler 3	x	x	x	x	x	x	x	x	x	x	x	x	x	x : Inhalt des Latches	-	-	x
20400228	Inhalt Latch B, Zähler 3	x	x	x	x	x	x	x	x	x	x	x	x	x		-	-	x
20400230	Register beschreiben und sofort Konvertierung starten: DAC 4	-	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
20400234	Inhalt Latch A, Zähler 4	x	x	x	x	x	x	x	x	x	x	x	x	x	x : Inhalt des Latches	-	-	x
20400238	Inhalt Latch B, Zähler 4	x	x	x	x	x	x	x	x	x	x	x	x	x		-	-	x

Adresse [HEX]	Funktion	Bit Nr.												Kommentar	Register verfügbar im Modul		
		31:16	15:10	9	8	7	6	5	4	3	2	1	0		Gold	Gold- DA	Gold- CO1
20400240	Register beschreiben und sofort Konvertierung starten: DAC 5	-	x	x	x	x	x	x	x	x	x	x	x	x : zu wandelnder Digitalwert	-	x	-
20400250	Register beschreiben und sofort Konvertierung starten: DAC 6	-	x	x	x	x	x	x	x	x	x	x	x		-	x	-
20400260	Register beschreiben und sofort Konvertierung starten: DAC 7	-	x	x	x	x	x	x	x	x	x	x	x		-	x	-
20400270	Register beschreiben und sofort Konvertierung starten: DAC 8	-	x	x	x	x	x	x	x	x	x	x	x		-	x	-
20400300	Zähler freigeben / sperren: CNT_ENABLE()	-	-	-	-	-	-	-	-	x	x	x	x	x = 0 : Zähler sperren x = 1 : Zähler freigeben	x	x	x
20400304	Umschalten zwischen Zählereingängen für single-ended oder differentiellen Betrieb (paarweise)	-	-	-	-	-	-	-	-	-	y	x	x	x : Eingänge für Zähler 1+2 y : Eingänge für Zähler 3+4 x,y = 0: single-ended x,y = 1: differentiell	x	x	x
20400310	Zähler löschen: CNT_CLEAR() ^a	-	-	-	-	-	-	-	-	x	x	x	x	x = 0 : kein Einfluss x = 1 : Zähler löschen	x	x	x
20400320	Zähler latches: CNT_LATCH() ^a	-	-	-	-	-	-	-	-	x	x	x	x	x = 0 : keine Einfluss x = 1 : Zähler latches	x	x	x
20400330	Zählereingang wählen: CLR oder LATCH	-	-	-	-	-	-	-	-	x	x	x	x	x = 0 : CLR-Eingang x = 1 : LATCH-Eingang	-	-	x
20400340	Impuls-/Ereigniszähler- oder Pulsbreiten-/Periodendauermessung	-	-	-	-	-	-	-	-	x	x	x	x	x = 0 : externer Takteingang x = 1 : interner Referenztakt (20MHz / 5MHz)	-	-	x
20400350	4-Flankenauswertung / CLK+DIR oder 20MHz / 5MHz Referenztakt	-	-	0	0	-	-	-	-	x	x	x	x	CNT_MODE=0: x=0: 4-Flanken-Auswertung; x=1: CLK+DIR CNT_MODE=1: x=0: 20MHz; x=1: 5MHz	-	-	x
20400370	Zähler: Fehlerregister ^b	div. Bits												Fehler-Bits (CNT_GETSTATUS)	-	-	x

a. Dieses Register wird nach der Durchführung automatisch wieder zurückgesetzt.

b. Dieses Register müssen Sie von Hand wieder zurücksetzen!

A.3 Hardware-Revisionen

Auf der Rückseite des Gold-Systems befindet sich ein Aufkleber mit der Revisionsbezeichnung des Geräts. Die Unterschiede der Revisionsstände sind nachfolgend dargestellt:

Revision	Erst- ausgabe	Änderung zur Vorgänger-Version
A	1998	Erst-Version mit Link-Datenverbindung.
B1	Nov. 2002	Prototyp (firmenintern, keine Auslieferung an Kunden)
B2	Apr. 2003	Datenverbindung zum PC nicht mehr über Link, sondern über Ethernet oder USB. Alle Analogeingänge und Zählereingänge sind nur in der Betriebsart differentiell verfügbar.
B3	Nov. 2003	Zusätzliche TTL-Zählereingänge für die Betriebsart single-ended (alternativ zu den Zählereingängen für die Betriebsart differentiell nutzbar). Neue Option Gold-D mit Sub-D-Buchsen anstelle von BNC-Buchsen.
B4	Dez. 2003	Verschiedene Verbesserungen
B5	März 2004	Verschiedene Verbesserungen Layout-Änderung
B6	Aug. 2004	Verbesserte Ethernet-Schnittstelle (ENET-2) mit höherem Datendurchsatz. Neue Option Gold-CAN mit verschiedenen Kommunikations-Schnittstellen.

A.4 Abbildungsverzeichnis

Abb. 1 – Konzept der <i>ADwin</i> -Systeme	3
Abb. 2 – Funktionsschema des <i>ADwin-Gold</i>	4
Abb. 3 – Stromversorgungsstecker (männlich)	7
Abb. 4 – Übersichtsbild <i>ADwin-Gold</i> (USB-Version)	9
Abb. 5 – Übersichtsbild <i>ADwin-Gold-D</i> (ENET-Version)	10
Abb. 6 – Pin-Belegung der Analogkanäle für Option Gold-D	11
Abb. 7 – Eingangsbeschaltung eines analogen Eingangs	11
Abb. 8 – Nullpunktverschiebung bei Standardeinstellung bipolar 10 Volt	13
Abb. 9 – Pin-Belegung bei der Konfiguration mit <code>CONF_DIO(12)</code>	15
Abb. 10 – Pin-Belegung der <i>DA</i> -Erweiterung	23
Abb. 11 – Schema der <i>Gold-CO1</i> -Zählererweiterung	24
Abb. 12 – Pin-Belegung der <i>CO1</i> -Erweiterung	25
Abb. 13 – Pin-Belegung Zähler-Spannungsversorgung (Gold-D)	26
Abb. 14 – Befehle der <i>Gold-CO1</i> -Zählererweiterung	26
Abb. 15 – Zahlenkreis als Interpretation von Zählerwerten	27
Abb. 16 – Schema <i>CO1</i> -Erweiterung im Modus „Takt und Richtung“	28
Abb. 17 – Schema <i>CO1</i> -Erweiterung im Modus „4-Flanken-Auswertung“	29
Abb. 18 – Schema <i>CO1</i> -Erweiterung im Modus „Periodendauermessung“	30
Abb. 19 – Schema <i>CO1</i> -Erweiterung im Modus „Impuls-/Pausenzeitmessung“	31
Abb. 20 – Hardware-Adressen der <i>CO1</i> -Zählererweiterung	32
Abb. 21 – Pin-Belegung SSI-Decoder	34
Abb. 22 – Listing: Konvertierung von Gray- in Binär-Code	35
Abb. 23 – CAN: Pinbelegungen	36
Abb. 24 – CAN: Gängige Baudraten einstellen	39
Abb. 25 – RS-xxx: Gängige Baudraten	41

A.5 Index

A

ADwin-System booten, *siehe* Booten
Ausgänge
 analog, Spannungsbereich · 12
Ausgänge digital · 14

B

Booten
 aus *ADbasic* · 8
 Automatisch · 45
Busfrequenz CAN *siehe* CAN-Bus

C

CAN-Bus
 Busfrequenz berechnen · 39
 Event · 39
 Globale Maske · 38

E

Eingänge
 analog, Spannungsbereich · 12
 offene · 9
Eingänge digital · 14
Encoder · 29
Erdung · 6
Event
 CAN-Bus · 39
 Eingangswiderstand · 9
 steigende Flanke · 14

F

Funktionsschema · 4

I

Impulsbreiten-Messung · 30
Innenwiderstand
 Spannungsquelle · 12
Installation
 Inbetriebnahme Hardware · 7
 Reihenfolge · 7
 Start · 1

K

Kalibrierung · 19

L

LSB · IV

M

Multiplexer · 11

N

Nicht-Linearität · 14

P

Prinzipschaltung, *siehe* Funktions-
schema

S

Schirmung · 6
Spannungsbereich · 12
Spannungsversorgung · 7
Stromversorgungs-Stecker · 7

U

Umrechnung
 Digit in Spannung · 14

V

Verstärkungsfaktor k_V · 13
Vierflanken-Auswertung · 29

Z

Zähler
 Betriebsarten · 24
 Gold-CO1 · 24
 Impulsbreiten-Messung · 30
 Inhalt auswerten · 27
 Konfigurieren · 27
 Vierflanken-Auswertung · 29