JÄGER
Computergesteuerte
Messtechnik GmbH

ADwin-Pro II

System and hardware description

ADwin-Pro II Hardware, manual version 1.72, September 2010

For any questions, please don't hesitate to contact us:

Hotline:	+49 6251 96320
Fax:	+49 6251 56819
E-Mail:	info@ADwin.de
Internet	www.ADwin.de

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

Table of contents

Typographical Conventions

"Warning" stands for information, which indicate damages of hardware or software, test setup or injury to persons caused by incorrect handling.

You find a "note" next to

– information, which absolutely have to be considered in order to guarantee an error free operation.

– advice for efficient operation.

"Information" refers to further information in this documentation or to other sources such as manuals, data sheets, literature, etc.

`<C:\ADwin\ …>`

File names and paths are placed in <angle brackets> and characterized in the font `Courier New`.

`Program text`

Program instructions and user inputs are characterized by the font `Courier New`.

`Var_1`

ADbasic source code elements such as instructions, variables, comments and other text are characterized by the font `Courier New` and are printed in color (see also the editor of the *ADbasic* development environment).

Bits in data (here: 16 bit) are referred to as follows:

Bit No.	15	14	13	…	01	00
Bit value	2^{15}	2^{14}	2^{13}	…	$2^1=2$	$2^0=1$
Synonym	MSB	-	-	-	-	LSB

1 The *ADwin-Pro II* system

The *ADwin-Pro II* system is an external processing system with modular expansion options. Depending on applications, the different enclosures can be equipped with classic *ADwin-Pro I* and new *ADwin-Pro II* modules.

When the *ADwin-Pro II* system was developed great attention was paid to the electromagnetic compatibility. The *ADwin-Pro II* system and all available input and output modules have the CE sign and can therefore be configured differently later if necessary.

Applicable modules

Each *ADwin-Pro II* system needs a processor module. It communicates via Ethernet with the PC or notebook.

In order to meet the various requirements for measurement and control tasks the system can be equipped with the following modules:

- analog input modules and analog output modules
- digital input modules and digital output modules
- counters
- filters, isolation amplifiers
- amplifiers for thermocouples and PT100 resistors
- serial communication interfaces (CAN, RSxxx, Fieldbus)
- storage / read module for PCMCIA storage media

Revision Identifier

All modules have a revision identifier written on the module front, e.g. Rev. A2, Rev. B3, Rev. C3. Earlier delivered modules have no identifier; they are to be considered as revision "Rev. A". *ADwin-Pro II* modules have the revision identifier Rev. E1 or higher.

Different revision characters mean different module properties and are described separately.

The revision identifier is followed by a minor counting number, which is mainly used for internal purposes of Jaeger Computergesteuerte Messtechnik GmbH.

2 How to Install an *ADwin-Pro II* System

Please keep strictly to the following order:

1. Start with the manual "*ADwin* installation":
   - Install software and interface drivers from the *ADwin*-CDROM.
   - Initialize the data connection from PC to *ADwin* system and do an operational test.
     The power connectors are described in chapter 4 "Enclosures for the ADwin-Pro II System".
   - Follow the notes in chapter 3 "Operating Environment".
2. Set module addresses, see chapter 5.1 on page 11.
3. Do your first steps with the *ADbasic* Tutorial.
4. Programming in *ADbasic*:

   The *ADbasic* manual describes the real-time development environment, the structure of an *ADbasic* program and gives hints for optimizations.

   The *ADbasic* instructions are described in the online help of the development environment or in these documents:
   - *ADbasic* manual: Basic instructions for calculation, program structure and process control.
   - *ADwin-Pro II* software manual: Instructions and hints for accessing the Pro modules.

For operation, please pay attention to the notes in this manual concerning the respective modules.

Please note:

For *ADwin* systems to function correctly, adhere strictly to the information provided in this documentation and in other mentioned manuals.

Qualified personnel

Programming, start-up and operation, as well as the modification of program parameters must be performed only by appropriately qualified personnel.

Qualified personnel are persons who, due to their education, experience and training as well as their knowledge of applicable technical standards, guidelines, accident prevention regulations and operating conditions, have been authorized by a quality assurance representative at the site to perform the necessary acivities, while recognizing and avoiding any possible dangers.
(Definition of qualified personnel as per VDE 105 and ICE 364).

Availability of the documents

This product documentation and all documents referred to, have always to be available and to be strictly observed. For damages caused by disregarding the information in this documentation or in all other additional documentations, no liability is assumed by the company *Jäger Computergesteuerte Messtechnik GmbH*, Lorsch, Germany.

Legal information

OEM products are mentioned without referring to possible patent rights, the existence of which, may not be excluded.

Subject to change.

Hotline address: see inner side of cover page.

3 Operating Environment

Earth protection

The *ADwin-Pro II* device must be earth-protected, in order to

– build a ground reference point for the electronic

– conduct interferences to earth.

Connect the GND clamp / plug via a short low-impedance solid-type cable to the central earth connection point of the controlled system. The GND plug is internally connected with ground and the enclosure.

Galvanic connection

In the Ethernet cable the data lines are galvanically isolated, but the ground potentials are connected, because the shielding of the Ethernet connector (RJ-45) is connected to GND.

Excluding transient currents

Transient currents, which are conducted via the aluminum enclosure or the shielding, have an influence on the measurement signal.
Please, make sure that the shielding is not reduced, for instance by taking measures for bleeding off interferences, such as connecting the shielding to the enclosure just before entering it. The more frequently you earth the shielding on its way to the controlled system the better the shielding will be.

Use cables with shielding on both ends for signal lines. Here too, you should reduce the bleeding off of interferences via the enclosure by using screen clips.

Supply voltage

Operate the device with the defined and fitting supply voltage. For operation with an external power supply, the instructions of the manufacturer apply. Close the device for operation, use cover plates to cover gaps between built-in modules.

Ambient atmosphere

ADwin-Pro II is designed for operation in dry rooms with an ambient temperature of +5°C … +50°C and a relative humidity of 0 … 80% (no condensation). The device may be operated in a control cabinet or mobile (e.g. in a car).

Chassis temperature

The temperature of the chassis (surface) must not exceed +60°C, even under extreme operating conditions – e.g. in a control cabinet or if the system is exposed to the sun for a longer period of time. You risk damages at the device or not-defined data (values) are output which can cause damages at your measurement device under unfavorable circumstances.

For use in a control cabinet, please note:

– The device shall not pe placed above strong heat sources, e.g. a high power transformer.

– Ventilation inside the control cabinet towards and from the *ADwin-Pro II* device mus be provided.
Especially, the ventilation slots of the device must be kept free, so that the device can lead off its generated heat completely.

4 Enclosures for the *ADwin-Pro II* System

The different sizes for the enclosures depend on the number of slots and the kind of power supply.

Enclosure	Number of Slots	Power supply	
ADwin-Pro II	16	100V…240V	AC
ADwin-Pro II-DC	16	10V…35V	DC
ADwin-Pro II-BM	15	100V…240V	AC
ADwin-Pro II-light	7	100V…240V	AC
ADwin-Pro II-light-DC	7	10V…35V	DC
ADwin-Pro II-mini	5	10V…36V	DC

The number of slots is given for Pro II modules. If Pro I modules be used – in combination with Pro II modules or not – less modules fit into the enclosure.

For the slot area (including power supply slot) the following dimensions apply:

1 HP = 1/5 inch = 5.08mm   1 U = 5/3 inch = 42.3mm

The slots mostly have a width of 5 HP = 1 inch.

Plug-in a module

You plug-in a module into the enclosre like this:

– Switch off the *ADwin* device! A module may sustain damage if you plug it in or out with the power supply switched on.

– Remove one or more cover plates at the wanted position, until the bearings be seen at the left edge: one upper and one lower bearing.
  - Pay attention to the color of the bearings. There are different, offset bearings for Pro I and Pro II modules:
    White bearings: Pro II modules.
    Black bearings: Pro II modules.
  - The processor module has a fixed position, no other position can be used.

– Insert the board carefully into both bearings, plug ahead. If positioned correctly the module cannot be skewed.

– Push the module into the enclosure. At the end the push gets harder while the module plug slides into the female connector of the back plane.

  The module's front panel should butt against the enclosure.

– Fix the module with the screws at top and bottom of the front panel.

– If there are, close the gaps between plugged-in modules using the cover plates. There are plates with 2, 3 or 5 HP width.

4.1 *ADwin-Pro II*

16 slots

The standard enclosure for the *ADwin-Pro II* systems. The backplane of the enclosure connects the processor module with the other modules.

The system fuse is located in a slot in the power supply unit above the female connector for the power supply cable (rear of the enclosure).

Number of Slots	16
Main dimensions (l x w x h) Slot area (w x h)	336 mm × 447.5 mm × 146 mm 84 HP × 3 U
Power supply unit	min. 70 W, 100 V…240 VAC at 50/60 Hz switching power supply
Fuse	5 A, delayed-action fuse

Fig. 1 – Enclosure *ADwin-Pro II*: Specification

At the rear of the enclosure, above the power supply connector you will find a label with the revision number:

Revision	Release	Previous versions
E1	Jun. 2006	*ADwin-Pro II*: New enclosure design and new back plane with Pro I and Pro II bus.

The Pro II enclosure is designed for both Pro I and Pro II modules: The back plane comprises the Pro I bus as well as the Pro II bus. The processor module runs both buses in parallel.

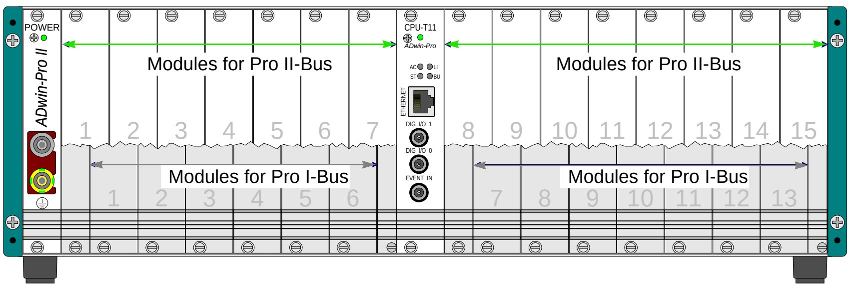


Fig. 2 – Enclosure *ADwin-Pro II*

Please note that modules for Pro I bus (grey in fig. 2) and for Pro II bus have different plug-in positions. You recognize the right position easily by the color of the bearings:

– White bearings: Modules for Pro II bus.

– Black bearings: Modules for Pro I bus.

  Output modules Pro-AOut-x with Rev. A may not be used for technical reasons.

The processor module must be plugged-in at the middle position (white bearings).

There is a gap of half a slot between processor module and Pro I modules (cover plates accompanied), while Pro II modules fit directly besides the processor module.

Pro II-DC with 16 slots

4.2 *ADwin-Pro II-DC*

The *ADwin-Pro II-DC* enclosure is similar to the standard enclosure *ADwin-Pro II*, but is equipped with a DC power supply.

If a current-limited power supply unit is used, it should be able to supply a multiple of the idle current during power-up to maintain proper performance of the system.

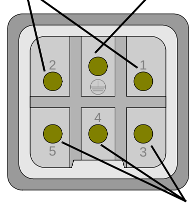


Fig. 3 – Enclosure *ADwin-Pro II-DC*:
Detailed view of the pin assignment

Number of Slots	16
Main dimensions (l x w x h) Slot area (w x h)	336mm × 447.5mm × 146mm 84 HP × 3 U
Power supply unit	min. 80W, DC-DC converter 10V…35V

Fig. 4 – Enclosure *ADwin-Pro II-DC*: Specification

At the rear of the enclosure, above the power supply connector you will find a label with the revision number:

Revision	Release	Änderung zur Vorgänger-Version
E1	Jun. 2005	*ADwin-Pro II*: New enclosure design and new back plane with Pro I and Pro II bus. New power supply female connector.

4.3 *ADwin-Pro II-BM*

Pro II "backmounted" with 15 slots

In the version "backmounted" of the standard enclosure, the modules are plugged-in at the rear of the enclosure.

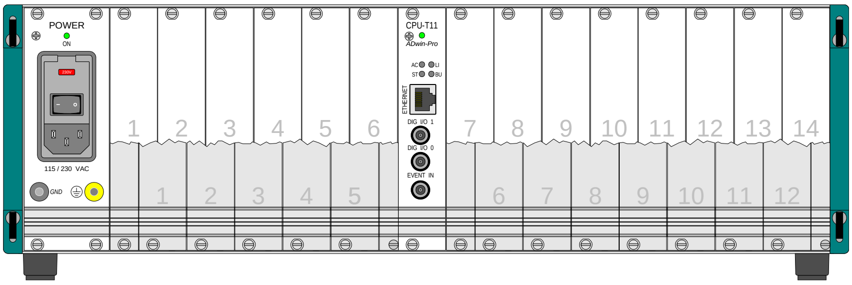


Fig. 5 – Enclosure of *ADwin-Pro II-BM* (rear panel)

The system fuse is located in a slot in the power supply unit above the female connector for the power supply cable (rear of the enclosure).

Number of Slots	16
Main dimensions (l x w x h) Slot area (w x h)	336 mm × 447.5 mm × 146 mm 84 HP × 3 U
Power supply unit	min. 70 W, 100 V…240 VAC at 50/60 Hz switching power supply
Fuse	5 A, delayed-action fuse

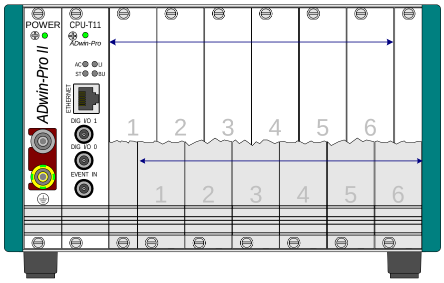
Fig. 6 – Enclosure of the *ADwin-Pro II*-BM: Specification

At the rear of the enclosure, above the power supply connector you will find a label with the revision number:

Revision	Release	Previous versions
E1	Jun. 2006	*ADwin-Pro II*: New enclosure design and new back plane with Pro I and Pro II bus.

Pro II-light with 7 slots

4.4 *ADwin-Pro II-light*



Enclosure *ADwin-Pro II-light*

The backplane of the enclosure connects the processor module with the other modules.

Number of Slots	7
Main dimensions (l x w x h) Slot area (w x h)	336 mm × 234 mm × 146 mm 42 HP × 3 U
Power supply unit	min. 40 W, 100 …240 VAC at 50/60 Hz switching power supply
Fuse	2 A, delayed-action fuse

Fig. 7 – Enclosure *ADwin-Pro II-light*: Specification

At the rear of the enclosure, above the power supply connector you will find a label with the revision number:

Revision	Release	Previous versions
E1	Jun. 2006	*ADwin-Pro II*: New enclosure design and new back plane with Pro I and Pro II bus.

4.5 *ADwin-Pro II-light-DC*

The enclosure *ADwin-Pro II-light-DC* is similar to *ADwin-Pro II-light*, but is equipped with a DC power supply.

Pro II-light-DC mit 7 Steckplätzen

If a current-limited power supply unit is used, it should be able to supply a multiple of the idle current during power-up to maintain proper performance of the system.

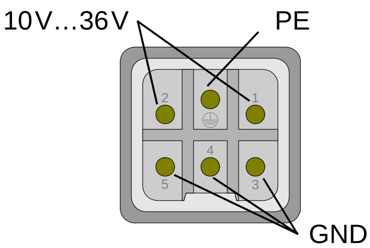


Fig. 8 – Enclosure *ADwin-Pro II-light-DC*:
Connector for power supply

Number of Slots	7
Main dimensions (l x w x h) Slot area (w x h)	336mm × 234mm × 146mm 42 HP × 3 U
Power supply unit	min. 80W, DC-DC converter 10V…35V
Fuse	2A, delayed-action fuse

Fig. 9 – Enclosure *ADwin-Pro II-light-DC*: Specification

At the rear of the enclosure, above the power supply connector you will find a label with the revision number:

Revision	Release	Previous versions
E1	Oct. 2006	First version, *ADwin-Pro II* only.

The enclosure is delivered with a power supply female connector. The plug may be ordered as follows:

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

Connectors: Plug enclosure Harting, series HA.3.XX.X,
Order no. HA.3.STO.1.11

female connector inset Harting, 5+PE, 400V, 16A,
Order no. HE.Q.5.BU.C

4 pcs. female connector contact pin 2.5mm^2, gold- plated,
Order no. HE-HA.C.BU.2,5.AU

4.6 *ADwin-Pro II-mini*

Pro II-mini with 5 slots

The smallest *ADwin-Pro II-mini* enclosure has 5 slots and requires an external power supply unit. The power supply connector is located at the rear of the enclosure.

Number of Slots	5
Main dimensions (l x w x h) Slot area (w x h)	253mm × 147.3mm × 146mm 20 HP × 3 U
External power supply unit	external power supply unit required: 10V…36V DC, >50W

Fig. 10 – Enclosure *ADwin-Pro-mini* Specification

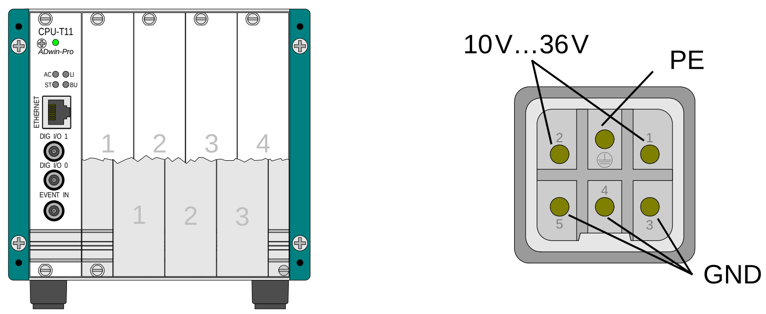


Fig. 11 – Enclosure *ADwin-Pro II-mini*
and power supply connector

At the rear of the enclosure you will find a label with the revision number:

Revision	Release	Previous versions
E1	Dec. 2006	*ADwin-Pro II*: New enclosure design and new back plane with Pro I and Pro II bus. Variable input voltage.

5 *ADwin-Pro* Modules

An *ADwin-Pro* module needs one slot (5 HP) in an *ADwin-Pro* system, some modules need 2 slots.

All technical data of the module refer to a device which is powered-up.

For pluggin-in a module into the enclosure please note the description on page 4, especially with *ADwin-Pro II* enclosures.

5.1 Setting the module's addresses

Any *ADwin-Pro* module (except CPU modules) is addressed in an *ADbasic* program via its module address. The module address is free selectable.

Selecting a module's address

Note the following rules for selecting a module's address:

– A module address mut be unique inside its module group.

  Each module is member of a module group:
  - Pro I modules, functional group CPU: processor modules.
  - Pro I modules, functional group ADC: analog input modules.
  - Pro I modules, functional group DAC: analog output modules.
  - Pro I modules, functional group DIO: digital input/output modules, relays and counter modules.
  - Pro I modules, functional group EXT: special modules of all kind.
  - All Pro II modules.

– A module address must be within the following limits:
  - Pro I modules: 1 … 255.
  - Pro II modules: 1 … 15.

  There are special limits for RSxxx- and fieldbus modules (see below)

It is true that you can select the same module address for modules of different groups. Nevertheless we recommend to use unique adresses in order to prevent a mix-up.

Setting the module's adress: Pro II modules

With Pro II modules you set the module address with the program *ADpro*. Settin the address also resets the Pro modules to initial state.

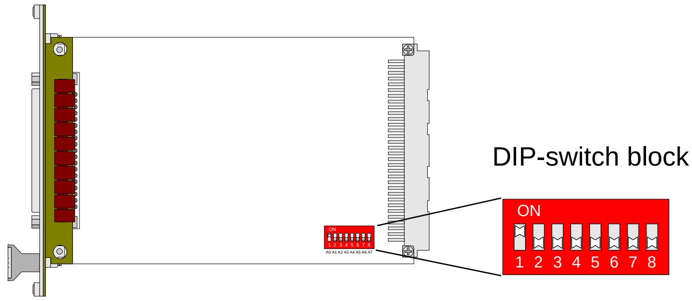
Switch off the power supply of the *ADwin* system and insert the Pro II modules into the casing; please note the description on page 4. Power up the *ADwin* system again.

Afterwards you start the program *ADpro* from the Windows Start menu under `Programs` ▸ `ADwin`. Select the module address with the menu entry `Edit` ▸ `Set module adresses`.

Setting the module's adress: Pro I modules

With Pro I modules you set the module address manually via DIP-switches. The on-board block of DIP-switches is located right to the bottom.

Wit 8 DIP switches the address is selectable between 1 and 255 (see fig. 12). Each module of the same group needs to have a different address.



Module no.	Settings of DIP switches							
	1	2	3	4	5	6	7	8
1	1	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0
3	1	1	0	0	0	0	0	0
4	0	0	1	0	0	0	0	0
5	1	0	1	0	0	0	0	0
6	0	1	1	0	0	0	0	0
7	1	1	1	0	0	0	0	0
8	0	0	0	1	0	0	0	0
…	…							
254	0	1	1	1	1	1	1	1
255	1	1	1	1	1	1	1	1

Fig. 12 – Address settings of the *ADwin-Pro* modules with DIP switches

Please note:

– A RSxxx module with 4 serial interfaces uses 2 addresses (group EXT): the set address and the following address.

– A fieldbus module uses 32 addresses (group EXT); the address allocation is shown in fig. 13.

Set module address	Addit. allocated addresses	Settings of DIP switches							
		1	2	3	4	5	6	7	8
1	160…191	1	0	0	0	0	0	0	0
2	192…223	0	1	0	0	0	0	0	0
3	224…255	1	1	0	0	0	0	0	0
4	128…159	0	0	1	0	0	0	0	0

Fig. 13 – Address settings of fieldbus modules with DIP switches

5.2 Processor modules

For each *ADwin-Pro* system one processor module is required. This processor module is the center of a Pro system and performs the following tasks:

– Communication with PC or laptop.

  The data connection is established via USB or Ethernet; former version used a serial link connection.

– Communication with other *ADwin-Pro* modules via internal Pro bus.

– Communication with possibly existing *TiCo* processors on *Pro II* modules via internal Pro bus.

– Start and run the user defined processes.

The processor module provides the memory for data and programs, divided into a fast internal memory (SRAM) and an external memory (DRAM).

At the time, the prcessor module Pro-CPU-T11 is available for *ADwin-Pro II*.

5.2.1 Pro-CPU-T11

The processor modul can only be run in a Pro II casing and works both with Pro I and Pro II modules.

The output module Pro-AOut-x runs with T11 up from Rev. B.

To be used for Pro system	Pro II
Processor	ADSP TS101S
Clock rate	300 MHz
Calculation resolution for float values	40 Bit
Data connection	Ethernet
Internal memory	768 KiB
External memory	256 MiB
TTL-signal inputs	Event In, with 4,7 kΩ pull-down resistor Dig I/O 0, with 4,7 kΩ pull-up resistor Dig I/O 1, with 4,7 kΩ pull-up resistor

Fig. 14 – Pro-CPU-T11: Specifikation

The processor module has a fixed position in the casing. Please see the notes on how to Plug-in a module on page 4.

The internal memory of the processor is divided into program memory (PM), data memory (DM) and free-for-use extra memory (EM). Each memory section has a size of 256 KiB.

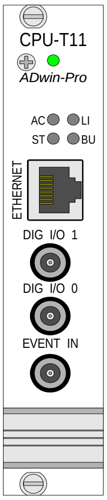


Fig. 15 – Pro-CPU-T11: Front panel

The processor module show the mode of operation by LEDs at the Ethernet connector. The meaning of the LEDs is described in manual ADwin Installation, chapter 10.3.

Input Event In

The external trigger input (`Event In`) enables the processor module to recognize an external signal as trigger for an event and trigger a process, that is processed immediately and completely (see *ADbasic* manual, chapter: Structure of the *ADbasic* program).

The event signal has to be present for 50 ns to be recognized.

Alternatively, the event input of an other module may be used. All event signals arrive at the same signal line of the processor module as the input `Event In` does.

Software changes when switching from T9 / T10

Existing *ADbasic* programs may – if still working with Pro I modules – be further used with few, but inevitable changes:

– The processor T11 needs the include file <`ADwinPro_All.inc`> to be included. In parallel all other include files for Pro modules should be deleted from the program.

PROCESSDELAY

– The time unit of the PROCESSDELAY (cycle time) is $3.\overline{3}$ ns for both high priority and low priority processes.

  All referring values and calculations must be adapted to the new time unit. The greatest possible Processdelay corresponds to 7.1 s; a greater cycle time can be achieved using an auxiliary variable.

READ_TIMER

– The time unit of $3.\overline{3}$ ns is also true for the internal counter, i.e. counter queries with **`Read_Timer`** must be adapted, too.

  Please note: The process timing in connection with I/O instructions has become more complex (see below). Thus, a time difference determined with **`Read_Timer`** will now refer to a part of the total process timing only.

SLEEP

– The instruction **`SLEEP`** must be replaced by one of the following new instructions.
  - **`CPU_SLEEP`** causes the processor to wait. The instruction **`SLEEP`** had the same function with the processors T9 and T10.
  - **`P1_SLEEP`** causes the Pro I bus to wait, e.g. to co-ordinate I/O instructions.
  - **`P2_SLEEP`** causes the Pro II bus to wait, e.g. to co-ordinate I/O instructions.

  The new instructions have a time unit of 10 ns (**`SLEEP`**: 100 ns).

  Which instruction is right? Normally **`SLEEP`** is used to bridge the waiting time of an I/O instruction, e.g. the settling time of a multiplexer with **`SET_MUX`**. In this case the instruction **`P1_SLEEP`** fits for previous modules (Pro I bus), and **`P2_SLEEP`** for Pro II modules.

  Please see the notes in the *ADbasic* manual about Using Waiting Times (chapter 6.2.5).

  Why are there new instructions? The processor T11 distinguishes processor instructions on theone hand and I/O instructions on the other hand. The processor architecture enables a quasi-parallel processing[1] of both instruction groups and obtains a much faster processing of *ADbasic* processes. This also means that the instruction groups are (mainly) processed independently in respect to timing. Since the process timing shall be controlled by waiting, there needs to be a separate instruction for each group. The separate instruction for each bus is required, because an I/O wait is effected by halting the appropriate bus.

1. The processor architecture differs from T9 and T10 in this point: T9 and T10 processed instructions of both groups sequentially. Thus, halting the processor with a SLEEP instruction did make the waiting time for subsequent I/O instructions, too.

5.2.2 Pro II-Boot

With Pro II-Boot you have a boot loader expansion which can

- boot an *ADwin-Pro II* system.
- load up to 10 processes.
- start process 10 automatically (if present).
- save data.

Pro II-Boot is an ordering option for processor modules with Ethernet interface. An upgrade is not possible.

Software

By installation of the *ADbasic* and the *ADwin* drivers from the CDROM, all files / programs necessary for the boot loader option have already been copied to the hard disk.

If you use the boot loader, an application, which you have written with a program for visualization of measurement data, must not reboot the *ADwin* system.

5.2.3 The Watchdog

You can monitor your processor module with a watchdog. The watchdog generates a reset, when a signal, generated by a program code, does unexpectedly not arrive (see also "*ADwin-Pro* System Specifications - Programming in *ADbasic*"). This reset sets the digital and analog outputs to those values, which correspond to the configuration after power-up, normally digital 0 or 0 Volt.

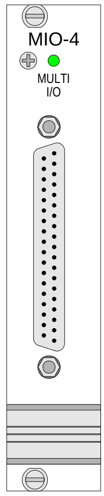
Notes in relation to the Pro II-Flash-Boot:

- Please pay attention to the fact that the watchdog has to be reset every 1.6s, since a longer time interval between two impulses will be interpreted as an error.
- The watchdog can also be used with the boot loader Pro-Flash-Boot, but does not automatically load and start the software.
- Test your programs always with the watchdog switched off. Activate it only when your programs work appropriately!

5.3 Pro II: Multi-IO Modules

Module	Rev.	Type	Chan-nels	Specific Data	Page
MIO-4	E	TTL input / output	8	U_{In} 5V TTL	18
		Analog input	16 s.e. 8 diff.	1 ADC, 18 bit, ±10V Max. conversion time 2µs max. sample 500 ksample/s	
		Analog output	4	4 DAC, 16 Bit, ±10V Settling time 9µs	
		TiCo Processor	–	128 KiByte internal memory 4MiByte external SRAM	
MIO-4-ET1	E	TTL input / output	8	U_{In} 5V TTL	25
		Analog input	16 s.e. 8 diff.	1 ADC, 18 bit, ±10V Max. conversion time 2µs max. sample 500 ksample/s	
		Analog output	4	4 DAC, 16 Bit, ±10V Settling time 9µs	
		Transistor output	4	5…30 V_{DC} (external), 200mA	
		Optocouple input	4 s.e.	selectable 5V, 12V, 24V	
		Counter block	1	Universal, 32 Bit, 5V diff.	
		SSI decoder	1	max. 2MHz	
		EtherCAT interface	1	Slave interface	
		TiCo Processor	–	128 KiByte internal memory 4MiByte external SRAM	

5.3.1 Pro II-MIO-4 Rev. E



The module Pro II-MIO-4 Rev. E is equipped with the following hardware:

– 16 analog inputs (16 single-ended or 8 differential), via multiplexer with 18 bit ADC, sample rate up to 500kHz

– 4 analog outputs with 16 Bit DAC.

– 8 digital input / output channels with TTL levels

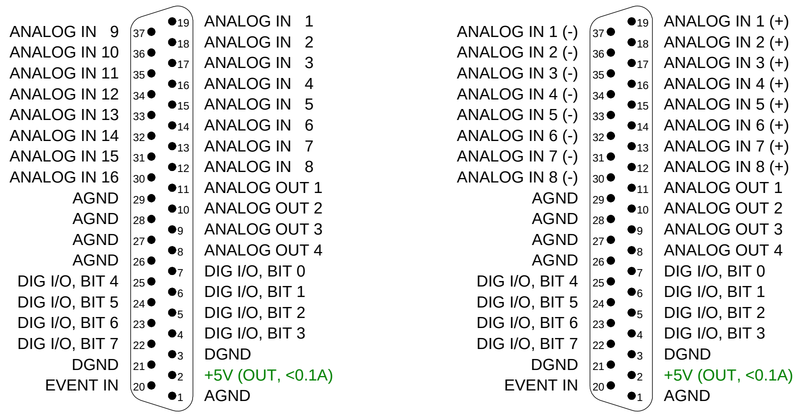
– 1 Event input

– *TiCo* processor with 128 KiByte internal memory and 4MiB external SRAM memory

  The *TiCo* processor has access to all inputs and outputs of the module. For more information about usage and programming of the *TiCo* processor please see the manual *TiCoBasic*.

If the *TiCo* bootloader is programmed, the module can work on its own and independently from the CPU moduleof the *ADwin-Pro II* system.

The extended module version Pro II-MIO-4-ET1 Rev. E is described starting on page 25.



Analog inputs single-ended Analog outputs differential

Fig. 16 – Pro II-MIO-4 Rev. E: Pin assignment

The module functions are described in the following sections:

– Analog Inputs

– Analog Outputs

– Digital Inputs / Outputs

– TiCo processor

– Technical Specification

– Programming

Analog Inputs

The module Pro II-MIO-4 Rev. E has 16 single ended inputs or 8 differential inputs (selectable by software). After power-up, the module is set to 8 differential inputs.

The inputs are equipped with a 37-pin D-Sub female connector; for pin assignment see fig. 16.

The inputs are connected via a multiplexer to the ADC. The ADC has a resolution of 18 bit and can run with a sampling rate of up to 500kSamples/s.

The module has an input voltage range of ±10V and a software selectable gain of 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

The module includes a sequential control, which can read measurement values from several or all input channels sequentially.

The module can monitor each input channel, if an upper or a lower limit–you can set the limits for each input channel separately–has been exceeded.

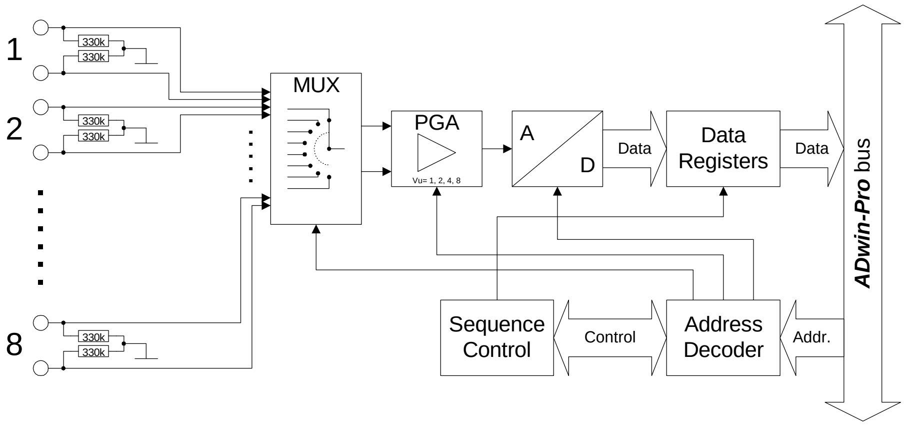


Fig. 17 – Pro II-MIO-4 Rev. E: Block diagram of analog inputs

Analog Outputs

The module Pro II-MIO-4 Rev. E has 4 analog outputs each with a 16 bit DAC. Each output has a fixed 1st order low-pass filters (fc = 10MHz) to suppress noise.

The output voltage range of the DACs is set to ±10V bipolar and can't be changed. Offset and gain are adjusted by software (see chapter 6 "Calibration").

The outputs are available on a 37-pin D-Sub female connector; for pin assignment see fig. 16.

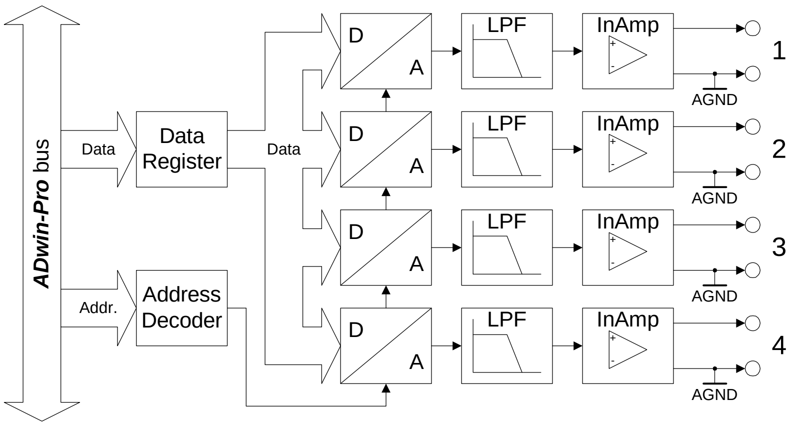


Fig. 18 – Pro II-MIO-4 Rev. E: Block diagram of analog outputs

Digital Inputs / Outputs

The digital input/output module Pro II-MIO-4 Rev. E provides 8 programmable digital input and output channels with TTL levels. The channels can be configured as blocks of 4 as inputs or outputs by *ADbasic* instructions. The channels are configured as inputs after power up.

Via the trigger input EVENT a signal can trigger a process, which will be processed at once and completely (see *ADbasic* manual).

The digital channels are available on a 37-pin D-Sub female connector; for pin assignment see fig. 16.

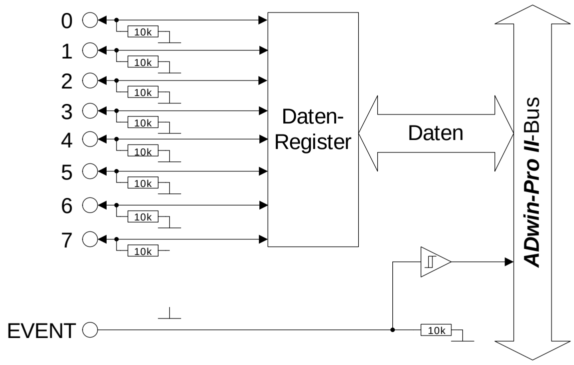


Fig. 19 – Pro II-MIO-4 Rev. E: Block diagram of digital channels

TiCo processor

The module provides the freely programmable *TiCo* processor with 128 kiB internal memory and 4 MiB external SRAM memory. The internal memory serves as data and program memory. You prgram the *TiCo* processor with *TiCoBasic*.

The *TiCo* processor has access to all analog and digital input and output channels, as does the *ADwin* CPU. Find more information about use and programming of the *TiCo* processor in the *TiCoBasic* manual.

If you store a *TiCoBasic* program in the *TiCo* bootloader, the program is automatically loaded into the *TiCo* processor and started on power-up. Thus, the module can run on its own and independently from the CPU module of the *ADwin-Pro II* system.

Technical Specification

Analog Inputs	
Input channels	16 single ended / 8 differential. via multiplexer
Resolution	18 Bit
Conversion time	max. 2µs
Sampling rate	max. 500ksps
Multiplexer settling time	5µs
Measurement range	±10V
Gain	1, 2, 4, 8 selectable by software
Accuracy INL	typical ±4 LSB
Accuracy DNL	max. ±1 LSB
Input resistance	330kΩ. ±2%
Input over-voltage	±20V
Offset error	adjustable
Offset drift	±30ppm/ºC
Analog Outputs	
Output channels	4
Resolution	16 Bit
Settling time	9µs (to 0.01% FSR)
Output voltage	±10V
Output current max.	±5mA per channel for optimal function
Accuracy INL	±2 LSB typical
Accuracy DNL	±1 LSB typical
Offset error	adjustable
Gain error	adjustable
Digital Inputs / Outputs	
Digital inputs / outputs	8 channel with TTL logic, configurable in groups of 4 channels
Pull down resistor	10kΩ
V_{IH}	min. 2V
V_{IL}	max. 0.8V
I_{IH}	max. 1µA
I_{IL}	max. 0.01mA
Voltage range	-0.5V … +5.5V
Output current	max. ±24mA per channel, max. ±50mA per block (4 channels) via V_{CC} or GND
Event input	TTL Logic
Power up status	All channels as inputs
General	
Memory size (TiCo)	128KiByte internal, 4MiByte external SRAM
Connector	37-pin D-Sub female connector

Fig. 20 – Pro II-MIO-4 Rev. E: Specification

Programming in ADbasic

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Range	Instructions
Analog Inputs	
Set inputs to single-ended or differential	`P2_SE_Diff`

Do a single conversion – complete or step by step	`P2_ADC`, `P2_ADC24`
	`P2_Set_Mux`, `P2_Start_Conv`
	`P2_Wait_EOC`
	`P2_Read_ADC`, `P2_Read_ADC24`

Read value and start new conversion	`P2_Read_ADC_SConv` `P2_Read_ADC_SConv24`

Use sequence control	`P2_Seq_Init`, `P2_Seq_Start`
	`P2_Seq_Read`, `P2_Seq_Read24`
	`P2_SEQ_Read_Packed`
	`P2_Seq_Wait`

Control input limits	`P2_ADC_Read_Limit` `P2_ADC_Set_Limit`
Analog Outputs	
Output values	`P2_DAC`, `P2_DAC4`, `P2_DAC4_Packed`

Output values step by step	`P2_Write_DAC`, `P2_Write_DAC4`
	`P2_Write_DAC4_Packed`
	`P2_Write_DAC32`
	`P2_Start_DAC`

Digital Inputs / Outputs	
Configure input/outputs	`P2_MIO_DigProg`
Query input signals	`P2_MIO_Digin_Long`

Use latch register	`P2_MIO_Dig_Latch`
	`P2_MIO_Dig_Read_Latch`
	`P2_MIO_Dig_Write_Latch`

Set and read back output signals	`P2_MIO_Digout`
	`P2_MIO_Digout_Long`
	`P2_MIO_Get_Digout_Long`

General	
Use LED	`P2_Check_LED`, `P2_Set_LED`
Synchronize	`P2_Sync_All, P2_Sync_Enable` `P2_Sync_Stat`
Use interrupt and event inputs	`P2_Event_Enable`, `P2_eVENT_rEAD` `P2_eVENT_cONFIG`

Programming in TiCoBasic

The module can be programmed with *TiCoBasic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `MIO_TiCo.inc` contains instructions for the functions:

Range	Instructions
Analog Inputs	
Set inputs to single-ended or differential	**SE_Diff**
Do a single conversion – complete or step by step	**ADC**, **ADC24** **Set_Mux**, **Start_Conv**, **Wait_EOC** **Read_ADC**, **Read_ADC24**
Read value and start new conversion	**Read_ADC_SConv** **Read_ADC_SConv24**
Use sequence control	**Seq_Init**, **Seq_Start** **Seq_Read**, **Seq_Read24** **Seq_Wait**
Control input limits	**ADC_Read_Limit** **ADC_Set_Limit**
Analog Outputs	
Output values	**DAC**
Output values step by step	**Write_DAC**, **Write_DAC32** **Start_DAC**
Digital Inputs / Outputs	
Configure input/outputs	**MIO_DigProg**
Query input signals	**MIO_Digin_Long**
Use latch register	**MIO_Dig_Latch** **MIO_Dig_Read_Latch** **MIO_Dig_Write_Latch**
Set and read back output signals	**MIO_Digout** **MIO_Digout_Long** **MIO_Get_Digout_Long**
General	
Use LED	**Check_LED**, **Set_LED**
Use interrupt and event inputs	**Event_Enable**, **Event_Eead** **Event_Config**, **Trigger_Event**

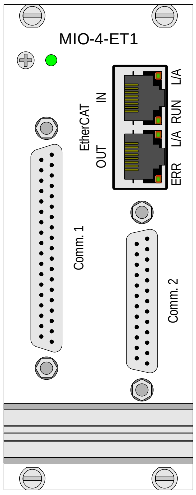
Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions
Data exchange with the *TiCo* processor using global variables and arrays	**P2_TDrv_Init** **P2_GetData_Long**, **P2_Get_Par**, **P2_Get_Par_Block** **P2_SetData_Long**, **P2_Set_Par**, **P2_Set_Par_Block** **P2_Get_TiCo_RingBuffer**, **P2_Set_TiCo_RingBuffer** **P2_RingBuffer_Empty** **P2_RingBuffer_Full**

Function	Instructions
Control *TiCo* processor	**P2_TiCo_Reset**, **P2_TiCo_Start**, **P2_TiCo_Stop** **P2_Get_TiCo_Bootloader_Status** **P2_Get_TiCo_Status**, **P2_Workload**
Control *TiCo* processes	**P2_Process_Status** **P2_TiCo_Get_Processdelay** **P2_TiCo_Set_Processdelay** **P2_TiCo_Start_Process** **P2_Tico_Stop_Process**
Transfer *TiCo* programs	**P2_TiCo_Flash**, **P2_TiCo_Load**

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



The module Pro II-MIO-4-ET1 Rev. E is equipped with the following hardware:

– 16 analog inputs (16 single-ended or 8 differential), via multiplexer with 18 bit ADC, sample rate up to 500kHz
– 4 analog outputs with 16 Bit DAC.
– 8 digital input / output channels with TTL levels
– 1 Event input
– 4 transistor outputs (TRA)
– 4 optically isolated inputs (OPT)
– 1 counter block with two 32-bit counters:
  • one up/down counter with clock/direction or four edge evaluation for connection of encoders.
  • one PWM counter to evaluate high and low times, duty cycle, or frequency.
– 1 SSI decoder to connect an incremental encoder
– 1 Ethercat interface (slave)
– *TiCo* processor with 128 KiByte internal memory and 4MiB external SRAM memory

  The *TiCo* processor has access to all inputs and outputs of the module. For more information about usage and programming of the *TiCo* processor please see the manual *TiCoBasic*.

If the *TiCo* bootloader is programmed, the module can work on its own and independently from the CPU moduleof the *ADwin-Pro II* system.

The module variant Pro II-MIO-4 Rev. E is described starting on page 18.

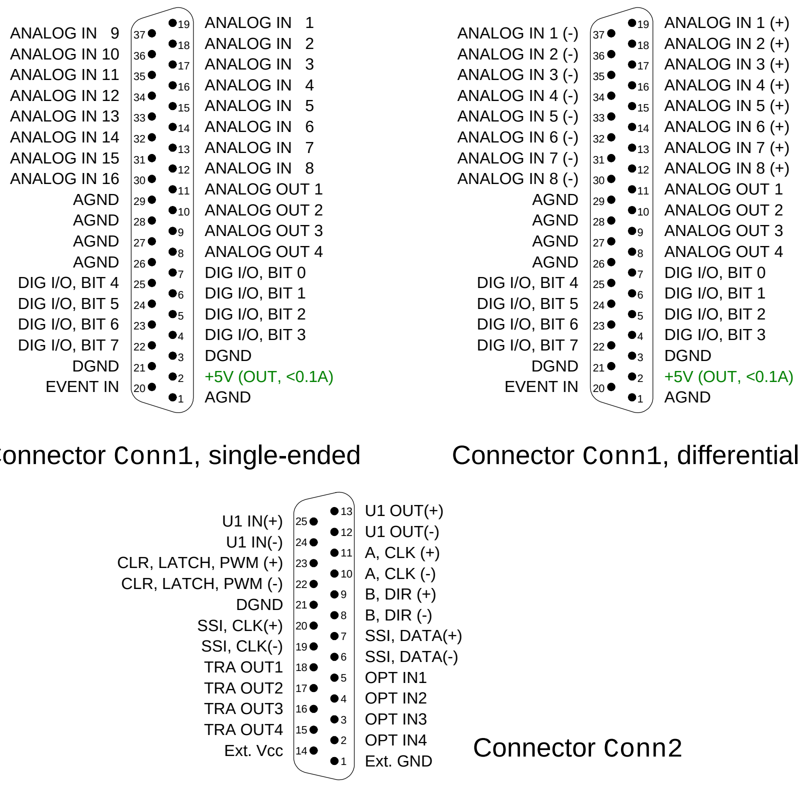


Fig. 21 – Pro II-MIO-4-ET1 Rev. E: Pin assignment

The module functions are described in the following sections:

– Analog Inputs
– Analog Outputs
– Digital Inputs / Outputs
– Transistor Outputs
– Optocouple Inputs
– SSI Decoder
– Counter block
– EtherCAT Interface
– TiCo processor
– Technical Specification
– Programming

Analog Inputs

The module Pro II-MIO-4-ET1 Rev. E has 16 single ended inputs or 8 differential inputs (selectable by software). After power-up, the module is set to 8 differential inputs.

The inputs are equipped with a 37-pin D-Sub female connector; for pin assignment see fig. 16.

The inputs are connected via a multiplexer to the ADC. The ADC has a resolution of 18 bit and can run with a sampling rate of up to 500kSamples/s.

The module has an input voltage range of ±10V and a software selectable gain of 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

The module includes a sequential control, which can read measurement values from several or all input channels sequentially.

The module can monitor each input channel, if an upper or a lower limit–you can set the limits for each input channel separately–has been exceeded.

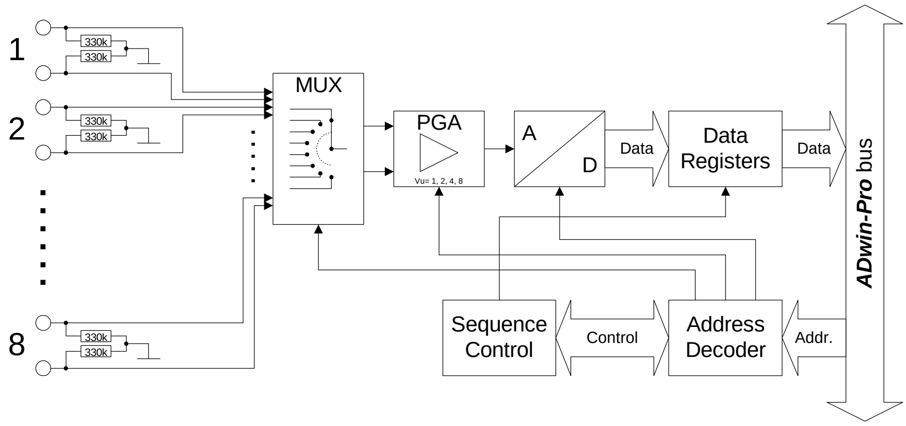


Fig. 22 – Pro II-MIO-4-ET1 Rev. E: Block diagram of analog inputs

Analog Outputs

The module Pro II-MIO-4-ET1 Rev. E has 4 analog outputs each with a 16 bit DAC. Each output has a fixed 1st order low-pass filters (fc = 10MHz) to suppress noise.

The output voltage range of the DACs is set to ±10V bipolar and can't be changed. Offset and gain are adjusted by software (see chapter 6 "Calibration").

The outputs are available on a 37-pin D-Sub female connector; for pin assignment see fig. 16.

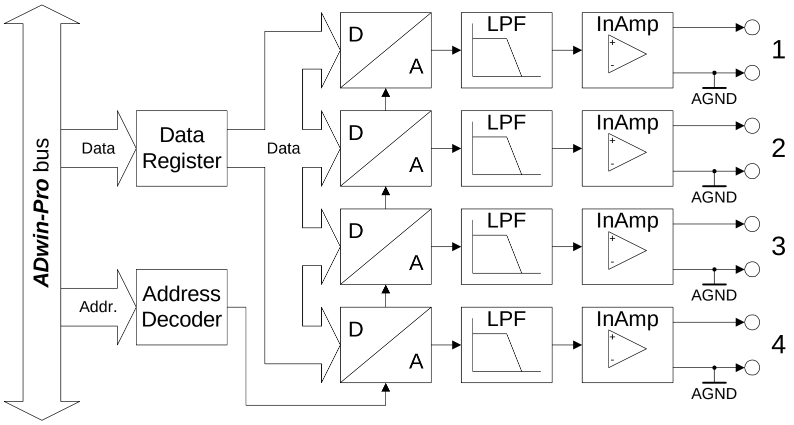


Fig. 23 – Pro II-MIO-4-ET1 Rev. E: Block diagram of analog outputs

Digital Inputs / Outputs

The digital input/output module Pro II-MIO-4-ET1 Rev. E provides 8 programmable digital input and output channels with TTL levels. The channels can be configured as blocks of 4 as inputs or outputs by *ADbasic* instructions. The channels are configured as inputs after power up.

Via the trigger input EVENT a signal can trigger a process, which will be processed at once and completely (see *ADbasic* manual).

The digital channels are available on a 37-pin D-Sub female connector; for pin assignment see fig. 16.

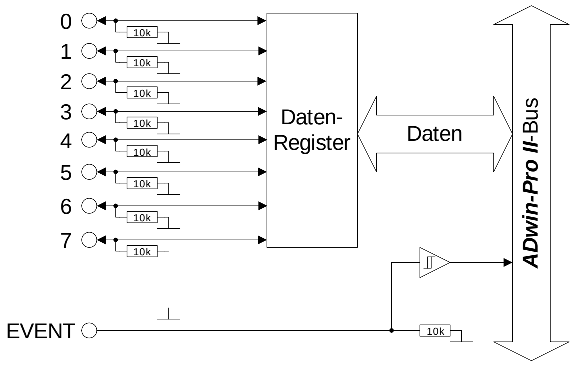


Fig. 24 – Pro II-MIO-4-ET1 Rev. E: Block diagram of digital channels

Transistor Outputs

The module Pro II-MIO-4-ET1 Rev. E provides 4 channels of galvanically isolated transistor outputs. The outputs switch to V_{CC}.

The switching voltage V_{CC} has to be provided by an external power supply.

The channels as well as the event-input are optically isolated from system circuitry.

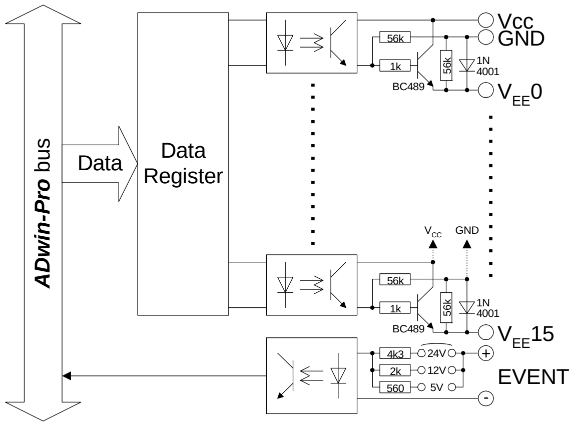


Fig. 25 – Pro II-MIO-4-ET1 Rev. E: Block diagram transistor outputs

The outputs are provided on the 25-pole D-Sub connector Conn2; pin assignment see fig. 21.

Optocouple Inputs

The module Pro II-MIO-4-ET1 Rev. E provides 4 channels of optically isolated digital inputs. The input voltage range can be set by jumpers (5V, 12V, 24V). The default setting of the input voltage range is 24V. The switching time of only 100ns allows the sampling of high-speed digital inputs.

The module can automatically monitor the edges of input channels. With every change the current input levels are saved together with a time stamp in a FIFO. The FIFO data can be read read and processed.

Each channel is optically isolated from the system circuitry and from the other inputs. The event-input is optically isolated from the system as well.

The inputs are provided on the 25-pole D-Sub connector Conn2; pin assignment see fig. 21.

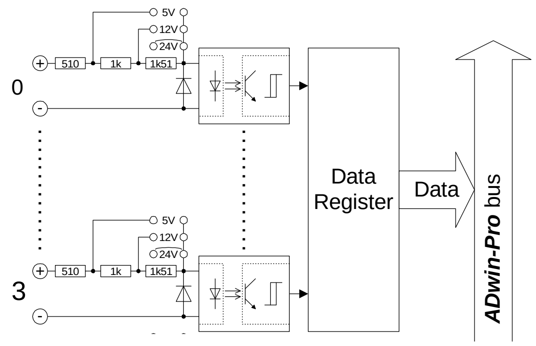


Fig. 26 – Pro II-MIO-4-ET1 Rev. E: Block diagram optocouple inputs

SSI Decoder

An incremental encoder with SSI interface can be connected to the decoder. The signals are differential and have RS422/485 levels.

You can set the clock rate (40kHz … 1MHz) as well as the data resolution (up to 32 bit) of the decoder via software.

A decoder either reads out an individual value (on request) or continously provides the current value.

The decoder inputs are provided on the 25-pole D-Sub connector Conn2; pin assignment see fig. 21.

Counter block

The module Pro II-MIO-4-ET1 Rev. E provides a configurable multi-purpose counter block. The counter block contains two 32-bit counters: First an up/down counter or four edge evaluation for connection of enceoders. Second, a PWM counter to evaluate high and low times, duty cycle, or frequency. Both counters of a block can be operated in parallel.

The counter inputs run differential.

Up/down counter

The up/down counter of a block can be operated in 2 modes:

– Clock / direction (CLK and DIR signals)

  A negative edge at the CLK input is the counting impulse for the 32-bit counter. The DIR signal sets the counting direction, TTL high means a count-up, TTL low means a count-down.

  You can latch the counter values program-controlled or you can influence the counter by an external CLR/LATCH signal.

  Depending on the programming the CLR/LATCH signal has either the effect that the counter values are cleared (CLR) or that the counter values are latched (LATCH). This function will only be effective when it is released by the instructions **P2_CNT_CLEAR_ENABLE** or **P2_CNT_LATCH_ENABLE**.

  The counter is cleared or latched with a rising adge at input CLR/LATCH. During the latch process the frequency of the measurement can be determined by getting the difference of two read latch values, because this difference defines the number of pulses between the two reading processes.

– Four edge evaluation (A and B signals)

  The four edge evaluation changes the signals (which should be 90° phase-shifted) of a connected incremental encoder at the inputs A and B to CLK and DIR signals. For this you have to program the inputs correspondingly (see "*ADwin-Pro* System Description, Programming in *ADbasic*").

  Since every edge of the A and B signals generates a count impulse, the resolution is increased by factor 4. If the encoder has a reference signal, it can be used to clear or latch the counter (after release of the CLR or LATCH input). The counter is cleared when the signals A, B and CLR are on logic "1" (software-selectable: clear, when only the CLR signal is on logic "1").

PWM Counter

The PWM counter of the counter block analyzes the signals at the PWM inputs. Via software instructions the following data can be read directly:

– frequency and duty cycle

– high and low time

The counter inputs are provided on the 25-pole D-Sub connector `Conn2`; pin assignment see fig. 21.

EtherCAT Interface

The module provides a fieldbus node with the functionality of an EtherCAT slave. All settings are done via software.

After power-on you must initialize the fieldbus node in *ADbasic*. The initialization determines the size of the input and output areas.

There is a range each for data input and data output; each range has a maximum size of 254 bytes. Please note, that the terms "input" and "output" are used as the fieldbus controller sees them.

You set the number and length of input and output areas separately.

The interface has a plug connector of type RJ45 for both data input (IN) and data output (OUT). Each connector has a top LED "Link / Activity" (L/A), which displaysd the operating status of the node in the EtherCAT bus. The two other LEDs display the status of the EtherCAT state machine (RUN) and the occurrence of comunication errors (ERR).

LED	Status	Meaning
Link / Activity	off	Offline (or no power).
	green	Fieldbus node online, no data exchange.
	green, flickering	Fieldbus node online, with data exchange.
RUN	off	Status INIT: interface being initialized (or no power).
	blinks green	Status PRE-OP: Interface has contact to bus master.
	flashs green once	Status SAFE-OP: Interface can read data from the bus, but not send.
	green	Status OP: Interface is completely ready, inputs and outputs are active.
	red	Status EXCEPTION.
ERR	off	No error (or no power).
	blinks red	Invalid configuration.
	flashs red once	Local error in the interface; EtherCAT status has been changed.
	flashs red twice	Application watchdog timeout.
	red	Critival communication error.

Fig. 27 – Pro II-MIO-4-ET1 Rev. E: Meaning of EtherCAT LEDs

If both LEDs RUN and ERR turn red, a serious error has occurred in the interface. Please inform the support of Jäger Messtechnik; you find the address on the inner side of the cover page of the manual.

The EtherCAT slave can only be accessed from *TiCoBasic*. You initialize slave using the instruction **MIO_ECAT_Init**. Afterwards, you project the EtherCAT bus with a configuration tool suitable for the bus master, e.g. the program "TwinCAT System Manager" of the Beckhoff company. To do so, you have to copy the description file `ADwin-EtherCAT.xml` of the EtherCAT interface from `C:\ADwin\Fieldbus\EtherCAT\` to the directory of the configuration tool.

TiCo processor

The module provides the freely programmable *TiCo* processor with 128 kiB internal memory and 4 MiB external SRAM memory. The internal memory serves as data and program memory. You program the *TiCo* processor with *TiCoBasic*.

The *TiCo* processor has access to all analog and digital input and output channels, as does the *ADwin* CPU. Find more information about use and programming of the *TiCo* processor in the *TiCoBasic* manual.

If you store a *TiCoBasic* program in the *TiCo* bootloader, the program is automatically loaded into the *TiCo* processor and started on power-up. Thus, the module can run on its own and independently from the CPU module of the *ADwin-Pro II* system.

Technical Specification

Analog Inputs	
Input channels	16 single ended / 8 differential. via multiplexer
Resolution	18 Bit
Conversion time	max. 2µs
Sampling rate	max. 500ksps
Multiplexer settling time	5µs
Measurement range	±10V
Gain	1, 2, 4, 8 selectable by software
Accuracy INL	typical ±4 LSB
Accuracy DNL	max. ±1 LSB
Input resistance	330kΩ. ±2%
Input over-voltage	±20V
Offset error	adjustable
Offset drift	±30ppm/ºC
Analog Outputs	
Output channels	4
Resolution	16 Bit
Settling time	9µs (to 0.01% FSR)
Output voltage	±10V
Output current max.	±5mA per channel for optimal function
Accuracy INL	±2 LSB typical
Accuracy DNL	±1 LSB typical
Offset error	adjustable
Gain error	adjustable
Digital Inputs / Outputs	
Digital inputs / outputs	8 channel with TTL logic, configurable in groups of 4 channels
Pull down resistor	10kΩ
V_{IH}	min. 2V
V_{IL}	max. 0.8V
I_{IH}	max. 1µA
I_{IL}	max. 0.01mA
Voltage range	-0.5V … +5.5V
Output current	max. ±24mA per channel, max. ±50mA per block (4 channels) via V_{CC} or GND
Event input	TTL Logic
Power up status	All channels as inputs
Transistor Outputs	

Fig. 28 – Pro II-MIO-4-ET1 Rev. E: Specification

Output channels	4
Switching voltage V_{CC}	5…30V DC durch externe Spannungsversorgung
Switching current	200mA max. per channel
Voltage drop	0.5V
Switching time	2.5µs
Isolation	42V channel-channel / channel-GND common ground for all OPT and TRA channels
Optocouple Inputs	
Input channels	4
Input current	typ. 3.5mA / max. 7.5mA
Input voltage range (selectable via jumpers)	0…5V / 0…12V / 0…24V
Switching threshold for 0-low	0…0.8V / 0…1.6V / 0…3.2V
Switching threshold for 1-high	4.5…5V / 10…12V / 20…24V
Input over-voltage	-5V … 8V / -5V … 16V / -5V … 30V /
Switching time	100ns
Isolation	42V channel-channel / channel-GND common ground for all OPT and TRA channels
Counter	
Number	1 universal counter block
Counter resolution	32 bit
Reference clock	50MHz
Clock frequency four edge evaluation	12.5MHz max. (at 90° phase-shift of the signals)
Clock frequency up/down counter	15MHz max.
Reference frequency PWM analysis	100MHz
Input / output level	compatible to RS422/485 (5V differential, 120 Ω bus termiinating resistor)
Isolation	none
SSI Decoder	
Number	1
Clock frequency SSI deccoder (CLK)	2MHz max.
Ethercat Interface	
Number	1, to be programmed in *TiCoBasic*
General	
Memory size (TiCo)	128KiByte internal, 4MiByte external SRAM
Connectors	37-pin D-Sub female connector `Conn1` 37-pin D-Sub female connector `Conn2` 2 female connectors type RJ45 for EtherCAT data input (`IN`) and output (`OUT`)

Fig. 28 – Pro II-MIO-4-ET1 Rev. E: Specification

Programming

Programming in ADbasic

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Range	Instructions
Analog Inputs	
Set inputs to single-ended or differential	**P2_SE_Diff**
Do a single conversion – complete or step by step	**P2_ADC**, **P2_ADC24** **P2_Set_Mux**, **P2_Start_Conv** **P2_Wait_EOC** **P2_Read_ADC**, **P2_Read_ADC24**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv24**
Use sequence control	**P2_Seq_Init**, **P2_Seq_Start** **P2_Seq_Read**, **P2_Seq_Read24** **P2_SEQ_Read_Packed** **P2_Seq_Wait**
Control input limits	**P2_ADC_Read_Limit** **P2_ADC_Set_Limit**
Analog Outputs	
Output values	**P2_DAC**, **P2_DAC4**, **P2_DAC4_Packed**
Output values step by step	**P2_Write_DAC**, **P2_Write_DAC4** **P2_Write_DAC4_Packed** **P2_Write_DAC32** **P2_Start_DAC**
Digital Inputs / Outputs	
Configure input/outputs	**P2_MIO_DigProg**
Query input signals	**P2_MIO_Digin_Long**
Use latch register	**P2_MIO_Dig_Latch** **P2_MIO_Dig_Read_Latch** **P2_MIO_Dig_Write_Latch**
Set and read back output signals	**P2_MIO_Digout** **P2_MIO_Digout_Long** **P2_MIO_Get_Digout_Long**
Transistor Outputs	
Set and reset output signals	**P2_Digout**, **P2_Digout_Long** **P2_Digout_Bits** **P2_Get_Digout_Long**
Use latch register	**P2_Dig_Latch**, **P2_Sync_All** **P2_Dig_Write_Latch**
Optocouple Inputs	
Query input signals	**P2_Digin_Edge** **P2_Digin_Long**
Use latch register	**P2_Dig_Latch**, **P2_Sync_All** **P2_Dig_Read_Latch**

Range	Instructions
Control edges at input channels	P2_Digin_FIFO_Enable P2_Digin_FIFO_Read P2_Digin_FIFO_Read_Timer P2_Digin_FIFO_Clear P2_Digin_FIFO_Full
Counter Block	
Configure counter	P2_Cnt_Enable, P2_Cnt_Mode
Use counter	P2_Cnt_Clear, P2_Cnt_Get_Status P2_Cnt_Latch P2_Cnt_Read, P2_Cnt_Read4 P2_Cnt_Read_Latch P2_Cnt_Read_Latch4
Use PWM counter	P2_Cnt_PW_Enable, P2_Cnt_PW_Latch P2_Cnt_Get_PW, P2_Cnt_Get_PW_HL
SSI Decoder	
Use SSI decoder	P2_SSI_Mode, P2_SSI_Set_Bits P2_SSI_Set_Clock P2_SSI_Set_Delay, P2_SSI_Read P2_SSI_Read2 P2_SSI_Start, P2_SSI_Status
EtherCat Interface	
Use interface	only possible in *TiCoBasic*
General	
Use LED	P2_Check_LED, P2_Set_LED
Synchronize	P2_Sync_All, P2_Sync_Enable P2_Sync_Stat
Use interrupt and event inputs	P2_Event_Enable, P2_Event_Read P2_Event_Config

Programming in TiCoBasic

The module can be programmed with *TiCoBasic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `MIO_TiCo.inc` contains instructions for the functions:

Range	Instructions
Analog Inputs	
Set inputs to single-ended or differential	SE_Diff
Do a single conversion – complete or step by step	ADC, ADC24 Set_Mux, Start_Conv, Wait_EOC Read_ADC, Read_ADC24
Read value and start new conversion	Read_ADC_SConv Read_ADC_SConv24
Use sequence control	Seq_Init, Seq_Start Seq_Read, Seq_Read24 Seq_Wait
Control input limits	ADC_Read_Limit ADC_Set_Limit
Analog Outputs	
Output values	DAC

Range	Instructions
Output values step by step	Write_DAC, Write_DAC32 Start_DAC
Digital Inputs / Outputs	
Configure input/outputs	MIO_DigProg
Query input signals	MIO_Digin_Long
Use latch register	MIO_Dig_Latch MIO_Dig_Read_Latch MIO_Dig_Write_Latch
Set and read back output signals	MIO_Digout MIO_Digout_Long MIO_Get_Digout_Long
Transistor Outputs	
Set and reset output signals	MIO_Digout, MIO_Digout_Long MIO_Get_Digout_Long
Use latch register	MIO_Dig_Latch MIO_Dig_Write_Latch
Optocouple Inputs	
Query input signals	MIO_Digin_Long
Use latch register	MIO_Dig_Latch MIO_Dig_Read_Latch
Counter Block	
Configure counter	Cnt_Enable, Cnt_Mode
Use counter	Cnt_Clear, Cnt_Get_Status Cnt_Latch, Cnt_Read Cnt_Read_Int_Register Cnt_Sync_Latch, Cnt_Read_Latch
Use PWM counter	Cnt_PW_Enable, Cnt_PW_Latch Cnt_Get_PW_HL
SSI Decoder	
Use SSI decoder	SSI_Mode, SSI_Set_Bits SSI_Set_Clock SSI_Set_Delay, SSI_Read SSI_Start, SSI_Status
EtherCat Interface	
Use interface	MIO_ECAT_Init MIO_ECAT_Get_Driver_Version MIO_ECAT_Check_State_Transition MIO_ECAT_Read_Data_16L MIO_ECAT_Write_Data_16L
General	
Use LED	Check_LED, Set_LED
Use interrupt and event inputs	Event_Enable, Event_Read Event_Config, Trigger_Event

Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions
Data exchange with the *TiCo* processor using global variables and arrays	**P2_TDrv_Init** **P2_GetData_Long**, **P2_Get_Par**, **P2_Get_Par_Block** **P2_SetData_Long**, **P2_Set_Par**, **P2_Set_Par_Block** **P2_Get_TiCo_RingBuffer**, **P2_Set_TiCo_RingBuffer** **P2_RingBuffer_Empty** **P2_RingBuffer_Full**
Control *TiCo* processor	**P2_TiCo_Reset**, **P2_TiCo_Start**, **P2_TiCo_Stop** **P2_Get_TiCo_Bootloader_Status** **P2_Get_TiCo_Status**, **P2_Workload**
Control *TiCo* processes	**P2_Process_Status** **P2_TiCo_Get_Processdelay** **P2_TiCo_Set_Processdelay** **P2_TiCo_Start_Process** **P2_Tico_Stop_Process**
Transfer *TiCo* programs	**P2_TiCo_Flash**, **P2_TiCo_Load**

5.4 Pro II: Analog Input Modules

This section describes analog input modules for *ADwin-Pro II*.

Analog input modules for *ADwin-Pro I* be in the manual "ADwin-Pro Hardware" on page 19 ff.

Module name	AIn 8/18-L2	AIn 8/18-L2-TiCo	AIn 32/18-D	AIn 32/18-D-TiCo	AIn 8/18-8B	AIn 16/18-8B
Revision	E	E	E	E	E	E
Number ADC	1	1	1	1	1	1
Resolution [Bit]	18	18	18	18	18	18
Max. conv. time [µs]	2	2	2	2	2	2
max. sampl. rate [ksample/s]	500	500	500	500	500	500
Channels diff.	8	8	16	16	8	–
Channels sng. end.	–	–	32	32	–	–
Channels 8B	–	–	–	–	8	16
Meas. range	±10V					
Gain	1, 2, 4, 8					
Calibration	by software					
TiCo processor	–	1	–	1	–	–
page	38	38	41	41	44	47

Module name	AIn F-4/14	AIn F-8/14	AIn F-4/16	AIn F-8/16	AIn F-4/18	AIn F-8/18
Revision	E	E	E	E	E	E
Number ADC	4	8	4	8	4	8
Resolution [Bit]	14	14	16	16	18	18
Max. conv. time [µs]	0.02	0.02	0.25	0.25	2	2
max. sampl. rate [ksample/s]	50000	4×50000 8×25000	4000	4000	500	500
Channels diff.	4	8	4	8	4	8
Meas. range	±10V					
Gain	1		1, 2, 4, 8		1	
Calibration	by software					
page	49	53	57	61	65	68

Note for open-ended inputs

Open-ended inputs can cause errors - above all in an environment where interferences may occur. You can avoid open-ended inputs this way:

– Separate unused inputs from open-ended lines.

– Apply a specified level (for instance GND) to unused inputs. Make the connection as close to the connector as possible.

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

Analog input module Pro II-AIn-8/18-L2 Rev. E with an 18 bit ADC, 8 differential inputs and a programmable amplifier (PGA). The module can be combined with amplifiers, Pro-TC and Pro-PT modules.

The variant Pro II-AIn-8/18-L2-TiCo Rev. E additionally provides a freely programmable *TiCo* processor, which has access to all inputs of the module. Find more information about use and programming of the *TiCo* processor in the manual *TiCoBasic*.

The inputs are available with shielded LEMO connectors (2-pole, CAMAC European norm, see fig. 31).

The module Pro II-AIn-8/18-L2 Rev. E has an input voltage range of ±10V and a software selectable gain of 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

The module includes a sequential control, which can read measurement values from several or all input channels sequentially.

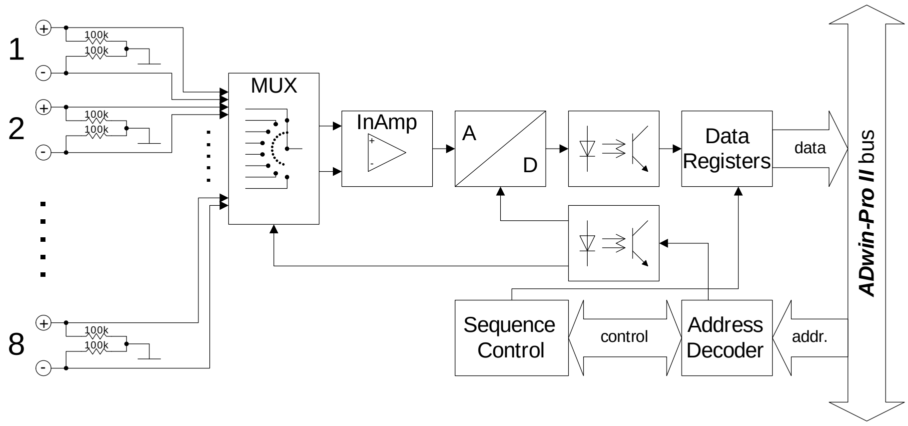
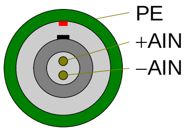


Fig. 29 – Pro II-AIn-8/18-L2 Rev. E: Block diagram

Input channels		8 differential via multiplexer
Resolution		18 Bit
Conversion time		max. 2µs
Sampling rate		max. 500ksps
Multiplexer settling time		2.5µs
Measurement range		±10V
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	±4 LSB typical
	DNL	max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30ppm/ºC
Connector		8 LEMO female connectors, 2-pole

Fig. 30 – Pro II-AIn-8/18-L2 Rev. E: Specification



2-pole LEMO female connector;
series 00 Multi-Contact, type EGG

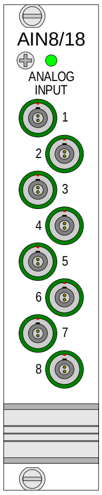


Fig. 31 – Pro II-AIn-8/18-L2 Rev. E: Front panel

Programming in ADbasic

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Do a single conversion – complete or step by step	**P2_ADC**, **P2_ADC24** **P2_Set_Mux**, **P2_Start_Conv** **P2_Wait_EOC** **P2_Read_ADC**, **P2_Read_ADC24**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv24**
Use sequence control	**P2_Seq_Init**, **P2_Seq_Start** **P2_Seq_Read**, **P2_Seq_Read24** **P2_Seq_Read24_Packed** **P2_Seq_Wait**
Control input limits	**P2_ADC_Read_Limit** **P2_ADC_Set_Limit**
Synchronize	**P2_Sync_All**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

Programming in TiCoBasic

The module can be programmed with *TiCoBasic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `AInTiCo.inc` contains instructions for the following functions:

Function	Instruction
Do a single conversion – complete or step by step	**SE_Diff** **ADC**, **ADC24** **Set_Mux**, **Start_Conv**, **Wait_EOC** **Read_ADC**, **Read_ADC24**
Read value and start new conversion	**Read_ADC_SConv** **Read_ADC_SConv24**
Use LED	**Check_LED**, **Set_LED**

Function	Instruction
Use interrupt and event inputs	`eVENT_eNABLE`, `eVENT_rEAD` `eVENT_cONFIG`, `Trigger_Event`

Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions

Data exchange with the *TiCo* processor using global variables and arrays	`P2_TDrv_Init`
	`P2_GetData_Long`, `P2_Get_Par`, `P2_Get_Par_Block`
	`P2_SetData_Long`, `P2_Set_Par`, `P2_Set_Par_Block`
	`P2_Get_TiCo_RingBuffer`,
	`P2_Set_TiCo_RingBuffer`
	`P2_RingBuffer_Empty`
	`P2_RingBuffer_Full`

Control *TiCo* processor	`P2_TiCo_Reset`, `P2_TiCo_Start`, `P2_TiCo_Stop`
	`P2_Get_TiCo_Bootloader_Status`
	`P2_Get_TiCo_Status`, `P2_Workload`

Control *TiCo* processes	`P2_Process_Status`
	`P2_TiCo_Get_Processdelay`
	`P2_TiCo_Set_Processdelay`
	`P2_TiCo_Start_Process`
	`P2_Tico_Stop_Process`

Transfer *TiCo* programs	`P2_TiCo_Flash`, `P2_TiCo_Load`

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

Analog input module Pro II-AIn-32/18-D Rev. E with an 18 bit ADC, 32 differential inputs and a programmable amplifier (PGA). The module can be combined with amplifiers, Pro-TC and Pro-PT modules.

The variant Pro II-AIn-32/18-D-TiCo Rev. E additionally provides a freely programmable *TiCo* processor, which has access to all inputs of the module. Find more information about use and programming of the *TiCo* processor in the manual *TiCoBasic*.

The module has 32 single-ended or 16 differential inputs (software selectable). The inputs are equipped with a 37-pin D-Sub female connector; for pin assignment see fig. 34 and 35.

After power-up the module is set to 16 differential inputs.

The module Pro II-AIn-32/18-D Rev. E has an input voltage range of ±10V and a software selectable gain of 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

Ex works the inputs are connected to the ground of the Pro device. Alternatively a GND level signal–common for all inputs–can be connected to one of the AGND pins. The ground connection to the Pro device should be split up, by switching the DIL switch (see fig. 36) to position `GND LIFT`.

Do not run the module without ground connection.

The module includes a sequential control, which can read measurement values from several or all input channels sequentially.

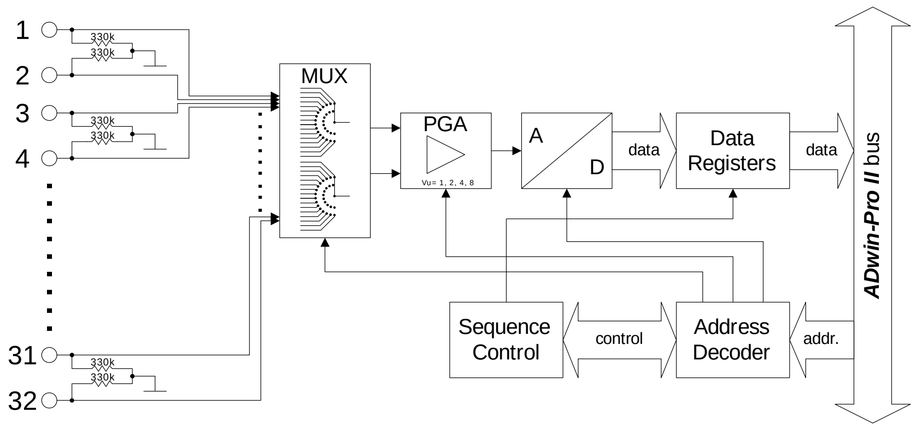


Fig. 32 – Pro II-AIn-32/18-D Rev. E: Block diagram

Input channels:		32 single ended or 16 differential; via multiplexer
Resolution:		18 bit
Conversion time:		max. 2µs
Sampling rate:		max. 500ksps
Multiplexer settling time:		2.5µs
Measurement range:		±10V
Gain:		1, 2, 4, 8 software selectable
Accuracy	INL	±4 LSB typical
	DNL	max. ±1 LSB
Input resistance:		330kΩ, ±2%
Input over-voltage:		±35V

Fig. 33 – Pro II-AIn-32/18-D Rev. E: Specification

Offset error:	adjustable
Offset drift:	±30 ppm/°C
Connector:	37-pin D-Sub female connector

Fig. 33 – Pro II-AIn-32/18-D Rev. E: Specification

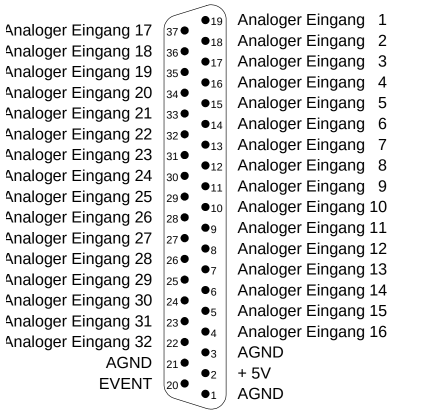


Fig. 34 – Pro II-AIn-32/18-D Rev. E: Pin assignment single ended

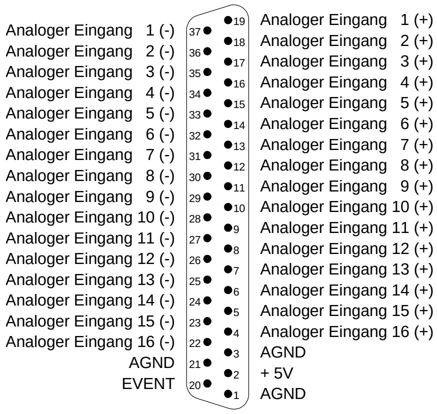


Fig. 35 – Pro II-AIn-32/18-D Rev. E: Pin assignment differential

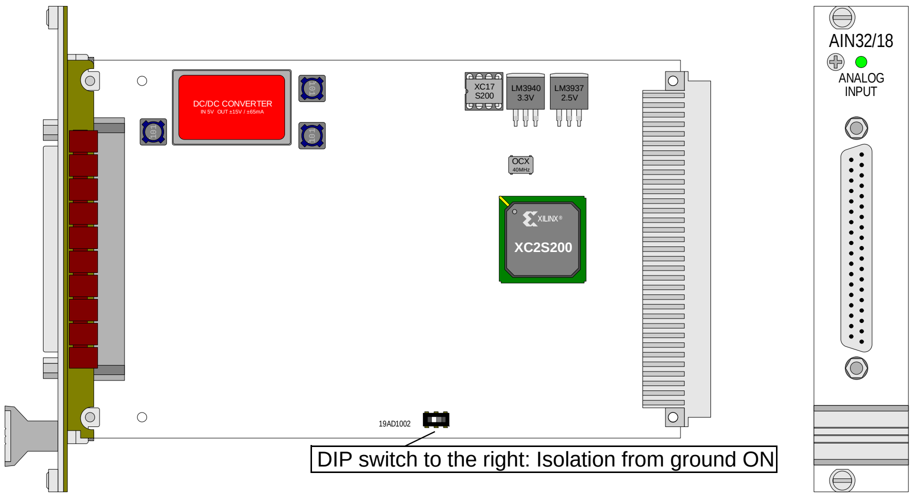


Fig. 36 – Pro II-AIn-32/18-D Rev. E: Board and front panel

Programming in ADbasic

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Do a single conversion – complete or step by step	**`P2_ADC`**, **`P2_ADC24`** **`P2_Set_Mux`**, **`P2_Start_Conv`** **`P2_Wait_EOC`** **`P2_Read_ADC`**, **`P2_Read_ADC24`**
Read value and start new conversion	**`P2_Read_ADC_SConv`** **`P2_Read_ADC_SConv24`**

Function	Instructions
Use sequence control	**P2_Seq_Init**, **P2_Seq_Start** **P2_Seq_Read**, **P2_Seq_Read24** **P2_Seq_Read24_Packed** **P2_Seq_Wait**
Control input limits	**P2_ADC_Read_Limit** **P2_ADC_Set_Limit**
Synchronize	**P2_Sync_All**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions
Data exchange with the *TiCo* processor using global variables and arrays	**P2_TDrv_Init** **P2_GetData_Long**, **P2_Get_Par**, **P2_Get_Par_Block** **P2_SetData_Long**, **P2_Set_Par**, **P2_Set_Par_Block** **P2_Get_TiCo_RingBuffer**, **P2_Set_TiCo_RingBuffer** **P2_RingBuffer_Empty** **P2_RingBuffer_Full**
Control *TiCo* processor	**P2_TiCo_Reset**, **P2_TiCo_Start**, **P2_TiCo_Stop** **P2_Get_TiCo_Bootloader_Status** **P2_Get_TiCo_Status**, **P2_Workload**
Control *TiCo* processes	**P2_Process_Status** **P2_TiCo_Get_Processdelay** **P2_TiCo_Set_Processdelay** **P2_TiCo_Start_Process** **P2_Tico_Stop_Process**
Transfer *TiCo* programs	**P2_TiCo_Flash**, **P2_TiCo_Load**

Programming in TiCoBasic

The module can be programmed with *TiCoBasic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `AInTiCo.inc` contains instructions for the following functions:

Function	Instruction
Do a single conversion – complete or step by step	**ADC**, **ADC24** **Set_Mux**, **Start_Conv**, **Wait_EOC** **Read_ADC**, **Read_ADC24**
Read value and start new conversion	**Read_ADC_SConv** **Read_ADC_SConv24**
Use LED	**Check_LED**, **Set_LED**
Use interrupt and event inputs	**eVENT_eNABLE**, **eVENT_rEAD** **eVENT_cONFIG**, **Trigger_Event**

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

Analog input module Pro II-AIn-8/18-8B Rev. E with an 18 bit ADC, 16 analog inputs and a programmable amplifier (PGA). The module is based upon Pro II-AIn-32/18-D Rev. E with an additional board holding 8 plug-in slots for 8B modules.

The 16 inputs are divided to 8 inputs for 8B modules and 8 differential inputs. The inputs are provided on two 37-pin D-Sub female connectors; for pin assignments see fig. 38.

The input voltage range after 8B modules is ±10V. Gain is software selectable to 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

Ex works the inputs are connected to the ground of the Pro device. Alternatively a GND level signal–common for all inputs–can be connected to one of the AGND pins. The ground connection to the Pro device should be split up, by switching the DIL switch (see fig. 36) to position `GND LIFT`.

Do not run the module without ground connection.

The module includes a sequential control, which can read measurement values from several or all input channels sequentially.

Input channels:		16 via multiplexer: 8 for 8B modules, 8 differential
Resolution:		18 bit
Conversion time:		max. 2µs
Sampling rate:		max. 500ksps
Multiplexer settling time:		2.5µs
Measurement range:		±10V
Gain:		1, 2, 4, 8 software selectable
Accuracy	INL	±4 LSB typical
	DNL	max. ±1 LSB
Input resistance:		330kΩ, ±2%
Input over-voltage:		±35V
Offset error:		adjustable
Offset drift:		±30ppm/ºC
Connector:		37-pin D-Sub female connector

Fig. 37 – Pro II-AIn-8/18-8B Rev. E: Specification

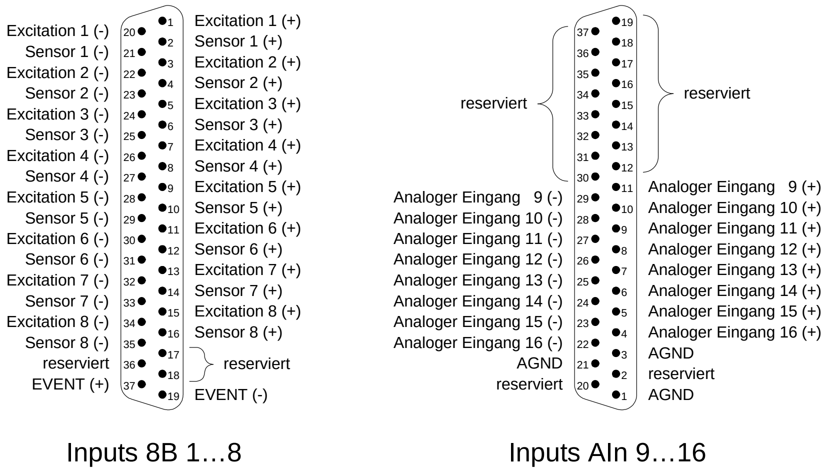


Fig. 38 – Pro II-AIn-8/18-8B Rev. E: Pin assignments

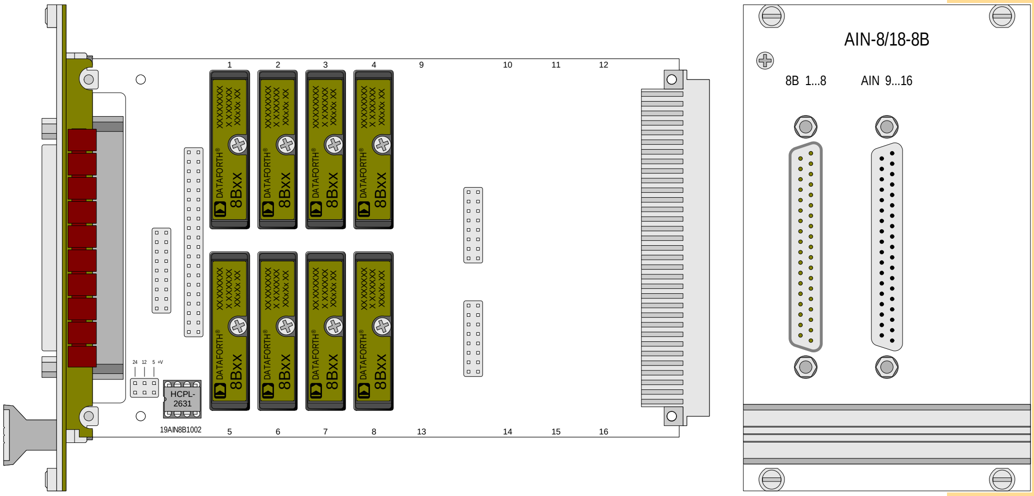


Fig. 39 – Pro II-AIn-8/18-8B Rev. E: Board and front panel

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Do a single conversion – complete or step by step	**P2_ADC**, **P2_ADC24** **P2_Set_Mux**, **P2_Start_Conv** **P2_Wait_EOC** **P2_Read_ADC**, **P2_Read_ADC24**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv24**
Use sequence control	**P2_Seq_Init**, **P2_Seq_Start** **P2_Seq_Read**, **P2_Seq_Read24** **P2_Seq_Read24_Packed** **P2_Seq_Wait**
Control input limits	**P2_ADC_Read_Limit** **P2_ADC_Set_Limit**
Synchronize	**P2_Sync_All**
Use LED	**P2_Check_LED**, **P2_Set_LED**

Function	Instructions
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

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

Analog input module Pro II-AIn-16/18-8B Rev. E with an 18 bit ADC, 16 analog inputs and a programmable amplifier (PGA). The module is based upon Pro II-AIn-32/18-D Rev. E with an additional board holding 16 plug-in slots for 8B modules.

The module has 16 inputs (via 8B modules) provided on two 37-pin D-Sub female connectors; for pin assignments see fig. 41.

The input voltage range after 8B modules is ±10V. Gain is software selectable to 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

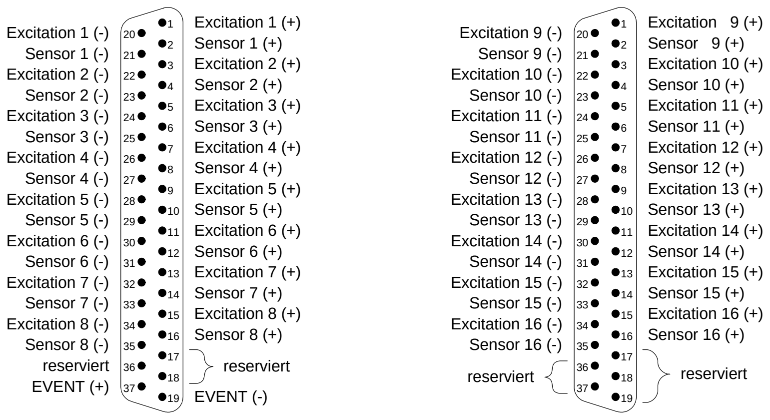
Ex works the inputs are connected to the ground of the Pro device. Alternatively a GND level signal–common for all inputs–can be connected to one of the AGND pins. The ground connection to the Pro device should be split up, by switching the DIL switch (see fig. 36) to position `GND LIFT`.

Do not run the module without ground connection.

The module includes a sequential control, which can read measurement values from several or all input channels sequentially.

Input channels:		16 via multiplexer for 8B modules.
Resolution:		18 bit
Conversion time:		max. 2µs
Sampling rate:		max. 500ksps
Multiplexer settling time:		2.5µs
Measurement range:		±10V
Gain:		1, 2, 4, 8 software selectable
Accuracy	INL	±4 LSB typical
	DNL	max. ±1 LSB
Input resistance:		330kΩ, ±2%
Input over-voltage:		±35V
Offset error:		adjustable
Offset drift:		±30ppm/ºC
Connector:		37-pin D-Sub female connector

Fig. 40 – Pro II-AIn-16/18-8B Rev. E: Specification



Eingänge 8B 1…8 Eingänge 8B 9…16

Fig. 41 – Pro II-AIn-16/18-8B Rev. E: Pin assignments

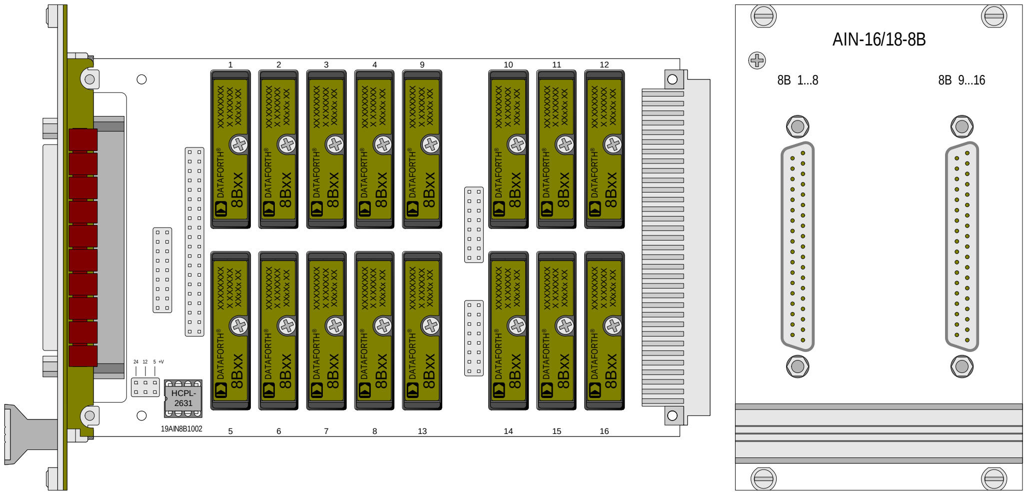


Fig. 42 – Pro II-AIn-16/18-8B Rev. E: Board and front panel

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Do a single conversion – complete or step by step	**P2_ADC**, **P2_ADC24** **P2_Set_Mux**, **P2_Start_Conv** **P2_Wait_EOC** **P2_Read_ADC**, **P2_Read_ADC24**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv24**
Use sequence control	**P2_Seq_Init**, **P2_Seq_Start** **P2_Seq_Read**, **P2_Seq_Read24** **P2_Seq_Read24_Packed** **P2_Seq_Wait**
Control input limits	**P2_ADC_Read_Limit** **P2_ADC_Set_Limit**
Synchronize	**P2_Sync_All**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

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

The analog input module Pro II-AIn-F-4/14 Rev. E has 4 Fast-ADC of 14 Bit and 4 differential inputs.

The inputs are available with the following connectors:

– Pro II-AIn-F-4/14-L2: shielded LEMO connectors, 2-pole, CAMAC European norm.

– Pro II-AIn-F-4/14-D: D-Sub female connector 37-pin.

– Pro II-AIn-F-4/14-B: BNC female connectors.

The module's converters run stand-alone with a fixed clock rate of 50MHz. The module memory enables to buffer a great amount of data–especially with burst sequences.

The module provides several operation modes for converting analog signals:

– Single measurement: The module converts stand-alone (with fixed clock rate). The *ADbasic* program queries the current measurement value and processes them.

– Single burst sequence: The *ADbasic* program starts a compelete burst measurement sequence, that is a defined number of single measurements.

– Continuous burst sequence: The *ADbasic* program starts a burst measurement sequence, that continuously does single measurements until the sequence is stopped. The data is stored in a fifo-like memory.

The module Pro II-AIn-F-4/14 Rev. E has an input voltage range of ±10V. The adjustment of gain and offset is made by software. (see chapter 6 "Calibration").

As an option, the module can return the moving average of 2…32 measurement values instead of simple measurement values.

Burst Sequence

The module processes burst sequences independent from the processor module of the *ADwin* system. The measurement values–number and measuring rate to be defined ahead in the program–are stored in the burst-memory of the module. The processor module then reads the stored values (even during a burst sequence) and processes them.

With a continuous burst-sequence the measuring rate must be syntonized to the reading rate. Please note:

– Measurement values are always read in blocks, the block size is selectable. The greater the data blocks are, the faster the average reading is done.

  Look out: While a block is read, other processes even with higher priority may be delayed. The probability of a delay rises with the block size.

– The time offset between continuous conversion and blockwise reading demands a data buffer. Therefore, the initialization of the burst sequence must allocate sufficient memory range (**`P2_BURST_INIT`**, parameter `samples`).

Event Inputs

With module version Pro II-AIn-F-4/14-D Rev. E (D-Sub female connector) a burst sequence can be controlled by external event signals, i.e. each (resulting) event signal has a measurement value stored.

As an option one channel of the burst sequence may be used as time channel, which holds the counter value of the internal module timer for each event signal.

The module is equipped with 3 differential event inputs: EVENT/A, B, ENABLE, whose signals are processed to the resulting event signal. This pre-processing of signals be configured with **P2_EVENT2_CONFIG**..

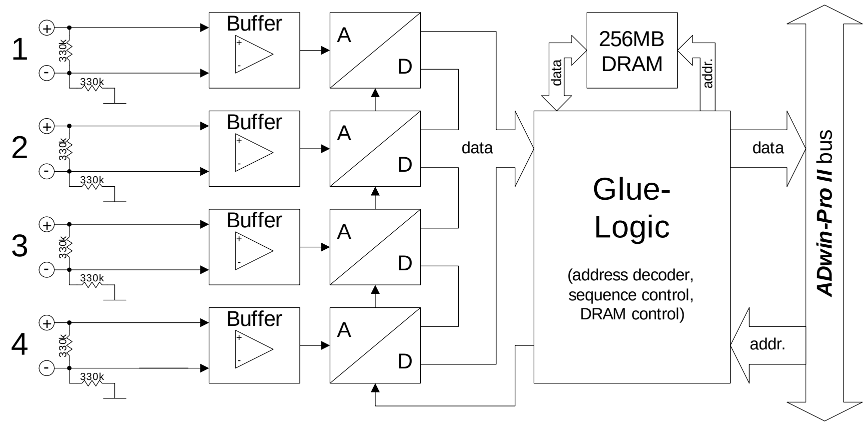


Fig. 43 – Pro II-AIn-F-4/14 Rev. E: Block diagram

Input channels	4 differential
Resolution	14 Bit
Conversion time	0.02µs (per ADC)
Sampling rate	50000ksps (per ADC)
Input band width	0 … 4MHz
Memory size	256MiB or 2^{27} = 134217728 values total
Measurement range	±10V with max. offset 3.5 V
Accuracy INL	typical ±1.2 LSB, max. ±5 LSB
Accuracy DNL	typical ±0.5 LSB, max. ±1 LSB
Input resistance	330kΩ, ±2%
Input over-voltage	±35V
Offset error	adjustable
Offset drift	±30ppm/°C of full scale range
Event input (D-Sub only)	3 differential; RS422/485 compatible (5V differential, 120 Ω bus terminating resistor) max. signal frequency 16MHz
Connector	4 LEMO female connectors. 2-pole optional: 37-pin D-Sub female connector optional: 4 BNC female connectors.

Fig. 44 – Pro II-AIn-F-4/14 Rev. E: Specification

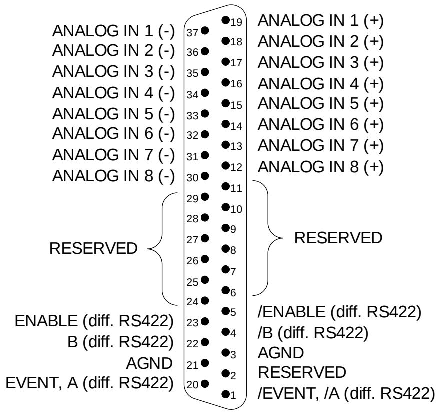


Fig. 45 – Pro-AIn-F-4/14-D Rev. E: Pin assignment differential

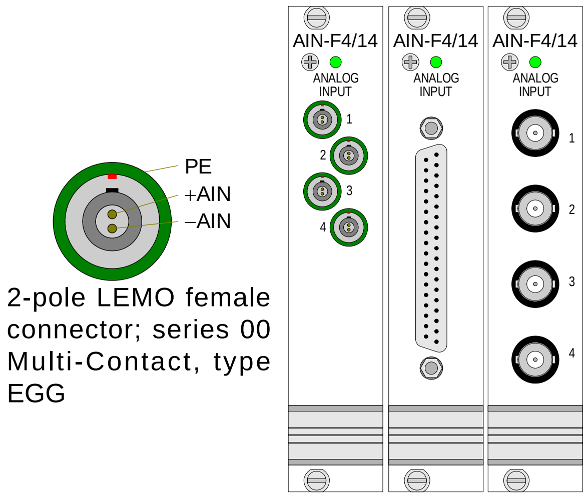


Fig. 46 – Pro II-AIn-F-4/14 Rev. E: Front panels

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure sequence control	**P2_ADCF_Mode**
Do a single conversion – complete or step by step	**P2_ADCF** **P2_Start_Conv**, **P2_Wait_EOCF** **P2_Read_ADCF**
Read several values	**P2_Read_ADCF32**, **P2_Read_ADCF4** **P2_Read_ADCF4_Packed**

Function	Instructions
Use burst conversion	P2_Burst_Init **P2_Burst_Start**, **P2_Burst_Stop** **P2_Burst_Status**, **P2_Burst_Reset** **P2_Burst_Read** **P2_Burst_Read_Index** **P2_Burst_Read_Unpacked1** **P2_Burst_Read_Unpacked2** **P2_Burst_Read_Unpacked4** **P2_Burst_CRead_Unpacked1** **P2_Burst_CRead_Unpacked2** **P2_Burst_CRead_Unpacked4**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Stat**
Control input limits	**P2_ADCF_Read_Limit** **P2_ADCF_Set_Limit**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

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

The analog input module Pro II-AIn-F-8/14 Rev. E has 8 Fast-ADC of 14 Bit and 8 differential inputs.

The inputs are available with the following connectors:

– Pro II-AIn-F-8/14-L2: shielded LEMO female connectors, 2-pole, CAMAC European norm.

– Pro II-AIn-F-8/14-D: D-Sub female connector 37-pin.

– Pro II-AIn-F-8/14-B: BNC female connectors.

The module's converters run stand-alone with a fixed clock rate of 50MHz. The great amount of data–especially with burst sequences–is buffered in the module memory. If converting measurement values from all 8 channels, the sampling rate is limited to 25MHz due to the maximum memory access rate.

The module provides several operation modes for converting analog signals:

– Single measurement: The module converts stand-alone (with fixed clock rate). The *ADbasic* program queries the current measurement value and processes them.

– Single burst sequence: The *ADbasic* program starts a compelete burst measurement sequence, that is a defined number of single measurements.

– Continuous burst sequence: The *ADbasic* program starts a burst measurement sequence, that continuously does single measurements until the sequence is stopped. The data is stored in a fifo-like memory.

The module Pro II-AIn-F-8/14 Rev. E has an input voltage range of ±10V. The adjustment of gain and offset is made by software. (see chapter 6 "Calibration").

As an option, the module can return the moving average of 2…32 measurement values instead of simple measurement values.

Burst Sequence

The module processes burst sequences independent from the processor module of the *ADwin* system. The measurement values–number and measuring rate to be defined ahead in the program–are stored in the burst-memory of the module. The processor module then reads the stored values (even during a burst sequence) and processes them.

With a continuous burst-sequence the measuring rate must be syntonized to the reading rate. Please note:

– Measurement values are always read in blocks, the block size is selectable. The greater the data blocks are, the faster the average reading is done.

  Look out: While a block is read, other processes even with higher priority may be delayed. The probability of a delay rises with the block size.

– The time offset between continuous conversion and blockwise reading demands a data buffer. Therefore, the initialization of the burst sequence must allocate sufficient memory range (**`P2_BURST_INIT`**, parameter `samples`).

Event Inputs

With module version Pro II-AIn-F-8/14-D Rev. E (D-Sub female connector) a burst sequence can be controlled by external event signals, i.e. each (resulting) event signal has a measurement value stored.

As an option one channel of a burst sequence may be used as time channel, holding the counter value of the internal module timer for each event signal.

The module is equipped with 3 differential event inputs: `EVENT/A`, `B`, `ENABLE`, whose signals are processed to the resulting event signal. This pre-processing of signals be configured with **P2_EVENT2_CONFIG**..

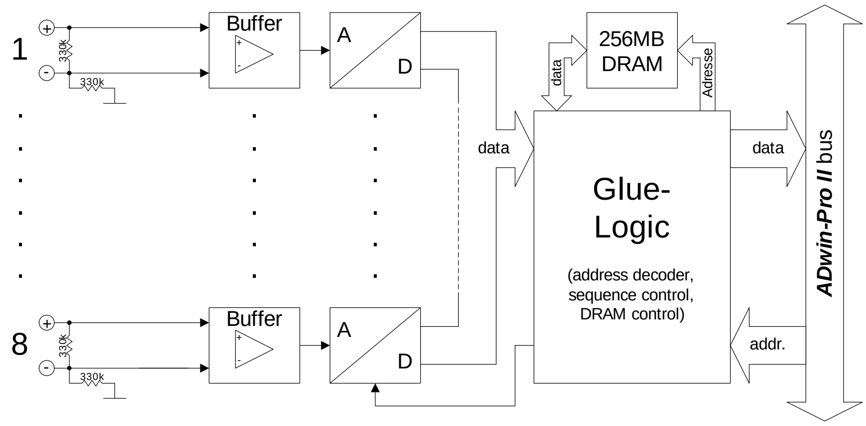


Fig. 47 – Pro II-AIn-F-8/14 Rev. E: Block diagram

Input channels		8 differential
Resolution		14 Bit
Conversion time		0.02µs (per ADC)
Sampling rate		4×50000ksps (per ADC) or 8×25000ksps (per ADC)
Input band width		0 … 4MHz
Memory size		256MiB or 2^{27} = 134217728 values total
Measurement range		±10V with max. offset 3.5 V
Accuracy	INL	typical ±1.2 LSB, max. ±5 LSB
	DNL	typical ±0.5 LSB, max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30ppm/°C of full scale range
Event input (D-Sub only)		3 differential; RS422/485 compatible (5V differential, 120 Ω bus terminating resistor) max. signal frequency 16MHz
Connector		8 LEMO female connectors. 2-pole optional: 37-pin D-Sub female connector optional: 8 BNC female connectors.

Fig. 48 – Pro II-AIn-F-8/14 Rev. E: Specification

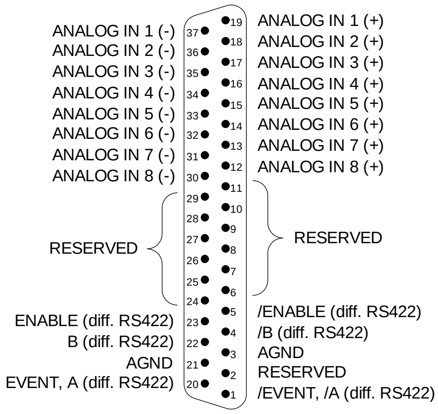


Fig. 49 – Pro-AIn-F-8/14-D Rev. E: Pin assignment differential

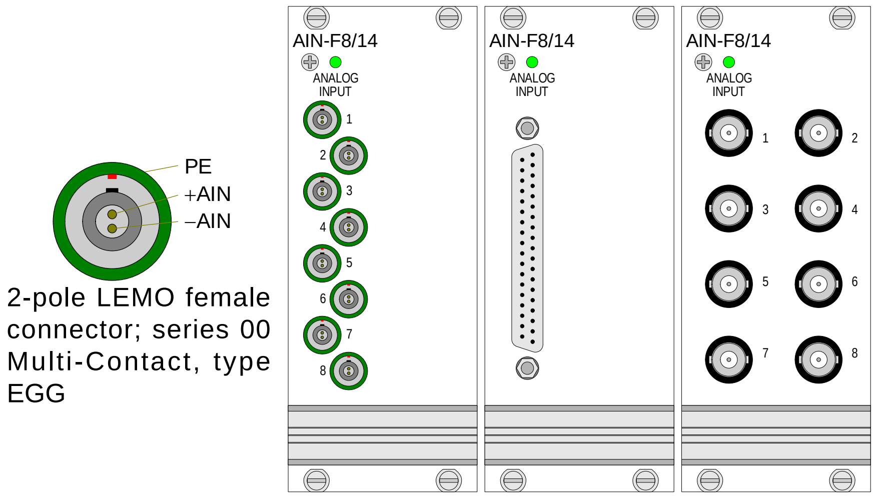


Fig. 50 – Pro II-AIn-F-8/14 Rev. E: Front panels

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure sequence control	**P2_ADCF_Mode**
Do a single conversion – complete or step by step	**P2_ADCF** **P2_Start_Conv**, **P2_Wait_EOCF** **P2_Read_ADCF**
Read several values	**P2_Read_ADCF32** **P2_Read_ADCF4**, **P2_Read_ADCF8** **P2_Read_ADCF4_Packed** **P2_Read_ADCF8_Packed**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv32**

Function	Instructions
Use burst conversion	**P2_Burst_Init** **P2_Burst_Start**, **P2_Burst_Stop** **P2_Burst_Status**, **P2_Burst_Reset** **P2_Burst_Read** **P2_Burst_Read_Index** **P2_Burst_Read_Unpacked1** **P2_Burst_Read_Unpacked2** **P2_Burst_Read_Unpacked4** **P2_Burst_Read_Unpacked8** **P2_Burst_CRead_Unpacked1** **P2_Burst_CRead_Unpacked2** **P2_Burst_CRead_Unpacked4** **P2_Burst_CRead_Unpacked8**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Mode**, **P2_Sync_Stat**
Control input limits	**P2_ADCF_Read_Limit** **P2_ADCF_Set_Limit** **P2_ADCF_Reset_Min_Max**, **P2_ADCF_Read_Min_Max4**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

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

The analog input module Pro II-AIn-F-4/16 Rev. E has 4 Fast-ADC of 16 Bit and 4 differential inputs.

The inputs are available with the following connectors:

– Pro II-AIn-F-4/16-L2: shielded LEMO connectors, 2-pole, CAMAC European norm.

– Pro II-AIn-F-4/16-D: D-Sub female connector 37-pin.

– Pro II-AIn-F-4/16-B: BNC female connectors.

The module’s converters run with a clock rate of up to 4MHz. The module memory enables to buffer data–especially with burst sequences.

The module provides several operation modes for converting analog signals:

– Single measurement: You start each conversion manually, query the measurement value and processes it in the *ADbasic* program.

– Single burst sequence: The *ADbasic* program starts a compelete burst measurement sequence, that is a defined number of single measurements.

– Continuous burst sequence: The *ADbasic* program starts a burst measurement sequence, that continuously does single measurements until the sequence is stopped. The data is stored in a fifo-like memory.

The module Pro II-AIn-F-4/16 Rev. E has an input voltage range of ±10V and a software selectable gain of 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

As an option, the module can return the average of 2…32 measurement values instead of simple measurement values. In this case the selected number of measurement values is converted before the average can be calculated.

Burst Sequence

The module processes burst sequences independent from the processor module of the *ADwin* system. The measurement values–number and measuring rate to be defined ahead in the program–are stored in the burst-memory of the module. The processor module then reads the stored values (even during a burst sequence) and processes them.

With a continuous burst-sequence the measuring rate must be syntonized to the reading rate. Please note:

– Measurement values are always read in blocks, the block size is selectable. The greater the data blocks are, the faster the average reading is done.

  Look out: While a block is read, other processes even with higher priority may be delayed. The probability of a delay rises with the block size.

– The time offset between continuous conversion and blockwise reading demands a data buffer. Therefore, the initialization of the burst sequence must allocate sufficient memory range (**`P2_BURST_INIT`**, parameter `samples`).

Event Inputs

With module version Pro II-AIn-F-4/16-D Rev. E (D-Sub female connector) a burst sequence can be controlled by external event signals, i.e. each (resulting) event signal has a measurement value stored.

As an option one channel of the burst sequence may be used as time channel, which holds the counter value of the internal module timer for each event signal.

The module is equipped with 3 differential event inputs: `EVENT/A`, `B`, `ENABLE`, whose signals are processed to the resulting event signal. This pre-processing of signals be configured with **`P2_EVENT2_CONFIG`**..

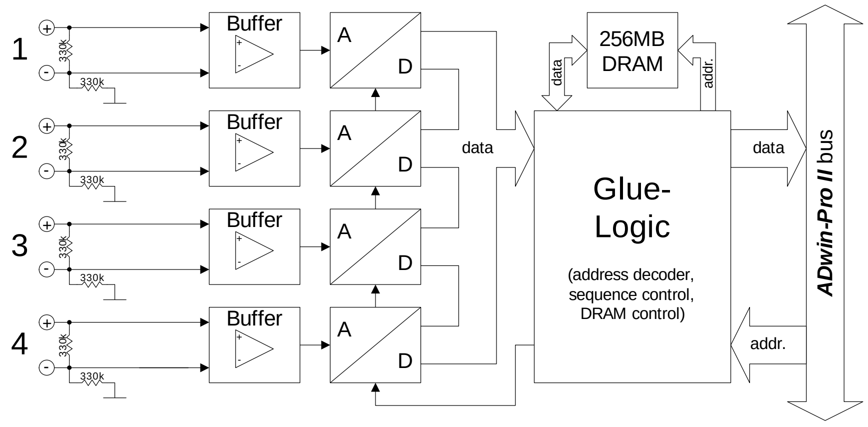


Fig. 51 – Pro II-AIn-F-4/16 Rev. E: Block diagram

Input channels		4 differential
Resolution		16 Bit
Conversion time		0.25µs (per ADC)
Input band width		0 … 600kHz
Memory size		256MiB or 2^{27} = 134217728 values total
Measurement range		±10V with max. offset 3.5 V
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	typical ±1.2 LSB, max. ±5 LSB
	DNL	typical ±0.5 LSB, max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±20V
Offset error		adjustable
Offset drift		±30ppm/°C of full scale range
Event input (D-Sub only)		3 differential; RS422/485 compatible (5V differential, 120 Ω bus terminating resistor)
Connector		4 LEMO female connectors. 2-pole optional: 37-pin D-Sub female connector optional: 4 BNC female connectors.

Fig. 52 – Pro II-AIn-F-4/16 Rev. E: Specification

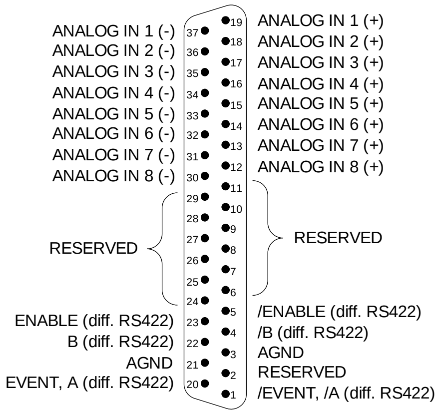


Fig. 53 – Pro-AIn-F-4/16-D Rev. E: Pin assignment differential

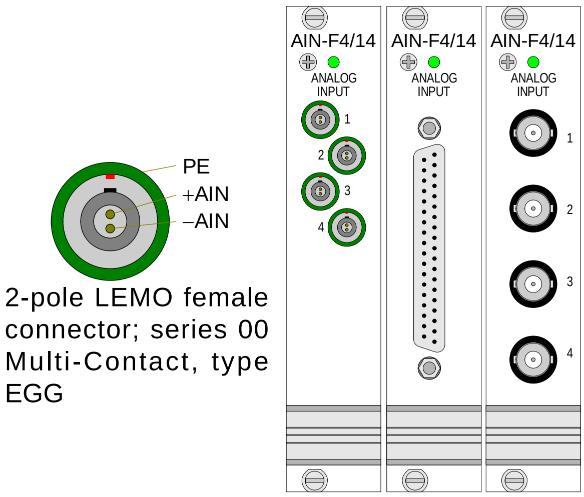


Fig. 54 – Pro II-AIn-F-4/16 Rev. E: Front panels

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure sequence control	**P2_ADCF_Mode**, **P2_Set_Average_Filter**, **P2_Set_Gain**
Do a single conversion – complete or step by step	**P2_ADCF** **P2_Start_ConvF**, **P2_Wait_EOCF** **P2_Read_ADCF**
Read several values	**P2_Read_ADCF32** **P2_Read_ADCF4** **P2_Read_ADCF4_Packed**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv32**

Function	Instructions
Use burst conversion	**P2_Burst_Init** **P2_Burst_Start**, **P2_Burst_Stop** **P2_Burst_Status**, **P2_Burst_Reset** **P2_Burst_Read** **P2_Burst_Read_Index** **P2_Burst_Read_Unpacked1** **P2_Burst_Read_Unpacked2** **P2_Burst_Read_Unpacked4** **P2_Burst_CRead_Unpacked1** **P2_Burst_CRead_Unpacked2** **P2_Burst_CRead_Unpacked4**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Mode**, **P2_Sync_Stat**
Control input limits	**P2_ADCF_Read_Limit** **P2_ADCF_Set_Limit** **P2_ADCF_Reset_Min_Max**, **P2_ADCF_Read_Min_Max4**, **P2_ADCF_Read_Min_Max8**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

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

The analog input module Pro II-AIn-F-8/16 Rev. E has 8 Fast-ADC of 14 Bit and 8 differential inputs.

The inputs are available with the following connectors:

– Pro II-AIn-F-8/16-L2: shielded LEMO female connectors, 2-pole, CAMAC European norm.

– Pro II-AIn-F-8/16-D: D-Sub female connector 37-pin.

– Pro II-AIn-F-8/16-B: BNC female connectors.

The module's converters run with a clock rate of up to 4MHz. The module memory enables to buffer data–especially with burst sequences.

The module provides several operation modes for converting analog signals:

– Single measurement: You start each conversion manually, query the measurement value and processes it in the *ADbasic* program.

– Single burst sequence: The *ADbasic* program starts a compelete burst measurement sequence, that is a defined number of single measurements.

– Continuous burst sequence: The *ADbasic* program starts a burst measurement sequence, that continuously does single measurements until the sequence is stopped. The data is stored in a fifo-like memory.

The module Pro II-AIn-F-8/16 Rev. E has an input voltage range of ±10V and a software selectable gain of 1, 2, 4 or 8. The adjustment of gain and offset is done by software (see chapter 6 "Calibration").

As an option, the module can return the average of 2…32 measurement values instead of simple measurement values. In this case the selected number of measurement values is converted before the average can be calculated.

Burst Sequence

The module processes burst sequences independent from the processor module of the *ADwin* system. The measurement values–number and measuring rate to be defined ahead in the program–are stored in the burst-memory of the module. The processor module then reads the stored values (even during a burst sequence) and processes them.

With a continuous burst-sequence the measuring rate must be syntonized to the reading rate. Please note:

– Measurement values are always read in blocks, the block size is selectable. The greater the data blocks are, the faster the average reading is done.

  Look out: While a block is read, other processes even with higher priority may be delayed. The probability of a delay rises with the block size.

– The time offset between continuous conversion and blockwise reading demands a data buffer. Therefore, the initialization of the burst sequence must allocate sufficient memory range (**P2_BURST_INIT**, parameter samples).

Event Inputs

With module version Pro II-AIn-F-8/16-D Rev. E (D-Sub female connector) a burst sequence can be controlled by external event signals, i.e. each (resulting) event signal has a measurement value stored.

As an option one channel of a burst sequence may be used as time channel, holding the counter value of the internal module timer for each event signal.

The module is equipped with 3 differential event inputs: EVENT/A, B, ENABLE, whose signals are processed to the resulting event signal. This pre-processing of signals be configured with **P2_EVENT2_CONFIG**..

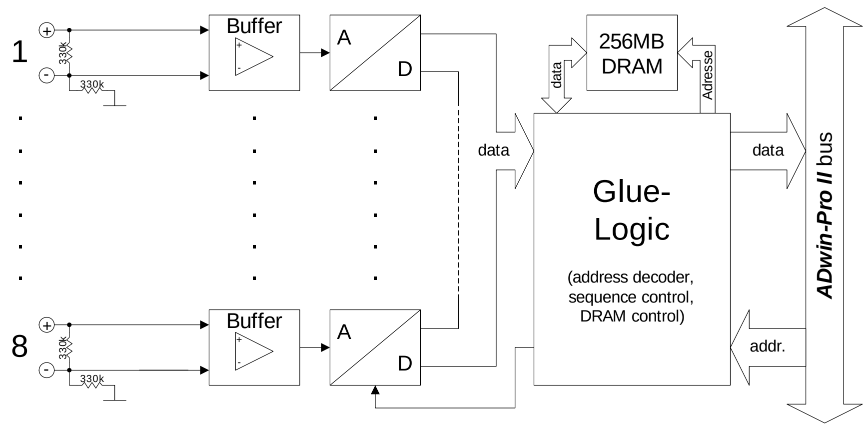


Fig. 55 – Pro II-AIn-F-8/16 Rev. E: Block diagram

Input channels		8 differential
Resolution		16 Bit
Conversion time		0.25µs (per ADC)
Input band width		0 … 600kHz
Memory size		256MiB or 2^{27} = 134217728 values total
Measurement range		±10V with max. offset 3.5 V
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	typical ±1.2 LSB, max. ±5 LSB
	DNL	typical ±0.5 LSB, max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±20V
Offset error		adjustable
Offset drift		±30ppm/ºC of full scale range
Event input (D-Sub only)		3 differential; RS422/485 compatible (5V differential, 120 Ω bus terminating resistor)
Connector		8 LEMO female connectors. 2-pole optional: 37-pin D-Sub female connector optional: 8 BNC female connectors.

Fig. 56 – Pro II-AIn-F-8/16 Rev. E: Specification

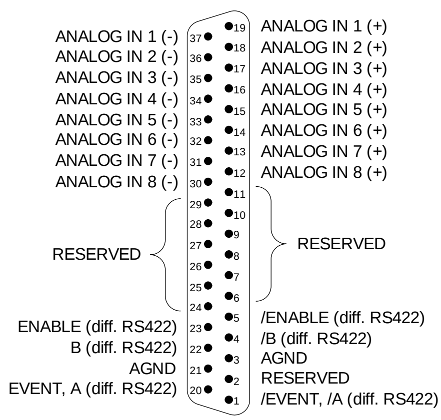


Fig. 57 – Pro-AIn-F-8/16-D Rev. E: Pin assignment differential

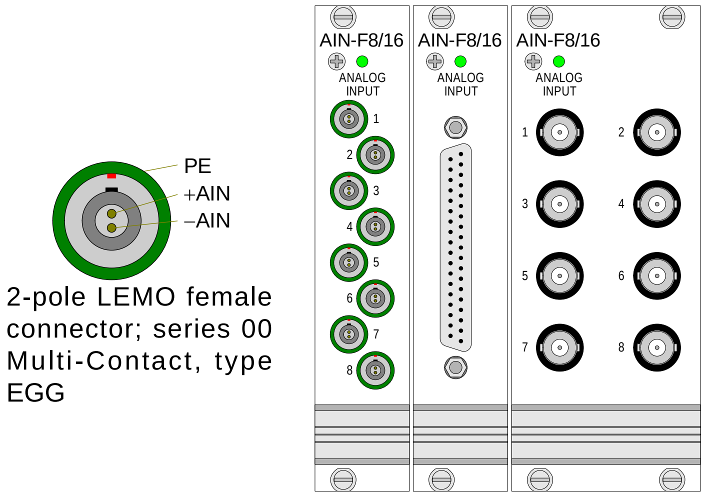


Fig. 58 – Pro II-AIn-F-8/16 Rev. E: Front panels

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure sequence control	**P2_ADCF_Mode**, **P2_Set_Average_Filter**, **P2_Set_Gain**
Do a single conversion – complete or step by step	**P2_ADCF** **P2_Start_ConvF**, **P2_Wait_EOCF** **P2_Read_ADCF**
Read several values	**P2_Read_ADCF32** **P2_Read_ADCF4**, **P2_Read_ADCF8** **P2_Read_ADCF4_Packed** **P2_Read_ADCF8_Packed**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv32**
Use burst conversion	**P2_Burst_Init** **P2_Burst_Start**, **P2_Burst_Stop** **P2_Burst_Status**, **P2_Burst_Reset** **P2_Burst_Read** **P2_Burst_Read_Index** **P2_Burst_Read_Unpacked1** **P2_Burst_Read_Unpacked2** **P2_Burst_Read_Unpacked4** **P2_Burst_Read_Unpacked8** **P2_Burst_CRead_Unpacked1** **P2_Burst_CRead_Unpacked2** **P2_Burst_CRead_Unpacked4** **P2_Burst_CRead_Unpacked8**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Stat**
Control input limits	**P2_ADCF_Read_Limit** **P2_ADCF_Set_Limit**
Use LED	**P2_Check_LED**, **P2_Set_LED**

Function	Instructions
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

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

The analog input module Pro II-AIn-F-4/18 Rev. E has 4 Fast ADC of 18 Bit and 4 differential inputs. The inputs are galvanically isolated from each other as well as from other modules.

The inputs are available with the following connectors:

– Pro II-AIn-F-4/18-L2: shielded LEMO female connectors, 2-pole, CAMAC European norm.

– Pro II-AIn-F-4/18-D: D-Sub female connector 37-pin.

The module Pro II-AIn-F-4/18 Rev. E has input voltage range of ±10V. The adjustment of gain and offset is made by software. (see chapter 6 "Calibration").

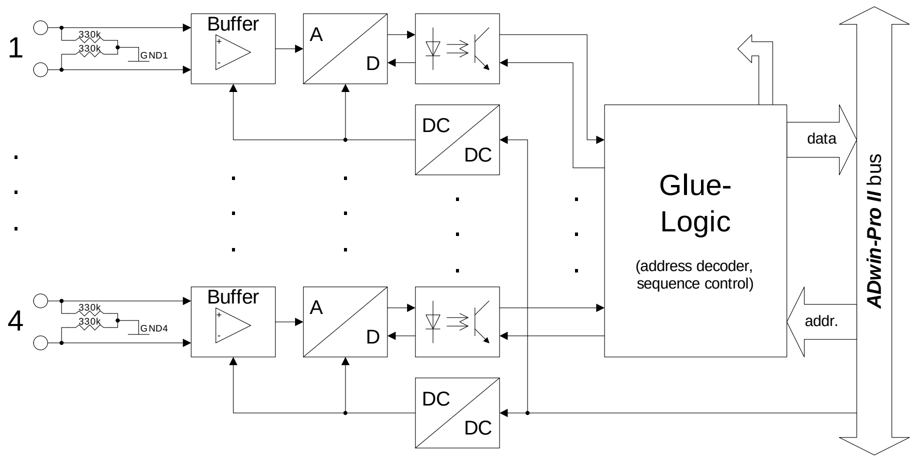


Fig. 59 – Pro II-AIn-F-4/18 Rev. E: Block diagram

Input channels		4 differential, galvanically isolated
Resolution		18 Bit
Conversion time		max. 2µs (per ADC)
Sampling rate		max. 500ksps (per ADC)
Input band width		0 … 0.6MHz
Measurement range		±10V
Accuracy	INL	max. ±4 LSB
	DNL	max. ±3 LSB
Input resistance		330kΩ, ±2%
Input over-voltage:		±35V
Offset error		adjustable
Offset drift		±30ppm/ºC of full scale range
Connector		4 LEMO female connectors, 2-pole optional: 37-pin D-Sub female connector

Fig. 60 – Pro II-AIn-F-4/18 Rev. E: Specification

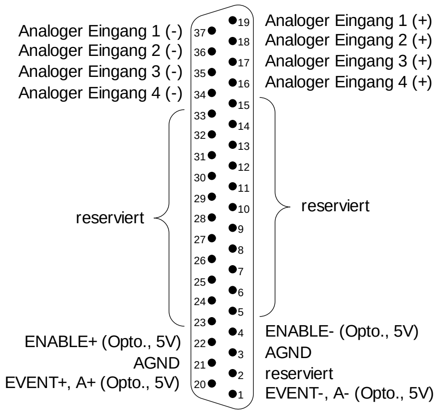


Fig. 61 – Pro II-AIn-F-4/18-D Rev. E: Pin assignment differential

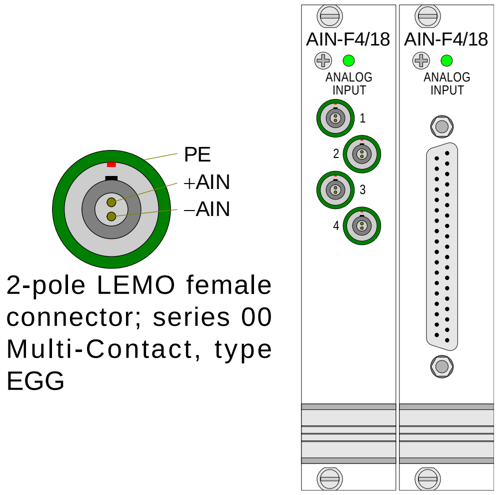


Fig. 62 – Pro II-AIn-F-4/18 Rev. E: Front panels

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure sequence control	**P2_ADCF_Mode**
Do a single conversion – complete or step by step	**P2_ADCF**, **P2_ADCF24** **P2_Start_ConvF**, **P2_Wait_EOCF** **P2_Read_ADCF**, **P2_Read_ADCF24**
Read several values	**P2_Read_ADCF32** **P2_Read_ADCF4**, **P2_Read_ADCF4_24B** **P2_Read_ADCF4_Packed**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv24** **P2_Read_ADC_SConv32**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Stat**
Control input limits	**P2_ADCF_Read_Limit** **P2_ADCF_Set_Limit**
Use LED	**P2_Check_LED**, **P2_Set_LED**

Function	Instructions
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**, **P2_Event2_Config**

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

The analog input module Pro II-AIn-F-8/18 Rev. E has 8 Fast-ADC of 18 Bit and 8 differential inputs. The inputs are galvanically isolated from each other as well as from other modules.

The inputs are available with the following connectors:

– Pro II-AIn-F-8/18-L2: shielded LEMO female connectors, 2-pole, CAMAC European norm.

– Pro II-AIn-F-8/18-D: D-Sub female connector 37-pin.

The module Pro II-AIn-F-8/18 Rev. E has input voltage range of ±10V. The adjustment of gain and offset is made by software. (see chapter 6 "Calibration").

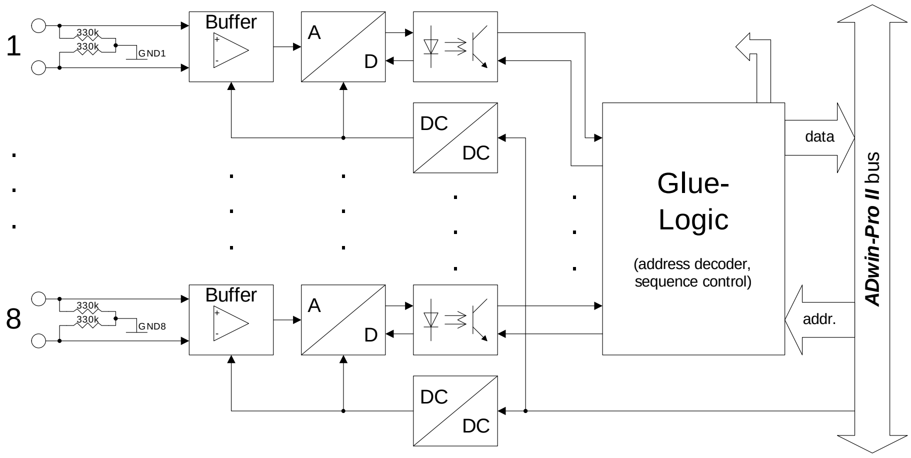


Fig. 63 – Pro II-AIn-F-8/18 Rev. E: Block diagram

Input channels		8 differential, galvanically isolated
Resolution		18 Bit
Conversion time		max. 2µs (per ADC)
Sampling rate		max. 500ksps (per ADC)
Input band width		0 … 0.6MHz
Measurement range		±10V
Accuracy	INL	max. ±4 LSB
	DNL	max. ±3 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30ppm/°C of full scale range
Connector		8 LEMO female connectors, 2-pole optional: 37-pin D-Sub female connector

Fig. 64 – Pro II-AIn-F-8/18 Rev. E: Specification

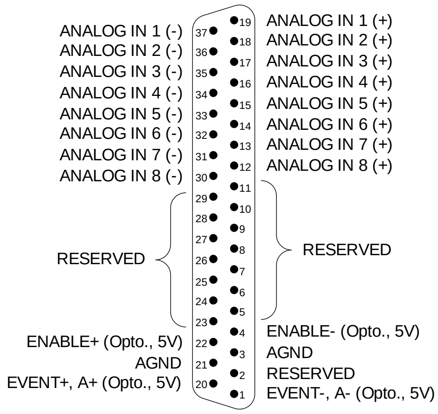


Fig. 65 – Pro II-AIn-F-8/18-D Rev. E: Pin assignment differential

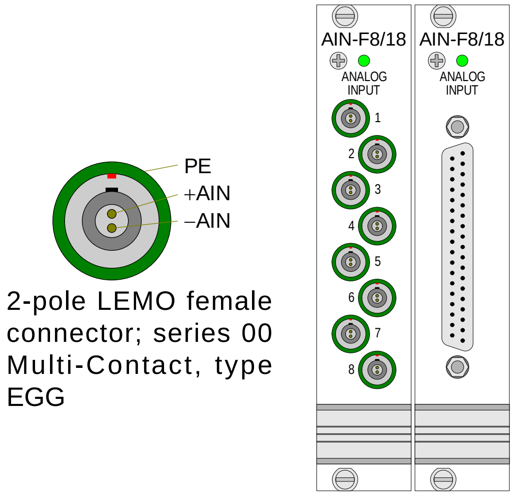


Fig. 66 – Pro II-AIn-F-8/18 Rev. E: Front panels

Programming in ADbasic

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure sequence control	P2_ADCF_Mode
Do a single conversion – complete or step by step	**P2_ADCF**, **P2_ADCF24** **P2_Start_ConvF**, **P2_Wait_EOCF** **P2_Read_ADCF**, **P2_Read_ADCF24**
Read several values	**P2_Read_ADCF32** **P2_Read_ADCF4**, **P2_Read_ADCF4_24B** **P2_Read_ADCF4_Packed** **P2_Read_ADCF8**, **P2_Read_ADCF8_24B** **P2_Read_ADCF8_Packed**
Read value and start new conversion	**P2_Read_ADC_SConv** **P2_Read_ADC_SConv24** **P2_Read_ADC_SConv32**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Stat**
Control input limits	**P2_ADCF_Read_Limit** **P2_ADCF_Set_Limit**

Function	Instructions
Use LED	`P2_Check_LED`, `P2_Set_LED`
Use interrupt and event inputs	`P2_Event_Enable`, `P2_Event_Read` `P2_Event_Config`, `P2_Event2_Config`

5.5 Pro II: Analog Output Modules

This section describes analog output modules for ***ADwin-Pro II***. Analog output modules for ***ADwin-Pro I*** be found in the manual "ADwin-Pro Hardware" from page 62.

Module name	AOut 4/16	AOut 8/16
Revision	E	E
Number DAC	4	8
Resolution [bit]	16	16
max. settling time [µs]	< 3	< 3
Channels sng. end.	4	8
Output voltage	±10V	±10V
Calibration[a]	SW	SW
page	72	74

a. SW: per Software

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

The analog output module Pro II-AOut-4/16 Rev. E has 4 DAC (16 bit) with fixed 1st order low-pass filters (fc = 10MHz).

The output voltage range of the DACs is set to ±10V bipolar and can't be changed. Offset and gain are adjusted by software (see chapter 6 "Calibration").

The outputs are available with the following connectors:

– Pro II-AOut-4/16-L: shielded LEMO female connectors CAMAC European norm.

– Pro II-AOut-4/16-D: D-Sub female connector 37-pin.

Modules with D-Sub female connector have an event input; an event given may be forwarded as trigger signal to the processor module.

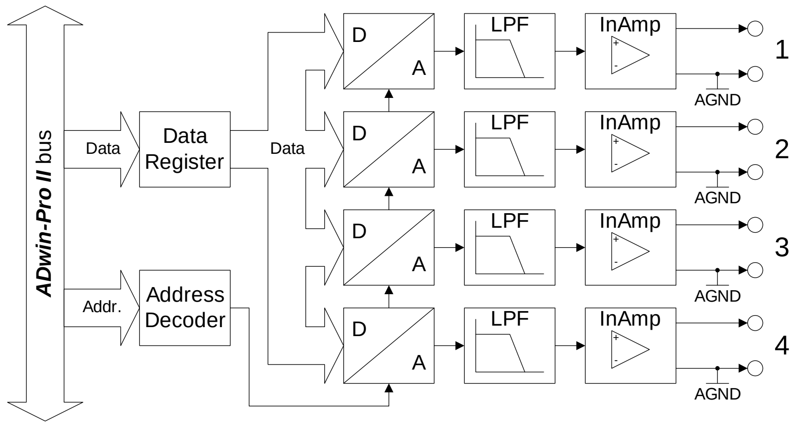


Fig. 67 – Pro II-AOut-4/16 Rev. E: Block diagram

Output channels		4 single-ended
Resolution		16 bit
Settling time to 0.01% FSR		< 3µs
Output voltage		±10V
Output current max.		±5mA per channel for optimal function ±35mA technically possible, short-circuit-proof
Accuracy	INL	±2 LSB typical
	DNL	±1 LSB typical
Offset error		adjustable
Gain error		adjustable
Offset drift		±10 µV/ºC
Connector		4 LEMO female connectors optional: 37-pin D-SUB female connector

Fig. 68 – Pro II-AOut-4/16 Rev. E: Specification

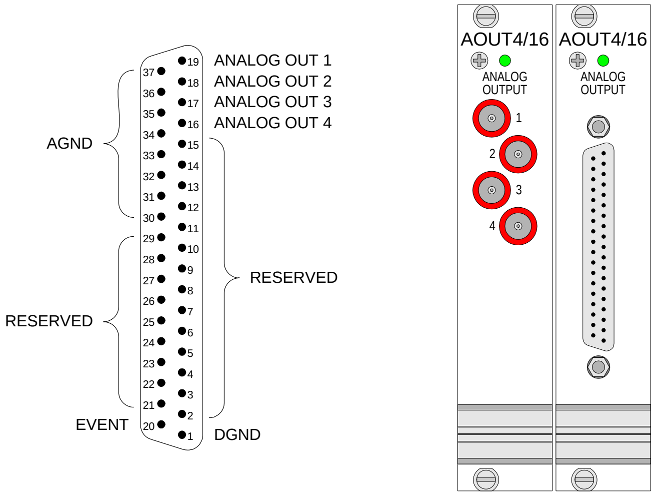


Fig. 69 – Pro II-AOut-4/16 Rev. E: Pin assignment D-Sub and front cover

Programming

The module is comfortably programmed with ***ADbasic*** instructions. The instructions are described in ***ADbasic*** online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Output values	**P2_DAC**, **P2_DAC4**, **P2_DAC4_Packed**
Output values step by step	**P2_Write_DAC**, **P2_Write_DAC4** **P2_Write_DAC4_Packed** **P2_Write_DAC32** **P2_Start_DAC**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Stat**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**

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

The analog output module Pro II-AOut-8/16 Rev. E has 8 DAC (16 bit) with fixed 1st order low-pass filters (fc = 10 MHz).

The output voltage range of the DACs is set to ±10V bipolar and can't be changed. Offset and gain are adjusted by software (see chapter 6 "Calibration").

The outputs are available with the following connectors:

– Pro II-AOut-8/16-L: shielded LEMO female connectors CAMAC European norm.

– Pro II-AOut-8/16-D: D-Sub female connector 37-pin.

Modules with D-Sub female connector have an event input; an event given may be forwarded as trigger signal to the processor module.

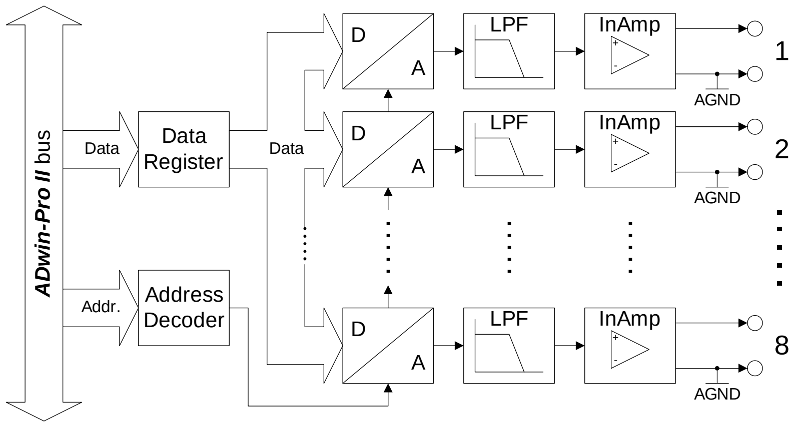


Fig. 70 – Pro II-AOut-8/16 Rev. E: Block diagram

Output channels		8 single-ended
Resolution		16 bit
Settling time to 0.01% FSR		< 3µs
Output voltage		±10V
Output current max.		±5 mA per channel for optimal function ±35 mA technically possible, short-circuit-proof
Accuracy	INL	±2 LSB typical
	DNL	±1 LSB typical
Offset error		adjustable
Gain error		adjustable
Offset drift		±10 µV/ºC
Connector		8 LEMO female connectors optional: 37-pin D-SUB female connector

Fig. 71 – Pro II-AOut-8/16 Rev. E: Specification

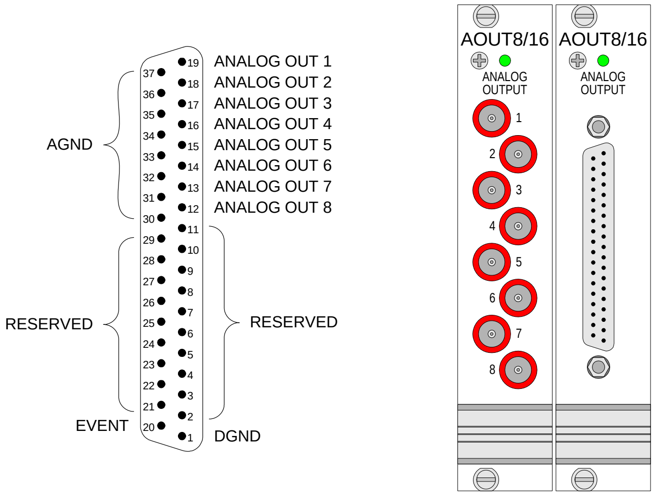


Fig. 72 – Pro II-AOut-8/16 Rev. E: Pin assignment D-Sub and front cover

Programming

The module is comfortably programmed with ***ADbasic*** instructions. The instructions are described in ***ADbasic*** online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Output values	**P2_DAC**, **P2_DAC4**, **P2_DAC4_Packed** **P2_DAC8**, **P2_DAC8_Packed**
Output values step by step	**P2_Write_DAC**, **P2_Write_DAC4** **P2_Write_DAC4_Packed** **P2_Write_DAC8** **P2_Write_DAC8_Packed** **P2_Write_DAC32** **P2_Start_DAC**
Synchronize	**P2_Sync_All**, **P2_Sync_Enable** **P2_Sync_Stat**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**

5.6 Pro II: Digital-I/O Modules

Digital I/O Modules

Module	Rev.	Type	Chan-nels	Input Voltage U_{In} [V]		High Level [mA]	Isolation [V]	Page
DIO-32	E	TTL input / output	32	5	TTL	–	–	77
DIO-32-TiCo	E	TTL input / output, with TiCo processor	32	5	TTL	–	–	77
OPT-16	E	Optocouple input	16	5, 12, 24	DC	–	42	80
OPT-32	E	Optocouple input	32	5 / 12 / 24	DC	–	42	82
REL-16	E	Relay output	16	max. 30	AC / DC	500	42	84
TRA-16	E	Transistor output	16	5...30	DC	200	42	86
PWM-16	E	PWM output signal	16	5	TTL	–	–	88
PWM-16-I	E	PWM output signal	16	5...30	DC	100	42	88

Counter Modules

Module	Rev.	Chan-nels	Counters			Input voltage. U_{In}		Isolation [V]	Page
			No.	Type[a]	Resol. [Bit]	[V]	Type		
CNT-T	E	4	1	U	32	5	TTL	–	91
CNT-I	E	4	1	U	32	5, 12, 24	DC	42	91
CNT-D	E	4 + 2 SSI	1	U	32	5 diff.	RS422/ RS485	–	91

a. U: Universal counter = UD + I + PWM

5.6.1 Pro II-DIO-32 Rev. E

The digital input/output module Pro II-DIO-32 Rev. E provides 32 programmable digital input and output channels at TTL levels. The channels can be configured as blocks of 8 bits as inputs or outputs by *ADbasic* instructions. The channels are configured as inputs after power up.

The module can automatically monitor the edges of input channels which is performed with a frequency of 100 MHz. With every change the current input levels are saved together with a time stamp in a FIFO; up to 511 of those value pairs (input level and time stamp) can be stored. The FIFO data can be read read and processed.
In addition, one can query whether a positive or negative edge has occurred at the input channels.

The variant Pro II-DIO-32-TiCo Rev. E provides in addition the freely programmable *TiCo* processor and 256 MiB memory. The *TiCo* processor has access to all digital input and output channels. Find more information about use and programming of the *TiCo* processor in the *TiCoBasic* manual.

The variant Pro II-DIO-32-TiCo from revision E 03 may output levels at defined points in time to digital outputs as stand-alone. A FIFO serves as buffer where the user-defined level patterns and points in time are stored.

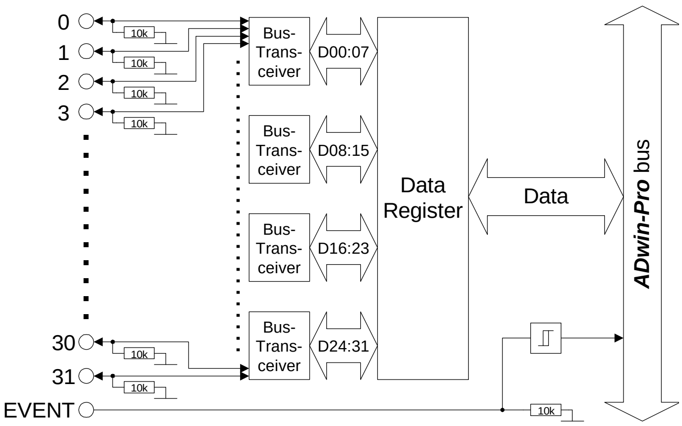


Fig. 73 – Pro II-DIO-32 Rev. E: Block diagram

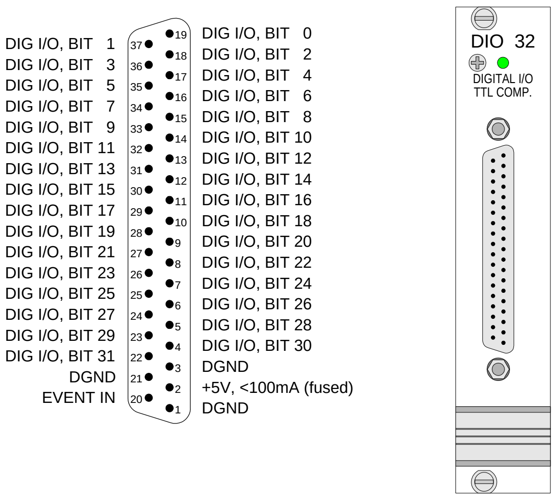


Fig. 74 – Pro II-DIO-32 Rev. E: Front panel and Pin assignment

Digital inputs	TTL logic

Fig. 75 – Pro II-DIO-32 Rev. E: Specification

Input/output channels	32; programmable via software as inputs/outputs in blocks of 8
Pull down resistor	10kΩ
V_{IH}	min. 2V
V_{IL}	max. 0.8V
I_{IH}	max. 1µA
I_{IL}	max. 0.01mA
Voltage range	-0.5V … +5.5V
Output current	max. ±35mA per channel, max. ±70mA per block (8 channels) via V_{CC} or GND
Event input	TTL logic
Power up status	All channels as inputs
Memory size	Pro II-DIO-32: no memory Pro II-DIO-32-TiCo: 256MiByte
Connector	37-pin D-Sub female connector

Fig. 75 – Pro II-DIO-32 Rev. E: Specification

Programming in ADbasic

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure input/outputs	**P2_DigProg**
Query input signals	**P2_Digin_Long**
Use latch register	**P2_Dig_Latch** **P2_Dig_Read_Latch** **P2_Dig_Write_Latch** **P2_Sync_All**
Monitor edges of input channels	**P2_Digin_FIFO_Enable** **P2_Digin_FIFO_Read** **P2_Digin_FIFO_Read_Timer** **P2_Digin_FIFO_Clear** **P2_Digin_FIFO_Full**
Query edges of input channels	**P2_Digin_Edge**
Set and read back output signals	**P2_Digout**, **P2_Digout_Bits** **P2_Digout_Long** **P2_Get_Digout_Long**
Set output signals automatically	**P2_Digout_FIFO_Clear** **P2_Digout_FIFO_Empty** **P2_Digout_FIFO_Enable** **P2_Digout_FIFO_Read_Timer** **P2_Digout_FIFO_Start** **P2_Digout_FIFO_Write**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**

Programming in TiCoBasic

The module variant Pro II-DIO-32-TiCo Rev. E can be programmed with *TiCo-Basic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `DIO32TiCo.inc` contains instructions for the following functions:

Function	Instructions
Configure input/outputs	**DigProg**
Query input signals	**Digin_Long**
Monitor edges of input channels	**Digin_FIFO_Enable** **Digin_FIFO_Read** **Digin_FIFO_Read_Timer** **Digin_FIFO_Clear** **Digin_FIFO_Full**
Query edges of input channels	**Digin_Edge**
Set and read back output signals	**Digout**, **Digout_Bits** **Digout_Set**, **Digout_Reset** **Digout_Long** **Get_Digout_Long**
Set output signals automatically	**Digout_FIFO_Clear** **Digout_FIFO_Empty** **Digout_FIFO_Enable** **Digout_FIFO_Read_Timer** **Digout_FIFO_Start** **Digout_FIFO_Write**
Use LED	**Check_LED**, **Set_LED**
Use interrupt and event inputs	**eVENT_eNABLE**, **Trigger_Event** **Event_Config**

Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions
Data exchange with the *TiCo* processor using global variables and arrays	**P2_TDrv_Init** **P2_GetData_Long**, **P2_Get_Par**, **P2_Get_Par_Block** **P2_SetData_Long**, **P2_Set_Par**, **P2_Set_Par_Block** **P2_Get_TiCo_RingBuffer**, **P2_Set_TiCo_RingBuffer** **P2_RingBuffer_Empty** **P2_RingBuffer_Full**
Control *TiCo* processor	**P2_TiCo_Reset**, **P2_TiCo_Start**, **P2_TiCo_Stop** **P2_Get_TiCo_Bootloader_Status** **P2_Get_TiCo_Status**, **P2_Workload**
Control *TiCo* processes	**P2_Process_Status** **P2_TiCo_Get_Processdelay** **P2_TiCo_Set_Processdelay** **P2_TiCo_Start_Process** **P2_Tico_Stop_Process**
Transfer *TiCo* programs	**P2_TiCo_Flash**, **P2_TiCo_Load**

5.6.2 Pro II-OPT-16 Rev. E

The input module Pro II-OPT-16 Rev. E provides 16 channels of optically isolated digital inputs. The input voltage range can be set by jumpers (5V, 12V, 24V). The default setting of the input voltage range is 24V. The switching time of only 100ns allows the sampling of high-speed digital inputs.

Each channel is optically isolated from the system circuitry and from the other inputs. The event-input is optically isolated from the system as well.

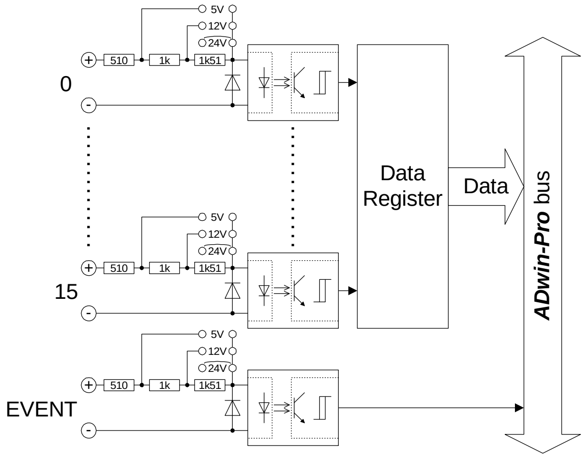


Fig. 76 – Pro II-OPT-16 Rev. E: Block diagram

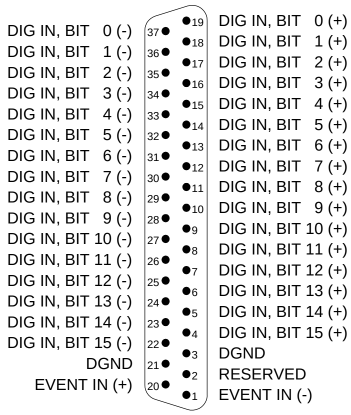


Fig. 77 – Pro II-OPT-16 Rev. E: Pin assignment

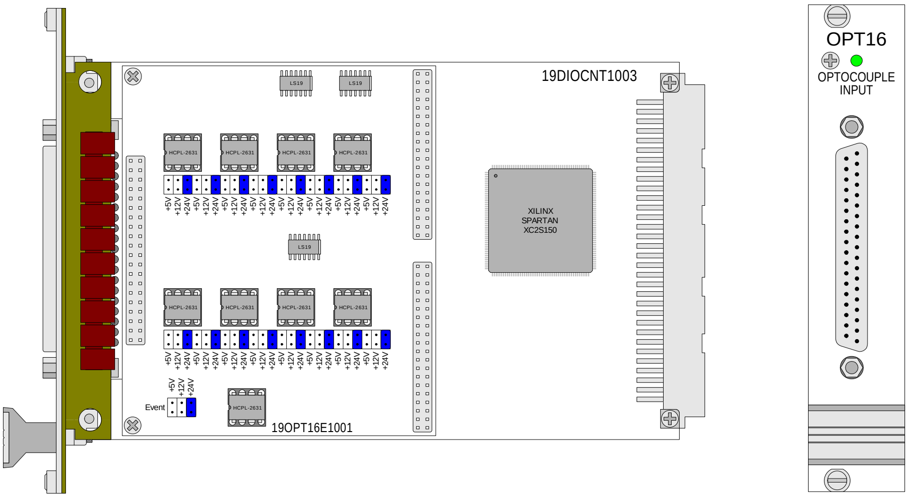


Fig. 78 – Pro II-OPT-16 Rev. E: Board and front panel

Input channels	16		
Event inputs	1		
Input current	typ. 3.5mA / max. 7.5mA		
Input voltage range (selectable via jumpers)	0…5V	0…12V	0…24V
Switching threshold for 0-low	0…0.8V	0…1.6V	0…3.2V
Switching threshold for 1-high	4.5…5V	10…12V	20…24V
Input over-voltage	-5V … 8V	-5V … 16V	-5V … 30V
Switching time	100ns		
Isolation	42V channel-channel / channel-GND		
Connector	37-pin D-Sub female connector		

Fig. 79 – Pro II-OPT-16 Rev. E: Specification

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Query input signals	**P2_Digin_Edge** **P2_Digin_Long**
Use latch register	**P2_Dig_Latch**, **P2_Dig_Read_Latch** **P2_Sync_All**
Monitor edges of input channels	**P2_Digin_FIFO_Enable** **P2_Digin_FIFO_Read** **P2_Digin_FIFO_Read_Timer** **P2_Digin_FIFO_Clear** **P2_Digin_FIFO_Full**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**

5.6.3 Pro II-OPT-32 Rev. E

The input module Pro II-OPT-32 Rev. E provides 32 channels of optically isolated digital inputs.

The input voltage range is fixed (ordering options: 5V, 12V, 24V) which is also true for the event input. The switching time of only 100ns allows the sampling of high-speed digital inputs.

The module can automatically monitor the edges of input channels. With every change the current input levels are saved together with a time stamp in a FIFO. The FIFO data can be read read and processed.

Each channel is optically isolated from the system circuitry and from the other inputs. The event-input is optically isolated from the system as well.

The channels are single ended and share the same ground potential (ext. GND) which has to be provided on the D-Sub connector.

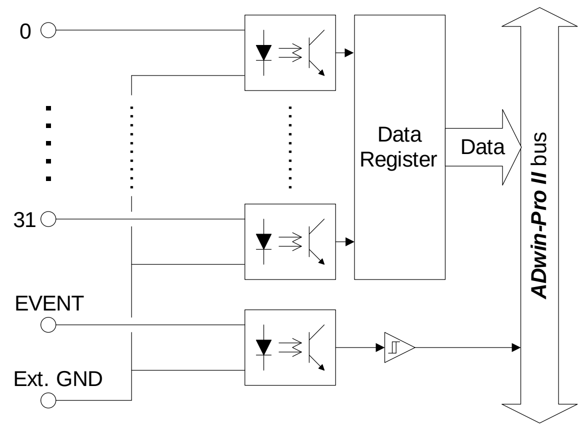


Fig. 80 – Pro II-OPT-32 Rev. E: Block diagram

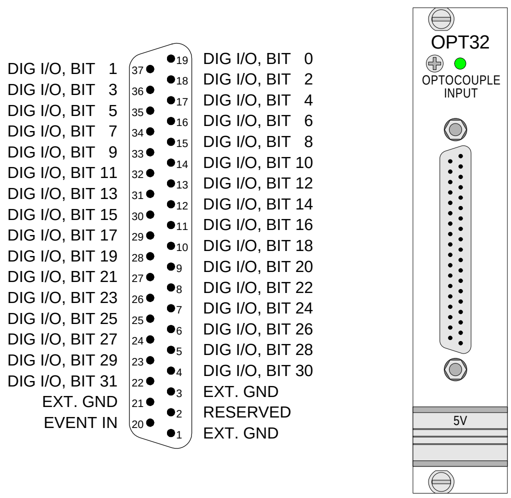


Fig. 81 – Pro II-OPT-32 Rev. E: Pin assignment and front panel

Input channels	32 single ended		
Event inputs	1		
Input current	typ. 3.5mA / max. 7.5mA		
Input voltage range (ordering option)	0…5V	0…12V	0…24V
Switching threshold for 0-low	0…0.8V	0…1.6V	0…3.2V
Switching threshold for 1-high	4.5…5V	10…12V	20…24V
Input over-voltage	-5V … 8V	-5V … 16V	-5V … 30V
Switching time	100ns		
Isolation	42V channel-channel / channel-GND		
Connector	37-pin D-Sub female connector		

Fig. 82 – Pro II-OPT-32 Rev. E: Specification

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Query input signals	**P2_Digin_Edge** **P2_Digin_Long**
Use latch register	**P2_Dig_Latch**, **P2_Dig_Read_Latch** **P2_Sync_All**
Monitor edges of input channels	**P2_Digin_FIFO_Enable** **P2_Digin_FIFO_Read** **P2_Digin_FIFO_Read_Timer** **P2_Digin_FIFO_Clear** **P2_Digin_FIFO_Full**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**

The instructions are described in the *ADbasic* online help and in the Pro Software manual.

5.6.4 Pro II-REL-16 Rev. E

The Pro II-REL-16 Rev. E output module provides 16 isolated relay outputs. Each channel is isolated from system circuitry and other output channels. The event-output is optically isolated from the system circuitry.

The module is equipped with normally open contacts, as an option also normally closed contacts are available.

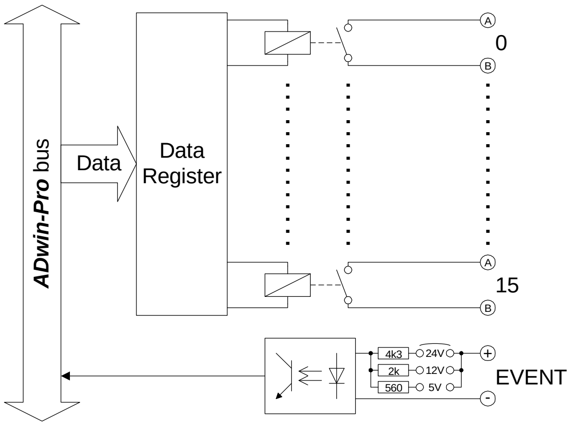


Fig. 83 – Pro II-REL-16 Rev. E: Block diagram

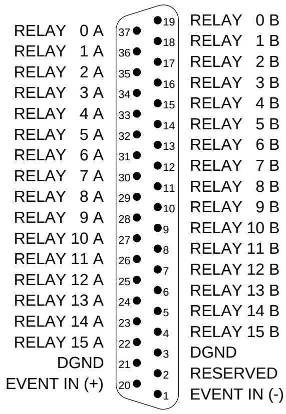


Fig. 84 – Pro II-REL-16 Rev. E: Pin assignment

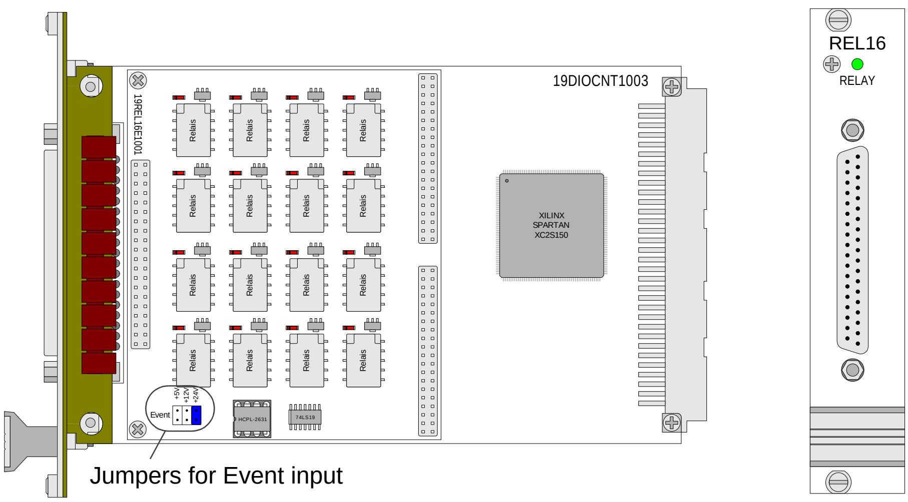


Fig. 85 – Pro II-REL-16 Rev. E: Board and front panel

Output channels	16
Switch voltage	30V AC/DC Maximum
Switch current	max. 500mA per channel
Contact	1 per channel, normally open (optional: normally closed)
Operate time	4ms
Release time	3ms
Bounce time	2ms
Event inputs	1
Isolation	42V channel-channel / channel-GND
Event input voltage	5V, 12V, 24V (selectable via jumpers)
Power up status	low (normally open contact: open / normally closed contact: closed)
Connector	37-pin D-Sub female connector

Fig. 86 – Pro II-REL-16 Rev. E: Specification

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Set and read back output signals	**P2_Digout**, **P2_Digout_Long** **P2_Digout_Bits** **P2_Get_Digout_Long**
Use latch register	**P2_Dig_Latch**, **P2_Dig_Write_Latch** **P2_Sync_All**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**

5.6.5 Pro II-TRA-16 Rev. E

The output module Pro II-TRA-16 Rev. E provides 16 channels of isolated transistor outputs. There are 2 module variants:

– Pro II-TRA-16 Rev. E: The outputs switch to V_{CC}.

– Pro II-TRA-16-G Rev. E: The outputs switch to ground.

The switching voltage V_{CC} has to be provided by an external power supply. The channels as well as the event-input are optically isolated from system circuitry.

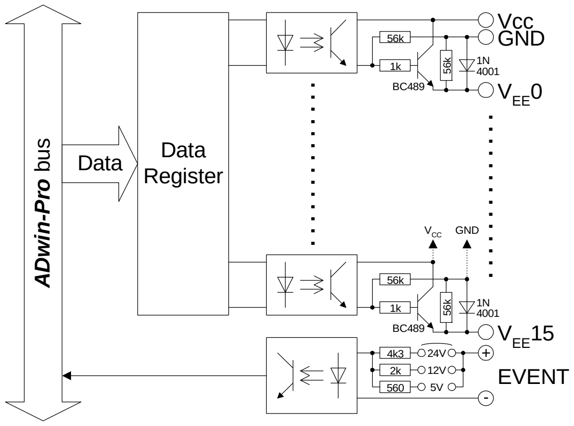


Fig. 87 – Pro II-TRA-16 Rev. E: Block diagram

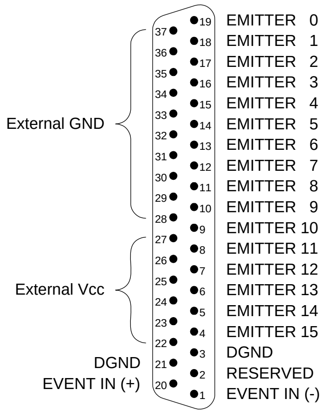


Fig. 88 – Pro II-TRA-16 Rev. E: Pin assignment

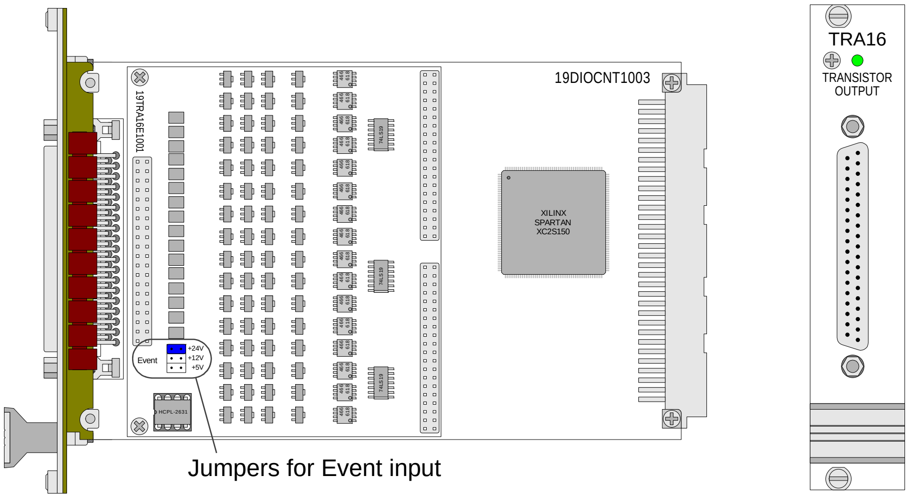


Fig. 89 – Pro II-TRA-16 Rev. E: Board and front panel

Output channels	16
Switching voltage V_{CC}	5…30V DC with external power supply
Switching current	Pro II-TRA-16: 200mA max. per channel Pro II-TRA-16-G: 100mA max. per channel
Voltage drop	0.5V
Switching time	2.5µs
Event input	1
Isolation	42V channel-channel / channel-GND
Event input voltage	5V, 12V, 24V (selectable via jumpers)
Power up status	low (GND external)
Connector	37-pin D-Sub female connector

Fig. 90 – Pro II-TRA-16 Rev. E: Specification

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Set and read back output signals	**P2_Digout**, **P2_Digout_Long** **P2_Digout_Bits** **P2_Get_Digout_Long**
Use latch register	**P2_Dig_Latch**, **P2_Dig_Write_Latch** **P2_Sync_All**
Use LED	**P2_Check_LED**, **P2_Set_LED**
Use interrupt and event inputs	**P2_Event_Enable**, **P2_Event_Read** **P2_Event_Config**

5.6.6 Pro II-PWM-16 Rev. E

The output module Pro II-PWM-16 Rev. E generates pulse width modulated signals (PWM signals) at 16 outputs. Each (PWM) signal can be configured individually via software; that means, they can be configured separately.

The module is available as version Pro II-PWM-16-I Rev. E, too. On this module, the outputs are optically isolated against the system circuit and against each other. The event input, too, is optically isolated from the system circuit.

The output signals are clocked with a reference clock speed of 100 MHz.

The lowest output frequency is about 0.6 Hz.

The highest output frequency where the duty cycle can be still defined in 1%-steps, is 1000 kHz.

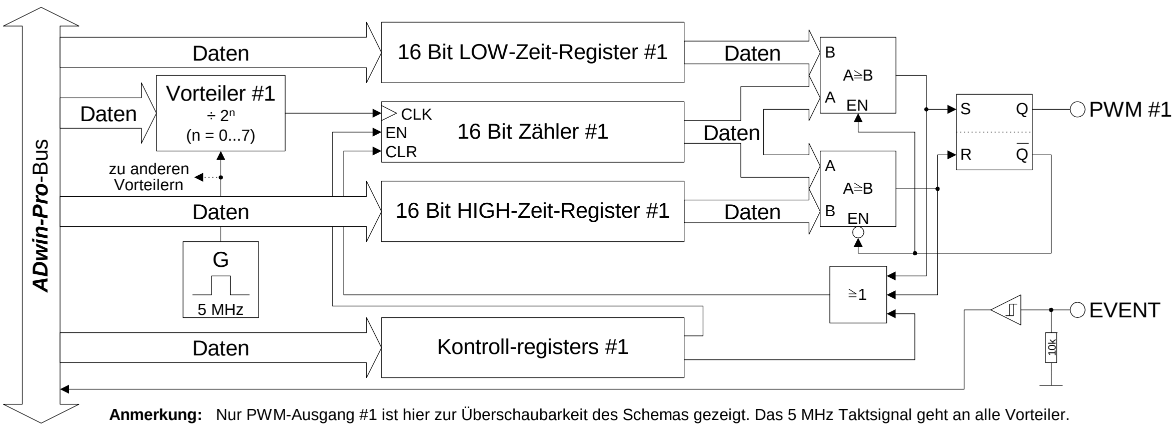


Fig. 91 – Pro II-PWM-16 Rev. E: Block diagram

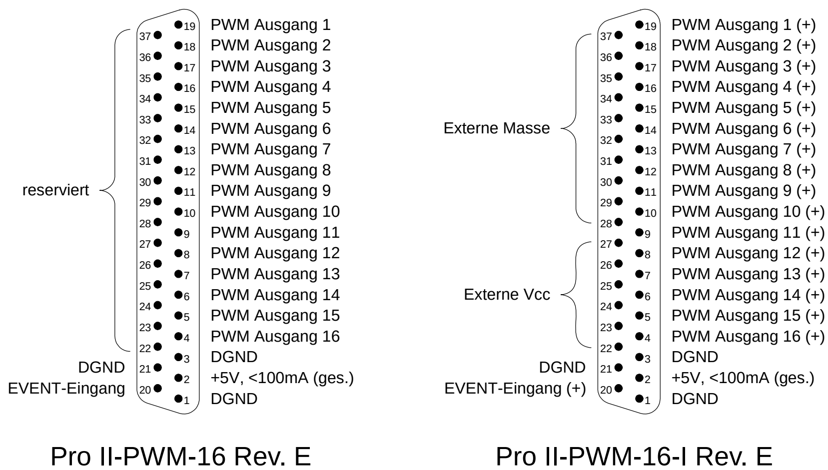


Fig. 92 – Pro II-PWM-16 Rev. E: Pin assignment

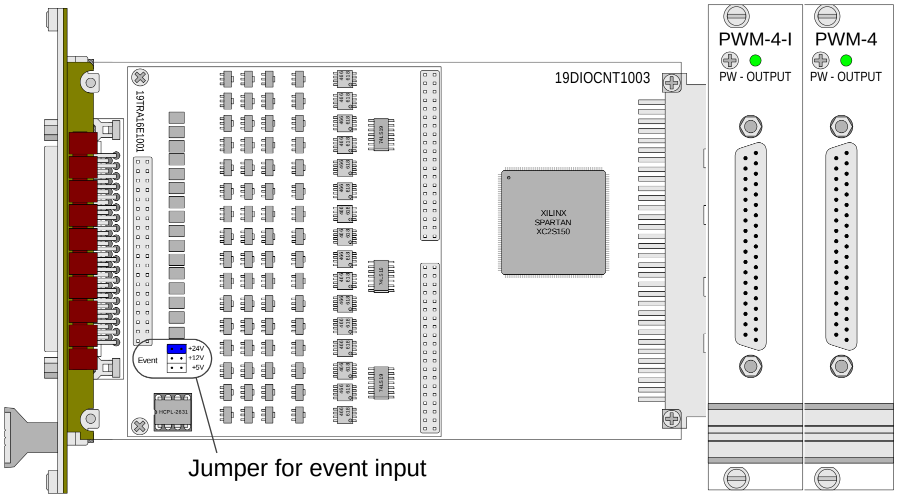


Fig. 93 – Pro II-PWM-16-I Rev. E: PlatineBoard and front panel

Output channels	16 PWM channels
Outputs	TTL
Counter / register width	32 bit
Reference clock	100MHz
Event input	1 input, positive TTL
Connector	37-pin D-Sub female connector
Pro II-PWM-16 Rev. E	
V_{OH}	2.4V min.
V_{OL}	0.8V max.
Output current	5mA per channel max.
Isolation	No (see page 88)
Pro II-PWM-16-I Rev. E	
Output voltage	5…30V DC via external power supply
Output current	100mA max. per channel
Voltage drop	0.5V max.
Switching time	without load: 3µs with load: 1…2µs
Event input voltage	5V, 12V, 24V (selectable via jumpers)
Isolation	42V channel-channel / channel-GND

Fig. 94 – Pro II-PWM-16 Rev. E: Spezifikation

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Initialize module	P2_PWM_Init P2_PWM_Standby_Value P2_PWM_Enable P2_PWM_Reset
Output and read back PWM signals	P2_PWM_Write_Latch P2_PWM_Latch, P2_Sync_All P2_PWM_Get_Status
Use LED	P2_Check_LED, P2_Set_LED
Use interrupt and event inputs	P2_Event_Enable P2_Event_Config P2_Event_Read

5.6.7 Pro II-CNT-x Rev. E

The module Pro II-CNT-x Rev. E provides 4 configurable multi-purpose counter blocks. Each counter block contains two 32-bit counters: First an up/down counter or four edge evaluation for connection of enceoders. Second, a PWM counter to evaluate high and low times, duty cycle, or frequency. Both counters of a block can be operated in parallel.

The following module versions are avaiable:

– *Pro II-CNT-T*: Counter inputs with TTL logic. The module is equipped with a *TiCo* processor.

– *Pro II-CNT-D*: Differential counter inputs; in addition, the counter contains 2 SSI decoders and a *TiCo* processor.

– *Pro II-CNT-I*: Counter inputs are optically isolated. The module is equipped with a *TiCo* processor.

  The counter inputs are optically isolated from the system circuit. The event input, too, is optically isolated from the system circuit.

  The input voltage range of the counter and event inputs can be set by jumpers to 0…5V, 0…12V or 0…24V. The default setting is 0…24V.

The *TiCo* processor is freely programmable and has access to all counter functions. Find more information about use and programming of the *TiCo* processor in the *TiCoBasic* manual.

Up/down counter

The up/down counter of a block can be operated in 2 modes:

– Clock / direction (CLK and DIR signals)

  A negative edge at the CLK input is the counting impulse for the 16-bit counter. The DIR signal sets the counting direction, TTL high means a count-up, TTL low means a count-down.

  You can invert the signals at the inputs CLK and DIR via software (instruction **P2_Cnt_Mode**) and thus change both the triggering edge and the counting direction.

  You can latch the counter values program-controlled or you can influence the counter by an external CLR/LATCH signal.

  Depending on the programming the CLR/LATCH signal has either the effect that the counter values are cleared (CLR) or that the counter values are latched (LATCH). This function will only be effective when it is released by the instructions **P2_CNT_CLEAR_ENABLE** or **P2_CNT_LATCH_ENABLE**.

  The counter is cleared or latched with a rising adge at input CLR/LATCH. During the latch process the frequency of the measurement can be determined by getting the difference of two read latch values, because this difference defines the number of pulses between the two reading processes.

– Four edge evaluation (A and B signals)

  The four edge evaluation changes the signals (which should be 90° phase-shifted) of a connected incremental encoder at the inputs A and B to CLK and DIR signals. For this you have to program the inputs correspondingly (see "*ADwin-Pro* System Description, Programming in *ADbasic*").

  Since every edge of the A and B signals generates a count impulse, the resolution is increased by factor 4. If the encoder has a reference signal, it can be used to clear or latch the counter (after release of the CLR or LATCH input). The counter is cleared when the signals A, B and CLR are on logic "1" (software-selectable: clear, when only the CLR signal is on logic "1").

PWM counter

The PWM counter of the counter block analyzes the signals at the PWM inputs. Via software instructions the following data can be read directly:

– frequency and duty cycle

– high and low time

Event input

This input, as far as it has been released, can start an externally triggered ***ADbasic*** process.

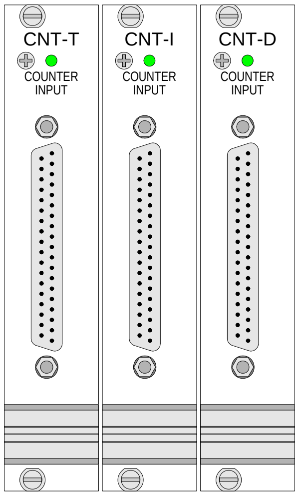


Fig. 95 – Pro II-CNT-x Rev. E: Front panel

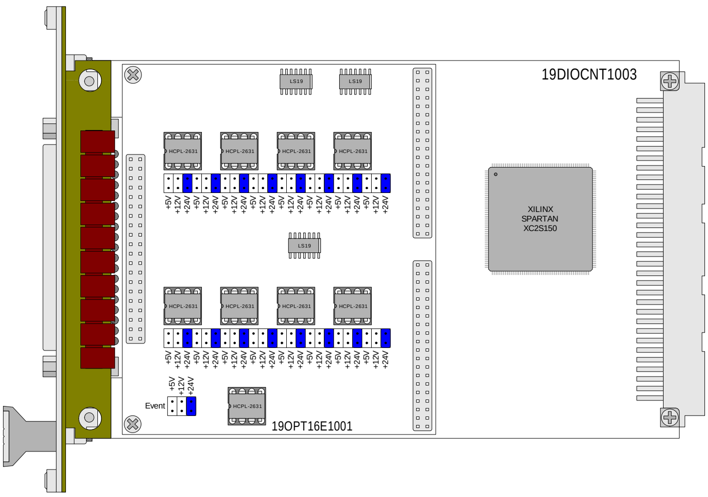
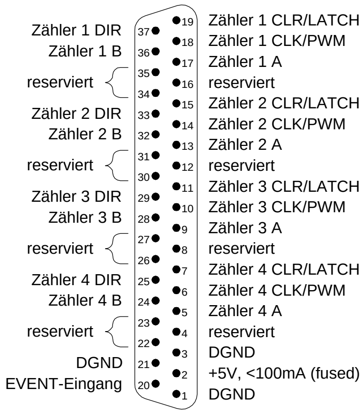
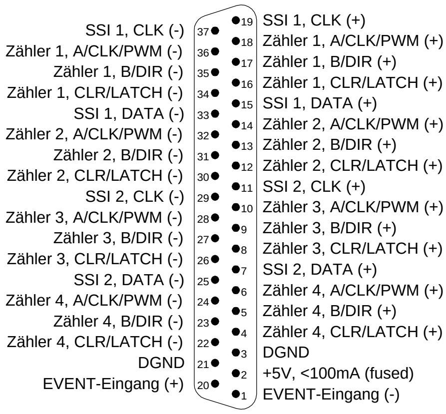


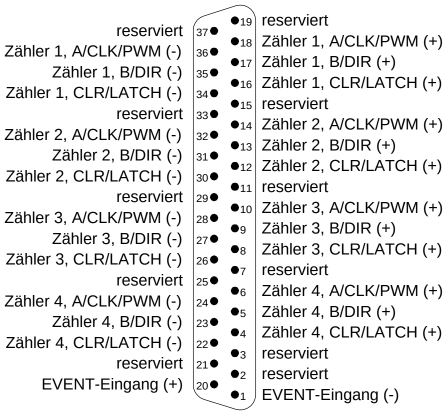
Fig. 96 – Pro II-CNT-I Rev. E: Board with jumpers



Pro II-CNT-T



Pro II-CNT-D



Pro II-CNT-I

Counter	4 multi-purpose counters Cnt-D: 2 SSI decoders in addtion
Counter resolution	32 bit
Event input	1
Reference clock	50 MHz
Clock frequency four edge evaluation	12.5 MHz max. (at 90° phase-shift of the signals)
Clock frequency up/down counter	CNT-T: 25 MHz max. CNT-I, CNT-D: 15 MHz max.
Reference frequency PWM analysis	100 MHz
Connector	37-pin D-Sub female connector
Power consumption	CNT-T approx. 150 mA CNT-D approx. 200 mA CNT-I approx. 200 mA

Fig. 97 – Pro II-CNT-x Rev. E: General specification

<table>
<tr><th colspan="4">Pro II-CNT-T</th></tr>
<tr><td>Input / output level</td><td colspan="3">TTL logic</td></tr>
<tr><td>Event input</td><td colspan="3">TTL logic</td></tr>
<tr><td>Isolation</td><td colspan="3">none</td></tr>
<tr><th colspan="4">Pro II-CNT-D</th></tr>
<tr><td>Input / output level</td><td colspan="3">compatible to RS422/485 (5V differential, 120 Ω bus termiinating resistor)</td></tr>
<tr><td>Event input</td><td colspan="3">1 differential
(singleded operation possible)</td></tr>
<tr><td>Clock frequency SSI deccoder (CLK)</td><td colspan="3">2MHz max.</td></tr>
<tr><td>Isolation</td><td colspan="3">none</td></tr>
<tr><th colspan="4">Pro II-CNT-I</th></tr>
<tr><td>Input current</td><td colspan="3">typ.3.5mA / max. 7.5mA</td></tr>
<tr><td>input voltage range
(selectable via jumpers)</td><td>0…5V</td><td>0 … 12V</td><td>0…24V</td></tr>
<tr><td>reliable switching threshold[1] for 0 (low)</td><td>0…0.8V</td><td>0…1.6V</td><td>0…3.2V</td></tr>
<tr><td>reliable switching threshold[1] for 1 (high)</td><td>4.5…5V</td><td>10…12V</td><td>20…24V</td></tr>
<tr><td>Input over-voltage</td><td>8V</td><td>16V</td><td>30V</td></tr>
<tr><td>Negative voltage</td><td colspan="3">-5V for all ranges</td></tr>
<tr><td>Switching time</td><td colspan="3">100ns</td></tr>
<tr><td>Isolation</td><td colspan="3">42V channel-channel / channel-GND</td></tr>
</table>

Fig. 98 – Pro II-CNT-x Rev. E: Specification

Programming in ADbasic

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Configure counters	**P2_Cnt_Enable**, **P2_Cnt_Mode**
Use counters	**P2_Cnt_Clear**, **P2_Cnt_Get_Status** **P2_Cnt_Latch** **P2_Cnt_Read**, **P2_Cnt_Read4** **P2_Cnt_Read_Latch** **P2_Cnt_Read_Latch4**
Use PWM counters	**P2_Cnt_PW_Enable**, **P2_Cnt_PW_Latch** **P2_Cnt_Get_PW**, **P2_Cnt_Get_PW_HL**
Use SSI decoders (CNT-D only)	**P2_SSI_Mode**, **P2_SSI_Set_Bits** **P2_SSI_Set_Clock** **P2_SSI_Set_Delay**, **P2_SSI_Read** **P2_SSI_Read2** **P2_SSI_Start**, **P2_SSI_Status**
Use event inputs	**P2_Event_Enable**, **P2_Event_Config** **P2_Event_Read**

1. A low/high signal will reliably be recognized in the indicated voltage ranges. But the switching process can also be effected outside these voltage ranges.

Function	Instructions
Use LED	**P2_Check_LED**, **P2_Set_LED**
Synchronize	**P2_Sync_All**

Programming in TiCoBasic

The module can be programmed with *TiCoBasic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `Cnt_TiCo.inc` contains instructions for the functions:

Function	Instructions
Configure counters	**Cnt_Enable**, **Cnt_Mode**
Use counters	**Cnt_Clear**, **Cnt_Get_Status** **Cnt_Latch**, **Cnt_Sync_Latch** **Cnt_Read**, **Cnt_Read_Latch** **Cnt_Read_Int_Register**
Use PWM counters	**Cnt_PW_Enable**, **Cnt_PW_Latch** **Cnt_Get_PW_HL**
Use SSI decoders (CNT-D only)	**SSI_Mode**, **SSI_Set_Bits** **SSI_Set_Clock** **SSI_Set_Delay**, **SSI_Read** **SSI_Start**, **SSI_Status**
Use event inputs	**Event_Enable**, **Event_Config** **Trigger_Event**
Use LED	**Check_LED**, **Set_LED**

Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions
Data exchange with the *TiCo* processor using global variables and arrays	**P2_TDrv_Init** **P2_GetData_Long**, **P2_Get_Par**, **P2_Get_Par_Block** **P2_SetData_Long**, **P2_Set_Par**, **P2_Set_Par_Block** **P2_Get_TiCo_RingBuffer**, **P2_Set_TiCo_RingBuffer** **P2_RingBuffer_Empty** **P2_RingBuffer_Full**
Control *TiCo* processor	**P2_TiCo_Reset**, **P2_TiCo_Start**, **P2_TiCo_Stop** **P2_Get_TiCo_Bootloader_Status** **P2_Get_TiCo_Status**, **P2_Workload**
Control *TiCo* processes	**P2_Process_Status** **P2_TiCo_Get_Processdelay** **P2_TiCo_Set_Processdelay** **P2_TiCo_Start_Process** **P2_Tico_Stop_Process**
Transfer *TiCo* programs	**P2_TiCo_Flash**, **P2_TiCo_Load**

5.7 Pro II: Extension and Interface Modules

Modul	RTD-8	TC-8-ISO
Revision	E	E
Function	temperature measurement	thermo couple interface
Measuring	2 wire, 3 wire or 4 wire	–
Temp. range	-200°C…+700°C	B: 250°C … 1820°C E: -200°C…1000°C J: -210°C…1250°C K: -200°C … 1372°C N: -200°C…1300°C R: -50°C … 1768°C S: -50°C … 1768°C T: -270°C … 400°C
Precision	±0.05…0.1K	±1K
Channels	8	8
Page	97	101

Module	CAN-2	CANL-2
Revision	E	E
Type	CAN interface	
CAN-Version	High speed	Low speed
Interfaces	2	
Page	103	

Module	RSxxx-2	RSxxx-4
Revision	E	E
Type	RSxxx interface	
RSxxx version	RS232 RS485	RS232 RS485
Interfaces	2	4
Data exchange rate [kBaud]	0.035… 2304	0.035… 2304
Page	109	

Module	LIN2	Profi-SL	EtherCAT-2	FlexRay-2
Revision	E	E	E	E
Function	LIN bus interface	Profibus interface	EtherCAT interface	FlexRay interface
Interfaces	2	1	1	2
Page	113	114	118	120

5.7.1 Pro II-RTD-8 Rev. E

The module Pro II-RTD-8 Rev. E has 8 inputs for connecting platinum resistance temperature detectors of the type Pt 100, Pt500, Pt1000, or Ni100. The inputs are connected via a multiplexer with an ADC. The maximum possible measurement range is -200°C…+700°C, depending on the temperature sensor (see data sheets of manufacturer).

There are 2 module variants:

– Pro II-RTD-8 Rev. E: Module inputs on a 37-pin D-Sub connector.

– Pro II-RTD-8-L Rev. E: Module inputs on Lemo connectors. Pin assignment see fig. 100.

Measurements can be done with 2, 3 or 4 wire technique; input circuit see fig. 99. Measuring method and sensor type are set vie software.
Measuring methods and wiring between sensor and module are described starting from page 99.

Measurements are performed automatically by a sequence control according to the settings. Thus, the *ADwin* CPU will only have to read the measured values.

Input signals run through a second order low-pass filter with 25 kHz. Additionally, you can use the instruction **P2_RTD_Channel_Config** to filter a single frequency from the digital signals; the measurement value is calculated as mean value of several measurements being done in defined sampling intervals.

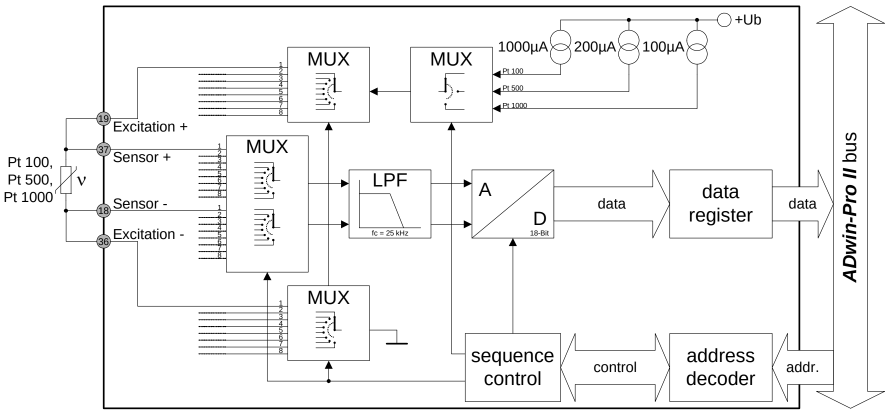


Fig. 99 – Pro II-RTD-8 Rev. E: Block diagram

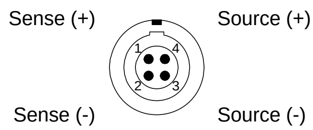


Fig. 100 – Pro II-RTD-8 Rev. E: Pin assignment Lemo connector

Inputs	8
Measurement	2 wire, 3 wire or 4 wire
Multiplexer settling time	100 µs
Input filter	25 kHz (2nd order)
Max. measurement range	-200°C…+700°C

Fig. 101 – Pro II-RTD-8 Rev. E: Specification

Accuracy	PT100: ±0.05K PT500: ±0.1K PT1000: ±0.1K Ni100: ±0.05K
Temp. resolution	0.015K
Drift	±10 ppm/K
$I_1 = I_2$	PT100: 1 mA PT500: 0.2 mA PT1000: 0.1 mA Ni100: 1 mA
Connector	37-pin D-Sub female connector or Lemo connectors

Fig. 101 – Pro II-RTD-8 Rev. E: Specification

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Range	Instruction
Configure module	`P2_RTD_Config` `P2_RTD_Channel_Config`

Access temperature channels	`P2_RTD_Start`
	`P2_RTD_Status`
	`P2_RTD_Read`
	`P2_RTD_Read8`

Convert measured value	`P2_RTD_Convert`
Use LED	`P2_Check_LED`, `P2_Set_LED`

Measurement Method

You can choose one of three measurement methods: 2 wire measurement, 3 wire measurement or 4 wire measurement.

– 2 wire measurement

Please pay attention to a very short connection with low impedance between the Pt sensor and the module input, because the voltage drop gets added to the measured voltage.

This is the reason why this measurement method is in general not to be recommended for precise measurements.

Running a 2 wire measurement on channel n, you connect the sensor to the inputs `Excitation +` and `Excitation -`. For channel 1 it would be pins 19 and 36 (see fig. 102) on the D-Sub connector.

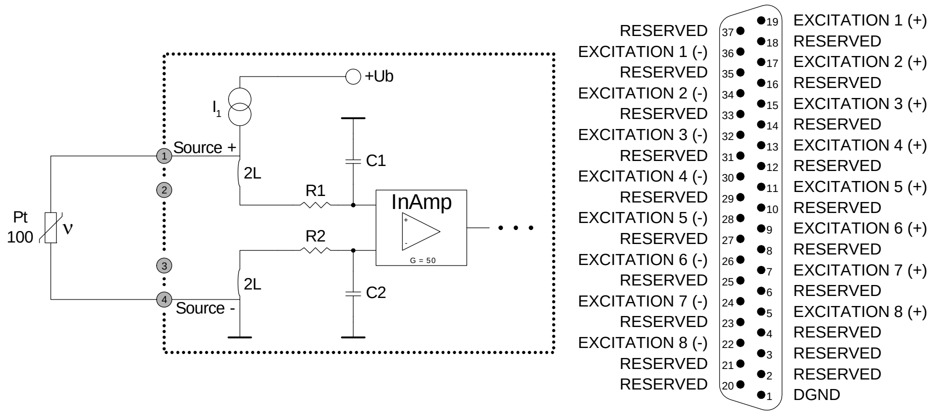


Fig. 102 – Pro II-RTD-8 Rev. E:
Block diagram and pin assignment with 2 wire measurement

– 3 wire measurement

In order to avoid the disadvantages of the 2 wire measurement, the voltage drop in the measurement lines is here compensated by a second voltage source I2.

To keep the measurement error as small as possible, the resistance value of the three measurement lines from the Pt sensor to the module input should be identical.

With a 3 wire measurement, you need 2 measurements for each measurement value. Thus, a 3 wire measurement requires double the time of 2 or 4 wire measurement.

Running a 3 wire measurement on channel n, you connect the sensor to the inputs `Excitation +`, `Sensor -` and `Excitation -`. For channel 1 it would be pins 18, 19 and 36 (see fig. 103) on the D-Sub connector.

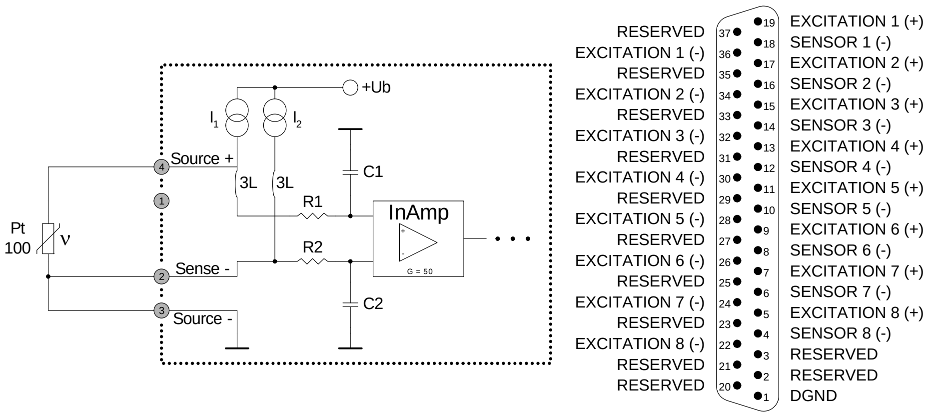


Fig. 103 – Pro II-RTD-8 Rev. E:
Block diagram and pin assignment with 3 wire measurement

– 4 wire measurement

The voltage drop at the Pt sensor is directly avoided with high impedance at the PCB by the two "sensor" inputs. The resistance of the measurement lines does not have an effect here any longer and need therefore not be compensated.

Running a 4wire measurement on channel n, you connect the sensor to the inputs `Excitation +`, `Sensor +`, `Sensor -` and `Excitation -`. For channel 1 it would be pins 18, 19, 36 and 37 (see fig. 104) on the D-Sub connector.

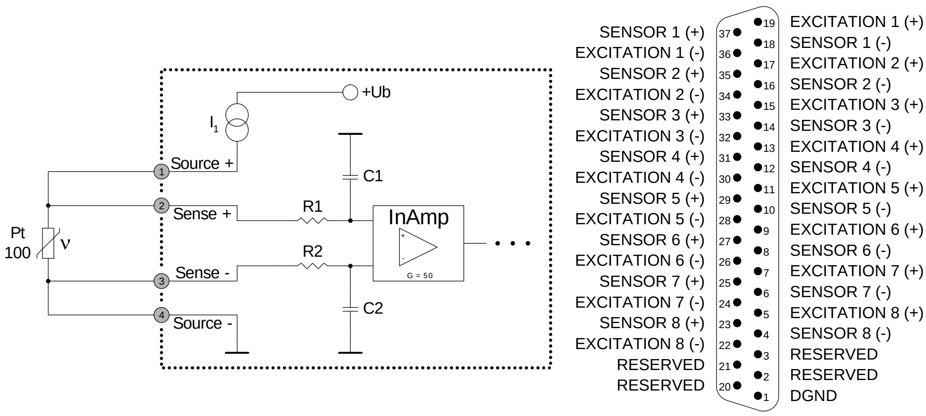


Fig. 104 – Pro II-RTD-8 Rev. E:
Block diagram and pin assignment with 4 wire measurement

5.7.2 Pro II-TC-8 ISO Rev. E

The module Pro II-TC-8 ISO Rev. E has 8 inputs for thermocouples and can be operated with thermocouple types B, E, J, K, N, R, S or T. For each channel the thermoelectric voltage (with or without cold junction correction) or the temperature may be queried separately via software.

Each channel is equipped with a separate ADC. The module provides a common cold junction compensation for all temperature inputs.

The jumper position (see fig. 107, left side) sets for each channel separately if the channel potentials are separated from each other:

– Position right: The channel potentials are separated from each other (default).

– Position left: The channel's negative input is connected to ground.

Input signals at the ADCs are digitzed at a stepwise adjustable sample rate. As soon as a value is queried via software, the module calculates the thermoelectric voltage or the temperature in °Celsius or °Fahrenheit from the last measurement value. All calculation is based upon the norm IEC 584-1. The thermoelectric voltage without cold junction correction can be returned, too.

Calibration of the module is performed by the manufacturer. If needed, please send the module to the address given on the back of the cover page.

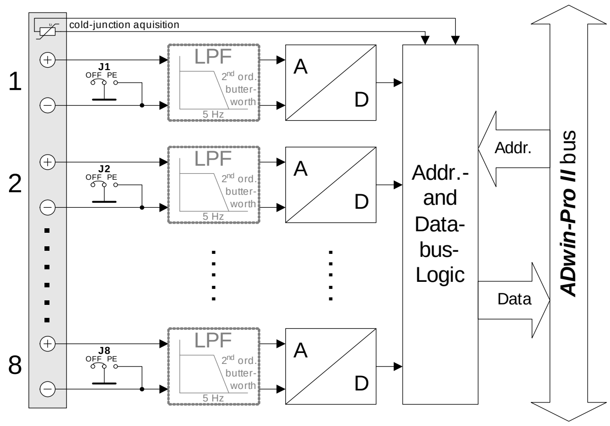


Fig. 105 – Pro II-TC-8 ISO Rev. E: Block diagram

Input channels	8
Sample rate	7Hz … 3500Hz
Thermocouple types, measuring range and accuracy	B: 250°C … 1820°C; ±5°C E: -200°C…1000°C; ±1°C J: -210°C…1250°C; ±1°C K: -200°C … 1372°C; ±1°C N: -200°C…1300°C; ±2°C R: -50°C … 1768°C; ±3°C S: -50°C … 1768°C; ±3°C T: -270°C … 400°C; ±1°C
Resolution	0.1°C
Input resistance	10MΩ
Input over-voltage	±20V
Offset drift	±30ppm/ºK of full scale range
Connector	Omega Subminiature Connector, Type SMP

Fig. 106 – Pro II-TC-8 ISO Rev. E: Specifikation

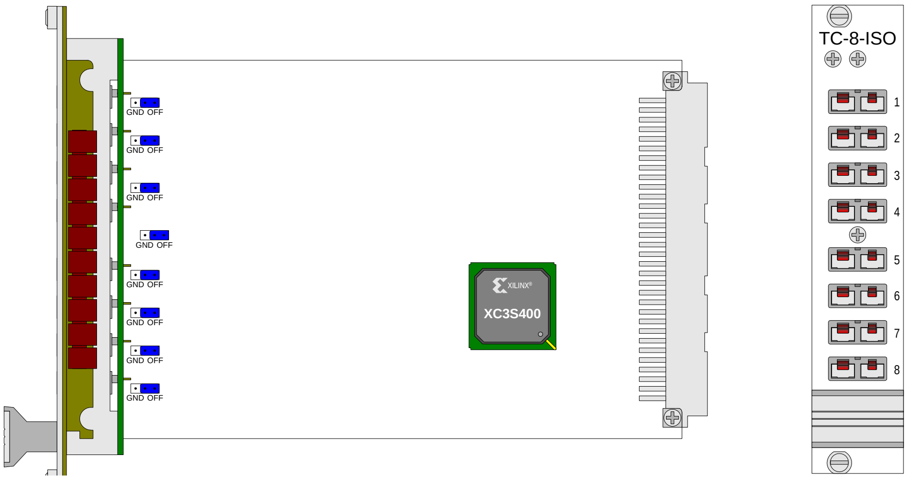


Fig. 107 – Pro II-TC-8 ISO Rev. E: Board and front panel

Programming

The following instructions are used to program the module:

Set sampling rate	**P2_TC_Set_Rate**
Copy input values to latches	**P2_TC_Latch**, **P2_Sync_All**
Read values	**P2_TC_Read_Latch** **P2_TC_Read_Latch4**, **P2_TC_Read_Latch8**

5.7.3 Pro II-CAN-2 Rev. E

The module Pro II-CAN-2 Rev. E has 2 CAN interfaces, a high speed or a low speed version. The names for the module versions are:

– Pro II-CAN-2 Rev. E: CAN interface high speed

– Pro II-CAN-2-LS Rev. E: CAN interface low speed

Each module is equipped with a freely programmable *TiCo* processor, which has full access to the CAN interfaces. Find more information about use and programming of the *TiCo* processor in the *TiCoBasic* manual.

The manual is divided into the following sections:

– CAN Controller

– Hardware design

– Message Management

– Setting the bus frequency

– Enable Interrupt / Trigger Event

– Module revisions

– Programming

CAN Controller

The CAN bus interface is equipped with the Intel® CAN controller AN82527 which works according to the specification CAN 2.0 parts A and B as well as to ISO 11898. You program the interface with *ADbasic* instructions, which are directly accessing the controller's registers.

Message

Messages sent via CAN bus are data telegrams with up to 8 bytes, which are characterized by so-called identifiers. The CAN controller supports identifiers with a length of 11 bit and 29 bit. The communication, that means the management of bus messages, is effected by 15 message objects.

The registers are used for configuration and status display of the CAN controller. Here the bus speed and interrupt handling, etc. are set (see separate documentation "82527 - Serial Communications Controller, Architectural Overview" by Intel®)

The CAN bus can be set to frequencies of up to 1 MHz and is usually operated with 1 MHz; with low speed CAN the max. frequency is 125 kHz. The CAN bus is galvanically isolated by optocouplers from the *ADwin* system.

An arriving message can trigger an interrupt which instantaneously generates an event at the processor. Therefore an immediate processing of messages is guaranteed.

Hardware design

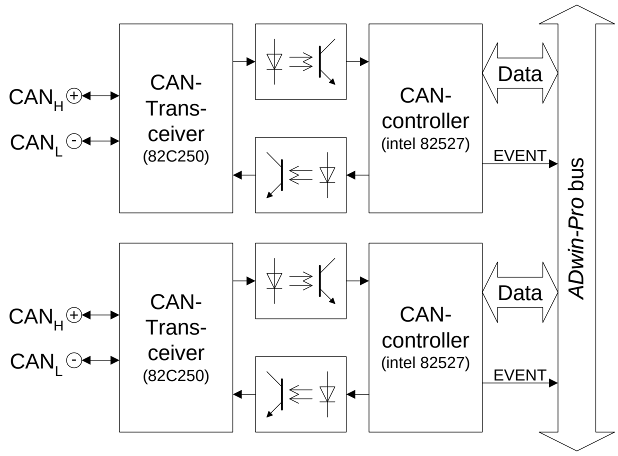


Fig. 108 – Pro II-CAN-2: Block diagram

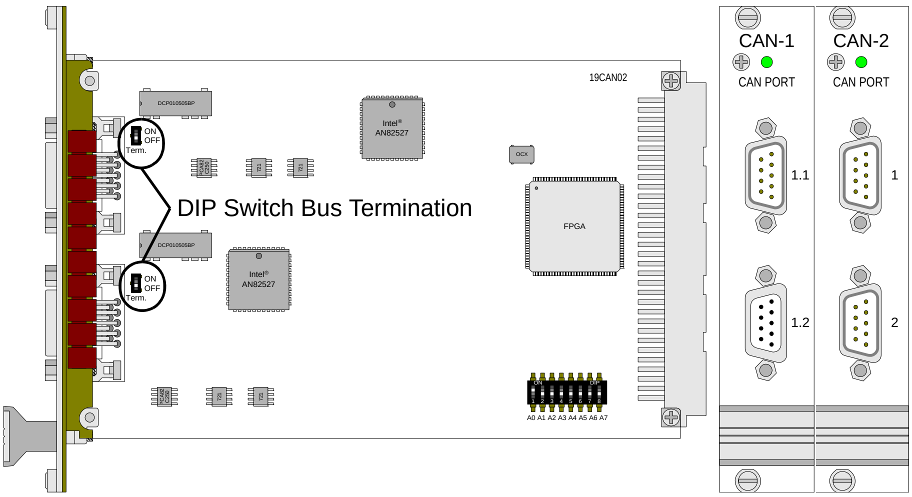


Fig. 109 – Pro II-CAN-2: PCB and front panels

The connections of the CAN bus interface are on the 9-pin D-SUB connector; the pin assignment is shown below.

CAN-2	CAN-2-LS

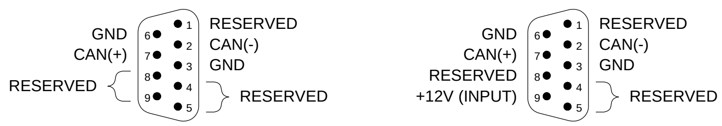


Fig. 110 – Pro II-CAN-2: Pin assignment (male)

Power supply (Low speed only)

The "low speed" version Pro II-CAN-2-LS requires an external power supply of 12V DC. The module needs a power supply for each controller separately.

Bus termination (High speed only)

If the CAN module functions as the physical termination of a high-speed CAN bus, it must be terminated with a 120Ω resistor (only the first or the last CAN node).

If a termination is necessary, move the DIP switch (see fig. 109) upward. CAN nodes, which are not positioned in an end-location, must not be terminated.

Message Management

Identifier

The CAN controller identifies messages by an identifier; these are parameters in a defined bit length. The parameters $0...2^{11}-1$ or $0...2^{29}-1$ result from the bit length.

Message objects

The controller stores each message (incoming or outgoing) in one out of 15 message objects. The message objects can either be configured to send or to receive messages. Message object 15 can only be used to receive messages. After initializing the CAN controller all message objects are not configured.

Each message object has an identifier, which enables the user to assign a message to a message object.

Transferring messages

In *ADbasic* a message is transferred to a message object using the array **can_msg**[], which can receive 8 data bytes plus the amount of data bytes (9 elements). When reading a message from the message object it can also be transferred to the array **can_msg**[].

Sending messages

Sending a message is made as follows:

– You configure a message object to send and define the identifier of the object (instruction **EN_TRANSMIT**).

– Save the message in **can_msg**[].

– Send the message (instruction **TRANSMIT**). The message in the array **can_msg**[] is transferred to the message object. As soon as the bus is ready, the message is sent (with the identifier of the message object).

Receiving messages

Receiving a message is made as follows:

– You configure a message object to receive and define the identifier of the object (instruction **EN_RECEIVE**).

– The controller monitors the CAN bus if there are incoming messages and saves messages with the right identifier in the message object.

– Transfer the message from the message object into the array **can_msg**[] (instruction **READ_MSG**) and read out the corresponding identifier.

An arriving message overwrites the old data in the message object, which will be definitely lost. Therefore pay attention to reading out the data faster than you are receiving them. A data loss is indicated by a flag.

The message object 15 has an additional buffer, so that 2 messages can be stored there.

Assigning messages

The allocation of an arriving message to a message object is automatically controlled by comparing its identifiers. The global mask (CAN registers 6...7 or 6...9) controls this comparison as follows:

– The identifier of the message is bit by bit compared to the identifier of the message object. If the relevant bits are identical, the message is transferred to the message object. Not relevant bits are not compared to each other, that is, the message is transferred to the object (if it depends on this bit).

– Relevant bits are set in the global mask.

Global mask

With the global mask a message object is used for receiving messages with **different identifiers** (ID). The following example shows the assignment of the message IDs 1...4 to the message object IDs 1...4, when all bits of the global mask are set, except the two least-significant bits (if you have an 11-bit identifier it is `11111111100b`).

Message ID	ID of the message object			
	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: Message is admitted
0: Message is not admitted

In this example the comparison of bit 2 is responsible for the assignment of the messages, because the bits 3...10 of the compared identifiers are identical (= 0) and the bits 0 and 1 are not compared, because they are set to zero in the global mask (= not relevant).

Setting the bus frequency

The **CAN bus frequency** depends on the configuration of the controller.

The initialization with **INIT_CAN** configures the controller automatically to a CAN bus frequency of 1 MHz. If the CAN bus is to operate with a different frequency, just use the instruction **SET_CAN_BAUDRATE**.

With low speed CAN the maximum bus frequency is 125kBit/s.

Bus frequency for special cases

In some special cases it may be better to select configurations other than those set with **SET_CAN_BAUDRATE**. For this purpose specified registers have to be set with the instruction **Poke**. The structure of the register is described in the controller documentation.

Enable Interrupt / Trigger Event

A message object can be enabled to trigger an interrupt when a message arrives. The interrupt output of the CAN controller is connected to the event input of the processor. The processor reacts immediately to incoming messages without having to control the message input (polling).

You can enable the interrupts of several message objects. Which object has caused the interrupt can be seen in the interrupt register (5Fh): It contains the number of the message object that caused the interrupt. If the interrupt flag (new message flag) is reset in the message object, the interrupt register will be updated. If there is no interrupt the register is set to 0. If another interrupt occurs during working with the first interrupt its source will be shown in the interrupt register. An additional interrupt does not occur in this case.

Module revisions

The differences between the revisions are described below:

Revision	Output date	Previous changes
A1		First version
A2	09/2003	New printed circuit board layout, bus termination with DIP switches instead of jumpers.

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Initializate CAN controller, set Baud rate	**P2_INIT_CAN** **P2_Set_CAN_Baudrate**
Transfer message objects	**P2_En_Receive** **P2_Read_Msg** **P2_En_Transmit** **P2_Transmit**
Set and read registers	**P2_Set_CAN_Reg** **P2_Get_CAN_Reg**
Use LED	**P2_Check_LED** **P2_Set_LED** **P2_CAN_Set_LED**
Use interrupt and event inputs	**P2_En_Interrupt** **P2_Event_Enable** **P2_Event_Config** **P2_Event_Read**

Programming in TiCoBasic

The module can be programmed with *TiCoBasic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `CAN_TiCo.inc` contains instructions for the functions:

Function	Instructions
Initialize CAN controller, set Baud rate	**INIT_CAN** **Set_CAN_Baudrate**
Transfer message objects	**En_Receive**, **Read_Msg** **Read_Msg_Con** **En_Transmit**, **Transmit** **Transmit_Status**
Set and read registers	**Set_CAN_Reg**, **Get_CAN_Reg**
Use LED	**Check_LED**, **Set_LED** **CAN_Set_LED**
Use interrupt	**En_Interrupt**

Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions
Data exchange with the *TiCo* processor using global variables and arrays	**P2_TDrv_Init** **P2_GetData_Long**, **P2_Get_Par**, **P2_Get_Par_Block** **P2_SetData_Long**, **P2_Set_Par**, **P2_Set_Par_Block** **P2_Get_TiCo_RingBuffer**, **P2_Set_TiCo_RingBuffer** **P2_RingBuffer_Empty** **P2_RingBuffer_Full**
Control *TiCo* processor	**P2_TiCo_Reset**, **P2_TiCo_Start**, **P2_TiCo_Stop** **P2_Get_TiCo_Bootloader_Status** **P2_Get_TiCo_Status**, **P2_Workload**

Function	Instructions

Control *TiCo* processes	`P2_Process_Status`
	`P2_TiCo_Get_Processdelay`
	`P2_TiCo_Set_Processdelay`
	`P2_TiCo_Start_Process`
	`P2_Tico_Stop_Process`

Transfer *TiCo* programs	`P2_TiCo_Flash`, `P2_TiCo_Load`

5.7.4 Pro II-RSxxx Rev. E

The Pro II-RSxxx module has 2 or 4 interfaces of the type RS-232 or RS-485. The type of interface is selected with the instruction **P2_rs485_send**.

The names for the module versions are:

– Pro II-RSx-2 Rev. E: 2 interfaces RS232/485

– Pro II-RSx-4 Rev. E: 4 interfaces RS232/485

All modules of the RSxxx-y modules are equipped with the "Quad Universal Asynchronous Receiver/Transmitter" (UART) controller, type TL16C754 from Texas Intruments®. Functionality and programming of the interfaces are based on this controller.

Each module is equipped with a freely programmable *TiCo* processor, which has full access to the RSxxx interfaces. Find more information about use and programming of the *TiCo* processor in the *TiCoBasic* manual.

The physical difference between the interface versions is their signal level, which is provided by appropriate drivers on the bus.

The description is divided into the following paragraphs:

– Hardware

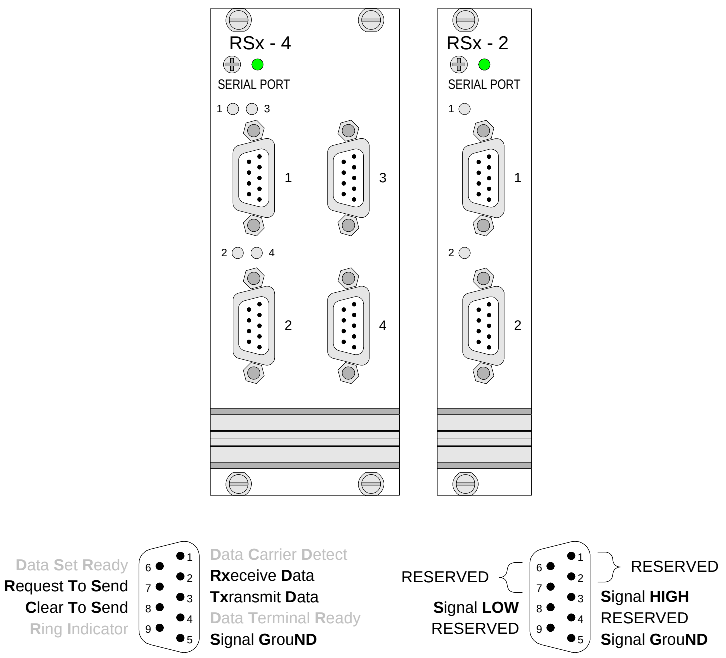
– Interface parameters

– Module revisions

– Programming

Hardware

These are the front panels and pin assignments of the modules Pro II-RSxxx. The pin assignment is switched together with the interface type.



RS-232 (male) RS-485 (male)

Fig. 111 – Pro II-RS-xxx: Front panels / pin assignments

Interface parameters

Each interface has an input and an output FIFO with a length of 64 bytes each. The settings of the interface parameters are made separately for each channel, using the controller register. Below the settings are described more detailed:

Handshake

– Handshake: The interface can be operated in 3 modes:
  1. Without handshake
  2. Software handshake
  3. Hardware handshake (RS232 only).
     When using the hardware handshake the signals RTS and CTS must be connected.

Parity

– Parity: In order to recognize an error or incorrect data during the transfer, a parity bit can be transferred at the same time. The parity can be even or odd or you can have no parity bit at all.

Data bits

– Data bits: the active data to be transferred may be 5...8 bits long.

Stop bits

– Stop bits: The number of stop bits can be set to 1, 1½ or 2. Here the number of stop bits depends on the number of data bits:
  - 5 data bits: 1 or 1½ stop bits.
  - 6…8 data bits: 1 or 2 stop bits.

Baud rate

– Baud rate: The physical data are between 35 Baud and 2.304 MBaud; when using an RS-232 interface the maximum Baud rate is 115.2 kBaud, according to the specification.

  The Baud rates are derived from the clock rate of the module; the basic clock rate has a frequency of 2.304MHz. Based on this fact, every Baud rate is possible, which can be derived from an integer division of the basic frequency. The divisor can have values between `1`…`0FFFFh`. The following table shows some common Baud rates and their divisors.

Baud rate	Divisor		Baud rate	Divisor	
	dec.	hex.		dec.	hex.
2.304.000	1	`0001h`	19.200	120	`0078h`
1.152.000	2	`0002h`	9.600	240	`00F0h`
460.800	5	`0005h`	4.800	480	`01E0h`
230.400	10	`000Ah`	2.400	960	`03C0h`
115.200	20	`0014h`	1.200	1920	`0780h`
57.600	40	`0028h`	600	3840	`0F00h`
38.400	60	`003Ch`	300	7680	`1E00h`

Fig. 112 – Pro II-RS-xxx: Baud rates

Special features of RS485

Contrary to the RS232 and RS422 interface, with RS485 more than 2 participants can communicate with each other. With an RS485 interface a bus is set up.

Consider the following:

– There is no handshake, because a handshake is only possible between two participants.

– The interface must know if it should write to the bus or get data from the bus (**`RS485_SEND`**).

Module revisions

The difference between the revisions is described below:

Revision	Release date	Previous changes
E	11/2007	First version

Programming

Programming in ADbasic

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Function	Instructions
Initialization	**P2_RS_INIT**, **P2_rs_reset**
Receiving and transmitting of data	**P2_read_fifo**, **P2_write_fifo**
Configure RS485 channel	**P2_RS485_Send**
Write and read access to the controller register	**P2_GET_RS**, **P2_SET_RS**
Use LED	**P2_Check_LED**, **P2_Set_LED** **P2_RS_Set_LED**

Programming in TiCoBasic

The module can be programmed with *TiCoBasic* instructions. The instructions are described in *TiCoBasic* online help.

The include file `RS_LIN_TiCo.inc` contains instructions for the following functions:

Function	Instructions
Initialization	**RS_INIT**, **rs_reset**
Receiving and transmitting of data	**read_fifo**, **write_fifo** **Check_Shift_Reg**
Configure RS485 channel	**RS485_Send**
Write and read access to the controller register	**GET_RS**, **SET_RS**
Use LED	**Check_LED**, **Set_LED** **RS_Set_LED**

Programming TiCo access

To access the *TiCo* processor from the ADwin CPU the following *ADbasic* instructions are defined in the include file `ADwinPro_All.inc`. The instructions are described in *ADbasic* online help and in the *TiCoBasic* manual.

Function	Instructions
Data exchange with the *TiCo* processor via global variables	**P2_TDrv_Init** **P2_GetData_Long**, **P2_Get_Par**, **P2_Get_Par_Block** **P2_SetData_Long**, **P2_Set_Par**, **P2_Set_Par_Block** **P2_Get_TiCo_RingBuffer**, **P2_Set_TiCo_RingBuffer** **P2_RingBuffer_Empty** **P2_RingBuffer_Full**

Function	Instructions
Control *TiCo* processor	**P2_TiCo_Reset**, **P2_TiCo_Start**, **P2_TiCo_Stop** **P2_Get_TiCo_Bootloader_Status** **P2_Get_TiCo_Status**, **P2_Workload**
Control *TiCo* processes	**P2_Process_Status** **P2_TiCo_Get_Processdelay** **P2_TiCo_Set_Processdelay** **P2_TiCo_Start_Process** **P2_Tico_Stop_Process**
Transfer *TiCo* programs	**P2_TiCo_Flash**, **P2_TiCo_Load**

5.7.5 Pro II-LIN-2 Rev. E

The module Pro II-LIN-2 Rev. E provides 2 LIN interfaces, which can be independently configured as LIN master or LIN slave.

LIN Interface

The LIN interfaces of the module are implemented according to spezification „LIN 2.1“ (Local Interconnect Network) of november 2006. You program the LIN interfaces with *ADbasic* instructions.

LIN is a serial communication protocol on a one-wire bus with a transfer rate of up to 20 KiBit/s. The bus efficiently supports the control of mechantronic nodes in distributed automotive applications.

Messages sent via LIN bus are data packets with up to 8 bytes payload, which are characterized by so-called identifiers. The management of bus messages is effected by 64 message boxes.

The bus concept refers to a single master node with multiple slave nodes. The master controls the total data transfer of the bus: Before each data packet the master sends a header with the identifier of the next data packet. Then, only this bus node will react (which can also be the master node itself) which manages a message box with the given identifier. Thus, this node will send a data packet to or receive a data packet from the LIN bus.

Hardware design

The connections of the LIN bus interfaces are on the 9-pin D-SUB connectors; the pin assignments are shown below.

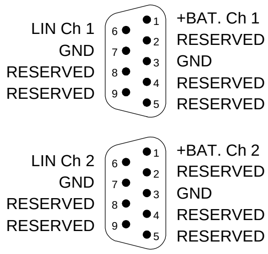


Fig. 113 – Pro II-LIN-2 Rev. E: Pin assignments (male)

Working with LIN bus

The 2 LIN interfaces of the module can be independently configured as LIN master or LIN slave with **`P2_LIN_Init_Write`**; this also true for the setting of the baudrate. The bus termination will be switched automatically according to the configuration.

The network master timing (NMT) is to be programmed in ***ADbasic***.

While configuring a message box with **`P2_LIN_Msg_Write`** you set the identifier and the sending mode (send, receive) of the message box. Any number of message boxes can be assigned to a LIN interface, but each identifier may be used only once for sending and once for receiving on the LIN bus; otherwise data collisions will occur.

After configuring, a message box will at once be active on the LIN bus, i.e. data packets can be sent or received.

If a LIN interface is configured as master, you use **`P2_LIN_Msg_Transmit`** to send a header with an identifier to the LIN bus. The message boxes configured with this identifier will react automatically.

The message box of a LIN master node operates different from a slave node:

– **Master node, send**: The LIN master sends both the header (see **P2_LIN_Msg_Transmit**) and then the data packet of the message box.
– **Master node, receive**: The LIN master sends the header (see **P2_LIN_Msg_Transmit**) on the LIN Bus and waits for the response of the appropriate slave node. The received data packet is stored into the message box.
– **Slave node, send**: The LIN slave waits until the master sends the header with the identifier which fits to the identifier of the message box. Only then the slave node will its data packet.
– **Slave node, receive**: The slave node waits until the master sends the header with the identifier which fits to the identifier of the message box. Then the slave receives the data packet and stores it into the message box.

Module Revisions

The differences between the revisions are described below:

Revision	Output date	Previous changes
E1		First version

Programming

The module is comfortably programmed with ***ADbasic*** instructions. The instructions are described in ***ADbasic*** online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Area	**Instructions**
Initialize and reset LIN interfaces	**P2_LIN_Init** **P2_LIN_Init_Write** **P2_LIN_Init_Apply** **P2_LIN_Reset**
Query LIN interface version	**P2_LIN_Get_Version**
Read or send data Send LIN header	**P2_LIN_Read_Dat** **P2_LIN_Msg_Write** **P2_LIN_Msg_Transmit**
Set module LED	**P2_Check_LED** **P2_Set_LED** **P2_LIN_Set_LED**

5.7.6 Pro II-Profi-SL Rev. E

The module Pro II-Profi-SL Rev. E provides a fieldbus node with the functionality of a Profibus slave. All settings are done via software.

Functions description

After power-on the fieldbus node must be initialized. The initialization determines the station address (slave node address) on the profibus as well as the size of the input and output areas.

There is a range each for data input and data output; each range has a maximum size of 76 bytes. Please note, that the terms "input" and "output" are used as the fieldbus controller sees them.

You set the number and length of input and output areas separately.

Hardware

The pin assignment of the 9-pin DSUB connector refers to DIN E 19245, part 3.

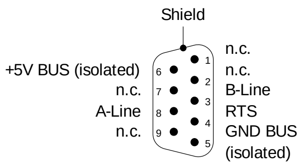


Fig. 114 – Pro II-Profi-SL Rev. E: Pin assignment

The Profibus has to be terminated at its physical beginning and at the end of its segments by an active terminator.

If required, you have to add the terminator yourself at the appropriate data lines of the fieldbus node or use an appropriate connector with integrated terminator.

Above and below the DSUB connector there are two LEDs, which display the operation status of the fieldbus node: operation mode (OP) and interface status (ST).

LED	Status	Meaning
OP	off	Offline or no power.
	green	Fieldbus node online, data exchange.
	flashing green	Fieldbus node online, status clear.
	flashing red, 1 flash	Error: Input/output configuration does not fit to master configuration.
	flashing red, 2 flashes	Error in Profibus configuration.
ST	off	Offline or no power.
	green	initialized.
	flashing green	initialized, diagnostiv event(s) present.
	red	Exception error.

Fig. 115 – Pro II-Profi-SL Rev. E: Meaning of LEDs

Projecting the Profibus

You are projecting the Profibus with a configuration tool suitable for the bus master. The following process description uses a Profibus master of the Hilscher company and the appropriate program `SyCon`.

The process description is valid for other configuration tools, correspondingly. Look for the exact process description of bus projection in the documentation of the configuration tool.

Copy the GSD file

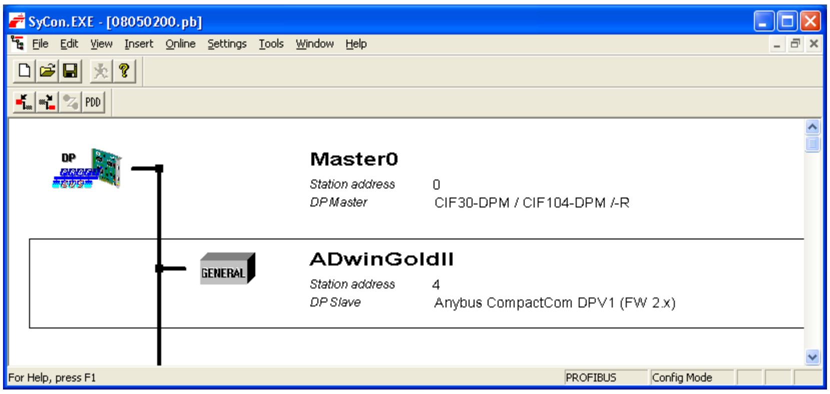
– Copy or import the GSD file `hmsb1811.gsd` of the fieldbus node from `C:\ADwin\Fieldbus\Profibus` into the source directory of the configuration tool.

  The configuration tool loads all required information about the new slave from the appropriate GSD file; the file content is determined by EN 50170. Afterwards, the slave can be accessed by any master.

Integrate the Slave

– In the configuration tool, add the Slave, i.e. the fieldbus node to the Profibus by selecting the GSD file `hmsb1811.gsd`. The station address must equal the address used for *ADbasic* initialization with **`P2_Init_Profibus`**.

Afterwards the bus could be structured as below:



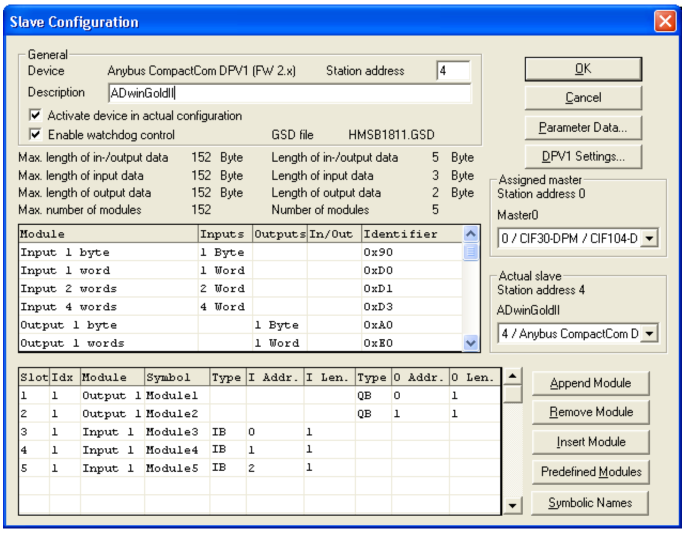
Configure the Slave

– Set number and length of input and output data in the fieldbus node memory. You may use only a single data length for input data and one for output data.

Input and output data can be set to a length of 1, 2, 4 or 8 Byte (2 byte = 1 word).

Number and length of data ranges must equal the data used for *ADbasic* initialization with **`P2_Init_Profibus`**.

Afterwards the slave configuration could be as below:



Module Revisions

The differences between the revisions are described below:

Revision	Output date	Previous changes
E1	July 2008	First version

Programming with *ADbasic*

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Area	Instructions
Initialize station address and data ranges	**`P2_Init_Profibus`**
Read and write data	**`P2_Run_Profibus`**

Initialization must be run with low priority since it takes some seconds; if it were a process with high priority, the PC interrupts the communication after a time (time-out). For the same reason, reading and writing data should be run with low priority.

Specifications

The fieldbus node is in agreement with the European Standard EN 50170, Volume 2. This norm is provided by the Profibus user organization:

Profibus Nutzerorganisation e.V.
Haid-und-Neu-Str. 7
76131 Karlsruhe, Germany
Phone: +49-72196-58590
Fax : +49-72196-58589
Order number: 0.042

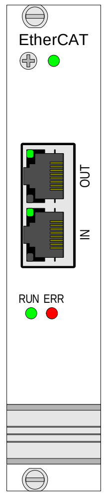
Operating modes of fieldbus node

The following table shows the operating modes, the fieldbus node supports and its behavior:

Operating mode	Behavior
Operate	The Profibus slave is part of the cyclic data exchange. Input data are transferred to the master via bus and output data are made ready for the master to transfer them.
Clear	The inputs are updated and the outputs are set to zero.
Stop	The slave is no longer part of the bus communication.

Fig. 116 – Pro-Profi-SL Rev. E: Operating modes

5.7.7 Pro II-EtherCAT-SL Rev. E



The module Pro II-EtherCAT-SL Rev. E provides a fieldbus node with the functionality of an EtherCAT slave. All settings are done via software.

Functions description

After power-on you must initialize the fieldbus node in *ADbasic*. The initialization determines the size of the input and output areas.

There is a range each for data input and data output; each range has a size of 16 Longs or 64 bytes. Please note, that the terms "input" and "output" are used as the fieldbus controller sees them.

Hardware

The interface has a plug connector of type RJ45 for both data input (`IN`) and data output (`OUT`). Each connector has a LED "`Link / Activity`" top right, which displays the operating status of the node in the EtherCAT bus. The two other LEDs (at the bottom of the plug) have no function.

Underneath the connectors there are LEDs displaying the status of the EtherCAT state machine (`RUN`) and the occurrence of communication errors (`ERR`).

LED	Status	Meaning
`Link / Activity`	off	Offline (or no power).
	green	Fieldbus node online, no data exchange.
	green, flickering	Fieldbus node online, with data exchange.
`RUN`	off	Status INIT: interface being initialized (or no power).
	blinks green	Status PRE-OP: Interface has contact to bus master.
	flashs green once	Status SAFE-OP: Interface can read data from the bus, but not send.
	green	Status OP: Interface is completely ready, inputs and outputs are active.
	red	Status EXCEPTION.
`ERR`	off	No error (or no power).
	blinks red	Invalid configuration.
	flashs red once	Local error in the interface; EtherCAT status has been changed.
	flashs red twice	Application watchdog timeout.
	red	Critival communication error.

Fig. 117 – Pro II-EtherCAT-SL Rev. E: Meaning of LEDs

If both LEDs `RUN` and `ERR` turn red, a serious error has occurred in the interface. Please inform the support of Jäger Messtechnik; you find the address on the inner side of the cover page of the manual.

Projecting the EtherCAT bus

You are projecting the EtherCAT bus with a configuration tool suitable for the bus master. The following process description uses the program "TwinCAT System Manager" of the Beckhoff company as EtherCAT bus master.

The process description is valid for other configuration tools, correspondingly. Look for the exact process description of bus projection in the documentation of the configuration tool.

– Copy the description file `ADwin-EtherCAT.xml` of the fieldbus node from `C:\ADwin\Fieldbus\EtherCAT` into the root directory of the configuration tool.

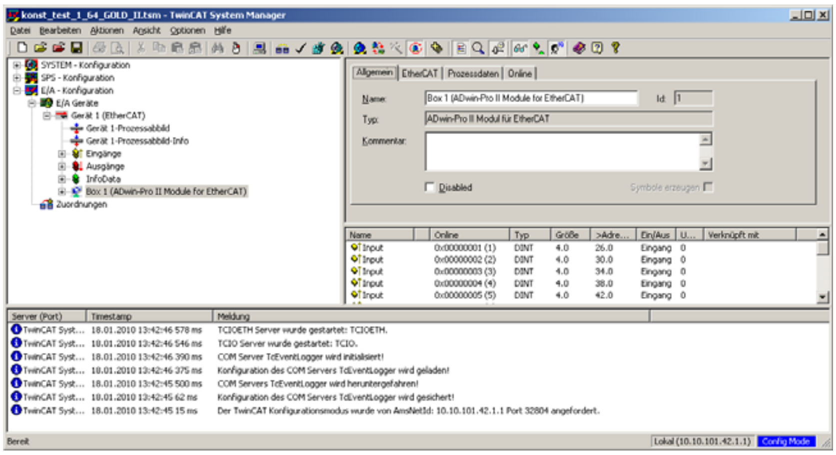
Upon start-up, the configuration tool loads the required information about the new slave from the appropriate description file.

– Add the *ADwin*-EtherCAT slave as bus member to the EtherCAT bus.

  Using TwinCAT System Manager, you mark the EtherCAT master and select the menu entry `Scan boxes` from the context menu (right mouse click).
  A list of all current bus members will be displayed.

– Select the *ADwin*-EtherCAT slave from the list; now the slave is confiremd as bus member.

  Afterwards the bus could be structured as below:



– Configure the *ADwin*-EtherCAT slave in an *ADbasic* program using the instruction **`P2_ECAT_Init`**.

  Though you can configure the slave from the configuration tool as well, the configuration in *ADbasic* has to be processed nevertheless–using the very same settings.

– Read the configuration into the configuration tool.

  Using the TwinCAT System Manager, you mark the *ADwin*-EtherCAT slave and clicken the button `Load PDO Info from the device`.

  The slave configuration is in any case as follows: 16 `DINT` (4 byte) as inputs and 16 `DINT` (4 byte) as outputs.

Module revisions

The difference between the revisions is described below:

Revision	Release date	Previous changes
E	12/2009	First version

The module revision is to be found on the front cover.

Programming with *ADbasic*

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Area	Instructions
Initialize station address and data ranges	**`P2_ECAT_Init`**
Get current interface version and operation mode	**`P2_ECAT_Get_Version`** **`P2_ECAT_Get_State`**
Read and write data in groups of 16 longs.	**`P2_ECAT_Read_Data_16L`** **`P2_ECAT_Write_Data_16L`**

Initialization must be run with low priority since it takes some seconds; if it were a process with high priority, the PC interrupts the communication after a time (time-out). For the same reason, reading and writing data should be run with low priority.

Specifications

The fieldbus node is in agreement with the international standard IEC 61158 and IEC 61784-2. More information is provided by the EtherCAT user organization:

EtherCAT Technology Group
Ostendstraße 196
D-90482 Nürnberg
Tel.: +49 9115405620
Fax : +49 9115405629
http://www.ethercat.org/

Operating modes of EtherCAT node

The following table shows the operating modes, the EtherCAT node supports and its behavior:

Operating mode	Behavior
Init	The EtherCAT slave is being initialized by the bus master.
PreOp	The interface is part of the data exchange, inputs and outputs are not active.
SafeOp	The interface can receive data, outputs are not active.
Op	The interface is completely ready; inputs and outputs are active.

Fig. 118 – EtherCAT: Operating modes

5.7.8 Pro II-FlexRay-2 Rev. E

The module Pro II-FlexRay-2 Rev. E is euqipped with 2 FlexRay interfaces, each interface represents a complete FlexRay bus with 2 channels.

The module can be configured to start a FlexRay bus without any other bus members. To do so, both FlexRay interfaces have to run as coldstarter nodes and the interfaces have to be combined to a FlexRay bus via DIP switches.

You can also set the bus termination for each interface and channel via DIP switch.

The description of the FlexRay module is divided into the following sections:

– FlexRay Controller
– Hardware
– Module Revisions
– Programming

FlexRay Controller

The module Pro II-FlexRay-2 is equipped with two FlexRay controllers MFR4310 from FreeScale® and runs according to the "FlexRay Communications System Protocol Specification V2.1". You program the interfaces with *ADbasic* instructions which directly access the controller registers.

For configuration and status display of the FlexRay controllers you use the appropriate registers. Here you can set all FlexRay parameters as e.g. bus speed, bus timing, etc.

You find more information on the web site http://www.freescale.com in the following documentations from FreeScale®:

– Engineering Bulletin EB683: MFR4310 and MFR4310 differences
– Data sheet MFR4300: MFR4300 FlexRay Communication Controller

Hardware

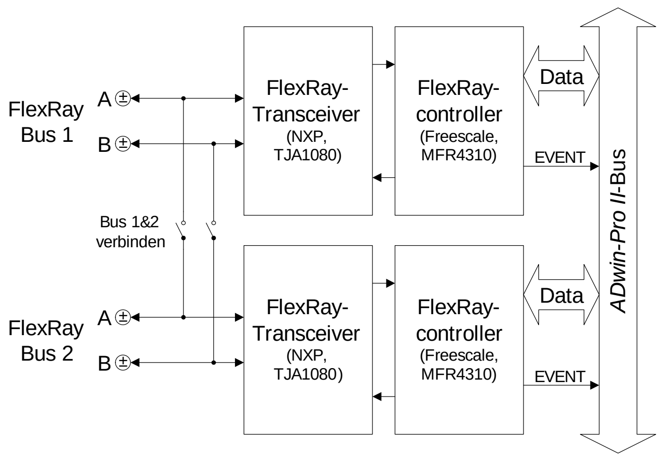


Fig. 119 – Pro II-FlexRay-2 Rev. E: Block diagram

The connections of the FlexRay interfaces are provided on a 9-pole DSUB connector; the pin assignment is shown below.

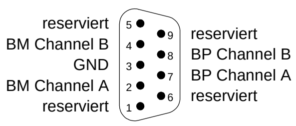


Fig. 120 – Pro II-FlexRay-2: Pin assignment

If an interface runs as physical end of a FlexRay bus, the bus bus must be terminated at this interface with the appropriate DIP switch (see fig. 121); the channels A and B can be terminated separately. Since both channels are run differential, you always have to switch both (!) DIP switches for each channel to the right for termination.
If the FlexRay module is not located at the end of the bus you may not terminate the interface.

Both FlexRay interfaces can be combined to a startable FlexRay cluster via DIP swithces. For combination of the channels A and B you always have to switch both (!) DIP switches for each channel upwards (see fig. 121).

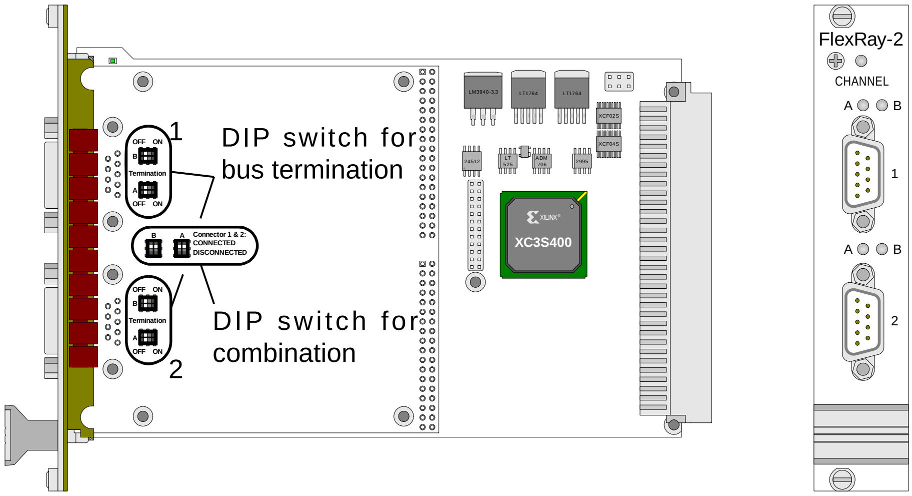


Fig. 121 – Pro II-FlexRay-2 Rev. E: PCB and front panels

Module Revisions

The differences between the revisions are described below:

Revision	Output date	Previous changes
E1	08 / 2009	First version

Programming

The module is comfortably programmed with *ADbasic* instructions. The instructions are described in *ADbasic* online help and in the Pro II Software manual.

The include file `ADwinPro_All.inc` contains instructions for the following functions:

Area	Instructions
Initialize a FlexRay interface	**P2_FlexRay_Init**
Reset a FlexRay controller	**P2_FlexRay_Reset**
Query interface version	**P2_FlexRay_Get_Version**
Set and read controller register	**P2_FlexRay_Read_Word** **P2_FlexRay_Write_Word**
Set module LED	**P2_Check_LED** **P2_Set_LED** **P2_FlexRay_Set_LED**

6 Calibration

6.1 General information

The digital-to-analog (DAC) and analog-to-digital (ADC) converters of the ***ADwin*** systems are calibrated in factory. According to the regulations for measurement accuracy it is recommended to calibrate the systems in regular time intervals.

Qualified personnel

Programming, start-up and operation, as well as the modification of program parameters must be performed only by appropriately qualified personnel.

> *Qualified personnel are persons who, due to their education, experience and training as well as their knowledge of applicable technical standards, guidelines, accident prevention regulations and operating conditions, have been authorized by a quality assurance representative at the site to perform the necessary acivities, while recognizing and avoiding any possible dangers.*
> *(Definition of qualified personnel as per VDE 105 and ICE 364).*

Availability of the documents

This product documentation and all documents referred to, have always to be available and to be strictly observed. For damages caused by disregarding the information in this documentation or in all other additional documentations, no liability is assumed by the company *Jäger Computergesteuerte Messtechnik GmbH*, Lorsch, Germany.

Tools

The following tools are necessary to calibrate the system:

– a reference voltage source with an accuracy of:
  - 10µV for calibration of 18 bit converters
  - 30µV for calibration of 16 bit converters
  - 100µV for calibration of 14 bit converters

– a digital multi meter with an accuracy of:
  - 10µV for calibration of 18 bit converters
  - 30µV for calibration of 16 bit converters
  - 100µV for calibration of 14 bit converters

– connecting cables from the input/outputs to the reference voltage and to the measurement device

Caution: Risk of electric shock.

ADwin-Pro systems have a power supply device, which gives access to high-voltage lines and connectors if the system is open. The ventilation slots are wide enough to pass through an alignment tool of 2.5 mm (=0.1inch).

Calibrate the system only when it is closed!
Do not pass any conductive objects through the ventilation slots!

6.2 Calculation basis

Voltage range

The standard voltage range of the analog inputs/outputs of the ***ADwin*** systems is -10V … +10V (bipolar 20 Volt).

Allocating digits to voltage

The 65536 (2^{16}) digits are allocated to the corresponding voltage ranges of the ADC and DAC in such a manner that the value for

- 0 (zero) digits corresponds to the maximum negative voltage.
- 65535 digits correspond to the maximum positive voltage.

The value for 65536 digits, exactly 10 Volt, is therefore just beyond the measurement range, therefore you get for the 16 bit AD or DA conversion a maximum voltage value of 9.999695 Volt, and for the 18 bit AD conversion a value of 9.999923706 Volt.

Zero offset

In bipolar settings this results in a zero offset, called offset in the following text. The offset has the value V_{OFF}=-10V.

Least Significant bit V_{LSB}

The value V_{LSB} defines the voltage, which corresponds to the least significant bit. The value in the standard setting is

- with 18 bit converters: 20V ./. 2^{18} = 76.294µV
- with 16 bit converters: 20V ./. 2^{16} = 305.175µV
- with 14bit converters: 20V ./. 2^{14} = 1220.7µV

Gain k_V

When using Pro-AIn modules with programmable gain arrays (PGA), you can amplify the input voltage by factors 2, 4 and 8. Thus, the measurement range gets smaller by the corresponding gain factor k_V.

Please pay attention to the fact that also the interference signals are amplified when using applications with k_V > 1. These can be reduced by programming digital filters in ***ADbasic***.

Allocating the bits

In order to get the same allocation of bits during measurements with 14 bit ADC as with a 16 bit ADC, the converted value is presented left-aligned in the lower word (16 bit) with the 14 bit ADC. The least 2 significant bits are always 0.

Bit no.	31…24	23…16	15…6	5…2	1…0
content	0	18 bit value in bits 6…23		0	0
	0	0	16 bit value in bits 0…15		
	0	0	14 bit value in bits 2…15		0
	upper word		lower word		

Fig. 122 – Bit allocation with different resolutions

The 16384 digits of a 14 bit ADC are mapped to the 65535 digits of a 16 bit ADC. Therefore, 4 digits of the 16 bit ADC correspond to one digit of the 14 bit ADC.

DAC

For DAC use the formula:

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

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

ADC

For ADC use the formula:

$$\mathrm{Digits} = \frac{U_{IN} - U_{OFF}}{U_{LSB}}$$

$$U_{IN} = \frac{\mathrm{Digits} \cdot U_{LSB} + U_{OFF}}{k_V}$$

Tolerance range

Slight variations regarding the calculated values may be within the tolerance range of the individual component. Two kinds of variations are possible (in LSB), which are indicated in your hardware manual.

INL

– The integral non-linearity (INL) defines the deviation from the ideal wave form covering the whole input voltage range.

DNL

– The differential non-linearity (DNL) defines the deviation from the ideal value of the quantization level.

6.3 Calibrating a module

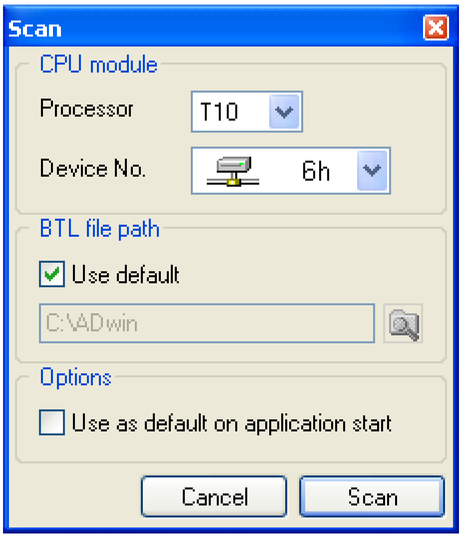
Initializing the hardware

At first define the voltage rangeof the module.

Calibration has to be made when the system reaches its operating temperature. 30 minutes after power-up of the system, the operating temperature is reached, provided the system has a (room) temperature of approx. 20…25°C before power-up.

Please note the general information in chapter 6.1.

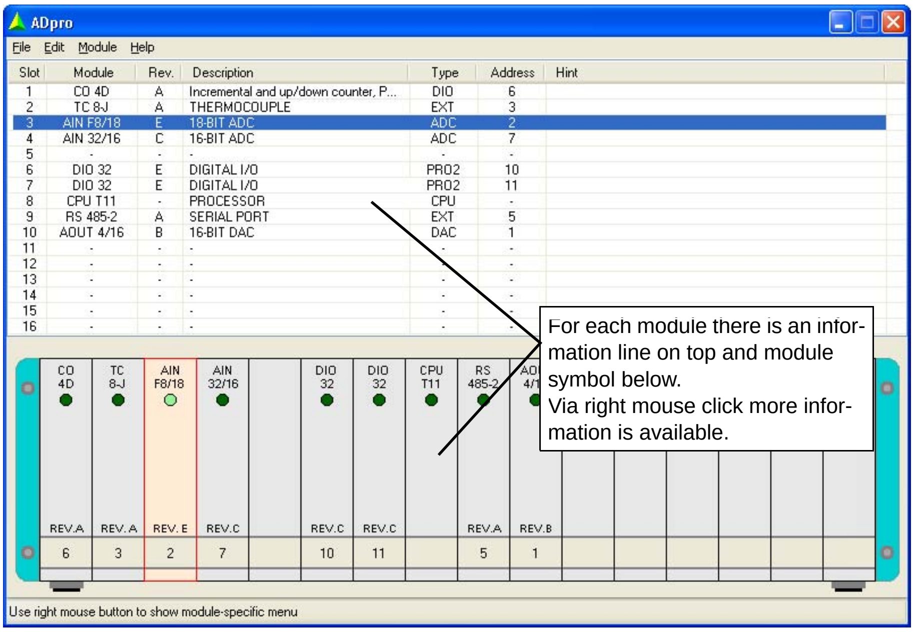
Call the program `ADpro.exe` from the Windows start menu under "`Programs\ADwin`". The program requires `Microsoft .NET Framework 2.0`. The dialog window `Scan` opens.



Please note: The next step will stop all processes and reset all module settings!

Enter the data for the ***ADwin*** system to be calibrated. The button `Scan` starts a connection to the ***ADwin*** system and reads system information. The program `ADpro.exe` will initlialize the ***ADwin*** system, i.e. it stops and deletes running processes.

If your ***ADwin*** system has booted successfully, the window "`ADwin - ADpro`" opens.



In the window `ADpro`, click on the module to calibrate and select the menu entry `Calibration` in the `Module` menu. If the `Calibration` entry is not displayed in the menu, the selected module cannot be calibrated.

Calibration

Connect the measurement device and the reference voltage source.

Follow the information displayed in the dialog window. Please note the difference of analog input modules with and without multiplexer:

– with the AIN modules the calibration of the ADC is effected via the input channel 1.

– with the AIN-F modules the connected channel is selected in the window "`Input channel`".

7 Accessories

7.1 LEMO Cable Sets for *ADwin-Pro* Systems

Pro-CS-1	4 cables with 200 mm (7.8 inch) and 4 cables with 400 mm (15.7 inch)
Pro-CS-2	4 cables with 400 mm (15.7 inch) and 4 cables with 800 mm (31.5 inch)
Pro-CS-3	4 cables with 1000 mm (39.4 inch) and 4 cables with 1500 mm (59 inch)
Pro-CS-4	4 cables with 5000 mm (196.8 inch)

All cables with LEMO connectors on each end

7.2 LEMO Adapter sets

Pro-AS-1	4 adapters: LEMO female connectors to BNC connectors (male)
Pro-AS-3	4 LEMO Y connector (male to double female)
Pro-AS-4	4 adapters: LEMO female connector to LEMO female connector
Pro-AS-5	4 terminators: 50 Ω, LEMO female connector
Pro-AS-6	4 cable adapters (length 4" / 10 cm): LEMO female connector to BNC connector (male)

7.3 Reference addresses

7.3.1 LEMO Connectors

Pro modules are equipped with the following LEMO connectors:

– Male connectors / female connectors of series 00 NIM-CAMAC, 1-pole
  - Cable connector: Type FFS
  - Built-in female connector: Type ERN

– Male connectors / female connectors of series 00 Multi-Contact, 2-pole
  - Cable connector: Type FGG
  - Built-in female connector: Type EGG

– Pt100 modules: Male connectors / female connectors of series 0B:
  - Cable connector: Type FGG
  - Built-in female connector: Type EGG

Manufacturer of LEMO connectors:

LEMO S.A. Chemin de Champs-Courbes 28 P.O. Box 194 CH-1024 Ecublens, Switzerland	Tel.: +41 21 695 16 00 Fax: +41 21 695 16 01 E-Mail: info@lemo,com Internet: www.lemo.com

7.3.2 Power Supply Pro-Mini

The plug connector for external power supply of the casing Pro-Mini is manufactured by Phoenix Contact GmbH:

> Combicon plug component, pitch 5.0 mm, Type MSTB 2,5/ 3-STF;
> order no. 1786844 (as of Dec. 2005)

Manufacturer of the connector:

Phoenix Contact GmbH & Co. KG Flachsmarktstraße 8 D-32825 Blomberg	Tel.: +49 5235 300 Fax: +49 5235 341200 E-Mail: info@phoenixcontact.com Internet: www.phoenixcontact.com

Annex

A.1 RoHS Declaration of Conformity

The directive 2002/95/EG of the European Union on the restriction of the use of certain hazardous substances in electrical und electronic equipment (RoHS directive) has become operative as from 1st July, 2006.

The following substances are involved:

– Lead (Pb)

– Cadmium (Cd)

– Hexavalent chromium (Cr VI)

– Polybrominated biphenyls (PBB)

– Polybrominated diphenyl ethers (PBDE)

– Mercury (Hg)

The product line ***ADwin-Pro II*** complies with the requirements of the RoHS directive in all delivered variants.

A.2 List of Modules