ADwin-Pro

System and hardware description

ADwin-Pro hardware, manual version 2.4, December 2004

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

ADwin-Pro hardware, manual version 2.4, December 2004

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 have absolutely to be considered in order to guarantee an operation without any errors
- 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 parentheses and characterized in the font Courier New.

`Program text`

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

`Var_1`

Source code elements such as `INSTRUCTIONS`, *`variables`*, `comment` 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* system

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

When the *ADwin-Pro* system was developed great attention was paid to the electromagnetic compatibility. The *ADwin-Pro* 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* system needs a processor module. It communicates via USB or Ethernet with the PC or notebook; previous versions used a serial link connection.

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

Since middle of 2002 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".

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* System

2.1 Order of the installation

Please keep absolutely the order of the individual installation steps:

1. Installation of the *ADwin* programs from the *ADwin*-CDROM
2. Installation of the interface drivers
3. Initialization of the hardware
4. First steps in *ADbasic*
5. Programming with the *ADbasic* Tutorial

2.2 Further Documentation

The installation of the *ADwin* programs and the *ADwin* drivers from the CDROM is described more detailed in the manual about interface installation of *ADwin* systems: *ADwin* Driver Installation.

The instructions for the real-time development tool *ADbasic* are described in the manual *ADbasic* as well as in the online help. The instructions for accessing an *ADwin-Pro* system with *ADbasic* are provided in special include files, which you have to include in your *ADbasic* program.

For this, read the manual: *ADwin-Pro* System Specifications - Programming in *ADbasic*.

3 Enclosures for the *ADwin-Pro* 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	16	100V…240V	AC
ADwin-Pro-DC	16	10V…35V	DC
ADwin-Pro-BM	15	100V…240V	AC
ADwin-Pro-light	7	100V…240V	AC
ADwin-Pro-mini	5	5V	DC
ADwin-Pro-mini-2	5	10V…18V	DC
ADwin-Pro-mini-3	5	20V…36V	DC

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 have a width of 5 HP = 1 inch.

Pro with 16 slots

3.1 *ADwin-Pro*

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

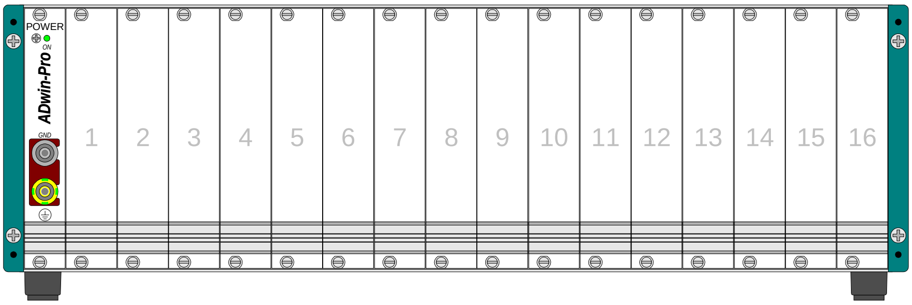


Fig. 1 – Enclosure *ADwin-Pro*

The system fuse is located in a slot in the power supply unit above the socket 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. 2 – Enclosure *ADwin-Pro*: 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
A	1997	First version with linear power supply.
B1	Sep. 1999	Prototype (internal use only, not delivered to customers)
B2	Jun. 2003	Internal structure modified, function unchanged.
B3	Jun. 2004	New power supply (switching power supply) with automatical voltage adaptation.

3.2 *ADwin-Pro-DC*

**Pro-DC
with 16 slots**

The *ADwin-Pro-DC* enclosure is similar to the standard enclosure *ADwin-Pro*, 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.

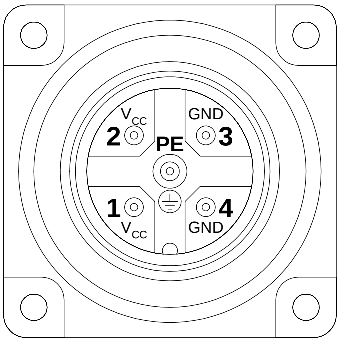


Fig. 3 – Enclosure *ADwin-Pro-DC*:
Detailed view of the pin assignment

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. 80 W, DC-DC converter 10 V…35 V

Fig. 4 – Enclosure *ADwin-Pro-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
A	1997	First version with linear power supply.
B1	Sep. 1999	Prototype (internal use only, not delivered to customers)
B2	Jun. 2003	Internal structure modified, function unchanged.
B3	Nov. 2003	Various improvements

Pro "backmounted" with 15 slots

3.3 *ADwin-Pro-BM*

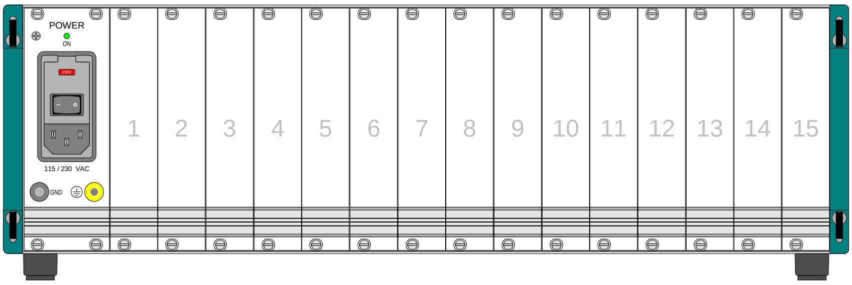


Fig. 5 – Enclosure of *ADwin-Pro-BM* (rear panel)

In the version "backmounted" of the standard enclosure, the modules are plugged-in at the rear of the enclosure.
The revisions are similar to those of the *ADwin-Pro* enclosure.

The system fuse is located in a slot in the power supply unit above the socket 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*-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
A	1997	First version with linear power supply.
B1	Sep. 1999	Prototype (internal use only, not delivered to customers)
B2	Jun. 2003	Internal structure modified, function unchanged.
B3	Nov. 2003	New power supply (switching power supply) with automatical voltage adaptation.

3.4 *ADwin-Pro-light*

Pro-light with 7 slots

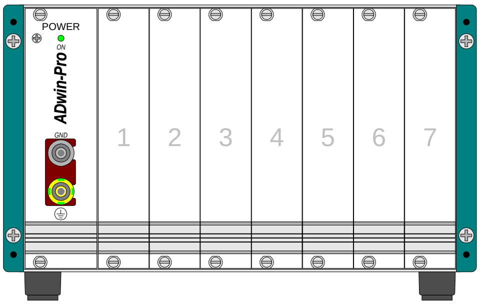


Fig. 7 – Enclosure *ADwin-Pro*-light

The *ADwin-Pro-light* enclosure. The backplane of the enclosure connects the processor module with the *ADwin-Pro* 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. 40W, 100…240VAC at 50/60Hz switching power supply
Fuse	2A, delayed-action fuse

Fig. 8 – Enclosure *ADwin-Pro-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
A1	1997	First version with linear power supply.
A2	Jun. 2004	New power supply (switching power supply) with automatical voltage adaptation. Internal structure modified.
A3	Aug. 2004	Various improvements

3.5 *ADwin-Pro-mini*

Pro-mini with 5 slots

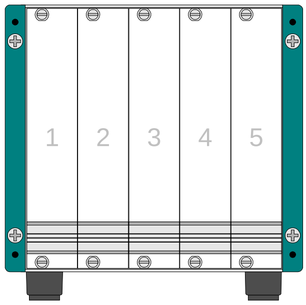


Fig. 9 – Enclosure *ADwin-Pro*-mini

The *ADwin-Pro-light* enclosure is available in 3 versions, each of which works with a different power supply voltage. In any case an external power supply unit is required.

The required power of the external power supply unit depends on the type and number of the used modules; we recommend a power of at least 20W.

Number of Slots		5
Main dimensions (l x w x h) Slot area (w x h)		253mm × 147.3mm × 146mm 20 HP × 3 U
Fuse		4A, delayed-action fuse
External power supply unit		external power supply unit required
	mini	5V DC, 4A
	mini-2	10V…18V DC
	mini-3	20V…36V DC

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

The connector for the external power supply unit as well as a 4A fuse can be found at the rear of the *ADwin-Pro*-mini enclosure (see fig. 11).

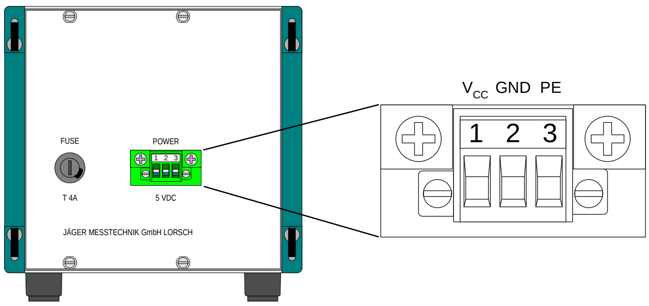


Fig. 11 – Enclosure *ADwin-Pro*-mini:
Rear and detail of the pin assignment

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

Revision	Release	Previous versions
A	1998	First version

4 *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.

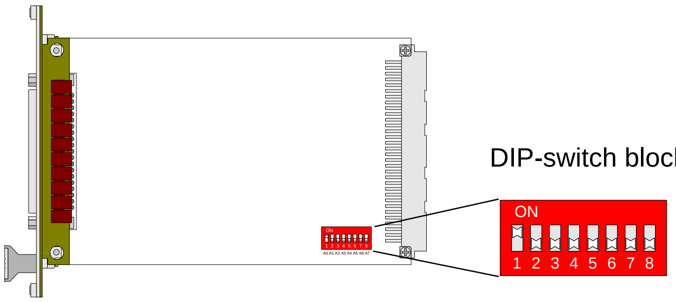
4.1 Setting the module's addresses

The ***ADwin-Pro*** modules are divided into 5 groups according to their function:

- CPU processor modules
- ADC analog input modules
- DAC analog output modules
- DIO digital input/output modules, relays and counter modules
- EXT add-on modules (thermocouples, 5B (MB) carrier boards, serial communication interfaces)

Each ***ADwin-Pro*** module has its own on-board block of DIP-switches for address-selection right to the bottom (except the processor modules). The address is selectable between 1 and 255 (see fig. 12). Each module of the same group needs to have a different address.

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.



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

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…233	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 – Adressierung mit DIP-Schaltern: Feldbus-Module

4.2 Processor modules

For each ***ADwin-Pro*** system a processor module is required. This processor module does all the communication with the PC or laptop and all other ***ADwin-Pro*** modules. Furthermore it runs the user defined processes. The data connection to the PC is established via USB or Ethernet.

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

Overview CPU modules

According to the requested computing power (clock rate) several processor modules are available:

Module	Pro-CPU-T9	Pro-CPU-T9-ENET / -USB	Pro-CPU-T10-ENET
Processor	ADSP 21062	ADSP 21062	ADSP 21162
Clock rate	40 MHz	40 MHz	80 MHz
Data connection	Link	ENET, USB	ENET, USB
Internal memory	256 kB opt. 512 kB	256 kB opt. 512 kB	512 kB
External memory	4 MB opt. 16/32 MB	16 MB opt. 64 MB	128 MB
Inputs	Event In	Event In opt. Digin 0	Event In Digin 0

Fig. 14 – Overview Pro-CPU modules

The external trigger input (EVENT) enables to recognize an external signal as trigger for an event. With this input processes are triggered by an external signal and can be 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 (specified value for CPU-T9 in basic version; 25 ns are typical).

4.2.1 Pro-CPU-T9

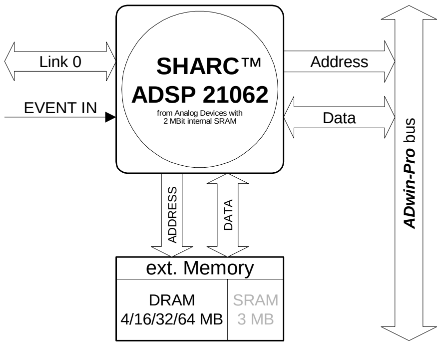


Fig. 15 – Pro-CPU-T9: Block diagram

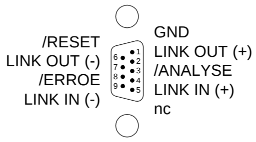


Fig. 16 – Pro-CPU-T9: Pin assignment

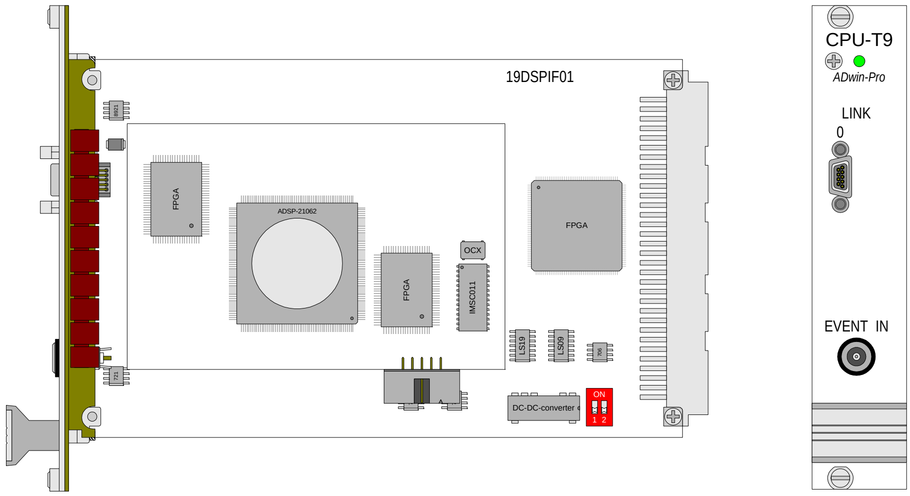


Fig. 17 – Pro-CPU-T9: Board and front panel

4.2.2 Pro-CPU-T9-ENET / -USB

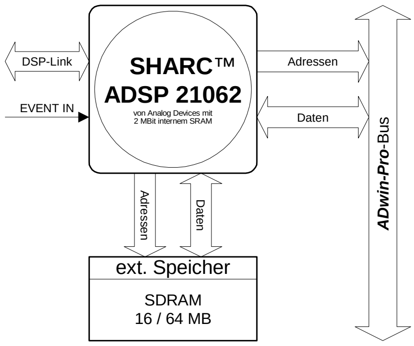


Fig. 18 – Pro-CPU-T9-ENET / -USB: Block diagram

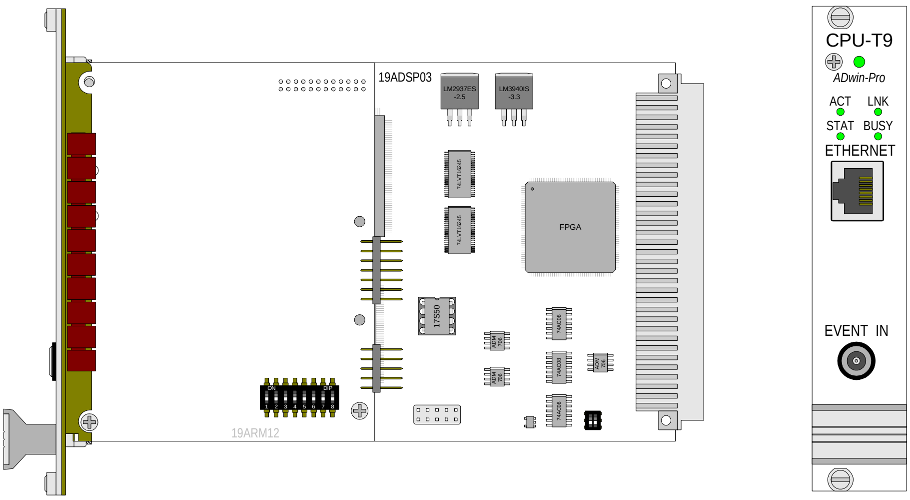


Fig. 19 – Pro-CPU-T9-ENET: Board and front panel

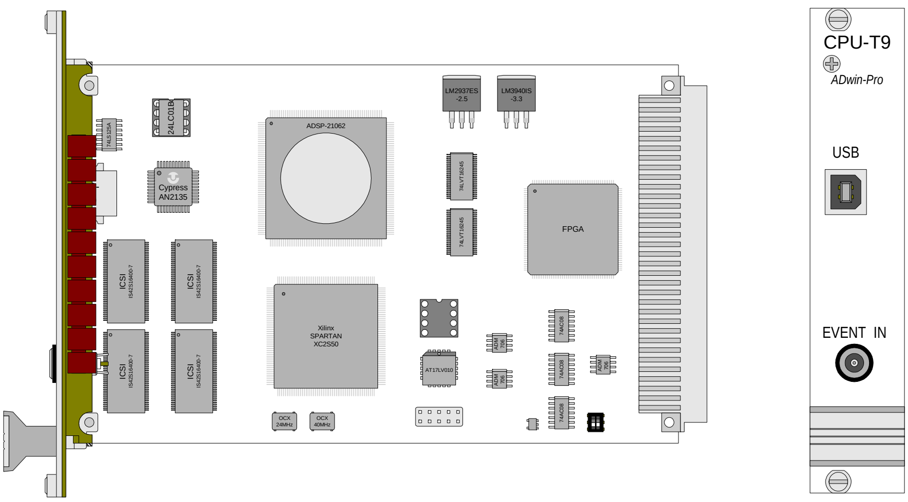


Fig. 20 – Pro-CPU-T9-USB: Board and front panel

The module may have an additional input `Digin 0` (ordering option). The input `Digin 0` is for use with TTL signals only.

4.2.3 Pro-CPU-T10-ENET

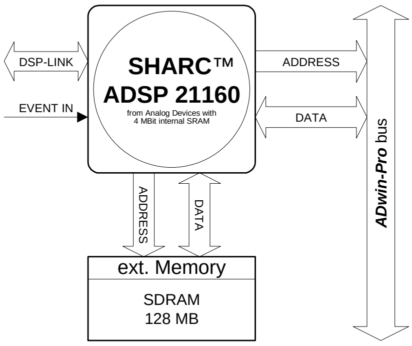


Fig. 21 – Pro-CPU-T10-ENET: Block diagram

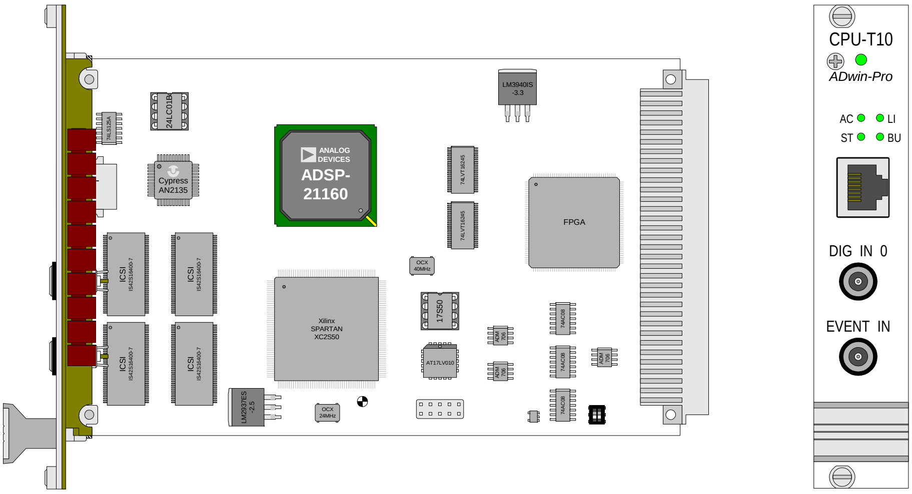


Fig. 22 – Pro-CPU-T10-ENET: Board and front panel

The input `Digin 0` is for use with TTL signals only.

4.2.4 Pro-Boot

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

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

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

Hardware

The bootloader unit (including Flash EEPROM) is placed on an additional board:

- Pro-CPU-T9-ENET, Pro-CPU-T10-ENET

  The bootloader is integrated on the Ethernet board.

  The bootloader is programmed with the program `ADethflash`. (windows start menu under `Programs\ADwin`). `ADethflash` contains online notes for use.

- Pro-CPU-T9 (link interface)

  The boot loader unit is placed on a separate board which is located between the SDRAM memory and the interface circuit board. The module's width is 10HP and needs 2 slots in the Pro system.

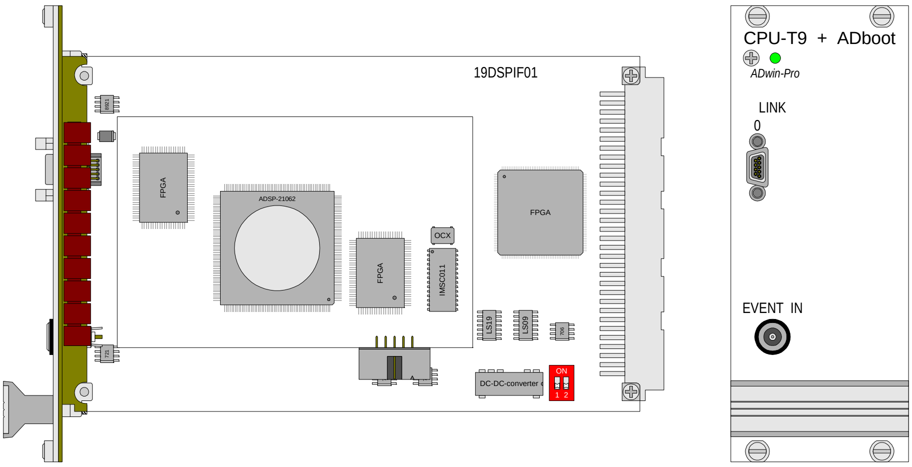


Fig. 23 – Example: Module Pro-CPU-T9 with Pro-Boot

The help file `ADBOOTLOAD.HLP` shows how to program of the bootloader (see folder <`C:\ADwin\Tools\ADbootload\`...>). Please open the help file and continue according to the listed description.

Software

By installation of the ***ADbasic*** and the ***ADwin*** drivers from the CDROM (version 3.0.22a6 or higher), 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.

4.2.5 The Watchdog

You can monitor your processor module with a watchdog (T9, T10). 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.

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.

Note in relation to the Pro-Flash-Boot:

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!

4.3 Analog Input Modules

Modulname	AIn 8/12	AIn 8/12	AIn 8/14	AIn 32/12	AIn 32/12	AIn 32/14	AIn 8/16	AIn 8/16	AIn 8/16	AIn 32/16	AIn 32/16	AIn F-4/12	AIn F-8/12	AIn F-4/14	AIn F-8/14	AIn F-4/16	AIn F-8/16
Revision	A	B	A	A	B	A	A	B	C	B	C	A	A	B	B	A	A
Number ADC	1	1	1	1	1	1	1	1	1	1	1	4	8	4	8	4	8
Resolution [Bit]	12	12	14	12	12	14	16	16	16	16	16	12	12	14	14	16	16
Conv. time [µs]	8.5	0.75	0.5	8.5	0.75	0.5	10	8	5	8	5	0.75	0.75	0.4	0.4	8	8
max. sampl. rate [ksample/s]	117[a]	1250[a]	2000	117[a]	1250[a]	2000	100[a]	100[a]	200	100[a]	200	1250[b]	1250[b]	2200[b]	2200[b]	100	100
Channels diff.	8	8	8	16	16	16	8	8	8	16	16	4	8	4	8	4	8
Channels sg. end.	–	–	–	32	32	32	–	–	–	32		–	–	–	–	–	–
Voltage ±5V	x	–	–	x	–	–	x	–	–	–	–	–	–	–	–	–	–
±10V	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
0...10V	x	x	–	x	x	–	x	–	–	–	–	–	–	–	–	–	–
optional ±20mA	–	–	x	–	–	x	–	–	–	–	–	–	–	–	–	–	–
Gain	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1, 2, 4, 8	1	1	1, 2, 4, 8	1, 2, 4, 8	1	1
Calibration[c]	TR	TR	SW	TR	TR	SW	TR	SW	SW	SW	SW	TR	TR	SW	SW	TR	TR
Page	19	21	23	25	28	30	32	34	36	38	40	42	44	46	48	50	52

a. To be achieved under favorable conditions: 1 input channel, time-optimized program
b. per ADC
c. SW: via software, TR: with trimmers; see chapter 5.3.

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 socket as possible.

4.3.1 Pro-AIn-8/12 REVA

To this module you find an improved successor module Pro-AIn-8/14 REVA (see page 23).

The analog input module Pro-AIn-8/12 REVA has a 12 bit ADC and 8 differential inputs. The inputs are equipped with shielded LEMO-sockets (CAMAC European norm). A programmable amplifier (PGA) and a multiplexer (MUX) are connected before the ADC.

The module can be combined with amplifiers, filters, thermocouples and PTC modules.

The input voltage range can be selected by jumpers (see page 20).

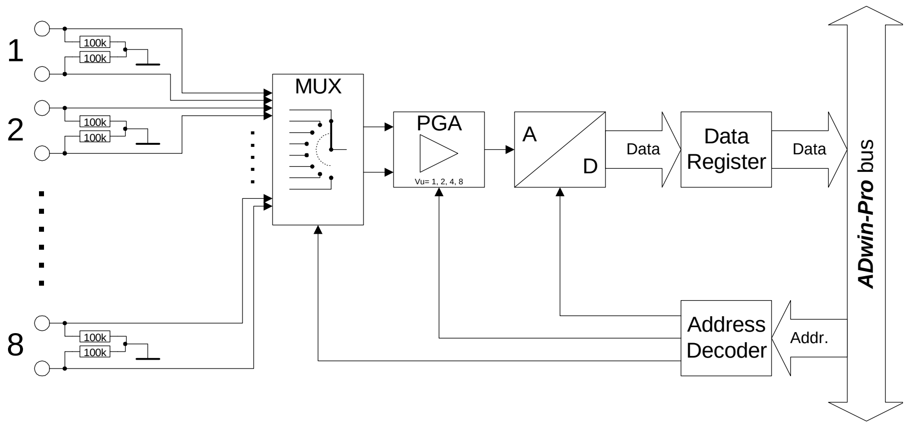


Fig. 24 – Pro-AIn-8/12 REVA: Block diagram

Input channels		8 differential via multiplexer
Resolution		12 bit
Conversion time		max. 8.5µs
Sampling rate		min. 117ksps
Measurement ranges		0…10V, ±5V, ±10V
Gain		software selectable: 1, 2, 4, 8
Accuracy	INL	max. ±1 LSB
	DNL	max. ±1 LSB
Input resistance		100kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC of full scale range
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 25 – Pro-AIn-8/12-D REVA: Specification

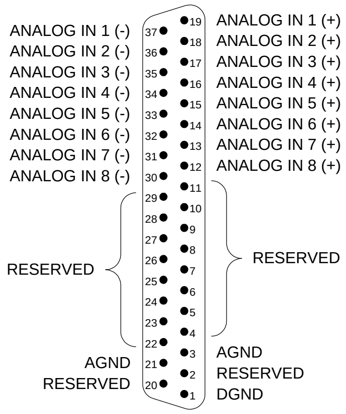


Fig. 26 – Pro-AIn-8/12-D REVA: Pin assignment

Setting the Input Voltage Range

The input module Pro-AIN-8/12 REVA is equipped with an ADC whose input voltage range is adjustable by 2 jumpers. As a default setting, the ADC is set to the voltage range of ±10V. The settings for other voltage ranges can be found in fig. 28.

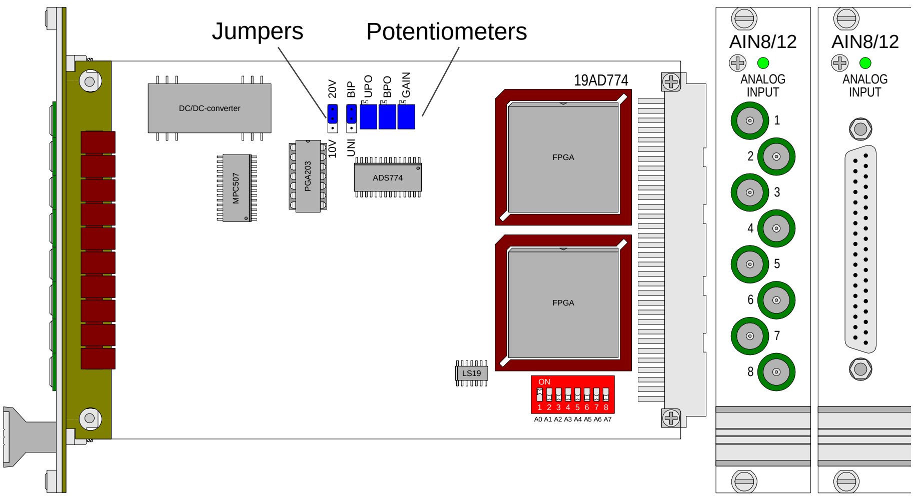


Fig. 27 – Pro-AIn-8/12 REVA: Board and front panel

After every jumper setting you have to recalibrate the ADC, in order to assure correct measurement results. The individual steps are described in the chapter 5 "Calibration" (page 173).

For the accurate adjustment of offset and gain the potentiometers UPO (unipolar) or BPO (bipolar) as well as GAIN are available (fig. 29).

Voltage range	J1	J2
±5V bipolar	BIP	10V
±10V bipolar (default)	BIP	20V
0...10V unipolar	UNI	10V
not allowed	UNI	20V

Fig. 28 – Pro-AIn-8/12 REVA: Jumper positions for the input voltage range

Potentiometer	Adjustment of
Gain	Gain factor
BPO	Offset (bipolar setting)
UPO	Offset (unipolar setting)

Fig. 29 – Pro-AIn-8/12 REVA: Function of the potentiometers

4.3.2 Pro-AIn-8/12 REVB

To this module you find an improved successor module Pro-AIn-8/14 REVA (see page 23).

The analog input module Pro-AIn-8/12 REV B has a 12 bit ADC and 8 differential inputs. The inputs are equipped with shielded LEMO-sockets (CAMAC European norm). A programmable amplifier (PGA) and a multiplexer (MUX) are connected before the ADC.

The module can be combined with amplifiers, filters, Pro-TC and Pro-PT modules.

The input voltage range of the ADC can be set by DIL switches (see page 22).

The module Pro-AIn-8/12 REV B is an advanced development of the module Pro-AIn-8/12 REVA with an input voltage range of ± 10V or 0...10V and a gain, programmable by software of 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software", page 176).

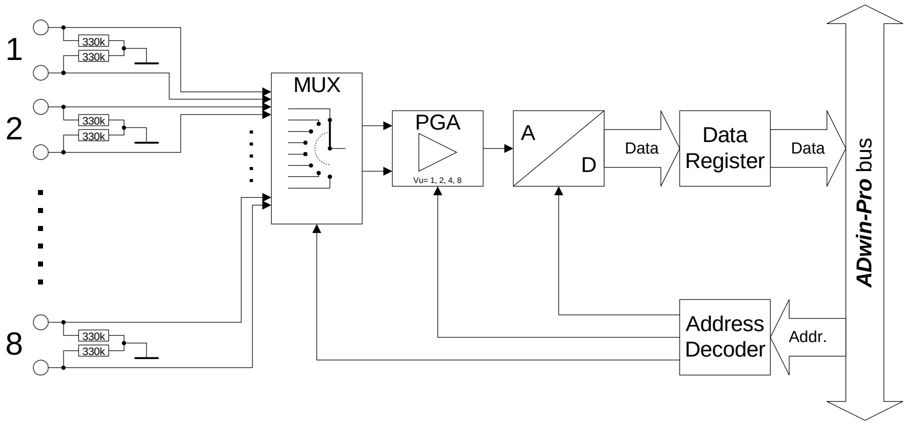


Fig. 30 – Pro-AIn-8/12 REVB: Block diagram

Input channels		8 differential via multiplexer
Resolution		12 bit
Conversion time		max. 0.75µs
Sampling rate		min. 1250ksps
Multiplexer settling time		3µs
Measurement ranges		0…10V, ±10V
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	typ. ±0.3 LSB, max. ±1 LSB
	DNL	typ. ±0.3 LSB, max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±17V
Offset error		adjustable
Offset drift		±30 ppm/ºC
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 31 – Pro-AIn-8/12 REVB: Specification

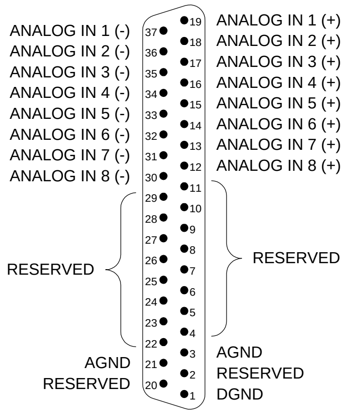


Fig. 32 – Pro-AIn-8/12-D REVB: Pin assignment

Setting the Input Voltage Range

The input module Pro-AIn-8/12 REVB is equipped with an ADC whose input voltage range is adjustable by 2 DIL switches. As a default setting, the ADC is set to the voltage range of ±10V. The settings for other voltage ranges can be found in fig. 34.

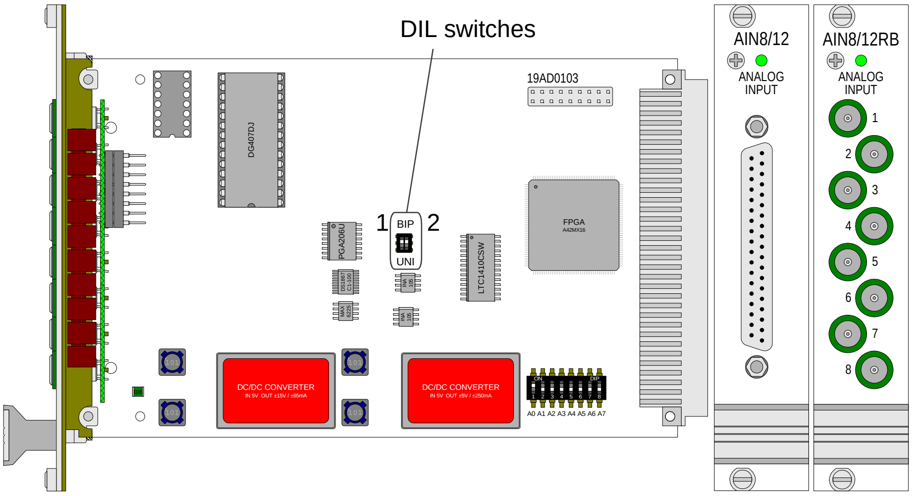


Fig. 33 – Pro-AIn-8/12 REVB: Board and front panel

After every jumper setting you have to recalibrate the ADC, in order to assure correct measurement results. The adjustment of gain and offset is made by software. The individual steps are described in the chapter 5.3.1 "Calibration per Software", page 176.

Voltage range	DIL1	DIL2
±10V bipolar (default)	BIP	BIP
0…10V unipolar	UNI	UNI
not allowed	BIP	UNI
not allowed	UNI	BIP

Fig. 34 – Pro-AIn-8/12: DIL switch settings for the input voltage range

4.3.3 Pro-AIn-8/14 REVA

The analog input module Pro-AIn-8/14 REVA has a 14 bit ADC and 8 single ended inputs. The input are equipped with shielded LEMO-sockets (CAMAC European norm). A programmable amplifier (PGA) and a multiplexer (MUX) are connected before the ADC. The module can be combined with Pro-TC and Pro-PT modules.

The module Pro-AIn-8/14 REVA is an advanced development of the module Pro-AIn-8/12 REVB with an input voltage range of ± 10V and a gain, programmable by software of 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").
The module includes a sequence control, which may read the measurement values of all input channel sequentially.

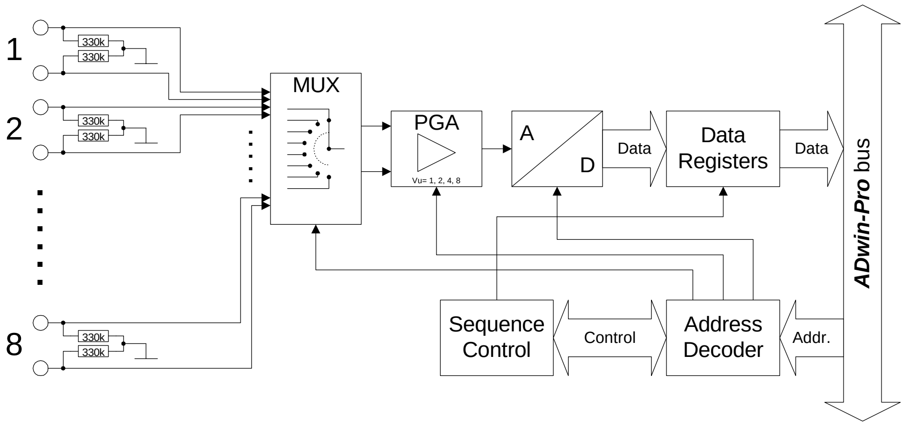


Fig. 35 – Pro-AIn-8/14 REVA: Block diagram

Input channels		8 differential via multiplexer
Resolution		14 bit
Conversion time		max. 0.5µs
Sampling rate		min. 2000ksps
Multiplexer settling time		3µs
Measurement ranges		±10V, optional: ±20mA / 8 channels
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	typ. ±0.6 LSB, max. ±2 LSB
	DNL	typ. ±0.3 LSB, max. ±1 LSB
	With the option ±20mA there is an additional inaccuracy of 0.05% of the measured voltage (by the 500Ω shunt).	
Input resistance		330kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 36 – Pro-AIn-8/14 REVA: Specification

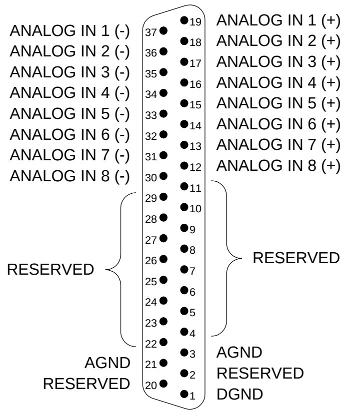


Fig. 37 – Pro-AIn-8/14-D REVA: Pin assignment

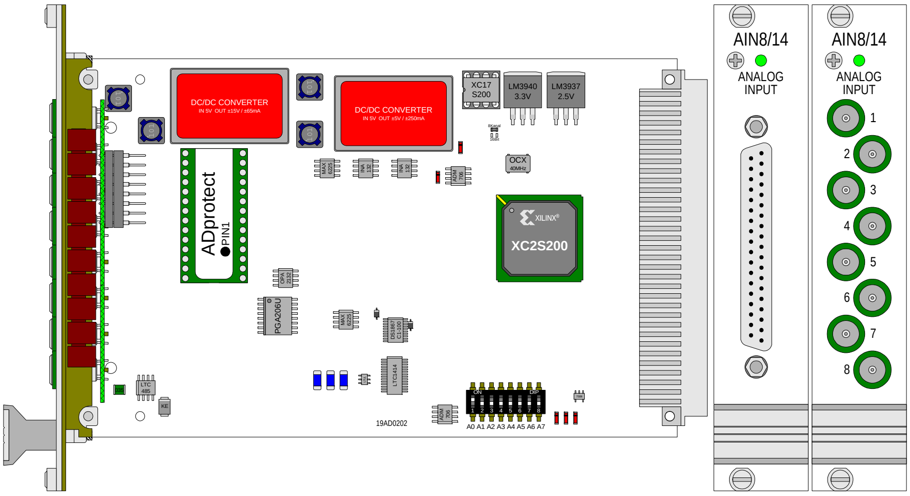


Abb. 38 – Pro-AIn-8/14 REVA: Board and front panel

4.3.4 Pro-AIn-32/12 REVA

To this module you find an improved successor module Pro-AIn-32/14 REVA (see page 30).

The analog input module Pro-AIn-32/12 REVA is equipped with a 12-bit ADC and a programmable amplifier. It has 32 single-ended or 16 differential inputs (software-selectable). The inputs are connected to a 37-pin DSub socket. The Pro-AIn-32/12 module offers options for combination with thermocouples and PTC-modules. The input voltage range of the ADC can be adjusted by jumpers (see page 26).

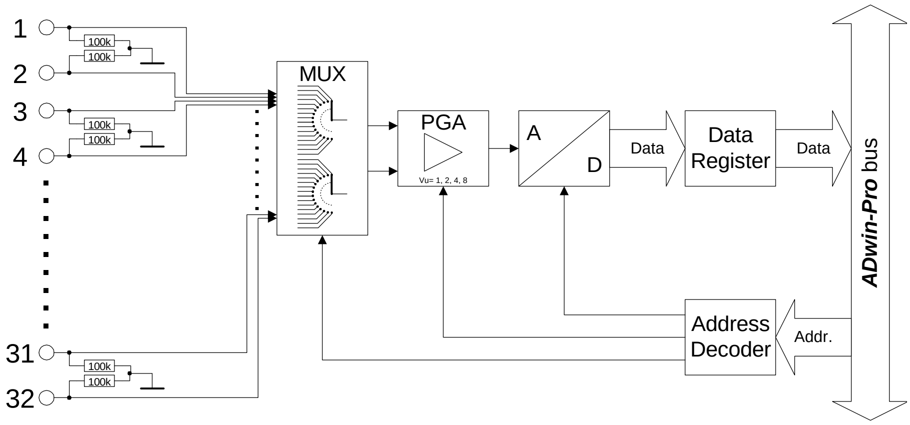


Fig. 39 – Pro-AIn-32/12 REVA: Block diagram

After power up the module Pro-AIn-32/12 is set to 16 differential inputs.

The figures 41 and 42 show the pin assignment of the module. Please consider the different pin assignment for differential and single-ended inputs.

Input channels		32 single-ended oder 16 differential
Resolution		12 bit
Conversion time		max. 8.5µs
Sampling rate		min. 117ksps
Measurement ranges		0…10V, ±5V, ±10V optional 0-20mA / 16 channels
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	max. ±1 LSB
	DNL	max. ±1 LSB
Input resistance		100kΩ, ±2%
Input over-voltage		±25V (peak ±35V)
Offset error		adjustable
Offset drift		±30 ppm/ºC of full scale range
Connector		37-pin DSub socket

Fig. 40 – Pro-AIn-32/12 REVA: Specification

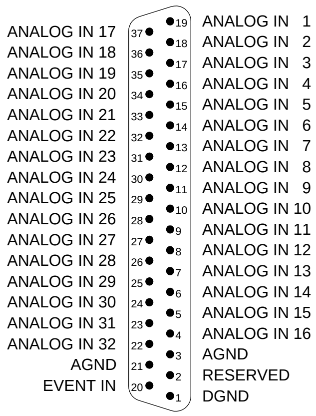


Fig. 41 – Pro-AIn-32/12 REVA: Pin assignment single-ended

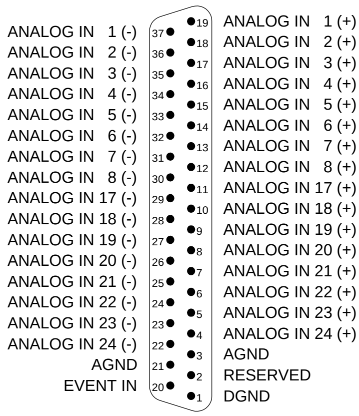


Fig. 42 – Pro-AIn-32/12 REVA: Pin assignment differential

Setting the Input Voltage Range

The input module Pro-AIN-32/12 REVA is equipped with an ADC whose input voltage range is adjustable by 2 jumpers. As a default setting, the ADC is set to the voltage range of ±10V. The settings for other voltage ranges can be found in fig. 44.

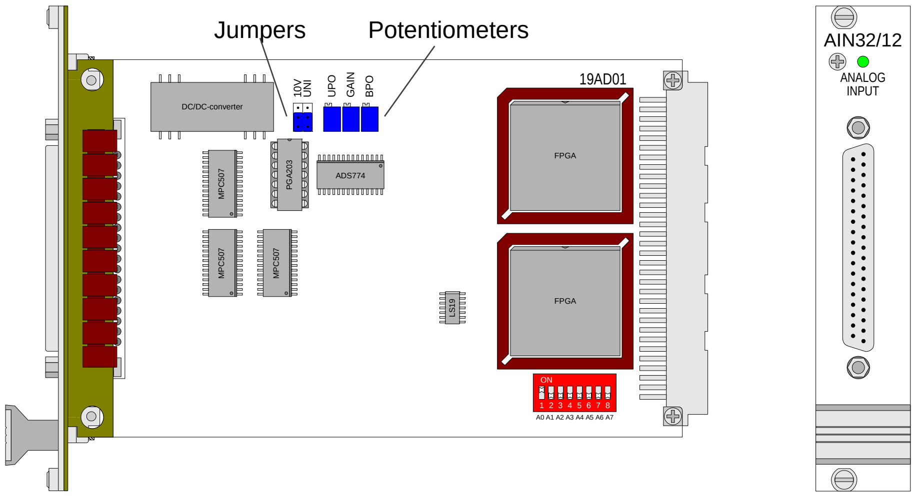


Fig. 43 – Pro-AIn-32/12 REVA: Board and front panel

After every jumper setting you have to recalibrate the ADC, in order to assure correct measurement results. The individual steps are described in the chapter 5 "Calibration".

If you have a board with the imprint "19AD774" (in the upright corner) the jumpers are arranged in positions different to this description. Please ask our support to get the correct jumper positions.

For the accurate adjustment of offset and gain the potentiometers UPO (unipolar) or BPO (bipolar) as well as GAIN are available (fig. 45).

Voltage range	J1	J2
±5V bipolar	BIP	10V
±10V bipolar (default)	BIP	20V
0...10V unipolar	UNI	10V
not allowed (0…20V)	UNI	20V

Fig. 44 – Pro-AIn-32/12 REVA: Jumper positions

Potenti-ometer	Adjustment of
Gain	Gain factor
BPO	Offset (bipolar setting)
UPO	Offset (unipolar setting)

Fig. 45 – Pro-AIn-32/12 REVA: Function of the potentiometers

4.3.5 Pro-AIn-32/12 REVB

To this module you find an improved successor module Pro-AIn-32/14 REVA (see page 30).

The analog input module Pro-AIn-32/12 REVB has a 12 bit ADC and a programmable gain (PGA). It is equipped with 32 single-ended inputs and 16 differential inputs (software-selectable). The inputs are connected on a 37-pin DSub socket. The module can be combined with Pro-TC and Pro-PT modules. The input voltage range of the ADC can be set by jumpers (see page 29).

The module Pro-AIn-32/12 REV B is an advanced development of the module Pro-AIn-32/12 with an input voltage range of ± 10V or 0-10V and a gain, programmable by software of 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").

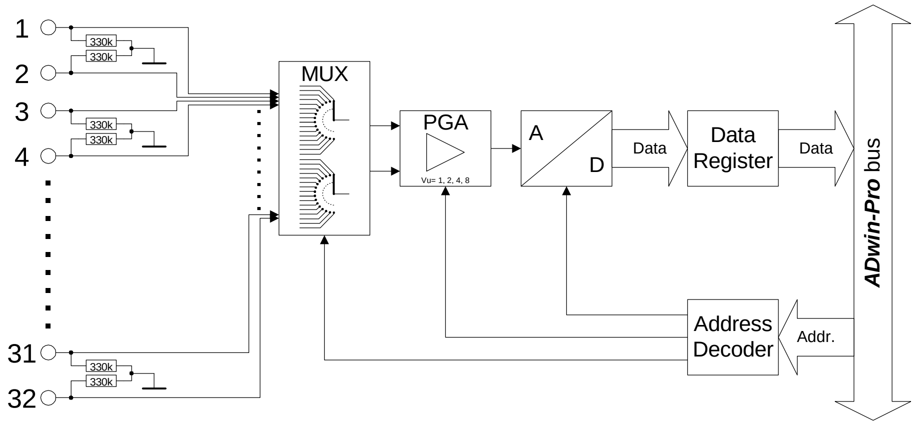


Fig. 46 – Pro-AIn-32/12 REVB: Block diagram

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

Figures 48 and 49 show the pin assignment of the module. Please consider the different pin assignment for differential and single-ended inputs.

Input channels		32 single-ended oder 16 differential
Resolution		12 bit
Conversion time		max. 0.75µs
Sampling rate		min. 1250ksps
Multiplexer settling time		3µs
Measurement ranges		0…10V, ±10V optional 0…20mA / 16 channels
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	typ. ±0.3 LSB, max. ±1 LSB
	DNL	typ. ±0.3 LSB, max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±17V
Offset error		adjustable
Offset drift		±30 ppm/ºC
Connector		37-pin DSub socket

Fig. 47 – Pro-AIn-32/12 REVB: Specification

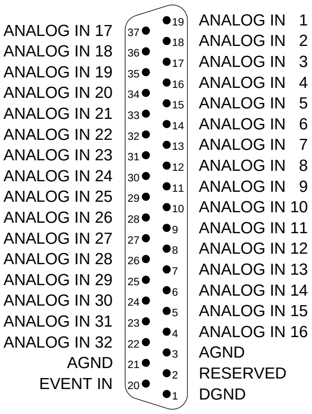


Fig. 48 – Pro-AIn-32/12 REVB: Pin assignment single-ended

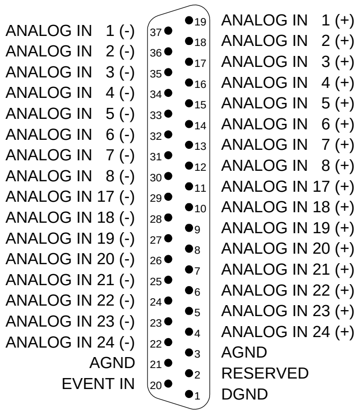


Fig. 49 – Pro-AIn-32/12 REVB: Pin assignment differential

Setting the Input Voltage Range

The input module Pro-AIn-32/12 REVB is equipped with an ADC whose input voltage range is adjustable by 2 DIL switches. As a default setting, the ADC is set to the voltage range of ±10V. The settings for other voltage ranges can be found in fig. 51.

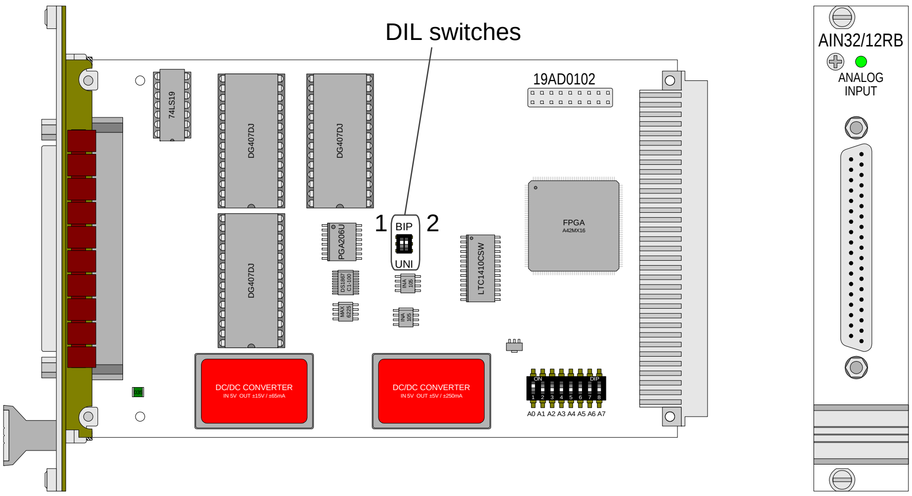


Fig. 50 – Pro-AIn-32/12 REVB: Board and front panel

After every jumper setting you have to recalibrate the ADC, in order to assure correct measurement results. The adjustment of gain and offset is made by software. The individual steps are described in the chapter 5.3.1 "Calibration per Software", page 176.

Voltage range	DIL1	DIL2
±10V bipolar (default)	BIP	BIP
0…10V unipolar	UNI	UNI
not allowed	BIP	UNI
not allowed	UNI	BIP

Fig. 51 – Pro-AIn-32/12 REVB: DIL switch settings for the input voltage range

4.3.6 Pro-AIn-32/14 REVA

The analog input module Pro-AIn-32/14 REVA has a 14 bit ADC and a programmable gain (PGA). It is equipped with 32 single-ended inputs or 16 differential inputs (software-selectable). The inputs are connected to a 37-pin DSUB socket. The module can be combined with Pro-TC and Pro-PT modules.

The module Pro-AIn-32/14 REVA is an advanced development of the module Pro-AIn-32/12 REVB. It has an input voltage range of ±10V and a gain, programmable by software of 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").
In addition, the module has a sequential control, which by request reads the measurement values from all input channels successively.

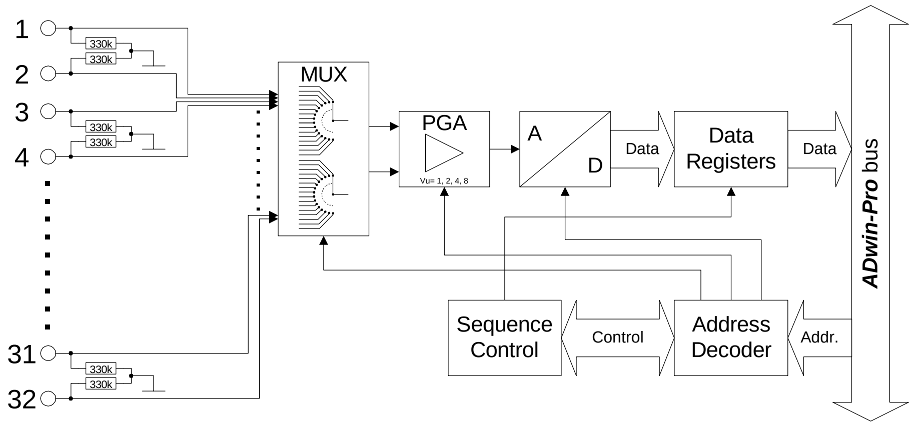


Fig. 52 – Pro-AIn-32/14 REVA: Block diagram

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

Figures 54 and 55 show the pin assignment of the module. Please consider the different pin assignment for differential and single-ended inputs.

Input channels		32 single-ended oder 16 differential
Resolution		14 bit
Conversion time		max. 0.5µs
Sampling rate		min. 2000ksps
Multiplexer settling time		3µs
Measurement ranges		±10V; optional 0…20mA / 16 channels
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	typ. ±0.6 LSB, max. ±1 LSB
	DNL	typ. ±0.3 LSB, max. ±1 LSB
	With the option ±20mA there is an additional inaccuracy of 0.05% of the measured voltage (by the 500Ω shunt).	
Input resistance		330kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC
Connector		37-pin DSub socket

Fig. 53 – Pro-AIn-32/14 REVA: Specification

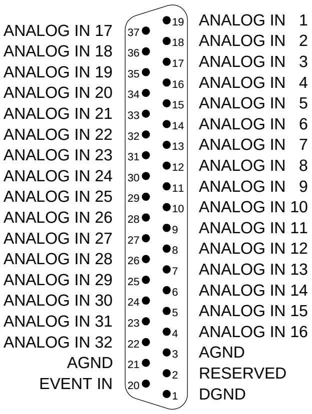


Fig. 54 – Pro-AIn-32/14 REVA:
Pin assignment single-ended

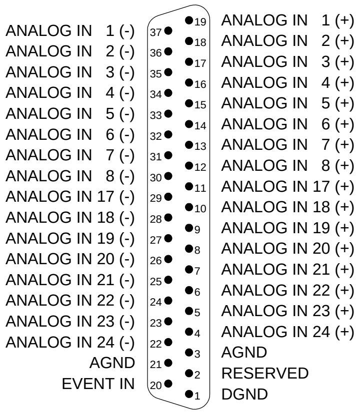


Fig. 55 – Pro-AIn-32/14 REVA:
Pin assignment differential

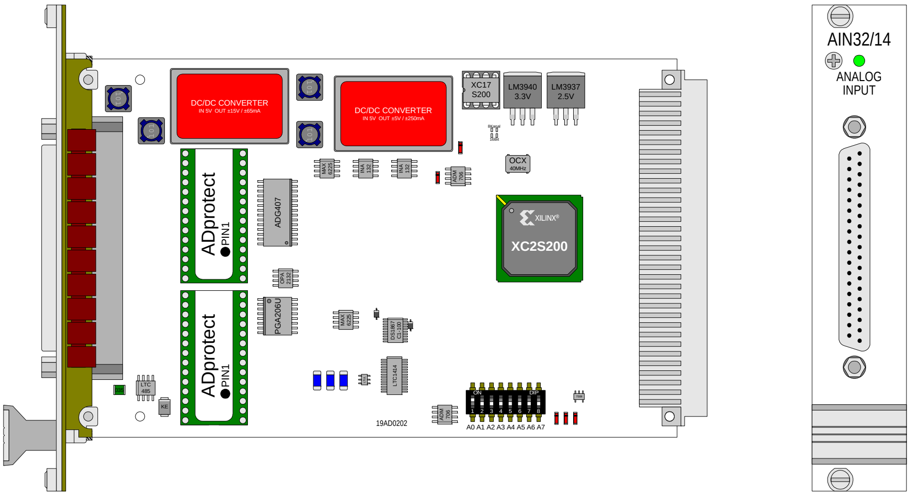


Abb. 56 – Pro-AIn-32/14 REVA: Board and front panel

4.3.7 Pro-AIn-8/16 REVA

To this module you find an improved successor module Pro-AIn-8/16 REVC (see page 36).

Analog input module Pro-AIn-8/16 REVA with a 16-bit ADC and 8 differential inputs.

The inputs are equipped with shielded LEMO-sockets (CAMAC European Norm). The module Pro-AIn-8/16 REVA has options for combination with thermocouples, filters and PTC-modules.

The input voltage range of the ADC can be adjusted by jumpers (see below).

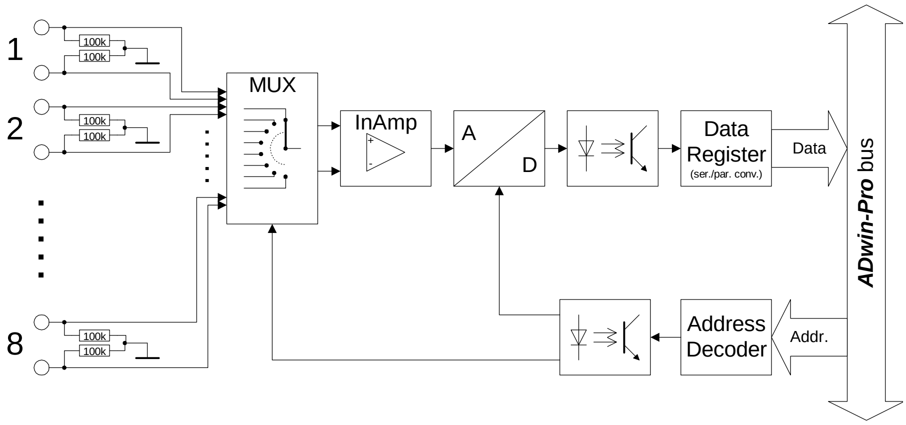


Fig. 57 – Pro-AIn-8/16 REVA: Block diagram

Input channels		8 differential via multiplexer
Resolution		16 bit
Conversion time		max. 10µs
Sampling rate	1 channel continuously	min. 100ksps
	channels multiplexed	min. 66ksps
	discontinuously	min. 50ksps
Measurement ranges		0…10V, ±5V, ±10V
Accuracy	INL	max. ±3 LSB
	DNL	max. +3, -2 LSB
Input resistance		100kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC of full scale range
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 58 – Pro-AIn-8/16 REVA: Specification

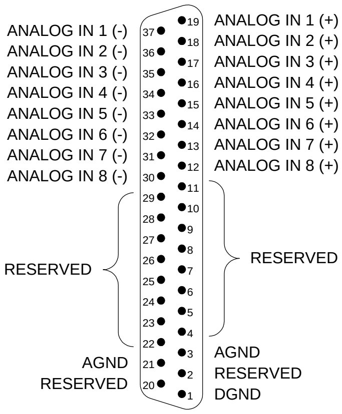


Fig. 59 – Pro-AIn-8/16-D REVA: Pin assignment

Setting the Input Voltage Range

The input module Pro-AIn-8/16 REVA is equipped with an ADC whose input voltage range is adjustable by 3jumpers. As a default setting, the ADC is set to the voltage range of ±10V. The settings for other voltage ranges can be found in fig. 61.

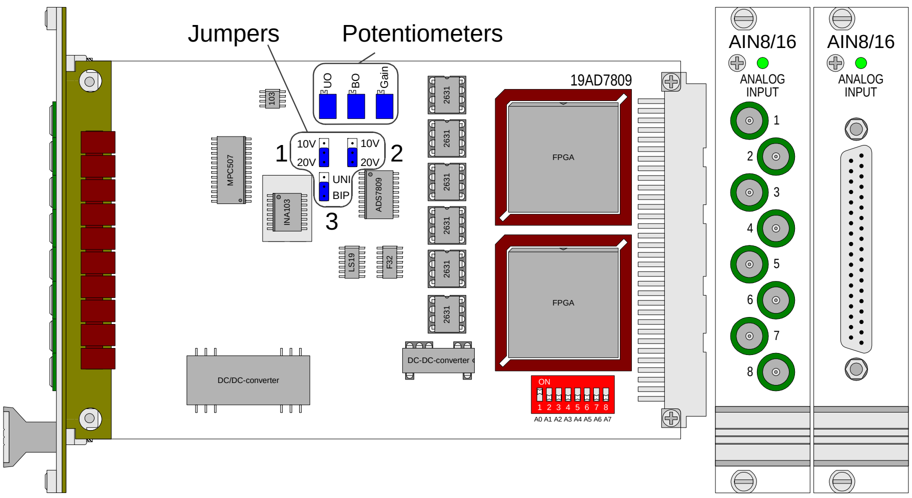


Fig. 60 – Pro-AIn-8/16 REVA: Board and front panel

After every jumper setting you have to recalibrate the ADC, in order to assure correct measurement results. The individual steps are described in the chapter 5 "Calibration".

For the accurate adjustment of offset and gain the potentiometers UO (unipolar) or BO (bipolar) as well as GAIN are available (fig. 62).

Voltage range	J1	J2	J3
±5V bipolar	10V	20V	BIP
±10V bipolar (default)	10V	20V	BIP
0...10V unipolar	10V	10V	UNI
not allowed (0...20V)	20V	20V	UNI

Fig. 61 – Pro-AIn-8/16 REVA: Jumper settings for the input voltage range

Potentiometer	Adjustment of
Gain	Gain factor
BPO	Offset (bipolar setting)
UPO	Offset (unipolar setting)

Fig. 62 – Pro-AIn-8/16 REVA: Function of the potentiometers

4.3.8 Pro-AIn-8/16 REVB

To this module you find an improved successor module Pro-AIn-8/16 REVC (see page 36).

The analog input module Pro-AIn-8/16 REV B has a 16-bit ADC, 8 differential inputs and a programmable gain (PGA). The inputs are equipped with shielded LEMO-sockets (CAMAC European norm).

The module can be combined with amplifiers, filters, Pro-TC and Pro-PT modules.

The module Pro-AIn-8/16 REV B is an advanced development of the module Pro-AIn-8/16 with an input voltage range of ± 10V and a gain, programmable by software of 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").

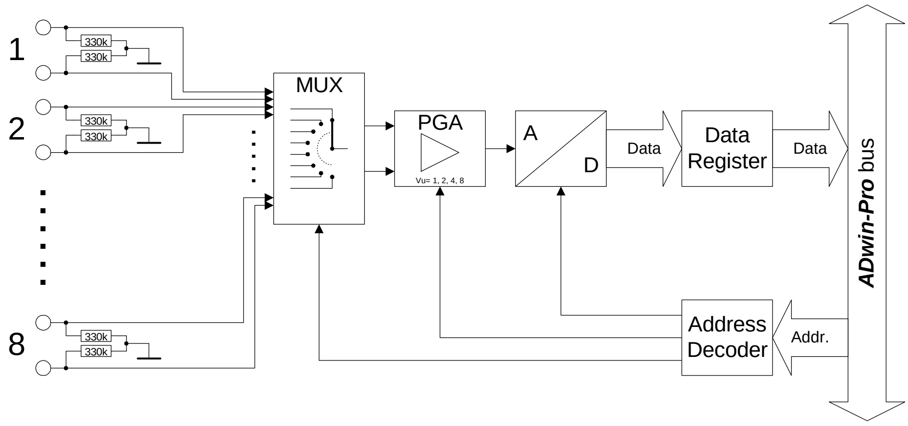


Fig. 63 – Pro-AIn-8/16 REVB: Block diagram

Input channels		8 differential via multiplexer
Resolution		16 bit
Conversion time		max. 8µs
Sampling rate		min. 100ksps
Multiplexer settling time		14µs
Measurement ranges		±10V
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	±3 LSB typical
	DNL	max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±17V
Offset error		adjustable
Offset drift		±20 ppm/ºC
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 64 – Pro-AIn-8/16 REVB: Specification

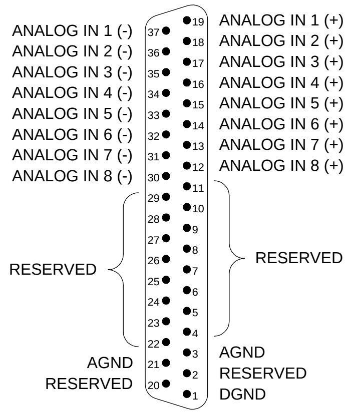


Fig. 65 – Pro-AIn-8/16-D REVB: Pin assignment differential

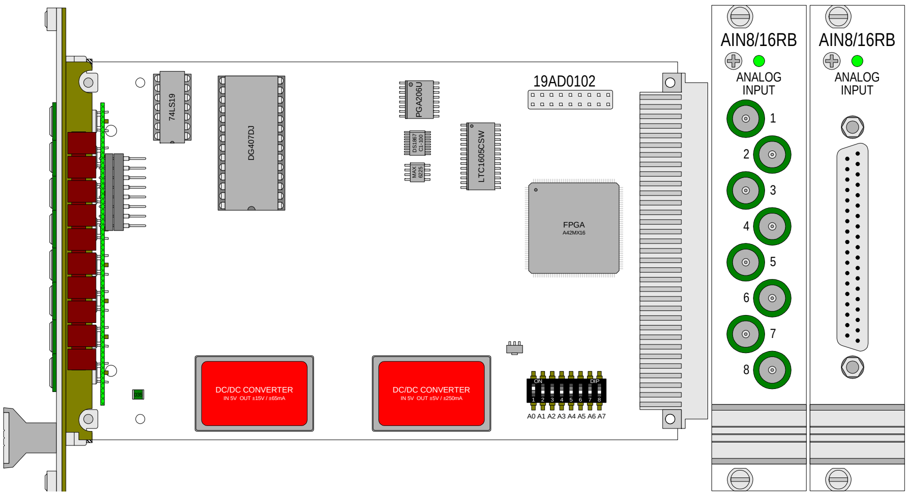


Fig. 66 – Pro-AIn-8/16 REVB: Board and front panel

4.3.9 Pro-AIn-8/16 REVC

The analog input module Pro-AIn-8/16 REVC has a 16-bit ADC, 8 differential inputs and a programmable gain (PGA). The inputs are equipped with shielded LEMO-sockets (CAMAC European norm). The module can be combined with amplifiers, filters, Pro-TC and Pro-PT modules.

The module Pro-AIn-8/16 REVC is an advanced development of the Pro-AIn-8/16 REVB with an input voltage range of ± 10V and a gain, programmable by software of 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").

In addition, the module has a sequential control, which by request reads the measurement values from all input channels successively.

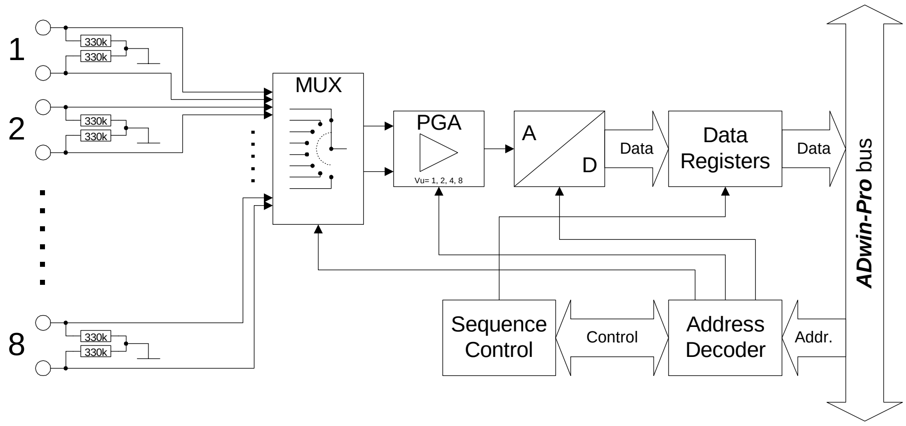


Fig. 67 – Pro-AIn-8/16 REVC: Block diagram

Input channels		8 differential via multiplexer
Resolution		16 bit
Conversion time		max. 5µs
Sampling rate		min. 200ksps
Multiplexer settling time		6µs
Measurement ranges		±10V
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	±2 LSB typical
	DNL	max. ±1 LSB
Input resistance		330kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 68 – Pro-AIn-8/16 REVC: Specification

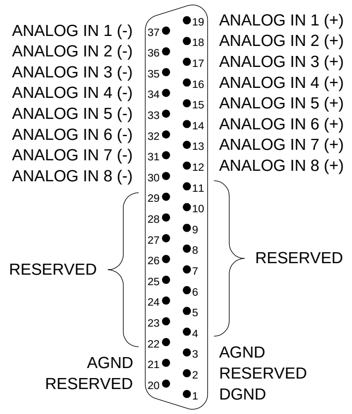


Fig. 69 – Pro-AIn-8/16-D REVC: Pin assignment differential

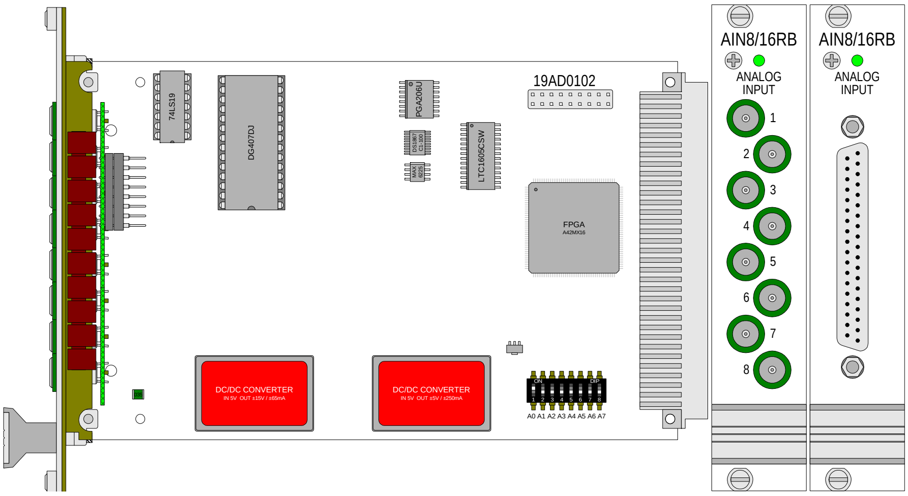


Fig. 70 – Pro-AIn-8/16 REVC: Board and front panel

4.3.10 Pro-AIn-32/16 REVB

To this module you find an improved successor module Pro-AIn-32/16 REVC (see page 40).

The analog input module Pro-AIn-32/16RB has a 16-bit ADC and a programmable gain amplifier (PGA). It has 32 single-ended or 16 differential inputs (software selectable). The inputs are connected on a 37-pin DSUB socket. The module can be combined with Pro-TC and Pro-PT modules.

The module Pro-AIn-32/16RB is equipped with an input voltage range of ± 10V and a gain amplifier, programmable by software to 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").

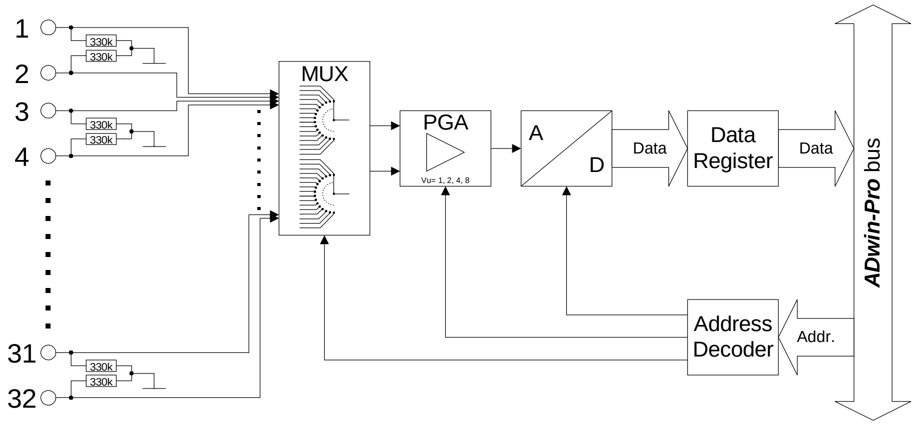


Fig. 71 – Pro-AIn-32/16 REVB: Block diagram

On Startup the module is set to 16 differential inputs.

Figures 73 and 74 show the pin assignments of the module. Please consider the different pin assignment for differential and single-ended inputs.

Input channels:		32 single-ended oder 16 differetial; via multiplexer
Resolution:		16 bit
Conversion time:		max. 8 µs
Sampling rate:		min. 100 ksps
Multiplexer settling time:		14 µs
Measurement ranges:		±10 V
Gain:		1, 2, 4, 8 software selectable
Accuracy	INL	max. ±3 LSB
	DNL	max. +3, -2 LSB
Input resistance:		330 kΩ, ±2%
Input over-voltage:		±17 V
Offset error:		adjustable
Offset drift:		±20 ppm/ºC
Connector:		37-pin DSub socket

Fig. 72 – Pro-AIn-32/16 REVB: Specification

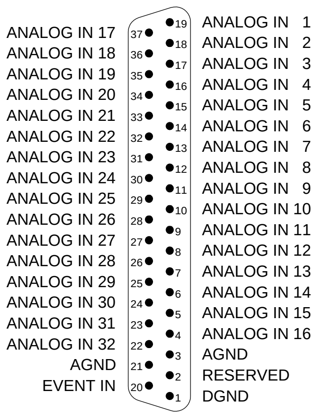


Fig. 73 – Pro-AIn-32/16 REVB: Pin assignment single-ended

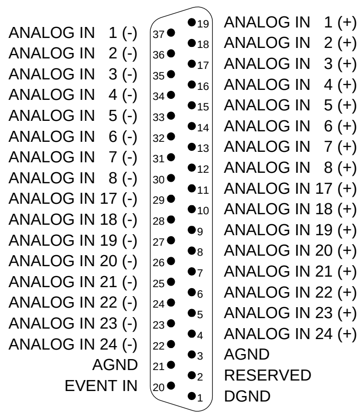


Fig. 74 – Pro-AIn-32/16 REVB: Pin assignment differential

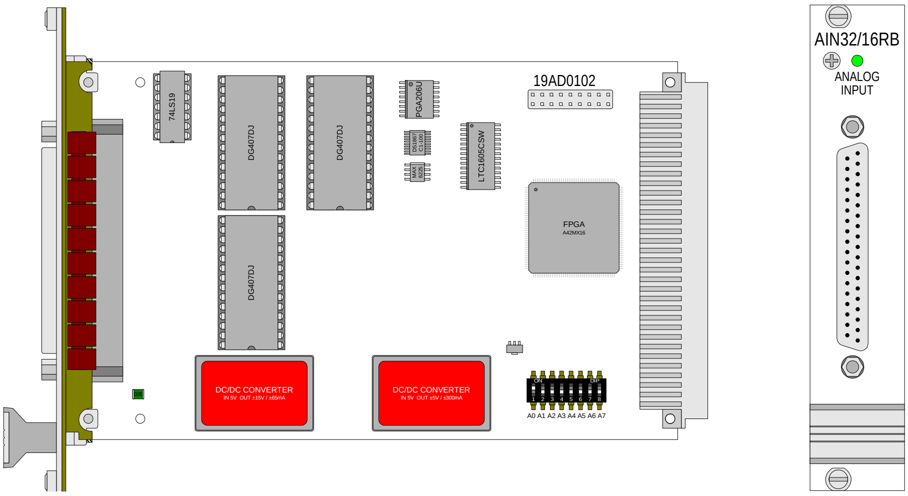


Fig. 75 – Pro-AIn-32/16 REVB: Board and front panel

4.3.11 Pro-AIn-32/16 REVC

The analog input module Pro-AIn-32/16 REVC has a 16-bit ADC and a programmable gain amplifier (PGA). It has 32 single-ended or 16 differential inputs (software selectable). The inputs are connected to a 37-pin DSUB socket. The module can be combined with amplifiers, filters, Pro-TC and Pro-PT modules.

The module Pro-AIn-32/16 REVC is equipped with an input voltage range of ±10V and a gain amplifier, programmable by software to 1, 2, 4 or 8. The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").

In addition, the module has a sequential control, which by request reads the measurement values from all input channels successively.

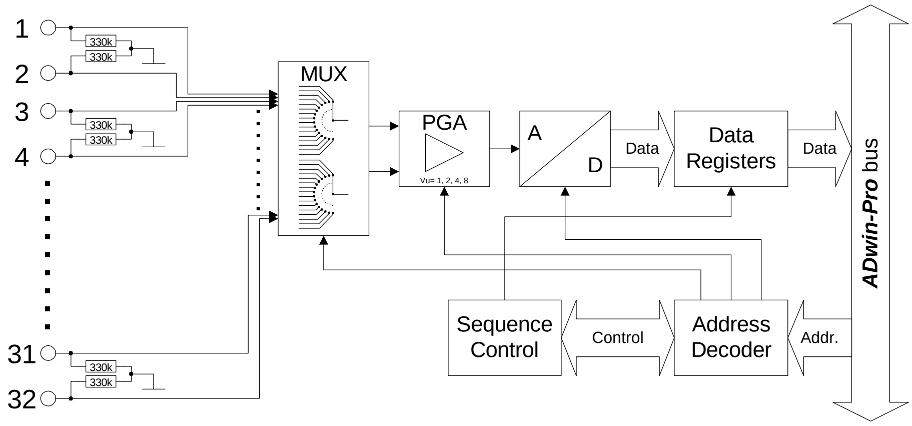


Fig. 76 – Pro-AIn-32/16 REVC: Block diagram

On Startup the module is set to 16 differential inputs.

Figures 78 and 79 show the pin assignments of the module. Please consider the different pin assignment for differential and single-ended inputs.

Input channels:		32 single-ended oder 16 differential; via multiplexer
Resolution:		16 bit
Conversion time:		max. 5µs
Sampling rate:		min. 200ksps
Multiplexer settling time:		6µs
Measurement ranges:		±10V
Gain:		1, 2, 4, 8 software selectable
Accuracy	INL	max. ±2 LSB
	DNL	max. +1 LSB
Input resistance:		330kΩ, ±2%
Input over-voltage:		±35V
Offset error:		adjustable
Offset drift:		±30 ppm/ºC
Connector:		37-pin DSub socket

Fig. 77 – Pro-AIn-32/16 REVC: Specification

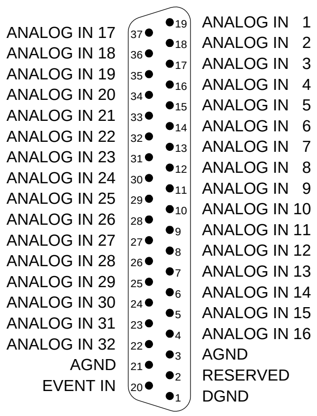


Fig. 78 – Pro-AIn-32/16 REVC: Pin assignment single-ended

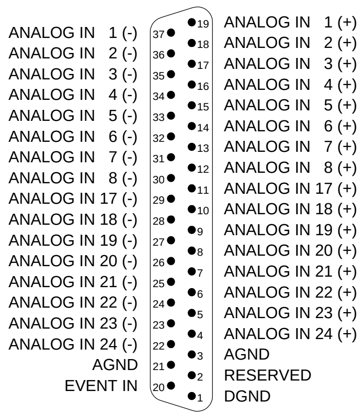


Fig. 79 – Pro-AIn-32/16 REVC: Pin assignment differential

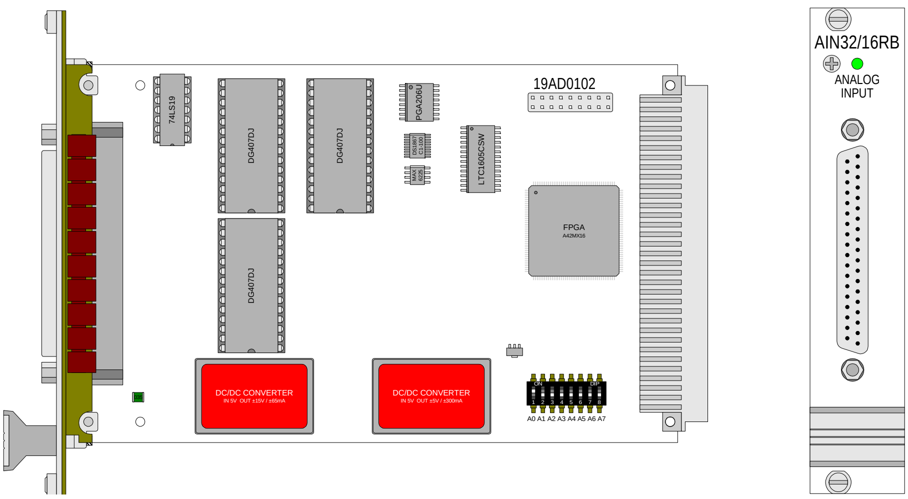


Fig. 80 – Pro-AIn-32/16 REVC: Board and front panel

4.3.12 Pro-AIn-F-4/12 REVA

Analog input module Pro-AIn-F-4/12 with 4 ADC (12 bit) and 4 differential inputs.

The inputs are equipped with shielded LEMO sockets (CAMAC European norm).

For the accurate adjustment of offset and gain the potentiometers Ox and Gx are available (fig. 84); you find information about the accurate adjustment on page 45. The "x" of the potentiometer's names are place holders for the number of the corresponding ADC. The potentiometer names are imprinted on the board.

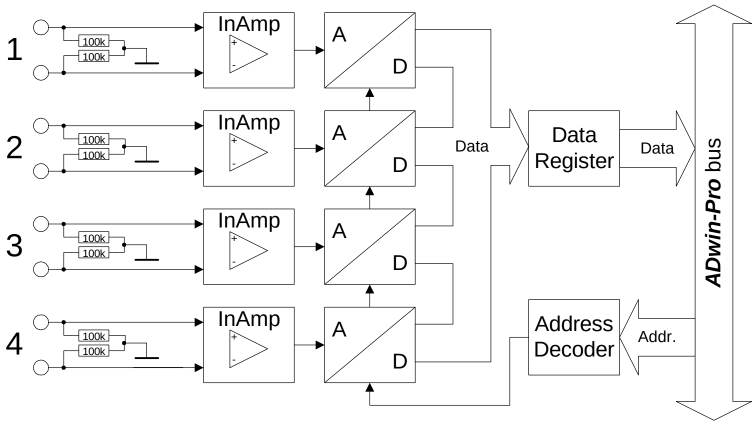


Fig. 81 – Pro-AIn-F-4/12 REVA: Block diagram

Input channels		4 differential
Resolution		12 bit
Conversion time		max. 0.75 µs (per ADC)
Sampling rate		min. 1250 ksps (per ADC)
Measurement ranges		±10 V
Accuracy	INL	typ. ±0.3 LSB, max. ±1 LSB
	DNL	typ. ±0.3 LSB, max. ±1 LSB
Input resistance		100 kΩ, ±2%
Input over-voltage		±35 V
Offset error		adjustable
Offset drift		±30 ppm/°C of full scale range
Connector		4 LEMO sockets optional: 37-pin DSub socket

Fig. 82 – Pro-AIn-F-4/12 REVA: Specification

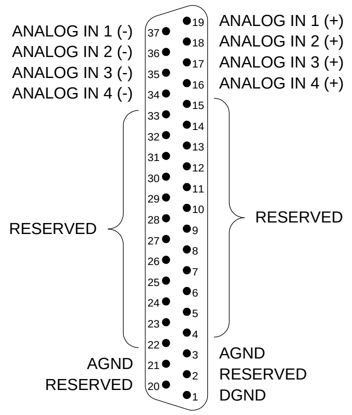


Fig. 83 – Pro-AIn-F-4/12-D REVA: Pin assignment differential

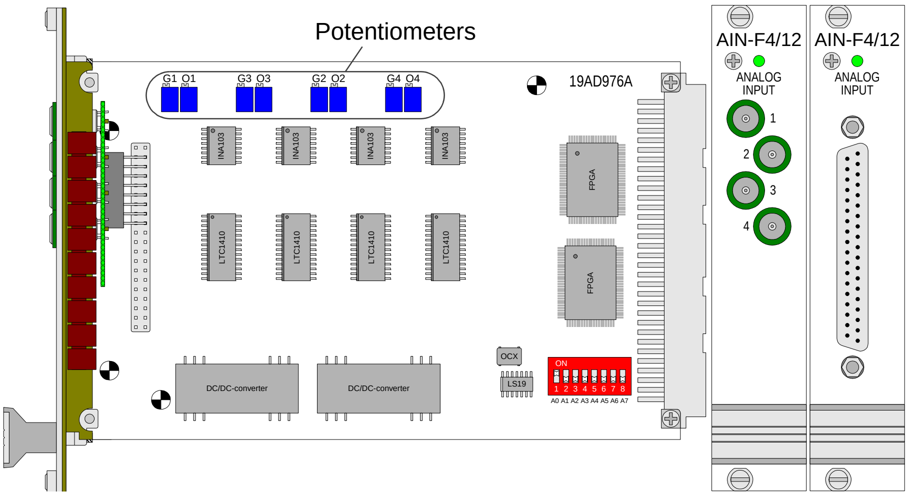


Fig. 84 – Pro-AIn-F-4/12 REVA: Board and front panel

4.3.13 Pro-AIn-F-8/12 REVA

Analog input module Pro-AIn-F-8/12 with 8 ADC (12 bit) and 8 differential inputs.

The inputs are equipped with shielded LEMO sockets (CAMAC European norm).

For the accurate adjustment of offset and gain the potentiometers Ox and Gx are available (fig. 88); you find information about the accurate adjustment on page 45.

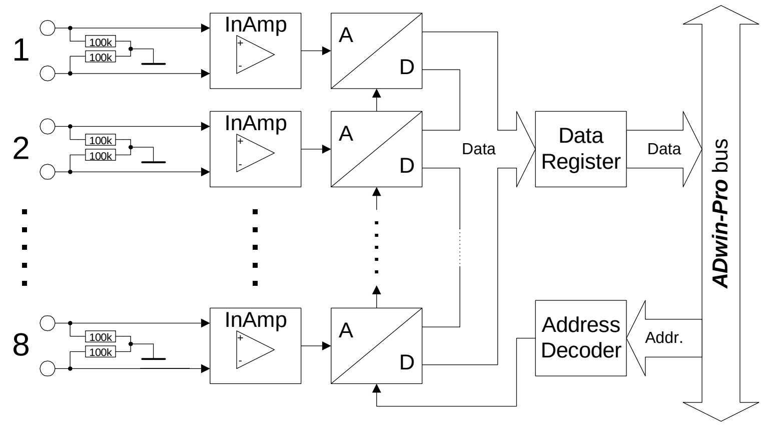


Fig. 85 – Pro-AIn-F-8/12 REVA: Block diagram

Input channels		8 differential
Resolution		12 bit
Conversion time		max. 0.75µs (per ADC)
Sampling rate		min. 1250ksps (per ADC)
Measurement ranges		±10V
Accuracy	INL	typ. ±0.3 LSB, max. ±1 LSB
	DNL	typ. ±0.3 LSB, max. ±1 LSB
Input resistance		100kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC of full scale range
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 86 – Pro-AIn-F-8/12 REVA: Specification

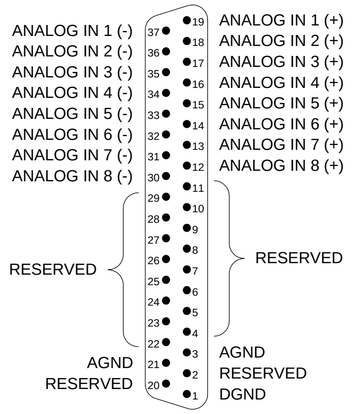


Fig. 87 – Pro-AIn-F-8/12-D REVA: Pin assignment differential

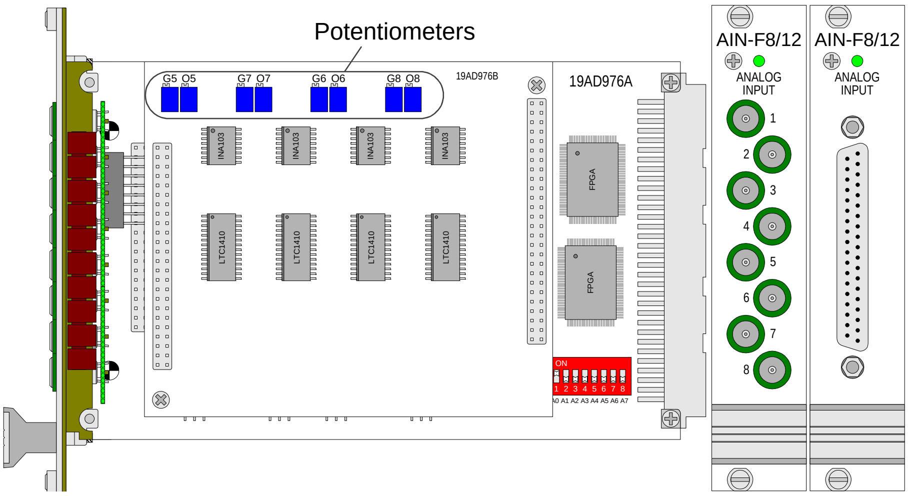


Fig. 88 – Pro-AIn-F-8/12 REVA: Board and front panel

Setting Offset and Gain factor

The analog input modules Pro-AIn-F-4/12 REVA and Pro-AIn-F-4/16 REVA are equipped with each 4 ADC, the modules Pro-AIn-F-8/12 REVA and Pro-AIn-F-8/16 REVA with each 8 ADC. The ADC 1 to 4 are on the base PCB and the ADC 5 to 8 are on an additional PCB which will be plugged into the base PCB. The input voltage of the ADC is fixed to ±10V.

The potentiometers Ox and Gx are used for an accurate adjustment of gain and offset (fig. 89). The "x" in the potentiometer's names stands for the number of the corresponding ADC. The potentiometer names are printed on the boards.

When testing the modules, the potentiometers have been optimally adjusted. Therefore we ask you to avoid adjusting the potentiometers if not necessary, because this may result in inaccuracy. The calibration of the ADC is described in the chapter 5 "Calibration".

Potentiometer	Adjustment of
Gx	Gain factor
Ox	Offset

Fig. 89 – Pro-AIn-F-8/12 REVA: Function of the potentiometers

4.3.14 Pro-AIn-F-4/14 REVB

Analog input module Pro-AIn-F-4/14 REVB with 4 fast-ADC (14 bit) and 4 differential inputs. The inputs are equipped with shielded LEMO sockets (CAMAC European norm).

The module has 2 operating modes, which you can use optionally for each of the channels:

– Standard individual measurement: Upon each process call an individual measurement can be executed by starting the conversion (waiting for its end), by reading out the measurement values and if necessary by processing them.

– Start of a burst measurement sequence: Upon each process call a complete measurement sequence, which consists of a large number of individual measurements, is started. The module executes the measurement sequence alone, independent of the ***ADwin*** system.

The measurement values - number and measurement frequency are to be defined in the program - are stored in a special burst memory of the module, which enables measurement frequencies of up to 2MHz. The size of the burst memory limits the number of measurement values or the total time of the measurement.

Only after end of the measurement sequence will you be able to read out the stored measurement values from the burst memory and process them.

During an individual measurement the processor of the ***ADwin-Pro*** system is checking the sequence of each individual measurement, whereas during a burst measurement it is the module which is checking the measurement sequence.

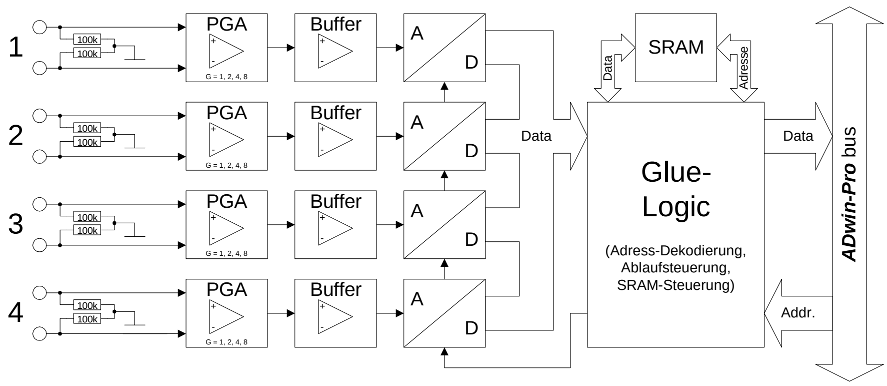


Fig. 90 – Pro-AIn-F-4/14 REVB: Block diagram

Input channels		4 differential
Resolution		14 bit
Conversion time		max. 0.4µs (per ADC)
Sampling rate		min. 2200ksps (per ADC)
Memory size		2^{20}-1 = 1048575 measurement values total
Measurement ranges		±10V
Accuracy	INL	max. ±2 LSB (>1000ksps ±3 LSB, >1500ksps ±4 LSB)
	DNL	±2 LSB typical
Input resistance		100kΩ, ±2%
Input over-voltage		±35V

Fig. 91 – Pro-AIn-F-4/14 REVB: Specification

Offset error	adjustable
Offset drift	±30 ppm/ºC of full scale range
Connector	4 LEMO sockets optional: 37-pin DSub socket

Fig. 91 – Pro-AIn-F-4/14 REVB: Specification

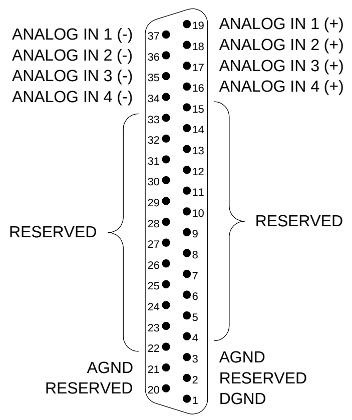


Fig. 92 – Pro-AIn-F-4/14-D REVB: Pin assignment differential

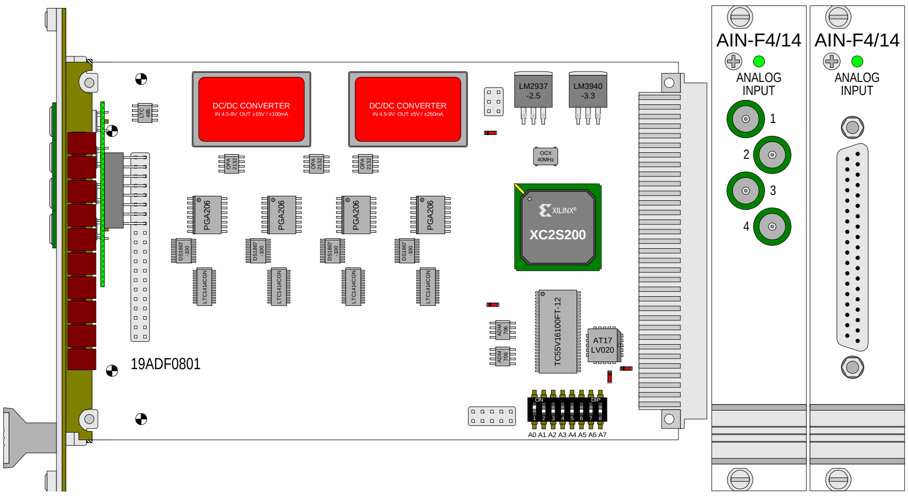


Fig. 93 – Pro-AIn-F-4/14 REVB: Board and front panel

4.3.15 Pro-AIn-F-8/14 REVB

Analog input module Pro-AIn-F-8/14 REVB with 8 fast-ADC (14 bit) and 8 differential inputs. The inputs are equipped with shielded LEMO sockets (CAMAC European norm).

The module has 2 operating modes, which you can use optionally for each of the channels:

- Standard single measurement: Upon each process call a single measurement can be executed by starting the conversion (waiting for its end), by reading out the measurement values and if necessary by processing them.
- Start of a burst measurement sequence: Upon each process call a complete measurement sequence, which consists of a large number of individual measurements, is started. The module executes the measurement sequence alone, independent of the ***ADwin*** system.

The measurement values - number and measurement frequency are to be defined in the program - are stored in a special burst memory of the module, which enables measurement frequencies of up to 2 MHz. The size of the burst memory limits the number of measurement values or the total time of the measurement.

Only after end of the measurement sequence will you be able to read out the stored measurement values from the burst memory and process them.

During a single measurement the processor of the ***ADwin-Pro*** system is checking the sequence of each individual measurement, whereas during a burst measurement it is the input module which is checking the measurement sequence.

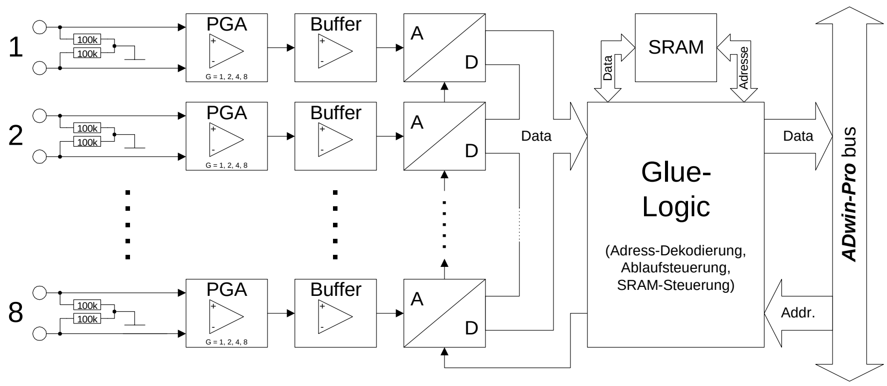


Fig. 94 – Pro-AIn-F-8/14 REVB: Block diagram

Input channels		8 differential
Resolution		14 bit
Conversion time		max. 0.4 µs (per ADC)
Sampling rate		min. 2200 ksps (per ACDC)
Memory size		2^{20}-1 = 1048575 meas. values total
Measurement ranges		±10 V
Accuracy	INL	max. ±2 LSB (>1000 ksps ±3 LSB, >1500 ksps ±4 LSB)
	DNL	±2 LSB typical
Input resistance		100 kΩ, ±2%
Input over-voltage		±35 V

Fig. 95 – Pro-AIn-F-8/14 REVB: Specification

Offset error	adjustable
Offset drift	±30 ppm/ºC of full scale range
Connector	8 LEMO sockets optional: 37-pin DSub socket

Fig. 95 – Pro-AIn-F-8/14 REVB: Specification

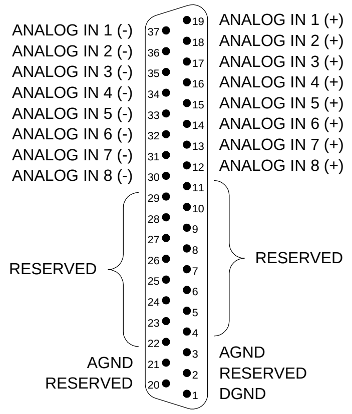


Fig. 96 – Pro-AIn-F-8/14-D REVB: Pin assignment differential

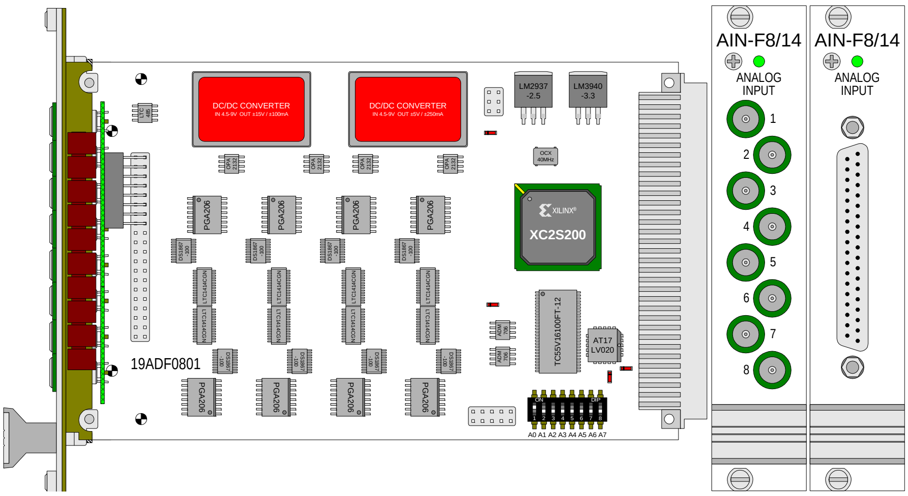


Fig. 97 – Pro-AIn-F-8/14 REVB: Board and front panel

4.3.16 Pro-AIn-F-4/16 REVA

Analog input module Pro-AIn-F-4/16 REVA with 4 ADC (16 bit) and 4 differential inputs.

The inputs are equipped with shielded LEMO sockets (CAMAC European norm).

For the accurate adjustment of offset and gain the potentiometers Ox and Gx are available (fig. 101); you find information about the accurate adjustment on page 45. The "x" of the potentiometer's names are place holders for the number of the corresponding ADC. The potentiometer names are imprinted on the board.

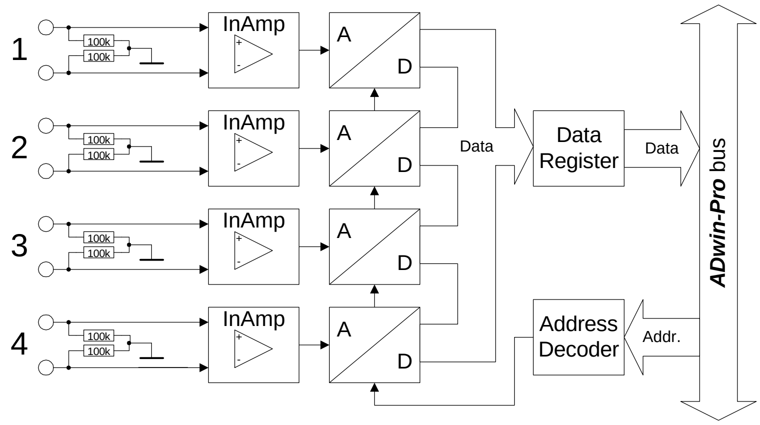


Fig. 98 – Pro-AIn-F-4/16 REVA: Block diagram

Input channels		4 differential
Resolution		16 bit
Conversion time		max. 8µs (per ADC)
Sampling rate		min. 100ksps (per ADC)
Measurement ranges		±10V
Accuracy	INL	±3 LSB typical
	DNL	max. ±1 LSB
Input resistance		100kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC of full scale range
Connector		4 LEMO sockets optional: 37-pin DSub socket

Fig. 99 – Pro-AIn-F-4/16 REVA: Specification

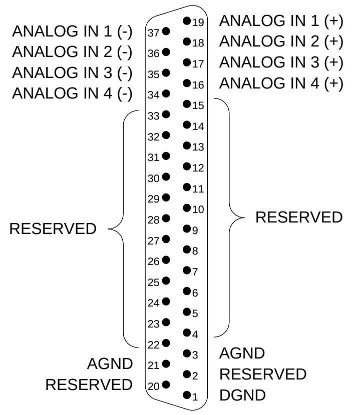


Fig. 100 – Pro-AIn-F-4/16-D REVA: Pin assignment differential

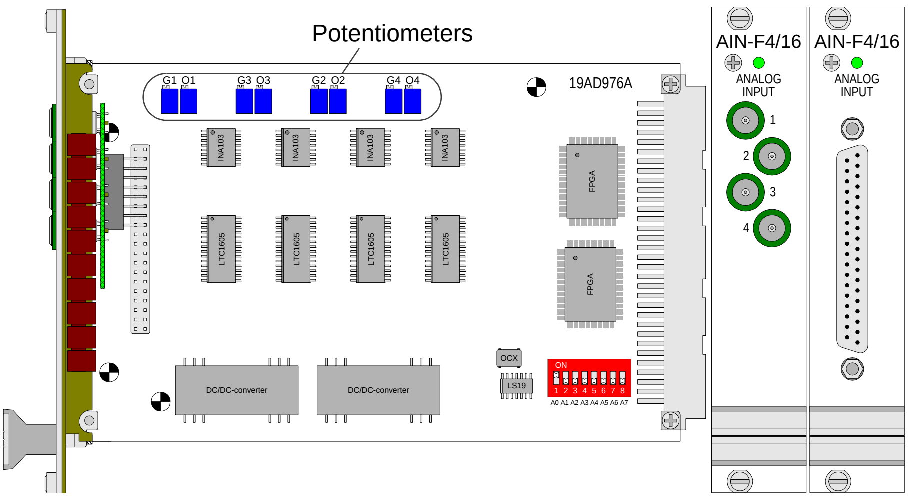


Fig. 101 – Pro-AIn-F-4/16 REVA: Board and front panel

4.3.17 Pro-AIn-F-8/16 REVA

Analog input module Pro-AIn-F-8/16 REVA with 8 ADC (16 bit) and 8 differential inputs.

The inputs are equipped with shielded LEMO-sockets (CAMAC European norm).

For the accurate adjustment of offset and gain the potentiometers Ox and Gx are available (fig. 105); you find information about the accurate adjustment on page 45. The "x" of the potentiometer's names are place holders for the number of the corresponding ADC. The potentiometer names are imprinted on the board.

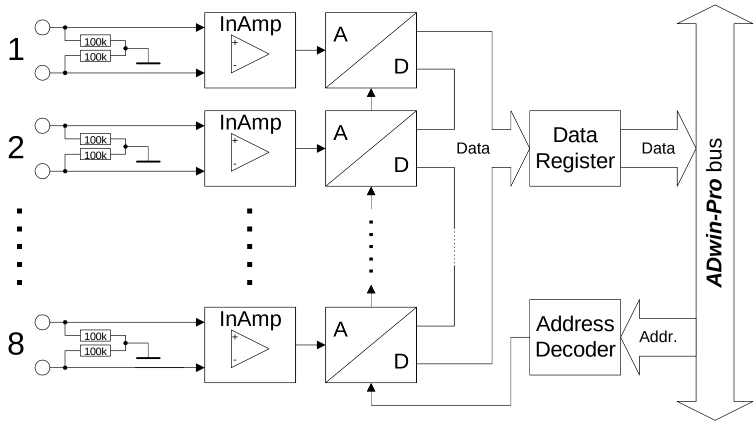


Fig. 102 – Pro-AIn-F-8/16 REVA: Block diagram

Input channels		8 differential
Resolution		16 bit
Conversion time		max. 8µs (per ADC)
Sampling rate		min. 100ksps (per ADC)
Measurement ranges		±10V
Accuracy	INL	±3 LSB typical
	DNL	max. ±1 LSB
Input resistance		100kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC of full scale range
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 103 – Pro-AIn-F-8/16 REVA: Specification

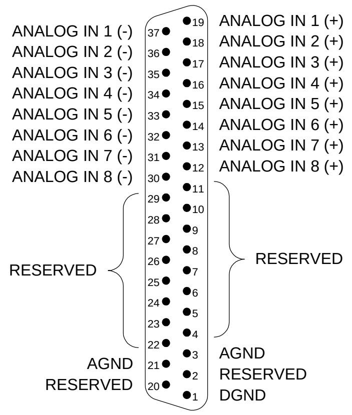


Fig. 104 – Pro-AIn-F-8/16-D REVA: Pin assignment differential

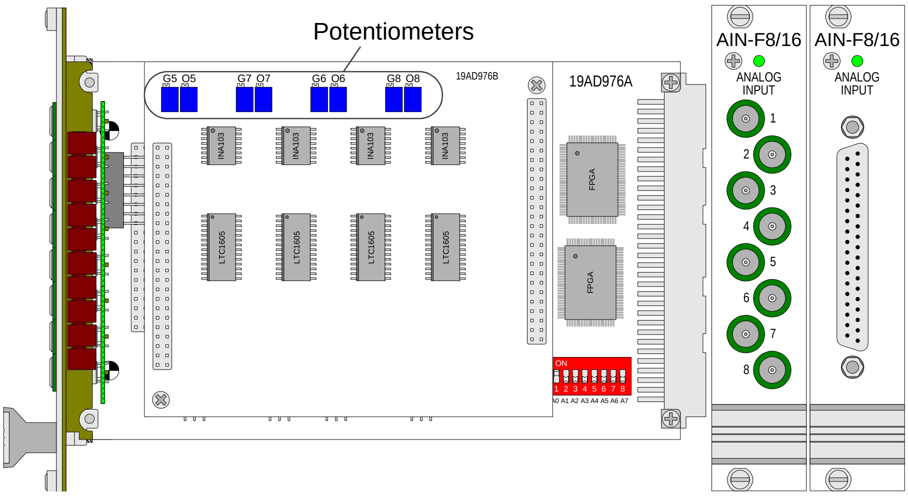


Fig. 105 – Pro-AIn-F-8/16 REVA: Board and front panel

4.4 Analog Output Modules

Modulname	AOut 4/16	AOut 4/16	AOut 4/16	AOut 4/16-M2	AOut 8/16	AOut 8/16	AOut 8/16
Rev.	A	B	C		A	B	C
No. ADC	4	4	4		8	8	8
Resolution [bit]	16	16	16		16	16	16
max. settling time [µs]	20 + 3	< 3	< 3		20 + 3	< 3	< 3
Channels sng. end.	4	4	4		8	8	8
Voltage range ±5V	x	x	–		x	x	–
±10V	x	x	x		x	x	x
0...5V	x	x	–		x	x	–
0…10V	x	x	–		x	x	–
add. memory (option)	–	–	–	2MB	–	–	–
Calibration[a]	TR	TR	SW		TR	TR	SW
page	56	60	63		58	60	65

a. SW: per software, TR: with trimmers

4.4.1 Pro-AOut-4/16 REVA

The analog output module Pro-AOut-4/16 REVA has 4 DAC (16 bit) with fixed 1st order low-pass filters (f_c = 100 kHz) to cut off glitches. The DAC are serial, causing an output delay of 3 µs.

The outputs are equipped with shielded LEMO sockets (CAMAC European norm).

Jumpers are used to set the output voltage range of the DAC (see page 59).

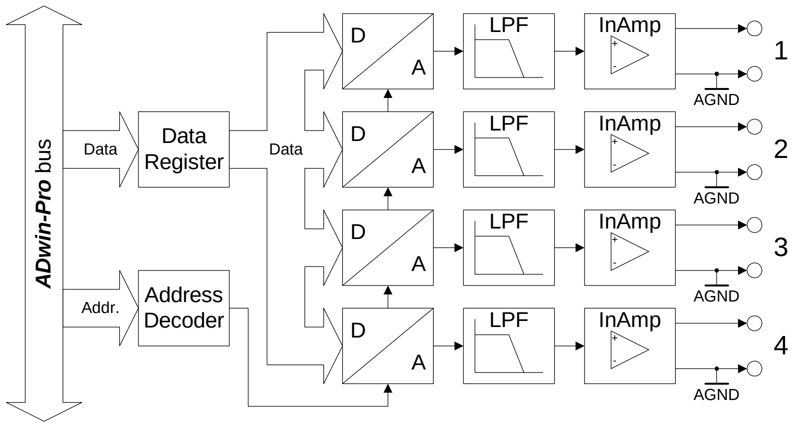


Fig. 106 – Pro-AOut-4/16 REVA: Block diagram

Output channels		4 single-ended
Resolution		16 bit
Settling time to 0.01% FSR		20 µs + 3 µs output delay, for full scale step
Output voltage		0…10 V, ±5 V, ±10 V
Output current max.		±5 mA per channel
Accuracy	INL	max. ±4 LSB
	DNL	max. ±4 LSB
Offset error		adjustable
Gain error		adjustable
Offset drift		±10 µV/ºC
Connector		4 LEMO sockets optional: 37-pin DSub socket

Fig. 107 – Pro-AOut-4/16 REVA: Specification

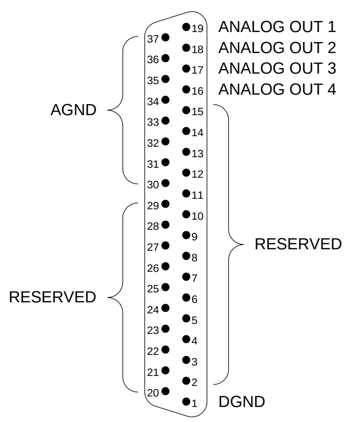


Fig. 108 – Pro-AOut-4/16-D REVA: Pin assignment differential

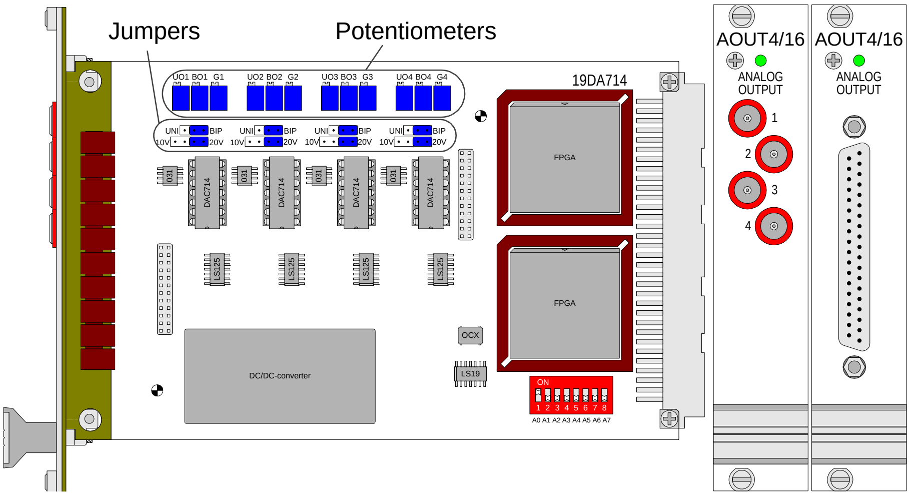


Fig. 109 – Pro-AOut-4/16 REVA: Board and front panel

4.4.2 Pro-AOut-8/16 REVA

The analog output module Pro-AOut-8/16 has 8 DAC (16 bit) with fixed 1st order low-pass filters (f_c = 100kHz) to cut off glitches. The DAC are serial, causing an output delay of 3µs.

The outputs are equipped with shielded LEMO sockets (CAMAC European norm).

Jumpers are used to set the output voltage range of the DAC (see page 59).

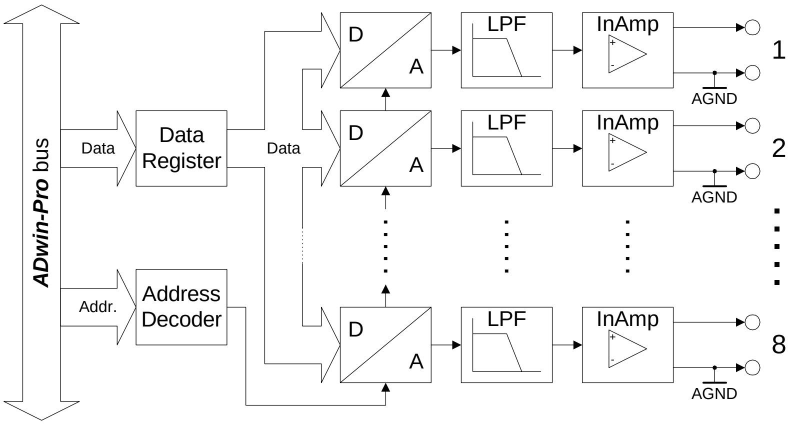


Fig. 110 – Pro-AOut-8/16 REVA: Block diagram

Output channels		8 single-ended
Resolution		16 bit
Settling time to 0.01% FSR		20µs + 3µs output delay, for full scale step
Output voltage		0…10V, ±5V, ±10V
Output current max.		±5mA per channel
Accuracy	INL	max. ±4 LSB
	DNL	max. ±2 LSB
Offset error		adjustable
Gain error		adjustable
Offset drift		±10µV/ºC
Connector		8 LEMO sockets optional: 37-pin DSub socket

Fig. 111 – Pro-AOut-8/16 REVA: Specification

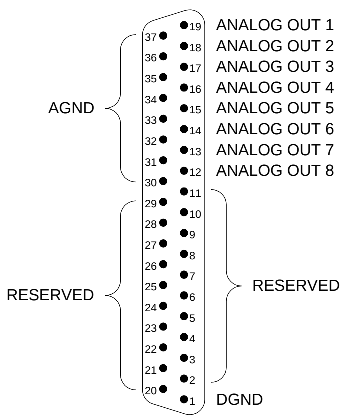


Fig. 112 – Pro-AOut-8/16-D REVA: Pin assignment differential

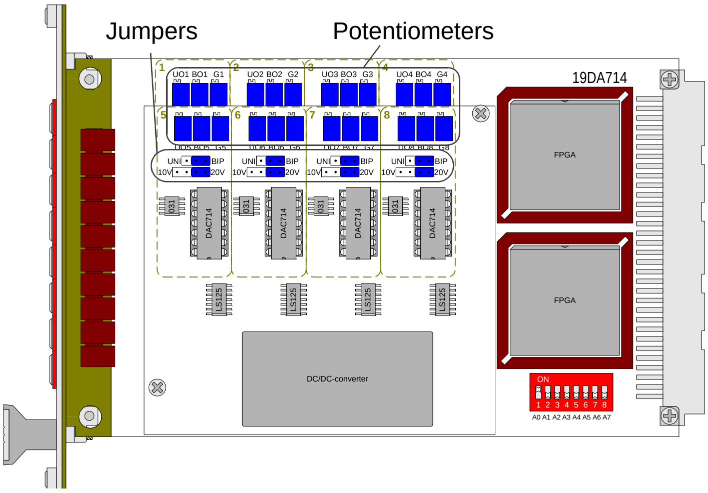


Fig. 113 – Pro-AOut-8/16 REVA: Board

Setting the Output Voltage Range

The module Pro-AOut-4/16 REVA is equipped with 4 DAC, the module Pro-AOut-8/16 REVA with 8 DAC. The DAC 1 to 4 are on the base PCB and the DAC 5 to 8 are mounted on an additional PCB which will be plugged into the base PCB.
The output voltage of the DAC can be set by two jumpers. As a default setting all DAC are set to the voltage range of ± 10V. If you want to adjust the DAC 1 to 4 on the base PCB of the output module Pro-AOut-8/16 you have to remove the additional PCB which is fixed by two screws.

In fig. 114 all possibilities to adjust the jumpers are listed The "x" in the potentiometer title stands for the number of the corresponding DAC (see PCB imprint near to the potentiometers).

The potentiometers U0x, Bx and Gainx are used for an accurate adjustment of gain and offset (fig. 115).

If nothing else has been said on ordering the module, the voltage range is set to ±10V. After every jumper setting you have to recalibrate the DAC, in order to assure correct measurement results. The individual steps are described in chapter 5.3.1 "Calibration per Software".

Voltage range	Jx1	Jx2
±5V bipolar	BIP	10V
±10V bipolar (default)	BIP	20V
0…10V unipolar	UNI	10V
not allowed (0…20V)	UNI	20V

Fig. 114 – Pro-AOut-8/16 REVA: Jumper positions for the output voltage range

Potentiometer	Adjustment of
Gain	Gain factor
BPO	Offset (bipolar setting)
UPO	Offset (unipolar setting)

Fig. 115 – Pro-AOut-8/16 REVA: Function of the potentiometers

4.4.3 Pro-AOut-4/16 REVB, Pro-AOut-8/16 REVB

The analog output module Pro-AOut-4/16 REVB (before: Version 2) has 4 DAC (16 bit). The module Pro-AOut-8/16 REVB (before: Version 2) has 8 DAC (16 bit).

Both modules have a fixed 1st order low-pass filter (f_c = 890kHz) in order to avoid interferences. The outputs are equipped with shielded LEMO-sockets (CAMAC European norm). The output voltage range of the DAC can be set by two DIL switches (see page 61). The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").

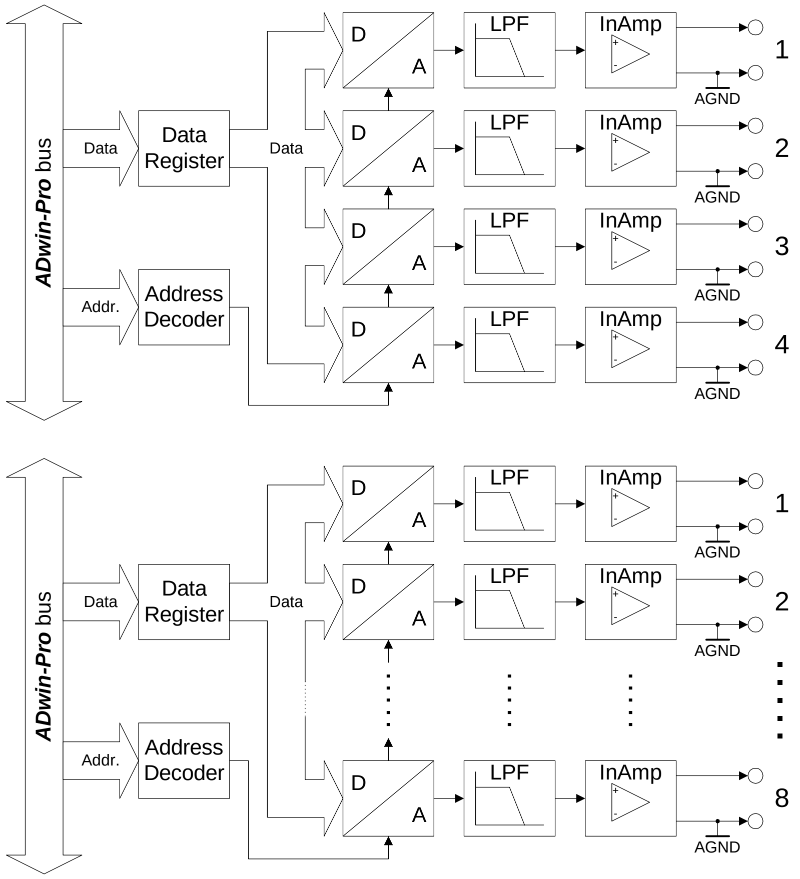


Fig. 116 – Pro-AOut-4/16 REVB, Pro-AOut-8/16 REVB: Block diagram

Output channels		4 bzw. 8 single-ended
Resolution		16 bit
Settling time to 0.01% FSR		< 3µs
Output voltage		0…10V, 0…5V, ±5V, ±10V
Output current max.		±5mA per channel
Accuracy	INL	±2 LSB typical
	DNL	±1 LSB typical
Offset error		adjustable
Gain error		adjustable
Offset drift		±10µV/ºC
Connector		4 or 8 LEMO sockets optional: 37-pin DSub socket

Fig. 117 – Pro-AOut-4/16 REVB, Pro-AOut-8/16 REVB: Specification

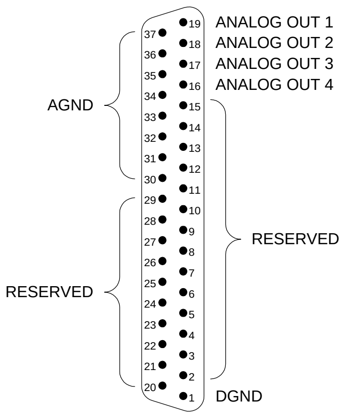


Fig. 118 – Pro-AOut-4/16 REVB: Pin assignment

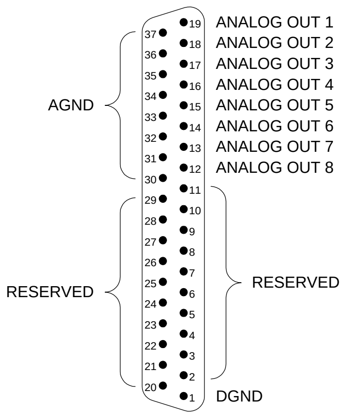


Fig. 119 – Pro-AOut-8/16 REVB: Pin assignment

Setting the Output Voltage Range

The output voltage range of every DAC can be set by 2 DIL switches. The default setting of the DAC is the voltage range ±10V.

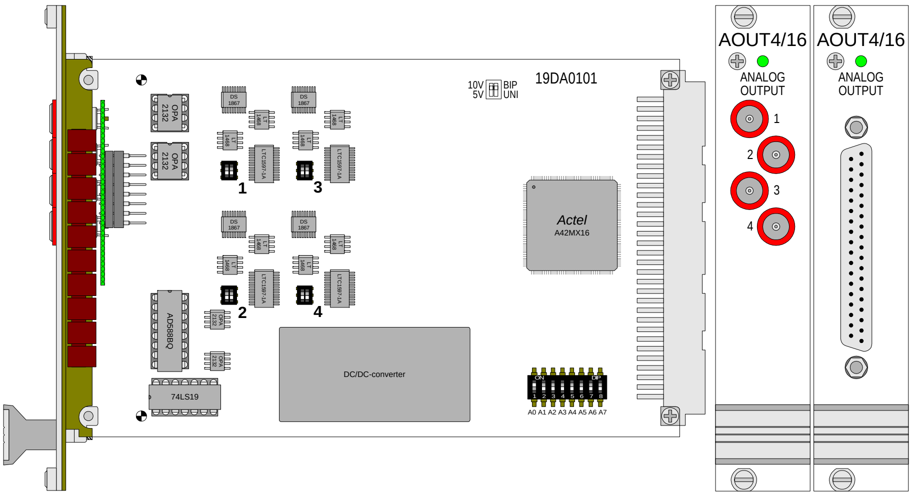


Fig. 120 – Pro-AOut-4/16 REVB: Board and front panel

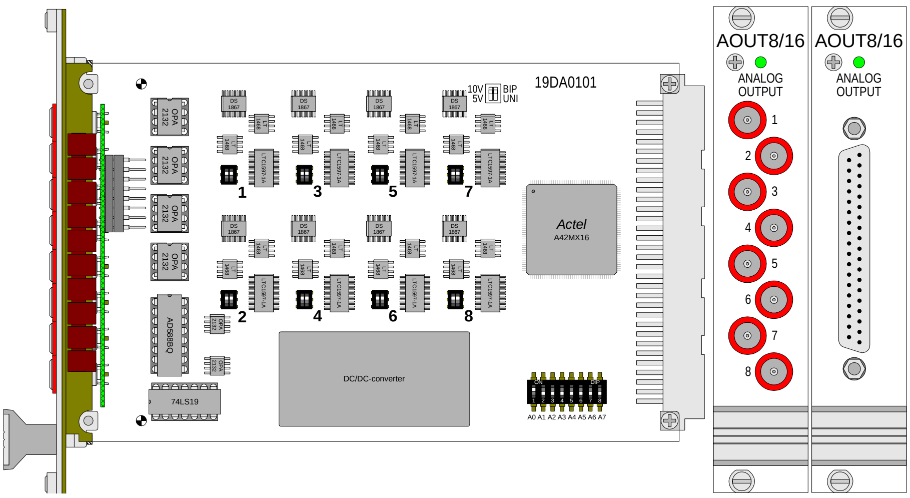


Fig. 121 – Pro-AOut-8/16 REVB: Board and front panel

Figure 122 illustrates all possible DIL switch positions. The "x" in the DIL switch title stands for the number of the corresponding DAC (see fig. 120/121). The adjustment of gain and offset is made by software (see chapter 5.3.1 "Calibration per Software").

If nothing else has been specified upon ordering the module, the voltage range is set to ±10V. Each time you change the DIL switch settings you have to recalibrate the ADC, in order to assure good measurement results.

Voltage range	DILx1	DILx2
±5V bipolar	5V	BIP
±10V bipolar (default)	10V	BIP
0…5V unipolar	5V	UNI
0…10V unipolar	10V	UNI

Fig. 122 – Pro-AOut-8/16 REVB: DIL switch settings for the output voltage range

4.4.4 Pro-AOut-4/16 REVC

The analog output module Pro-AOut-4/16 REVC has four 16-bit DACs and has in its basic version the same functions as the previous version (Rev. B).

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 5.3.1 "Calibration per Software").

The outputs have shielded LEMO sockets (CAMAC European norm). As option a 37-pin D-SUB socket can also be used.

function generator

In the version …-M2 the module has an additional internal memory (SRAM) of 2MB for a function generator. In the memory data any wave forms are stored, which the function generator outputs with a specified output frequency. For each output channel wave form data, output frequency, output start and end can be set individually.

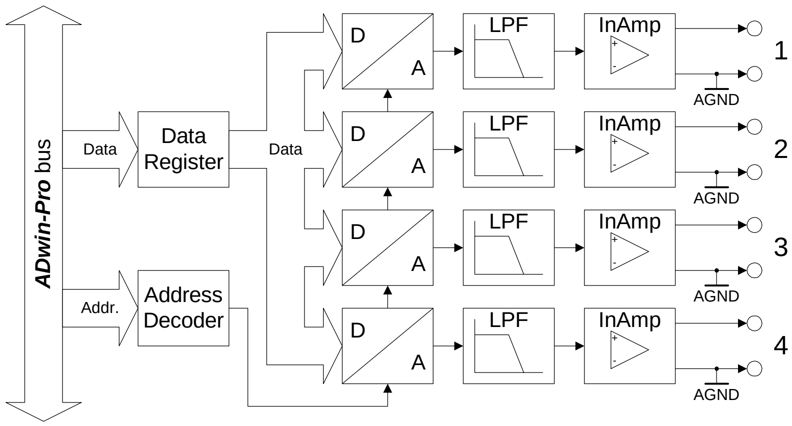


Fig. 123 – Pro-AOut-4/16 REVC: Block diagram

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
Addition memory for the function generator (optional)		2MB
Connectors		4 LEMO sockets optional: 37-pin D-SUB socket

Fig. 124 – Pro-AOut-4/16 REVC: Specification

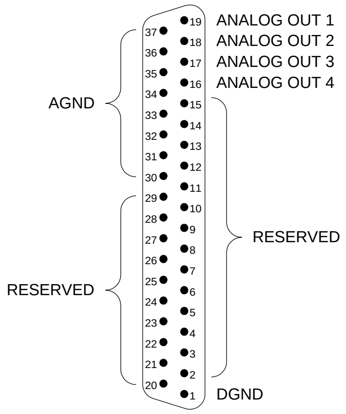


Fig. 125 – Pro-AOut-4/16-D REVC: Pin assignment

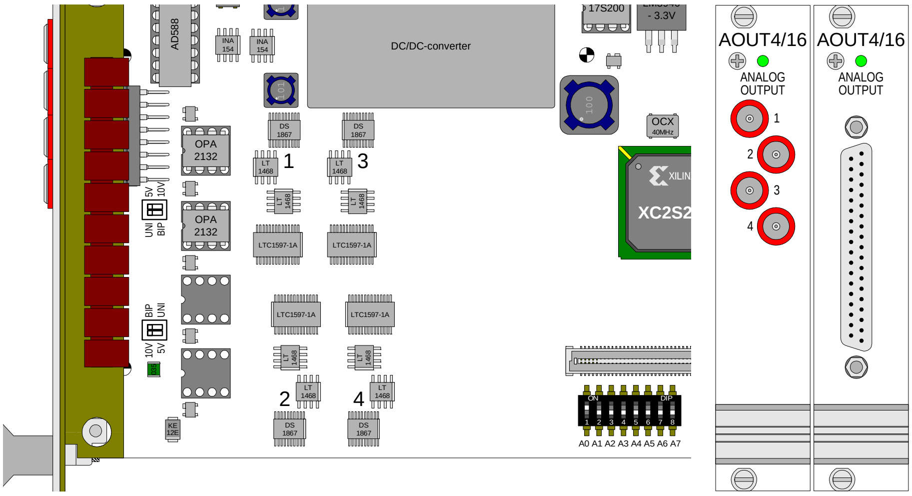


Fig. 126 – Pro-AOut-4/16 REVC: Printed circuit board (detail) and front panels

4.4.5 Pro-AOut-8/16 REVC

The analog output module Pro-AOut-8/16 REVC has eight 16-bit DACs and the same functions as the previous module (Rev. B).

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 5.3.1 "Calibration per Software").

The outputs have shielded LEMO sockets (CAMAC European norm). As option a 37-pin D-SUB socket can also be used.

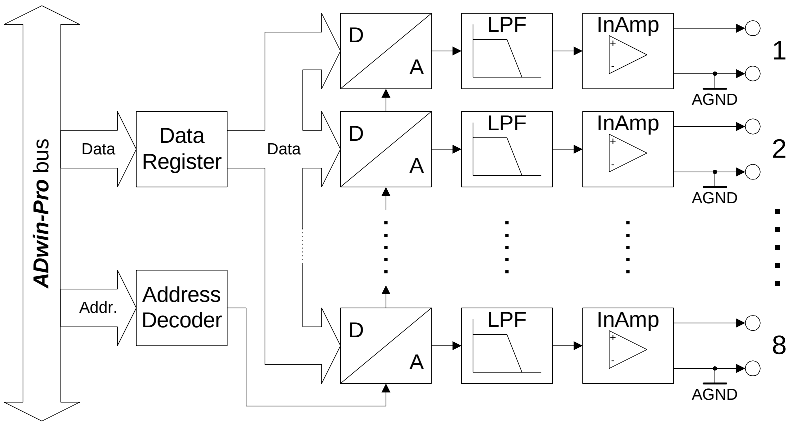


Fig. 127 – Pro-AOut-8/16 REVC: Block diagram

Output channels		8 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
Connectors		8 LEMO sockets optional: 37-pin D-SUB socket

Fig. 128 – Pro-AOut-8/16 REVC: Specification

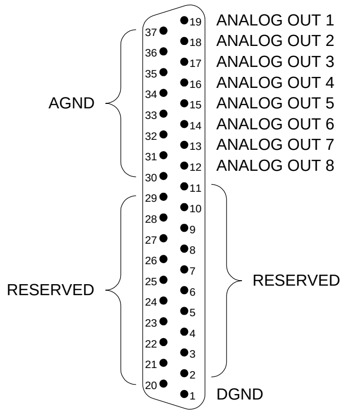


Fig. 129 – Pro-AOut-8/16-D REVC: Pin assignment

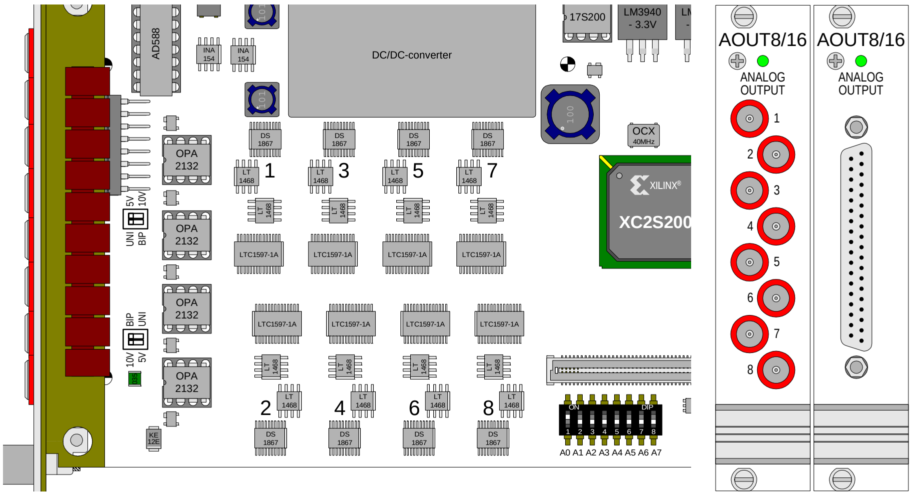


Fig. 130 – Pro-AOut-8/16 REVC: Printed circuit board(detail) and front panels

4.5 Analog Input and Output Modules

4.5.1 Pro-AO-16/8-12 REVA

The analog input/output module Pro-AO-16/8-12 REVA includes a ADC (12 bit) with 16 multiplexed channels and 8 DAC (12 bit). The outputs are equipped with a fixed 1st order low-pass filter (f_c=100 kHz) to cut off glitches.

The 16 inputs and 8 outputs are connected with a 37-pin DSUB connector.

The voltage ranges of the ADC and DAC can be set by 3 jumpers and adjusted by potentiometers (see page 68).

This module has an address in the group of the analog input modules as well as in the group of the analog output modules.

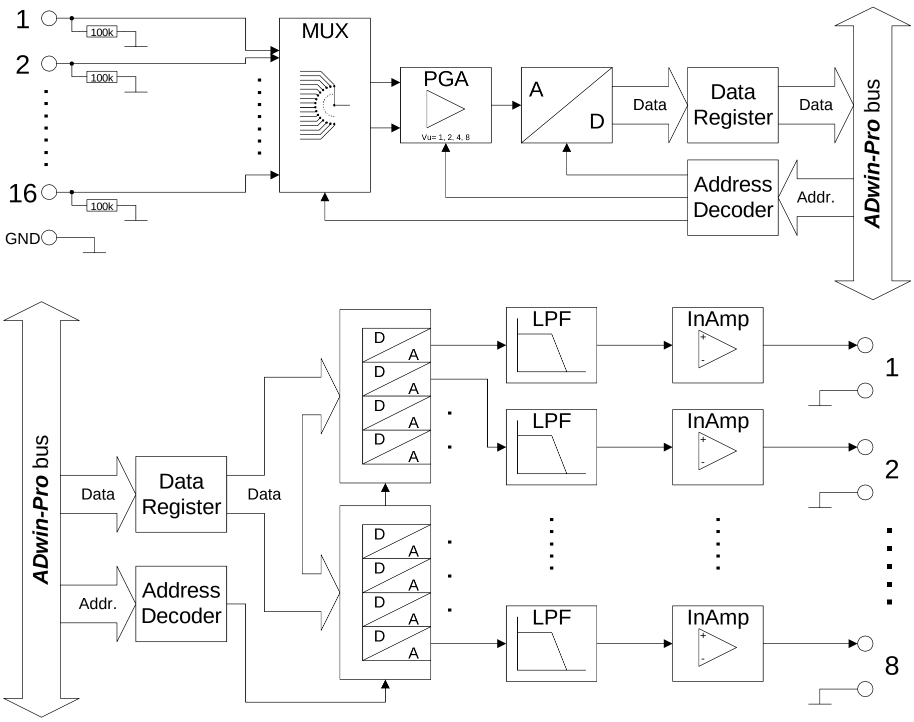


Fig. 131 – Pro-AO-16/8-12 REVA: Block diagram

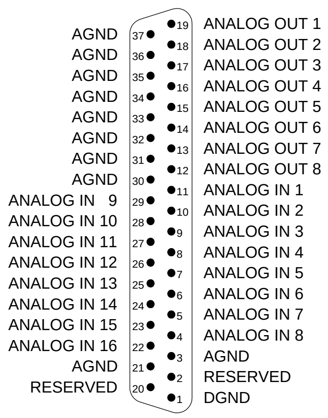


Fig. 132 – Pro-AO-16/8-12 REVA: Pin assignment

ADC		
Input channels		16 single-ended via multiplexer
Resolution		12 bit
Conversion time		7.5µs
Measurement ranges		0…10V, ±5V, ±10V; optional 0…20mA
Gain		1, 2, 4, 8 software selectable
Accuracy	INL	max. ±1 LSB
	DNL	max. ±1 LSB
Input resistance		100kΩ, ±2%
Input over-voltage		±35V
Offset error		adjustable
Offset drift		±30 ppm/ºC of full scale range
DAC		
Output channels		8 single-ended
Resolution		12 bit
Settling time to 0.01%		10µs for a full scale step
Output voltage		0…10V, ±5V, ±10V
Output current max.		5mA per channel
Accuracy	INL	max. ±1 LSB
	DNL	max. ±1 LSB
Offset error		adjustable
Gain error		adjustable
Offset drift		±10µv/ºC
Common		
Connector		37-pin DSub socket

Fig. 133 – Pro-AO-16/8-12 REVA: Specification

Setting the Input and Output Voltage Range

The 12 bit input/output module Pro-AO-16/8-12 REVA is equipped with 1 ADC and 8 DAC. The input voltage range of the ADC and the output voltage range of the DAC can be set by 3 jumpers each. The default settings of the ADC as well as of the DAC is the voltage range ±10V.

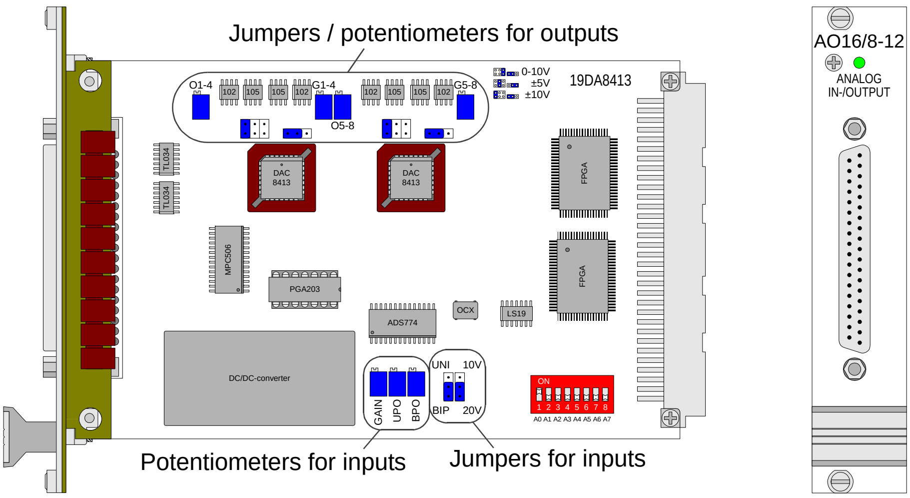


Fig. 134 – Pro-AO-16/8-12 REVA: Board and front panel

Input voltage range

In figure 135 all (allowed) possibilities to adjust the jumpers for setting the input voltage range are listed.

The potentiometers UPO, B and G are used for an accurate adjustment of gain and offset (fig. 136). The jumpers as well as the potentiometers for setting the input voltage range can be found on the lower part of the module.

If nothing else has been said on ordering the module, the voltage range is set to ±10V. After every jumper setting you have to recalibrate the ADC, in order to assure correct measurement results. The individual steps are described in chapter 5 "Calibration".

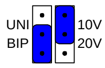
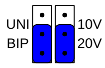
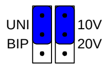
Voltage range	Jumper settings
±5V bipolar	
±10V (default)	
0…10V	

Fig. 135 – Pro-AO-16/8-12 REVA: Jumper settings for the input voltage range

Potentiometer	Adjustment of
Gain	Gain factor
BPO	Offset (bipolar setting)
UPO	Offset (unipolar setting)

Fig. 136 – Pro-AO-16/8-12 REVA: Function of the potentiometers for the inputs

Output voltage range

In fig. 137 all possible jumper settings for the output voltage range are listed.

The potentiometers `O1-4` for the outputs 1…4 and `O5-8` for the outputs 5…8 or respectively, `G1-4` and `G5-8` (fig. 138) are used for an accurate adjustment of gain and offset.

The jumpers as well as the potentiometers for setting the output voltage range can be found on the lower part of the module.

If nothing else has been said on ordering the module, the voltage range is set to ±10V. After every jumper setting you have to recalibrate the DAC, in order to assure correct measurement results. The individual steps are described in chapter 5.3.1 "Calibration per Software".

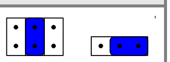
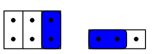
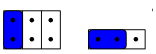
Voltage range	Jumper settings
±5V bipolar	
±10V (default)	
0…10V	

Fig. 137 – Pro-AO-16/8-12 REVA: Jumper settings for the output voltage range

Potentiometer	Adjustment of
G1-4 G5-8	Gain factor
O1-4 O5-8	Offset

Fig. 138 – Pro-AO-16/8-12 REVA: Function of the potentiometers for the outputs

4.6 Digital-I/O- and Counter Modules

Digital I/O Modules

Module	Rev.	Type	Chan-nels	Input Voltage U_{In} [V]		High Level [mA]	Isolation [V]	Page
DIO-32	A	TTL input / output	32	5	TTL	–	–	72
DIO-32	B	TTL input / output	32	5	TTL	–	–	74
OPT-16	A, B	Optocouple input	16	5, 12, 24	DC	–	500	76
REL-16	A, B	Relay output	16	max. 30	AC / DC	500	500	78
TRA-16	A	Transistor output	16	5...30	DC	200	500	80
TRA-16	B	Transistor output	16	5...30	DC	200	500	
Comp-16	A	comparator input, switching thresholds separately adjustable	16 s.e.	-2 ... +8,23	DC	–	–	119
Storage	A	Module to read / write a transportable mass media (PCMCIA card, Compact Flash card, hard disk) from *ADbasic*. With integrated real time clock.						121

Zähler-Module

Module	Rev.	Chan-nels	Counter			Input voltage. U_{In}		Isolation [V]	Page
			No.	Type[a]	Resol. [Bit]	[V]	Type		
CNT-VR4 CNT-VR4-L	A	4	1	UD	32	5	TTL	–	82
CNT-VR4-I CNT-VR4-L-I	A	4	1	UD	32	5, 12, 24	DC	500	84
CNT-8/32	A	8	1	I	32	5	TTL	–	86
CNT-8/32-I	A	8	1	I	32	5, 12, 24	DC	500	88
CNT-16/16	A	16	1	I	16	5	TTL	–	90
CNT-16/16-I	A	16	1	I	16	5, 12, 24	DC	500	92
CNT-16/32	A	16	1	I	32	5	TTL	–	94
CNT-16/32-I	A	16	1	I	32	5, 12, 24	DC	500	96
CNT-VR2PW2	A	4	2	I, UD	32	5	TTL	–	98
CNT-VR2-PW2-I	A	4	2	I, UD	32	5, 12, 24	DC	500	98
CNT-PW4	A	4	1	PWM	32	5	TTL	–	99
CNT-PW4-I	A	4	1	PWM	32	5, 12, 24	DC	500	101
CO4-T	A	4	1	U	32	5	TTL	–	103
CO4-I	A	4	1	U	32	5, 12, 24	DC	500	105
CO4-D	A	4 + 2 SSI	1	U	32	5 diff.	RS422/ RS485	–	107
PWM-4	A	4	1	PWM	32	5	TTL	–	110
PWM-4-I	A	4	1	PWM	–	5...30	DC	500	112

a. UD: Up / Down counter; I = Incremental counter; PWM: PWM analysis;
U: Universal counter = VR + I + PWM

4.6.1 Pro-DIO-32 REVA

To this module you find an improved successor module Pro-DIO-32 REVB (see page 74).

The digital input/output module Pro-DIO-32 REVA provides 32 programmable digital input and output channels at TTL levels. The channels can individually be configured as inputs or outputs by ***ADbasic*** instructions. The channels are configured as inputs after power up.

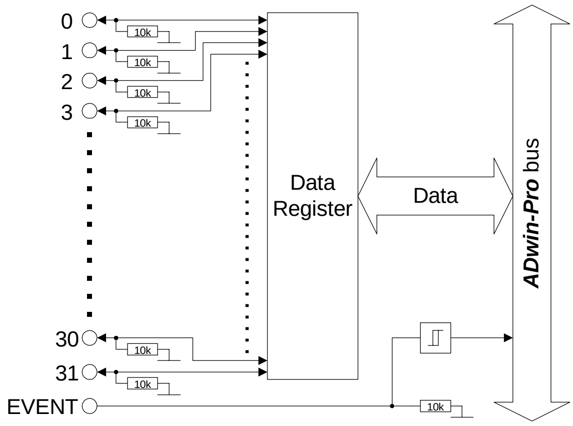


Fig. 139 – Pro-DIO-32 REVA: Block diagram

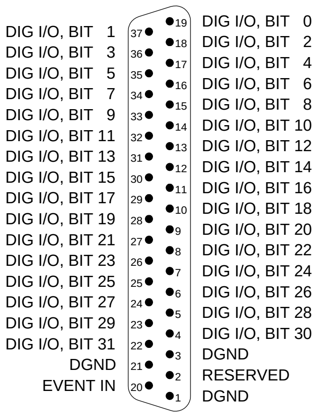


Fig. 140 – Pro-DIO-32 REVA: Pin assignment

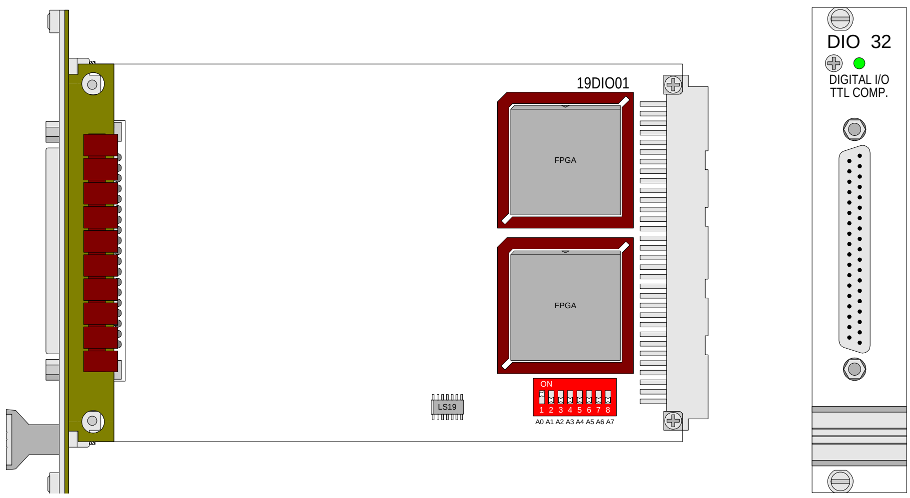


Fig. 141 – Pro-DIO-32 REVA: Board and front panel

Input / output channels	32, each configurable via software as input or output
Digital inputs	TTL logic
Pull-Down Resistor	10 kΩ
V_{IH}	min. 2.4 V
V_{IL}	max. 0.8 V
I_{IH}	max. 0.55 mA
I_{IL}	max. 0.01 mA
Voltage range	-0.5 V … +5.5 V
Output current	max. 6 mA per channel (outputs are short-circuit proof)
Event input	TTL logic
Power up status	All channels as inputs
Connector	37-pin DSub socket

Fig. 142 – Pro-DIO-32 REVA: Specification

4.6.2 Pro-DIO-32 REVB

The digital input/output module Pro-DIO-32 REVB 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 (not individually as previous model). The channels are configured as inputs after power up.

With the new command `DIGOUT_F` the setting or resetting of single outputs is realized quicker and needs much less program memory (the previous command `DIGOUT` can still be used).

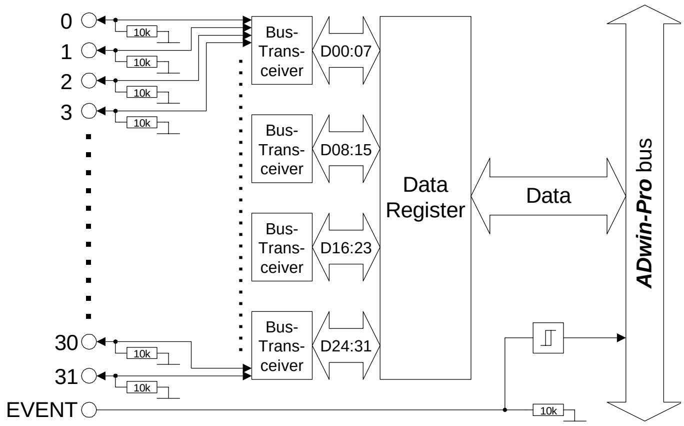


Fig. 143 – Pro-DIO-32 REVB: Block diagram

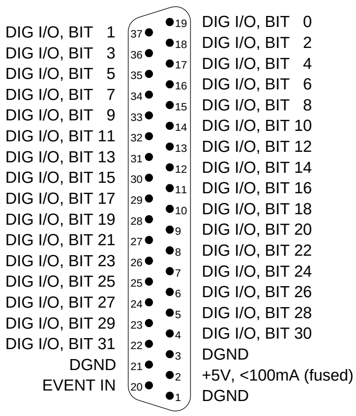


Fig. 144 – Pro-DIO-32 REVB: Pin assignment

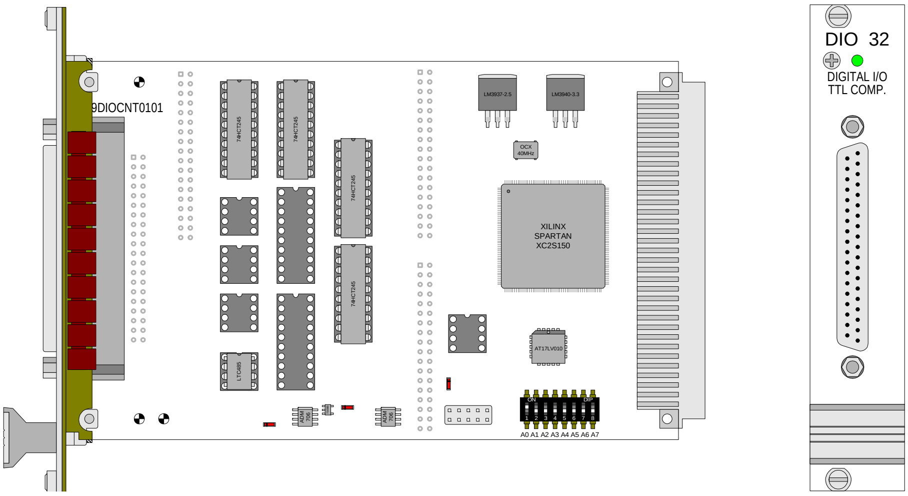


Fig. 145 – Pro-DIO-32 REVB: Board and front panel

Input/output channels	32; programmable via software as inputs/outputs in blocks of 8
Digital inputs	TTL logic
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
Connector	37-pin DSub socket

Fig. 146 – Pro-DIO-32 REVB: Specification

4.6.3 Pro-OPT-16 REVA

The input module Pro-OPT-16 REVA 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 200ns 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.

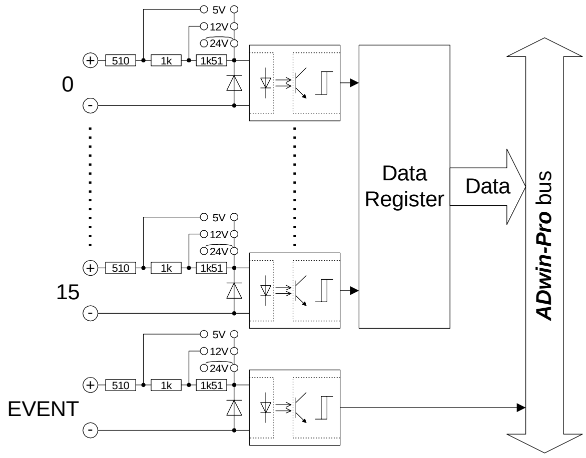


Fig. 147 – Pro-OPT-16 REVA: Block diagram

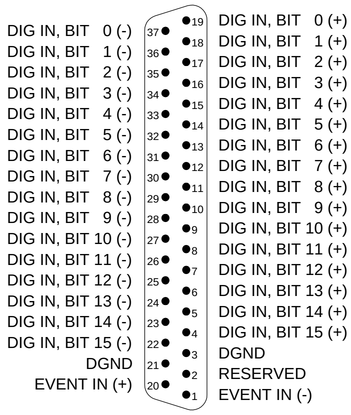


Fig. 148 – Pro-OPT-16 REVA: Pin assignment

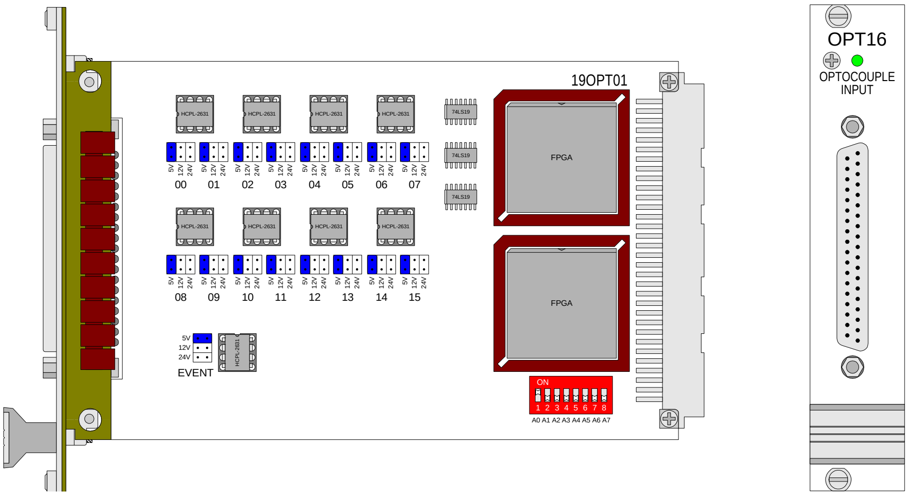


Fig. 149 – Pro-OPT-16 REVA: Board and front panel

Input channels	16		
Event inputs	1		
Input current	typ. 7 mA / max. 15 mA		
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	200 ns		
Isolation	500V channel-channel / channel-GND		
Connector	37-pin DSub socket		

Fig. 150 – Pro-OPT-16 REVA: Specification

4.6.4 Pro-REL-16 REVA, REVB

The Pro-REL-16 REVA / REVB 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.

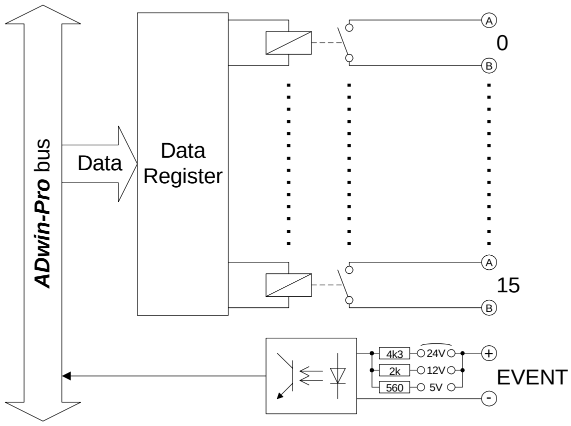


Fig. 151 – Pro-REL-16 REVA, REVB: Block diagram

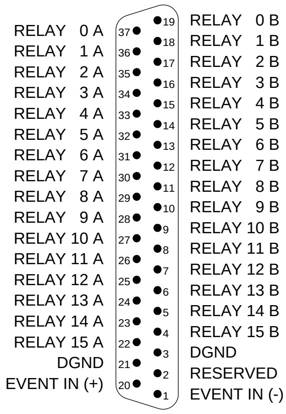


Fig. 152 – Pro-REL-16 REVA, REVB: Pin assignment

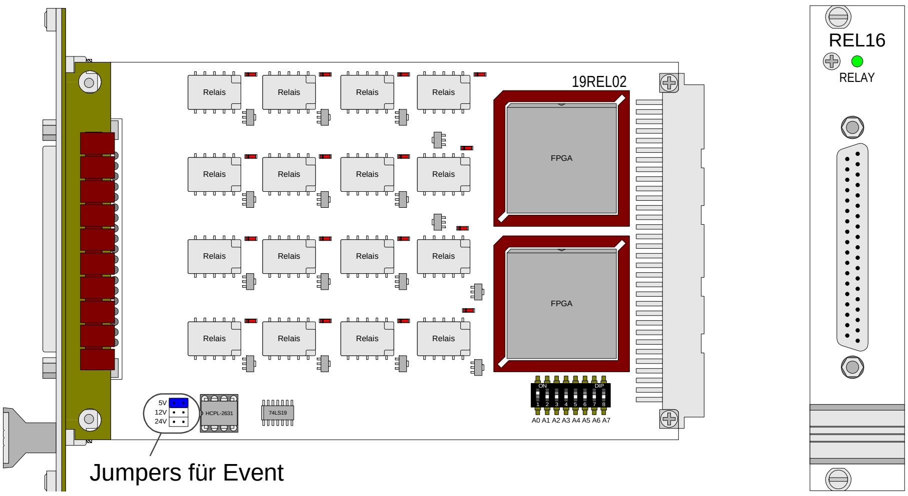


Fig. 153 – Pro-REL-16 REVA, REVB: Board and front panel

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	500V 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 DSub socket

Fig. 154 – Pro-REL-16 REVA, REVB: Specification

4.6.5 Pro-TRA-16 REVA, REVB

The output module Pro-TRA-16 REVA / REVB provides 16 channels of isolated transistor outputs. The supply voltage V_{CC} has to be provided by an external power supply. The channels are isolated from system circuitry. The event-input is optically isolated from system circuitry.

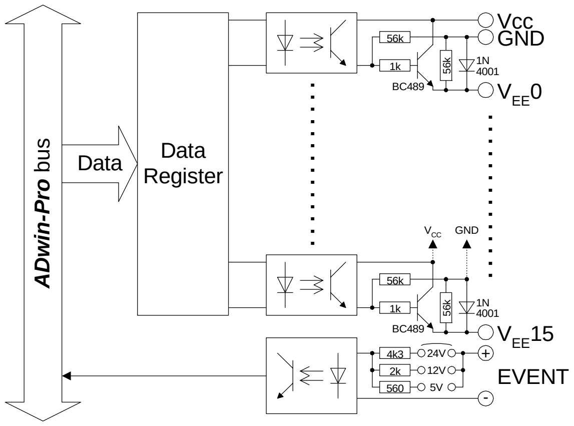


Fig. 155 – Pro-TRA-16 REVA: Block diagram

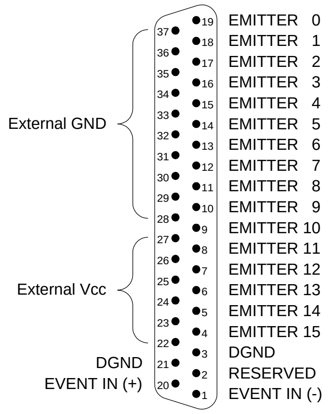


Fig. 156 – Pro-TRA-16 REVA: Pin assignment

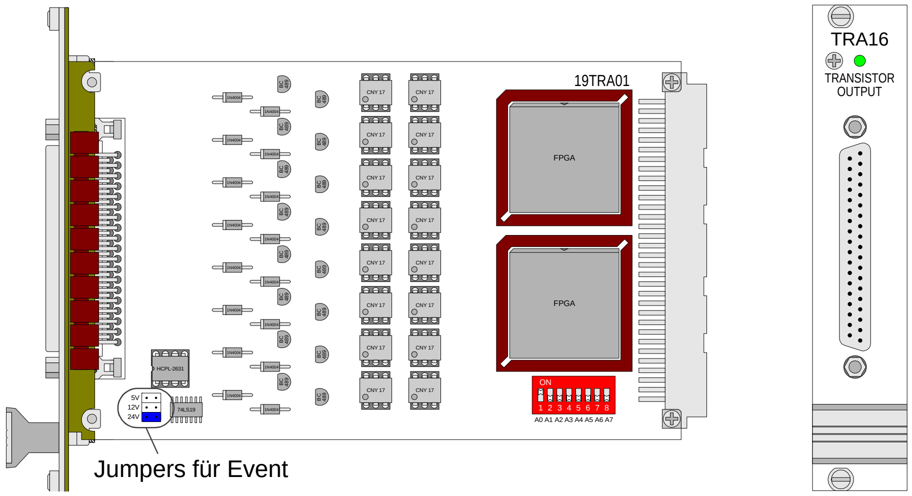


Fig. 157 – Pro-TRA-16 REVA: Board and front panel

Output channels	16
Switching voltage	5…30V DC with external power supply
Switching current	200mA max. per channel
Voltage drop	0.5V
Switching time	10µs
Event input	1
Isolation	500V channel – channel / channel – GND
Event input voltage	5V, 12V, 24V (selectable via jumpers)
Power up status	low (GND external)
Connector	37-pin DSub socket

Fig. 158 – Pro-TRA-16 REVA: Specification

4.6.6 Pro-CNT-VR4 (-L) REVA

To this module you find an improved successor module Pro-CO4-T REVA (see page 103).

The Pro-CNT-VR4 REVA counter module has 4 up/down counters (32 bit), an edge evaluation circuit, and a register (latch) for read out during the count process. All counter values can be loaded (latched) into the register simultaneously with instruction `CNT_LATCH`. It is also possible to latch the counters individually.

Each counter has 2 inputs which are decoded internally by an edge evaluation logic (quadruple evaluation). The maximum frequency is 1.25 MHz at each input A and B (maximum internal count rate: 5 MHz). Optionally the counters can be used with one clock input and a direction input at a maximum count rate of 10 MHz. The operating mode is selectable by software, for each counter individually.
According to the mode of operation, either the inputs A/B are active or the inputs CLK/DIR.

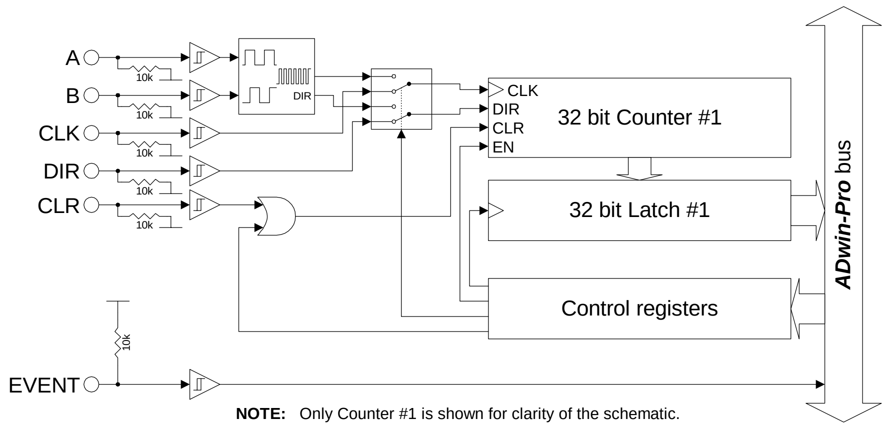


Fig. 159 – Pro-CNT-VR4 REVA: Block diagram

The module Pro-CNT-VR4 REVA is also available as Pro-CNT-VR4-L REVA version. With this version each counter is equipped with a LATCH-input instead of a CLR-input. The LATCH-inputs must be enabled before use with the instruction `EXTLCH_ENABLE`.
(see also example program `<Pro-CNT-VR4-L-I.BAS>`).

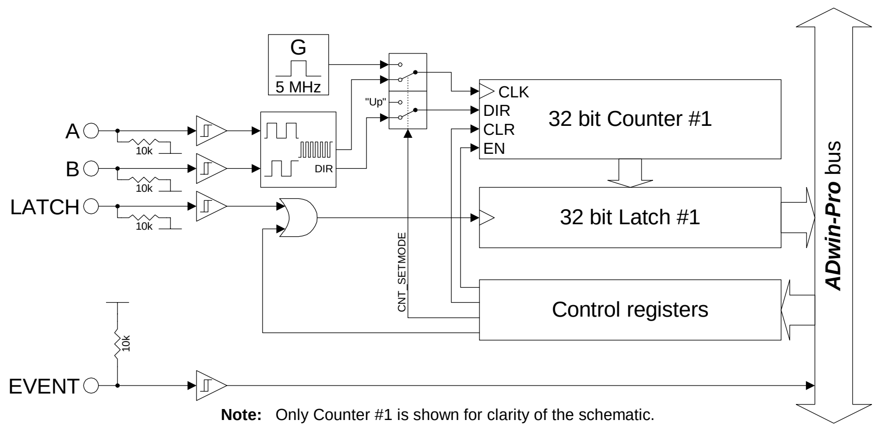


Fig. 160 – Pro-CNT-VR4-L REVA: Block diagram

The modules Pro-CNT-VR4 REVA and Pro-CNT-VR4-L REVA are equipped with 4 times the components shown in the block diagram; exception: the event input and the control register, which can only be found once on the modules.

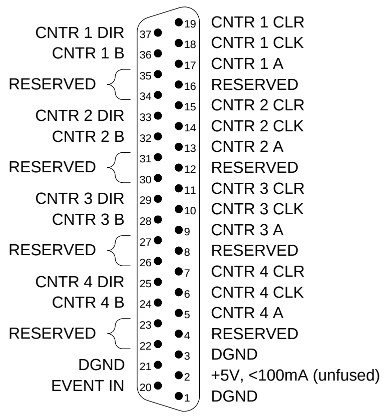


Fig. 161 – Pro-CNT-VR4 REVA: Pin assignment

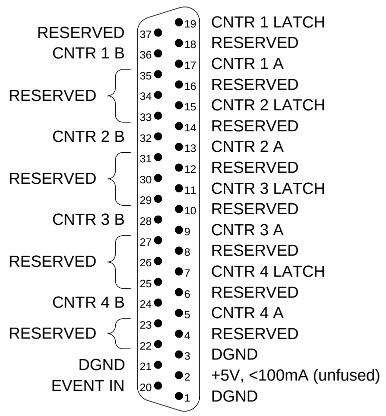


Fig. 162 – Pro-CNT-VR4-L REVA: Pin assignment

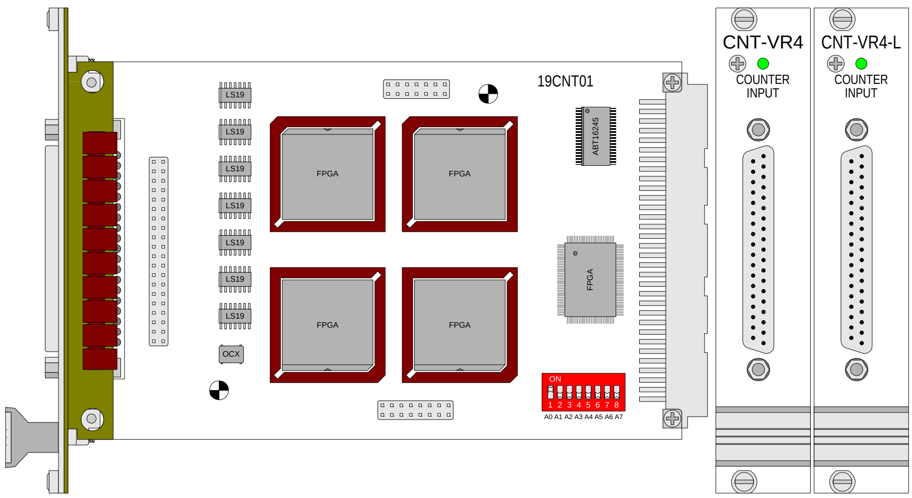


Fig. 163 – Pro-CNT-VR4(-L) REVA: Board and front panel

Counter		4 up/down counters
Counter resolution		32 bit
Input clock rate	edge evaluation	1.25MHz max. per channel A,B
	clock, direction	10MHz max.
Signal pulse width	edge evaluation	min. 800ns per channel A,B
	clock, direction	min. 50ns
Inputs		TTL
Trigger Input		pos. TTL
Pull down resistor		10kΩ
V_{IH}		min. 2.4V
V_{IL}		max. 0.8V
I_{IH}		max. 0.55mA
I_{IL}		max. 0.01mA
Voltage range, absolute		-0.3V … 7V
Connector		37-pin DSub socket
Isolation		No (see page 84)

Fig. 164 – Pro-CNT-VR4 REVA: Specification

4.6.7 Pro-CNT-VR4(-L)-I REVA

To this module you find an improved successor module Pro-CO4-I REVA (see page 105).

The counter module Pro-CNT-VR4(-L)-I REVA has 4 up/down counters (32 bits), an edge evaluation circuit, and a register (latch) for read out during the count process. All count rates can be loaded (latched) simultaneously into the register with the instruction `CNT_LATCH`. It is also possible to latch the counters individually.
Each counter has 2 inputs which are decoded internally by an edge evaluation logic (quadruple evaluation). The maximum throughput rate is 1.25MHz at each input A and B (maximum internal count rate: 5MHz). Optionally the counters can be used with a clock input (CLK) and a direction input (DIR) at a maximum count rate of 10MHz. The operating mode is selectable per software, for each counter individually.
According to the mode of operation, either the inputs A/B are active or the inputs CLK/DIR.

The voltage range of the counter and event inputs can be selected by jumpers. The default setting of the input voltage range is 24V. The counter inputs are optically isolated from the system circuitry as well as from other inputs. The event input is also isolated from the system circuitry.

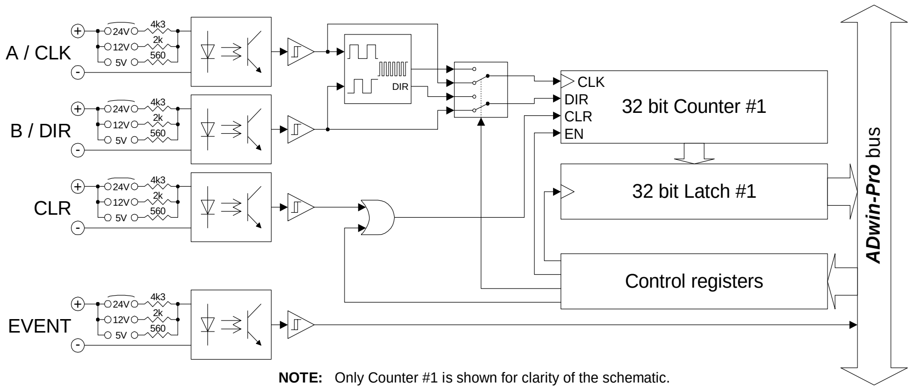


Fig. 165 – Pro-CNT-VR4-I REVA: Block diagram

On the module version Pro-CNT-VR4-L-I REVA each counter is equipped with a LATCH-input instead of a CLR-input. The LATCH-inputs must be enabled before use with the instruction `EXTLCH_ENABLE` (see also example program `<Pro-CNT-VR4-L-I.BAS>`).

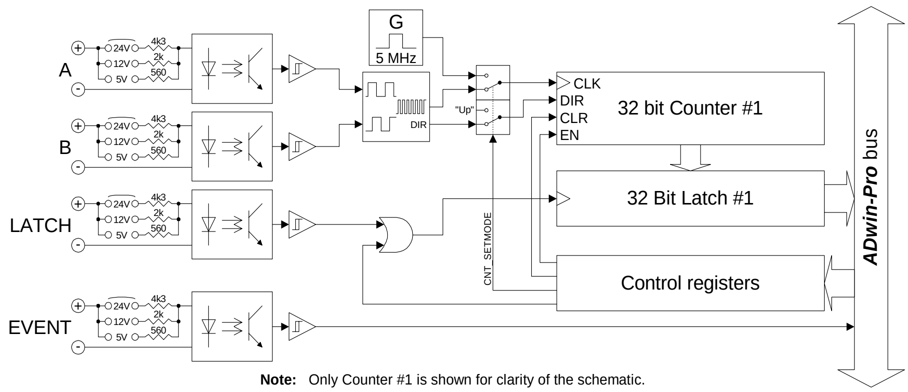


Fig. 166 – Pro-CNT-VR4-L-I REVA: Block diagram

The modules Pro-CNT-VR4-I and Pro-CNT-VR4-L-I are equipped with 4 times the components shown in the block diagram; exception: the event input and the control register, which can only be found once on the modules.

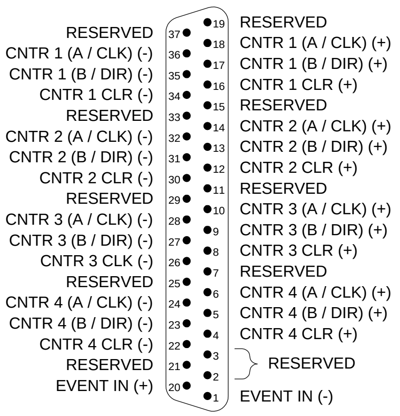


Fig. 167 – Pro-CNT-VR4-I REVA: Pin assignment

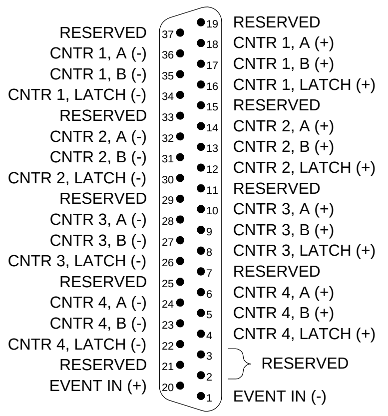


Fig. 168 – Pro-CNT-VR4-L-I REVA: Pin assignment

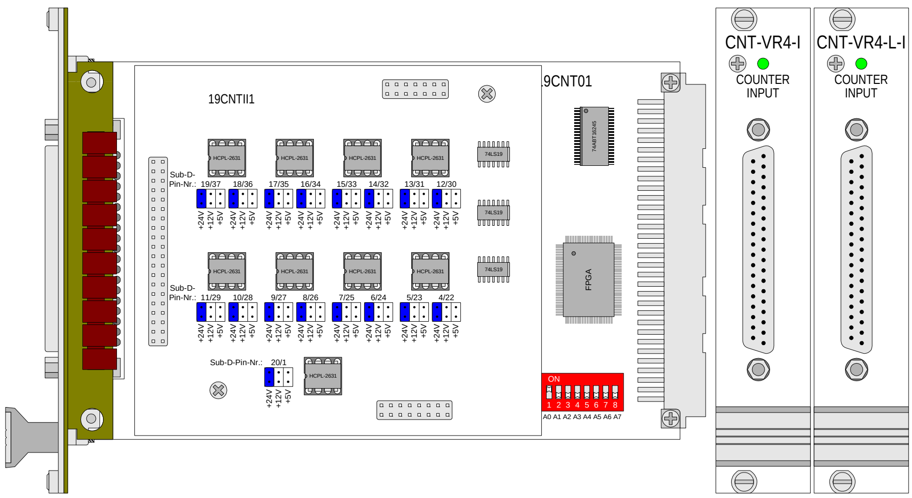


Fig. 169 – Pro-CNT-VR4(-L)-I REVA: Board and front panel

Counter		4 up/down counters		
Counter resolution		32 bit		
Input clock rate	edge evaluation	1.25MHz max. per channel A,B		
	clock, direction	10MHz max.		
Signal pulse width	edge evaluation	min. 800ns per channel A,B		
	clock, direction	min. 50ns		
Event inputs		1		
Input current		typ. 7mA / max. 15mA		
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 resistance		560 Ω	2 kΩ	4.3 kΩ
Input over-voltage		-5V … 8V	-5V … 16V	-5V … 30V
Switching time		200ns		
Connector		37-pin DSub socket		
Isolation		500V channel-channel / channel-GND		

Fig. 170 – Pro-CNT-VR4(-L)-I REVA: Specification

4.6.8 Pro-CNT-8/32 REVA

To this module you find an improved successor module Pro-CNT-16/32 REVA (see page 94).

The counter module Pro-CNT-8/32 REVA has 8 counters (32 bit). All count rates can be loaded into the register with a single ***ADbasic*** command so that all counter values can be latched simultaneously. It is also possible to latch the counters individually. With a rising edge of a TTL pulse the 32-bit counter increments by one. The counter can be cleared by software command. The count rate can be determined by the difference between two successive register values. Since the register access is discrete-time, the frequency can be calculated online.

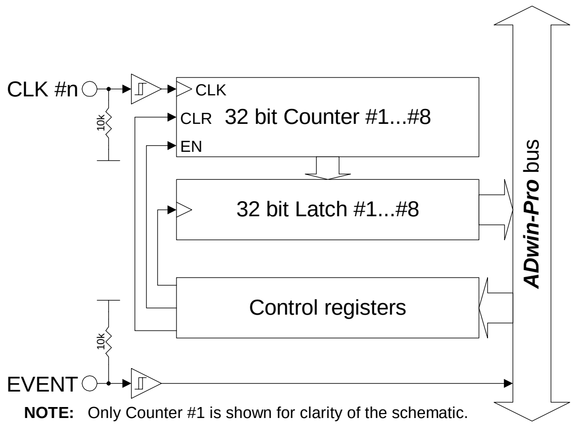


Fig. 171 – Pro-CNT-8/32 REVA: Block diagram

The module Pro-CNT-8/32 REVA is equipped with 8 times the components shown in the block diagram; exception: the event input and the control register, which can only be found once on the module.

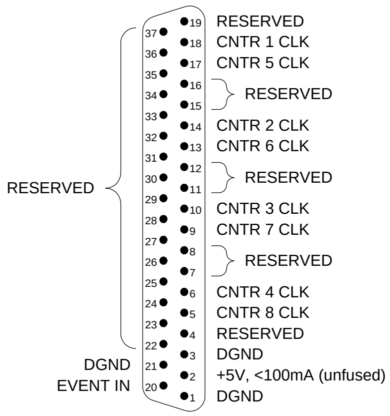


Fig. 172 – Pro-CNT-8/32 REVA: Pin assignment

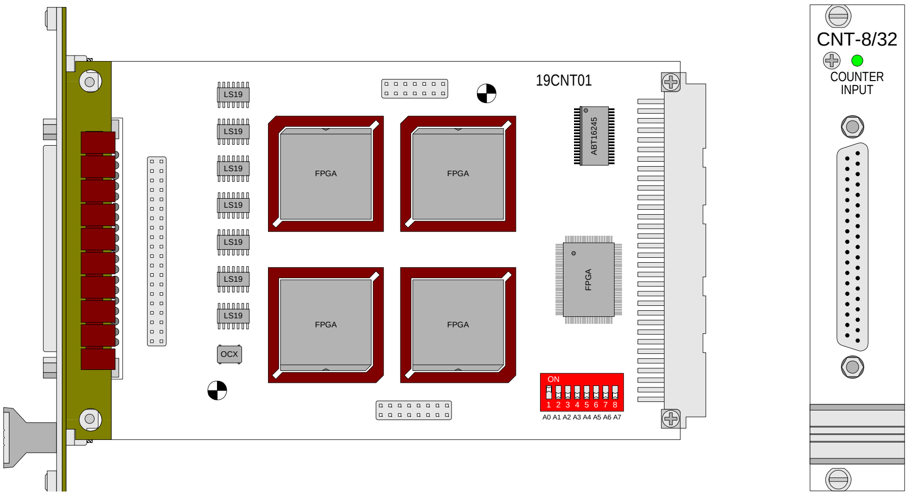


Fig. 173 – Pro-CNT-8/32 REVA: Board and front panel

Counter	8 up counters
Counter resolution	32 bit
Input clock rate	10 MHz max.
Signal pulse width	min. 50 ns
Inputs	TTL
Trigger Input	pos. TTL
Pull down resistor	10 kΩ
V_{IH}	min. 2.4 V
V_{IL}	max. 0.8 V
I_{IH}	max. 0.55 mA
I_{IL}	max. 0.01 mA
Voltage range, absolute	-0.3 V … 7 V
Connector	37-pin DSub socket
Isolation	No (see page 88)

Fig. 174 – Pro-CNT-8/32 REVA: Specification

4.6.9 Pro-CNT-8/32-I REVA

To this module you find an improved successor module Pro-CNT-16/32-I REVA (see page 96).

The counter module Pro-CNT-8/32-I REVA has 8 counters (32 bit). All count rates can be loaded into the register with a single ***ADbasic*** command so that all counter values can be latched simultaneously. It is also possible to latch the counters individually. With a rising edge of a TTL pulse the 32-bit counter increments by one. The counter can be cleared by software command. The count rate can be determined by the difference between two successive register values. Since the register access is discrete-time, the frequency can be calculated online.

The counter inputs are optically isolated from the system circuitry as well as from other inputs. The event input is also isolated from the system circuitry. The input voltage range can be selected by three jumpers. The default setting of the input voltage range is 24V.

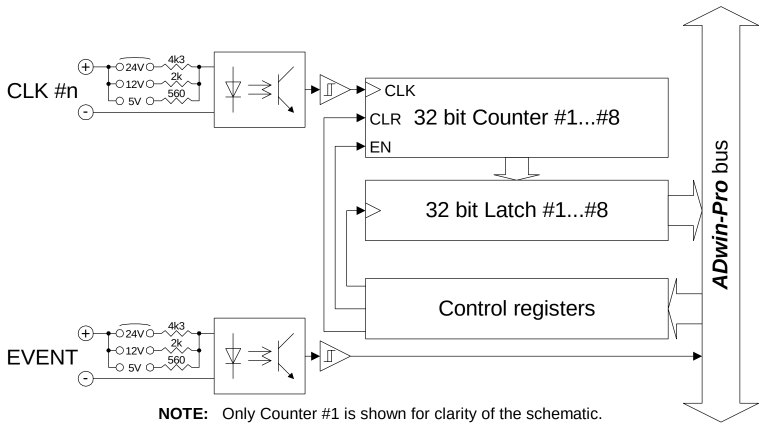


Fig. 175 – Pro-CNT-8/32-I REVA: Block diagram

The module Pro-CNT-8/32-I is equipped with 8 times the components shown in the block diagram; exception: the event input and the control register.

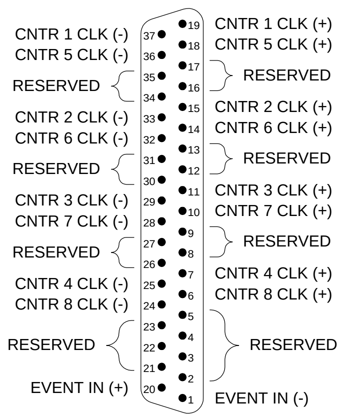


Fig. 176 – Pro-CNT-8/32-I REVA: Pin assignment

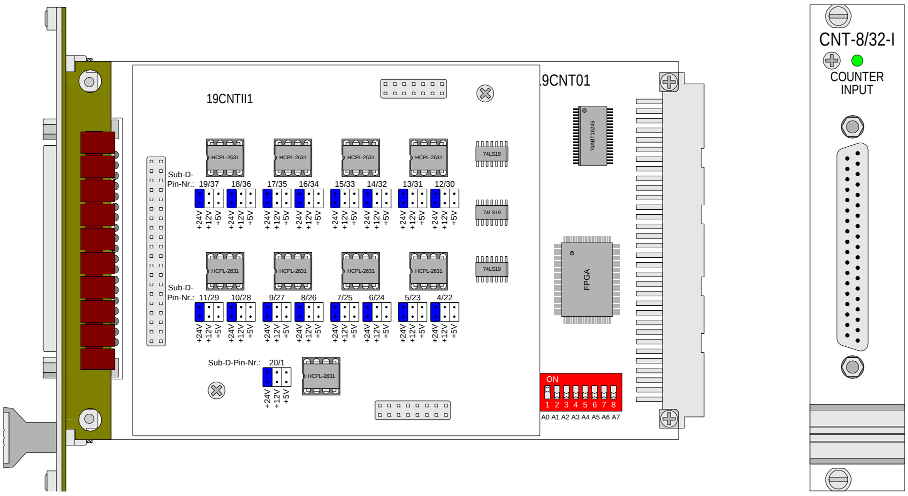


Fig. 177 – Pro-CNT-8/32-I REVA: Board and front panel

Counter	8 up counters		
Counter resolution	32 bit		
Event input	1		
Input current	typ. 7mA / max. 15mA		
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 resistance	560 Ω	2 kΩ	4.3 kΩ
Input over-voltage	-5V … 8V	-5V … 16V	-5V … 30V
Switching time	200ns		
Connector	37-pin DSub socket		
Isolation	500V channel-channel / channel-GND		

Fig. 178 – Pro-CNT-8/32-I REVA: Specification

4.6.10 Pro-CNT-16/16 REVA

To this module you find an improved successor module Pro-CNT-16/32 REVA (see page 94).

The counter module Pro-CNT-16/16 REVA has 16 counters (16 bit). All count rates can be loaded into the register with a single ***ADbasic*** command so that all counter values can be latched simultaneously. It is also possible to latch the counters individually. With a rising edge of a TTL pulse the 16-bit counter increments by one The counter can be cleared by software command.

The count rate can be derived from the difference of two successive readings of the latch.

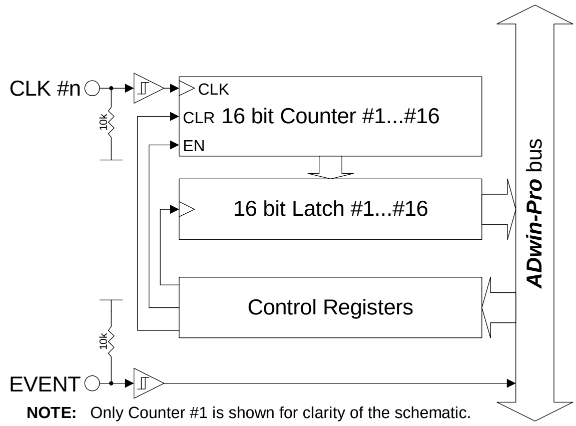


Fig. 179 – Pro-CNT-16/16 REVA: Block diagram

The module Pro-CNT-16/16 REVA is equipped with 16 times the components shown in the block diagram; exception: the event input and the control register.

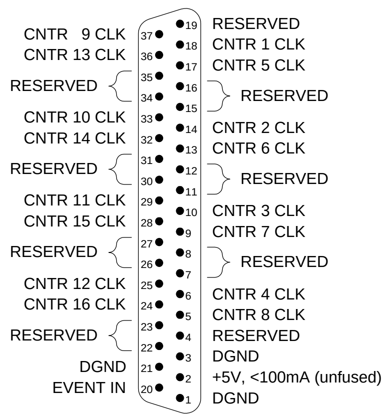


Fig. 180 – Pro-CNT-16/16 REVA: Pin assignment

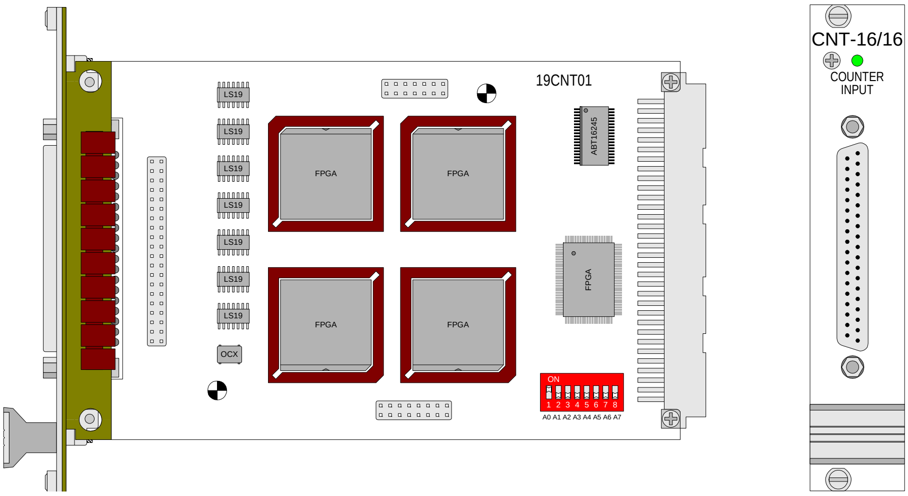


Fig. 181 – Pro-CNT-16/16 REVA: Board and front panel

Counter	16 up counters
Counter resolution	16 bit
Input clock rate	10MHz max.
Signal pulse width	min. 50ns
Inputs	TTL
Trigger Input	pos. TTL
Pull down resistor	10kΩ
V_{IH}	min. 2.4V
V_{IL}	max. 0.8V
I_{IH}	max. 1mA
I_{IL}	max. 0.2mA
Voltage range, absolute	-0.3V … 7V
Connector	37-pin DSub socket
Isolation	No (see page 92)

Fig. 182 – Pro-CNT-16/16 REVA: Specification

4.6.11 Pro-CNT-16/16-I REVA

To this module you find an improved successor module Pro-CNT-16/32-I REVA (see page 96).

The counter module Pro-CNT-16/16-I REVA has 16 counters (16 bit). All counter values can be loaded into the register with a single ***ADbasic*** command so that all counter values can be latched simultaneously. It is also possible to latch the counters individually. With a rising edge of a TTL pulse the 16-bit counter increments by one. The counter can be cleared by software command. The count rate can be derived by the difference between two successive readings of the latch. Since the register access is discrete-time, the frequency can be calculated online.

The counter inputs are optically isolated from the system circuitry as well as from other inputs. The event input is also isolated from the system circuitry. The input voltage range can be selected by jumpers. The default setting of the input voltage range is 24V.

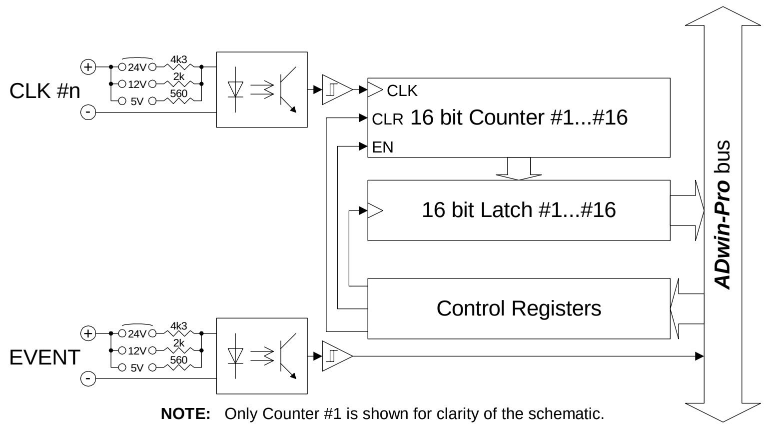


Fig. 183 – Pro-CNT-16/16-I REVA: Block diagram

The module Pro-CNT-16/16-I REVA is equipped with 16 times the components shown in the block diagram; exception: the event input and the control register.

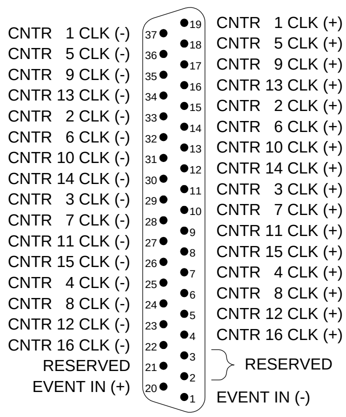


Fig. 184 – Pro-CNT-16/16-I REVA: Pin assignment

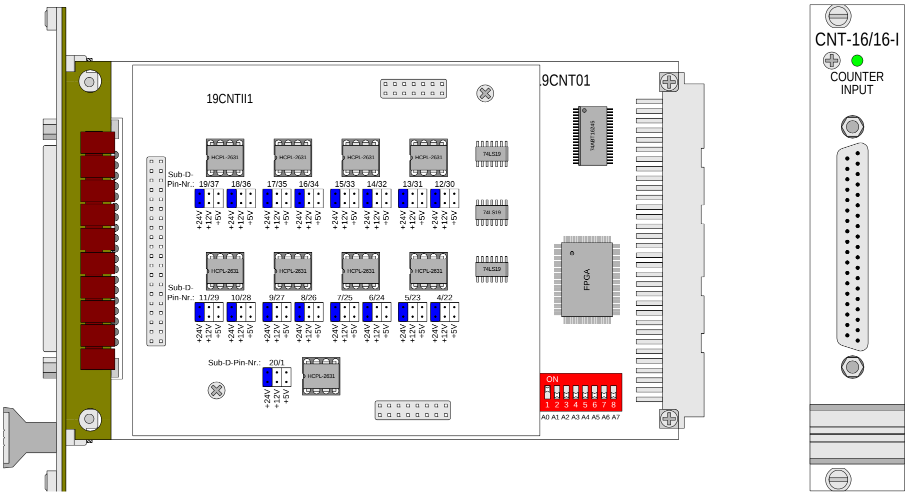


Fig. 185 – Pro-CNT-16/16-I REVA: Board and front panel

Counter	16 up counters		
Counter resolution	16 bit		
Event input	1		
Input current	typ. 7mA / max. 15mA		
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 resistance	560 Ω	2 kΩ	4.3 kΩ
Input over-voltage	-5V … 8V	-5V … 16V	-5V … 30V
Switching time	200ns		
Connector	37-pin DSub socket		
Isolation	500V channel-channel / channel-GND		

Fig. 186 – Pro-CNT-16/16-I REVA: Specification

4.6.12 Pro-CNT-16/32 REVA

The counter module Pro-CNT-16/32 REVA has 16 counters (32 bit). All counter values can be loaded into the register with a single ***ADbasic*** command so that all counter values can be latched simultaneously. It is also possible to latch the counters individually. With a rising edge of a TTL pulse the 32-bit counter increments by one. The counter can be cleared by software command. The count rate can be derived by the difference between two successive readings of the latch. Since the register access is discrete-time, the frequency can be calculated online.

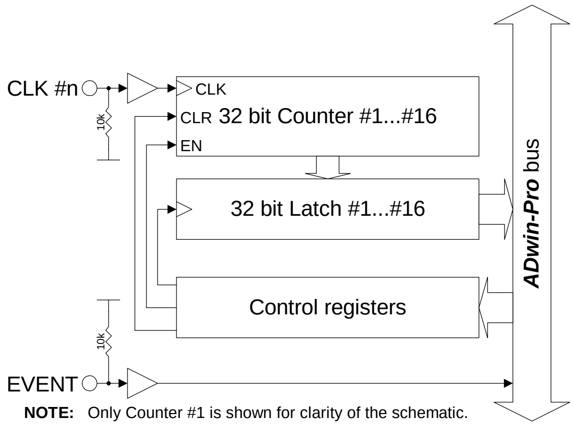


Fig. 187 – Pro-CNT-16/32 REVA: Block diagram

The module Pro-CNT-16/32 REVA is equipped with 16 times the components shown in the block diagram, exception: the event input and the control register.

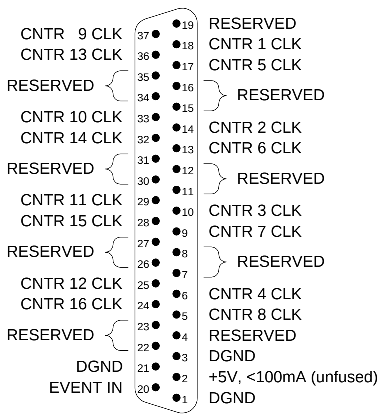


Fig. 188 – Pro-CNT-16/32 REVA: Pin assignment

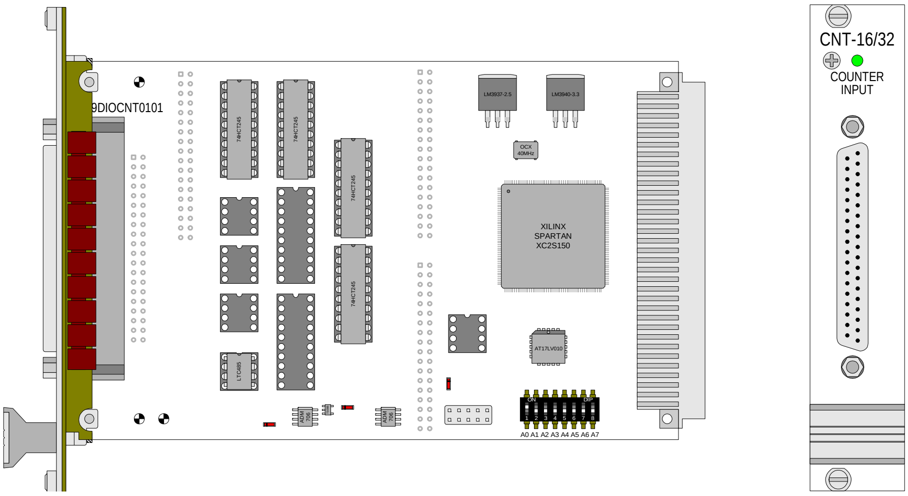


Fig. 189 – Pro-CNT-16/32 REVA: Board and front panel

Counter	16 up counters
Counter resolution	32 bit
Input clock rate	20MHz max.
Signal pulse width	min. 25ns
Inputs	TTL
Trigger Input	pos. TTL
Pull down resistor	10kΩ
V_{IH}	min. 2.4V
V_{IL}	max. 0.8V
I_{IH}	max.1mA
I_{IL}	max. 0.2mA
Voltage range, absolute	-0.3V … 7V
Connector	37-pin DSub socket
Isolation	No (see page 96)

Fig. 190 – Pro-CNT-16/32 REVA: Specification

4.6.13 Pro-CNT-16/32-I REVA

The counter module Pro-CNT-16/32-I REVA has 16 counters (32 bit). All counter values can be loaded into the register with a single ***ADbasic*** command so that all counter values can be latched simultaneously. It is also possible to latch the counters individually. With a rising edge of a TTL pulse the 32-bit counter increments by one. The counter can be cleared by software command. The count rate can be derived by the difference between two successive readings of the latch. Since the register access is discrete-time, the frequency can be calculated online.

The counter inputs are optically isolated from the system circuitry as well as from other inputs. The event input is also isolated from the system circuitry. The input voltage range can be selected by jumpers. The default setting of the input voltage range is 24V.

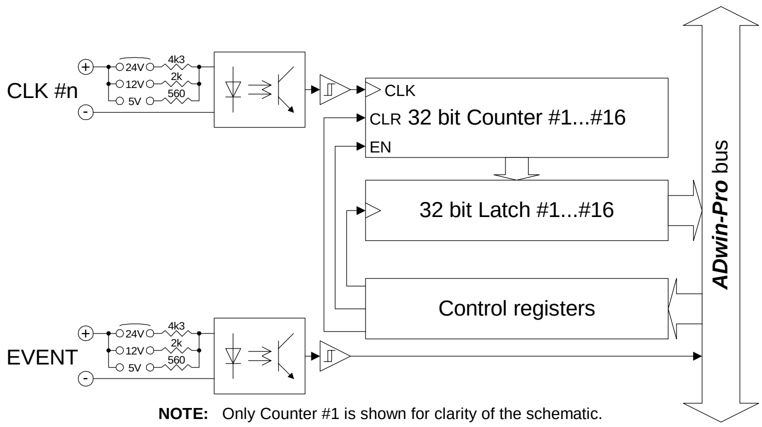


Fig. 191 – Pro-CNT-16/32-I REVA: Block diagram

The module Pro-CNT-16/32-I REVA is equipped with 16 times the components shown in the block diagram, exception: the event input and the control register.

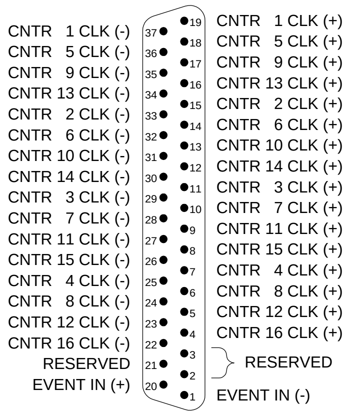


Fig. 192 – Pro-CNT-16/32-I REVA: Pin assignment

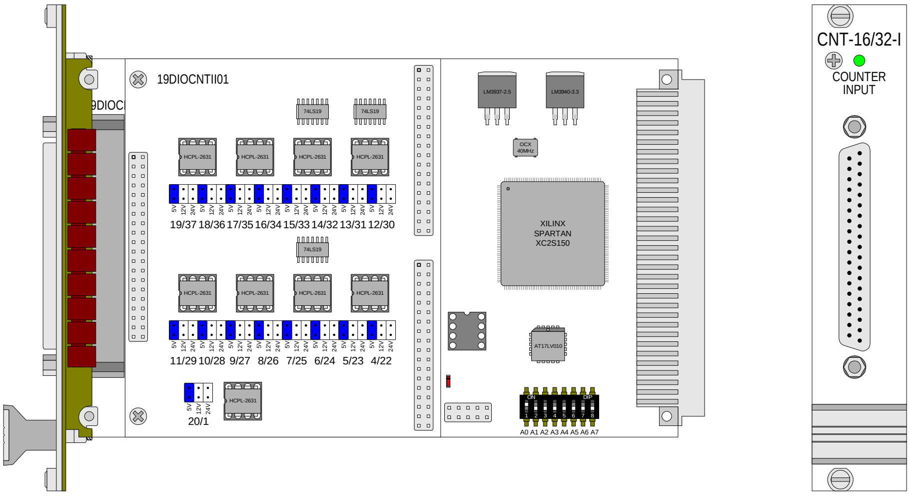


Fig. 193 – Pro-CNT-16/32-I REVA: Board and front panel

Counter	16 up counters		
Counter resolution	32 bit		
Event input	1		
Input current	typ. 7 mA / max. 15 mA		
input voltage range (selectable via jumpers)	0…5 V	0 … 12 V	0…24 V
Switching threshold for 0-low	0…0.8 V	0…1.6 V	0…3.2 V
Switching threshold for 1-high	4.5…5 V	10…12 V	20…24 V
Input resistance	510 Ω	1.51 kΩ	3.2 kΩ
Input over-voltage	-5 V … 8 V	-5 V … 16 V	-5 V … 30 V
Switching time	200 ns		
Connector	37-pin DSub socket		
Isolation	500 V channel-channel / channel-GND		

Fig. 194 – Pro-CNT-16/32-I REVA: Specification

4.6.14 Pro-CNT-VR2PW2(-I) REVA

To this module you find an improved successor module Pro-CO4-T REVA and Pro-CO4-I REVA (see page 103 / page 105).

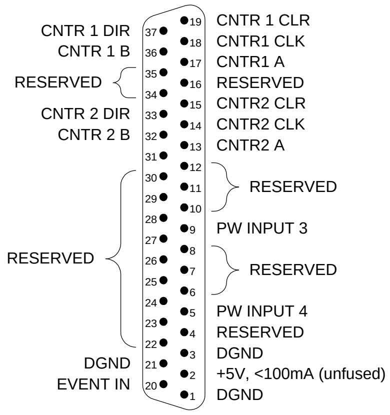


Fig. 195 – Pro-CNT-VR2PW2 REVA: Pin assignment

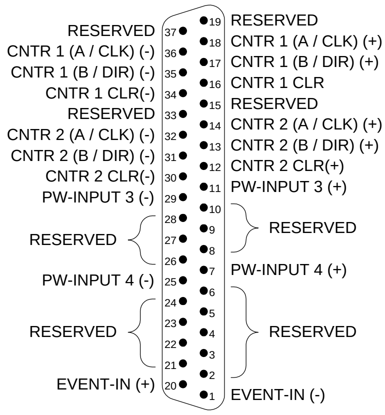


Fig. 196 – Pro-CNT-VR2PW2-I REVA: Pin assignment

Programming

The module Pro-CNT-VR2/PW2(-I) REVA has 2 up/down-counters (UD) and 2 impulse width counters (IW). It is a combination of the modules Pro-CNT-VR4(-I) (4 UD) and Pro-CNT-PW4(-I) (4 IW), so you can use the same instructions as for these modules.

The 1^{st} and 2^{nd} UD-counter is related to the 1^{st} and 2^{nd} counter of the module CNT-VR4 and the 1^{st} and 2^{nd} PW-counter is related to the 3^{rd} and 4^{th} counter of the module CNT-PW4.

4.6.15 Pro-CNT-PW4 REVA

To this module you find an improved successor module Pro-CO4-T REVA (see page 103).

The digital counter module Pro-CNT-PW4 REVA has four inputs for pulse width modulated signal acquisition. With this module you are able to determine the positive and negative pulse widths, the duty cycle, period time, and frequency. The 4 counters (32 bit) are clocked with a fixed 5MHz clock signal. At the rising and falling edges of the PW-input signal, the counter value will be stored in two separate latches.

Please, make sure that the delay of the event (via internal or external timer) is shorter than the period width of the highest input frequency to be measured.

Example: The signal whose positive and negative pulse widths you want to know has a frequency of 3.3kHz. The event has to arrive in a time interval of less than 303µs (= 1/3.3kHz).

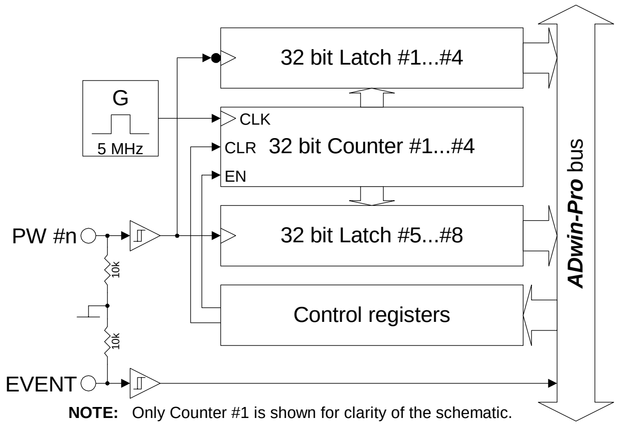


Fig. 197 – Pro-CNT-PW4 REVA: Block diagram

The module Pro-CNT-PW4 REVA is equipped with 4 times the components shown in the block diagram, exception: the event input and the control register which can only be found once on the modules.

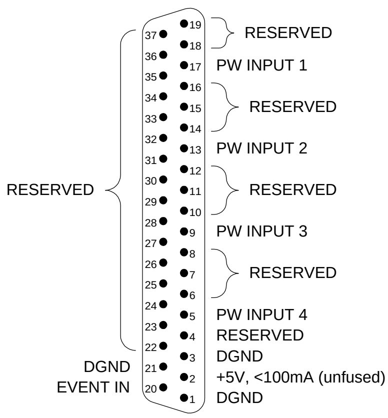


Fig. 198 – Pro-CNT-PW4 REVA: Pin assignment

	falling edge	rising edge
Input PW1	Latch 1	Latch 5
Input PW2	Latch 2	Latch 6
Input PW3	Latch 3	Latch 7
Input PW4	Latch 4	Latch 8

Fig. 199 – Pro-CNT-PW4 REVA: Allocation of the latches

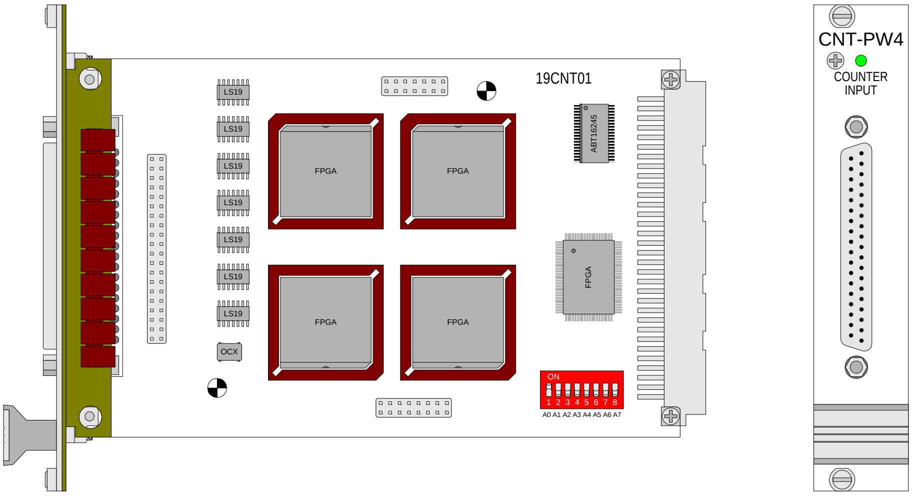


Fig. 200 – Pro-CNT-PW4 REVA: Board and front panel

Counter	4 impulse counters
Counter resolution	32 bit
Reference clock	5MHz
Inputs	4 TTL
V_{IH}	min. 2.4V
V_{IL}	max. 0.8V
I_{IH}	max. 20µA
I_{IL}	max. -50µA
Voltage range	-0.3V … 7V
Event input	1
Input resistance	10kΩ
Connector	37-pin DSub socket
Isolation	No (see page 101)
Power consumption	approx. 120mA

Fig. 201 – Pro-CNT-PW4 REVA: Specification

4.6.16 Pro-CNT-PW4-I REVA

To this module you find an improved successor module Pro-CO4-I REVA (see page 105).

The digital counter module Pro-CNT-PW4-I REVA has four inputs for pulse width modulated signal acquisition. The inputs are optically isolated from each other as well as from the system circuit. The switching time of only 200ns permits the reading of fast digital signals. The event input is isolated from the system, too. The input voltage range of the counter inputs can be selected by jumpers. The default setting of the input voltage range is 24V.

With this module you are able to determine the positive and negative pulse widths of up to four signals and to calculate the duty cycle, period time and frequency The 4 counters (32 bit) are clocked with a fixed 5MHz clock signal. At the rising and falling edges of the PW-input signal, the counter value will be stored in two separate latches.

Please, make sure that the delay of the event (via internal or external timer) is smaller than the period width of the highest input frequency. to be measured.

Example: The signal whose positive and negative pulse widths you want to know has a frequency of 3.3kHz Þ The event has to arrive in a time interval of less than 303µs (= 1/3.3kHz).

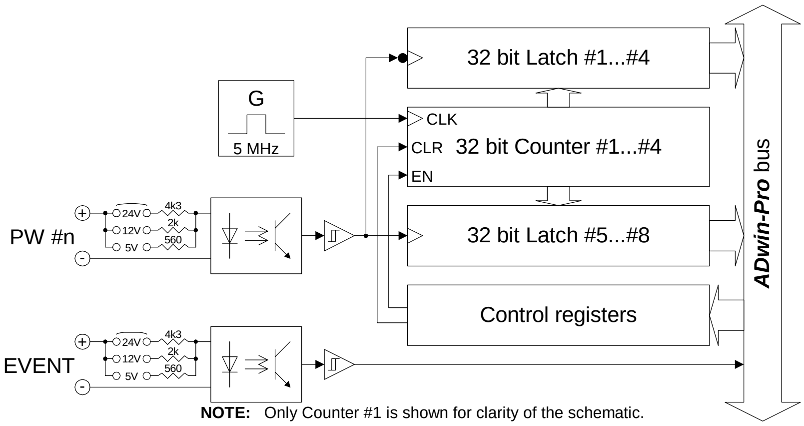


Fig. 202 – Pro-CNT-PW4-I REVA: Block diagram

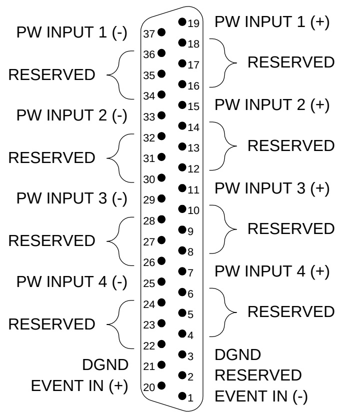


Fig. 203 – Pro-CNT-PW4-I REVA: Pin assignment

The module Pro-CNT-PW4-I is equipped with 4 times the components shown in the block diagram; exception: the event input and the control register which can only be found once on the modules.

	falling edge	rising edge
Input PW1	Latch 1	Latch 5
Input PW2	Latch 2	Latch 6
Input PW3	Latch 3	Latch 7
Input PW4	Latch 4	Latch 8

Fig. 204 – Pro-CNT-PW4-I REVA: Allocation of the latches

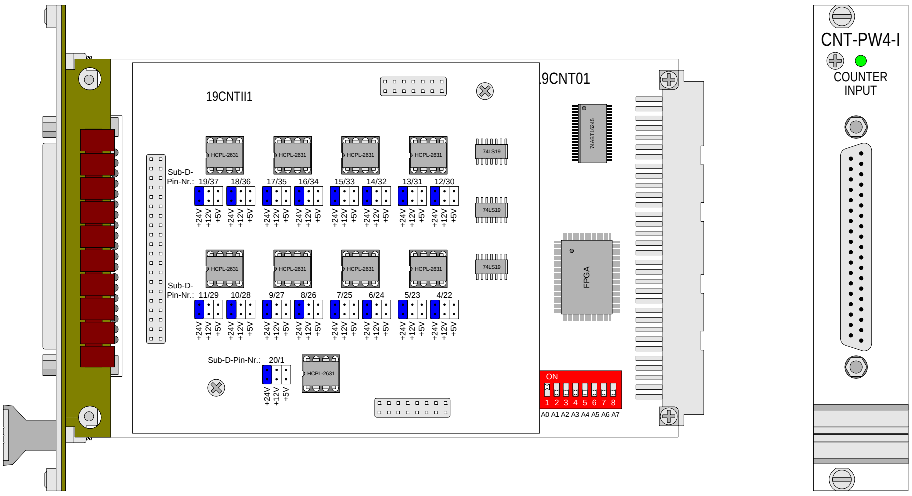


Fig. 205 – Pro-CNT-PW4-I REVA: Board and front panel

Counter	4 impulse counter		
Counter resolution	32 bit		
Event inputs	1		
Reference clock	5MHz		
Input current	typ. 7mA / max. 15mA		
input voltage range (selectable via jumpers)	0…5V	0 … 12V	0…24V
Switching threshold 0 (low)	0 … 0.8V	0 … 1.6V	0 … 3.2V
Switching threshold 1 (high)	4.5 … 5V	10 … 12V	20 … 24V
Series resistor	560 Ω	2kΩ	4.3kΩ
Input over-voltage	8V	16V	30V
Negative voltage	-5V for all ranges		
Switching time	200ns		
Isolation	500V channel-channel / channel-GND		
Connector	37-pin DSub socket		

Fig. 206 – Pro-CNT-PW4-I REVA: Specification

4.6.17 Pro-CO4-T REVA

The module Pro-CO4-T REVA is a configurable multi-purpose counter which provides 4 up/down counters as well as the analysis of up to 4 PWM signals. The counter inputs are designed for TTL logic. The functionality of the counter inputs and of the counters themselves can individually be selected via registers.

You may set different operating modes for the counters: Up/down counter, PWM analysis or four edge evaluation. After power-up of the Pro system the default setting of the counters is four edge evaluation with CLR input (the CLR input is not yet released).

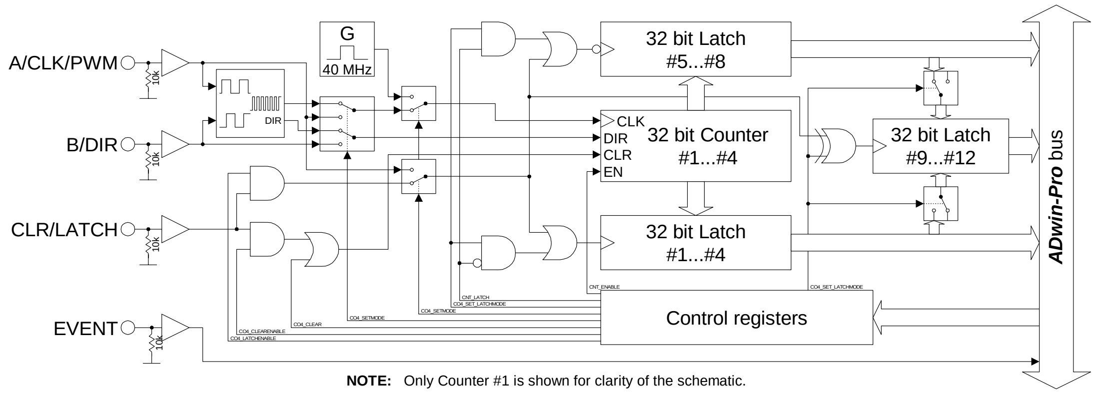


Fig. 207 – Pro-CO4-T REVA: Block diagram

Up/down counter (CLK and DIR signals)

The CLK and DIR signals are going directly to the 32-bit counter.

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 `CO4_CLEARENABLE()` or `CO4_LATCHENABLE()`.

You can clear or latch the counter when you get the signal CLR/LATCH logic "1". 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.

PWM analysis

With the PWM analysis the signal, which is to be measured, goes directly to the trigger inputs of the latches. For instance, the counter value in counter 1 is latched into latch 1 at a rising edge, at a falling edge it is latched into latch 5.

The ***ADbasic*** process is responsible for evaluating from the latch contents the high and low times, the duty cycle, period duration or frequency of the PWM signal.

Four edge evaluation of incremental encoders (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 signalsA, B and CLR are on logic "1" (software-selectable: clear, when only the CLR signal is on logic "1").

Event input

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

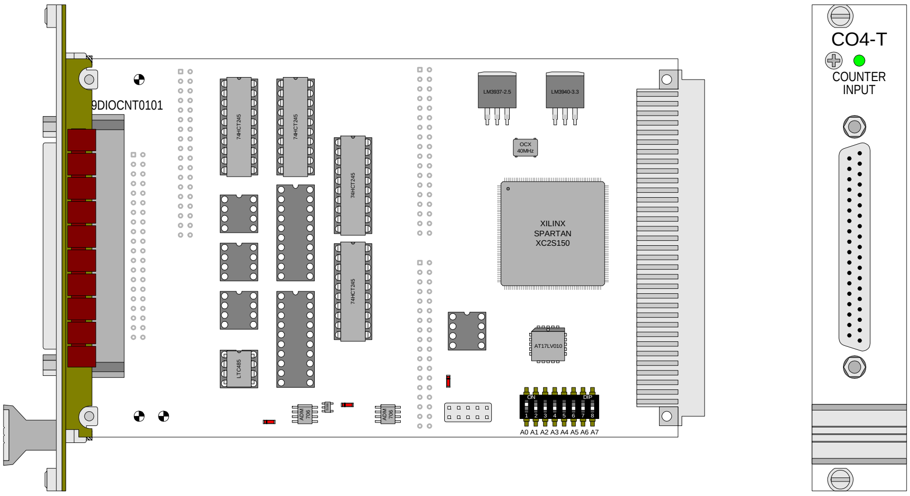


Fig. 208 – Pro-CO4-T REVA: Board and front panel

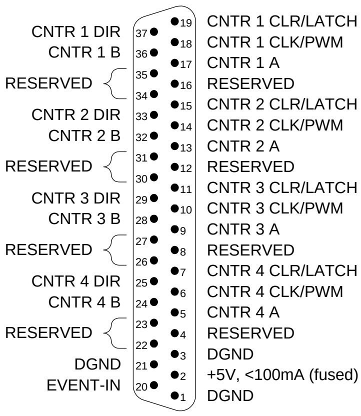


Fig. 209 – Pro-CO4-T REVA: Pin assignment Pro-CO4-T

Counter	4 multi-purpose counters
Counter resolution	32 bit
Input / output level	TTL logic
Event input	TTL logic
Reference clock	40 MHz (100 ppm)
Clock frequency four edge evaluation	5 MHz max. (at 90° phase-shift of the signals)
Clock frequency up/down counter	20 MHz max.
Reference frequency PWM analysis	40 MHz
Connector	37-pin DSub socket
Power consumption	approx. 150 mA
Isolation	No (see page 105)

Fig. 210 – Pro-CO4-T REVA: Specification

4.6.18 Pro-CO4-I REVA

The basic functions of the module Pro-CO4-I REVA are similar to those of the module Pro-CO4-T REVA (see page 103).

What is different is the fact that the counter inputs of the module Pro-CO4-I 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.

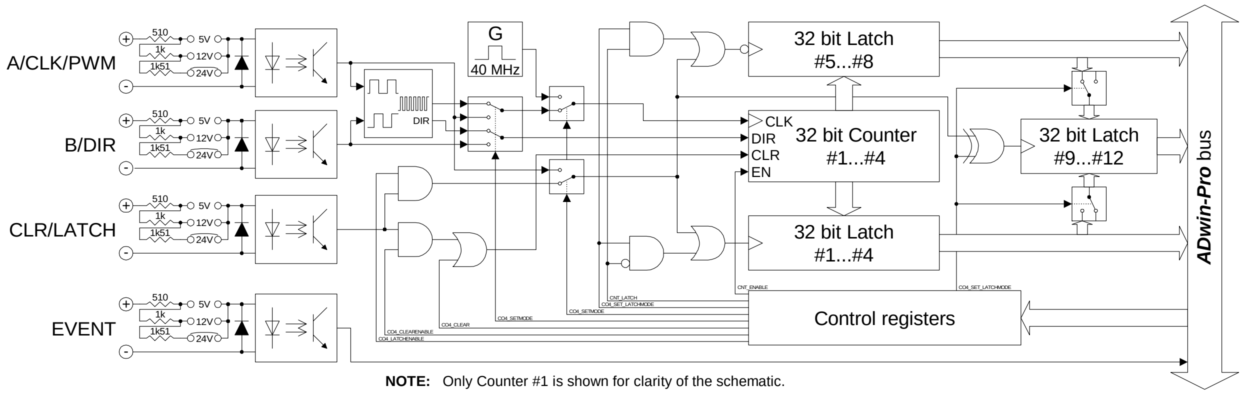


Fig. 211 – Pro-CO4-I REVA: Block diagram

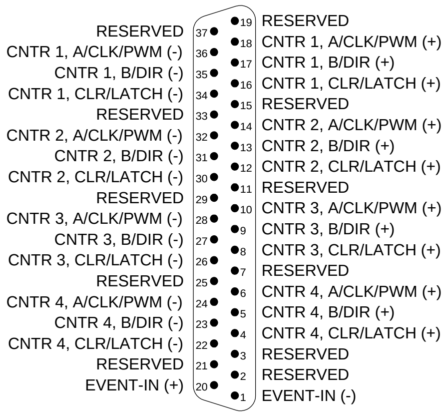


Fig. 212 – Pro-CO4-I REVA: Pin assignment

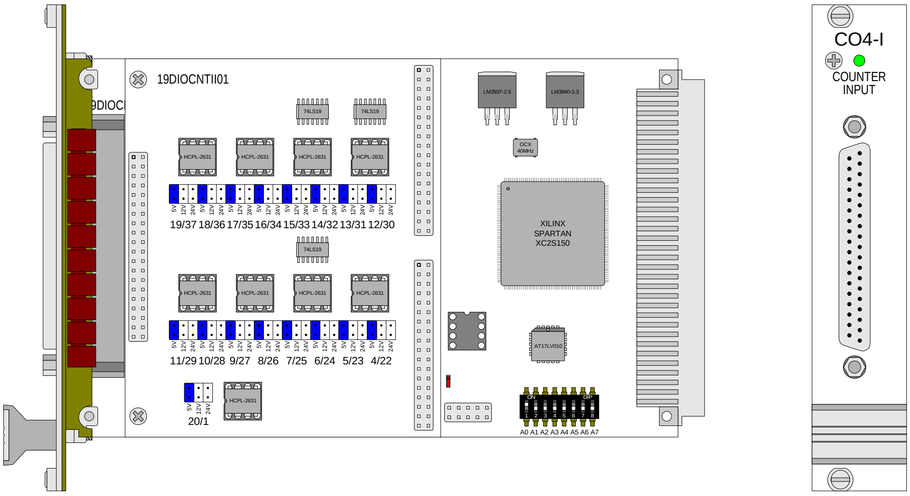


Fig. 213 – Pro-CO4-I REVA: Board and front panel

Counter	4 multi purpose counters		
Counter resolution	32 bit		
Event inputs	1		
Reference clock	40MHz (100ppm)		
Clock frequency four edge evaluation	5MHz max. (at 90° phase shift of the signals)		
Clock frequency up/down counter	10MHz max.		
Reference frequency PWM analysis	40MHz		
Input current	typ. 7mA / max. 15mA		
input voltage range (selectable via jumpers)	0…5V	0 … 12V	0…24V
reliable switching threshold[1] for 0 (low)	0…0.8V	0…1.6V	0…3.2V
reliable switching threshold[1] for 1 (high)	4.5…5V	10…12V	20…24V
Series resistor	510 Ω	1.51kΩ	3.02kΩ
Input over-voltage	8V	16V	30V
Negative voltage	-5V for all ranges		
Switching time	100ns		
Isolation	500V channel-channel / channel-GND		
Connector	37-pin DSub socket		
Power consumption	approx. 200mA		

Fig. 214 – Pro-CO4-I REVA: Specification

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.

4.6.19 Pro-CO4-D REVA

The basic functions of the module Pro-CO4-D REVA are similar to those of the module Pro-CO4-I REVA (see page 105).

In addition, the module Pro-CO4-D REVA is equipped with 2 decoders for the connection of incremental encoders with SSI interface. All inputs are differential and can be used for RS422/485 levels (5V). Finally, the signals A, B and CLR are checked if they show short circuits or a cable break; you can obtain this information with the instruction `CO4_GETSTATUS`.

It is possible to operate the EVENT input in differential mode as well as in single-ended mode. If only a single-ended signal is available, it is to be set at EVENT. The negative EVENT input is not set.

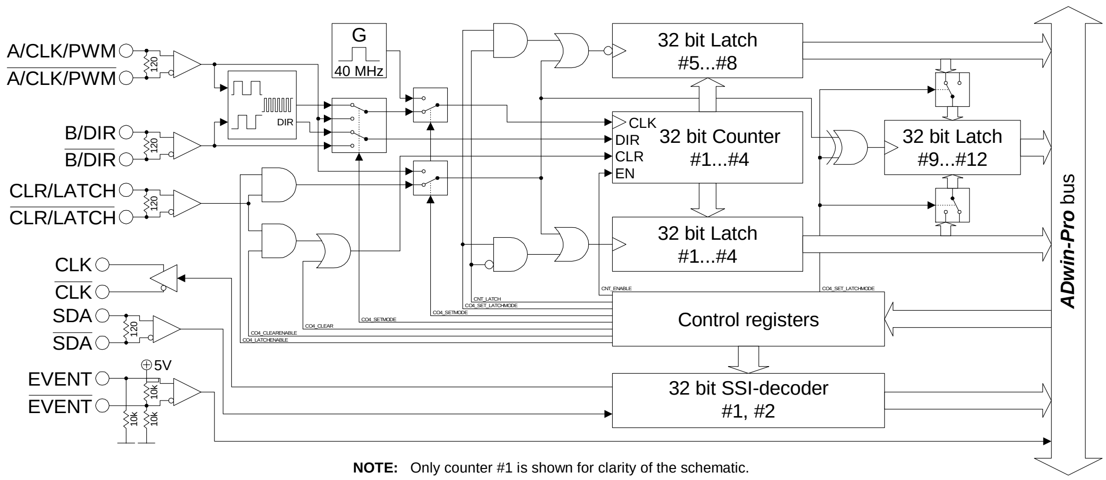


Fig. 215 – Pro-CO4-D REVA: Block diagram

SSI decoder

An incremental encoder with SSI interface can be connected to one of the two decoders. The signals are differential, too, and have RS422/485 levels.

The clock rates as well as the resolution of the encoder (up to 32 bits) are programmable via pre-scaler (of approx. 40 kHz to 1 MHz). A conversion from gray into binary code is made by a routine to be programmed in the ***ADbasic*** process (see below).

```
'PAR_1 = gray-value to be converted
'PAR_9 = result of the gray to binary conversion

DIMm, n AS LONG

EVENT:
 IF(par_2=1) THEN     'start of conversion
   m=0                'clear values of previous conv.
    PAR_9=0            ' -"-
    FOR n=1 TO 32      'go through all 32 bits
     m=(SHIFT_RIGHT(PAR_1,(32-n)) AND 1) XORm
      PAR_9=(SHIFT_LEFT(m,(32-n))) OR PAR_9
    NEXT n
    PAR_2=0            'enable next conversion
 ENDIF
```

Fig. 216 – Listing: Conversion from gray code into binary code

DIP switch on the module (component side, middle)

You can determine slow and fast input signals with high measurement rate without switching, by connecting 1 counter input with 2 counters. For this you have to change the DIP switch positions on the module:

– Switch counter 3 (additionally to counter 1) to counter input 1:

  Push the switches of the *upper* double DIP switches upwards.

– Switch counter 4 (additionally to counter 2) to counter input 2:

  Push the switches of the *lower* double-DIP switches upwards.

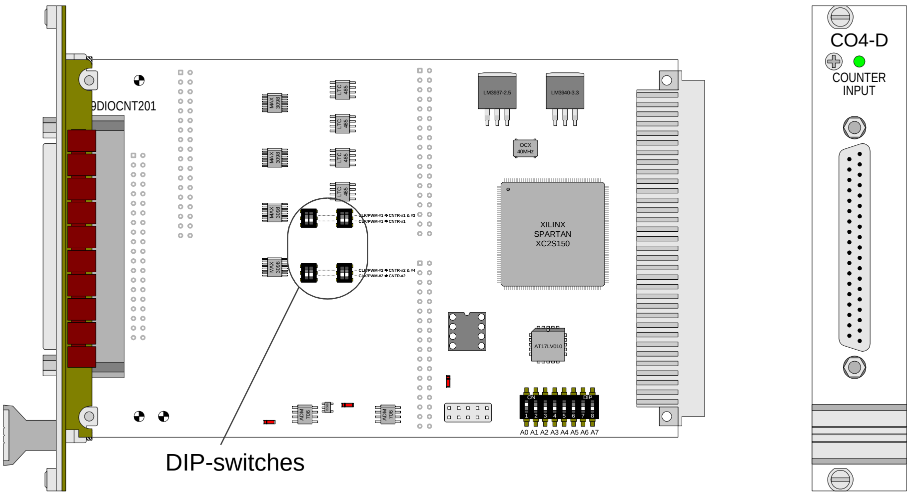


Fig. 217 – Pro-CO4-D REVA: Board and front panel

The figure below illustrates the DIP switch positions and the resulting input-counter-connections.

Program one of the two counters with CLK and DIR signal inputs (up/down counter), the other with PWM-input (PWM analysis). In a corresponding ***ADbasic*** process you can now determine the frequency or period duration of the signal in a wide frequency range.

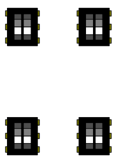
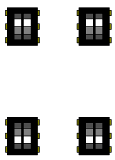
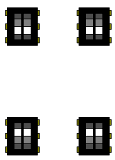
DIP switch position	Input counter# A/CLK/PWM	Counter			
		CNTR-#1	CNTR-#2	CNTR-#3	CNTR-#4
	1	✓	-	-	-
	2	-	✓	-	-
	3	-	-	✓	-
	4	-	-	-	✓
	1	✓	-	✓	-
	2	-	✓	-	-
	3	-	-	-	-
	4	-	-	-	✓
	1	✓	-	-	-
	2	-	✓	-	✓
	3	-	-	✓	-
	4	-	-	-	-

Fig. 218 – Pro-CO4-D REVA: Allocation of Input to Counter with DIP switches

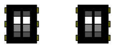
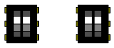
DIP switch position	Input counter# A/CLK/PWM	Counter CNTR-#1	CNTR-#2	CNTR-#3	CNTR-#4
	1	✓	-	✓	-
	2	-	✓	-	✓
	3	-	-	-	-
	4	-	-	-	-

Fig. 218 – Pro-CO4-D REVA: Allocation of Input to Counter with DIP switches

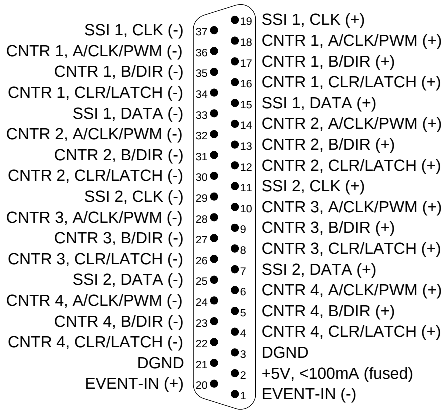


Fig. 219 – Pro-CO4-D REVA: Pin assignment

Counter	4 multi purpose counters + 2 SSI decoders
Counter resolution	32 bit
Input/output levels	RS422/485 compatible (5V differential, 120 Ω bus terminating resistor, see also block diagram)
Event input	1 differential (single-ended operation possible)
Reference clock	40MHz (100 ppm)
Clock frequency four edge evaluation	5MHz max. (at 90° phase shift of the signals)
Clock frequency up/down counter	20MHz max.
Reference frequency PWM analysis	40MHz
Clock frequency SSI decoder (CLK)	1MHz max.
Connector	37-pin DSub socket
Power consumption	approx. 200mA

Fig. 220 – Pro-CO4-D REVA: Specification

4.6.20 Pro-PWM-4 REVA

The output module Pro-PWM-4 REVA generates pulse width modulated signals (PWM signals) at four outputs. Each (PWM) signal can be configured individually via software; that means, they can be configured separately.

The function of the module is significantly characterized by its 4 counters (16 bit) and 8 registers (16 bit), 4 for the duration of the low voltage level and 4 for the high voltage level.
The counters are clocked by a crystal oscillator with a fixed frequency of 5MHz. This frequency can roughly be prescaled in 2^n steps ($0 < n < 7$, $n \in N$). The prescaling in conjunction with the duration of the high and low time defines the output frequency of the PWM signals. The output of the PWM signals is attained by evaluating the register values and the counter values via an RS-flip-flop.

By setting a register, the counters can be enabled or disabled. But do not confuse this with enabling or disabling the PWM output (putting it into a "static" mode). This can only be made by the command `PWM_OUT()`, which sets the output to a defined status, when the counter is enabled.

The lowest output frequency at a still definable duty cycle of approx. 0...100% is about 0.6Hz.

The highest output frequency where the duty cycle can be still defined in 1%-steps, is 50kHz.

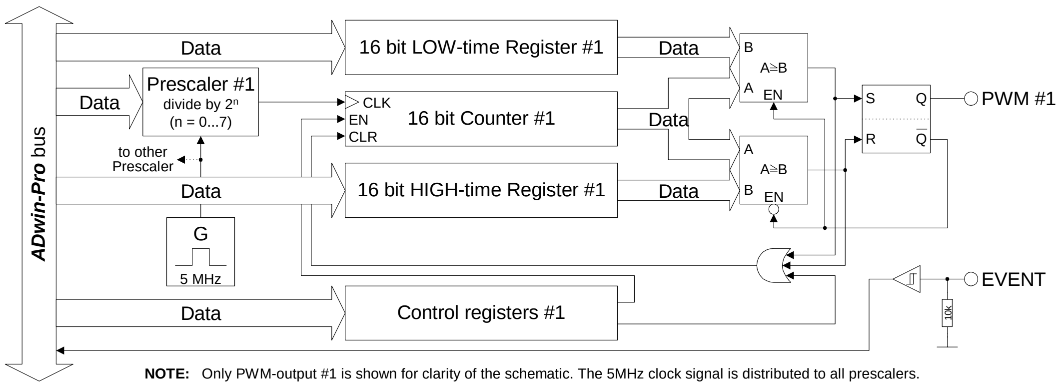


Fig. 221 – Pro-PWM-4 REVA: Block diagram

The module Pro-PWM-4 is equipped with 4 times the components shown in the block diagram; exception: the event input and the 5MHz reference oscillator, which can only be found once on each module.

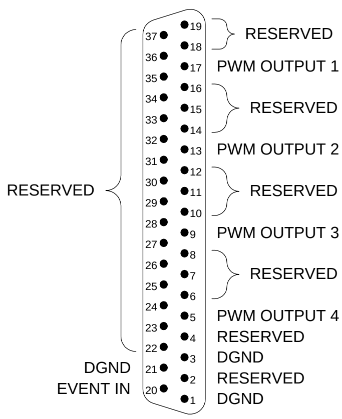


Fig. 222 – Pro-PWM-4 REVA: Pin assignment

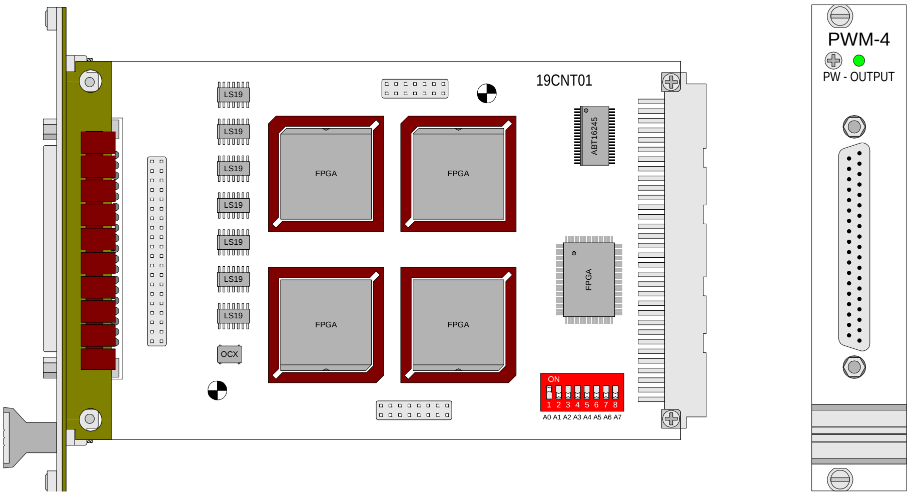


Fig. 223 – Pro-PWM-4 REVA: Board and front panel

Output channels		4 PWM channels
Outputs		TTL
Counter-/register width		16 bit
f_{clk} after Prescaler	Div. by 1 (2^0)	200 ns (5 MHz)
	Div. by 2 (2^1)	400 ns (2.5 MHz)
	Div. by 4 (2^2)	800 ns (1.25 MHz)
	…	
	Div. by 128 (2^7)	25.6 µs (≈ 39 kHz)
V_{OH}		2.4 V min.
V_{OL}		0.8 V max.
Output current		5 mA per channel max.
Event input		Positive TTL
Connector		37-pin DSub socket
Isolation		No (see page 112)

Fig. 224 – Pro-PWM-4 REVA: Specification

Information about programming and programming examples can be found after the description of the module Pro-PWM-4-I REVA.

4.6.21 Pro-PWM-4-I REVA

The module Pro-PWM-4-I REVA generates pulse width modulated signals (PWM signals) at four outputs. Each (PWM-) signal can be configured individually via software, that means they can be configured separately.

The function of the module is significantly characterized by its 4 counters (16 bit) and 8 registers (16 bit), 4 for the duration of the low voltage level and 4 for the high voltage level.
The counters are clocked by a crystal oscillator with a frequency of 5MHz. By defining a (scale) register the frequency can be roughly prescaled in 2^n steps ($0 < n < 7$, $n \in N$). By presetting the duration of the high- and low-pulses, the output frequency of the PWM signals is defined.

The output of the PWM signals is attained by evaluating the register values and the count rate via an RS-flip-flop.

The inputs are optically isolated from the system circuitry and from other inputs. The event-input is isolated from the system circuitry as well. The input voltage range of the counter inputs can be selected by jumpers. The default setting of the input voltage range is 24V.

By setting a register, the counters can be enabled or disabled. But do not confuse this with enabling or disabling the PWM output (putting it into a "static" mode). This can only be made by the command `PWM_OUT(…)`, which sets the output to a defined status, when the counter is enabled.

The lowest output frequency at a still definable duty cycle of approx. 0...100% is about 0.6Hz.

The highest output frequency where the duty cycle can be still defined in 1%-steps, is 50kHz.

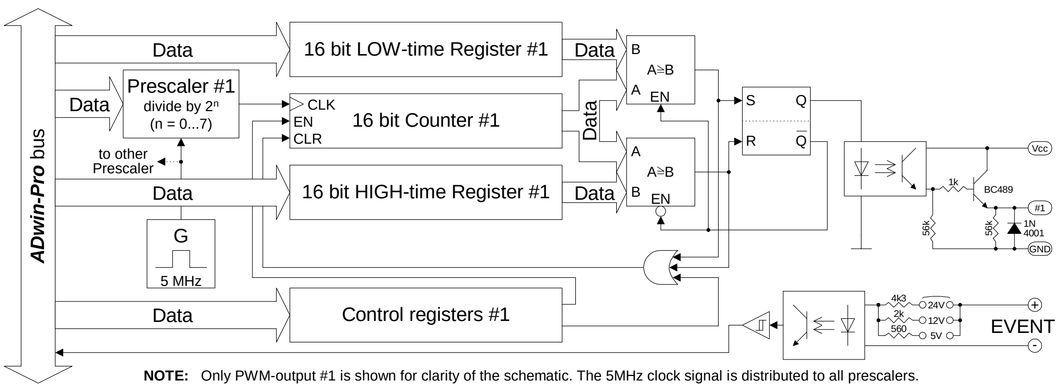


Fig. 225 – Pro-PWM-4-I REVA: Block diagram

The module Pro-PWM-4-I REVA is equipped with 4 times the components shown in the block diagram; exception: the event input and the 5MHz reference oscillator, which can only be found once on each module.

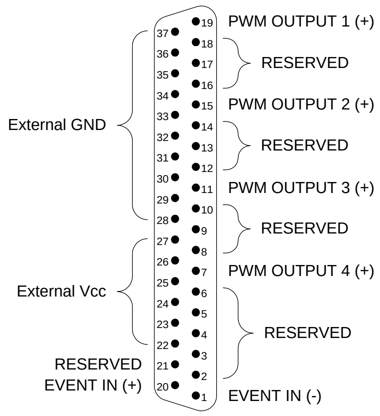


Fig. 226 – Pro-PWM-4-I REVA: Pin assignment

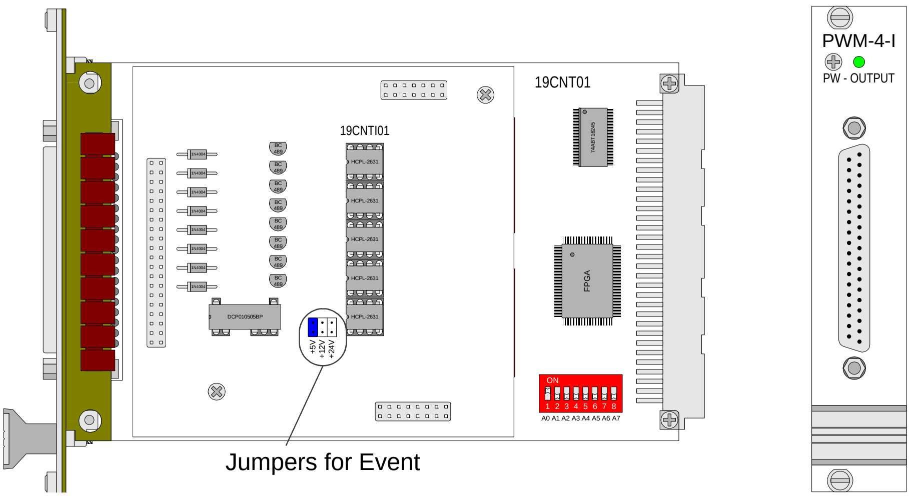


Fig. 227 – Pro-PWM-4-I REVA: Board and front panel

Output channels		4
Counter-/register width		16 bit
f_{clk} after Prescaler	Div. by 1 (2^0)	200 ns (5 MHz)
	Div. by 2 (2^1)	400 ns (2.5 MHz)
	Div. by 4 (2^2)	800 ns (1.25 MHz)
	…	
	Div. by 128 (2^7)	25.6 µs (≈ 39 kHz)
Output voltage		5…30 V DC with an external power supply
Output current		100 mA max. per channel
Event inputs		Pos. TTL
Voltage drop		0.5 V max.
Switching time		10 µs
Event input		1
Event input voltage		5 V, 12 V, 24 V (selectable via jumpers)
Connector		37-pin DSub socket
Isolation		500 V channel-channel / channel-GND

Fig. 228 – Pro-PWM-4-I REVA: Specification

4.6.22 Programming the Module Pro-PWM-4(-I)

Programming the four PWM outputs on the ***ADwin-Pro*** module PWM-4 is made by using the PWM commands which are available when the file <`ADWPDIO.INC`> has been included into the ***ADbasic*** source code.

The equation shows the formula for calculating the output frequency of the PWM signal.

$$f_{out} = \left(\frac{5\text{MHz}}{\text{presc}}\right) \cdot \left(\frac{1}{ts_{per}}\right) = \left(\frac{5\text{MHz}}{\text{presc}}\right) \cdot \left(\frac{1}{ts_{low} + ts_{high}}\right)$$

Legend for the equation:

ts_{low} = duration of the low level

ts_{high} = duration of the high level

ts_{per} = period duration

presc = prescaler factor

f_{out} = output frequency

Instructions for the PWM modules

PWM_SET

`PWM_SET` does the settings of the defined module for the prescaler and the duration of the high- and low-pulses of the PWM output channel.

`PWM_SET(module, channel, prescale, low, high)`

Parameters

		LONG	FLOAT	VAR	CONST
module	defined module address	✓	-	✓	✓
channel	PWM output channel (1...4)	✓	-	✓	✓
prescale	pre-scaler value 0…7, corresponds to a division by 2^0 … 2^7	✓	-	✓	✓
low	number of cycles for the low-time, after the prescaler.	✓	-	✓	✓
high	number of cycles for the high-time, after the prescaler	✓	-	✓	✓

Description

The values of the parameters *low* und *high* represent the number of cycles after the prescaler, which have to be achieved by the counter to change the logic level.

The prescaler is clocked with a frequency of 5MHz.

PWM_ENABLE

`PWM_ENABLE` enables or stops the counters corresponding to the specified PWM outputs.

`PWM_ENABLE(module, pattern)`

Parameters

		LONG	FLOAT	VAR	CONST
module	defined module address	✓	-	✓	✓
pattern	Bit pattern for the PWM outputs. Bits 0-3 correspond to outputs 1-4: bit=0: enable counter bit=1: stop counter	✓	-	✓	✓

Description

This instruction does not affect the PWM outputs, only the counters corresponding to the outputs. It should be used only in combination with the instruction `PWM_OUT`.

PWM_OUT

`PWM_OUT` sets the specified PWM output channel to high or low level.

`PWM_OUT(module, channel, level)`

Parameters

		LONG	FLOAT	VAR	CONST
module	defined module address	✓	-	✓	✓
channel	PWM output channel (1…4)	✓	-	✓	✓
level	Output value to be set 0: U_{out} = logical "0" 1: U_{out} = logical "1" (or V_{EE} at PWM-4-I)	✓	-	✓	✓

Description

This instruction does only work, when the counters corresponding to the PWM outputs are enabled with `PWM_ENABLE`.

Programming Example / Program Description

The program <`Pro_PWM_4.bas`> provided on the ***ADwin*** CDROM in the folder <`C:\ADwin\ADbasic3\Samples_ADwin_Pro`> generates identical PWM signals at the outputs 1…4 with a frequency of 1kHz.

With the parameters `PAR_1`…`PAR_14` you will be able to change in ***ADbasic*** the following values in the parameter window:

– The high-level duration of the PWM signals: `PAR_1`, `PAR_3`, `PAR_5`, `PAR_7`.

– The low-level duration of the PWM signal: `PAR_2`, `PAR_4`, `PAR_6`, `PAR_8`.

– The division factor (prescaler) for the frequency of 5MHz, which will be generated by the crystal oscillator on the ***ADwin-Pro*** module PWM-4: `PAR_9` … `PAR_12`.

– The enabling of the 16-bit counters (only the internal counter and not the PWM outputs!) : `PAR_13`.

– The module address: `PAR_14`.

4.6.23 Pro-Comp-16 REVA

The input module Pro-Comp-16 REVA provides 16 input channels with a comparator each. The analog signals are acquired parallel and converted with a 10-bit resolution. Depending on switching thresholds, which are separately adjustable by software for each channel, digital signals are generated from these measurement values (1/0).

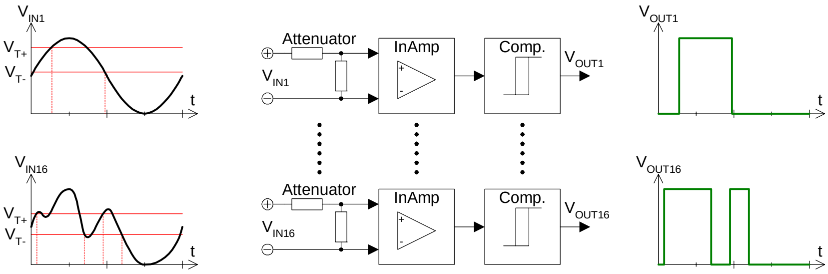


Fig. 229 – Pro-Comp-16 REVA: Block Diagram

The input voltage range of the module is between -2V…+8.23V, other voltage ranges on request. The signals are acquired with 20MHz per channel.

Providing information

Information about the acquired signals is available per software:

– The digital signals (1/0) of all input channels
– The current converted measurement values
– The maximum and minimum of the acquired measurement values
– The last 1024 measurement values of 2 selected channels
– The digital signals (1/0) of measurement differences

After all measurement values have been acquired they are evaluated quasi-differential, which means that for all 8 channel pairs (1/2, 3/4, ..., 15/16) the differences of the measurement vaues is calculated. Depending on the switching thresholds, digital signals (1/0) are generated from these differences, which can be selected via software.

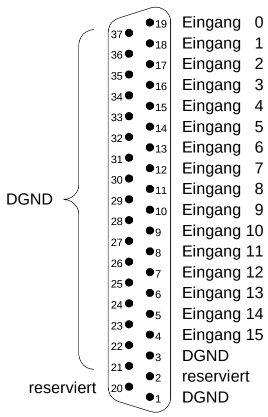


Fig. 230 – Pro-Comp-16 REVA: Pin assignment

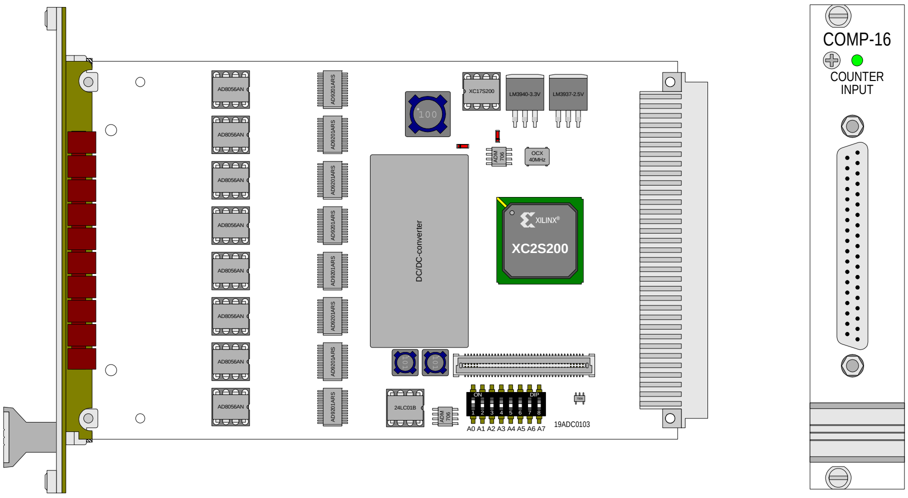


Fig. 231 – Pro-Comp-16 REVA: Printed circuit board and front panel

Technical data

Input/output channels	16 single-ended inputs with a comparator each Comparators with individually selectable switching thresholds
Input resistor	10kΩ
Input voltage range	-2V ... +8.23V
Sampling rate	typ. 20Msps
Resolution	10-bit
Accuracy	±4 digits
Connectors	37-pin D-SUB socket

Fig. 232 – Pro-Comp-16 REVA: Specification

4.6.24 Pro-Storage Rev. A

The module Pro-Storage writes (or reads out) large quantities of data from an *ADbasic* process to an (exchangeable) storage medium. Therefore data can be stored during stand-alone operation of the *ADwin* system at long-term measurements. On the module there is a battery-backed real-time clock to datestamp the data.

Storage media

The Pro-Storage module supports PCMCIA memory cards, Compact-Flash-cards in combination with an adapter card as well as 1" and 1.8" hard disks.

Basically the storage media can be used with a reading device of a PC to read out the data or write the data to the media. In any case, the medium must be initialized by the program <Pro-Storage.exe> before using it.

The user must program writing to (or reading) the medium during operation in a low-priority process. The process is running additionally to the already running open-loop, closed-loop or measurement processes; data is exchanged via a global memory (FIFO). A standard example for such a low-priority process is included in delivery, illustrating how to write data to the storage medium.

The further module description is into the following sections:

General Operating Information

In any case, before using the storage medium it should be initialized with the program <Pro-Storage.exe>.

Do not remove the storage medium as long as data are accessed (left, lower LED is blinking). As a consequence, the data exchange would be interrupted.

If data of the Pro-Storage module is processed differently than with the program Pro-Storage.exe, for instance using a read access device, the following items are to be considered:

1. The size of the files must not be changed.
2. The files must not be deleted or generated.
3. During the writing process the file end must not be exceeded.
4. The data has to absolutely remain in its physical position.

If the rules are not kept, the data will be overwritten or destroyed.

Module Design

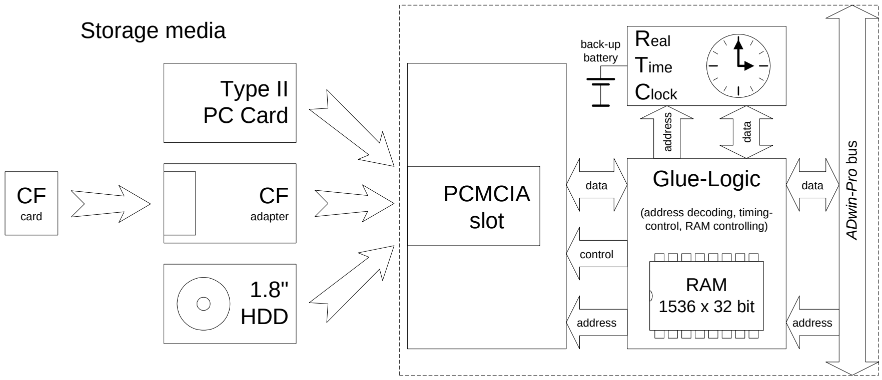


Fig. 233 – Pro-Storage: Block Diagram

The glue-logic of the Pro-Storage module has an internal buffer, which can receive 12 sectors with 128 data words à 32-bits. The glue-logic buffers all data, which is read from or written into the storage medium. Data is exchanged with the storage medium sector by sector.

The real-time clock works independently of all other components and is battery-backed. It provides date and time.

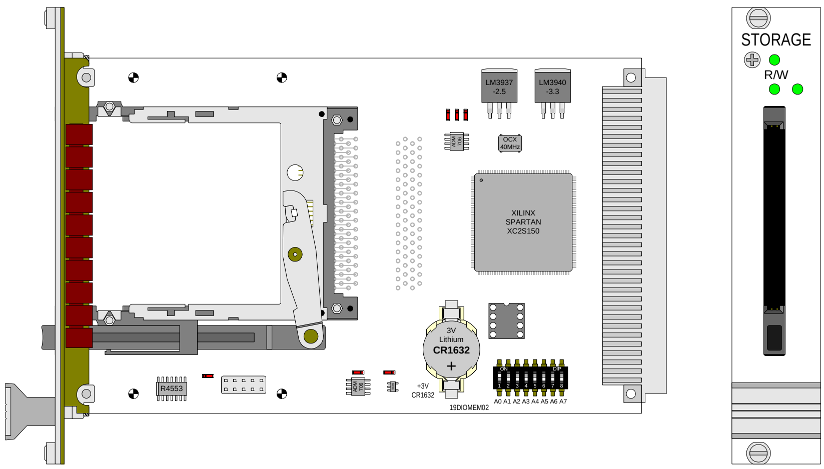


Fig. 234 – Pro-Storage: Printed circuit board and front panel

The module has a width of 1 HP and requires one slot.

LED function

On the front panel there are 3 two-colored LEDs.

The lower left LED displays the status of the storage medium:

- Green: The storage medium is correctly inserted or
  Read access to the storage medium
- Red: Storage medium is removed or
  Write access to the storage medium

The top and lower right LEDs are individually programmable. (see *ADbasic* instruction **SETLED**).

The Real-Time Clock

The module is equipped with a real-time clock from Epson RTC-4553AA. With simple *ADbasic* instructions, date and time are set and read out, so that a specified time can be assigned to measurements. Time and glue-logic work autonomously.

The time must be specified by a valid date and time of day; it has a resolution of one second. Leap years are considered.

The clock is battery-backed and can remain up to 2 years without any external power supply (when the Pro-system is turned off). Replace the buffer battery after 2 years by a 3V Lithium cell, type CR1632.

Select the storage medium

Select the Storage Medium

For selecting the storage medium type there are the following criteria (as of middle 2003); the difference in memory size is not essential.

Medium	Re-writable	Mechan. Strain	Access Time[a]
PCMCIA memory card (type II)	o[b]	+	+
Compact-Flash card, also with adapter card	o[b]	+	+
1"- und 1.8" hard disks	++	–	–

a. After a communication break

b. approx. 1 million times

We recommend that PCMCIA cards of the manufacturer SanDisk are used as industrial grade version (memory sizes of up to 2 GB). PCMCIA cards, which are identical in construction, but in a more rugged enclosure (also IP54 and IP68) are also available from the manufacturer Altec.

The reading and writing rates depend on the storage medium and on the writing and reading processes. Depending on the type and manufacturer there are great differences between the storage media.
For the PCMCIA cards mentioned above, the writing rate was evaluated under ideal conditions by the example process `<Pro-STORAGE-WR.bas>`: approx. 140kB/s for a file of 1MB and approx. 190kB/s for a file of 10MB.

The following factors reduce the writing rate:

– Interruption of the (low-priority) writing/reading process

– The small size of the file to be written to (storage medium-specific)

– In user-specific writing/reading processes: low efficiency

– Hard disks: Longer breaks between writing and reading sequences

Hard disks turn into sleep mode after some seconds (for the exact value see the datasheet of the manufacturer); the rotation of the hard disk stops. A new writing process must therefore wait until the hard disk reaches its full rotation speed again (which takes some seconds, see datasheet). Dimension the FIFO array so that it is large enough to buffer all incoming data during the waiting period.

Before a storage medium can be used it must be partitioned and formatted, and initialized by the program `<Pro-Storage.exe>` (see "Installing the Storage Medium" on page 123).

Insert the storage medium

Insert the storage medium in the correct position (the edge connector up front, double guide bar at the top). After the storage medium has been correctly inserted, the lower green LED flashes for a short moment.

Remove the storage medium

To remove the storage medium press the lever under the storage medium up to the stop contact and remove it. After the storage medium has been removed the left, lower red LED flashes for a short moment.

Installing the Storage Medium

Normally storage media are already partitioned and formatted upon delivery. In any case the storage medium should be initialized.

For questions about the partitioning pay attention to the notes of the manufacturer. If necessary, you may format the storage medium yourself:

Formatting

– Format the partition with the file system FAT16 or FAT32.

  FAT16 is necessary for Windows versions up to Win95 SR1 and can be used for storage media of up to 2 GB. If the hard disk has more than 504MB FAT32 should be used.

  Ensure that the partition table in the Master Boot Record is not overwritten after formatting (using the PC).

Initializing

– Initialize the storage medium using the program `<Pro-Storage.exe>` (in the Windows start menu under `Programs\ADwin`); the program is described in the section "To initialize / to change initialization".

  During initialization up to 10 files are generated on the storage medium, into which the data is written. During initialization the final size of the files is defined. The file information is stored twice, so that the file management accesses the data both under DOS / Windows as well as on the *ADwin* system.

  Writing data to or reading data from the *ADwin* system is made in a low-priority *ADbasic* process (section "Use the storage medium"). To keep the data structure as simple as possible, the data are stored in a linerary form on the storage medium. All sectors of a file are stored one after the other (= no fragmentation).

In addition to the initialization the program `<Pro-Storage.exe>` executes the following tasks:

– Transfer data between PC and storage medium (page 126) and

– To set the real-time clock (page 120).

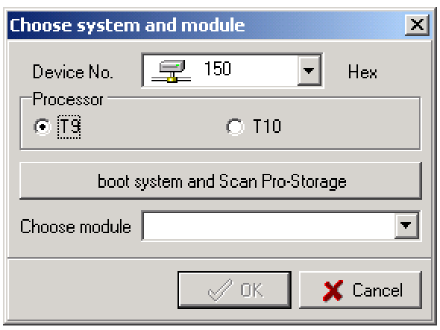
To initialize / to change initialization

Program is in process

If the program `<Pro-Storage.exe>` is already running, activate the `Read File Structure` menu item (at the lower edge of the window). Continue with the section "Read file structure".

After program start

After program start this window appears:



– Insert a storage medium into the Pro-Storage module.

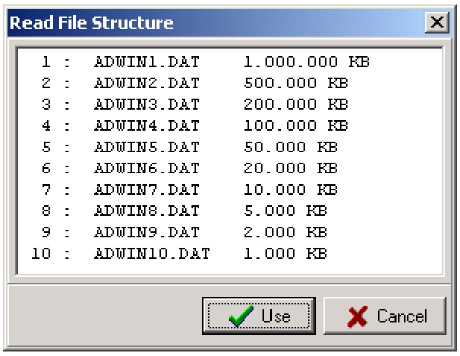
– Select the device no. of the *ADwin-Pro* system and press the button "boot ADwin-System and scan Pro-Storage".

Now the Pro-system boots and several processes for initialization of the storage medium are transferred. One of these processes checks, if there are Pro-Storage modules in the system and displays them in a list under `Choose module`. The modules differ from each other in the specified module address (see "Setting the module's addresses" on page 9).

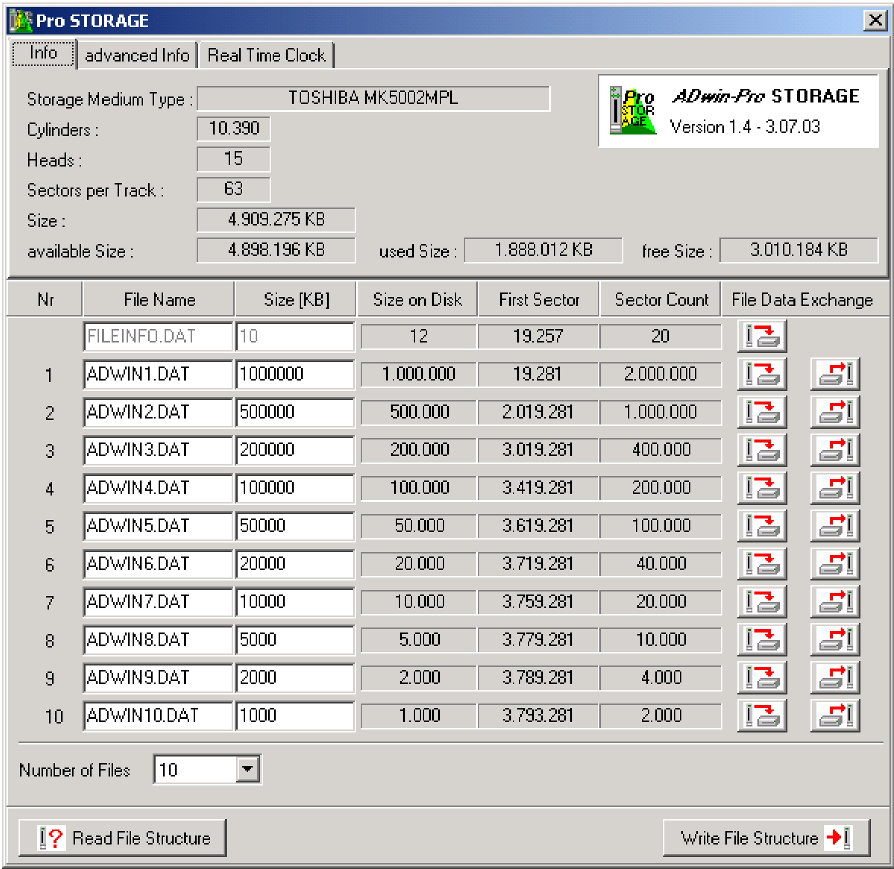
– Select a Pro-Storage module and confirm with `OK`.

Read file structure

If the storage medium has already been initialized, the existing file structure is displayed, otherwise the following window remains empty:



With `Use` the data structure is accepted, with `Cancel` you accept a standard data structure (a single file, with the size of 1000 kB) or the previous one. The data structure can be changed in the next window.



– Set first in the line `Number of Files`, how many files (1...10) you would like to use. In the table above the corresponding line number is activated.

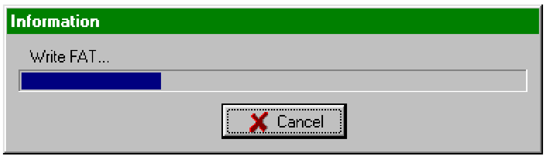
– If necessary adapt for each file the name and size (except for `FILEINFO.DAT`). The file name must correspond to the DOS convention (8+3), and the file size is indicated in `kB` (= 1.024 byte).

The value `Free Size` (bottom right, under "Info"), indicates how many kilobytes can still be used on the storage medium.

The column `Size on Disk` indicates how many kilobytes the file needs on the storage medium. Several sectors (see `advanced Info`; 1 sector = 512 bytes) make a cluster, therefore the file can have more kilobytes than entered.

Write file structure

– Transfer the specified data structure with `Write File Structure` to the storage medium.



The storage medium can only be removed after the initialization is completed.

– Now the storage medium is initialized and data can be written to or read from the Pro-Storage module.

With the initialization of the storage medium information about the glue-logic is saved twice:

1. It is completely stored in the FAT: This file information is used by DOS / Windows.
2. It is stored in the sector 2 and in the file `FILEINFO.DAT`:
   This information is used for the file management on the *ADwin* system.

   In sector 2 (absolute) the start and end sectors are saved, in the file `FILEINFO.DAT` the current, relative write and read positions of the generated files.

The initialization does not physically change data on the storage medium. But data can be lost when the file size is changed.

If the contents of the storage medium is displayed by using a reading device (e.g. Explorer), all files are displayed with their file size (this is the information of the FAT).

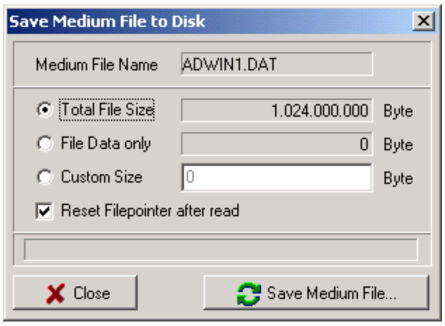
Name	Size	Type	Modified
ADWIN1.DAT	1.000.000KB	DAT File	24.04.03 14:17
ADWIN10.DAT	1.000KB	DAT File	24.04.03 14:17
ADWIN2.DAT	500.000KB	DAT File	24.04.03 14:17
ADWIN3.DAT	200.000KB	DAT File	24.04.03 14:17
ADWIN4.DAT	100.000KB	DAT File	24.04.03 14:17
ADWIN5.DAT	50.000KB	DAT File	24.04.03 14:17
ADWIN6.DAT	20.000KB	DAT File	24.04.03 14:17
ADWIN7.DAT	10.000KB	DAT File	24.04.03 14:17
ADWIN8.DAT	5.000KB	DAT File	24.04.03 14:17
ADWIN9.DAT	2.000KB	DAT File	24.04.03 14:17
FILEINFO.DAT	10KB	DAT File	24.04.03 14:17

From the PC to the storage medium

Transfer data between PC and storage medium

With the program `<Pro-Storage.exe>` data transfer is possible from the PC to the storage medium being inserted in a Pro-Storage module and vice versa.

The data of a storage medium file is stored with the button (`Save Medium File to Disk`) on the PC. After pressing the button the following dialog opens:



This dialog determines how many data of the file are copied to the PC.

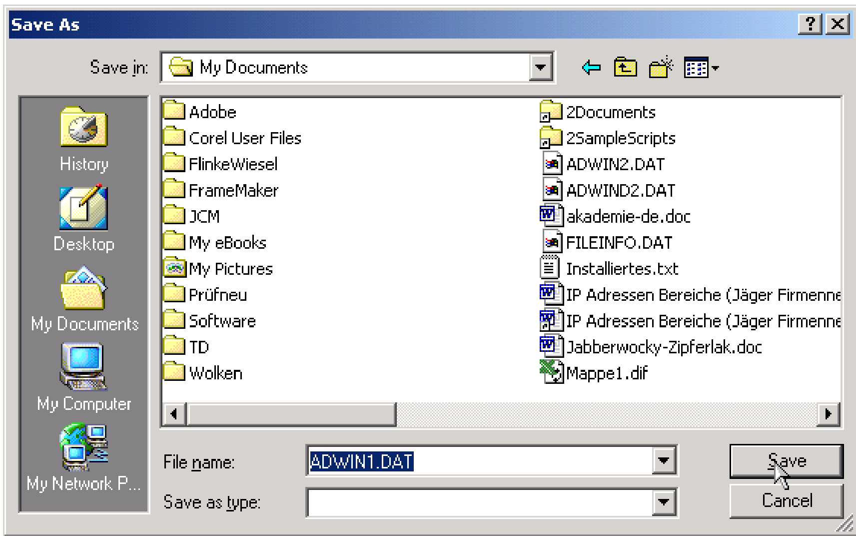
– `Total Filesize` saves the whole file (including the data sectors where nothing is written into).

– `File data only` saves only the data sector where data has been written into (the file is indicated by a pointer; see also chapter 7.3).

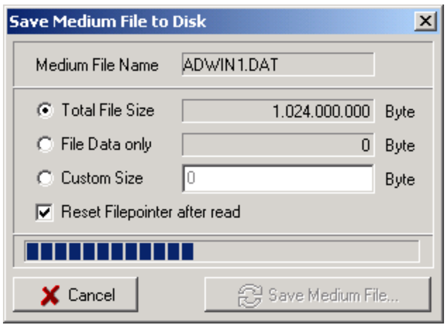
– `Custom Size` determines manually the amount of the bytes to be saved.

If the `Reset file pointer after read` option is activated, the write/read pointer of this file is set to the beginning after data transfer. The standard saving process (see below) works with this pointer.

When clicking the `Copy to Harddisk` menu item, a file saving dialog opens.



Enter a file name and confirm by clicking on `Save`. The data is stored into the selected file.

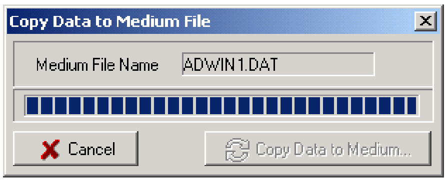


The bar in the dialog box shows the saving process. After saving the data, the dialog is automatically closed.
If you interrupt the data transfer with `Cancel`, you can restart it with `Save Medium File …` .

From the storage medium to the PC

With the button (`Copy Data to Medium File`) data are transferred from a source file into the corresponding file. After pressing the button a "file open"-dialog is opened.

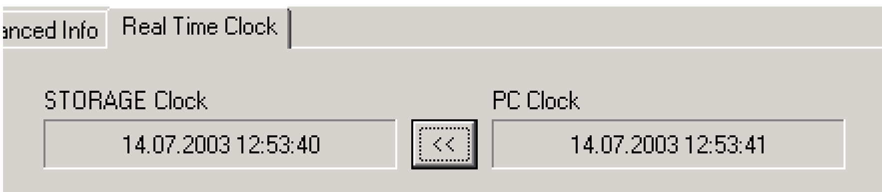
Select the source file in the dialog box, where the data is to be transferred. The amount of data must not be higher than the size of the destination file. Confirm the selection by clicking on `Open`. The dialog box closes and a bar in another window shows the status of the data transfer.



After writing the data, the dialog box is automatically closed. Due to the data transfer, the previous data in the destination file will be overwritten.
If you interrupt the data transfer with `Cancel`, you can restart it by clicking on `Copy Data to Medium …` .

Set the real-time clock

Select the tag "Real-Time Clock" at the top of the window. If you use the double-headed arrow in the middle, date and time of your PC are transferred to the real-time clock of the Pro-Storage module.



After data is transferred, the real-time clock continues running independently.

Use the storage medium

Measurement process + data process

In a measurement process you can use the storage medium in the *ADwin* system as provider of data or as data memory. The measurement process should not, if possible, access the storage medium itself, so that the process can be processed as usual and as fast as before. Therefore you additionally require a low-priority data process, which serves as "data messenger" between the storage medium and the high-priority measurement process.

The additional data process leads you to the following tasks:

– extend the measurement process

Insert into your measurement process (mp) the control of data process (dp) and the data transfer as additional tasks.

Use 2 global fields for information exchange between mp and dp:
- Field 1 for the data transfer
- Field 2 for the control of the data process

– create the data process

We provide a standard process as dp, which writes data to storage media. The text below describes how to adapt the standard saving process to your needs.

Generally, the dp is based on the simple data structure and management, which is already installed on the storage medium.

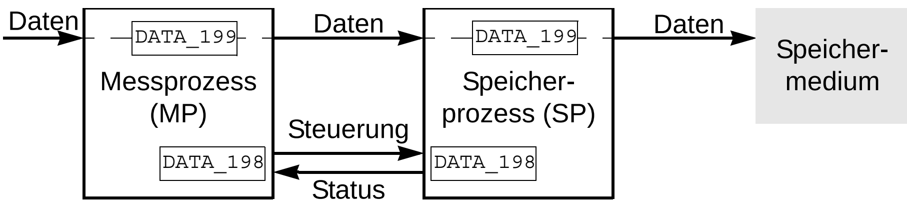
– adjust the timing of processes

The fact that 2 (or more) processes are running synchronously, requires that you coordinate the timing characteristics of these processes, so that the tasks of the MP and the data flow can run without any interruption. Preferably you will adapt the cycle times (globaldelay) of both processes to one another.

If you want develop an own write or read process, please pay attention to the rules of importance on page 132.

Standard saving process (SP)

The standard saving process (SP) gets data via a FIFO array (`DATA_199`) from a high-priority measurement process (MP) and writes it into a specified file of the storage medium. The MP controls functions of the SP using an additional global array (`DATA_198`), and vice versa receives status messages.



To use the saving process, proceed as follows:

1. Determine basic parameters

1. Determine the basic parameters:

   - Number (1…10) of the destination file.
     The SP works only with one of the 10 files. The SP determines the file information of the destination file only once during the start process.
   - Write mode "Append" or "New":
     Upon restart the SP may either overwrite the destination file or append the data to the end of file.

The SP writes the number of the saved values (write pointer) into the file `Fileinfo.dat` each time when saving a complete sector is saved. Upon restart the SP reads this write pointer and appends new files.

- Duration of the time-out:
  The time-out is the longest time interval, after which data from the FIFO array are buffered (backup copy).

  Normally the SP stores data, if it can write up to 128 values into one or more (12) sectors. But if the time-out is reached earlier, all data in the FIFO are saved immediately.
- Approximate size of the FIFO array for data transfer:
  The array must be large enough to buffer fluctuations of the data flow, e.g. when data arrives irregularly or if there is a delay when the data memory is accessed (for hard disks see page 122).
  A later adaptation of the process cycle times may require to change the array size.
- Data type (FLOAT or LONG) of the data to be saved.
  The SP can only be used for processing one of the two data types.

2. Which process starts?

2. Determine, which process is to start first (SP or MP). For both possibilities the pros and cons have to be considered:

Measurement process starts
+ **Flexible**: If the MP starts the SP, it can re-define the number of the destination file, the time-out and the write mode for the SP upon each restart. Thus the MP saves the data in several files. Therefore, if there are changes, the SP does not have to be compiled again.
– **Latency**: The MP must wait with the transfer of data into the FIFO array until the SP reports that it is ready to save the data. Alternatively the FIFO array can be dimensioned as buffer for this latency and already receive data. Normally the SP is interrupted by high-priority processes, so that the latency cannot exactly be determined before.
+ **Data loss preventable**: If the MP buffers more data in the FIFO array than the destination file can take, the surplus data must be written into another file, otherwise it is lost. For this the MP must stop the SP and restart it with a new destination file number.
Saving process starts
– **Programming effort**: If the number of the destination file, the time-out or the write mode are changed, the SP must be changed and re-compiled.
– **Possible data loss**: If the MP buffers more data in the FIFO array than the destination file can take, those data will be lost.
+ **No delay**: If the SP starts the MP, the MP can pass the data into the FIFO array without any delay. Also possible is to monitor the status of the SP via PC and then start the MP.
– **Start time not determinable**: The time when the SP starts the MP cannot exactly be determined.

We assume that both processes are already on the *ADwin* system, but have not yet been started. The process started first determines the basic parameters. Either the first process or the PC start the second process.

3. Adapt the source code

3. Copy the source code files `Pro-Storage_SP.bas` and `Pro-Storage_MP.bas` from the directory `C:\ADwin\ADbasic\samples_ADwin_PRO` into the directory of your project. Add the information you have selected according to the items 1 and 2 to your file. In the source code the areas are marked which must be adapted or moved to the MP.

   Note that the global arrays have the same size in both processes.

4. Test the program

4. Test the programs (compile SP with low-priority!). Note, how the MP uses the 2 global arrays for data transfer and the control of the SP:

– `DATA_199[]` or `fb[]`: This FIFO array is used for passing the saved data from MP to SP.

– `DATA_198[]` or `f_cmd[]`: The elements of this array are used for the control of the saving process and as feedback for the MP.

- `f_cmd[1]` Number (1…10) of the file
- `f_cmd[2]` Write mode in the SP:
  0: Write data starting at the beginning of the file ("New" means to overwrite previous data).
  1: Append data at the end of the data ("Append").
- `f_cmd[3]` Duration of the time-out in the SP:
  ≤0:Write data immediately.
  >0:Time interval in 100µs until buffering starts.
- `f_cmd[4]` Total size of the file in 32-bit values.
- `f_cmd[5]` Status of the SP: Amount of data values, which have already been written into the file.
- `f_cmd[6]` Status of the SP: Amount of free elements in the FIFO `DATA_199[]`.
- `f_cmd[7]` Status or error messages of the SP:
  128: Error - storage medium does not respond in a defined time interval (time-out).
  64: Error - end of file is reached during writing into the file.
  32: Status - SP is stopped as soon as possible (see `f_cmd[9]`)
  16: Error - Start sector of the file is larger than the end sector.
  8: Error - Write pointer of the file is invalid.
  4: Error - File is full (before first saving).
  2: Error - File does not exist.
  1: Error - No storage medium in the module.
  0: Status - End of file not yet reached = data can be saved.
- `f_cmd[8]` Status of the SP:
  0: Section **`INIT:`** is finished.
  1: Section **`INIT:`** just in progress.
- `f_cmd[9]` Command of the MP to the SP:
  0: Sp continues working.
  1: Save remaining data and stop SP, for feedback see `f_cmd[7]`.

If the MP setting the parameter `f_cmd[9]` the SP is not stopped at once, but at the moment when there is no data left in the FIFO array. Thus, the SP continues working as long as the MP writes data into the FIFO array.

The SP stops automatically when an error occurs. The cause for the error is reported to the MP using `f_cmd[7`.

5. Complete the measurement process

5. Integrate the standard measurement process `Pro-Storage_MP.bas` in your individual measurement process. Sometimes it is necessary to consider security queries regarding a FIFO overflow.

   Adapt afterwards the cycle time of the MP to the cycle time of the SP (`Globaldelay`). It may be necessary to change the size of the data FIFO `DATA_199[]` additionally to the cycle time, to get the necessary results.

   The SP should be configured with a high cycle time, so that a sufficient memory rate is guaranteed. Then the cycle time of the MP (and of other processes) can individually be set to obtain a processor workload less than 100%.

   If the cycle time of the SP is short so that calling the SP requires a longer period of time than processing it, an endless loop will be the result.

Use the storage medium

Ths standard saving process is now integrated in your measurement program.

Individual data process

The low-priority process described on the pages above is a standard example for fast data storage. Even if this process does not fully meet your requirements we recommend that you first improve your practical knowledge before developing your own individual data process.

Examples

For your own application several instructions are available to write or read data sector by sector to/from the storage medium. With an individual process the following tasks can be executed:

- To work simultaneously with several files on one storage medium.
- To save a specified amount of data and to read it again later, e.g. parameters of a test stand for an initialization after restart.
- To save data with a time stamp, that is with date and time of the module-inherent real-time clock.
- To access sectors of the storage medium. In extreme cases data are nevertheless written ore read, independent of the initialized data structures. In this case it is not possible to access data via a PC reading device.

Important: Compared to the standard saving process a user-defined process with the instructions described above, is necessarily slower, because the instructions include additional test routines. These instructions are principally used for the non-time-critical exchange of data of a specific length.
If you do prefer a data process with faster access times, call our support hotline.

Rules of importance

The following rules have to be definitely kept when developing an individual data process for the Pro-Storage module:

- The data process must have low-priority.
  If the instructions are used in high-priority processes errors occur, or even data loss, when working with the storage medium.
- Only one process has access to each Pro-Storage module. Otherwise data will be lost.
- In order to work with files the data structure on the storage medium should not be changed, that is all sectors of a file are located one after the other (= no fragmentation).
- During data transfer the smallest data block has the size of a sector, these are 128 values.
- In a sector only values of the same data type can be used (LONG or FLOAT).
- The information about the data type of a value (or the values in a sector) cannot be saved, and must if necessary be saved separately. In a file different data types could be used in the sectors, but it is not reasonable, as we have described above.
- As long as the write/read process accesses the storage medium, it is not allowed to remove it. Otherwise data of the running process will be lost on the storage medium (by overwriting them).

For data exchange between data process and measurement process as well as for the exchange of control and status information, we recommend that you use a programm design similar to the standard saving process.

4.7 Signal Conditioning and Interface Modules

Signal conditioning modules

Please note any signal conditioning module can only be used with an analog input module (Pro-AIn-8/12, Pro-AIn-8/14 or Pro-AIn-8/16).

Module	TC-4	TC-8	TC-16
Revision	A	A	A
Type	thermocouple amplifier		
Type	J: 0°C…750°C K: -200°C…950°C		
Accuracy in bits	12		
Channels	4	8	16
Page	135		

Module	PT100-4	PT100-8
Revision	A	A
Type	RTD amplifier	
Version	2, 3 or 4 wires	
Temperature range	-50°C…250°C (other ranges on request)	
Accuracy	±0.2°C	
Channels	4	8
Page	135	

Module	LPSH-4-FI	LPSH-8-FI
Revision	A	A
Type	Filter isolation amplifier	
Filter	low pass 4^{th} order	
Cut-off frequency	fixed (options on request)	
Channels	4	8
Page	135	

Module	MB-8 (-D)
Revision	A
Function	Passive carrier module for insertion of input modules of the type 5B oder MB.
Page	135

Fig. 235 – Overview signal conditioning modules

Interface modules

Module	CAN-1	CANL-1	CAN-2	CANL-2
Revision	A	A	A	A
Type	CAN interface			
CAN-Version	High speed	Low speed	High speed	Low speed
Interfaces	1		2	
Page	135			

Module	PROFI-DP-SL	Inter-SL
Revision	A	A
Type	Fieldbus interface	
Fieldbus version	Profibus	Interbus
Size of DP-RAM	2kB	
Data exchange rate	9.6kBit/s … 12MBit/s	500kBit/s
Page	135	

Module	RS-232-2	RS-485-2	RS-232-4	RS-485-4
Revision	A	A	A	A
Type	RSxxx interface			
RSxxx version	RS232	RS485	RS232	RS485
Interfaces	2		4	
Data exchange rate [kBaud]	0.035… 115.2	0.035… 2304	0.035… 115,2	0.035… 2304
Page	135			

Abb. 236 – Overview interface modules

4.7.1 Pro-TC-4 REVA, Pro-TC-8 REVA, Pro-TC-16 REVA

With the modules Pro-TC-xx REVA you have purchased thermocouple amplifiers, including cold junction compensation, with 4 (Pro-TC-4), 8 (Pro-TC-8), or 16 (Pro-TC-16) channels.

The amplifier outputs are connected to a LEMO-socket via multiplexer. The output must be connected to an additional analog input module. The multiplexer can be set by an ***ADbasic*** instruction.

Pro-TC-4

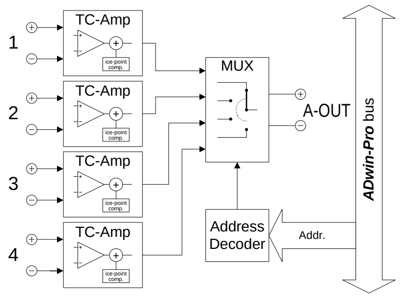


Fig. 237 – Pro-TC-4 REVA: Block diagram

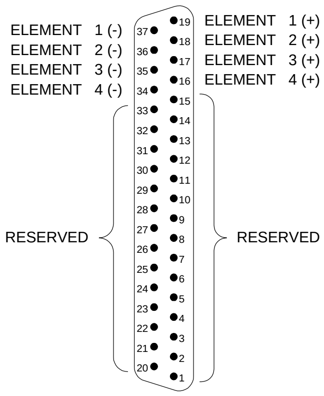


Fig. 238 – Pro-TC-4-x-D REVA: Pin assignment differential

Pro-TC-8

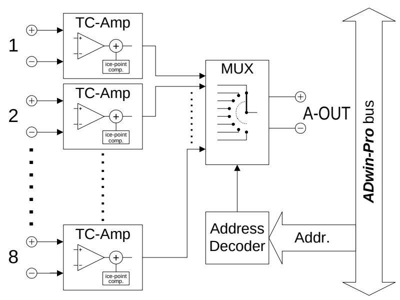


Fig. 239 – Pro-TC-8 REVA: Block diagram

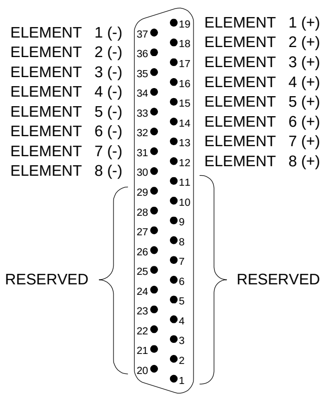


Fig. 240 – Pro-TC-8-x-D REVA: Pin assignment differential

Pro-TC-16

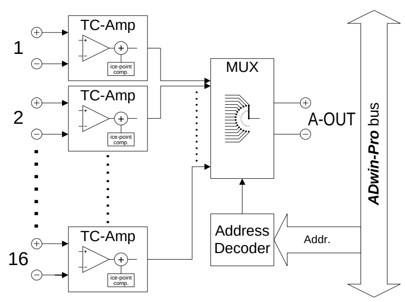


Fig. 241 – Pro-TC-16 REVA: Block diagram

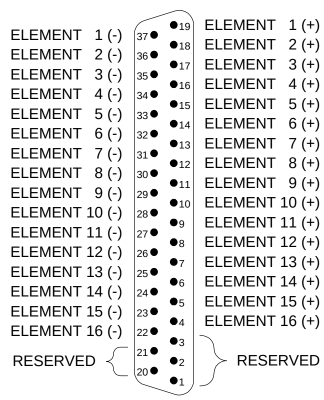


Fig. 242 – Pro-TC-4-16-D REVA: Pin assignment differential

Input channels	Pro-TC-4	4
	Pro-TC-8	8
	Pro-TC-16	16
Multiplexer settling time		50µs
Type, measurement range		J: 0°C…750°C K: -200°C…950°C
Output voltage range		±10V to LEMO socket A-OUT
Accuracy		±1°C
Connector	Pro-TC-4, Pro-TC-8	Omega subminiature connector, type: SMP-K-F optional: SMTC-37F, 37-pin DSub socket
	Pro-TC-16	Omega subminiature connector, type: SMTC-37F, 37-pin DSub socket

Fig. 243 – Pro-TC-x REVA: Specification

A conversion table is available for the conversion of the temperature values into the corresponding integer/float values.
After the installation of the ***ADwin***-CDROM you will find the conversion table in the ***ADbasic*** online help, topic "hardware information".

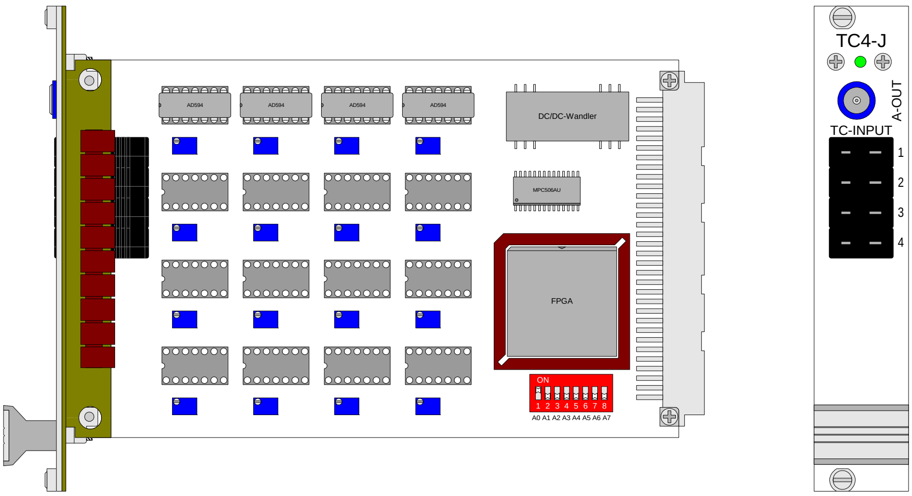


Fig. 244 – Pro-TC-4-J REVA: Board and front panel

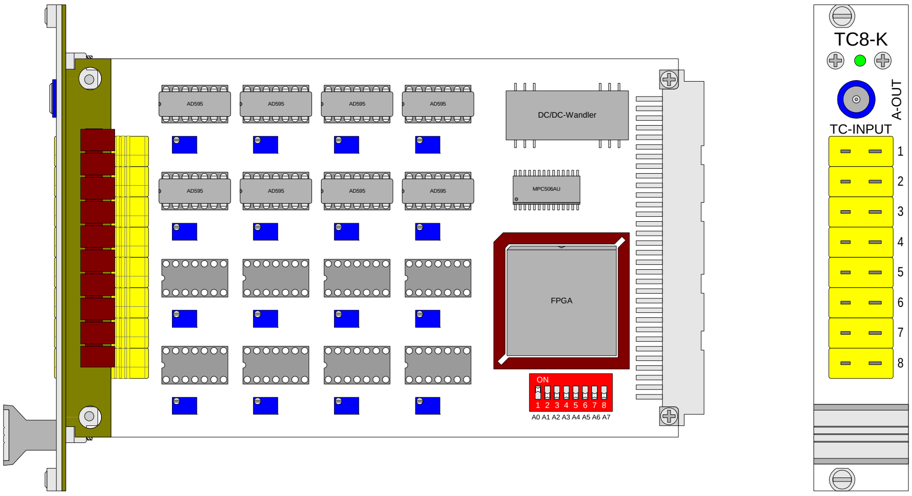


Fig. 245 – Pro-TC-8-K REVA: Board and front panel

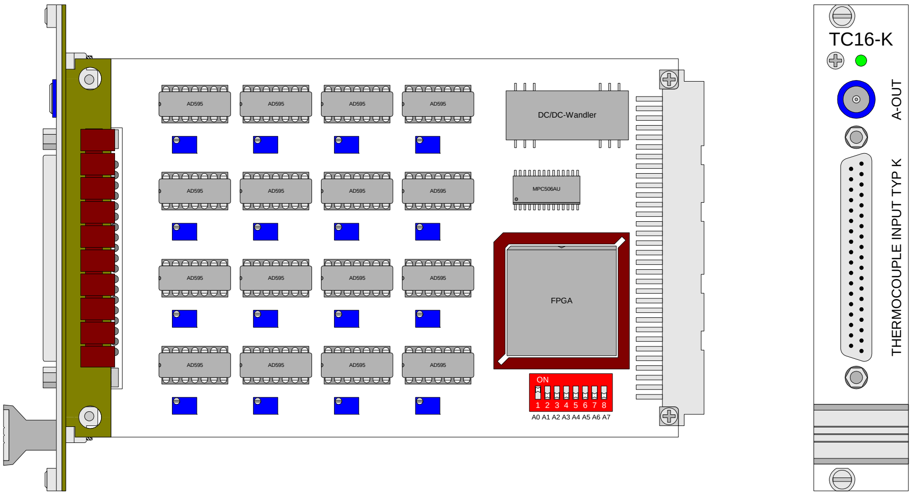


Fig. 246 – Pro-TC-16-K-D REVA: Board and front panel

4.7.2 Pro-PT100-4 REVA, Pro-PT100-8 REVA

The module Pro-PT100 has 4 or 8 inputs for connecting platinum temperature sensors of the type Pt 100. The maximum possible measurement range is -200°C…+266°C, depending on the temperature sensor (see the data sheets of the manufacturers, e.g. Betatherm, Ephy-Mess, Heraeus, Jomo, Omega, Sensycon, etc.).

The amplifier outputs are connected to a LEMO-socket via multiplexer. The output must be connected to an additional analog input module. The multiplexer can be set by an ***ADbasic*** instruction.

Measurements are done with 2, 3 or 4 wire technique (input circuit see fig. 247). Measuring method, zero point and gain are set via jumpers and trimmers on the circuit board (page 142).
The measuring methods and the wiring between senso and Pro-PT100 module is described on page 141.

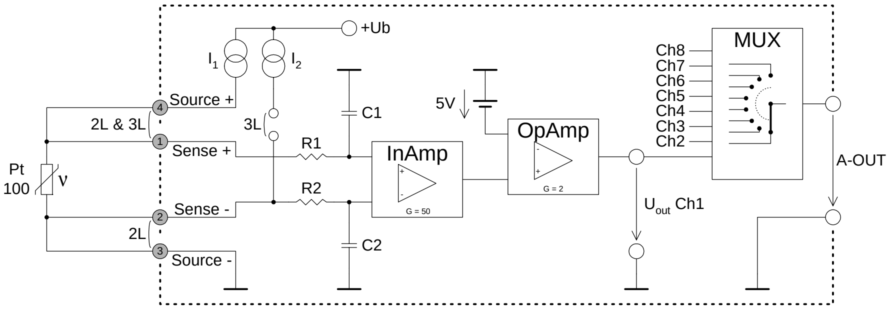


Fig. 247 – Pro-PT100-x REVA: Block diagram

Inputs		4 or 8
Multiplexer settling time		15µs
Max. measurement range		-200°C…+266°C
Output voltage range		±10V to LEMO socket (A-OUT)
$I_1 = I_2$		1mA
Connector		Omega subminiature connector, type: SMP-K-F optional: 37-pin DSub socket
Module width	Pro-PT100-4	5HP wide / 1 slot
	Pro-PT100-8-D	5HP wide / 1 slot
	Pro-PT100-8	10HP wide / 2 slots

Fig. 248 – Pro-PT100-x: Specification

A conversion table is available for the conversion of the temperature values into the corresponding integer values. You will find the conversion table in the ***ADbasic*** online help, topic "hardware information".

Pro-PT100-8

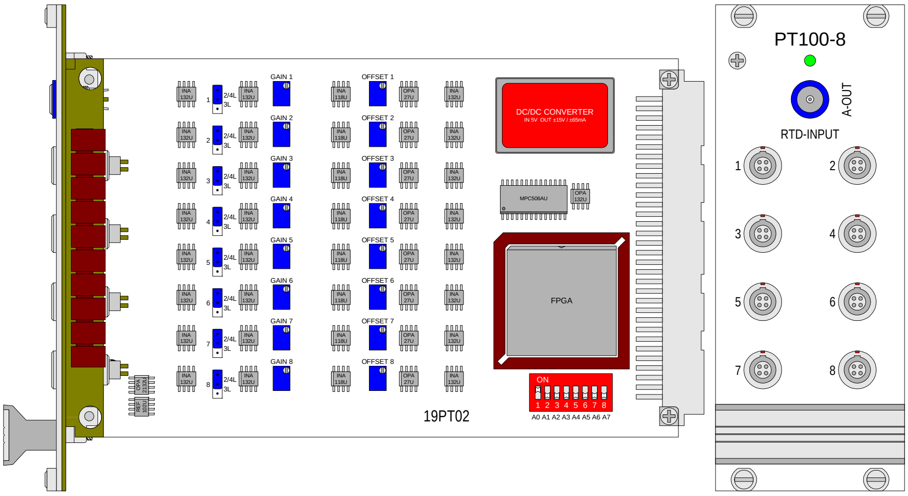


Fig. 249 – Pro-PT100-8 REVA: Board and front panel

Pro-PT100-4

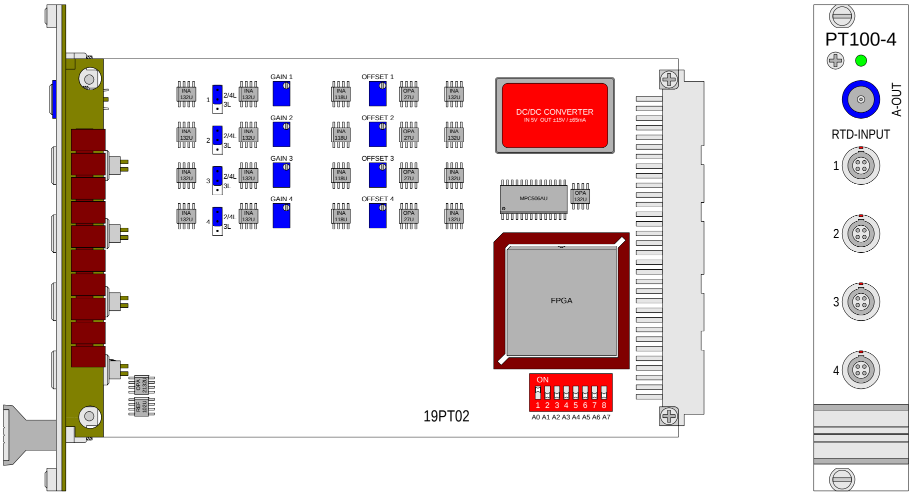


Fig. 250 – Pro-PT100-4 REVA: Board and front panel

Pro-PT100-8-D

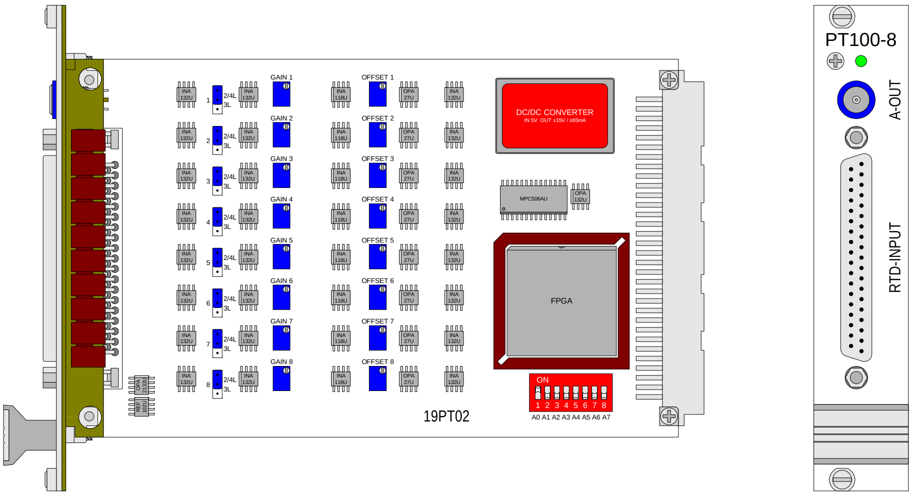


Fig. 251 – Pro-PT100-8-D REVA: Board and front panel

Pro-PT100-4-D

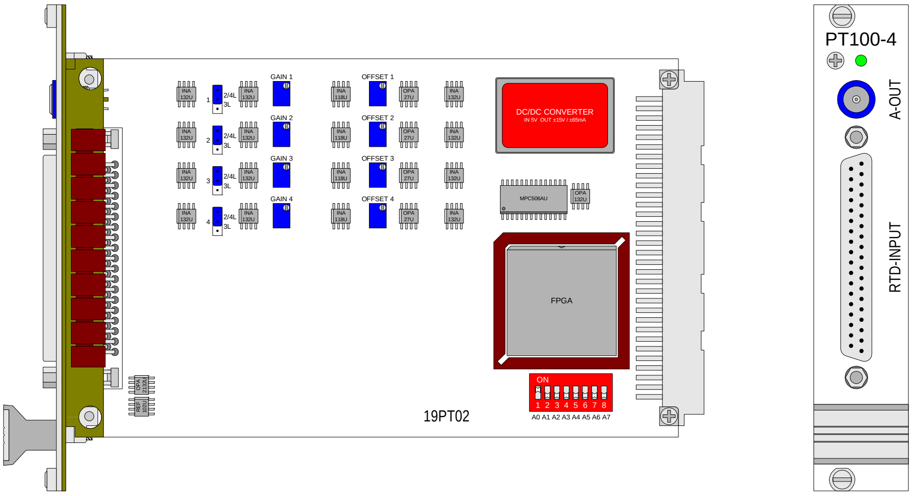


Fig. 252 – Pro-PT100-4-D REVA: Board and front panel

Pin assignments

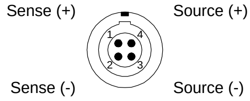


Fig. 253 – Pro-PT100-x: LEMO socket

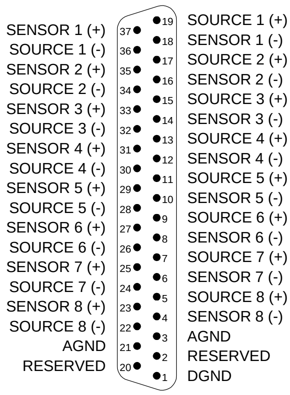


Fig. 254 – Pro-PT100-8-D: Pin assignment

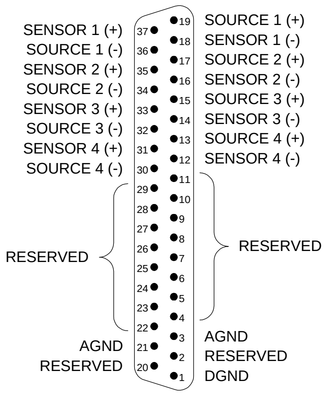


Fig. 255 – Pro-PT100-4-D: Pin assignment

Measurement Method

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

– 2 wire measurement

2 wire

Please pay attention to a very short connection with low impedance between the Pt 100 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.

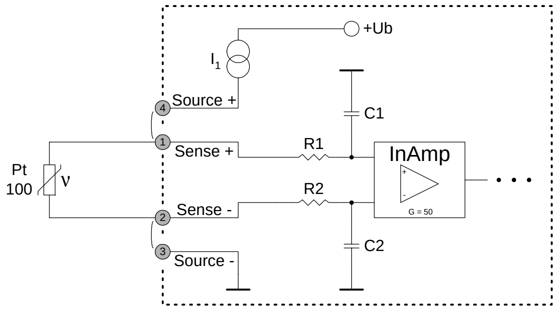


Fig. 256 – Pro-PT100-x: 2 wire measurement

For a 2 wire measurement the following connections have to be made:
- LEMO connector: Connect "source+", pin 4 with "sensor+", pin 1.
- LEMO connector: Connect "source -", pin 3 with "sensor -", pin 2.
- Set the jumper on the PCB to the position "2/4L".

– 3 wire measurement

3 wire

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 100 to the module input should be identical.

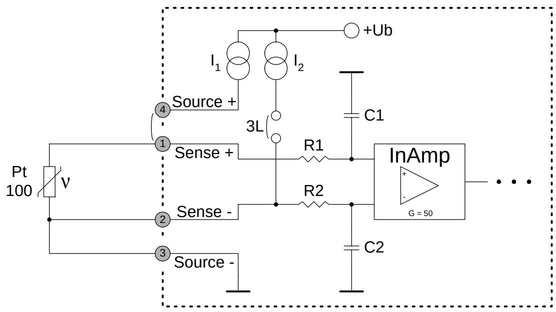


Fig. 257 – Pro-PT100-x: 3 wire measurement

For a 3 wire measurement the following connections have to be made:
- LEMO connector: Connect "source +", pin 4 with "sensor +", pin 1.
- Set the jumper on the PCB to the position "3L", in order to activate the second voltage source.

4 wire

– 4 wire measurement

The voltage drop at the Pt 100 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.

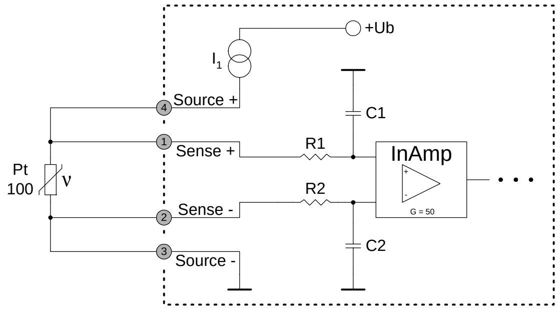


Fig. 258 – Pro-PT100-x: 4 wire measurement

For a 4 wire measurement the following connections have to be made:

- LEMO connector: Using this method no connections have to be made here.
- Set the jumper on the PCB to the position "2/4L".

Setting Zero Point and Gain

The figure 259 below shows the schematic of the printed circuit board and where you can set measuring method, zero point and gain.

With the jumpers 1 to 8 the selected measurement method can be set:

– Upper position "2/4L": 2 or 4 wire measurement

– Lower position "3L": 3 wire measurement

The zero point (at 0°C) is set with the trimmers "OFFSET 1" to "OFFSET 8", the scale factor or gain with "GAIN 1" to "GAIN 8".

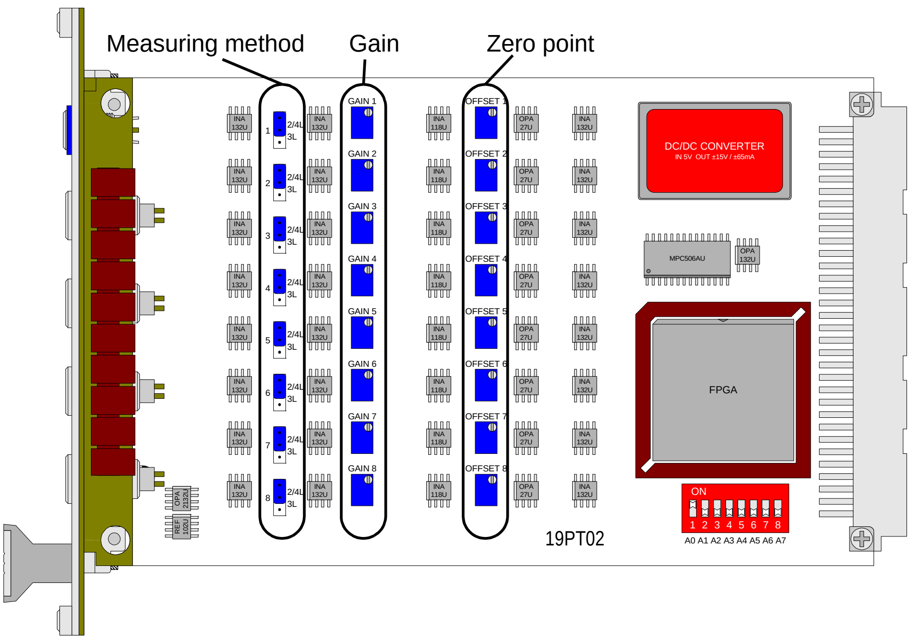


Fig. 259 – Pro-PT100-x: Position of jumpers and potentiometers

4.7.3 Pro-LPSH-4-FI REVA, Pro-LPSH-8-FI REVA

The module Pro-LPSH-4-FI REVA has 4 low-pass filters of 4th order with Sample & Hold and isolation amplifiers, the module Pro-LPSH-8-FI REVA has 8 of these low-pass filters. The filters are Butterworth filters with a fixed cut-off frequency. The frequency has to be indicated when you place an order. The inputs are all optically isolated from system circuitry and from each other.

A low-pass filter module has to be used in combination with an analog input module. (Pro-AIn-8/12 or Pro-AIn-8/16). A low-pass filter module and an analog input module are then forming one unity, which is 2 inches (10 HP) wide and therefore needs two slots.

The switching from sample to hold mode has to be made by the instruction `SH_SETMODE(module,mode)`.

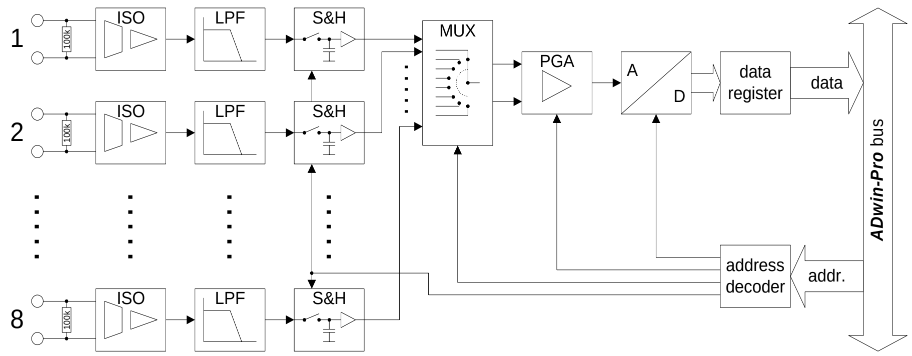


Fig. 260 – Pro-LPSH-8-FI REVA in combination with Pro-AIN-8/12 REVA

Input channels	4 at Pro-LPSH-4-FI REVA, isolated 8 bei Pro-LPSH-8-FI REVA, isolated
Input voltage range	±10V
Isolation voltage	1kV
Offset drift	40 ppm/°C
Non-linearity	0.016%
Input resistance	100kΩ
Filter	Butterworth 4^{th} order
Cut-off frequency	5kHz, 10kHz, 20kHz (other frequencies on request)
Cut-off frequency error	±5%
Sample & Hold drift	1.5mV/s
Linearity	±1 LSB (12 bit)
Connector	4 / 8 LEMO sockets (optional: 37-pin DSub socket)

Fig. 261 – Pro-LPSH-4/8-FI REVA: Specification

4.7.4 Carrier module Pro-MB-8 (-D)

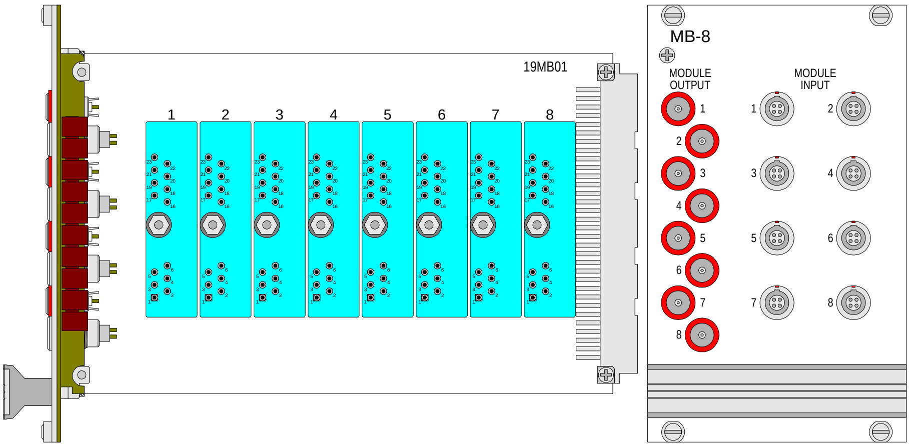


Fig. 262 – Pro-MB-8: Board and front panel

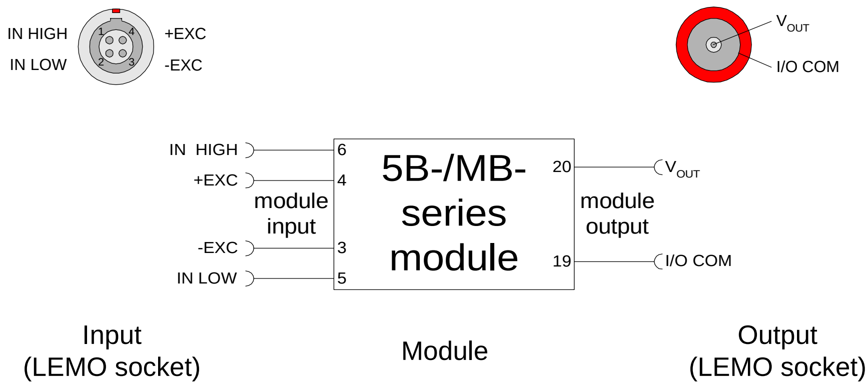


Abb. 263 – Pro-MB-8: Pin assignment input, module and output

The module Pro-MB-8 is a passive carrier module providing up to 8 slots for insertion of 5B input modules (Analog Devices or Burr Brown) or MB input modules (Keithley). The power supply (pin 17, V_{CC} = +5V; pin 16, GND = 0V) of the 5B or MB module is directly connected with the back plane PC bus of the ***ADwin-Pro*** system. Pin 22 (READ EN) und 23 (WRITE EN) are connected with GND. Therefore the output of the 5B or MB modules is always enabled. Pins 18 and 20 are connected.

The module Pro-MB-8 has LEMO connectors, the module Pro-MB-8-D has DSub connectors.

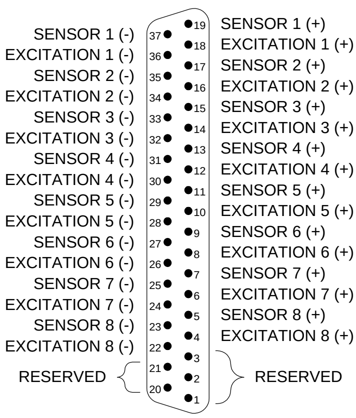


Fig. 264 – Pro-MB-8-D: Pin assignment IN

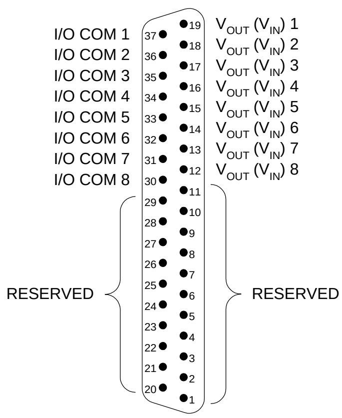


Fig. 265 – Pro-MB-8-D: Pin assignment OUT

4.7.5 Pro-CAN-x Rev. A

The module Pro-CAN has 1 or 2 CAN interfaces, a high speed or a low speed version. The names for the module versions are shown in the table below:

	High speed	Low speed
1 CAN interface	Pro-CAN-1	Pro-CAN-1-LS
2 CAN interfaces	Pro-CAN-2	Pro-CAN-2-LS

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 of the DIO1 add-on 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 255 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. 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.

The manual is divided into the following sections:

– Hardware design
– Message Management
– Setting the bus frequency
– Interrupt / Event
– Module revisions
– Programming

Hardware design

1 CAN interface

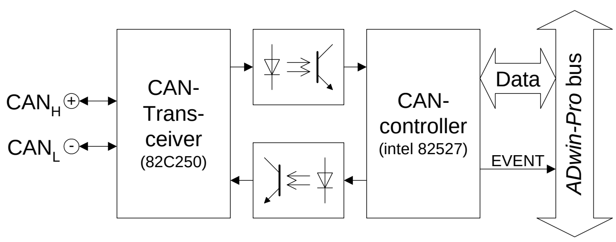


Fig. 266 – Pro-CAN-1: Block diagram for 1 interface

2 CAN interfaces

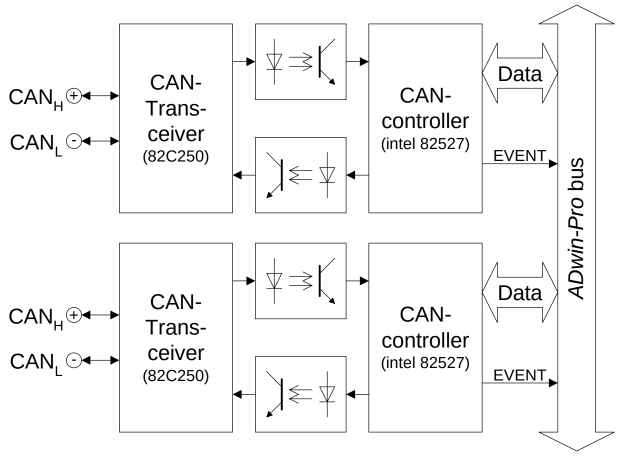


Fig. 267 – Pro-CAN-2: Block diagram for 2 interfaces

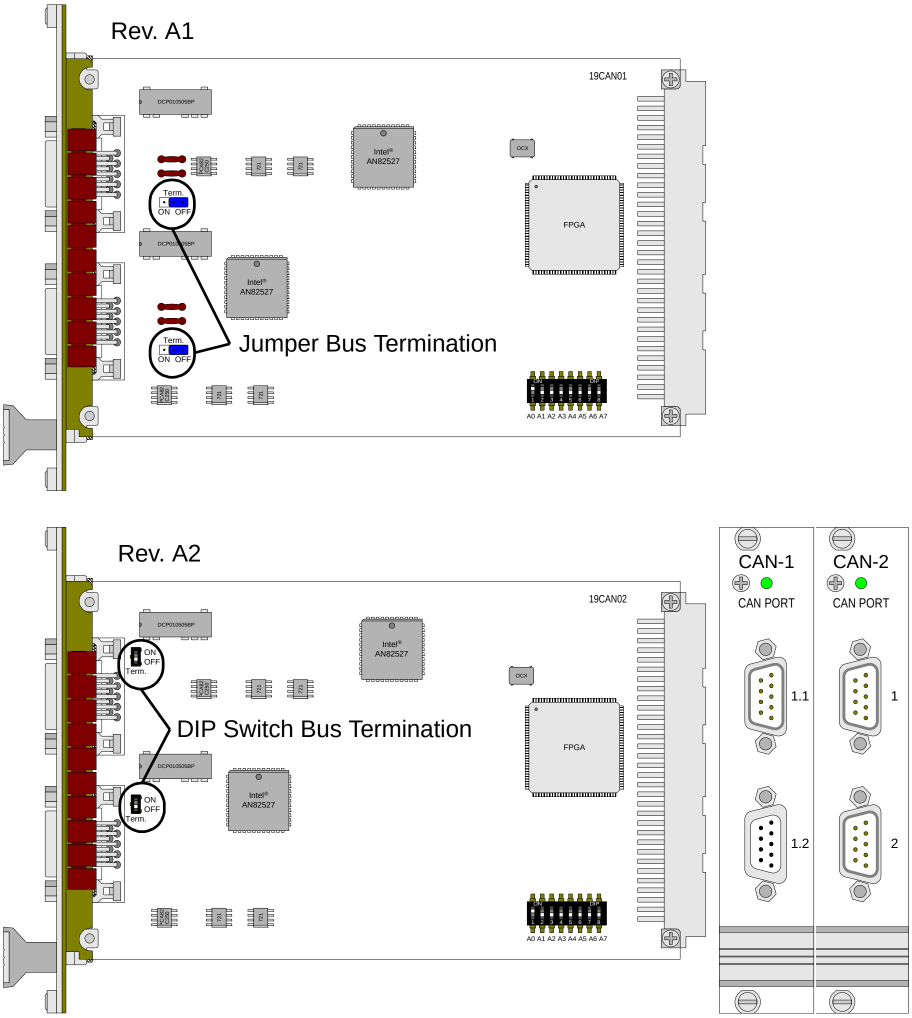


Fig. 268 – Pro-CAN-1/-2: PCB and front panels

The connections of the CAN bus interface are on the 9-pin D-SUB connector; the pin assignment is shown below. On the CAN-1 and CAN-1-LS modules both D-SUB connectors are internally connected with each other.

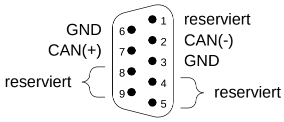
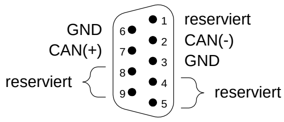
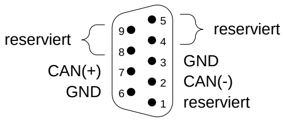
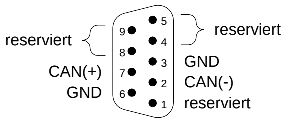
CAN-1, CAN-1-LS	CAN-2, CAN-2-LS
	
Connection 1.1 (male connector)	Connection 1 (male connector)
	
Connection 1.2 (female connector)	Connection 2 (male connector)

Fig. 269 – Pro-CAN: Pin assignment

Bus termination (High speed only)

If the CAN module functions as the physical termination of a high-speed CAN bus, the terminator must be a resistor (only the first or the last CAN node). If a termination is necessary, move the DIP switch (Rev. A2, see fig. 268) upward; when using the module revision A1 set the jumper to the left position (ON). 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 routine configures the controller automatically so that the CAN bus frequency is 1 MHz. If the CAN bus is to operate with another frequency, the values in the "Bit Timing Register 0 (BTR0, address 3Fh) and in the "Bit Timing Register 1" (BTR1, address 4Fh) have to be changed. Just use the instruction **SET_CAN_BAUDRATE** for setting a large quantity of bus frequencies.

Special cases

In some special cases it may be better to select configurations other than those set with the instruction mentioned above. For this purpose specified registers have to be set with the instruction `POKE`. The structure of the register is described below.

	Bit Timing Register 0 (BTR0)		Bit Timing Register 1 (BTR1)		
Bits	7…6	5…0	7	6…4	3…0
Sub-Reg.	SJW	BRP	SPL	TSEG2	TSEG1

The following table shows the admitted values and the meaning of the individual ranges:

Range	Admitted values	Meaning
SJW	0 … 3	Max. pulse elongation during bus synchronization
BRP	0 … 63	Pre-scaler
SPL	0 … 1	Sampling mode
TSEG1	2 … 15	Time segments before sampling
TSEG2	1 … 7	Time segments after sampling

The default setting of the ranges SJW and SPL is 0 and should only be changed if necessary. Select the sample point (specified by TSEG1 and TSEG2) in such a way that it is between 50% and 80% of the total bit length.

The CAN bus frequency is calculated as follows:

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

The following table illustrates all common settings for the Baud rates.

Baud rate [kBit/s]	125	250	500	1.000
BRP	3	1	0	0
TSEG1	6	6	6	2
TSEG2	7	7	7	3
BTR0	`03h`	`01h`	`00h`	`00h`
BTR1	`76h`	`76h`	`76h`	`32h`
Sample point[%]	54	54	54	60

Fig. 270 – CAN: Setting the Baud rates

Access to the two timing registers is only possible, when the access has been enabled before. This is done by the CCE-bit in the control register. The bit has to be reset afterwards.

Interrupt / 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 is 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 Pro-CAN is easily programmed with ***ADbasic*** instructions.

The instructions for the following ranges are in the include file `<ADPEXT.INC>`:

Range	Instructions
Initialization of the CAN controller	**INIT_CAN**
Setting and reading of registers	**SET_REG**, **GET_REG**
Initialization of message objects	**EN_RECEIVE**, **EN_TRANSMIT**
Sending and receiving of data sets	**TRANSMIT**, **READ_MSG**
Enabling interrupts	**EN_INTERRUPT**
Setting the Baud rate	**SET_CAN_BAUDRATE**

The instructions are described in the Pro software manual or the online help.

4.7.6 Pro-Fieldbus Modules

The fieldbus modules Pro-PROFI-DP-SL Rev. A and Pro-Inter-SL Rev. A provide a fieldbus interface with the functionality of a "slave". The modules have a DP-RAM (Dual Port - Random Access Memory) with a size of 2kB. For the user the communication is defined as access to this DP-RAM, the bus-specific data exchange is realized with hardware. Therefore the communication is more or less independent of the fieldbus type.

In the following text the characteristics of the fieldbus modules is described. Later the special features of the fieldbusses are explained. The description is divided into the following paragraphs:

– Functions description of the fieldbus modules
– Data exchange by handshake
– Programming
– Specifications
– Pro-PROFI-DP-SL Rev. A
– Pro-Inter-SL Rev. A

Functions description of the fieldbus modules

Initialization

After switching on the Pro system the fieldbus interface must be initialized. The module must not be accessed before the initialization. The initialization determines the size of the input and output areas and the behavior of the module. A second initialization is not possible. If the interface is not correctly parameterized the Pro system must be switched on and off.

DP-RAM

Each module has a DP-RAM (Dual Port - Random Access Memory), which transfers data between the fieldbus and the program. Both, the program and the fieldbus have access to this memory. The memory is divided into 6 large areas and has a total size of 2kB. The table shows the areas of the memory. Please take into account that the terms "input" and "output" are used as the fieldbus controller sees them.

Address Area	Content / Function
`000h - 1FFh`	Data input (of the fieldbus)
`200h - 3FFh`	Data output (of the fieldbus)
`400h - 51Fh`	Mailbox input (of the fieldbus)
`520h - 63Fh`	Mailbox output (of the fieldbus)
`640h - 7BFh`	Fieldbus-specific data
`7C0h - 7FFh`	Control register

Fig. 271 – Pro-Fieldbus-SL: Areas of the DP-RAM

Data input / Data output

In this address area data for cyclic and acyclic data exchange is saved. The size of both areas is determined in the initialization phase. The data for the cyclic data exchange is located at the beginning of each area, the data for the acyclic data exchange follows directly. If the maximum memory size of 512 bytes is not obtained, the remaining area is not used.

Mailbox input and output

The mailbox area is used for initialization of the fieldbus module. The mailbox area is the interface to the bus-specific part of the module. The initialization of the module is made using the instruction `INIT_SLAVE`, so that the user does not have to pay attention to how the area functions.

Control register

The control area consists of two registers, which enable the handshake to access the DP-RAM, and of registers, from which information about the module and its configuration can be read out. The content of the individual areas are shown in the table below. Important: Only experienced users may write values directly into the control area.

Area	Size (Byte)	Meaning
7C0h-7C1h	2	Version number of the bootloader
7C6h-7C9h	4	Serial number
7CAh-7CBh	2	Manufacturer
7CCh-7CDh	2	Identification of the field bus type: 0001h: Profibus 0010h: Interbus
7CEh-7CFh	2	Version number of the software
7D4h-7D5h	2	Watchdog counter (counter increments every ms)
7DAh-7DFh	6	Status of the LED, meaning depends on the field-bus: 1. Byte: LED bottom left 2. Byte: LED top left 3. Byte: LED top right 4. Byte: LED bottom right
7E0h-7E1h	2	Module type: 0101h = Slave
7E2h	1	Bit 0: Status of the inputs when the program stops: Bit = 0: Set inputs to 0. Bit = 1: Freeze inputs Bit 1: Status message: Changed output data Bit = 0: Message is disabled Bit = 1: Message is enabled (see 7E4H-7E5H)
7E3h	1	Bit 0 = 0: Bus is offline Bit 0 = 1: Bus is online Bit 1 = 0: Clear outputs, when bus is offline Bit 1 = 1: Freeze outputs when bus is offline
7E4h-7EBh	8	The bits in this area show if the data in the output area have changed. Each bit stands for 8 data bytes in the output area.
7EDh	1	Interrupt source
7EEh	1	Released interrupts
7F0h-7F1h	2	Size of the input area for cyclic data transfer (in bytes)
7F2h-7F3h	2	Size of the total input area in the DP-RAM (in bytes)
7F4h-7F5h	2	Size of the total input area (in bytes)
7F6h-7F7h	2	Size of the output area for cyclic data transfer (in bytes)
7F8h-7F9h	2	Size of the total output area in the DP-RAM (in bytes)
7FAh-7FBh	2	Size of the total output area (in bytes)
7FEh-7FFh	2	Handshake register

Fig. 272 – Pro-Fieldbus-SL: Control register

Data exchange by handshake

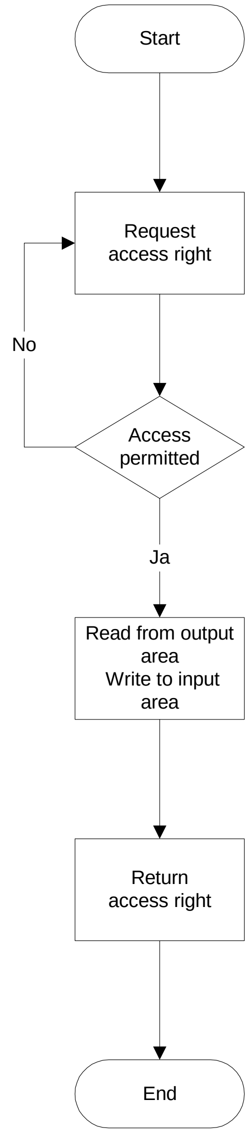
When the DP-RAM is accessed it must be assured that only one side has access to the memory. In order to ensure that this happens, there is a handshake process between the two sides fieldbus and program.

There is a separate handshake each for the input, output and control areas. Therefore, both sides may have access to the memory simultaneously, without disabling access for the other side.

After initializing the module and starting of the cyclic data transfer of the fieldbus, first the bus side has the right to access the DP-RAM.

The data, sent by the fieldbus master to the ***ADwin*** system (slave in the bus), is transferred by the fieldbus and is received and processed by the bus-specific electronic equipment of the ***ADwin*** module. The active data is written into the output area of the DP-RAM.

The program accesses the DP-RAM or parts of it at any time, (see figure at right). As soon as the fieldbus permits the access, the user reads out the active data from the output area of the DP-RAM and writes data into the input area of the DP-RAM.



As long as the user side has the access right, the fieldbus cannot access the DP-RAM. The Pro module continues to execute its tasks as fieldbus slave, that is the slave meets all requirements of the bus master. Data arriving from the fieldbus is buffered in the bus-specific electronic equipment of the module.

If the user has exchanged all data with the DP-RAM, he has to return the access right to the bus side. Now the electronic equipment of the module writes the buffered data into the output area of the DP-RAM and gets data from the input area of the DP-RAM. At the next bus cycle the bus master gets the input data and writes data into the output area.

fig. 273 explains the data flow in the fieldbus module.

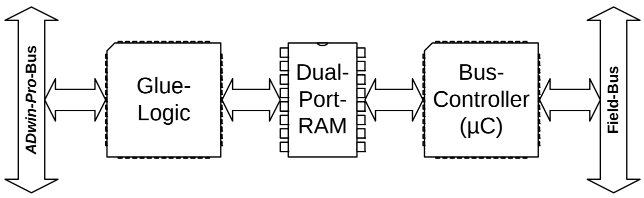


Fig. 273 – Pro-Fieldbus-SL: Data flow in the module

Programming

All fieldbus modules are easily programmed with the ***ADbasic*** instructions. Therefore the programs can be used for various fieldbus modules, without having to change them.
The following instructions can be found in the include file `<ADPEXT.INC>` :

Area	Instructions
Initialization	INIT_SLAVE
Access right to the DP-RAM (handshake)	CHECK_ACCESS REQUEST_ACCESS REQUEST_RELEASE_ACCESS
Writing to and reading out the memory areas	CHANGED_DATA GET_PRO_BYTE SET_PRO_BYTE GET_READ_BUFFER SET_WRITE_BUFFER

The instructions are described in the Pro Software manual or the online help.

Initialization

Initializing the fieldbus module

This program initializes a fieldbus module in the **LOWINIT:** section (see also page 153, Initialization). The initialization must be a low-priority process, because it takes some seconds; if it is a process with high priority, the PC interrupts the communication after a time (time-out).

```
#INCLUDE adwpext.inc
DIM adr AS LONG

LOWINIT:
adr = 1
REM Initialization of the anybus-module
par_1 = INIT_SLAVE(adr,16,0,32,0,2,2,0)
```

After initialization the module has the following parameters:

– 16 bytes input data in the cyclic data exchange
– 0 byte input data in acyclic data exchange
– 32 bytes output data in the cyclic data exchange
– 0 byte output data in the acyclic data exchange
– CHANGED_DATA function is enabled
– Outputs are frozen at bus OFF
– No interrupt

For more information see the Pro Software manual or the online help.

Data exchange

Exchanging data with the fieldbus

The following program is an example for exchanging data with the fieldbus. It takes into consideration that initializing the fieldbus interface has already been made (see above).

Cyclically (timer-controlled), access to the DP-RAM is requested, the access right is checked, data are exchanged and access is returned again to the bus side. Before the exchange, the data is checked if it has been modified and only then will it be read out and transferred again.

```
#INCLUDE adwpext.inc
DIM data_1[1000] AS LONG 'Field for input data
DIM data_2[1000] AS LONG 'Field for output data
DIM n AS LONG

INIT:
 n = 1
 par_14 = 0
 FOR n = 1 TO 100          'Initialization of the
                           'transmission data
  data_2[n] = n
 NEXT n

EVENT:
 IF (par_14 = 0) THEN
  INC par_8
  par_8 = par_8 AND 0ffh
  REQUEST_ACCESS(0,6)      'request access to input and
                           'output areas

  par_14 = 1
 ENDIF
 IF (par_14 = 1) THEN
   par_1 = CHECK_ACCESS(0)'Check access right
   IF (par_1 = 6) THEN     'If it has access right....
    par_14 = 2
   ELSE
    REQUEST_ACCESS(0,6)    'else, request once more
   ENDIF
 ENDIF
 IF (par_14 = 2) THEN
  par_9 = CHANGED_DATA(0,32) 'Check if there are new data
  IF (par_9 <> 0) THEN     'If there are new data,then...
   INC par_7
   data_2[1] = par_8
   SET_WRITE_BUFFER(0,data_2,0,60) 'Write data (60 bytes)
   GET_READ_BUFFER(0,data_1,0200h,40) 'Read data
                           '(40 bytes)
  ENDIF
  REQUEST_RELEASE_ACCESS(0,6)'Release access rights
  par_14 = 3
 ENDIF
 IF (par_14 = 3) THEN
  par_1 = CHECK_ACCESS(0) 'Has the bus access rights
                           'again?
  IF (par_1 = 0) THEN
   INC par_11              'Count access cycles
   par_14 = 0
  ENDIF
 ENDIF
```

4.7.7 Pro-PROFI-DP-SL Rev. A

General information about this fieldbus module is given in chapter 4.7.6 "Pro-Fieldbus Modules", page 153.

fig. 274 shows the side-view (printed circuit board) and the front panel of the Pro-PROFI-DP-SL module.

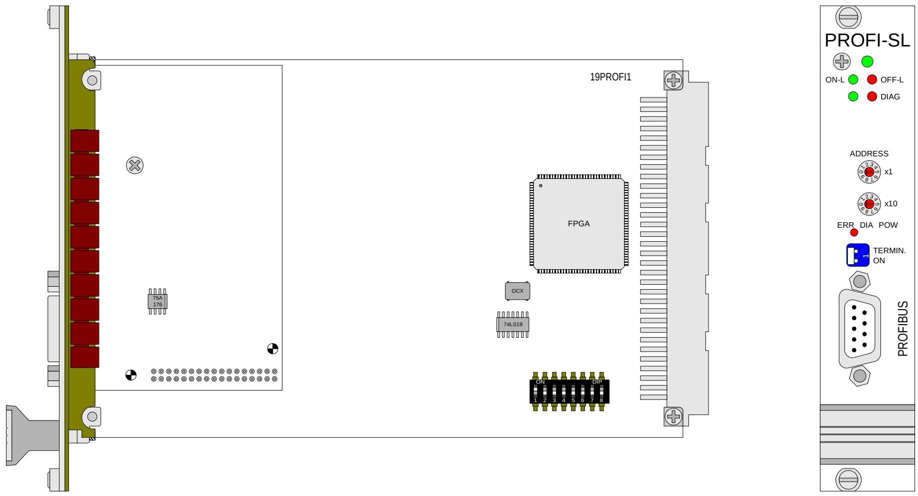


Fig. 274 – Pro-PROFI-DP-SL: Printed circuit board and front panel

Pin assignment

fig. 275 shows the pin assignment of the 9-pin D-SUB socket for the connection to the Profibus. The pin assigment is in agreement with the norm DIN E 19245, Part 3.

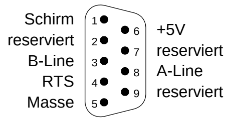
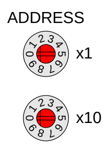


Fig. 275 – Pro-PROFI-DP-SL: Pin assignment

Profibus address



The Profibus address of the module is set using two switches at the front panel of the module (see graphic). The lower switch ("x10") is for the tens place, the upper switch ("x1") for the units place of the address. The address range is between 0 and 99.

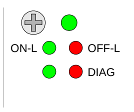
If for instance the lower switch is positioned on "4" and the upper on "3", address 43 is set.

The address will only be set during the internal initializing of the slave module, e.g. after power-up. Changing the switch position during operation will have no effect on the module on the Profibus, only after system restart.

Bus termination

The Profibus has to be terminated at its physical beginning and end of its segments by an active terminator. The module has a termination, which can be switched on or off with a switch at the front panel (see graphic). Switched to the position "ON" means: the termination is active.

Status display



On the front panel of the PROFI-DP-SL module there are four LEDs. 3 of them show the status of the module, (see table). The graphic shows the position of the LED.

LED	Meaning
ON-L	On (green): The module is online Off (red): No meaning
OFF-L	On (green): The module is offline Off (red): No meaning
DIAG	Fieldbus diagnosis: A flash of 1Hz (red): Input/output configuration does not match with the master configuration A flash of 4Hz (red): Error occured during initialization of the Profibus ASIC.

Fig. 276 – Pro-PROFI-DP-SL: Meaning of the status-LED

Integration into the Profibus

GSD file

The configuration of a Profibus is made using a configuration tool, which depends on the user and the selected master system. The tool gets its information about the slaves to be integrated from standardised files. This enables each master to access each slave. All files are in ASCII format and they are in agreement with the norm EN 50170. The file for the Pro-PROFI-DP-SL module will be delivered with the system and is called:

`hms1003.gsd`

The following details apply for all configuration tools: For more details for the bus configuration, see the documentation of the configuration tool.

Integrating the slave

Copy the GSD file into the source code of the configuration tool. Include the slave (the module) into the configuration tool. Afterwards the bus could be structured as shown below:

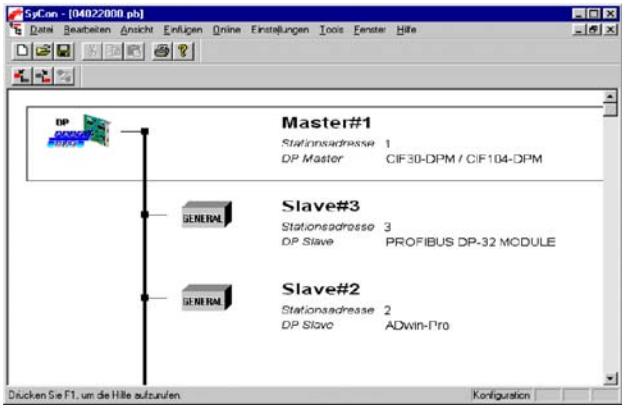


Fig. 277 – Pro-PROFI-DP-SL: Bus layout in the configuration tool

Configuring the slave

The memory of a slave is divided into areas, the memory modules. Three different types are available: IN/OUT, INPUT and OUTPUT. Each memory determines an area which has a specified size in the input or output area; the sizes of 1, 2, 4, 8, 16, 32, 64 and 128 bytes are available.
A memory module IN/OUT of the size 4 bytes needs 4 bytes in the input area and 4 bytes in the output area.

The configuration of the memory size for the input and output area must be in accordance with the configuration of the module made during initialization. Therefore you have to add the memory sizes of all memory modules - separated in input and output areas - and compare the result with the configuration you made during initialization.

The memory size for the input and output data must be in the range of 0 … 244 bytes. The totoal memory size of both areas must be in the range of 1 … 416 bytes.

fig. 278 shows the possible configuration of a module (slave):

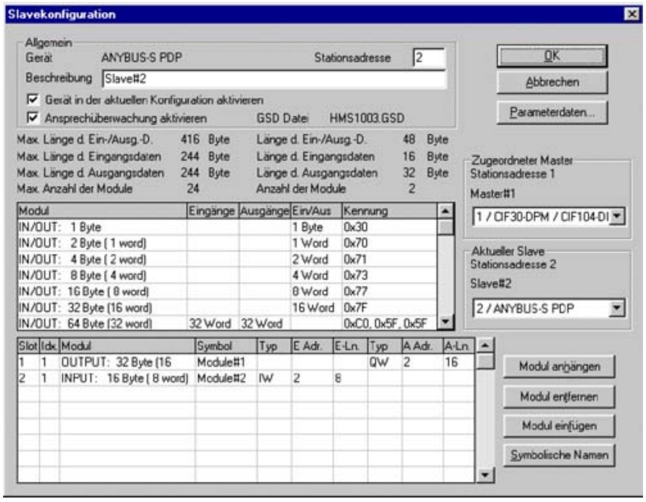


Fig. 278 – Pro-PROFI-DP-SL: Slave configuration

Example

If the module is initialized with 32 bytes input data and 32 bytes output data, the configuration tool can be configured as follows:

– IN/OUT (16 bytes)

– INPUT (16 bytes)

– OUTPUT (16 bytes)

The two areas with 32 bytes can also be combined differently. But it is important that the total of the bytes for the input and the total of the bytes for the output each match with the configuration made during initialization of the module. If not, the module cannot be part of the data exchange with the bus.

The module supports only the cyclic data exchange. Acyclic data cannot be transferred.

When the station address is set, indicate the value, which you have set with the address switches of the module. With another address the master cannot access the slave and therefore the module cannot exchange data with the bus.

Specifications

Norm

The module 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

Supported Baud rates

The following Baud rates are supported:

9.6kBit/s	187.5kBit/s	3MBit/s
19.2kBit/s	500kBit/s	6MBit/s
93.75kBit/s	1.5MBit/s	12MBit/s

Supported services

The following table shows which service the module provides and the description of the functions:

Service	Description of the functions
Cyclic Data Exchange	The module is part of the cyclic data exchange. The data coming from the master are accepted and the data requested by the master are transferred. The master controls this process.
Slave Diagnostic	The slave transmits - after request of the master - the standard diagnostic data, according to EN 50170.
Freez	If the slave gets a "freeze" message from the bus, it goes into the freeze mode. All input data are kept. This means that the data, which are presently in the input area are transfrerred via cyclic data exchange to the master. If this area is changed later, the data, being stored on the Profibus, will not be influenced. They will only change, if a new message arrives in the freeze mode from the master, or if the freeze mode is cancelled.
Unfreez	Cancels the freeze mode.
Sync	If the slave gets a "sync" message from the bus it goes into the sync mode. All output data are kept. This means that the data, which are presently in the output area are stable. If the master transmits other data via cyclic data exchange, they will not be transferred to the output area. This is only done, if a new sync message arrives from the bus and the sync mode will be cancelled.
Unsync	Cancels the sync mode.
Clear_Data	Sets all output data to 0. That means, all memory cells, which are located in the area of the output data, have the value 0 afterwards.

Fig. 279 – Pro-PROFI-DP-SL: Supported services

Operating modes

The following table shows the operating mode, the module 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. 280 – Pro-PROFI-DP-SL: Supported operating modes

4.7.8 Pro-Inter-SL Rev. A

General information about this fieldbus module is given in chapter 4.7.6 "Pro-Fieldbus Modules", page 153.

fig. 281 shows the side-view (printed circuit board) and the front panel of the Pro-Inter-SL module.

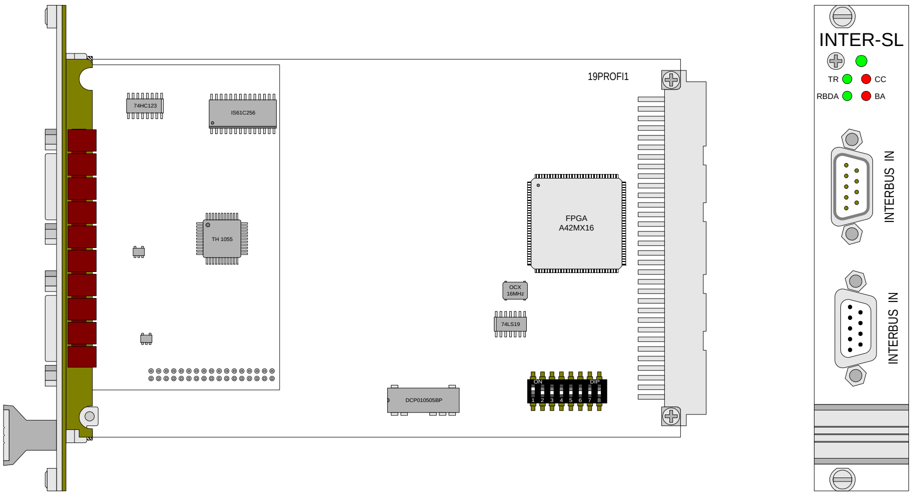


Fig. 281 – Pro-INTER-SL: Printed circuit board and front panel

Pin assignment

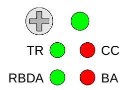
fig. 282 shows the pin assignment of the D-SUB sockets for the connection of the Interbus (input and output).

Input (socket) Output (socket)

Fig. 282 – Pro-INTER-SL: Pin assignment

Status display

On the front panel of the Pro-INTER-SL module are 4 status LEDs, which inform about the module communication. Table 6 illustrates the meaning of the LEDs:



LED	Name	Meaning when the LED is active
1	RBDA	Interbus output is switched off.
2	TR	PCP communication is active.
3	CC	Physical connection to the master, master will not be initialized.
4	BA	Bus is active.

Fig. 283 – Pro-INTER-SL: Meaning of the status LED

Integration into the Interbus

After having connected the bus (hardware), the master will be able to read the bus configuration. After reading the master has all necessary information about the connected slaves. For example the size of the input and output areas for cyclic data exchange and the size of the PCP communication. After reading the bus configuration the communication can be started immediately. Moreover, the master identifies the participants (DIO, PC, AIN, ...).

Depending on the configuration, the module can be a digital slave (ID 3) or a PCP participant (ID 243). As digital slave the module cannot exchange acyclical data with the bus master.

If during initialization of the fieldbus interface with **INIT_SLAVE** an area for parameter data is indicated, then the module is a PCP participant. (The return parameters `Par_in` and `Par_out` are not equal to zero). If the return parameters are equal to zero the module is a digital slave. fig. 284 shows a bus layout with a module as PCP participant.

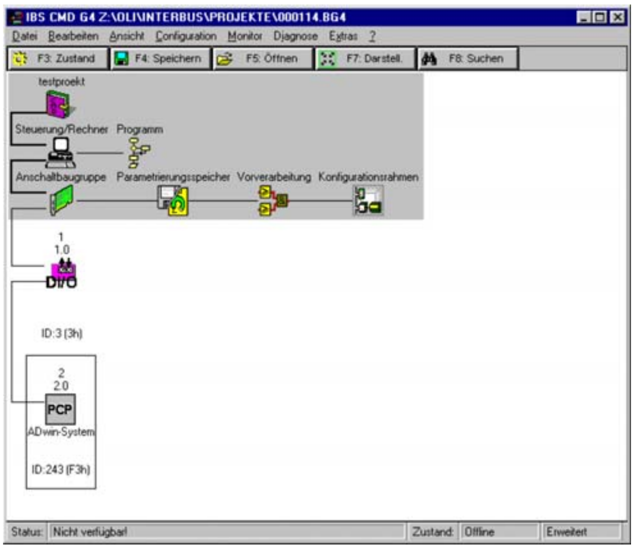


Fig. 284 – Pro-INTER-SL: Bus layout in the configuration tool

Baud rate

The Baud rate in the Interbus is 500 kBit/s, other transfer rates are not available. The module supports this Baud rate.

Configuring the slave

Input and output areas

When using an Interbus, an area for cyclic and acyclic data can be indicated during initialization of the fieldbus interface (see **INIT_SLAVE**). In the Interbus area sizes are indicated in words (1 word = 2 bytes); thus an input and output area of this size is specified, because the structure of the Interbus requires equally large areas for the input and output data.

The size of the area for cyclic data can be in the range of 1 ... 10 words.

The standard size für the PCP channel in the Interbus is 32 words for either input or output. This size is recommended, but not absolutely required. For the Pro-INTER-SL module the size of the area can be in the range of 1 ... 200 words.

PCP channel

The parameter channel is considered in the protocol with 1 word. That means that 2 bytes from the acyclic data are transmitted each cycle. Therefore the (acyclic) transmission of the parameter data is slower than the transmission of the cyclic data. The time needed for transmitting the parameter data is evaluated by the product of the bus cycle time and the number of the words to be transmitted in the parameter data set.

4.7.9 Pro-RSxxx Rev. A

The Pro-RSxxx module has 2 or 4 interfaces of the type RS-232 or RS-485. The names for the module types are shown in the table below:

	RS-232	RS-485
2 interfaces	Pro-RS232-2	Pro-RS485-2
4 interfaces	Pro-RS232-4	Pro-RS485-4

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.

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

The Pro-RS-xxx-4 module with 4 serial interfaces requires 2 module addresses. Therefore the module address +1 is set additionally to the module address set manually.

The description is divided into the following paragraphs:

– Hardware

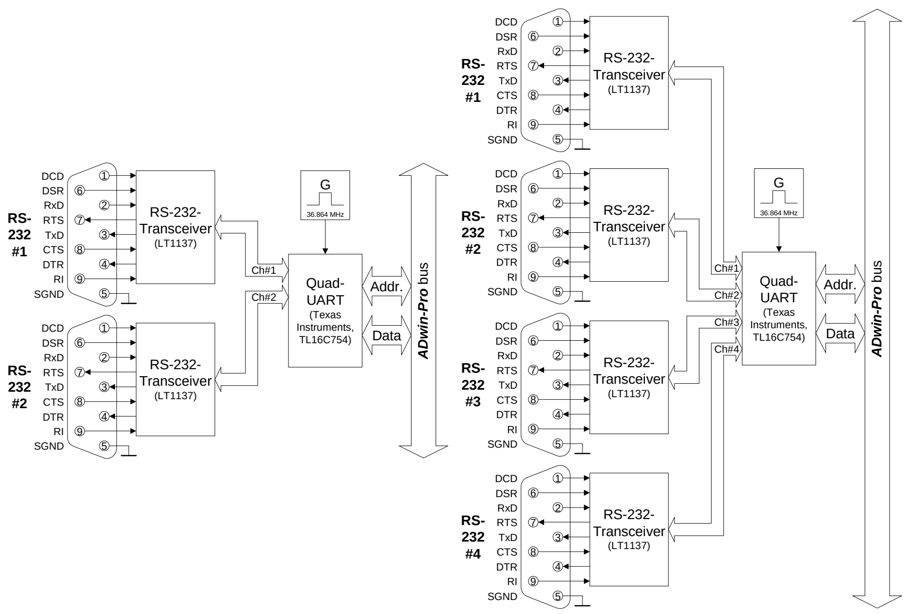
– Interface parameters

– Module revisions

– Programming

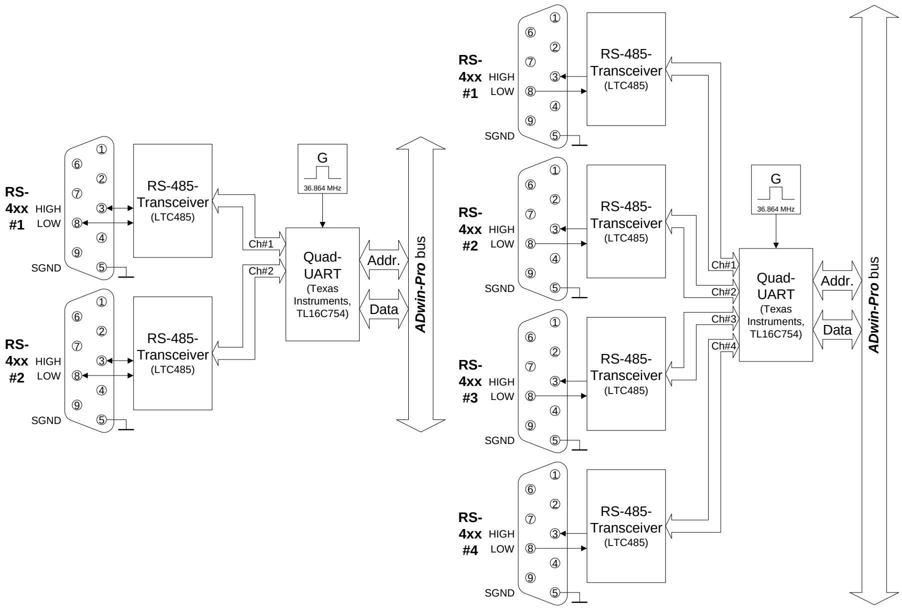
Hardware

Block diagrams



Pro-RS-232-2 Pro-RS-232-4

Fig. 285 – Pro-RS232: Block diagrams



Pro-RS-485-2 Pro-RS-485-4

Fig. 286 – Pro-RS485: Block diagrams

Printed circuit board, front panels

fig. 287 shows the side-view (printed circuit boad) and the front panel of the modules Pro-RS232-x and Pro-RS485-x.

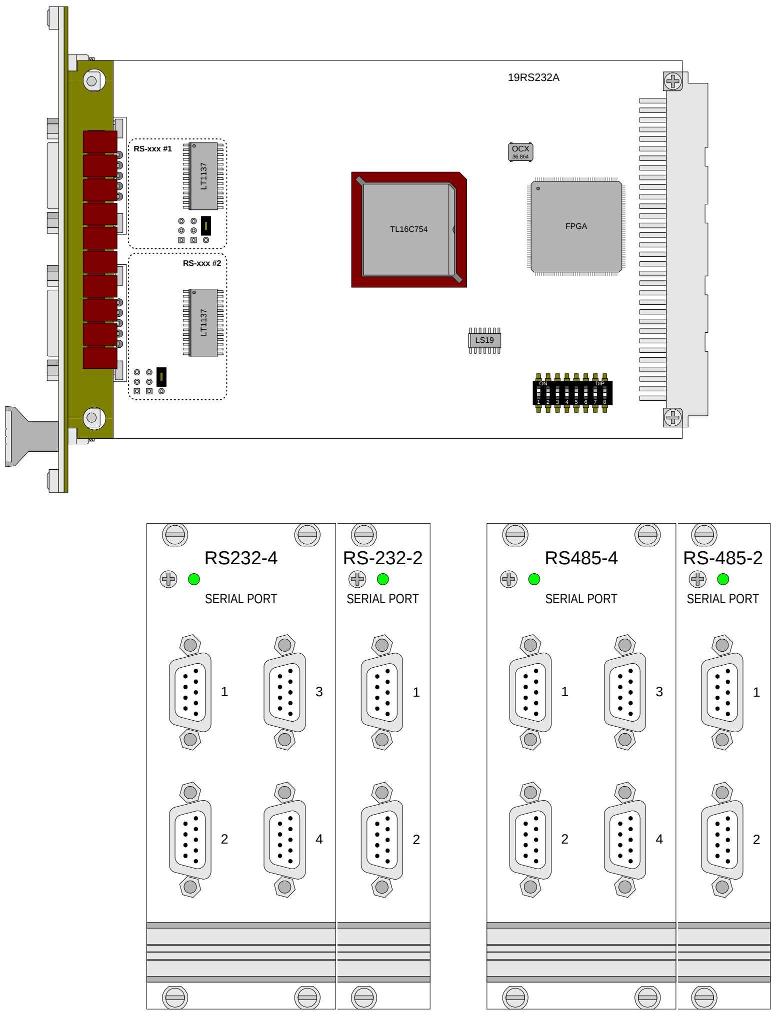
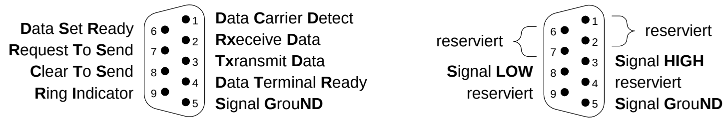


Fig. 287 – Pro-RSxxx: Printed circuit board and front panels

fig. 288 shows the pin assignment of the D-SUB sockets.



RS-232 (socket) RS-485 (socket)

Fig. 288 – Pro-RS-xxx: 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 by 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 by 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. 289 – Pro-RS-xxx: Baud rates

Special features of RS485

Contrary to the RS232 interface 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
A1		First version
A2	09/2002	New printed circuit layout
A3	10/2003	Internal program optimization

Programming

All RSxxx-y modules are equipped with the Quad Universal Asynchronous Receiver/Transmitter (UART), type TL16C754 from Texas Instruments®.

Functionality and programming of the interface depend on this controller. The modules are easily programmed by ***ADbasic*** instructions, which can be found in the include file `<ADPEXT.INC>`:

Area	Instructions
Initialization	`RS_INIT`, `RS_RESET`
Receiving and transmitting of data	`RS485_SEND`, `READ_FIFO`, `WRITE_FIFO`
Write and read access to the controller register	`GET_RS`, `SET_RS`

The instructions are described more detailed in the Pro software manual or the online help.

RS232: Send and receive

Example programs

The following program illustrates the initialization of the serial RS232 interface in the `INIT:` section. The process is timer-controlled.

```
REM The program initializes the serial interface
REM in the section Init:
REM In the section Event: data is exchanged between the interfaces
1 & 2 of the RS module.
REM The interfaces are tested with this program.
REM For this you have to connect the interfaces with
REM each other before starting the program.

#INCLUDE adwpext.inc
DIM DATA_1[1000] AS LONG  'Transmitted data
DIM DATA_2[1000] AS LONG  'Received data
DIM i AS LONG             'Control variable

INIT:
 FOR i = 1 TO 1000        'Initialization of the
                          'transmitted data
  DATA_1[i] = i AND 0FFh
 NEXT i
 RS_INIT(1,1,9600,0,8,1,0) 'Initializing interface 1:
                          '9600 Baud;
                          'No parity bit;
                          '8 data bits;
                          '2 stop bits;
                          'No handshake

 RS_INIT(1,2,9600,0,8,1,0) 'Initializing interface 2
                          'same as interface 1
 PAR_1 = 1
 PAR_4 = 1

EVENT:
 REM Read and write a data set
 IF (PAR_1 <= 1000) THEN  'Send data
  PAR_2 = WRITE_FIFO(1,1,DATA_1[PAR_1])
  IF (PAR_2 = 0) THEN INC PAR_1
 ENDIF

 PAR_3 = READ_FIFO(1,2)   'Read data
 IF (PAR_3 <> -1) THEN
  DATA_2[PAR_4] = PAR_3
  INC PAR_4
 ENDIF
 IF (PAR_4 > 1000) THEN END 'All data are transmitted
```

RS232:
Send string instruction

Many devices with an RS232 interface can be controlled using string instructions. The following programs show how to send a string in one process and how to receive the string with another process. Both programs are available on the ***ADwin*** CDROM.

The programs can be used on the same module but with different interfaces. Please pay attention to the remarks in the programs.

The program `RS232_send_string.BAS` first initializes interface 1. In the **EVENT** section the interface 1 sends a string char by char. In the **FINISH** section the character "#" is used as an end marker. It may be replaced by any other character.

```
' Process for RS232-communication: sending a string
' ++++++++++++++++++++++++++++++++++++++++++++++++++++++++
' The program may run together with RS232_receive_string.BAS
' on the same module.  If so, please follow these instructions:
' - connect the interfaces with each other
' - compile and start RS232_receive_string.BAS
' - compile and start RS232_send_string.BAS

#INCLUDE adwinpro.inc
#INCLUDE adwpext.inc

REM import string library
#IF PROZESSOR = T10 THEN
IMPORT string.lia
#ELSE
IMPORT string.li9
#ENDIF

#DEFINE rs_adr 1           'module address
#DEFINE rs_no 1            'interface number
#DEFINE s_endchar "#"      'end marker "#"
#DEFINE s_send DATA_1
#DEFINE str_len 50         'length of send string

DIM s_send[str_len] AS STRING 'send string
DIM s_temp[1] AS STRING    'single char
DIM sp AS LONG             'send pointer

INIT:
 GLOBALDELAY = 10000000    '0.25 s
 'A reset is allowed only once on a module!
 'RS_RESET(rs_adr)         'reset RS module
 RS_INIT(rs_adr,rs_no,9600,0,8,0,0) 'init RS interface
 sp=1                      'initialize pointer
 s_send = "This is a TESTSTRING" 'send string

EVENT:
 STRMID(s_send, sp, 1, s_temp) 'read next char of string
 PAR_11 = ASC(s_temp)      'get ascii code of char
 IF (PAR_11 = 0) THEN END 'quit when all chars are sent
 PAR_12 = WRITE_FIFO(rs_adr, rs_no, PAR_11) 'send code
 REM increase pointer, else send again
 IF (PAR_12 = 0) THEN INC sp
 REM quit when max. string length is reached
 IF (sp > str_len) THEN END

FINISH:
 DO                        'send end marker "#"
  PAR_11 = ASC(s_endchar) 'get ascii code
  PAR_12 = WRITE_FIFO(rs_adr, rs_no, PAR_11) 'send code
 UNTIL (PAR_12 = 0)
```

**RS232:
Receive string instruction**

The program `RS232_receive_string.BAS` first initializes interface 2. In the **EVENT** section the interface 2 receives a string until the end marker char is received (or the receiving string is full)

```
' Process for RS232-communication: Receiving a string.
' ++++++++++++++++++++++++++++++++++++++++++++++++++++++++
' The program may run together with RS232_send_string.BAS
' on the same module. If so, please follow these instructions:
' - connect the interfaces with each other
' - compile and start RS232_receive_string.BAS
' - compile and start RS232_send_string.BAS

#INCLUDE adwinpro.inc
#INCLUDE adwpext.inc

REM import string library
#IF PROZESSOR = T10 THEN
IMPORT string.lia
#ELSE
IMPORT string.li9
#ENDIF

#DEFINE rs_adr 1            'module address
#DEFINE rs_no 2             'interface number
#DEFINE s_receive DATA_2
#DEFINE str_len 50          'max. length of received string

DIM s_receive[str_len] AS STRING 'received string
DIM s_temp[1] AS STRING     'single char
DIM s_endchar[1] AS STRING 'end marker
DIM endflag AS LONG         '
DIM rp AS LONG              'receive pointer

INIT:
 GLOBALDELAY = 10000000     '0.25 s
 RS_RESET(rs_adr)           'reset RS module
 RS_INIT(rs_adr,rs_no,9600,0,8,0,0) 'init RS interface
 rp = 0                     'initialize receive pointer
 s_receive = ""             'initialize receive string
 s_endchar = "#"            'end marker

EVENT:
 PAR_21 = READ_FIFO(rs_adr, rs_no) 'receive status / char
 IF (PAR_21 <> -1) THEN
  CHR(PAR_21,s_temp)        'get char from ascii value
  INC rp                    'increase receive pointer
  REM end marker received or string full?
  endflag = STRCOMP(s_temp, s_endchar)
  IF ((endflag=0) OR (rp>str_len)) THEN END
  s_receive = s_receive + s_temp 'save char to string
 ENDIF
```

RS485: Receive and send

In this example the RS485 interface 2 is a passive participant, which reads data coming from the bus. If a specified byte (55) is received, the interface becomes active and starts sending the value 44.

```
REM This program implies a RS485 interface with the address 1.
#INCLUDE adwpext.inc

#DEFINE rs_adr 1

DIM ret_val AS LONG
DIM val AS LONG

INIT:
 RS_RESET(rs_adr)
 RS_INIT(rs_adr,2,38400,0,8,0,3)'Initialize channel 2
 RS485_SEND(rs_adr,2,0)    'channel 2 = receiving

EVENT:
 val = READ_FIFO(rs_adr,2) 'Read data

 IF (val = 55) THEN
  RS485_SEND(rs_adr,2,1)  'channel 2 = sending
  ret_val = WRITE_FIFO(rs_adr,2,44) 'Write data
 ENDIF
```

5 Calibration

5.1 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.

Please note: On several modules the input or output voltage range is adjustable via jumper or DIL switch. After every new setting you have to recalibrate the ADC/DAC, in order to assure correct measurement results.

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 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:
  - 30µV for calibration of 16 bit converters
  - 100µV for calibration of 12 bit converters

– a digital multi meter with an accuracy of:
  - 10µV for calibration of 16 bit converters
  - 100µV for calibration of 12 bit converters

– connecting cables from the input/outputs to the reference voltage and to the measurement device

– adapter board with a connector according to DIN 41612 with 96 pins[2]

– insulated adjusting tools[2]

2. only for modules with trimmers for calibration

5.2 Calculation basis

Voltage range

The standard voltage range of the analog inputs/outputs of the ***ADwin*** systems is -10V … +10V (bipolar 20 Volt).

The voltage ranges of the ***ADwin-Pro*** system can additionally be set to -5V…+5V (bipolar 10 Volt) and 0V…+10V (unipolar 10 Volt) by jumper.

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 12 bit AD conversion a value of 9.995117 Volt.

Zero offset

In bipolar settings this results in a zero offset, called offset in the following text. The offset has the following value:

V_{OFF}

Offset V_{OFF}	with setting
-10V	bipolar ±10 Volt (-10V…+10V)
-5V	bipolar ±5 Volt (-5V…+5V)
0V	unipolar 10V (0V…+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 16 bit converters: 20V ./. 2^{16} = 305.175µV

– with 12 bit converters: 20V ./. 2^{12} = 4.8828mV

– Further values of V_{LSB} can be found in fig. 291 (page 178).

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 the 12 bit ADC as with a 16 bit ADC, the converted value is presented left-aligned in the lower word (16 bit) with the 12 bit ADC. The least 4 significant bits are always 0 (see fig. 290).

<table>
<tr><td>bit-No.</td><td>31…16</td><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>09</td><td>08</td><td>07</td><td>06</td><td>05</td><td>04</td><td>03</td><td>02</td><td>01</td><td>00</td></tr>
<tr><td rowspan="2">32 bit-Memory</td><td rowspan="2">0
(upper word)</td><td colspan="12">12 bit value of the 12 bit-ADC in the lower word (left aligned)</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td colspan="16">16 bit value of the 16 bit-ADC in the lower word</td></tr>
</table>

Abb. 290 – Bit allocation of 12 bit-ADC and 16 bit-ADC

Explanation to fig. 290

The 4096 digits of the 12 bit ADC are mapped to the 65535 digits of the 16 bit ADC. Therefore, 16 digits of the 16 bit ADC correspond to one digit of the 12 bit ADC.The following equations apply therefore for both ADC types.

For DAC use the formula:

DAC

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

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

For ADC use the formula:

ADC

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

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

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.

5.3 Calibrating a module

Initializing the hardware

At first define the voltage range by setting the jumpers when the system is switched off.

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.

Depending on the module you calibrate with one of the following methods:

– Calibration per Software, chapter 5.3.1

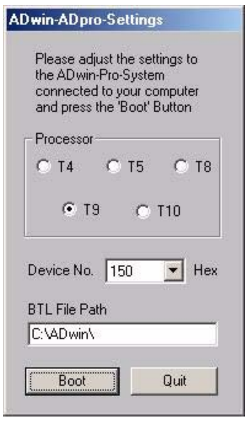
– Calibration with Trimmers, chapter 5.3.2

5.3.1 Calibration per Software

Please note the general information in chapter 5.1.

Call the program `ADpro.exe` from the Windows start menu under "`Programs\ADwin`".

If your ***ADwin*** system has booted successfully, the window "`ADwin - ADpro`" opens.



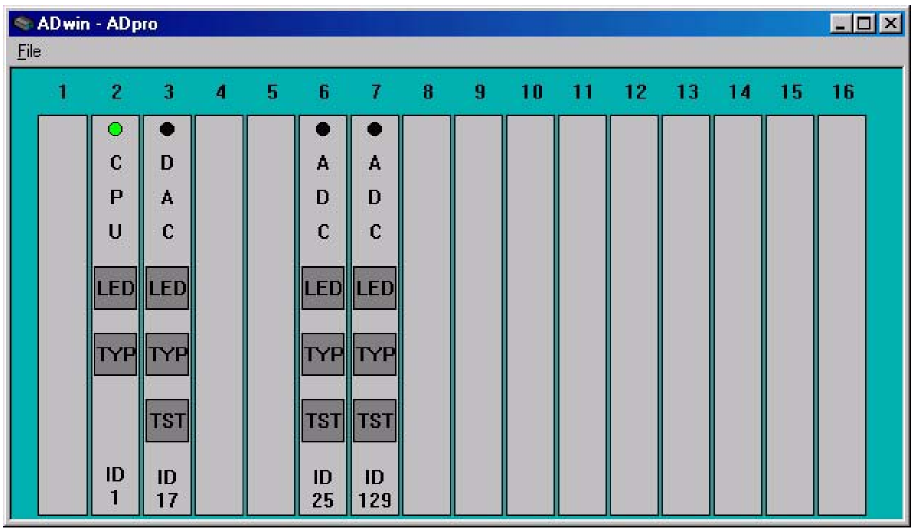
The window "`ADwin-ADpro Settings`" to the left opens only,

– if your ***ADwin*** system did not boot successfully.

– if the settings in this window are not correct.

Check and, if necessary, correct the settings in this window. Boot your system again with the "`Boot`" button.

Notes for operating several ***ADwin*** systems at one PC: ADpro saves the values, indicated in the window "`ADwin ADpro Settings`" after each successful booting. If you change the system you can adapt the settings under "`ADwin-ADpro\File\Option`" in the window "`Connection`".



Select the "`TST`" button in the "`ADwin-ADpro`" window for the module to be calibrated, then select the "`Calibrate`" button in the "`Dialog`" window.

If the button "`Calibrate`" is not displayed in the "`Dialog`" window, the selected module is calibrated by trimmers, see the following chapter 5.3.2 "Calibration with Trimmers".

Calibration

Connect the measurement device and the reference voltage source.

The individual calibration steps are described in the windows "`AOUT Calibrate`" or "`AIN Calibrate`". Please note that

– 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`".

5.3.2 Calibration with Trimmers

Please note the general information in chapter 5.1.

> 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 ventiation slots!

If access to the trimmers is not easily possible, use an adapter board with a connector according to DIN 41612 with 96 pins. Please take into consideration that the DAC and ADC are rapidly cooling down: Finish the calibration after some minutes.

Offset and Gain

Program start

Start ***ADbasic*** and boot your system.

Connecting

Connect the measurement device and the reference voltage source.

Calibration programs

In the annex of this chapter you will find the programs for easy and fast calibration. You will find:

– The position of the offset and gain trimmers in the figures of your module description in this manual.

– The configurations in the following fig. 291, if you do not want to work with the calibration programs.

– Please pay also attention to the previous notes about INL and DNL.

Vol-tage	Digits dec. Digits hex.	Max. value 65535 `FFFFh`	Max. test value 64080 `FA50h`	Mean test value 32768 `800h`	Min. test value 1456 `5B0h`	Min. value 0 `0h`
-10V to +10V	16 bit U_{LSB}: 305.1758µV	+ 9.9996948V	+ 9.5556641V	0V	- 9.5556641V	- 10V
	12 bit U_{LSB}: 4.88281mV	+ 9.9951172V				
-5V to +5V	16 bit U_{LSB}: 152.5879µV	+ 4.9998474V	+ 4.7778320V	0V	- 4.7778320V	- 5V
	12 bit U_{LSB}: 2.44141mV	+ 4.9975586V				

		Max. value	Max. test value	Mean test value	Min. test value	Min. value
Vol-tage	Digits dec.	65535	64080	32768	1456	0
	Digits hex.	FFFFh	FA50h	800h	5B0h	0h
0V to +10V	16 bit U_{LSB}: 152.5879µV	+ 9.9998474V	+ 9.7778320V	+ 5V	+ 0.2221680V	0V
	12 bit U_{LSB}: 2.44141mV	+ 9.9975586V				

Abb. 291 – Assignment of digits to voltage at the inputs/outputs dependent on the configuration of ADC and DAC

Depending on the module group, the trimmers have different effects. Please take into consideration that the calibration has absolutely to be made according to the order shown in the following section "Calibration with ADbasic Programs", even if you work with different test values.

5.4 Calibration with *ADbasic* Programs

In the annex of this chapter you will find ***ADbasic*** programs for fast and easy calibration of the different ***ADwin-Pro*** modules. Please proceed according to the following order:

AOut-4/16, AOut-8/16

– AOut-4/16, AOut-8/16 (bipolar and unipolar)

1. Calibrate Offset:
   - Enter the digital **min. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the offset trimmer for the relevant channel (BPO, UPO).
2. Calibrate Gain:
   - Enter the digital **max. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the corresponding gain trimming potentiometer.
3. Check:
   - Check all 3 test values from fig. 291.

AIn 8/16 Rev. A bipolar

– AIn 8/16, bipolar

1. Calibrate Offset:
   - Enter the digital **mean test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the offset trimmer for the relevant channel (BPO).
2. Calibrate Gain:
   - Enter the digital **max. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the corresponding gain trimming potentiometer.
3. Check:
   - Check all 3 test values from fig. 291.

AIn 8/16 Rev. A unipolar

– AIn 8/16, unipolar

1. Calibrate Offset:
   - Enter the digital **min. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the offset trimmer for the relevant channel (UPO).
2. Calibrate Gain:
   - Enter the digital **max. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the corresponding gain trimming potentiometer.
3. Check:
   - Check all 3 test values from fig. 291.

AIn 8/12

– AIn 8/12

1. Calibrate Offset:
   - Enter the digital **min. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the offset trimmer for the relevant channel.
2. Calibrate Gain:
   - Enter the digital **max. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the corresponding gain trimming potentiometer.
3. Check:
   - Check all 3 test values from fig. 291.

AIn 32/12

– AIn 32/12 (bipolar und unipolar)

1. Calibrate Offset:
   - Enter the digital **min. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the offset trimmer for the relevant channel (BPO, UPO).
2. Calibrate Gain:
   - Enter the digital **max. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the corresponding gain trimming potentiometer.
3. Check:
   - Check all 3 test values from fig. 291.

AIn F 8/12, AIn F 4/12

– AIn F 8/12, AIn F 4/12

1. Calibrate Offset:
   - Enter the digital **mean test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the offset trimmer for the relevant channel.
2. Calibrate Gain:
   - Enter the digital **max. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the corresponding gain trimming potentiometer.
3. Check:
   - Check all 3 test values from fig. 291.

AIn F 8/16, AIn F 4/16

– AIn F 8/16, AIn F 4/16

1. Calibrate Offset:
   - Enter the digital **mean test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the offset trimmer for the relevant channel.
2. Calibrate Gain:
   - Enter the digital **max. test value** for `PAR_8` in the parameter window and confirm with "`Send`" or by pressing [RETURN].
   - Set the voltage value with the corresponding gain trimming potentiometer.
3. Check:
   - Check all 3 test values from fig. 291.

5.5 Programs for calibration

You will find the ***ADbasic*** programs for calibration as source files in the directory <`C:\ADwin\Tools\Calibration\`…> with standard installation from the ***ADwin*** CDROM (beginning with version 3.00.30xx).

5.5.1 *Pro* AOut 4/16 und 8/16 (DAC)

```
'Process for the ADwin-Pro in order to
'output voltage with an AOUT module.
'Last modification on July 18, 2000 ur
'Usage of the variables:
'PAR_6 : module address        (1 to 255)
'PAR_7 : channel number        (1 to 8)
'PAR_8 : output value          (0 to 65535)

#INCLUDE adwpad.inc
#INCLUDE adwpda.inc

'#####################################################
INIT:
GLOBALDELAY=2000
IF (PAR_6=0) then PAR_6=1 'prevent module address 0
IF (PAR_7=0) then PAR_7=1 'prevent channel number 0
IF (PAR_8=0) then PAR_8=32768
'64080 => +9.555664V (at a voltage range of +/- 10V)
'32768 => 0V
'1456 => -9.555664V

,#####################################################
EVENT:
DAC(PAR_6,PAR_7,PAR_8)'output value
```

5.5.2 Pro AIn 8/16 Rev. A (ADC)

```
'Process for the ADwin-Pro in order to
'read voltage with an AIN-8/16 module.
'The mean value is calculated in FPAR_1.
'Last modification on August 08, 2000 ur
'Usage of the variables:
'PAR_1 : module address      (1 to 255)
'PAR_2 : channel number      (1 to 8)
'PAR_3 : read value          (0 bis 65535)
'FPAR_1: mean value

#INCLUDE adwpad.inc
#INCLUDE adwpda.inc

'#######################################################
INIT:
GLOBALDELAY=2000
IF (PAR_1=0) then PAR_1=1 'prevent module address 0
IF (PAR_2=0) then PAR_2=1 'prevent channel number 0 (not
allowed
IF (PAR_3=0) then PAR_3=32768
'64080 => +9.555664V (at a voltage range of +/- 10V)
'32768 => 0V
'1456 => -9.555664V

'#######################################################
EVENT:
PAR_3=ADC16(PAR_1,PAR_2)'read value
FPAR_1=FPAR_1*0.95 + PAR_3*0.05'mean value
```

5.5.3 Pro AIn 8/12 (ADC), -Pro AIn 32/12 (ADC)

```
'Process for the ADwin-Pro in order to read a voltage
'with an AIN-8/12 or AIN-32/12 module.
'A mean value is calculated in FPAR_1.
'Last modification on August 08, 2000 ur
'Usage of the variables:
'PAR_1 : module address       (1 to 255)
'PAR_2 : channel number       (1 to 32)
'PAR_3 : read value           (0 to 65535)
'FPAR_1: mean value

#INCLUDE adwpad.inc
#INCLUDE adwpda.inc

'#######################################################
INIT:
GLOBALDELAY=2000
IF (PAR_1=0) then PAR_1=1 'prevent module address 0
IF (PAR_2=0) then PAR_2=1 'prevent channel number 0 (not
allowed)
IF (PAR_3=0) then PAR_3=32768
'64080 => +9.555664V (at a voltage range of +/- 10V)
'32768 => 0V
'1456 => -9.555664V

'#######################################################
EVENT:
PAR_3=ADC(PAR_1,PAR_2)  'read value
FPAR_1=FPAR_1*0.95 + PAR_3*0.05'mean value
```

5.5.4 Pro AIn F-4/16 und 8/16 (ADC) , Pro AIn F-4/12 und 8/12 (ADC)

```
'Process for the ADwin-Pro in order to read a voltage
'with an AIN-F module.
'A mean value is calculated in FPAR_1
'Last modification on August 08, 2000 ur
'Usage of the variables:
'PAR_1 : module address   (1 to 255)
'PAR_2 : channel number   (1 to 8)
'PAR_3 : read value       (0 to 65535)
'FPAR_1: mean value

#INCLUDE adwpad.inc
#INCLUDE adwpda.inc

'########################################################
INIT:
GLOBALDELAY=2000
IF (PAR_1=0) then PAR_1=1 'prevent module address 0
IF (PAR_2=0) then PAR_2=1 'prevent channel number 0 (not
allowed)
IF (PAR_3=0) then PAR_3=32768
'64080 => +9.555664V (at a voltage range of +/- 10V)
'32768 => 0V
'1456 => -9.555664V

'########################################################
EVENT:
PAR_3=ADCF(PAR_1,PAR_2) 'read value
FPAR_1=FPAR_1*0.95 + PAR_3*0.05'mean value
```

6 Accessories

6.1 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

6.2 Adapter sets

Pro-AS-1	4 adapters: LEMO sockets to BNC connectors (male)
Pro-AS-3	4 LEMO Y connector (male to double female)
Pro-AS-4	4 adapters: LEMO socket to LEMO socket
Pro-AS-5	4 terminators: 50 Ω, LEMO socket
Pro-AS-6	4 cable adapters (length 4" / 10 cm): LEMO socket to BNC connector (male)

6.3 Ordering the LEMO connectors

You can directly order the needed LEMO connectors at the manufacturer address.

LEMO GmbH

Hanns-Schwindt-Straße 6
Postfach 820529
D-81829 München

Tel.: +49 89 42770-3
Fax: +49 89 4202192

E-Mail: lemo@info.de
Internet: www.lemo.ch

In general, we use the following connectors:

– Male connectors / sockets of series 00 (NIM-CAMAC):
  - Cable connector: Type FFS
  - Built-in socket: Type ERN

– Pt100 modules: Male connectors / sockets of series 0B:
  - Cable connector: Type FGG
  - Built-in socket: Type EGG

Prior to changes.

Annex

A-1 Table of Figures

A-2 Alphabetic List of Modules