



Technical support:  
+49 (0)7223 / 9493-0

**Technical description**

**ADDINUM PA 150**

**Digital input and output board**

8<sup>th</sup> edition 09/2004

## Product information

This manual contains the technical installation and important instructions for correct commissioning and usage, as well as production information according to the current status before printing.

The content of this manual and the technical product data may be changed without prior notice.

ADDI-DATA GmbH reserves the right to make changes to the technical data and the materials included herein.

## Warranty and liability

The user is not permitted to make changes to the product beyond the intended use, or to interfere with the product in any other way.

ADDI-DATA shall not be liable for obvious printing and phrasing errors. In addition, ADDI DATA, if legally permissible, shall not be liable for personal injury or damage to materials caused by improper installation and/or commissioning of the board by the user or improper use, for example, if the board is operated despite faulty safety and protection devices, or if notes in the operating instructions regarding transport, storage, installation, commissioning, operation, thresholds, etc. are not taken into consideration. Liability is further excluded if the operator changes the board or the source code files without authorisation and/or if the operator is guilty of not monitoring the permanent operational capability of working parts and this has led to damage.

## Copyright

This manual, which is intended for the operator and its staff only, is protected by copyright. Duplication of the information contained in the operating instructions and of any other product information, or disclosure of this information for use by third parties, is not permitted, unless this right has been granted by the product licence issued. Non-compliance with this could lead to civil and criminal proceedings.

## ADDI-DATA software product licence

Please read this licence carefully before using the standard software. The customer is only granted the right to use this software if he/she agrees with the conditions of this licence.

The software must only be used to set up the ADDI-DATA boards.

Reproduction of the software is forbidden (except for back-up and for exchange of faulty data carriers). Disassembly, decompilation, decryption and reverse engineering of the software are forbidden. This licence and the software may be transferred to a third party if this party has acquired a board by purchase, has agreed to all the conditions in this licence contract and the original owner does not keep any copies of the software.

## Trademarks

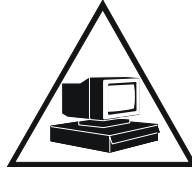
- ADDI-DATA is a registered trademark of ADDI-DATA GmbH.
- Turbo Pascal, Delphi, Borland C, Borland C++ are registered trademarks of Borland Insight Company.
- Microsoft C, Visual C++, Windows XP, 98, Windows 2000, Windows 95, Windows NT, EmbeddedNT and MS DOS are registered trademarks of Microsoft Corporation.
- LabVIEW, LabWindows/CVI, DasyLab, Diadem are registered trademarks of National Instruments Corp.
- CompactPCI is a registered trademark of PCI Industrial Computer Manufacturers Group.
- VxWorks is a registered trademark of Wind River Systems Inc.

## **WARNING**

**In case of wrong uses and if the board is not used for the purpose it is intended:**



**people may be injured**



**the board, PC and peripheral may be destroyed**



**the environment may be polluted**

**★★★ Protect yourself, the others and the environment★★★**

- **Read carefully the safety leaflet (yellow)!**

If this leaflet is not with the documentation , please contact us and ask for it.

- **Observe the instructions of the manual!**

Make sure that you do not forget or skip any step. We are not liable for damages resulting from a wrong use of the board.

- **Any question?**

Our technical support is at your disposal.

<b>1</b>	<b>INTENDED PURPOSE OF THE BOARD .....</b>	<b>5</b>
1.1	Intended use .....	5
1.2	Usage restrictions.....	5
1.3	General description of the board.....	5
<b>2</b>	<b>OPERATION .....</b>	<b>6</b>
2.1	Inputs.....	6
2.2	Outputs.....	6
<b>3</b>	<b>FUNCTIONING .....</b>	<b>8</b>
<b>4</b>	<b>MECHANICAL SET-UP .....</b>	<b>8</b>
<b>5</b>	<b>SIGNALS - DIRECT CONNECTOR.....</b>	<b>8</b>
<b>6</b>	<b>PERIPHERAL SIGNALS - SUB-D PIN CONNECTOR.....</b>	<b>9</b>
6.1	General technical data .....	9
6.2	Technical data for inputs.....	9
6.3	Technical data for outputs .....	9
<b>7</b>	<b>ADJUSTING THE BOARD.....</b>	<b>10</b>
7.1	Base address.....	10
7.2	Watchdog .....	11
<b>8</b>	<b>INSTALLATION INTO THE SYSTEM .....</b>	<b>12</b>
<b>9</b>	<b>SOFTWARE .....</b>	<b>12</b>
<b>10</b>	<b>PROGRAMMING .....</b>	<b>12</b>
10.1	Inputs.....	12
10.2	Outputs.....	13
<b>11</b>	<b>APPENDIX.....</b>	<b>14</b>
11.1	Circuit configuration .....	14
11.2	Component scheme .....	15
11.3	62-pin direct connector .....	16
11.4	37-pin SUB-D connector .....	17
11.5	Programming examples: C, Pascal, Assembler, Basic .....	18
11.6	Connection examples .....	22

## Figures

Fig. 7a: DIP switches and Jumpers .....	10
Fig 7b - watchdog active or out-of-action .....	11

# 1 INTENDED PURPOSE OF THE BOARD

## 1.1 Intended use

The board PA 150 must be inserted in a PC with ISA slots, which is used as electrical equipment for measurement, control and laboratory pursuant to the norm IEC 61010-1.

## 1.2 Usage restrictions

The PA 150 board must not to be used as safety related part for securing emergency stop functions.

The board must not be used in potentially explosive atmospheres.

## 1.3 General description of the board

Data exchange between the PA 150 board and the peripheral is to occur through a shielded cable. This cable must be connected to the 37-pin SUB-D male connector of the board.

The board has 16 inputs and 16 outputs intended for processing 24 V digital signals. An external 24 V supply voltage is necessary to run the outputs. The terminal board PX 901 and the relay board PX 8500 allow to connect the 24 V supply voltage through a shielded cable.

The use of the board in combination with external terminal or relay boards is to occur in a closed switch cabinet; the installation is to be effected competently.

The use of the board in a PC could change the PC features regarding to noise emission and immunity. Increased noise emission or decreased noise immunity could result in the system not being conform anymore. Check the PC's and cable's shielding capacity prior to putting the device into operation.

The connection with our standard cable ST010 complies with the specifications:

- metallized plastic hood
- shielded cable
- cable shield folded back and firmly screwed to the connector housing.

The use of the board according to its intended purpose includes observing all advises given in this manual and the safety leaflet. Uses beyond these specifications are not allowed. The manufacturer is not liable for any damages which would result from the non-observance of this clause.

For a quick and safe installation of this board please read absolutely following chapters :

- 7. Adjusting the board
- 9. Programming

## 2 OPERATION

The board PA 150 is intended for parallel data input and output, with simultaneous separation of peripheral and system sides. The signal inputs and outputs are divided into two groups of eight bits each. All inputs and outputs are filtered. Each input and output has a go line to the front connector. The outputs and the inputs use each a separate return line.

The board needs no software initialisation. It is ready to operate immediately after applying the working voltage and after power ON reset. Data transfer between the CPU and board PA 150 occurs directly over a buffer for inputs and over a data latch register for outputs.

### 2.1 Inputs

The optically decoupled inputs have a maximum load of 24 mA. To avoid damage to the optical coupler through pole reversal of input voltage, a diode is connected antiparallel to the LED of the optical coupler.

Through the galvanic separation via optical coupler, disturbances from the peripheral side are filtered to the system bus side, so that the effect of inductive and capacitive disturbances is reduced.

A high dynamic and switching threshold of the input signal is achieved by inserting a Z-diode in each input line. All inputs comply with industrial standard of +24 V for logic "1".

### 2.2 Outputs

The transistor outputs have a switching capability of 30 V, 200 mA.

External disturbances, which can occur in industrial environments, are suppressed through the galvanic decoupling of optical couplers. The outputs correspond to the industrial standard, and switch the +24 V outwards via a transistor.

The board is equipped with a watchdog. It disconnects the digital outputs, if the board has not responded within a preset time of more than 4 seconds.

In this way the computer can automatically withdraw released switching commands, if the computer loses control over inputs/outputs due to hardware and software failings. This watchdog-function can be turned on or off over jumper.

Each input/output byte occupies 1 byte within the 64K I/O address space. The base address is selected in steps of 4 bytes by a 10 pole DIP switch and four jumpers.

The board is equipped for direct use in an IBM or compatible PC with a 62 pole direct connector. The periphery is connected over a 37 pole MIN-D pin connector. Thus the following connection possibilities (see connection examples in the appendix):

- over a cable directly to contact and signal transmitters as well as to output loads (relays, lamps ...);
- over our standard cable ST010 to terminal board PX 900 in order to connect to the inputs and outputs. From there over cable ST020 to relay board PX 850 for outputs;
- over our standard cable ST010 to terminal board PX 901-D in order to connect to the inputs and outputs.

For all the above mentioned possibilities, the test adapter PX 910 is a useful help for hardware and software set-ups. The test adapter can also be used without peripheral connection.

### 3 FUNCTIONING

The board PA 150 needs no initialising. It is immediately ready to work after turning power on and successful RESET of the CPU.

The address decoding relates to the 64K I/O address area. Both lower address bits are decoded for the selection of 2 blocks of eight inputs each and 2 blocks of eight outputs each. The higher address bits are compared with the on-board switches ; the board is enabled when they are identical.

The control logic takes over co-ordination between board and microcomputer.

The peripheral signals are connected over the front connector and are carried to the optical couplers over the protection and filter circuitries.

### 4 MECHANICAL SET-UP

A 1,6 mm thick printed circuit supports the mechanical and electrical connection (230 x 99 mm). The connection with the microcomputer bus system occurs over the 62 pole gold-plated direct connector. The 37 pole MIN-D connector is the connection with the periphery. The board is plugged directly into the PC and is screwed onto the back panel of the appliance with an hinge.

### 5 SIGNALS - DIRECT CONNECTOR

The board is connected with the microcomputer directly over a direct connector. You will find the signal setting pattern of the direct connector in appendix 10.3.

The signals have the following meanings:

A0 - A15	: Address Lines
D0 - D7	: Data Lines
IOR/	: I/O Read
IOW/	: I/O Write
AEN	: Address Enable
RESET DRV	: System Initialising
GND	: 0V, Ground
+ 5 V	: 5V Supply

Board consumption

- all outputs logic "0": 200 mA typ.
- all outputs logic "1": 450 mA typ.



## 6 PERIPHERAL SIGNALS - SUB-D PIN CONNECTOR

The board is connected with the periphery over a 37 pole MIN-D pin connector. You will find the connector setting pattern in appendix 10.4.

### 6.1 General technical data

Board dimensions ..... : 230 x 99 mm  
 Temperature range ..... : 0-60°C

### 6.2 Technical data for inputs

Input current : logic "1" ..... :  $\geq 6,3$  mA (at 24V)  
 Signal delay ..... : 65  $\mu$ s, nominal  
 Maximal input frequency ..... : 5 kHz  
 Minimal input voltage ..... : 16 V  
 Upper limit of input voltage ..... : 30 V  
 Test voltage of individual  
 inputs against PC side ..... : 0.5 kV

### 6.3 Technical data for outputs

Output current : logic "1" ..... : 200 mA  
 Upper limit of output current  
 (only 1 channel) ..... : 300 mA (24V)  
 Voltage drop to the output  
 transistor at 200mA ..... :  $\leq 0.5$ V  
 Upper limit of the working voltage  
 switched by the output ..... : 30 V  
 Test voltage of individual outputs  
 against PC side ..... : 0.5 kV  
 Fuse ..... : slow 1.6 A

Dynamic data with an output current  
 of 100mA

Output pulse (logic "1") ..... : typ. 15  $\mu$ s  
 Turn on switching delay ..... : 2  $\mu$ s  
 Turn off switching delay ..... : 10  $\mu$ s

## 7 ADJUSTING THE BOARD

### 7.1 Base address

For responding the board, the address is adjusted with switches. A block of 4 addresses is assigned to the board for data transfer with the CPU. This address block of the available I/O address space is freely selectable.

The board is adjusted over 4 solder jumpers and a 10 pole DIP switch. The jumpers are designated J1 to J4 and the switches S1 to S10. They can be set in "ON" or "OFF" position.

Caution ! - a switch in "ON" position is equivalent to the logic "0"  
 - a switch in "OFF" position is equivalent to the logic "1"

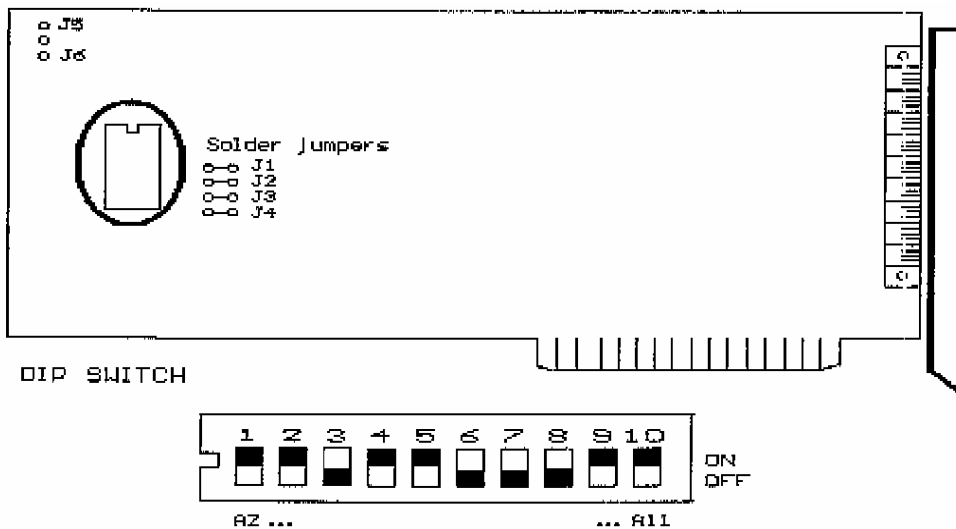
Switch :	J4*	J3*	J2*	J1*	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1
Addr. bit :	A15	A14	A13	A12	A11	A10	A9	A8	A7	A6	A5	A4	A3	A2

The board is delivered with the following configuration : I/O address H0390

Switch :	J4*	J3*	J2*	J1*	S10	S9	S8	S7	S6	S5	S4	S3	S2	S1
State:	0	0	0	0	0	0	1	1	1	0	0	1	0	0

\* Permanently coded at "0" over a solder jumper.

**Fig. 7a: DIP switches and Jumpers**

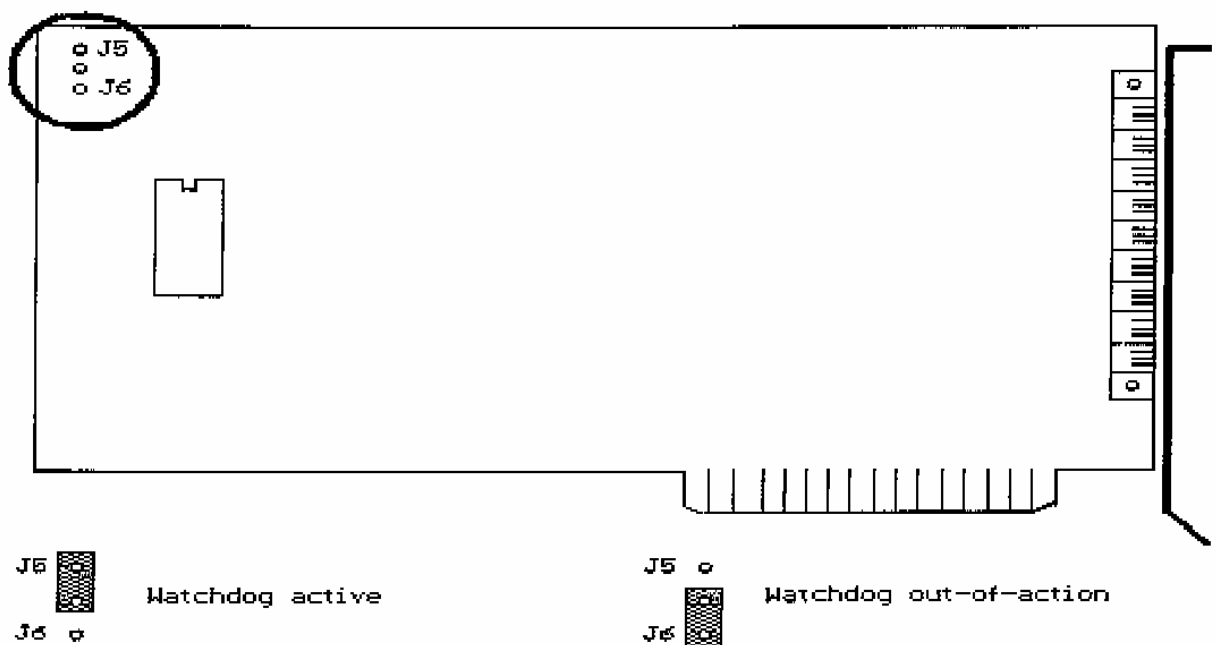


## 7.2 Watchdog

The watchdog circuit consists of a monostable multivibrator which preset time is comprised between 4 and 5 sec. The monostable multivibrator is rearmed each time when the board is responded by reading or writing. If the board has not be responded within more than 4 sec, the watchdog circuit is activated and sets to "0" all the outputs.

The watchdog circuit acts over jumper J5. When putting software into operation or for certain uses, this function can cause disturbances. They can be suppressed by replacing jumper J5 with J6.

**Fig. 7b: Watchdog active or out-of-action**



## 8 INSTALLATION INTO THE SYSTEM

The board fits into a free slot of your PC.

For installation, open the appliance following manufacturer instructions. Be sure to turn power off first. On the back panel remove the covering plate of your free slot. Then press the board into the slot and fasten it to the back side with an hinge. The machine may now be closed and power may be turned on.

Afterwards system tests are carried out. The periphery can be connected. We suggest to use terminal board ADDIVARIOUS PX 900 and a standard cable ST010, since this needs no inconvenient wiring.

The test adapter PX 910 is useful for simulating and testing input and output functions. With this aid you will have a quick and economical peripheral connection (see appendix 10.6).

## 9 SOFTWARE



### IMPORTANT!

Further information for installing and uninstalling the different drivers is to be found in the delivered description "**Installation instructions for the ISA bus**".

A link to the corresponding PDF file is available in the navigation pane (Bookmarks) of Acrobat Reader.

## 10 PROGRAMMING

For information exchange between the CPU and the board 4 addresses are assigned, which are differentiated by the 2 lowest address bits. The higher address bits A2 - A15 are compared with the DIP switch and jumper adjustment described in chapter 7 "Adjusting the board".

Board PA 150 needs no software initialisation. It can be directly used with the adjusted address.

### 10.1 Inputs

Let's suppose that the address is set on 0390H. The 16 digital inputs are read with these commands:

INP (&H0390)

INP (&H0391)

Digital input 1 corresponds with bit 0 of INP (&H0390)

Digital input 16 corresponds with bit 7 of INP (&H0391).

In digital signal processing the state of a given input must be checked often. For example digital input 4.

Following formulation is a simple way for checking whether an input is on logic "1" or "0".

```
A = INP (&H0390)
```

```
IF (A AND &H08) = &H08 THEN
```

```
PRINT "INPUT 4 = 1" ELSE
```

```
PRINT "INPUT 4 = 0"
```

Next formulation allows to verify if digital inputs 10 and 16 are set.

```
B = INP (&H0391)
```

```
IF (B AND &H82) = &H82 THEN
```

```
PRINT "INPUT 10 and 16 = 1" ELSE
```

```
PRINT "INPUT 10 and/or 16 = 0"
```

## 10.2 Outputs

Let's suppose that the address is set on 0390H. The 16 digital outputs are responded with these commands:

```
OUT &H0392, Data
```

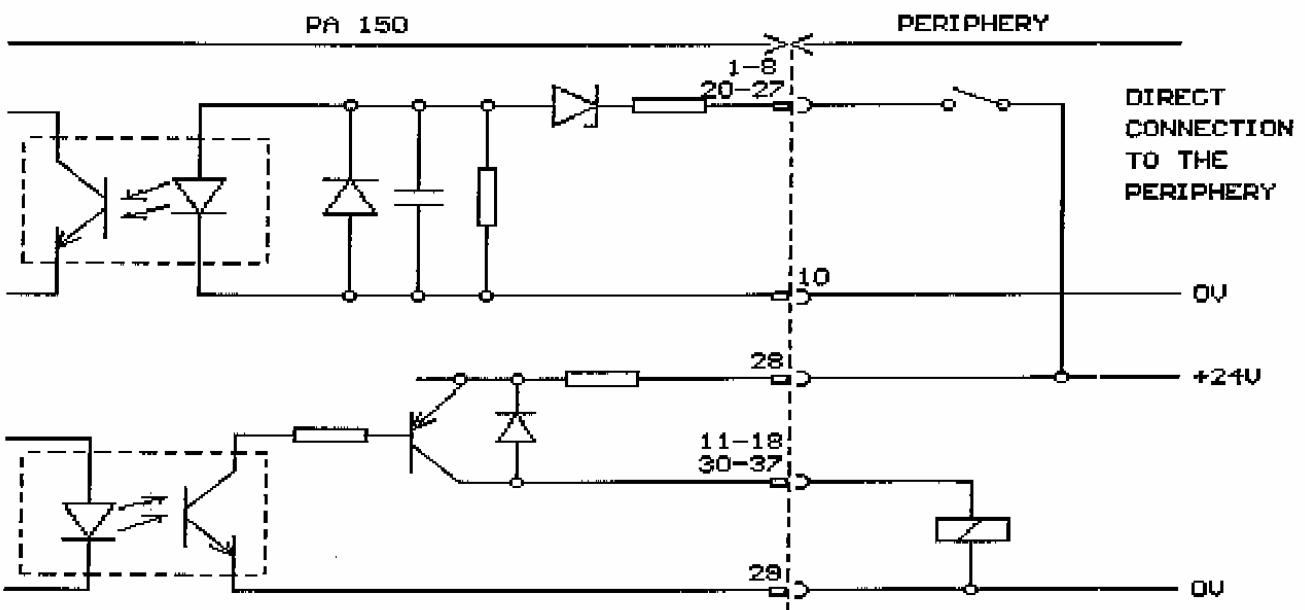
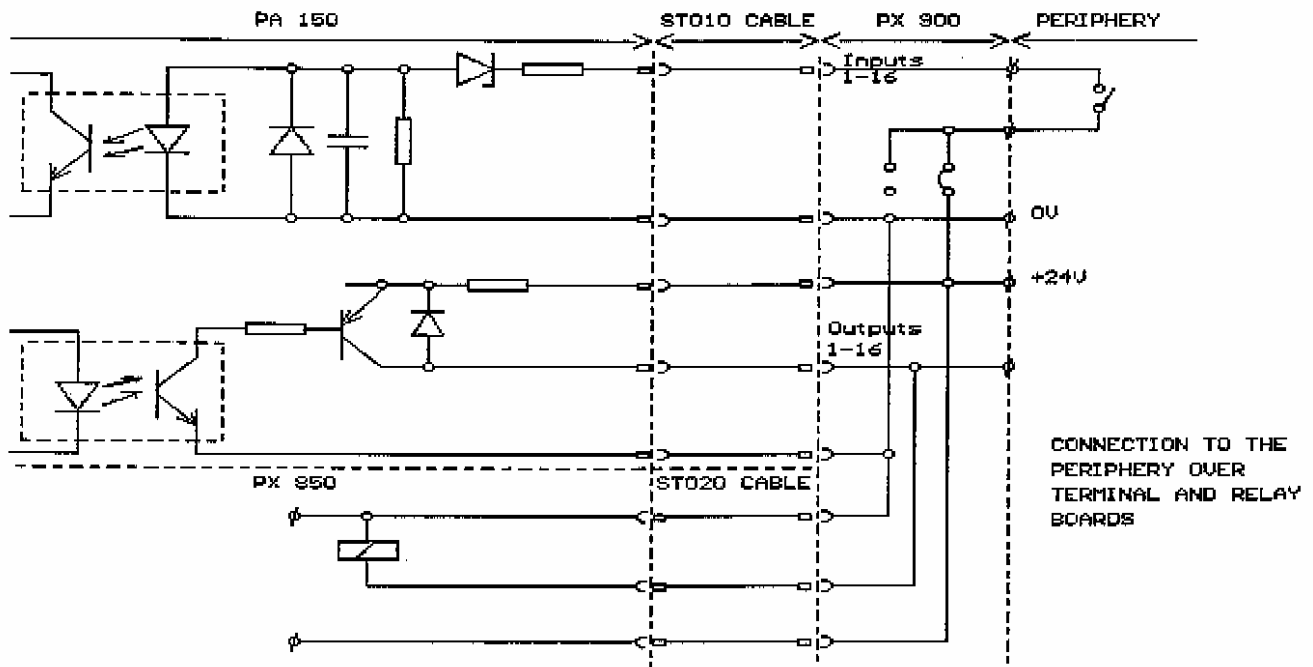
```
OUT &H0393, Data
```

Output data remain on the outputs until they are replaced with another output. The reset occurs also with power ON reset or when the watchdog circuit responds.

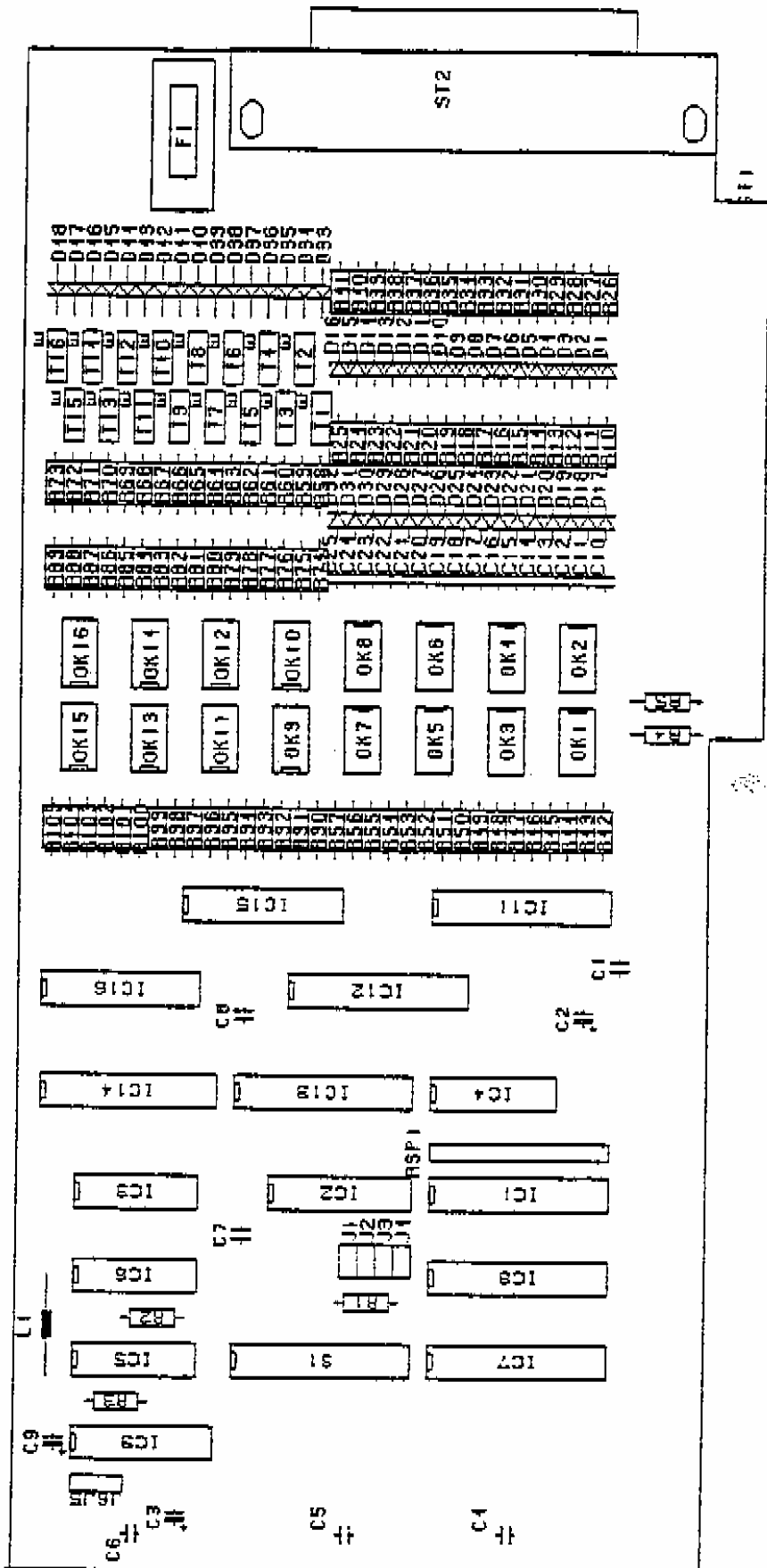
You will find programming examples in chapter 10.5.

# 11 APPENDIX

## 11.1 Circuit configuration

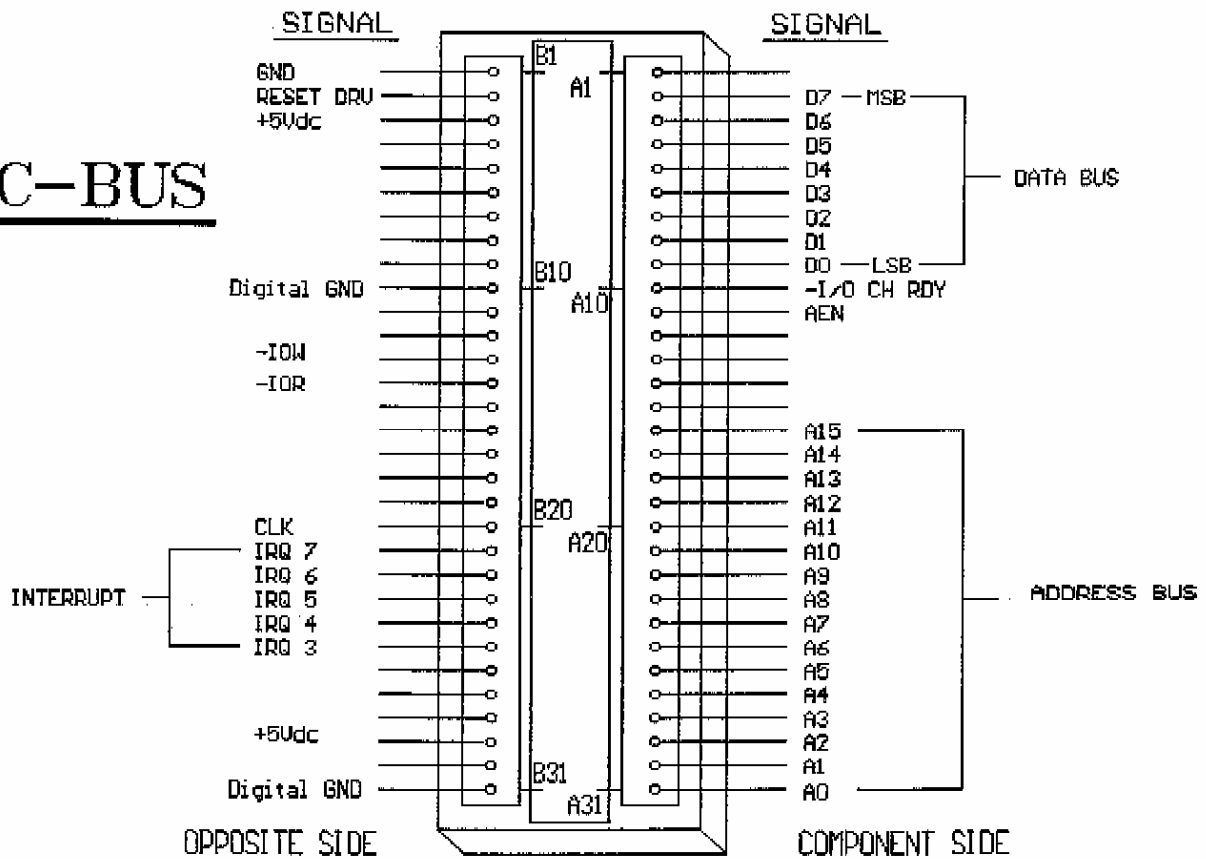


### 11.2 Component scheme



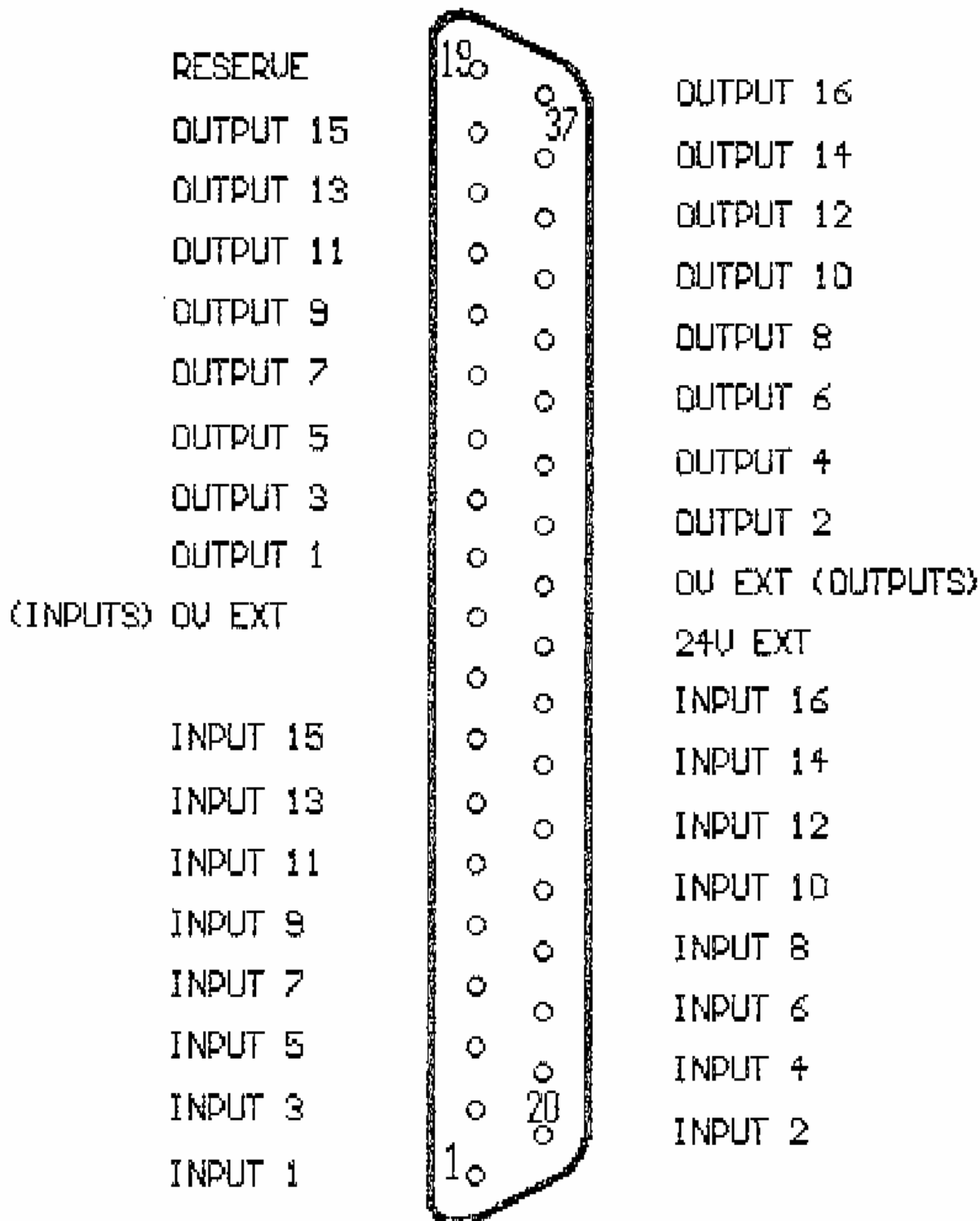
### 11.3 62-pin direct connector

## PC-BUS





### 11.4 37-pin SUB-D connector



## 11.5    **Programming examples: C, Pascal, Assembler, Basic**

```

/* TURBO C                                                    */
/* Reads digital inputs of a PA150 board                      */
/* Used compiler: TURBO C version 2.0 mode SMALL             */
/*
#define Adr_PA150  0x390 /* Basic address of PA150 board      */

void main (void)
{
  char in_LSW,
  char in_MSW;

  int  in_Word;

/* Digital inputs are read either by two readings of 8 bits or by one reading */
/* of 16 bits                                                                    */

  in_LSW = inportb (Adr_PA150);
  in_MSW = inportb (Adr_PA150 + 1);
  in_Word = inport (Adr_PA150);
}

(* TURBO PASCAL                                             *)
(* Reads digital inputs of a PA150 board                    *)
(* Used compiler: TURBO PASCAL version 4.0                 *)

PROGRAM  PA150;
USES DOS;
CONST
  basic_adr_PA150 = $390;      (* Basic address of PA150 board*)
VAR in_1 : BYTE;
    in_2 : BYTE;
    in_3 : INTEGER;
                                (*      **** Main program **** *)
BEGIN

(* Digital inputs are read either by two readings of 8 bits or by one
reading *)
(* of 16 bits                                                                    *)

  in_1 := PORT [basic_adr_PA150];
  in_2 := PORT [basic_adr_PA150 + 1];
  in_3 := PORTW [basic_adr_PA150];

END.

```

```

; TASM
; Reads digital inputs of a PA150 board
; Used compiler: TASM version 1.0 mode SMALL

Basic_Address_PA150 = 390h          ; Basic address of PA150 board

DOSSEG
.MODEL SMALL
.STACK 100hdd
.DATA
in_1  db  ?
in_2  db  ?
in_3  dw  ?

.CODE
mov  ax,@data
mov  ds,ax          ; Initialises data segment

; Example 1 :
; Reading occurs when 8 bits have been read twice

mov  dx,Basic_Address_PA150
in   al,dx
mov  [in_1],al      ; Bit 0 of byte in_1 is
inc  dx            ; digital input number 1
in   al,dx
mov  [in_2],al      ; Bit 7 of byte in_2 is
                        ; digital input number 16

; Example 2 :
; Reading occurs when 16 bits have been read once

mov  dx,Basic_Address_PA150
in   ax,dx
mov  [in_3],ax      ; Bit 0 of word in_3 is
                        ; digital input number 1
                        ; Bit 15 of word in_3 is
                        ; digital input number 16

mov  ah,4ch         ; Back to DOS
int  21h
END

' TURBO BASIC
' Reads digital inputs of a PA150 board
' Used compiler: TURBO BASIC version 1.00

Basic_AddressPA150% = &H390          ' Basic address of PA150 board

' Reading occurs when 8 bits have been read twice

in_1% = inp (Basic_AddressPA150%)   ' Bit 0 of word
                                        ' in_1 is digital input number 1
in_2% = inp (Basic_AddressPA150% + 1) ' Bit 7 of word
                                        ' in_2 is digital input number 16

```

```

/* TURBO C                                                    */
/* Writes digital outputs of a PA150 board                    */
/* Used compiler : TURBO C version 2.0 mode SMALL            */

#define Basic_Adr_PA150  0x390          /* Basic address of PA150 board */

void main (void)
{

/* Example 1                                                    */
/* Writing occurs in 2 x 8 bits                                  */

    outportb (Basic_Adr_PA150 + 2, 0xAA) ; /* Bit 0 value 0 x AA on digital */
                                           /* output number 1          */

        outportb (Basic_Adr_PA150 + 3, 0x55); /* Bit 7 of value 0 x 55 on */
                                           /* digital output number 16 */

}

(* TURBO PASCAL                                                    *)
(* Writes digital outputs of a PA150 board                    *)
(* Used compiler: TURBO PASCAL version 4.0                    *)

PROGRAM  PA150;

USES  DOS;

CONST
    Basic_Adr_PA150 = $390; (* Basic address of PA150 board  *)

BEGIN
    (*      ****  Main program  ****      *)

    (* Example 1 :                                                    *)
    (* Writing occurs in 2 x 8 bits                                  *)

        PORT [Basic_Adr_PA150 + 2] := $0AA; (* Bit 0 of value $0AA on digital *)
                                           (* output number 1          *)
        PORT [Basic_Adr_PA150 + 3] := $055; (* Bit 7 of value $055 on digital *)
                                           (* output number 16        *)

END.

```

```
; TASM
; Writes digital outputs of a PA150 board
; Used compiler: TASM version 1.0 mode SMALL
```

```
Basic_Address_PA150 = 390h ; Basic address of PA150 board
```

```
DOSSEG
.MODEL SMALL
.STACK 100h
.DATA

.CODE
mov ax,@data
mov ds,ax ; Initialises data segment
```

```
; Example 1 :
; Writing occurs when 8 bits have been written twice
```

```
mov dx,Basic_Address_PA150 + 2

mov ax,0AA55h
out dx,al ; Bit 0 of value "AL" on
digital ; output number 1

mov al,ah
inc dx
out dx,al ; Bit 7 of value "AL" on
digital ; output number 16

mov ah,4ch ; Back to DOS
int 21h
END
```

```
' TURBO BASIC
' Writes digital outputs of a PA150 board
' Used compiler: TURBO BASIC version 1.00
```

```
Basic_AddressPA150% = &H390 ' Basic address of PA150 board
```

```
' Writing value &H0AA on Less Significant Byte
```

```
' Writing value &H055 on Most Significant Byte
```

```
out Basic_AddressPA150% + 2, &H0AA ' Bit 0 of value &H0AA
' on digital output number 1
out Basic_AddressPA150% + 3, &H055 ' Bit 7 of value &H055
' on digital output number 16
```

### 11.6 Connection examples

