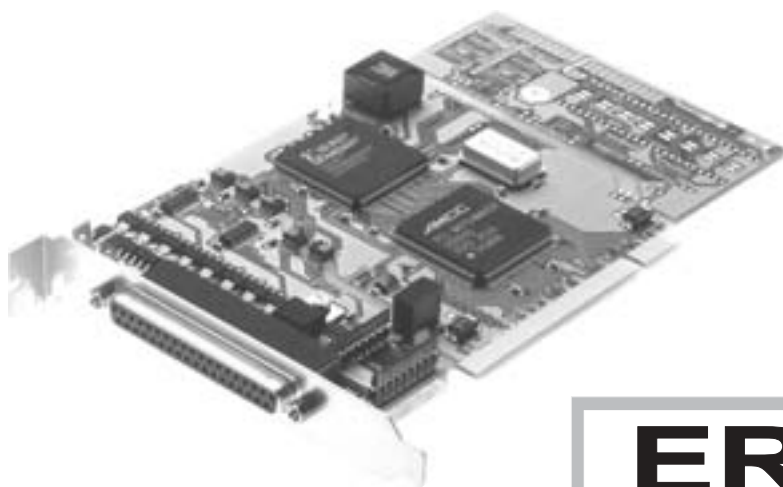
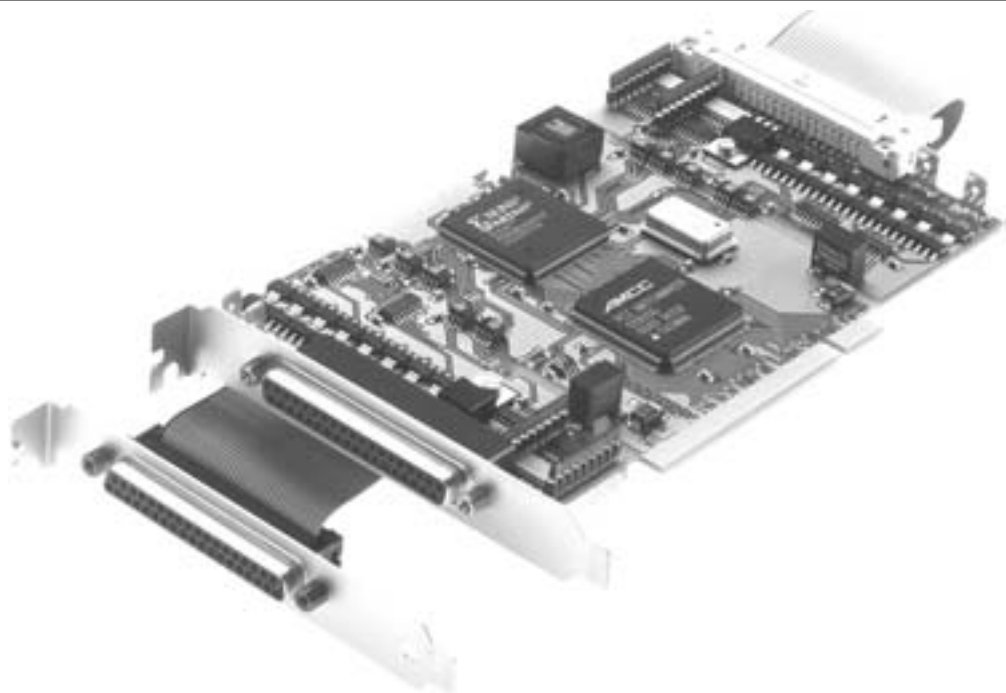

IO 1388 Digital IO-Card

Manual



ERMA

Electronic GmbH

Warranty

For delivered products our "Allgemeine Lieferungs- und Zahlungsbedingungen" are effective. In no event ERMA-Electronic or its suppliers shall be liable for any other damages whatsoever (including, without limitation, damages for loss of business profits, business interruption or other pecuniary loss) arising out of or inability to use this product.

All products from ERMA-Electronic are warranted against defective material and workmanship for a period of two (2) years from date of delivery. If it is necessary to return the product to ERMA, the sender is responsible for shipping charges, freight, insurance and proper packaging to prevent breakage in transit. ERMA's warranty does not apply to defects resulting from action of the buyer, such mishandling, improper interfacing, operation outside of design limits, improper repair or unauthorized modification.

Trademarks

Turbo Pascal, Delphi are registered trademarks of Borland International, INC.

MS-DOS, Windows, Visual Basic are registered trademarks of Microsoft Corporation.

IBM, PC XT/AT, OS/2 are registered trademarksin of the International Business Machines Corporation.

All other trademarks named or portrayed in the text are registered trademarks of its ownwer and are recognized by ERMA-Electronic.

CONTENTS

1. Safety Instructions	5
2. Symbol Explanation	5
3. General	6
4. Function	6
5. Block diagram	7
6. Hints against noisy environment	7
7. Installation	8
7.1. Hardware configuration	8
7.1.1. Component layout	8
7.1.2. Digital Input Channels	9
7.1.3. Digital Output Channels	10
7.2. Open PC	10
7.3. Card insertion	10
7.4. Close PC	10
7.5. Driver installation	11
7.5.1. Windows 95 / 98 / ME	11
7.5.2. Windows NT4.0	11
7.5.3. Windows 2000	11
7.5.4. Windows XP	11
7.6. Configure software	11

7.7. Test of the card	11
8. Registers of the card IO 1388	12
8.1. Output Register	12
8.2. Input Register	12
8.3. Interrupt Enable Register	12
8.4. Timer-Status-Register	13
8.5. Watchdog-Register	14
8.6. Watchdog-Register	14
8.7. Releasing the IRQ of the Target-Controller	15
8.8. Programming under Windows	15
8.9. Programming with IO 1388-DLL	15
8.9.1. Functions	15
8.10. Usage within own applications	18
8.10.1. Visual Basic	18
8.10.2. Visual C++/ LabWindowsCVI	18
8.10.3. Delphi	19
8.10.4. Tips and Tricks	19
9. Troubleshooting	19
10. Pin assignment	20
10.1. Pin Assignment of the card	20

10.2. Pin assignment of the additional slot	21
11. Technical datas	22
12. Ordering information	23

Date : 28. 03.2003
IO1388AE.PUB
Technical subjects to change

1. Safety Instructions

This instrument is produced in accordance with Class II of IEC 348 and VDE 0411. When delivered the instrument has been tested to meet all functions described. Before installing the instrument please read the mounting and servicing instructions. We have no liability or responsibility to customer or any other person or entity with respect to any liability, loss or damage caused or alleged to be caused directly or indirectly by equipment or software sold or furnished by us. Read the installation instruction carefully. No liability will be assumed for any damage caused by improper installation.

Inspect the instrument module carton for obvious damage. Be sure there are no shipping and handling damages on the module before processing. Do not apply power to the instrument if it has damaged.

The warranty does not apply to defects resulting from action of buyer, such as mishandling, improper interfacing, operation outside of design limits, improper repair or unauthorized modifications.

2. Symbol Explanation



Caution



Attention



Instruction



Tip

Caution: Will be used at **dangerous for life and health !**

Attention: Will cause **damage.**

Instruction: If not noticed, **Trouble** may occur.

Tip: Useful hints for **better operation.**

3. General

The board IO 1388 was designed to interface with the PC's internal PCI-BUS through any available expansion slot. There are 2 different models available:

- model IO 1388 / 0 - supplied with 16 / 16 input / output channels
- model IO 1388 / 1 - supplied with 32 / 32 input / output channels

All input and output channels are optically isolated. In addition input and output voltage levels of 24 V are also available. By this reason all input and output channels can directly connected to a PLC.

The output channels are short circuit protected. Each channel can switch currents up to 1 A.

A watchdog is implemented. In a programmable register a distinct output channel mask configuration can be programmed. If the watchdog timeout time is over, the output channels are set to the predetermined state.

By an interrupt-enable-register the input channels can be programmed to generate interrupts. The interrupts can be generated by a positive-edge or a negative-edge trigger.

In addition there is a timer implemented. The timer e. g. can be used for reading input channels in distinct intervals.

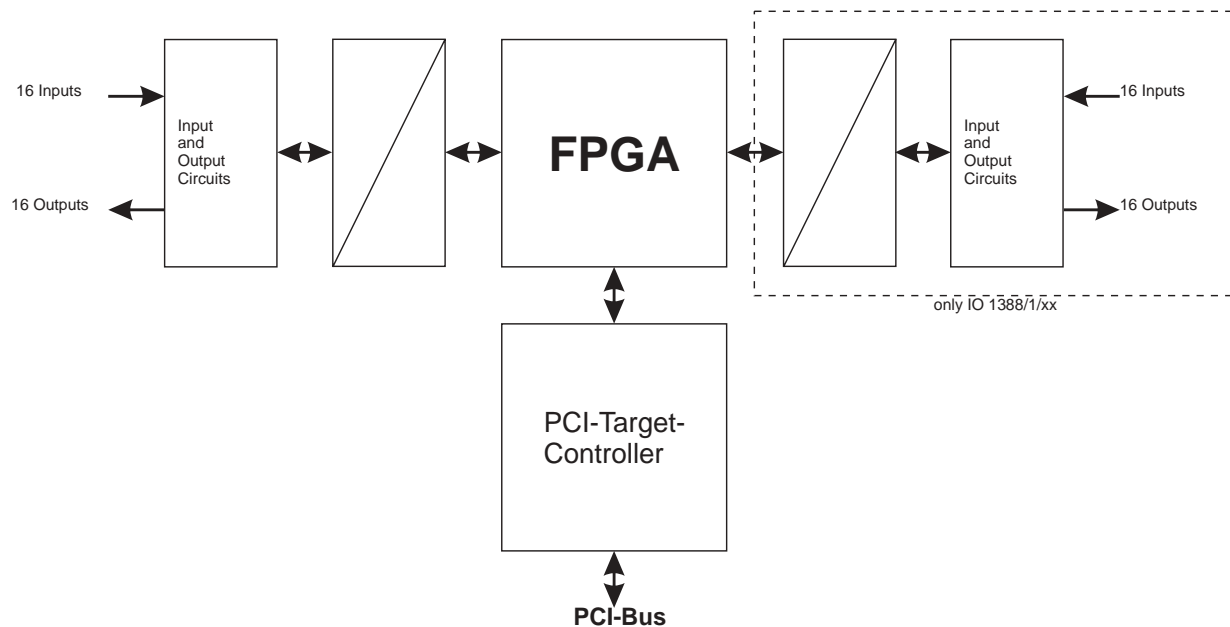
4. Function

The IO 1388 will use one PCI-slot within the pc. Because the PCI-bus supports plug and play, there are no jumpers or switches to change configuration. The card will use 8 x 32-bit-IO-addresses for the functions and 16 x 32-bit-IO-addresses for the PCI-target-controller. You can use any number of IO 1388 within one pc (depends on number of PCI-slots). The standard drivers supports up to 4 cards. If more than 4 cards should be used, you can ask for a driver for more cards. The card has one 37-pin SUB-D male connector for the input and output channels.



The functions of the card will be handled in a FPGA. This free programmable device makes it possible to support customer specific functions without changing the hardware. It is possible to implement additional special functions or fast controllers within the FPGA.

5. Block diagram



6. Hints against noisy environment

All inputs and outputs are protected against noisy environment and high voltage spikes. Nevertheless the location should be selected to ensure that no capacitive or inductive interference can have an effect on the instrument or connection lines.

It is advisable:

- To use shielded, twisted pair cables.
- The wiring of shields and ground (0V) should be star-shaped.
- The distance to interference sources should be as far as possible. If necessary, protective screen or metal enclosures must be provided.
- Coils of relays must be supplied with filters.
- Parallel wiring of input signals and AC power lines should be avoided.



7. Installation



The installation of the card should only be done by qualified personal only. Before installation all components have to be disconnected from power supply. Because within PC's and the peripherals are high voltages it is dangerous to life!

7.1. Hardware configuration

The IO 1388 uses plug and play and so there is no need for any settings.

7.1.1. Component layout

Figure 1 and 2 show the component layout.

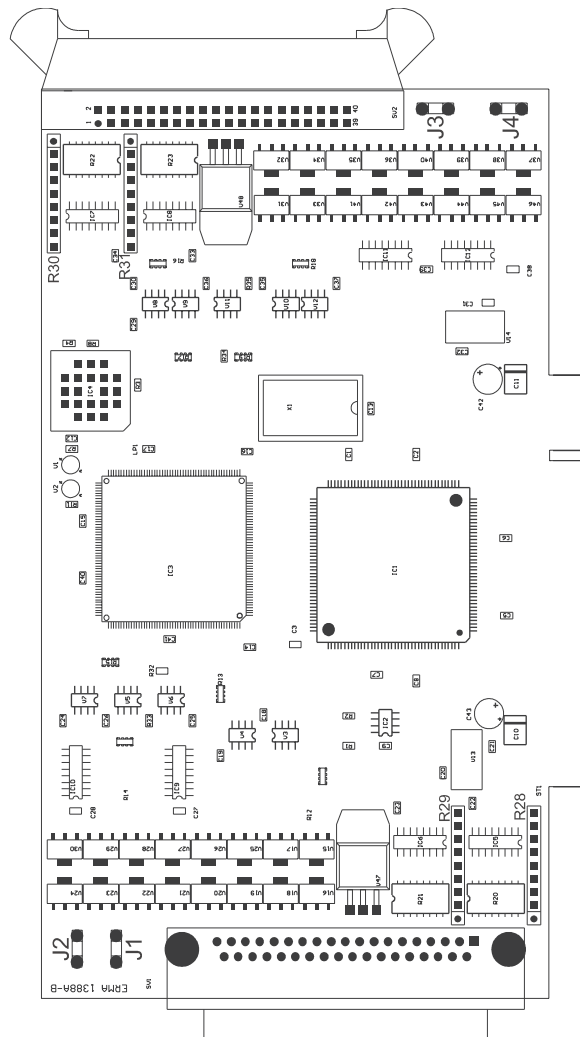


Figure 1

7. Installation

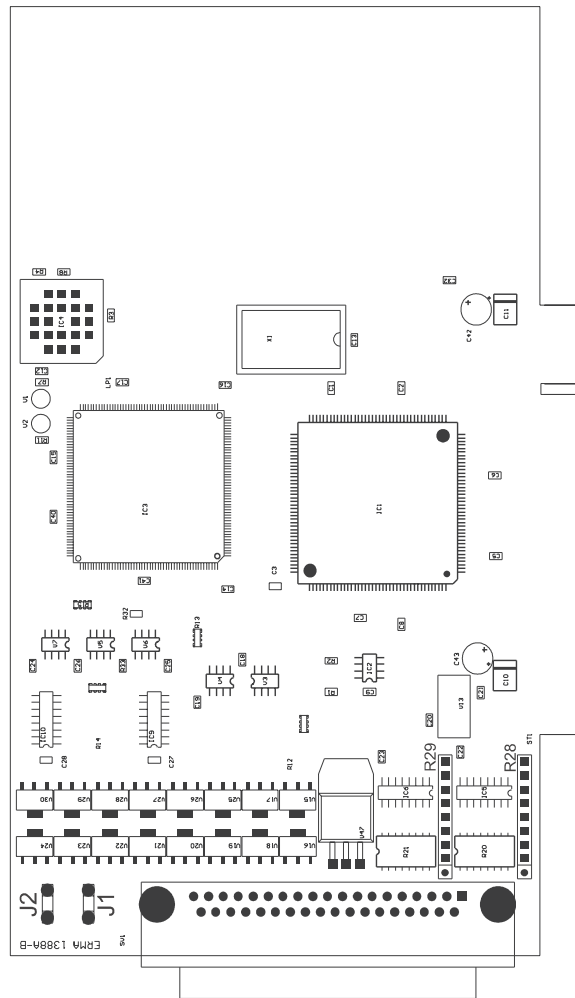


Figure 2

7.1.2. Digital Input Channels

Figure 3 shows one of the input circuits.

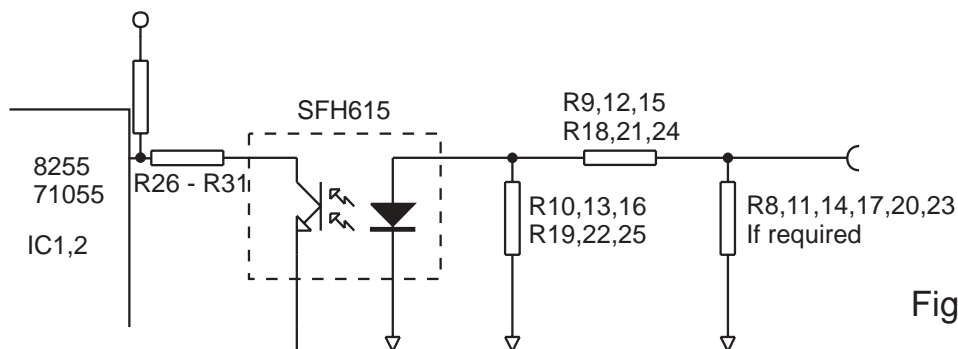


Figure 3

7.1.3. Digital Output Channels

Figure 4 shows one of the output circuits.

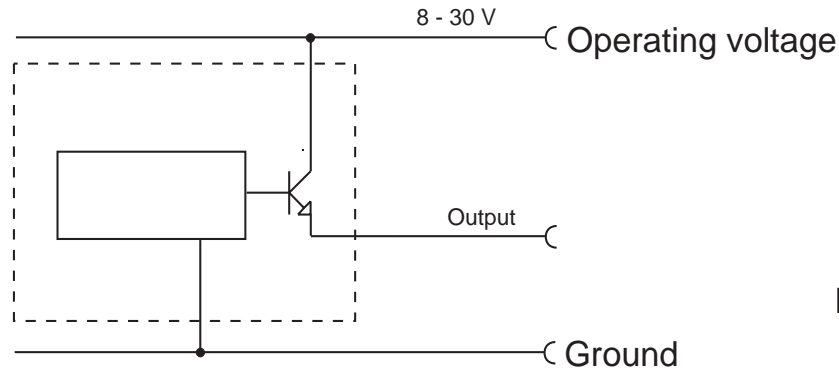


Figure 4

7.2. Open PC

Before opening the pc-case disconnect power supply! The case should be opened as described by the manufacturer of the PC.

7.3. Card insertion

While inserting the card you could be hurt. This is because all components on the cards and in the PC have sharp pins. So this work have to be done carefully. Thee IO 1388 should be placed in a free PCI-Slot. The card should be installed vertically from above. Afterwards the slot plate of the card have to be screwed to the backside of the PC-case.

The slot plate is used for mounting and also for shielding. Keep in mind that removal of the plate will result in loss of shielding and card and PC will be more sensitive to EMC. Additionaly the card won't be held in slot-position when external force will happen to the connected cables. Then the card could move within the slot and that can result in damage to the IO 1388 and the PC! So, don't remove the slot plate!

7.4. Close PC

The case should be closed as described by the manufacturer of the PC.

7.5. Driver installation

On most Windows-versions you must have administrator rights to install drivers.

7.5.1. Windows 95 / 98 / ME

Windows will recognize the new card and ask for the driver. The driver is on the disc within the directory Driver\Win9x.



ATTENTION: Windows 95 does not support PCI-SubvendorID and SubsystemID of the PCI-BUS. So it is possible that there can occur problems, if two different cards (for instance 1 x IO 1388 and 1 x PCI 1398-S01) will be used. In this case please use Windows 98 or Windows ME.

7.5.2. Windows NT4.0

You have to start the program SETUP.EXE. The program is on the disc within the directory Driver\WINNT40. The program will install the driver and the DLL.

7.5.3. Windows 2000

Windows will recognize the new card and ask for the driver. The driver is on the disc within the directory Driver\Win2000.

7.5.4. Windows XP

Windows will recognize the new card and ask for the driver. The driver is on the disc within the directory Driver\WinXP.

7.6. Configure software

All example-programs are shipped as executable and source-code. With the "Conf"-button you can assign encoder-numbers to each encoder. All example-programs will only work with encoders 1 and 2.

7.7. Test of the card

After installation the card could be tested.
Therefore you can use the example-programs on the disc.

8. Registers of the card IO 1388

The card will use 8 x 32-bit-IO-addresses for the function and 16 x 32-bit-IO-addresses for the PCI-target-controller. The address range for the PCI-target-controller is addressed by the base address0, the address range for the functions by the base address1. These addresses you can get from the PCI configuration section (see BIOS.functions for the PCI-BUS). The functions of the card can be controlled by read or write commands with these addresses.

8.1. Output Register

Read / Write

BaseAddress1 + 0	output channel write / read back
------------------	----------------------------------

Each bit of the 32-bit-register is controlling one output channel. A bit = 0 will generate a non conducting output, a bit = 1 will cause a conducting state of the output channel, that means, the external applied positive supply voltage is switched through to the output pin.

8.2. Input Register

Read

BaseAdresse1 + 4	read input channels
------------------	---------------------

Each bit of this 32-bit-register corresponds to one input channel. Write commands to this register will have no effect.

8.3. Interrupt Enable Register

Read / Write

BaseAddress1 + 8	Enable for positive-edge trigger write / read
BaseAddress1 + 12	Enable for negative-edge trigger write / read

Each bit of these registers corresponds to one input channel. Positiv or negativ-edge triggering can be selected by the register. Bit = 0 means "no interrupt is activ". If the value of a bit is set to "1", interrupt mode is activ for the selected input channel provided that the IO-interrupt-enable and the global-interrupt-enable is set.

8.4. *Timer-Status-Register*

Read / Write BaseAddress1+16

bit 31	Global-Interrupt-enable
bit 30	IO-Interrupt-Enable
bit 29	Watchdog-Enable
bit 28	Timer-Enable
bit 27	Timer-Status (IRQ by the timer) read only !
bit 26	Watchdog-Status (IRQ by WD) read only !
bit 25	Input-Status (IRQ by input signal) read only !
bit 24	Configuration of the card
bit 23 - 16	not used
bit 15 - 0	Timer-Value (>0)

By the “Global-Interrupt-Enable-bit” the interrupts can be made active or not active.

Attention must be paid that for enabling interrupts. Not only the “Global-Interrupt-Enable-Bit” must be set but also the “Interrupt-Enable-Bit” of the PCI-Target-Controller !

The bits “Timer-Enable” and “Input-Enable” are used to activate the timer respectively the watchdog.

The 3 bits for the status are indicating which function has generated an interrupt.

The “Configuration-bit” is used to recognize the used IO card.

“Configuration-Bit” = 0 means model **IO 1388/ 0** (16 input / output channels),
 “Configuration-bit” = 1 model **IO 1388/1** (32 input / output channels). The bits 24 - 16 should be set to “0”.

The bits 15 - 0 are used to program the timer. The value must be 1 or higher.

The measuring rate is:

$$f = 20 \text{ kHz} / (\text{timer value})$$

Attention! The shortest measuring rate is determined by the used system, because Windows is not real time capable. The timer can be disabled by the “Timer-Enable-Bit” or the “Global-Interrupt-Enable-Bit”.

8.5. Watchdog-Timer-Register

Read / Write

BaseAdresse1 + 20	timer value for watchdog, read / write
-------------------	--

The watchdog is a 16-bit counter (bit 15 - 0). When programming the watchdog-register, the bits 31 - 16 should be set to "0".

The bits 15 - 0 are used to program the timer. The value must be 1 or higher. The time-out time is:

$$t = 1 \text{ ms} * (\text{watchdog-counter-value})$$

Every command to the card is resetting the watchdog counter. If the host stops sending commands and the programmed timeout time is exceeded, the output channels of the IO 1388 card are set to the programmed output configuration of the watchdog-register.

The watchdog-timer-interrupt can be disabled by the "Watchdog-Enable" or the "Global-Interrupt-Enable".

8.6. Watchdog-Register

Read / Write

BaseAdresse1 + 24	value for watchdog-time-out, read / write
-------------------	---

Each bit corresponds to an output channel (IO 1388 / 0 bits 15 - 0, IO 1388 / 1 bits 31 - 0)

8.7. Releasing the IRQ of the Target-Controller

Using the software IO 1388-DLL, releasing of the IRQ of the target controller must not be observed, because the DLL will do it. Only if the customer uses an other driver it can be of interest.

Releasing of interrupts:

Write the value 0x00002C0C to BaseAddress0+56

Closing of interrupts:

Write the value 0x00000C0C to BaseAddress0+56

8.8. Programming under Windows

To build applications for Windows the disc includes different drivers for all Windows-versions. To keep the programming of the IO 1388 as easy as possible, all functions of the drivers are handled within one DLL. So all programming of the card is done by using the DLL.

The file ERMA_SSI.DLL should be copied to the Windows system directory or the application directory.

8.9. Programming with IO 1388-DLL

The DLL IO 1388-DLL includes all functions necessary for the work with the IO 1388.



The return value $\neq 0$ of all functions (except GetLastError) signalize a correct performance. The function IO1388_GesLastError is used to report an error-code.

Programs written for using the IO1388-DLL will run under Windows 9x / ME, Windows NT 4.0 / 2000 / XP. Therefore only the DLL and the corresponding driver is necessary.

8.9.1. Functions

General functions

char IO1388_InitDriver()

The function must be called once at the beginning of the program. The function initialises the card, the drivers, and the DLL.

char IO1388_DelnitDriver()

The function IO1388_DelnitDriver must be called once at the end of the program. All pending datas will be erased and all interrupts closed.

char IO1388_GetCardCount(unsigned char *number)

This function returns the number of installed IO 1388 cards.

char IO1388_GetCardNumber (*unsigned char Card, unsigned long *number*)

This function returns the production number of the IO 1388 card. This is used and necessary, if more than one IO 1388 card is installed in the same pc. That's why the BIOS and the operating system when booting can differently allocate the cards. So it is necessary to identify the cards by there production number.

int IO1388_GetLastError ()

If one of the IO 1388 functions returns a failure, the function **IO1388_GesLastError** is used to return the error-code. Following error codes are available:

IO1388_ERR_OK	no failure
IO1388_ERR_ALREADYOPEN	already initialised
IO1388_ERR_NOTOPEN	not initialised
IO1388_ERR_NOCARD	wrong number of the card
IO1388_ERR_VXDNOTFOUND	Win9x/ME-driver not found
IO1388_ERR_DRVNOTFOUND	WinNT4.0/2000-driver not found
IO1388_ERR_INTERNALERROR	internal DLL/failure of the driver
IO1388_ERR_WRONGVALUE	invalid value
IO1388_ERR_DEVICERUNNING	card is running
IO1388_ERR_DEVICESTOPPED	card is not yet running

char IO1388_WriteOutputs (*unsigned char Card, unsigned long value*)

This function is setting the output channels of the IO 1388. Each bit corresponds to one output channel. (see register)

char IO1388_ReadOutputs (*unsigned char Card, unsigned long *value*)

This function is reading back the state of the output channels of the IO 1388. Each bit corresponds to one input channel. (see register).

char IO1388_ReadInputs (*unsigned char Card, unsigned long *value*)

This function is reading the input channels of the IO 1388. Each bit corresponds to one input channel. (see register).

char IO1388_ReadIntRise (*unsigned char Card, unsigned long *value*)

This function is reading the register for positive-edge triggering of the IO 1388. Each bit corresponds to one input channel. (see register).

char IO1388_WriteIntRise (*unsigned char Card, unsigned long value*)

This function is writing the register for positive-edge triggering of the IO 1388. Each bit corresponds to one output channel. (see register).

char IO1388_ReadIntFall (*unsigned char Card, unsigned long *value*)

This function is reading the register for negative-edge triggering of the IO 1388. Each bit corresponds to one input channel. (see register).

char IO1388_WriteIntFall (*unsigned char Card, unsigned long value*)

This function is writing the register for negative-edge triggering of the IO 1388. Each bit corresponds to one output channel. (see register).

char IO1388_ReadTimerStatus (*unsigned char Card, unsigned long *value*)

This function is reading the timer-status-register of the IO 1388 (see register).

char IO1388_WriteTimerStatus (*unsigned char Card, unsigned long value*)

This function is writing the timer-status-register of the IO 1388 (see register).

char IO1388_ReadWdTimer (*unsigned char Card, unsigned long *value*)

This function is reading the timer-value of the watchdog counter (see register).

char IO1388_WriteWdTimer (*unsigned char Card, unsigned long value*)

This function is writing the timer-value of the watchdog counter (see register).

char IO1388_ReadWdValue (*unsigned char Card, unsigned long *value*)

This function is reading the watchdog-value of the watchdog counter (see register).

char IO1388_WriteWdValue (*unsigned char Card, unsigned long value*)

This function is writing the watchdog-timeout-time of the watchdog counter (see register).

char IO1388_StartIRQ (*unsigned char Card*)

This function is starting the interrupt recording. Once started the function **IO1388_ReadTimerStatus** should no more be used. Before starting the interrupt mode the desired values should be written to the appropriate registers. If there are any datas in the FIFO, they will be removed.

char IO1388_StopIRQ (*unsigned char Card*)

This function is stopping interrupt recording. Datas are stored in the FIFO until red or recording is started again or the driver is deinitialised.

char IO1388_GetIRQStatus (*unsigned char Card, unsigned long *value*)

This function returns the state of the interrupt recording. The following messages are sent:

IO1388_RUNNING	Interrupt recording is started
IO1388_IOINT	Since the last command an IO-Interrupt has occurred
IO1388_TMINT	Since the last command a Timer-Interrupt has occurred
IO1388_WDINT	Since the last command a Watchdog-Interrupt has occurred
IO1388_VALUE	There are data in the FIFO
IO1388_FIFOOV	Since the last command an overflow of the FIFO has occurred (data loss)

char IO1388_GetIRQInputs (unsigned char Card, long (*buffer)[], unsigned long buffersize, unsigned long *n)

This function is reading data from the FIFO in packages. This function should be used when the function IO1388_GetIRQStatus returns the Status IO1388_VALUE.

Hints: buffersize and n are not the values expressed through bytes but by the number of values !

8.10. Usage within own applications

8.10.1. Visual Basic

Within directory Include\VB on the cd there is the file ERMA_SSI.BAS. If this file is included within a VisualBasic-project, all DLL functions for IO 1388 are available in the project.

8.10.2. Visual C++ / LabWindowsCVI

For C/C++ applications there are an import library ERMA_SSI.LIB and an header-file ERMA_SSI.H. The library has to be inserted in the linker-options of the project-settings (see documentation of VisualC++/LabWindowsCVI). Include the header-file in all source-files where IO 1388 functions are needed.

An additional method is to use the Windows-function LoadLibrary to connect the application to the DLL. See documentation of VisualC++ / LabWindowsCVI for further information.

8.10.3. Delphi

In the directory Include\Delphi of the cd there is a Delphi unit source file ERMA_SSI.PAS. If this file is inserted within a Delphi-project all DLL-functions of the IO 1388 are available. Insert this unit to all uses-declarations where IO 1388 functions are needed.

8.10.4. Tips and Tricks

- While programming interrupts keep in mind:
 - Set global interrupt enable.
(is done by SSISStart of ERMA_SSI.DLL when used)
 - Set used interrupt-enables
- If drivers are developed, the following points should be considered:
 - While writing to registers, set all unused bits to 0
 - While reading registers, mask out all unused bits.
This will safe compatibility with future changes.

9. Troubleshooting

All PC-boards of ERMA-Electronic GmbH are tested during manufacturing as well as when delivered. Nevertheless it can occur that a new board doesn't work. That must not be a failure of the board. There are many minor details when a new board doesn't work. In that case at first some points should be paid attention to.

- One of the most frequent failures are dirty slots. It is recommended to clean the connector of the board with a cloth and a little spirit.
- Further it can happen that the board doesn't fit correctly into the slot or into the PC case. The dimensions of the board are standardized. But the dimension of the PC cases are sometimes out of the prescribed dimensions.
- The cable of the peripherie should be tested.

If you have observed the hints above and the board doesn't work at all call the ERMA - Team. The ERMA-Team will help you.

10. Pin assignment

10.1. Pin Assignment of the card

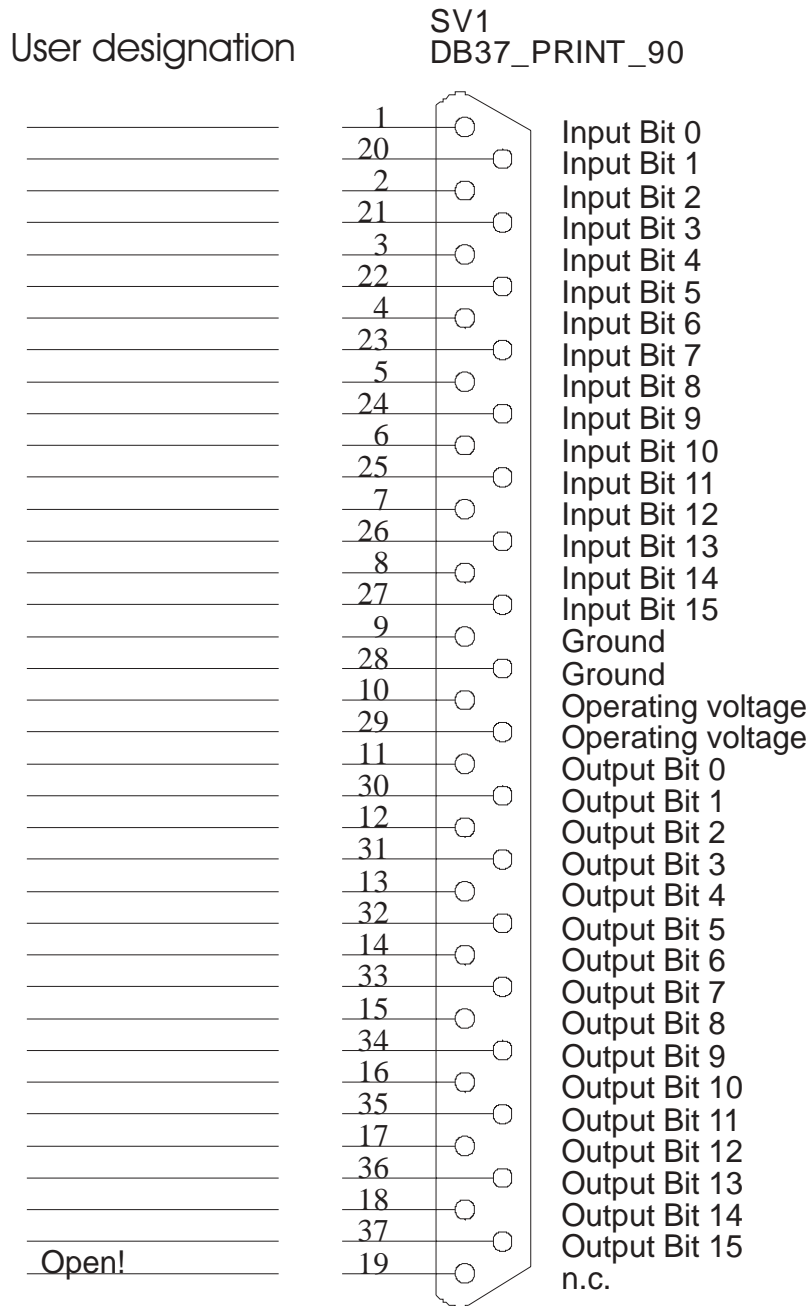


Figure 5

Figure 5 shows the pin connections of the 37 pole SUB-D-connector of the IO 1388

10.2. Pin assignment of the additional slot

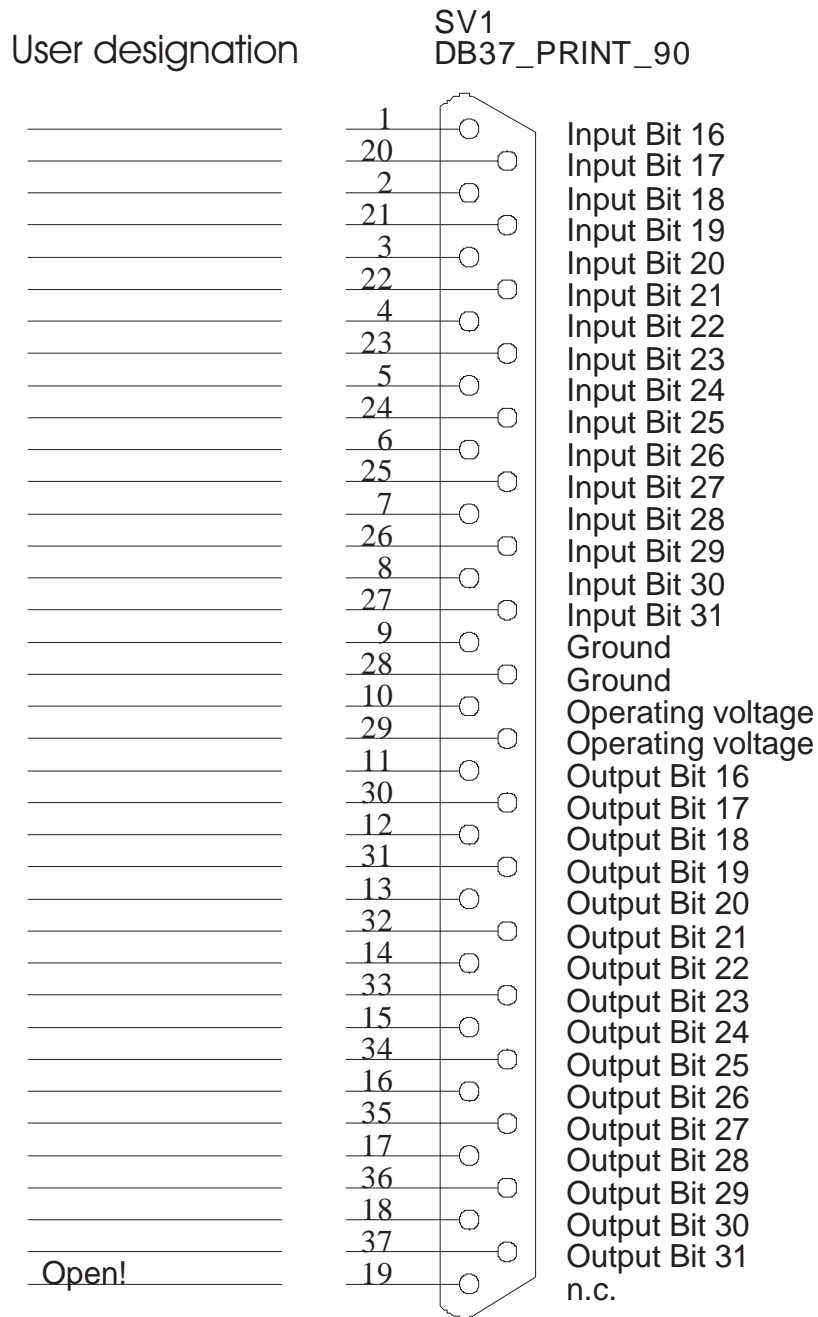


Figure 6

Figure 6 shows the pin assignment of the additional slot connector of the card IO 1388 / 1.

11. Technical datas

Digital output channels	: 16 / 32
	: high side switching
Voltage	: 8 - 30 V DC
Current	: max. 1 A, short circuit protected
Isolation voltage	500 V DC
Digital input channels	: 16 / 32, high active
Voltage	: 5 / 24 V or customer defined
Impedance	: > 10 k
Isolation voltage	500 V DC
Connectors	: 37 pole SUB-D-female
EMC	: corresponding to 89/336/EWG
Operating Temperature	: 0 bis 40 °C
Storage Temperature	: -25 bis +85°C
Dimensions	: 107 x 190 mm

12. Accessories

There are two cables for high currents applications available. These cables are necessary for currents of more than 10 A DC through the SUB-D-connector respectively if the current through the additional flat cable exceeds 2 A. DC.

KIO 1388-1

This cable is used for the card IO 1388/0/xx for currents of more than 10 A DC. This cable is also suited for the IO 1388/1/xx for increasing the supply current of the flat cable.

KIO 1388-2

This cable is used for the card IO 1388/1/xx for supply currents of more than 10 A DC and 2 A DC

13. Ordering information

IO 1388/	x/	x/	xx	
				Encoder Power Supply
			00	Without power supply
			05	With 5 V power supply
			12	With 12 V power supply
			24	With 24 V power supply
				Backup-Battery
			0	Without battery
			1	With battery
				Galvanic (opto) Isolation
			0	Without galvanic isolation
			1	With galvanic isolation

ERMA - Electronic GmbH
Max-Eyth-Straße 8
78194 Immendingen

Telefon +49 (0)7462 2000-0
Fax +49 (0)7462 2000-29
email erma-electronic@erma-electronic.com
Web www.erma-electronic.com

