User manual for

PC-192

High Performance Digital I/O Board for IBM PC, PC/XT, PC/AT and PS/2 Compatible Computers

First edition.

August 1990

August 1990 Printing

Table of Contents

[This page intentionally left blank.]

Chapter 1

Introduction

The PC-192 is a 196 line parallel digital I/O interface board for the IBM PC/XT/AT, PS/2 Model 25 and 30 and compatible computers. The PC-192 is built around eight industry standard 8255 type Programmable Peripheral Interface (PPI) adapter chips.

1.1. Features

- TTL/CMOS compatible
- Industry standard 50-way connectors
- Jumper selectable wait-state generator allows trouble free operation in high speed systems
- Driver software supplied, complete with source code in C and QuickBasic
- Supplied complete with LabWindows (V1.2) drivers.

1.2. Description

A block diagram of the PC-192 is shown in figure 1.1. Each 8255 device has its own I/O locations dedicated to it, as well as its own output connector.

The PC-192 can plug into any fully bussed slot of a PC/XT/AT or PS/2 model 25 or 30, and uses 32 consecutive I/O addresses. The base address of the PC-192 can be set via the DIP switch on the board.

Also included on the PC-192 is a wait-state generator. This ensures that bus I/O cycles are long enough for the 8255 devices on the board. The wait-state generator can either be enables or disabled via the on-board DIP switch. A utility program is provided which tests the operation of the PC-192 to determine if the wait-state generator is required.

Provision for obtaining +5 volt power from the host PC is made on the I/O connectors. Jumpers are provided for each individual connector, allowing the host PC's +5 volt power to either be used, or isolated from the connector.

Connection to the PC-192 is made via 50 pin insulation displacement connectors. The required connectors are 3M type 3425-6050 or equivalent.

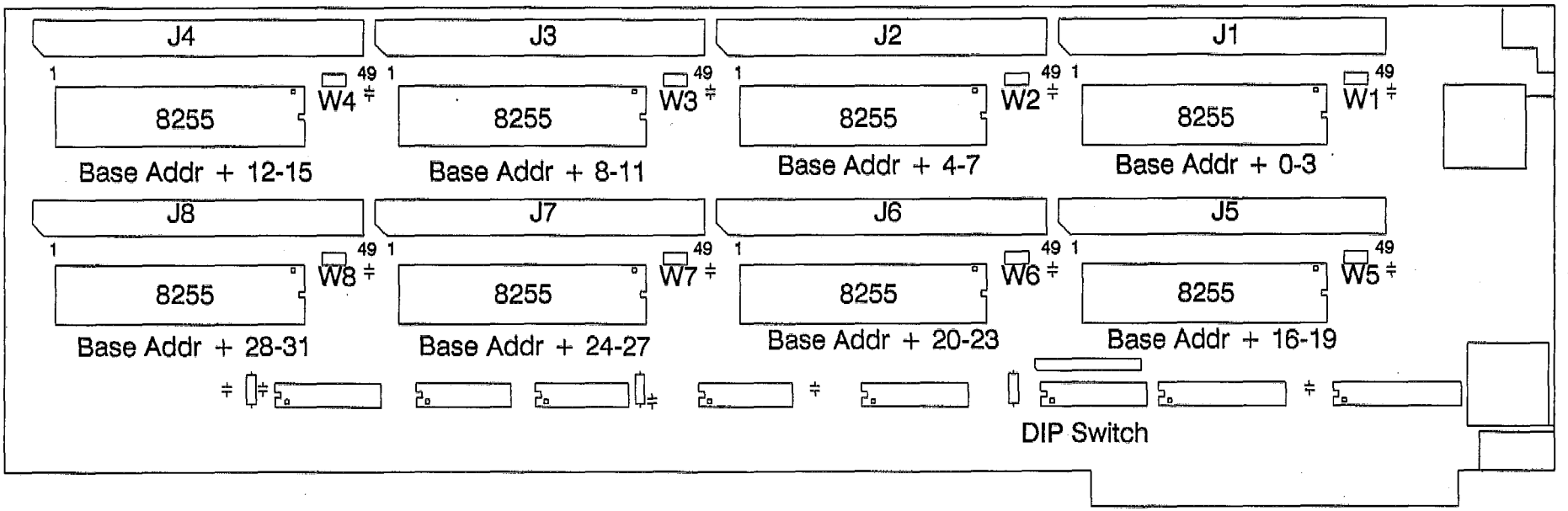

Figure 1.1. PC-192.

[This page intentionally left blank.]

Chapter 2

Installation

This chapter describes how to configure the PC-192, how to install it into your computer, and how to test the installation.

2.1. Configuration

There are three aspects of the PC-192's operation that can be configured. These are:

- Base address. The base address is where the computer will find the PC-192. This is set via seven of the switches on the DIP switch block.
- Wait state generation. If your computer generates abnormally fast bus cycles, it may be necessary to slow these down. The PC-192 has the ability to this, without affecting I/O cycles that are not directed at it. This is controlled by one of the switches on the DIP switch block.
- +5 Volt power. The PC-192 can supply a limited amount of power to external devices. This can be enabled or disabled individually for each I/O connector via a jumper.

2.1.1. Base address setting

The PC-192 uses a block of 32 8-bit I/O address locations, corresponding to 1F locations in HEX. The base address setting controls the address at which this block begins. This must be chosen such that the address block used by the PC-192 does not overlap any other device.

Base address on the PC-192 may be assigned to any location from 000 to 7E0 HEX. The I/O address used by standard devices are listed below:

Address	Device
000-1FF	Internal system board
200-20F	Games port
210-217	Expansion unit
220-24F	Reserved
250-257	Not assigned
258-25F	Intel "Above Board"
260-277	Not assigned
278-27F	Reserved
280-2EF	Not assigned
2F0-2F7	LPT2:
2F8-2FF	COM2:
300-31F	Prototype board
320-32F	Hard disk
330-377	Not assigned
378-37F	LPT1:
380-38F	SDLC communications
390-39F	Not assigned
3A0-3AF	Binary communications
3B0-3BF	Mono display adapter
3C0-3CF	Reserved
3D0-3DF	Color graphics
3E0-3E7	Reserved
3E8-3EF	Not assigned
3F0-3F7	Floppy disk
3F8-3FF	COM1:
400-7FF	Not used; see note below.

Note that addresses from 400-7FF cannot normally be used, because these addresses are not normally decoded. The PC-192 however (and most other members of the "PC-XX" family) can use these address, if (and ONLY if) the board at the address 400 HEX less than the address of the PC-192 also decodes the extra addresses. For example, two PC-192s can be installed, the first at address 300 HEX and the second at address 700 HEX. Most other members of the "PC-XX" family of boards decode the extra addresses.

If your computer has boards not listed here, such as LAN adapters, back-up boards or other engineering boards, you should consult the manuals for these boards for information on the address ranges used. In most cases, a base address of 300 HEX is a good choice. This is also the factory default base address.

Base Address	Switches						
	SW2	SW3	SW4	SW5	SW6	SW7	SW8
0H	On	On	On	On	On	On	Off
20H	On	On	On	On	On	Off	Off
40H	On	On	On	On	Off	On	Off
60H	On	On	On	On	Off	Off	Off
80H	On	On	On	Off	On	On	Off
A0H	On	On	On	Off	On	Off	Off
C0H	On	On	On	Off	Off	On	Off
E0H	On	On	On	Off	Off	Off	Off
100H	On	On	Off	On	On	On	Off
120H	On	On	Off	On	On	Off	Off
140H	On	On	Off	On	Off	On	Off
160H	On	On	Off	On	Off	Off	Off
180H	On	On	Off	Off	On	On	Off
1A0H	On	On	Off	Off	On	Off	Off
1C0H	On	On	Off	Off	Off	On	Off
1E0H	On	On	Off	Off	Off	Off	Off
200H	On	Off	On	On	On	On	Off
220H	On	Off	On	On	On	Off	Off
240H	On	Off	On	On	Off	On	Off
260H	On	Off	On	On	Off	Off	Off
280H	On	Off	On	Off	On	On	Off
2A0H	On	Off	On	Off	On	Off	Off
2C0H	On	Off	On	Off	Off	On	Off
2E0H	On	Off	On	Off	Off	Off	Off
300H	On	Off	Off	On	On	On	Off
320H	On	Off	Off	On	On	Off	Off
340H	On	Off	Off	On	Off	On	Off
360H	On	Off	Off	On	Off	Off	Off
380H	On	Off	Off	Off	On	On	Off
3A0H	On	Off	Off	Off	On	Off	Off
3C0H	On	Off	Off	Off	Off	On	Off
3E0H	On	Off	Off	Off	Off	Off	Off

Figure 2.1. Base address switch settings.

Base Address	Switches						
	SW2	SW3	SW4	SW5	SW6	SW7	SW8
400H	Off	On	On	On	On	On	Off
420H	Off	On	On	On	On	Off	Off
440H	Off	On	On	On	Off	On	Off
460H	Off	On	On	On	Off	Off	Off
480H	Off	On	On	Off	On	On	Off
4A0H	Off	On	On	Off	On	Off	Off
4C0H	Off	On	On	Off	Off	On	Off
4E0H	Off	On	On	Off	Off	Off	Off
500H	Off	On	Off	On	On	On	Off
520H	Off	On	Off	On	On	Off	Off
540H	Off	On	Off	On	Off	On	Off
560H	Off	On	Off	On	Off	Off	Off
580H	Off	On	Off	Off	On	On	Off
5A0H	Off	On	Off	Off	On	Off	Off
5C0H	Off	On	Off	Off	Off	On	Off
5E0H	Off	On	Off	Off	Off	Off	Off
600H	Off	Off	On	On	On	On	Off
620H	Off	Off	On	On	On	Off	Off
640H	Off	Off	On	On	Off	On	Off
660H	Off	Off	On	On	Off	Off	Off
680H	Off	Off	On	Off	On	On	Off
6A0H	Off	Off	On	Off	On	Off	Off
6C0H	Off	Off	On	Off	Off	On	Off
6E0H	Off	Off	On	Off	Off	Off	Off
700H	Off	Off	Off	On	On	On	Off
720H	Off	Off	Off	On	On	Off	Off
740H	Off	Off	Off	On	Off	On	Off
760H	Off	Off	Off	On	Off	Off	Off
780H	Off	Off	Off	Off	On	On	Off
7A0H	Off	Off	Off	Off	On	Off	Off
7C0H	Off	Off	Off	Off	Off	On	Off
7E0H	Off	Off	Off	Off	Off	Off	Off

Figure 2.2. Base address switch settings (cont).

Base address is controlled by switches 2 thru 8 of the DIP switch block on the PC-192 board. Tables 2.1 and 2.2 show the available base addresses, and the corresponding DIP switch settings.

2.1.2. Wait state generation

Wait state generation is controlled by switch 1 on the PC-192's DIP switch block. If this switch is ON, then wait states are enabled. If the switch is OFF, then wait states are disabled. Only a very small number of computers will require that wait-states be enabled. You can find if your computer requires this as follows:

a) Set switch 1 of the DIP switch to OFF, disabling wait states.

b) Install the PC-192 in your computer. Installation is discussed later in the chapter.

c) Run the PC-192 test program. This is on the disk supplied with your PC-192, and is called "PC192TST.EXE". Note that to run this program, no external devices may be connected to the PC-192's I/O connectors.

d) The program will run for some time. If any errors are reported, then your computer requires wait-states, and you should set switch 1 to the ON position.

e) If errors persist after the switch 1 is on, then the either the PC-192, or the host computer, are defective and should be serviced.

2.1.3. +5 Volt power

Each of the eight I/O connectors has its own individual jumper to enable or disable the supply of +5 V power to external devices.

If this jumper is IN, then power will be supplied to external devices. If the jumper is OUT, then no power will be supplied, and the computer's +5 V supply is isolated from external devices.

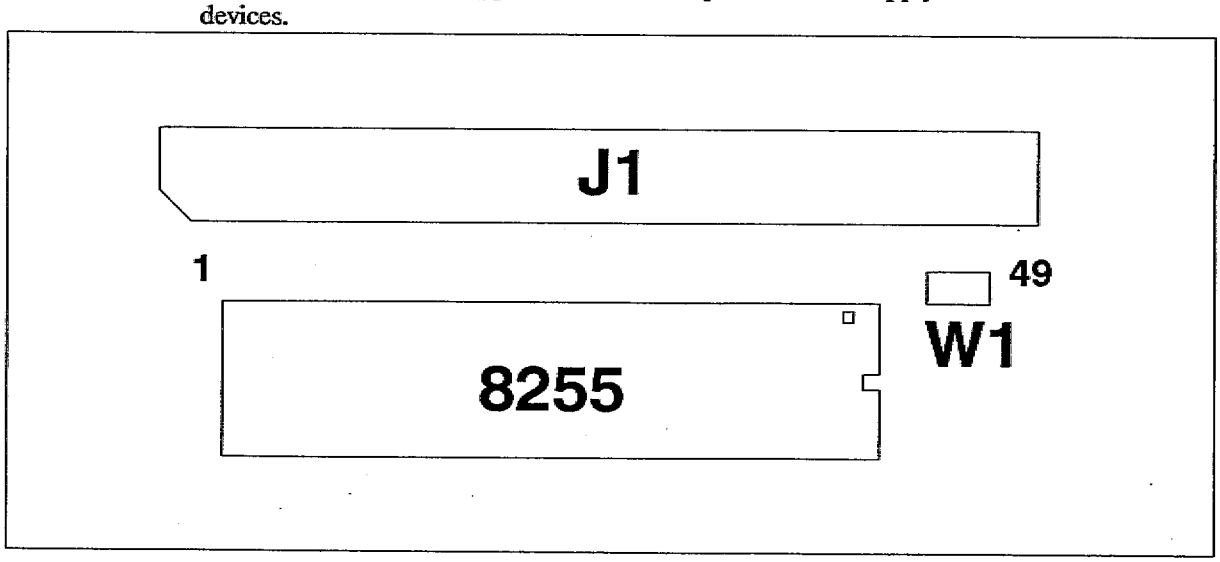

Figure 2.3. +5 V power jumper location.

Figure 2.3 shows the position of the jumper relative to connector J1. The jumper corresponding to connector J1 is labeled W1, that for J2 is labeled W2 etc.

NOTE

The PC-192 can supply a maximum of 100 mA per I/O connector. Exceeding this may cause irreparable damage to the PC-192.

DO NOT connect the +5 V supply from the computer to an external power supply. If an external device has its own power, the corresponding connector's jumper should be removed.

2.2. Installation

In order to install the PC-192 board:

a) Switch off the computer and all attached devices.

b) Unplug the power cord from the computer and all attached devices.

WARNING

Failure to disconnect all power cables can result in hazardous conditions, as dangerous voltage levels may be present in externally connected cables.

---

c) Remove the cover from the PC. If you are not sure how to do this, consult the manual supplied with your system unit.

d) Ensure that the jumper and DIP switch settings on the PC-192 are appropriate for your application.

e) Install all cables required by your application. The cables should be routed out through the bracket. If are using several cables, you may find it easier to use the an adjacent open slot to route some of the cables. NOTE: If you intend to test the PC-192 as described in the next section, no cables should be plugged in.

f) Choose an unused option slot. You may find it easier to choose a slot with an additional empty slot to the right.

g) Loosen the screw at the top of the blank bracket installed in the chosen slot, and remove the blank bracket.

h) Feed the cables through the rear of the computer via the slot.

i) Align the gold plated edge connector with the edge socket and the rear adapter slot with the board bracket. Firmly press the board down into the socket on the computer's system board. Ensure that the board's edge connector is in the socket, and has not slipped sideways past the socket.

j) Re-install the bracket's screw, and replace the computer's cover.

k) Plug in all cords and cables. Turn the power back on. The PC-192 is now installed.

2.3. Testing the PC-192

The PC-192 comes complete with a comprehensive diagnostics package. This is the program "PC192TST.EXE" and may be found on the disk supplied with your PC-192.

In order to use this program, the following condition must be met:

- All cables must be unplugged from the PC-192's I/O connectors.

To use the test program, proceed as follows.

a) Insert the supplied disk into your computer's A: drive

b) If you have NOT changed the base address from the factory default setting of 300 HEX, type the following in

A:PC192TST

c) If you have changed the PC-192 base address, then type in the following:

A:PC192TST [*Base Address*]

where the base address is given in HEX. For example, if the base address is set to 700 HEX, type in:

A:PC192TST 700

The program will run for several minutes, comprehensively checking the operation of the PC-192. If errors are found, the program will suggest possible steps to correct the problem.

[This page intentionally left blank.]

Chapter 3

Connections

This chapter describes the PC-192's I/O connectors, and how to make connections to the PC-192.

3.1. Connector

All connections to the PC-192 are made via eight 50 pin type plugs. Each connector carries the digital I/O lines from one of the eight 8255 devices on the board. The pin-out of all eight connectors is shown in figure 3.1. Figure 1.1, in chapter 1, shows the layout of the connectors on the PC-192 board.

3.2. Making connections

The digital I/O lines on the PC-192 are directly compatible with most forms of TTL and CMOS logic operating from 5 V supplies. Detailed specifications on the digital I/O lines may be found in chapter 6.

The +5 V power line on each connector may be individually enabled or disabled by jumpers. This is discussed in chapter 2.

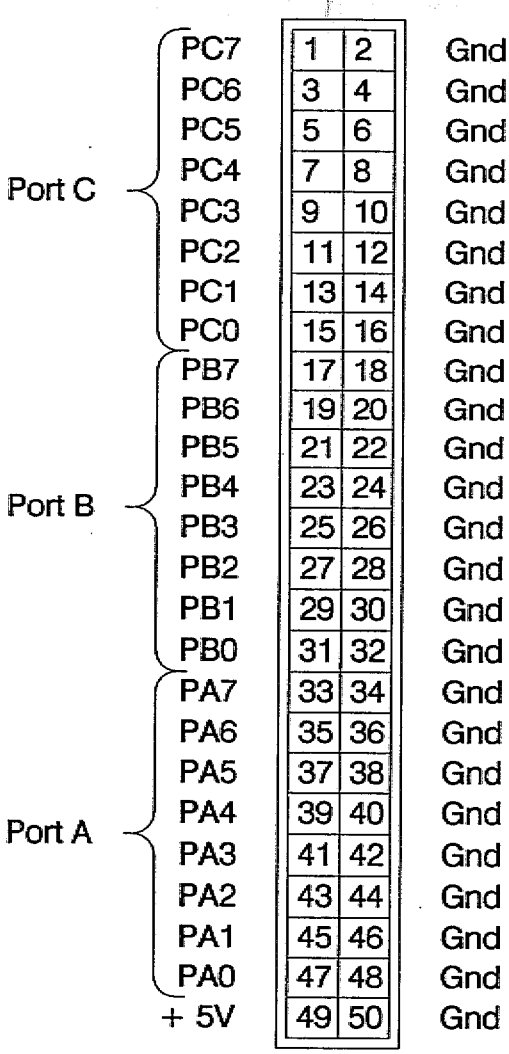

Figure 3.1. PC-192 I/O connectors.

WARNING

No more than 100 mA of current may be drawn from the +5 V power line.

If the device to which a PC-192 I/O connector is to be coupled has its own +5 V power, then that I/O connector's +5 V power jumper MUST be removed.

Chapter 4

Programming

This chapter describes how to program the PC-192.

4.1. Low level programming

At the lowest level, the PC-192 can be programmed using I/O input and output commands. In order to do this you must have detailed knowledge of the operation of the 8255 PPI's used on the PC-192. This information can be obtained from the following sources :

a) 8555A/8255A-5 Programmable Peripheral Interface, Intel Corporation data sheet.

b) 8255A Programmable Peripheral Interface Applications, Intel Corporation application note.

As an alternative to low level programming, driver software is supplied with the PC-192. This is described in the next section.

4.2. Driver software

The driver software consists of a set of real-time functions for use with the PC-192 board. These device drivers are supplied in C and BASIC, and are callable from most compiled languages, including the following:

- Microsoft C version 5.1
- Turbo C version 1.5
- Microsoft QuickBasic version 4.5

The driver software allows programmers to control the PC-192 via high level function calls, so allowing the user to write custom software without understanding the low level operation of the PC-192. Also included with the driver package is complete source code, in C and BASIC, for the entire driver package. This allows advanced users to modify existing code, rather than having to start writing low level code from scratch.

4.2.1. Driver format

The driver package comes in form of a single module. This module consists of a single program file, and a single include file. Identical versions are supplied in C and in BASIC. Microsoft C, Turbo C and Microsoft QuickBasic compilers are specifically supported, but most other compilers should also be able to compile these modules.

Both the C and BASIC modules are also fully compatible with the LabWindows integrated environment.

All driver functions are performed by calling functions, which return a status code. Possible status return codes are described in the next section.

The driver system is designed to support multiple boards.

4.2.2. Differences between C and QuickBasic versions

The driver is supplied with source code for both C and QuickBasic. These versions are identical, with the exception of the naming of procedures and constants. Where C names use an underscore ("_"), QuickBasic names make use of a period (".").

For example, the initialization function is called Init_brd in C, and Init.brd in QuickBasic.

4.2.3. Ports

PC-192 ports are numbered from 0 to 23. Port 0 is port A, J1; port 1 is port B, J1 etc.

4.2.4. Return codes

Two possible return codes are supported:

ERR_NOT_AVAIL (-1)

The ERR_NOT_AVAIL return code (ERR.NOT.AVAIL in Basic) indicates that the board was not available, or the requested function is not available for that board.

RETURN_OK (0)

The RETURN_OK return code (RETURN.OK in Basic) indicates that the function was correctly performed.

4.2.5. Microsoft C/QuickC

All supplied modules are directly compatible with Microsoft C version 5.1, as well as Quick C version 2.0. You can compile the DACQPC.C module using any supported memory model. For example, using the large memory model:

```
cl /AL /c dacqpc.c
```

4.2.5.1. Required files

DACQPC.C, DACQPC.H

4.2.5.2. Examples

DEMO1.C

4.2.6. Turbo C

All supplied modules are directly compatible with Turbo C version 1.5. You can compile the DACQPC.C module using any supported memory model.

4.2.6.1. Required files

DACQPC.C, DACQPC.H

4.2.6.2. Examples

DEMO1.C

4.2.7. Microsoft QuickBasic

Microsoft QuickBasic version 4.5 is fully supported for both the stand-alone and integrated environment operation. As in the case of the other languages, the driver is supplied in source code form, and is intended for use as a separately compiled module.

4.2.7.1. Required files

DACQPC.BAS, DACQPC.INC

4.2.7.2. Examples

DEMO1.BAS

4.2.8. LabWindows support

Included with the supplied driver is full support for LabWindows V1.2. This support is in the form of an instrument module, and its associated information.

The LabWindows code and functions are identical to the standard libraries described above, but are supplied in the form of three files:

DACQPC.LBW, DACQPC.LWI, DACQPC.FP

These three files contain, in addition to the program code, instrument front panels and help information for use in the LabWindows integrated environment.

In order to install the driver, all that is necessary is to copy the three files to the LabWindows instrument directory. You can then use the instrument module as you would any other.

4.3. Function reference

The DACQPC driver as supplied with the PC-192 requires only seven function calls. These are the following:

Init_brd	Initializes one of up to eight boards.
DIO_prt_cfg	Configures the operating mode of a port.
DIO_in_port	Returns digital data from the specified port.
DIO_out_port	Writes digital data to the specified port.
DIO_in_line	Return the state of a particular line of the specified port.
DIO_out_line	Sets a particular line of the specified port.
DIO_prt_status	Returns a status word indicating the status of a particular I/O port.

Each function will now be described in detail.

4.3.1. Init_brd

Name	Init_brd - Initializes a PC-192
Boards Supported	PC-192
C Usage	#include <DACQPC.H> int Init_brd(int iBrd_num, int iBrd_type, int iBase_addr);
QuickBasic Usage	REM $INCLUDE: 'DACQPC.INC'

FUNCTION Init.brd%(BYVAL iBrd.num%, BYVAL iBrd.type%, BYVAL iBase.addr%)

Description

This function initializes the board at the address iBase_addr. In subsequent call to the driver, the board will be identified by the board number.

The iBrd_type parameter must be set to 13 for PC-192 boards.

The Init_brd function Initializes all ports as inputs.

Return Value

RETURN_OK - Board initialized

Example

DEMO1.C, DEMO1.BAS

4.3.2. DIO_prt_cfg

Name

DIO_prt_cfg - Configure a PC-192 port

Boards Supported

PC-192

C Usage

#include <DACQPC.H>

int DIO_prt_cfg(int iBrd_num, int iPort_num, int iLatch, int iDir);

QuickBasic Usage

REM $INCLUDE: 'DACQPC.INC'

FUNCTION DIO.prt.cfg%(BYVAL iBrd.num%, BYVAL iPrt_num%, BYVAL iLatch%, BYVAL iDir%)

Description

This function configures the specified port for direction and handshaking mode. The iBrd_num parameter is the number with which the board was initialized (via Init_brd).

iPort_num is the number of the port to be configured. This ranges from 0 to 23.

iLatch indicates the handshake mode. If this is 0, then the port is in normal (non-latched) mode. If this is 1, then latched mode operation is selected. Latched mode operation configures the port into operating mode 1. Note that latched mode operation is valid only for the A and B ports of 8255 devices. These are ports 0 and 1, 3 and 4, 6 and 7 etc. For information on latched mode operation, consult the 8255 data sheet.

iDir indicates the direction (input or output) that the port is configured for. 0 indicates input, and 1 output.

Return Value

RETURN_OK - Port initialized.

ERR_NOT_AVAIL - Port does not exist, or cannot be set to requested mode.

Example

DEMO1.C, DEMO1.BAS

4.3.3. DIO_in_port

Name

DIO_in_port - Reads a PC-192 port

Boards Supported

PC-192

C Usage	#include <DACQPC.H> int DIO_in_port(int iBrd_num, int iPort_num, int *iVal);
QuickBasic Usage	REM $INCLUDE: 'DACQPC.INC' FUNCTION DIO.in.port%(BYVAL iBrd.num%, BYVAL iPrt_num%, SEG iVal%)
Description	This function reads digital input data from the specified port. The iBrd_num parameter is the number with which the board was initialized (via Init_brd). iPort_num is the number of the port to be configured. This ranges from 0 to 23. iVal is the result of the read operation. Note that iVal is passed by reference.
Return Value	RETURN_OK - Port read. ERR_NOT_AVAIL - Port does not exist.
Example	DEMO1.C, DEMO1.BAS

4.3.4. DIO_out_port

Name	DIO_out_port - Writes a PC-192 port
Boards Supported	PC-192
C Usage	#include <DACQPC.H> int DIO_out_port(int iBrd_num, int iPort_num, int iVal);
QuickBasic Usage	REM $INCLUDE: 'DACQPC.INC' FUNCTION DIO.out.port%(BYVAL iBrd.num%, BYVAL iPrt_num%, BYVAL iVal%)
Description	This function writes digital output data to the specified port. The iBrd_num parameter is the number with which the board was initialized (via Init_brd). iPort_num is the number of the port to be configured. This ranges from 0 to 23. iVal is the digital pattern to be written.
Return Value	RETURN_OK - Port written. ERR_NOT_AVAIL - Port does not exist.
Example	DEMO1.C, DEMO1.BAS

4.3.5. DIO_in_line

Name	DIO_in_line - Reads a digital input line
Boards Supported	PC-192
C Usage	#include <DACQPC.H>

	int DIO_in_line(int iBrd_num, int iPort_num, int iLine, int *iVal);
QuickBasic Usage	REM $INCLUDE: 'DACQPC.INC'
	FUNCTION DIO.in.line%(BYVAL iBrd.num%, BYVAL iPrt_num%, BYVAL iLine%, SEG iVal%)
Description	This function reads a single digital input line from the specified port. The iBrd_num parameter is the number with which the board was initialized (via Init_brd).
	iPort_num is the number of the port to be configured. This ranges from 0 to 23.
	iLine specifies the line to be read. 0 is the LSB, and 7 the MSB.
	iVal is the result of the read operation. Note that iVal is passed by reference. The result is either 0 or 1.
Return Value	RETURN_OK - Line read.
	ERR_NOT_AVAIL - Port does not exist.

4.3.6. DIO_out_line

Name	DIO_out_line - Writes a digital output line
Boards Supported	PC-192
C Usage	#include <DACQPC.H>
	int DIO_out_line(int iBrd_num, int iPort_num, int iLine, int iVal);
QuickBasic Usage	REM $INCLUDE: 'DACQPC.INC'
	FUNCTION DIO.out.line%(BYVAL iBrd.num%, BYVAL iPrt_num%, BYVAL iLine%, BYVAL iVal%)
Description	This function writes a single digital output line in the specified port. Other lines are not changed. The iBrd_num parameter is the number with which the board was initialized (via Init_brd).
	iPort_num is the number of the port to be configured. This ranges from 0 to 23.
	iLine specifies the line to be read. 0 is the LSB, and 7 the MSB.
	iVal is the value to be written. It may be either 0 or 1.
Return Value	RETURN_OK - Line read.
	ERR_NOT_AVAIL - Port does not exist.
Example	DEMO1.C, DEMO1.BAS

4.3.7. DIO_prt_status

Name	DIO_prt_status - returns the status of the selected port.
Boards Supported	PC-192

C Usage	#include <DACQPC.H> int DIO_prt_status(int iBrd_num, int iPort_num, int *iStatus);
QuickBasic Usage	REM $INCLUDE: 'DACQPC.INC' FUNCTION DIO.prt.status%(BYVAL iBrd.num%, BYVAL iPrt_num%, SEG iStatus%)
Description	This function return the handshake status of a port. The port must have been configured as a latched port. The iBrd_num parameter is the number with which the board was initialized (via Init_brd). iPort_num is the number of the port to be configured. This ranges from 0 to 23. iStatus is the port status. If the port is an output port, then if iStatus is 1, an external device has accepted the previous data, and the next value can be written. If iStatus is 0, then the external device has not yet accepted the current data. The status return corresponds to the 8255 OBF bit in this mode. If the port is an input port, then if iStatus is 1, an external device has latched data into the port, which is available to be read. If iStatus is 0, then the external device has not yet written any data into the port. The status return corresponds to the 8255 IBF bit in this mode.
Return Value	RETURN_OK - Status returned. ERR_NOT_AVAIL - Port does not exist.

Chapter 5

Specifications

5.1. Digital I/O

Number of lines	192
Compatibility	TTL
Input logic low	0.8 V max.
Input logic high	2.0 V min
Input load current	+- 10 uA max
Output low voltage	0.45 V (I_{sink} = 1.7 mA)
Output high voltage	2.40 V (I_{source} = 200 uA)
Output source current	4 mA max, (V_{out} = 1.5 V, 8 outputs max.)
I/O connector	50 pin header, user supplied mating connector 3M 3425-6050

5.2. PC Interface

Base address	DIP switch selectable, 000H to 7E0H
Power requirements	+5 V @ 1.6 A max.
Operating temperature	0°C to 55°C

Relative humidity	0 to 90% noncondensing