

Technical Manual

JRC1

(JUMPtec[®] Remote Control)

1. TABLE OF CONTENTS

1. TABLE OF CONTENTS	2
2. USER INFORMATION	3
2.1. Trademarks	3
2.2. General.....	3
2.3. Warranty.....	3
3. INTRODUCTION	4
4. REQUIREMENTS	5
5. OPERATING INSTRUCTIONS	6
6. AVAILABLE COMMANDS	7
6.1. Overview.....	7
6.2. connect[<port>[<maxbaud>]].....	7
6.3. resume	7
6.4. reboot	8
6.5. memread <mem#> <file>[/b].....	8
6.6. memwrite <mem#> <file>[/b]	8
6.7. diskread <drive> <file>[<sectorcount>]	9
6.8. diskwrite <drive> <file>[<sectorcount>][/vv]	9
6.9. flashread <file>[<blockcount>[<blockstart>]]	10
6.10. flashwrite <file>[<blockcount>[<blockstart>]][/v] [/vv]	10
6.11. flasherase[<blockcount>[<blockstart>]]	10
6.12. ipwrite <ipaddr> <ipmask>[<iprouter>][/r]	11
6.13. boardinfo[/f]	11
6.14. server <file> . [<port>[<maxbaud>]][/u]	12
6.15. imgcreate <drive> <file>	12
7. HINTS.....	13
8. DOCUMENT REVISION HISTORY.....	14

2. USER INFORMATION

Copyright 2000 **JUMPtec**® Industrielle Computertechnik AG.

In this document **JUMPtec**® Industrielle Computertechnik AG will also be referred to by the short form "**JUMPtec**®".

The information in this document has been carefully checked and is believed to be accurate and reliable. However, no responsibility is assumed for inaccuracies. Furthermore, **JUMPtec**® reserves the right to make changes to any portion of this manual to improve reliability, function or design. **JUMPtec**® does not assume any liability for any product or circuit described herein.

2.1. Trademarks

AT and IBM are trademarks of International Business Machines

XT, AT, PS/2 and Personal System/2 are trademarks of International Business Machines Corporation.

Microsoft is a registered trademark of Microsoft Corporation.

Intel is a registered trademark of Intel Corporation.

All other products and trademarks mentioned in this manual are trademarks of their respective owners.

2.2. General

For the circuits, descriptions and tables indicated no responsibility is assumed as far as patents or other rights of third parties are concerned.

The information in the chapter Technical Descriptions describes the type of the boards and shall not be considered as assured characteristics.

The reproduction, transmission or use of this document or its contents is not permitted without expressed written authority.

Offenders will be liable for damages. All rights created by patent grant or registration of a utility model or design, are reserved.

(C) **JUMPtec**® AG 2000

2.3. Warranty

Each board is tested carefully and thoroughly before being shipped. If, however, problems should occur during operation, please check your user specific settings of all boards included in your system. This is often the source of problems due to resource conflicts. If a board is defective, it can be sent to your supplier for repair. Please take care of the following steps:

1. The board returned should have the factory default settings since a test is only possible with these settings.
2. In order to repair your board as fast as possible we require some additional information from you (detailed description of how and when did the problem occur, what is exactly the problem and so on).
3. Upon receipt of returned boards please be aware that your user specific settings could have been changed during repair.

Within the warranty period the repair is free of charge as long as the warranty conditions are observed. Because of the high test expenditure you will be charged with the test cost if no fault is found. Repair after the warranty period will be charged.

This **JUMPtec**® product is warranted against defects in material and workmanship for the warranty period from the date of shipment. During the warranty period **JUMPtec**® will at its option either repair or replace defective products.

For warranty service or repair the product must be returned to a service facility designated by **JUMPtec**®.

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance or handling by buyer, unauthorized modification or misuse, operation outside of the product's environmental specifications or improper installation or maintenance.

JUMPtec® will not be responsible for any defects or damages to other products not supplied by **JUMPtec**® that are caused by a faulty **JUMPtec**® product.

3. INTRODUCTION

The **JUMPtec**® Remote Control (JRC) is an extension of the PC BIOS that provides a way to intercept and reroute certain BIOS functionality over a serial port at an early stage during the system's boot process.

There are two distinct software components involved:

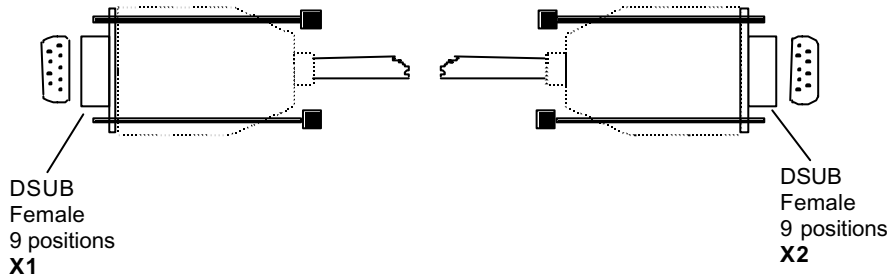
- The BIOS extension. The PC that contains this component is referred to as **client**. This component is part of the **JUMPtec**® Extension BIOS.
- The second component runs on a different machine referred to as **host** that is connected with a serial cable to the client. This second component is an application that can be run at the Command Prompt or from batch files. This component is available as a 32 bit Windows console for Windows 9x or NT and as a MS-DOS application.

There are two different aspects how this interception is utilized:

- The **Server Mode** provides a way of intercepting screen output from and keyboard input to the client on the host machine. In Server Mode it is possible to map the host's floppy drive to the client's drive specifier A:. The host can also emulate a single disk drive as a file on the host machine's hard disk. This file can be created with the command *IMGCREATE* and is accessible from the client machine as drive A:.
- The second aspect is **Image Mode** which provides a way of halting the boot process of the client. This allows the host machine to read or write the contents of the CMOS RAM, EEPROM CMOS mirror, NVRAMs of other **JUMPtec**® boards as well as the contents of any flash, hard or floppy disks of the client.

4. REQUIREMENTS

- JUMPtec® PC with BIOS that supports JRC (Client PC)
- Regular PC (Host PC) running either
 - Windows 9x or Windows NT with JRC.EXE installed, or
 - MS-DOS with JRCD.EXE installed.
- Serial Cable that connects any of the two PCs COM ports



X1		X2	
Signal	Pin	Pin	Signal
Receive Data	2	3	Transmit Data
Transmit Data	3	2	Receive Data
GND	5	5	GND

5. OPERATING INSTRUCTIONS

- Open a Command Prompt window on the Host PC when running Windows95/NT.
- Issue a connect command with the appropriate COM port and the desired baud rate:

```
jrc connect COM2 38400          for Windows95/NT systems
or
jrzd connect COM2 38400        for DOS systems
```

NOTE! The DOS version of the host program JRCD.EXE does not run under Windows95/NT and may halt the command prompt window. Use the Windows95/NT version JRC.EXE instead.

- Reboot the client.
- Both PCs should now be connected
- Issue any other commands

Please note that you can write batch files that can issue any number of commands in succession. You can use `if errorlevel` to determine if a command was successful.

Please note that the `server` command to enter the server mode contains an implicit connect and by default uses the baud rate and port of the last successful manual connect.

The disk image for the server mode can be created with the `imgcreate` command.

To create an image file from a floppy, put the floppy into the host's floppy drive and use

```
jrc imgcreate a: disk.img      for Windows95/NT systems
or
jrzd imgcreate a: disk.img     for DOS systems
```

After this a file `disk.img` is available on the host's hard disk, that can be accessed from the client as drive A: in server mode.

For further informations and examples please see the application note 27 (JAP0027.zip) on the **JUMPtec**® homepage: www.jumptec.de.

6. AVAILABLE COMMANDS

6.1. Overview

Image Mode Commands:

```

connect[ <port>[ <maxbaud>]]
resume
reboot
memread <mem#> <file>[ /b]
memwrite <mem#> <file>[ /b]
diskread <drive> <file>[ <sectorcount>]
diskwrite <drive> <file>[ <sectorcount>][ /vv]
flashread <file>[ <blockcount>[ <blockstart>]]
flashwrite <file>[ <blockcount>[ <blockstart>]][ /v] [ /vv]
flasherase <blockcount>[ <blockstart>]]
ipwrite <ipaddr> <ipmask>[ <iprouter>][ /r]
boardinfo [ /f]

```

Server Mode Commands:

```

server <file>|. [ <port>[ <maxbaud>]][ /u]
imgcreate <drive> <file>

```

Each command can be abbreviated by usually two letters. These are underlined.

6.2. connect[<port>[<maxbaud>]]

Parameters:

<port>
Serial port. The default is COM2.

<maxbaud>
The default is 9600 baud. The actual baud will be negotiated.

Description:

Waits for a client on the given port. The command tries to reboot the client first if an connections already exists, otherwise the client must be reset manually. See Operating Instructions for details.

6.3. resume

Parameters:

none

Description:

The client will continue with the current boot process. After this command there is no connection between host and client anymore.

6.4. reboot

Parameters:

none

Description:

The client will be rebooted. After this command there is no connection between host and client anymore.

6.5. memread <mem#> <file>[/b]

Parameters:

<mem#> (for further description see below)

0: CMOS

1: EEPROM CMOS mirror

2: NVRAM of JIDA board #1

n: NVRAM of JIDA board #n-1

<file>

filename under which the data will be stored on the host

/b

If specified the file will be written binary. By default a text file ASCII hex will be written.

Description:

Writes the contents of the specified client memory into a file on the host.

6.6. memwrite <mem#> <file>[/b]

Parameters:

<mem#> (for further description see below)

0: CMOS

1: EEPROM CMOS mirror

2: NVRAM of JIDA board #1

n: NVRAM of JIDA board #n-1

<file>

filename under which the data have been stored on the host

/b

If specified the file will be written binary. By default a text file ASCII hex will be parsed.

Description:

Reads a file and writes the contents into the specified client memory.

Description of the parameter <mem#>:

The parameter 0 determines the reading/writing of the current data from/to the **CMOS**. The CMOS is a RAM memory where the CMOS Setup configuration is stored. In all **JUMPtec**® units supporting JRC, the CMOS settings are also stored in an EEPROM to avoid the need of a battery that supplies the CMOS RAM when the system power is turned off. The writing of a file content to the CMOS will not affect the contents of the EEPROM. After a reboot the BIOS settings are restored from the contents of the EEPROM and changes made to the CMOS are overwritten.

Using the parameter 1 (**EEPROM CMOS mirror**) the CMOS Setup configuration stored in the EEPROM can be read/written. During each systems boot up, the BIOS settings are read out of the EEPROM and written to the CMOS RAM.

The **NVRAM** contains besides user informations (alphanumeric-LCD settings, darkboot settings, ...) **JUMPtec®** specific data like the module's serial number, fabrication date, last repair date,

The command memread/memwrite 2 includes the data read/written using the parameter 1 (CMOS Setup changes stored in the EEPROM). Besides this the user informations can be accessed too.

Explanation of the parameter **n**: **NVRAM of JIDA board #n-1**.

To get information about the JIDA (**JUMPtec® Intelligent Device Architecture**) board number under which the CPU unit can be accessed, the following procedure is recommended:

Call "Get Device ID" with CL=1. The name of the first device installed will be returned. If result was "Board exists" (CL=0), increment CL and call "Get Device ID" again. Repeat until result is "Board not present" (CL#0). You now know the names of all boards within your system that follow the JIDA standard. More information about a specific board may then be obtained by calling the appropriate inquiry function with the board's number in CL.

WARNING: Association between board and board number may change due to configuration changes. Do not rely on any association between board and board number. Instead, always use the procedure described in the preceding paragraph first, to determine the association between board and board number.

6.7. diskread <drive> <file>[<sectorcount>]

Parameters:

<drive>

- 0: Floppy A:
- 1: Floppy B:
- 128: Hard Drive (Usually C:)
- 129: Hard Drive (Probably D:)

<file>

File name under which the disk image will be stored on the host.

<sectorcount>

Amount of sectors to be read. If this parameter is omitted then all sectors will be read.

Description:

Writes the contents of the specified client disk into a file on the host.

6.8. diskwrite <drive> <file>[<sectorcount>][/v]

Parameters:

<drive>

- <file>0: Floppy A:
- 1: Floppy B:
- 128: Hard Drive (Usually C:)
- 129: Hard Drive (Probably D:)

<file>

File name under which the disk image has been stored on the host.

<sectorcount>

Amount of sectors to be written. If this parameter is omitted then all sectors will be written.

/v

Verify the contents of the disk after the all sectors have been written

Description:

Reads a file and writes the contents into the specified client disk.

NOTE! The file containing the disk image must have been created with the diskread command. Don't use images created with other commands.

6.9. flashread <file>[<blockcount>[<blockstart>]]

Parameters:

<file>

File name under which the flash contents will be stored on the host.

<blockcount>

Amount of 64k flash blocks to be read. If this parameter is omitted then all flash blocks will be read.

<blockstart>

Number of first block (0-based). The default value is 0.

Description:

Writes the contents of the specified flash blocks into a file on the host.

NOTE! On some systems there may be flash blocks reserved for the system BIOS. These blocks cannot be accessed.

6.10. flashwrite <file>[<blockcount>[<blockstart>]][/v] [/vv]

Parameters:

<file>

File name under which the flash contents has been stored on the host.

<blockcount>

Amount of 64k blocks to be written. If this parameter is omitted then all flash blocks will be written.

<blockstart>

Number of first block (0-based). The default value is 0.

/v

Verify the contents of the flash after each block has been written.

/vv

Verify the contents of the flash after the all blocks have been written.

Description:

Reads a file and writes the contents into the specified flash block.

NOTE! On some systems there may be flash blocks reserved for the system BIOS. These blocks cannot be accessed.

NOTE! The file containing the flash data must have been created with the flashread command. Don't use files created with other commands.

6.11. flasherase[<blockcount>[<blockstart>]]

Parameters:

<blockcount>

Amount of 64k blocks to be erased. If this parameter is omitted then all flash blocks will be erased.

<blockstart>

Number of first block (0-based). The default value is 0.

Description:

Erases the contents of the specified flash blocks.

NOTE! On some systems there may be flash blocks reserved for the system BIOS. These blocks cannot be accessed.

6.12. ipwrite <ipaddr> <ipmask>[<iprouter>][/r]**Parameters:**

<ipaddr>
Client's IP address.
<ipmask>
Client's IP mask.
<iprouter>
Client's router's IP address.

Description:

This sets the specified IP addresses and stores them in the client's EEPROM. This is used to setup WEBtoNET boxes. Refer to the WEBtoNET documentation for details.

6.13. boardinfo[/f]**Parameters:**

/f
Write a empty flag file (*.FLG) with the client's board name as core file name.

Description:

This command displays information about the client's CPU board in that order:

Flags
Primary Class Name
Board Name
Board Sub Name
Manufacturer Name
Manufacturing Date
Last Repair Date
Serial Number
Hardware Revision
Firmware Revision
Jida Revision
Feature Number

The /f switch is very useful for detecting the board in a batch file:

```
del LEU1.FLG
del P488.FLG
jrc bi /f
if exist LEU1.FLG echo littleMonster
if exist P488.FLG echo MOPSlcd4
```

6.14. server <file>|. [<port>[<maxbaud>]] [/u]

Parameters:

<file>

There are two ways of using this parameter:

1. File name of a disk image on the host that will be used in the emulation of drive A: on the client. A period specifies that no remote disk should be emulated. Use the imgcreate command to create a disk image file. See introduction for details.
You can also specify a file name that contains wild cards (? or *) then up to 16 disk image files will be read and sorted alphabetically. You can then use the Scroll Lock key to cycle thru those files. Note that one key stroke unmounts the current disk image and a second stroke mounts the next disk image in the list. When the end of the list is reached the first image will be mounted again.
The wildcard feature is not supported by the DOS version JRCD.EXE.
2. You can also specify a physical disk drive i.e. A: to map the host drive directly without using a disk image. All accesses to client drive A: are then routed to host drive A:.

<port>

Serial port. The default is COM2

<maxbaud>

The default is 9600 baud. The actual baud rate will be negotiated.

/u

If this switch is present then the disk image will not be mounted, i.e. there is no disk in the emulated drive until the image is mounted manually via the Scroll Lock key. This switch works with disk images only, i.e. it has no effect if a physical disk drive was specified.

Description:

Waits for a client on the given port and enters server mode. The command tries to reboot the client first if an connections already exists. See Operating Instructions for details.

You can exit the server mode by pressing both control keys on the keyboard simultaneously. To reboot the client press the left shift and control key and then the right control key. To mount or unmount a disk image press the Scroll Lock key.

NOTE! On JRCD.EXE version 1.7 and below the meaning of the shift keys and the control keys is exchanged, i.e. you must use both shift keys to exit, and both shift and any control key to reboot. Version 1.8 and higher uses the same key combinations as JRC.EXE

NOTE! The disk image file must have been created with the imgcreate command. Don't use images created with other commands.

6.15. imgcreate <drive> <file>

Parameters:

<drive>

Drive from which the disk image should be created, i.e. a:

<file>

File name of the disk image that can be used in the emulation of a drive on the client. See introduction for details.

Description:

Creates a disk image from a floppy disk and writes it on the host's hard disk. You can use this to speed up the disk emulation in server mode.

7. HINTS

- Switching between server mode and image mode

The commands 'connect' and 'server' both check for existing connections and operate accordingly. If a connection already exists, then an implicit 'reboot' will be issued first. This makes it easy to switch between server mode and image mode.

If you want to switch from image mode to server mode simply execute the 'server' command. If you want to switch from server mode to image mode then terminate the server mode without rebooting the client first and then issue a new 'connect' command.

- Purpose of the /u switch in server mode

This is useful in case you want to emulate a disk drive in server mode with a disk image file. If this image is not bootable and the boot sequence on the client is set up to boot from drive A:, then there is no way to boot the client. In this case simply add the /u switch when executing the 'server' command. This will emulate an empty drive, thus causing the client to boot from drive C:. Later on you can 'insert' the image into the emulated drive, i.e. mount and access the image by pressing the Scroll Lock key.

- Usage of the Scroll Lock key for installation programs

The mount/unmount feature in server mode via the Scroll Lock key is useful when running a installation program that requires several disk images. For example when you have a program that requires eight diskettes and you want to install it on the client. In this case using the floppy drive of the host system would take an extremely long amount of time for installation.

To speed up the installation process, create images of the installation diskettes first and name them disk1.img to disk8.img. Then execute the 'server' command and specify 'disk*.img' as filename for the disk image. Now you can use the Scroll Lock key to 'exchange' the disk images in the emulated drive.

8. DOCUMENT REVISION HISTORY

Filename	Date	Edited by	Alteration to previous document revision
JRC1M110.DOC	20.02.1998	H. Eß	Initial version created based on JRC.DOC version 1.6 / 1998.02.19 by DP
JRC1M111.DOC	11.03.1998	H. Eß	Minor changes in section Introduction / warning added for not to run JRCD.EXE under Windows95/NT / figure and diagram for serial I/O cable added / info about server exit key codes corrected / term ‚host mode‘ replaced with ‚server mode‘
JRC1M112.DOC	30.07.1998	DP	Verify command options
JRC1M113.DOC	14.10.1998	H. Bruhn	Added some more explanations for easier customer understanding
JRC1M114.DOC	25.02.1999	CH	Layout of the manual revised and missing information added.
JRC1M115.DOC	04.02.2000	DP	Added ipwrite and boardinfo commands