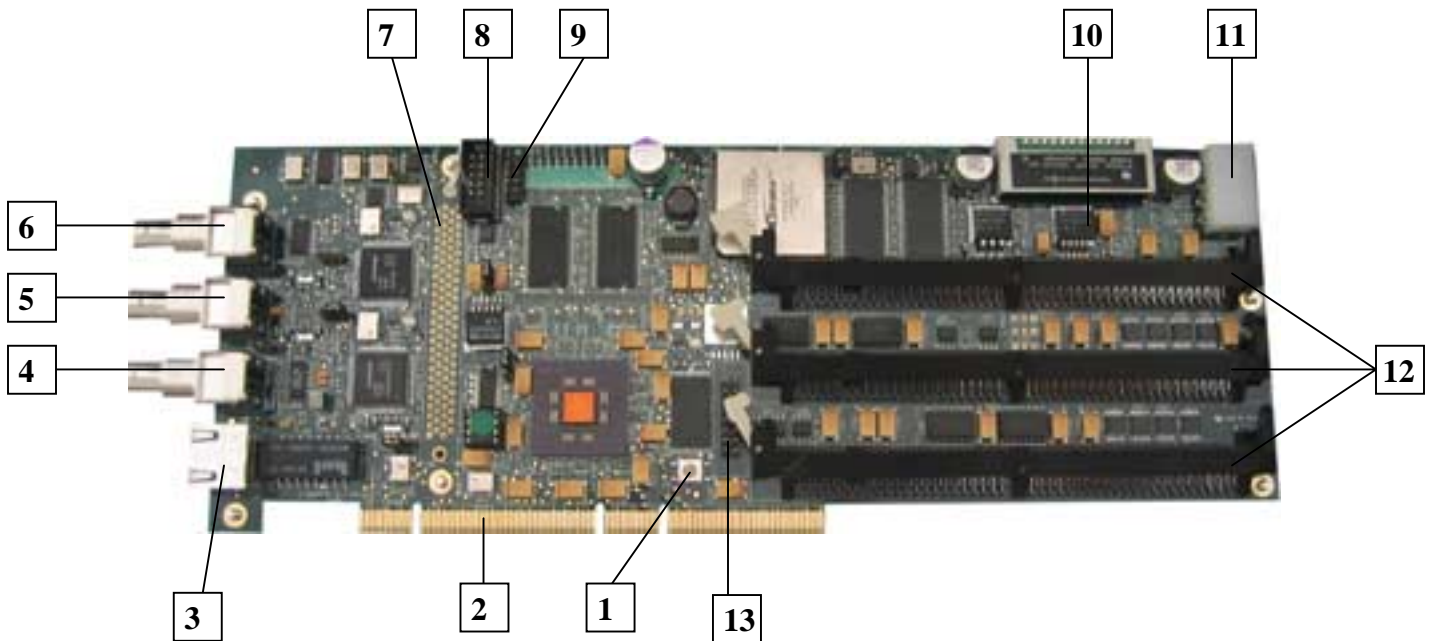


ORCA Embedded Linux Operating manual version 006

1.Introduction

The ORCA encodes a standard definition or high definition video stream using MPEG-2. It also accepts uncompressed audio or it can compress a single stereo pair using MPEG1 Layer 2 or AC-3. The ORCA consists of a PCI mother-card that can run by itself (standalone mode) using an external power supply. The ORCA may also be placed in a computer PCI slot. The board has 3 DIMM slots, each of which can receive an MPEG-2 daughter board. The board is shown below with the major components highlighted:

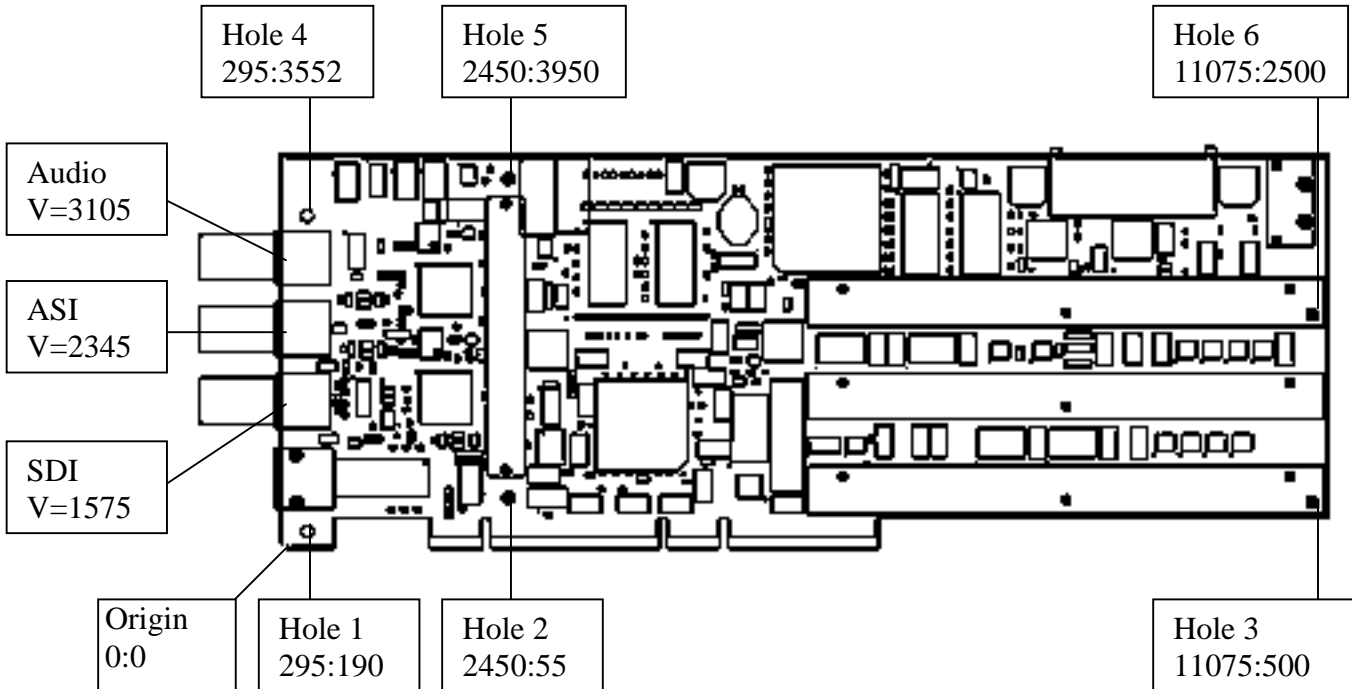


- 1) Reset button,
- 2) PCI connector.
- 3) On-board ethernet connector.
- 4) Multi-format HD/SD SDI input (270Mbits, 1.485Gbits or 1.485/1.001 GBits).
- 5) Multi-format HD/SD/DVB-ASI output.
- 6) AES-EBU stereo audio input.
- 7) Pre-processing expansion connector, currently not in use.
- 8) Primary serial port.
- 9) Secondary serial port and/or GPIO port.
- 10) 3.3V Power regulator, used only in standalone mode. When the ORCA is used as a PCI expansion card, this regulator should be disabled or not present.
- 11) Power connector for standalone mode.
- 12) Encoder board slots.
- 13) Test port, reserved for factory use only.

The board is shipped with a serial port cable and a serial port IDC-10 to DB-9 adapter. This cable combination allows the user to connect the ORCA to a PC terminal using any COM port.

2. Hardware setup

A mechanical drawing of the ORCA is shown below:



The mounting holes coordinates are shown in units of 1/1000 of an inch. The format is X:Y. The BNC connectors vertical position is also indicated as V=xxxx.

2.1. Standalone mode

The ORCA is shipped in a 1U rack mount unit.

2.2 PCI mode

In PCI mode, the ORCA gets its 5V power and 3.3V power through the PCI connector. The internal 3.3V power regulator should be disabled or absent. If not, DO NOT PLUG THE BOARD INTO A PC. Contact the factory for support.

Find 2 empty PCI slots in your computer and connect the ORCA to the leftmost slot (the back of the PC should be facing you). Connect the serial port cable to the ORCA on one side, to a COM port on the other.

3. Console setting

READ THIS SECTION IF YOU PLAN TO USE THE ORCA WITHOUT GUI.

On a PC running Windows, start a Hyperterminal session and set it for baud rate = 115200, 8 bits, no parity, 1 stop, hardware flow control off, software flow control off.

Also set “ECHO ON” so that the commands you type are visible in the console. Set line wrap on.

On a PC running another OS, the setting should match the parameters above.

4. Operation

READ THIS SECTION IF YOU PLAN TO USE THE ORCA WITHOUT GUI.

Power-up the ORCA. You should first see some boot code information, then the BIOS is going to wait for 5 seconds before initiating automatic boot. This delay is necessary to allow user intervention, necessary for firmware upgrades. After the boot delay, the ORCA firmware will start, wait until the command prompt shows up:

~#

at the prompt, type the following:

```
cd orca <ENTER>
```

This command changes the current directory to the directory “orca”.

The ORCA defaults to encoding a long gop size of 15, and a bit rate of 9Mbps. You can now use the serial console to modify the parameters and operate the unit. The ORCA will automatically detect the input format and configure itself accordingly. Before encoding, ensure that the video format detected is correct. Contact Doremi if you notice any discrepancies.

5. Command description

READ THIS SECTION IF YOU PLAN TO USE THE ORCA WITHOUT GUI.

The commands are divided into status and control commands. Status commands start with a “g” characters, as in “Get”. The control commands start with an “s”, as in “Set”. Each command consist of 2 characters. If a command requires a parameter, the parameter immediately follows the command. The parameter consist of 3 numerical characters. For example to set the bit rate to 95Mbits per seconds, you need to type “sb 095”. The command is executed the moment you hit <enter>.

5.1 Control commands

Control commands allow you to set the encoding parameters. The command always starts with “./host_com ...”. The list of acceptable commands and their description are shown below:

5.1.1 Set bit rate command

./host_com sb nnn

This command sets the bit rate to the desired value. The range is (0 – 100). The limitation is in software, contact Doremi if you need to go beyond this bit rate.

5.1.2 Set GOP size

./host_com ss nnn

This command allows you to set the GOP size to a value between (1-15). The limitation is in software, contact Doremi if you need to go beyond 15.

5.1.3 Set number of B frames

./host_com sm nnn

This command sets the number of B frames between 2 reference frames. Reference frames are I frames or P frames. The number of B frames value is between 0-2.

5.1.4 Set chroma format

./host_com sc 00n

This command sets the chroma format. If n=1, 422 is used, if n=0, 420 is used. This command only applies for cards that are delivered for 422 encoding.

5.1.5 Set vertical search range

./host_com sr nnn

This command sets the vertical search range for motion estimation. Parameter range is (0-128). Default is 128.

5.1.6 Set horizontal search range

./host_com sR nnn

This command sets the horizontal search range for motion estimation. Parameter range is (0-200). Default value is 200.

5.1.7 Set Constant bit rate mode

./host_com sv

This command sets the encoding mode to constant bit rate.

5.1.8 Set fixed quality mode

./host_com sf nnn

This command sets the encoding mode to fixed quality. The bit rate is variable. The parameter range is (3-31) where 3 is the highest bit rate (highest quality), 31 being the lowest bit-rate (lowest quality). When using this mode, the user should ensure that the bit rate is set to a level that is high enough. Otherwise, the DVB-ASI will not work properly.

5.1.9 Set Custom CBR mode

./host_com sd

This command sets the encoding mode to a Doremi Custom CBR mode. This mode is particularly useful at low bit rates where seam artifacts will show up if regular CBR is used.

5.1.10 Constrain low bit rate

./host_com sQ nnn

This command is used in CBR mode and in custom CBR mode to reduce the influence of frames that have limited content, which cause the bit rate to drop below the CBR average rate. Parameter range is (3-31). Setting this parameter is content dependant. To disable this feature, set value to 31.

5.1.11 Constrain high bit rate

./host_com sq nnn

This command is used in CBR mode and in custom CBR mode to limit the bit rate for frames that are very rich in content, which cause the bit rate to exceed the CBR average rate. Parameter range is (1-29). Setting this parameter is content dependant. To disable this feature, set value to 1. This value should always be below the Constrain low bit rate value by at least an offset value of 2.

5.1.12 Start encoding

./host_com st nnn

This command starts encoding after the number of frames specified in the parameter.
Parameter range is 0-999.

5.1.13 Stop encoding

./host_com se nnn

This command ends encoding after the number of frames specified in the parameter.
Parameter range is 0-999.

5.2 Status Commands

Status commands allow the user to verify setting before encoding starts. The list of acceptable commands is shown below.

5.2.1 Get bit rate

./host_com gb

This command returns the bit rate currently being used.

5.2.2 Get GOP size

./host_com gs

This command returns the GOP size currently being used.

5.2.3 Get number of B frames

./host_com gm

This command returns the number of B frames between reference frames being used.

5.2.4 Get low bit rate constraint

./host_com gq

This command returns the value set by the command “sq”.

5.2.5 Get high bit rate constraint

./host_com gQ

This command returns the value set by the command “sQ”.

5.2.6 Get bit rate mode

./host_com gT

This command returns the mode currently programmed (CBR, Custom CBR, Fixed...)

5.2.7 Get vertical search range

```
./host_com gr
```

This command returns the vertical motion estimation search range.

5.2.8 Get horizontal search range

```
./host_com gR
```

This command returns the horizontal motion estimation search range.

5.2.9 Get chroma mode

```
./host_com gc
```

This command returns the chroma mode being used (422 or 420). This command is only valid for boards with that capability.

6. Firmware update

To update the ORCA firmware, it is necessary to have the serial connection in place and operational on a console (hyperterminal on Windows, kermit on Linux). Reboot the board by pressing the reset button. After reset, you have 5 seconds to hit any key on the keyboard to stop the boot process and enter into BIOS update mode. In the section below, it is assumed that the customer has received a new firmware from Doremi that consist of 2 files, “new_image.bin”, “new_rd.bin”.

6.1 firmware update using Linux on serial console only

- a) type “loadb 1000000 <enter>” at the command prompt, this command will setup the ORCA to receive a file through the serial port.
- b) Hold the control button down and type “\”, release the control button, type the character “c”. This operation will switch between the console and the kermit command line.
- c) Type “send new_image.bin”. this will start the send process on the host PC and the receive process on the ORCA. You should see on the console a numerical progress report that is continuously updated until the end of the transfer.
- d) Once the transfer is done, type “c <enter>” to switch back to the ORCA console mode.
- e) To verify type “imi 1000000 <enter>” this will print the header of the firmware along with the firmware version. If the firmware version is ok, it means the download has been successful.
- f) Type “protect off ffc10000 fff70000” to unprotect the flash BIOS,
- g) Type “erase ffc10000 fff70000 <enter>” to erase the area in flash reserved for the ORCA firmware, you should normally see a numerical progress report which will end with the message, “erased 55 sectors”.
- h) Type “cp.b 1000000 ffc10000 \$(filesize) <enter>”. This command will copy the new firmware to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- i) type “loadb 1000000 <enter>” at the command prompt, this command will setup the ORCA to receive a file through the serial port.
- j) Hold the control button down and type “\”, release the control button, type the character “c”. This operation will switch between the console and the kermit command line.
- k) Type “send new_rd.bin”. this will start the send process on the host PC and the receive process on the ORCA. You should see on the console a numerical progress report that is continuously updated until the end of the transfer.
- l) Once the transfer is done, type “c <enter>” to switch back to the ORCA console mode.
- m) Type “cp.b 1000000 ffd60000 \$(filesize) <enter>”. This command will copy the new ram disk to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- n) Open the file orca_bargs_xxx.txt in a text editor. Every line in the file represent a boot variable and what its value should be. Type “printenv <enter>”. This command will print on the screen all the current variables. If a variable value is different when compared to the text file, you need to modify it. For example, if the text file says “bootcmd=bootm ffc10000 ffd60000”, and the “printenv” result on the screen for that variable is “bootcmd=bootm ffc50000”, then you need to execute the following commands:
 - a. “setenv bootcmd bootm ffc10000 ffd60000 <enter>” notice that when you type the command, you should not type the sign “=”.
 - b. “savenv <enter>” to save the changes,
 - c. “printenv <enter>” to verify that the change has been done.
 - d. You need to execute this sequence for every variable that needs update.

6.2 firmware update using Linux on serial console and ethernet

This mode allows faster transfer of the binary files to the ORCA card. You should have a tftp server running under Linux. Copy The files “new_image.bin”, and “new_rd.bin” to the tftp server root directory.

- a) type “tftpboot 1000000 new_image.bin <enter>” at the command prompt, this command will transfer the file to the ORCA via the TFTP server.
- b) To verify successful transfer, type “imi 1000000 <enter>” this will print the header of the firmware along with the firmware version. If the firmware version is ok, it means the download has been successful.
- c) Type “protect off ffc10000 fff70000” to unprotect the flash BIOS,
- d) Type “erase ffc10000 fff70000 <enter> to erase the area in flash reserved for the ORCA firmware, you should normally see a numerical progress report which will end with the message, “erased 55 sectors”.
- e) Type “cp.b 1000000 ffc10000 \$(filesize) <enter>”. This command will copy the new firmware to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- f) type “tftpboot 1000000 new_rd.bin<enter>” at the command prompt, to transfer the second file to the ORCA.
- g) Type “cp.b 1000000 ffd60000 \$(filesize) <enter>”. This command will copy the new ram disk to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- h) Open the file orca_bargs_XXX.txt in a text editor. Every line in the file represent a boot variable and what its value should be. Type “printenv <enter>”. This command will print on the screen all the current variables. If a variable value is different when compared to the text file, you need to modify it. For example, if the text file says “bootcmd=bootm ffc10000 ffd60000”, and the “printenv” result on the screen for that variable is “bootcmd=bootm ffc50000”, then you need to execute the following commands:
 - a. “setenv bootcmd bootm ffc10000 ffd60000 <enter>” notice that when you type the command, you should not type the sign “=”.
 - b. “savenv <enter>” to save the changes,
 - c. “printenv <enter> to verify that the change has been done.
 - d. You need to execute this sequence for every variable that needs update.

6.3 firmware update using Windows on serial console only

- a) Type “loadb 1000000 <enter>” at the command prompt, this command will setup the ORCA to receive a file through the serial port.
- b) Go to the Transfer menu of the Hyperterminal and select “Send File...”. A window will pop-up, browse and select “new_image.bin”, then choose “kermit protocol”, then hit Ok or Send.
- c) Wait until the end of the transfer.
- d) To verify that you successfully transferred the file, type “imi 1000000 <enter>” this will print the header of the firmware along with the firmware version. If the firmware version is ok, it means the download has been successful.
- e) Type “protect off ffc10000 fff70000” to unprotect the flash BIOS,
- f) Type “erase ffc10000 fff70000 <enter>” to erase the area in flash reserved for the ORCA firmware, you should normally see a numerical progress report which will end with the message, “erased 55 sectors”.
- g) Type “cp.b 1000000 ffc10000 \$(filesize) <enter>”. This command will copy the new firmware to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- h) type “loadb 1000000 <enter>” at the command prompt, this command will setup the ORCA to receive a file through the serial port.
- i) Go to the Transfer menu of the Hyperterminal and select “Send File...”. A window will pop-up, browse and select “new_rd.bin”, then choose “kermit protocol”, then hit Ok or Send.
- j) Wait until the end of the transfer.
- k) Type “cp.b 1000000 ffd60000 \$(filesize) <enter>”. This command will copy the new ram disk to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- l) Open the file orca_bargs_xxx.txt in a text editor. Every line in the file represent a boot variable and what its value should be. Type “printenv <enter>”. This command will print on the screen all the current variables. If a variable value is different when compared to the text file, you need to modify it. For example, if the text file says “bootcmd=bootm ffc10000 ffd60000”, and the “printenv” result on the screen for that variable is “bootcmd=bootm ffc50000”, then you need to execute the following commands:
 - e. “setenv bootcmd bootm ffc10000 ffd60000 <enter>” notice that when you type the command, you should not type the sign “=”.
 - f. “saveenv <enter>” to save the changes,
 - g. “printenv <enter>” to verify that the change has been done.
 - h. You need to execute this sequence for every variable that needs update.

6.4 firmware update using Windows on serial console and ethernet

This mode allows faster transfer of the binary files to the ORCA card. You should have a tftp server running under Windows. Copy The files “new_image.bin”, and “new_rd.bin” to the tftp server root directory.

- a) type “tftpboot 1000000 new_image.bin <enter>” at the command prompt, this command will transfer the file to the ORCA via the TFTP server.
- b) To verify successful transfer, type “imi 1000000 <enter>” this will print the header of the firmware along with the firmware version. If the firmware version is ok, it means the download has been successful.
- c) Type “protect off ffc10000 fff70000” to unprotect the flash BIOS,
- d) Type “erase ffc10000 fff70000 <enter>” to erase the area in flash reserved for the ORCA firmware, you should normally see a numerical progress report which will end with the message, “erased 55 sectors”.
- e) Type “cp.b 1000000 ffc10000 \$(filesize) <enter>”. This command will copy the new firmware to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- f) type “tftpboot 1000000 new_rd.bin<enter>” at the command prompt, to transfer the second file to the ORCA.
- g) Type “cp.b 1000000 ffd60000 \$(filesize) <enter>”. This command will copy the new ram disk to flash. You should type the command as is above, do not replace the sentence “filesize” with the numerical file size.
- h) Open the file orca_bargs_xxx.txt in a text editor. Every line in the file represent a boot variable and what its value should be. Type “printenv <enter>”. This command will print on the screen all the current variables. If a variable value is different when compared to the text file, you need to modify it. For example, if the text file says “bootcmd=bootm ffc10000 ffd60000”, and the “printenv” result on the screen for that variable is “bootcmd=bootm ffc50000”, then you need to execute the following commands:
 - e. “setenv bootcmd bootm ffc10000 ffd60000 <enter>” notice that when you type the command, you should not type the sign “=”.
 - f. “savenv <enter>” to save the changes,
 - g. “printenv <enter>” to verify that the change has been done.
 - h. You need to execute this sequence for every variable that needs update.