

# Orix/Twilighte Manual

v2022.4.1

rev 09/11/2022



## EXTRA DOCUMENTATION

---

### Rom loader and firmware twilighte board menu

<https://github.com/orix-software/systemd/blob/master/docs/pdf/systemd.pdf>

## INTRODUCTION

### What is new (v2022.4) ?

[https://orix-software.github.io/update/2022\\_4/](https://orix-software.github.io/update/2022_4/)

### General informations

This documentation must be use when you have installed orix version **2022.3** (see at the top of the banner on the oric at boot).

On <http://orix.oric.org>, you will have some youtube videos links showing how to use some functionnality.

The board has a firmware version. This firmware can be upgraded see « Hardware and firmware upgrade » section.

The board can be upgraded too but you have to send it to upgrade the board see « Hardware and firmware upgrade » section » too.

The card has a 512KB of eeprom, and 512KB of RAM. This RAM is saved with a battery. For instance, only bank 4, 3, 2 and 1 can be switched to see others sets. It's a software limit. In the future, you will be able to displays all bank and starts any binary from theses banks. If you wants to change the set, you can use twil command. This command can switch to eeprom bank or ram bank and can switch to any set.

Some extra devices (TOM2, logitech joypad) are explained a bit in this manual, but it's not advert, we don't ear anything:) It explains some ways to use joystick, others hardware exists in the same way)

### Features

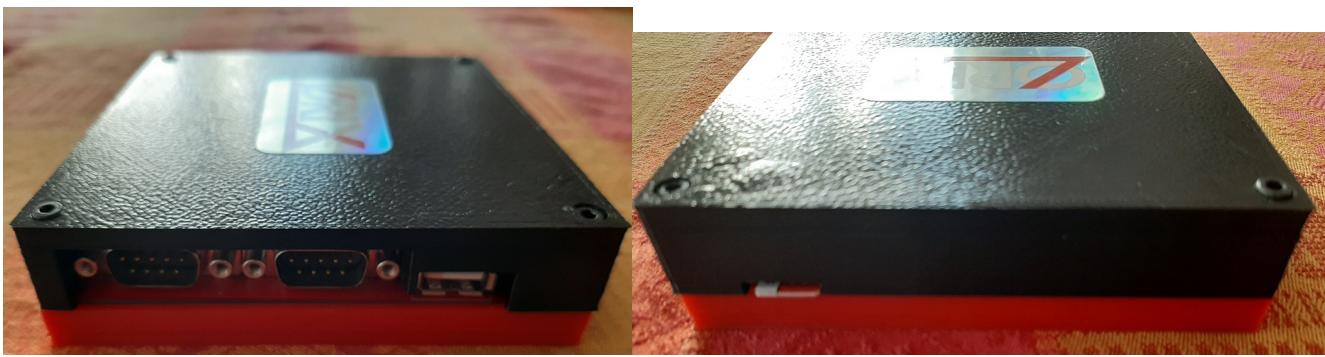
- .tap file fast loading (with multitap files)
- Joysticks support for a lot of games on atmos mode
- the hobbit, defence-force (and others games) works without any patch for loading
- in system : kernel update, roms and ram update (with orixcfg binary)
- 2 DB9 Joysticks (atari)
- 512KB of EEPROM (banking mode)
- 512KB of RAM (banking mode)
- read/write from sdcard (MAX 64GB) or usb drive (mass storage)
- drag and drop from the PC to the oric : It will be available on the oric (with WIFI connexion) : It requires extra hardware with a raspberry pi zero
- fast loading : 46KB per second. A game require less than one second to load and start
- cumulus compatible with the help of an other amplibus board (not provided)

# GETTING STARTED

## Content



## Physical ports



## Hardware limits

The usb controller manage FAT32 only. Sdcard and usb key must be formatted with FAT32 filesystem. If you want to use pi zero gadget trick, you need to do a mkfs to FAT32 file system.

All tests had been done with samsung evo sdcard and sandisk usb key. A lot of sdcard works, and we did not see incompatibility with sdcard.

Sdcard controller and usb key controller can work with 32GB storage Max. But it can handle 64 GB sdcard (tested). It can handle larger sdcard/usb key reader, but only 32 and 64 GB devices was used.

## Software limits

The sdcard/usb controller can handle long filename, but Orix handles 8+3 filename only.

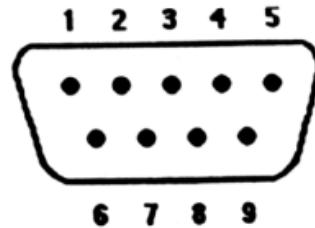
## Information about joysticks part

The left port has only 1 button. The right port has 3 buttons. The joystick pinout is atari pinout. You can use standard DB9 joystick. You can also plug « TOM2 » hardware (not provided), it can connect a usb mouse or usb joypad (wireless) to theses ports. For example, logitech joypad F710 (wireless) works with TOM2.

Please note that TOM2 can only handle 2 buttons. It means that the third button can't work with TOM2 connected.

# Prise Joystick

- |                       |                       |
|-----------------------|-----------------------|
| 1. ↑                  | 6 . Souris bouton n°1 |
| 2. ↓                  | 7 . +5V.              |
| 3. ←                  | 8 . Masse             |
| 4. →                  | 9 . Souris bouton n°2 |
| 5 . Souris bouton n°3 |                       |



## {#anchor-12}First boot : Initialize the storage

When the card is sent, kernel is built with a default storage. In order to know which device is the default one, you can type « mount ». You can only use one device at the same time, but you can swap easily theses devices from command line.

If you see « sdcard », then sdcard will be read by default. You can change it, with a command : « twil -u », it will switch to usbdrive. If you want to have usb drive by default, you can program kernel with the tool « orixcfg ». See Orixcfg section.

Now, if you know which device you will use by default, you can install all software on it.

Plug the device on your PC (sdcard or usb key). If you have a pi zero w, you can do this with drag and drop solution from the PC.

Download sdcard.tgz from this : <http://repo.orix.oric.org/dists/official/tgz/6502/>

It contains all software for orix there is others which are not available in this archive.

Now, use 7zip on your PC (or tar/gzip under linux), and unzip all files from this sdcard.tgz. Put all theses new files in your device root folder.

Now, you can insert the device (sdcard or usbkey – or pi zero) in the twilighte board and play.

```
ORIX v2021.1 CPU:6502
560 KB RAM/512 KB ROM - 2020-12-03 23:37
/#mount
rootfs on / type FAT32 /dev/sda1 <sdcard>
/#

```

## Upgrade from v2022.4 to v2022.4.1

If your orix version is below v2022.3 version, please go to annexes part at the end of this document, before you try to upgrade to v2022.4.1

- Download <http://repo.orix.oric.org/dists/official/tgz/6502/sdcard.tgz>
- untar/gunzip sdcard.tgz (use 7zip under windows) on your device usb or sdcard : It could require some time to copy because there is a lot of small files (tap, hlp etc)
- you can start orix on real machine, and type :

```
/#cd usr
/usr/#cd share
```

```
/usr/share#cd carts  
/usr/share/carts#cd 2022.4
```

If you want to use usb drive for default device :

```
/usr/share/carts/2022.4#orixcfg -r -s 4 kernelus.r64
```

If you want to use sdcard for default device :

```
/usr/share/carts/2022.4#orixcfg -r -s 4 kernelsd.r64
```

- press ‘y’, and **wait until Orix reboots**

(Don’t switch off the Oric at this step)

## Optionnal step for upgrade

Now bank displays all banks from 1 to 64. It means that you should have some strange bank signature for eeprom. Now an empty set is provided in */usr/share/carts/2021.4* folder. With Orixcfg you can initialize your set with this cart. Don’t use « -s 4 » flag for orixcfg when you want to load emptyset.

## First step : type a command

You can access to available command from many ways :

- From /bin folders, there is binary available on current device, ‘ls’ will show you available commands
- From banks : type « help -b5 » you will see available commands

# Basic10 & Basic11

## Launch

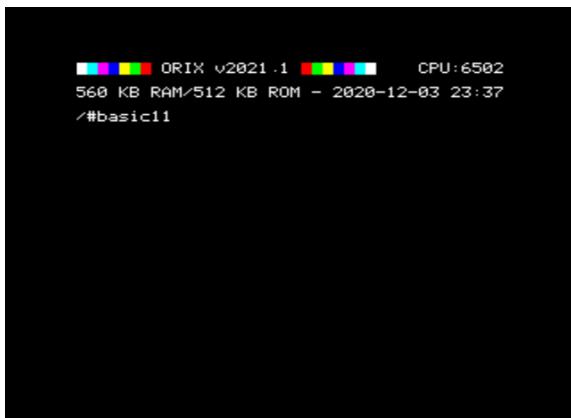
Basic10 starts Oric-1 rom with sdcard/usb key support

Basic11 starts Atmos rom with sdcard/usb key support

You can type basic11 or press FUNCT+B to start

## Load a personal .tap file

When you starts basic11 commands, the default path is « /home/basic11/ ». Each action on the basic11 mode will be done in this folder (cload/csave). If you cload a tape file, it must be in « /home/basic11 » folder.



You have downloaded a .tap file, and want to use it. Then, you can create a folder /home/basic11/

Under Orix

```
/#mkdir home  
/#cd home  
/home#mkdir basic11  
/home#cd basic11
```

Put you file in this folder from your PC, and start basic11 (you don't need to be in the « /home/basic11 » folder to start basic11 with no parameter. By default, basic11 starts in « /home/basic11/ »)

## Oric.org tape file

When you downloaded sdcard.tgz and unzip it into sdcard or usbkey device, there is many tape file included in this archive. You don't need to move these type file, if you know the key, you can starts it from commands line. In this case, it will load the correct basic1.1 rom to start the tape file (see below), and the correct joystick configuration if it's correct.

## Oric.org tape file update

Each week a new software.tgz is generated. You can download it from « repo » and unzip it on the device. It will generate last tape file and last joysticks configuration.

## Search a tape file from command line

#basic11 -l	key	NAME
1815HACK	1815	
J2048	2048	
256CIRCL	256	BYTES ZOOMING CIRCLE DEM
3D	3D	
3DFONCUS	3D	FONCUS
3DLUFAR	3D	LUFAR-SCHACK
3DMAZE	3D	MAZE/ BREAKOUT
3DWALK	3D	MUNCH
FREECELL	3D	FREECELL
4GAMES	4	GAMES FOR CHILDREN
4KLONG	4K	LONG
4KBOX	4K	BOX
4KFIRE	4K	FIRE
4KQIX	4K	QIX
6502ORIC	6502	MAN CMPI DEMO
ATM	A.T.M	(AIR TERRE MER)
ACCOUNT	A.COUNT	BOOK
ACHERON	ACHERON	RAGE / LA RAGE D'AC
ACROBATE	ACROBATES	
ADDBOOK	ADDRESS BOOK	
ADDSOUS	ADDSOUS	

Basic11 has also many.tap files inserted in sdcard.tgz

Try to find the software with option -l

```
/# basic11 -l
```

If you find your software, you can do perform **ctrl+c**.

You can type space to do a pause.

On that case, you can launch the tape file like :

```
/# basic11 «KEYDISPLAYED
```

When KEYDISPLAYED is the key displayed in key column. Please note that the key must be in **UPPERCASE**

## Load a tap file from command line

Note that MYFILE must be in **UPPERCASE**

```
/# basic11 «MYFILE
```

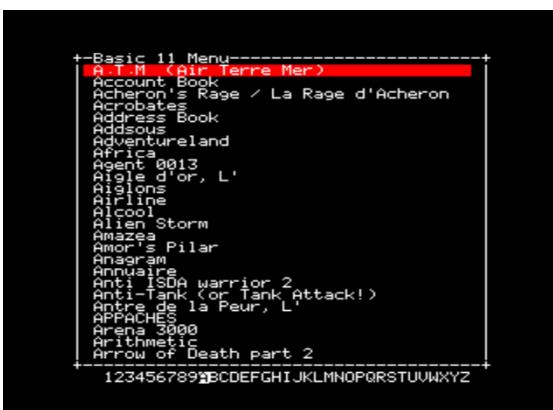
If MYFILE is in the oric.org database, it will launch the software with the filename MYFILE.

If basic11 command does not find MYFILE in the oric.org database, it will try to load it from /home/basic11/ folder.

## Save your program

If you start « basic11 » with no options, basic rom will starts and each csave (or cload) actions will store files in « */home/basic11* » folder

## Start basic11 menu



If you type « basic11 -g » on command line or FUNCT+G, you will have a menu with all software which have a download link on oric.org (only atmos version and when a tape file is available).

```
/#basic11 -g
```

You can use left and right letters to change to a new letter. If the letter is empty, it means that there is no available tap file for this letter.

You can use up and down link to navigate into software. If you press enter, the software will starts.

Note that not all games are working yet. Some times, chars are corrupted. If the joysticks does not works, there is two case :

- the game does not call rom routine to manage keyboard
- keyboard mapping is not done yet

You can use arrows to navigate into the menu :

- up and down to select the software
- right and left to switch to the menu letters

Some letters are empty. It means that there is no software with tape file available on oric.org for this letter

## Quit basic11

If you want to quit basic11 from interpreter command line, you can type « QUIT ». This will force to reboot to Orix (you can also use reset button)

## How the .tap file starts

If you only type « basic11 », this will start bank 6 (normal basic rom). The default folder in that case is « /home/basic11 »

If you type « basic11 » with a tape file as an argument, there is 2 cases

1. The tape file (key) is already known in oric.org website, then basic11 try to find it in its databank file (/var/cache/basic11/ folder). If the key is found, it will start the tape file located in « /usr/share/basic11/... »
2. If the key is unknown, it will try to find it in « /home/basic11 »

If the tap file is in the oric.org db file, basic11 will load the software configuration from the db software file (as joystick configuration, and the id of the rom). Basic11 load the right rom into ram bank, override the default basic11 path to the tape file folder (« *usr/share/basic11/[firstletter software]* »).

It means that if you load this kind of software and you can quit the software, each file action in basic11 rom, will be performed in « *usr/share/basic11/[firstletter software]* ». »

## Not working tapes (for instance)

- All Oric-1 games can be started with FUNCT+L in ROM menu : start oric-1 (depending of your device), and put .tap files in /home/basic10
- Software which does not work (25), but the number can be reduced in future release.

cobra	Cobra pinball	Damsel in distress
Rush hour 4K		
Le diamant de l'ile maudite	Durendal	HU*BERT
Hunchback	Schtroumpfs	Stanley (ROM 0,1 tested)
Them	Titan	Visif
Xenon III	Dig Dog	Elektro Storm
Kilburn Encounter	Le tresor du pirate	L'aigle d'or (ROM 0,1 tested)
Compatible (micropuce)	Volcanic demo	Clavidact
DAO Cobra Soft	CW-Morse	The Hellion
MARC	Caspak	Kryllis : when we lost one life, the game does not restart

## Tape with altered charset

Fire flash	Scuba Dive	3D fongus (i,f letters)
------------	------------	-------------------------

## **Joysticks issues**

We did keyboard/joystick mapping for a lot of games, but we did not set the keyboard mapping for all software. If you want to help us, contact us.

Some game does not work because they handle their own keyboard routine. It could be handle with hardware tricks but, it's not done.

Some others games uses special keys (SHIFT, CTRL) for direction or the first button. Theses cases are not handle yet : but it could in the future.

404: Not Found

## **asm2k2**

launch asm2k2 demo

/#asm2k2

# **bank**

## **Introduction**

Bank command is command line tool to see which bank are loaded into EEPROM bank and RAM bank. Each bank has a “signature”. Bank allows to see theses banks. Bank can also starts a ROM with his id. In that case, you don’t need to have a rom “orix friendly” and you can start it from command line. In the current bank version, there is restriction to launch a command.

## **SYNOPSYS**

**List all bank (when ROM signature is valid)**

```
/#bank Bank 1 to 32 is eeprom bank and bank 33 to 64 are ram bank
```

**Displays all signature even when ROM is not valid**

```
/#bank
```

**List all commands from a bank**

```
/#help -b5
```

**Start a specific bank**

```
/#bank 1
```

If you need to load a rom into a bank, you need to have a look to orixcfg binar

## **DESCRIPTION**

This command displays bank when the command is called without parameter. WIth a parameter, you can switch to a the id of the bank passed to the argument :

bank : displays all the bank (if a signature is found) bank 4 : switch to bank 4 bank -a : displaues all bank (empty bank too)

## **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/bank.asm>

## **Barboric/Barbitoric demo**

Launch barbitoric demo from orix (sdcard/usbdrive)

/# barboric

## **Blakes7**

Start Blakes7

/#blakes7

If you want to remove your saved game :

/#cd /usr/share/blakes7

/usr/share/blakes7#/ rm 136.o

## **Command: bootfd**

### **Starts microdisc bootsector**

bootfd is a tool to boot the boot sector when a drive is connected. Insert a disk and type :

```
/#bootfd
```

The binary is located to bin folder. It will load microdisc rom and start it. If the binary displays that it does not found microdis.rom, then place microdis.rom file in the right folder.

If you have firmware 1, you will be able to load « blake's 7 ». If you have firmware 2, all sedoric .dsk should start.

For instance, only Space99 does not work, it stops after intro.

### **SYNOPSYS**

- bootfd

### **EXAMPLES**

- bootfd

Boots only microdisc boot sector with the help of microdisc.rom. In this version, we don't verify if a dsk is inserted or not. If there is no disk, the oric will crash

### **SOURCE**

Src code : <https://github.com/orix-software/bootfd.git> (Assembly and C)

## Born1983

Launch born 1983 demo

/#born1983

## **cat**

### **Introduction**

Display a file to the screen

### **SYNOPSIS**

- cat FILENAME

### **NOTES**

Displays content of the FILENAME

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/cat.asm>

# **cd**

## **Introduction**

Change directory

## **SYNOPSYS**

- cd DIRECTORY

## **EXAMPLES**

- cd /usr/bin
- cd ..
- cd /

## **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/cd.asm>

## Command: cksum

checksum utility

### SYNOPSIS

- cksum file [...]
- cksum @batchfile
- cksum -h

### EXAMPLES

- cksum /bin/cksum

### DESCRIPTION

checksum and count the bytes in a file

### OPTIONS

- -h show this help message and exit

### SOURCE

<https://github.com/orix-software/cksum>

```
/usr/share/basic11#ls
basicsd.rom  basicsd0.rom  basicsd1.rom
basicsd2.rom  basicsus.rom  basicsus0.rom
basicus1.rom  basicsus2.rom  basicsus1.rom
basicus2.rom

/usr/share/basic11#cksum basicsus2.rom
2153917165 16384 basicsus2.rom
/usr/share/basic11#
```

## **clear**

### **Introduction**

Clear the screen

### **SYNOPSYS**

- clear

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/clear.asm>

# **cp**

## **Introduction**

Copy a file to another file

## **SYNOPSYS**

- #cp from to

## **DESCRIPTION**

Copy a file to another. Only 40KB can be copied. It's a software limit. cp tries to allocate 40KB, if there is not enough memory it displays Out of memory.

## **EXAMPLES**

- cp from.txt to.txt

## **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/cp.asm>

df

```
[  ] ORIX v2021.2 [  ] CPU:6502
[  ] 560 KB RAM/512 KB ROM - 2021-04-25 00:38
/#[df -h
512-blocks Used Avail . Use% Mounted on
168430090 293831626 xxx /dev/sda1
/#[
```

Displays storage device informations

## **Command: dsk-util**

Disk image utility

### **SYNOPSIS**

- dsk-util -f|-s file.dsk
- dsk-util -h

### **EXAMPLES**

- dsk-util -f ftodos.dsk
- dsk-util -s sedoric3.dsk

### **DESCRIPTION**

dsk-util display the directory of a disk image file.

### **OPTIONS**

- -h show this help message and exit
- -f FT DOS disk image
- -s Sedoric disk image

### **SOURCE**

<https://github.com/orix-software/dsk-util>

## **echo**

### **Introduction**

Display a message

### **SYNOPSYS**

- #echo hello

### **DESCRIPTION**

Display a message

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/echo.asm>

## **env**

### **Introduction**

Displays env variables

### **SYNOPSIS**

- #env

### **DESCRIPTION**

Displays env variables. It can't be used in command line or shell script for instance

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/env.asm>

## Command: forth

Launch forth for Orix

### SYNOPSIS

- forth

### DESCRIPTION

**Forth** is a language based on teleforth (telestrat). You can type **VLIST** to list all words. If you need to switch to hires, you can type **GRAFX** and **HRIES**. If you need to switch to text: **GRAFX** and **TEXT**

You can use forth language. It works the same ways than Tele forth (it's teleforth but it write file on sdcard/usbdrive).

You can download Teleforth langage in order to see how to program in forth.

When you type « forth » forth starts with the current folder directory.

If you were in /, forth will load and save file in this folder.

In others cases, you can create a forth folder in home and goes into it in forth for example :

- mkdir home
- #cd home
- #mkdir forth
- forth

if you type « cd forth» in forth environnement, all files actions will be perform in « */home/forth* »

### SOURCE

<https://github.com/assinie/Tele-Forth>

# Ftdos

## Introduction

ftdos command starts ftdos .dsk file without jasmin floppy disc controller. It's a beta version.

## Usage

To start a .dsk file

```
/#ftdos mydsk
```

To display the ftdos command version :

```
/#ftdos -v
```

ftdos disks must be present in /home/basic11/dsk/

## Informations

- Write on .dsk file is not supported
- refers to ftdos manual for usages
- you need to switch off/on the oric when you reset into ftdos mode, because there is a bug when we reset during ftdos mode. Ftdos does not load and reboot into Orix

## **Command: grep**

**grep utility**

### **SYNOPSIS**

- grep [-ncisw] string filename
- grep -h

### **DESCRIPTION**

**grep** display all lines of a text file that contain a string.

### **OPTIONS**

- -h^GDisplay command syntax
- -n^GShow line numbers
- -c^GCount only the matching lines
- -i^GIgnore case
- -w^GString can use wildcards \*, ?, ^ and \$
- -s^GSilent mode

### **EXAMPLES**

- grep error menu.sub
- grep -n “level 1” menu.sub
- grep -i ERROR menu.sub
- grep -ni ‘level 2’ menu.sub
- grep -w ’\*lev?? 2\*’menu.sub
- grep -w ‘^if’ menu.sub
- grep -w ‘error\$’ menu.sub
- grep -w ‘if\*level ??’ menu.sub

### **SOURCE**

<https://github.com/orix-software/grep>

## **help**

### **Introduction**

Display commands

### **SYNOPSYS**

- help

### **DESCRIPTION**

- No parameter : Displays all internals shell command
- bX parameter : Displays all internals commands in bank X

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/help.asm>

## Command: hexdump

hexdump utility

### SYNOPSIS

- hexdump file

### EXAMPLES

- hexdump /bin/hexdump

### DESCRIPTION

Display file contents in hexadecimal. You can use [SPACE] to pause the display ou [CTRL]+C to abort.

### SOURCE

<https://github.com/orix-software/hexdump>



The screenshot shows a terminal window with the ORIX v2021.1 logo at the top. Below it, the CPU is listed as 6502 and the memory usage as 560 KB RAM/512 KB ROM - 2020-12-09 13:58. The command entered is '/bin#hexdump file'. The output shows the hex dump of the file, with columns for address, bytes, and ASCII representation. The ASCII output includes recognizable strings like '.ori...', '...j', '...H', '...b...j...', '...l...', '...@.3...', '...w...', '...P...@...', '...P...LB...', and 'P....'. A vertical ellipsis indicates more data.

Address	Bytes	ASCII
0000	01 00 6F 72 69 01 00 00	.ori...
0008	00 00 00 00 00 00 00 00	
0010	09 32 00 00 00 00 0A 22	"...j"
0018	0A 35 22 00 00 00 0E 5A 22	"...H"
0020	15 20 23 15 00 00 03 3D 23	'...b...j...'
0028	A2 20 B0 67 00 00 03 5B 20	'...l...'
0030	15 F0 00 00 00 00 00 00 00	'...@.3...'
0038	A5 15 A8 22 00 00 20 20 20	'...w...'
0040	19 A0 04 20 78 19 60 00 00	'...P...@...'
0048	20 B2 19 20 AC 19 A0 00 02	
0050	20 78 19 4C 42 00 60 00 00	'...P...LB...@...'
0058	0C B2 20 20 AC 19 A0 00 02	
0060	20 70 19 A9 BE A2 10 20	'P....'

## **iports**

### **Introduction**

Display I/O Ports of the board

### **SYNOPSYS**

- #iports

### **DESCRIPTION**

Displays I/O ports.

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/iports.asm>

## **ls**

### **Display catalog**

List all the file in the current folder. Token are supported (\*,?) ex : « ls ?e.tap » will list all files with a ‘e’ in the second letter

### **SYNOPSIS**

list all the files in the current folder

- ls

List all \*.tap files

- ls \*.tap

List size and datetime of the file + ls -l

### **DESCRIPTION**

Directories are in <sup>FBLUE</sup>G color. It manages ‘-l’ and Pattern works in different ways : ls \*.tap

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/ls.asm>

## **lscpu**

### **Introduction**

Displays cpu info. It detects : 6502, 65c02 and 65816

### **SYNOPSIS**

- #lscpu

### **DESCRIPTION**

Displays cpu info

### **EXAMPLES**

- lscpu

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/lscpu.asm>

## **lsmem**

### **Introduction**

Displays malloc table

### **SYNOPSIS**

- #lsmem

### **DESCRIPTION**

Displays malloc table. Free chunks and busy chuncks are displayed with ranges.

### **EXAMPLES**

- lsmem

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/lsmem.asm>

## **man**

### **Introduction**

display manual pages

### **SYNOPSYS**

- man ls

### **DESCRIPTION**

Displays manual. All .hlp files are located in /usr/share/man/. It manages multiples text screen (if .hlp is bigger than a screen when space is pressed, it switch to next page).

### **Keys**

- SPACE to switch to next page
- ESC to exit

### **EXAMPLES**

- man ls

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/man.asm>

# **mkdir**

## **Introduction**

Create a folder

## **SYNOPSIS**

- /#mkdir PATH

## **DESCRIPTION**

Create a folder. -p (recursive mode) option is not available

## **EXAMPLES**

- mkdir /opt

## **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/mkdir.asm>

# mount

## Introduction

Displays mounts

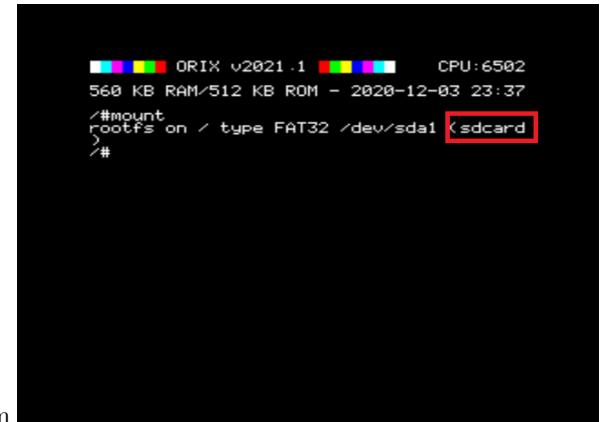
## SYNOPSYS

- #mount

## DESCRIPTION

Displays mount (usb key or sdcard)

## SOURCE



ORIX v2021.1 CPU:6502  
560 KB RAM/512 KB ROM - 2020-12-03 23:37  
/#mount  
rootfs on / type FAT32 /dev/sda1 <sdcard  
/#

<https://github.com/orix-software/shell/blob/master/src/commands/mount.asm>

## **otimer**

### **Introduction**

Display timer since the Oric is power on

### **SYNOPSYS**

- otimer

### **NOTES**

Each time Oric reboots, this timer is reset

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/otimer.asm>

## **pwd**

### **Introduction**

Display the current path

### **SYNOPSIS**

- `pwd`

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/pwd.asm>

## **ps**

### **Introduction**

Displays process list

### **SYNOPSIS**

- #ps

### **DESCRIPTION**

Displays process list

### **EXAMPLES**

- ps

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/ps.asm>

## Quintessential demo

Start quintessential demo

/#quintes

## **reboot**

### **Introduction**

reboot machine

### **SYNOPSYS**

- reboot

### **DESCRIPTION**

Reboot the machine: it call NMI VECTORS and flush page 2 and page 0

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/reboot.asm>

## **rm**

### **Introduction**

remove file

### **SYNOPSIS**

- /# rm /myfile

### **DESCRIPTION**

rm file or directory. For instance, rm can only remove file in root folder, relative arg does not work yet

### **EXAMPLES**

- rm /myffile

### **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/rm.asm>

# sh

## Introduction

When kernel has finished to initialize at boot time, sh command is started in interactive mode

## Interactive mode

*Esc-b* : move cursor at the beginning of the previous word

*Esc-f* : move cursor at the end of the next word

*Esc-l* : switch current word into lowercase, and put cursor at the end of the word

*Esc-u* : switch current word into uppercase, and put cursor at the end of the word

*Ctrl-a* : move cursor at the beginning of the line

*Ctrl-e* : move cursor at the end of the line

*Ctrl-c* : cancel current line

*Ctrl-k* : delete the end of the line

*Ctrl-l* : clear screen, and displays the line, the cursors keeps his position

*Ctrl-u* : clear the line and put cursor at the beginning of the line

*Ctrl-t* : swap char under the cursor with the previous one, and shift the cursor to the right

*Ctrl-o* : Switch into replacement or insertion mode

## Shortcuts for tools

*Funct+T* : Start Twilighte setup

*Funct+L* : Start Loader

*Funct+B* : Start Basic11

*Funct+G* : Start basic11 gui

## Available commands

The command line is limited in characters (37). If you reach this limit, you won't be able to type the complete command line

Known bugs

- « ./ » can not be used to launch a binar

# **twil**

## **Introduction**

Twil command can displays the current firmware of twilighte card, and can swap root folder to usbkey or sdcard.

## **SYNOPSIS**

- `/#twil -f` : displays Twilighte board firmware
- `/#twil -u` : switch default device : usbdrive
- `/#twil -s` : swap default device to : sdcard

## **DESCRIPTION**

## **SOURCE**

<https://github.com/orix-software/shell/blob/master/src/commands/twil.asm>

# **MONITOR**

## **Usage**

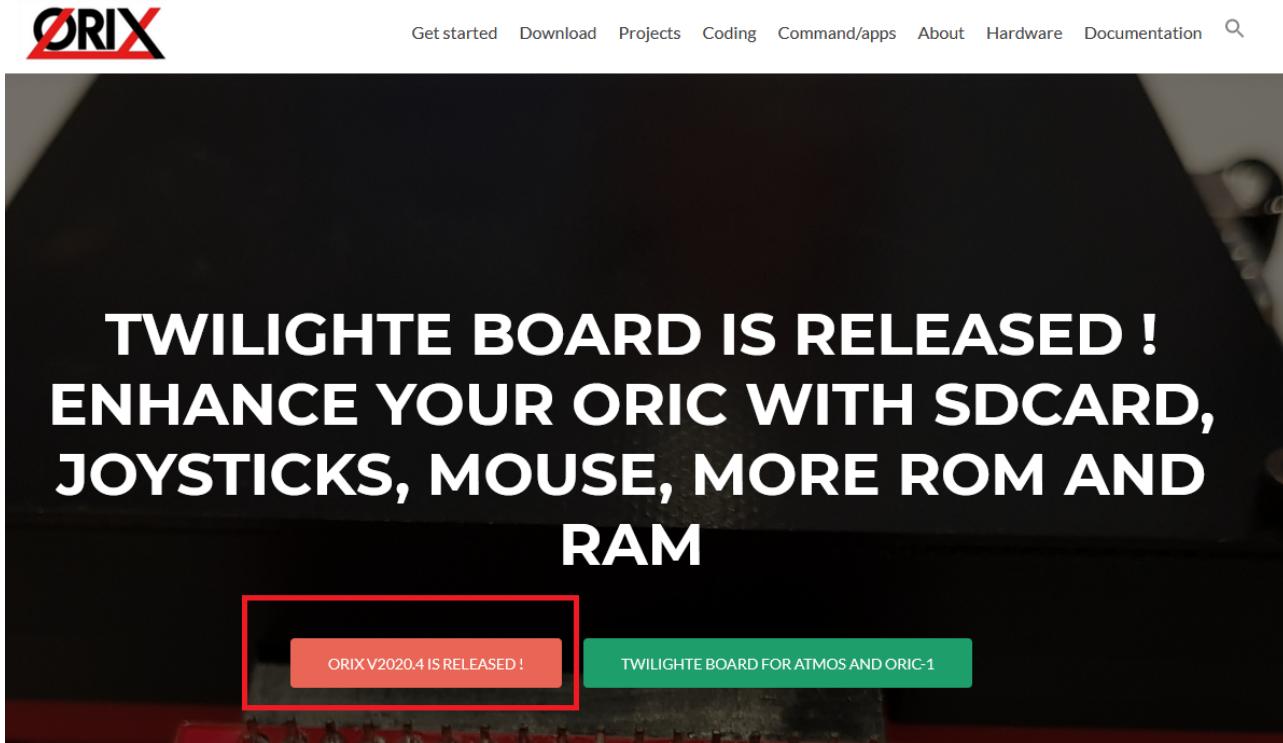
Monitor is a rom which can displays a monitor. It's teleass without assembler part.

# ORIXCFG

## Update kernel, shell : orixcfg

When a new released is done, you can update the eeprom with the new kernel and new roms.

If you want to know if you need to update the kernel, you can compare your current version, and the last release version. You can go to <http://orix.oric.org> You need to have a look to this release below :



If on your Oric screen, it's not the same value, you can update it. The sequence of the Orix release is Year.X. There is 4 releases a year, and each one must be done until you reach the final one, in order to avoid some case. If your version is v2020.3 and the last available version is v2021.4. You need to update to v2020.4, then v2021.1, v2021.2, v2021.3, v2021.4, v2022.1, v2022.4

It's maybe possible to jump to version v2022.3, but it's at your own risk because there is a « chance » that some kernel calls changed, and orixcfg could do wrong step.

### Update kernel, shell

When you need to update kernel, you can update it with orixcfg. You just need to define set 4 on the command line. This step is very **dangerous** if you don't load the right file. There is no verification and any file on the command line will be load in the kernel set. Usually, kernel set file is named kernxxxx.r64.

If you did something wrong on this step, you won't be able to start orix again. It means that you will need to remove eeprom from the card and program it with a eeprom programmer

This next command will load kernel.r64 to kernel set. Please wait until Orix reboots. If you have a kernel 2020.3 and you need to load a kernel 2021.1, you will need to load previous kernel set before the update of 2021.1.

.r64 extension means that it's a 64KB set. It's usually used to define that the file contains 4 roms of 16KB.

Please note that we provide 2 kernels version. One named « kernelsd.r64 » which means that the default device will be sdcards, and the other one « kernelus.r64 » which means that default device will be « usb » (usbkey). If you load the wrong kernel at this step, you can use twil command to switch to the right device, and you can start again kernel update with the right file (kernelsd.r64 or kernelus.r64 depending of your configuration).

```
/#orixcfg -r -s 4 kernelsd.r64
```

### Load a ROM into a ram slot

Space between values and switches are not optionnal, orixcfg needs theses spaces

```
/#orixcfg -b XX -l myrom.rom
```

This command will load myrom.rom (in the current path), in RAM bank XX

Older usage as : orixcfg -r -s X -b Y myrom.rom is no longer included in orixcfg since orixcfg v2021.3

## Load a set of ROM into ROM slot

```
/#orixcfg -r -s 0 myrom.r64
```

This command will load myrom.r64 (in the current path), in set 0. For instance, you can not load one bank, you need to load 64KB set.

## Clear bank ram or initialize it

Ram bank are not initialized when the board is tested. If you have garbage on screen when you uses bank (after you used twil -w). You have to clear all ram bank (ram bank are battery saved).

If you want to clear bank 4 of the set 0, you can do this command. You need to do this command for each bank of each set. For instance, there is no switch to clear all the ram with one command.

```
/#orixcfg -w -s 0 -b 4 -c
```

## Flush all ram bank

```
/#orixcfg -w -f
```

## **OSID MUSIC**

### **How to play osid music ?**

You need to check if you have twilight board firmware 2 :

```
/#twil -f
```

If it returns 2 or greater, you can download some osid files :

[https://www.oric.org/software/osid\\_music-2534.html](https://www.oric.org/software/osid_music-2534.html)

Place all .tap files in /home/basic11

And launch :

```
/#basic11
```

Load patch to avoid to load sedoric routines (in basic command line)

```
CLOAD«OSID
```

And then load the osid file you want :

```
CLOAD«OSNEVER
```

## **PWD**

### **Introduction**

Displays current PWD

## **SHA1**

### **Usage**

Sha1 is a tool to displays a string into sha1 encoding

# **STORMLORD**

## **Introduction**

Stormlord is Stormlord game port to Orix. You can use joysticks to plays to this game.

**Only one joystick port is working on this version**

## SYSTEMD

Systemd is a rom which can load another ROM in ram slot. When you type systemd, it will reads `/etc/systemd/banks` and will load rom declared in this file sequencialy. It means that the first rom will be load in bank id 33, the second one in bank id 34.

This roms can be used in a eeprom bank, you can load it with orixcfg

You can set roms in « `/etc/systemd/banks` » as :

`[MYROMNAME1]`

`path=/usr/share/rom/my.rom`

`[MYROMNAME2]`

`path=/usr/share/rom/my2.rom`

## **Command: submit**

**submit utility**

### **SYNOPSIS**

- submit file [arg...]

### **EXAMPLES**

- submit help.sub

### **DESCRIPTION**

**submit** is a command language interpreter that executes commands read from a file.

### **OPTIONS**

- no options

### **SOURCE**

<https://github.com/orix-software/submit>

## **Command: untar**

**untar utility**

### **SYNOPSIS**

- `untar -t[v]f tarfile`
- `untar -x[v]f tarfile`

### **EXAMPLES**

- `untar -tf /home/test.tar`

### **DESCRIPTION**

**untar** list and extract files in a TAR archive

### **OPTIONS**

- `-h ^B`show this help message and exit
- `-t ^B`list files
- `-x ^B`extract files
- `-v ^B`verbose mode

### **SOURCE**

<https://github.com/orix-software/untar>

# CUMULUS COMPATIBILITY

## How to connect a cumulus

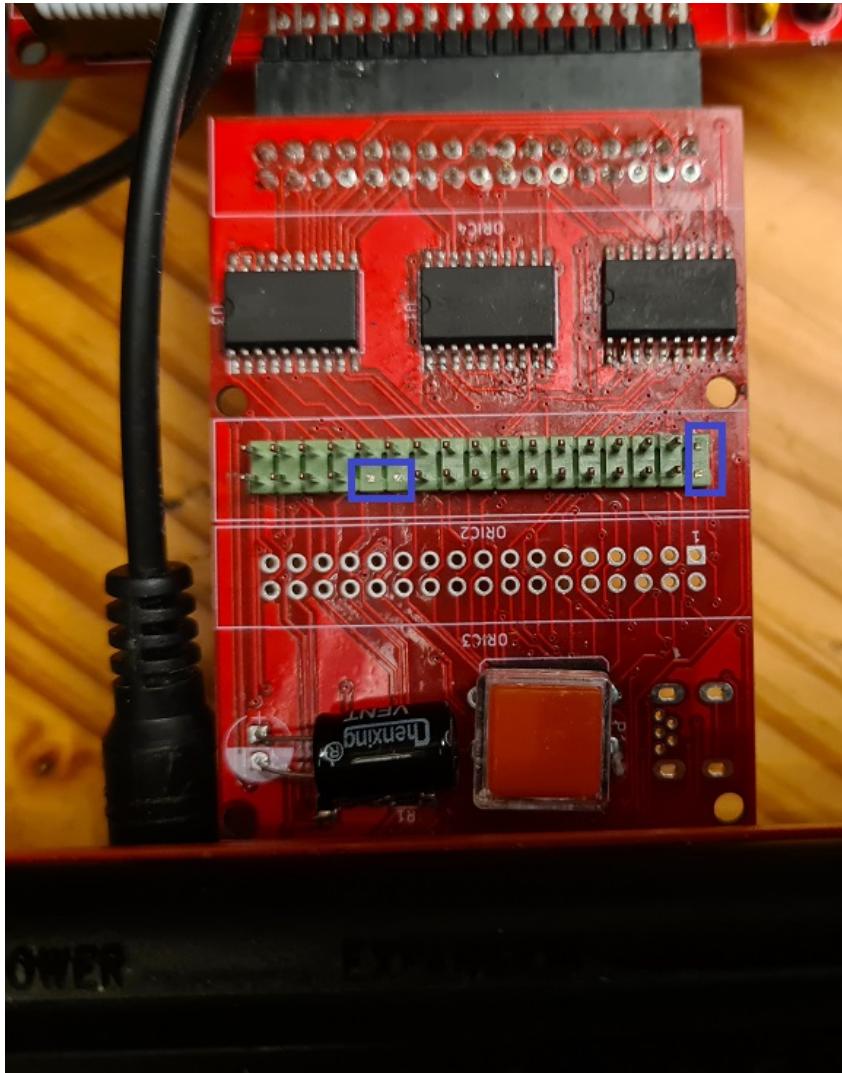
On the current firmware (Firmware 1) : and current hardware (board version v0.65), we have to do some hacks to have cumulus working. But, you will only launch two diskfile. Anyway, you can access to drive with no restriction, except bank switching. See « Hardware and firmware upgrade », if you want to avoid theses modifications

In firmware 1, and with board modification, there is only two working disk : Blake's 7 and VIP2015.

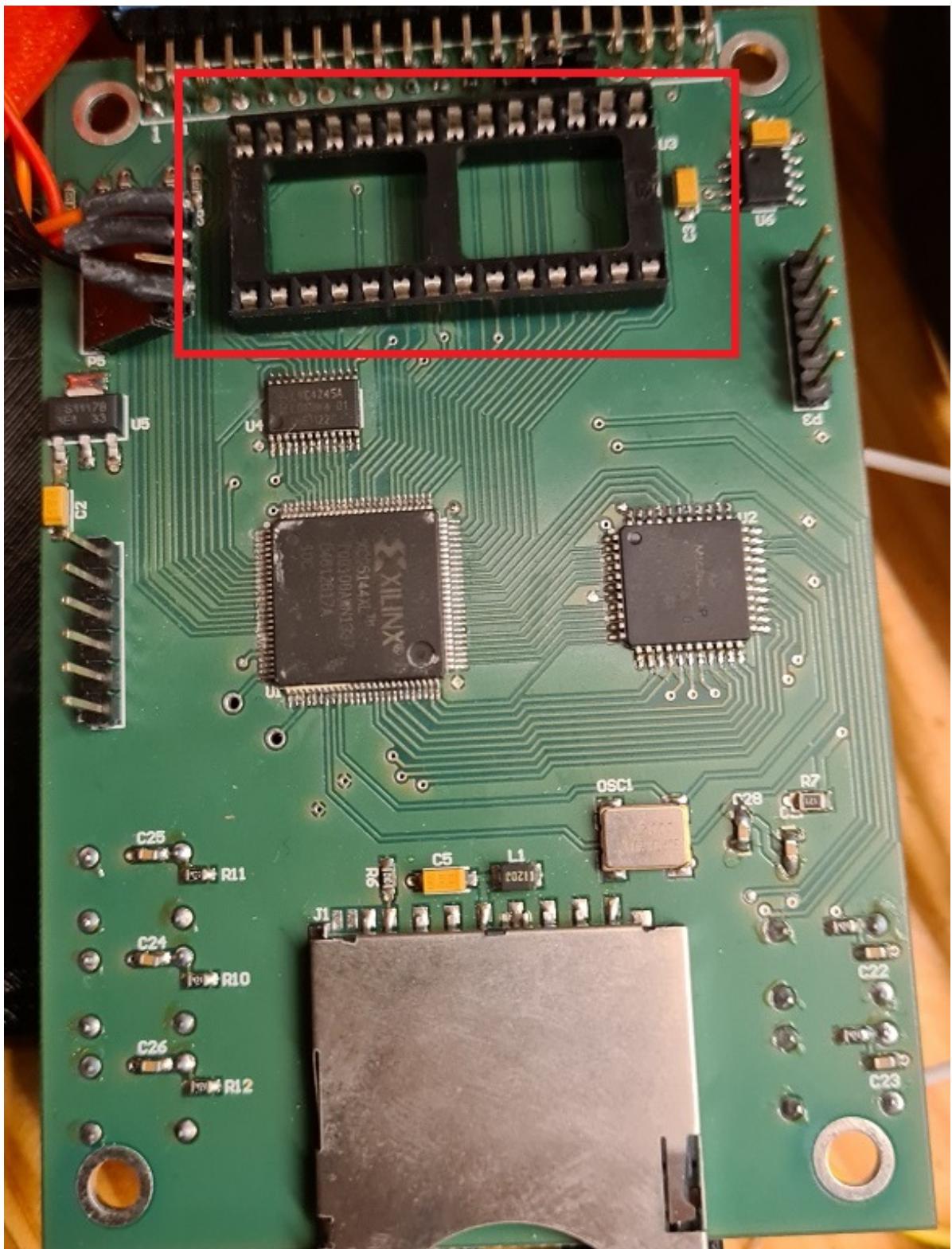
Please, test your cumulus on the oric connected to the board. If it does not work on your Oric, it won't work too with card plugged !

If you want to use cumulus, you have to :

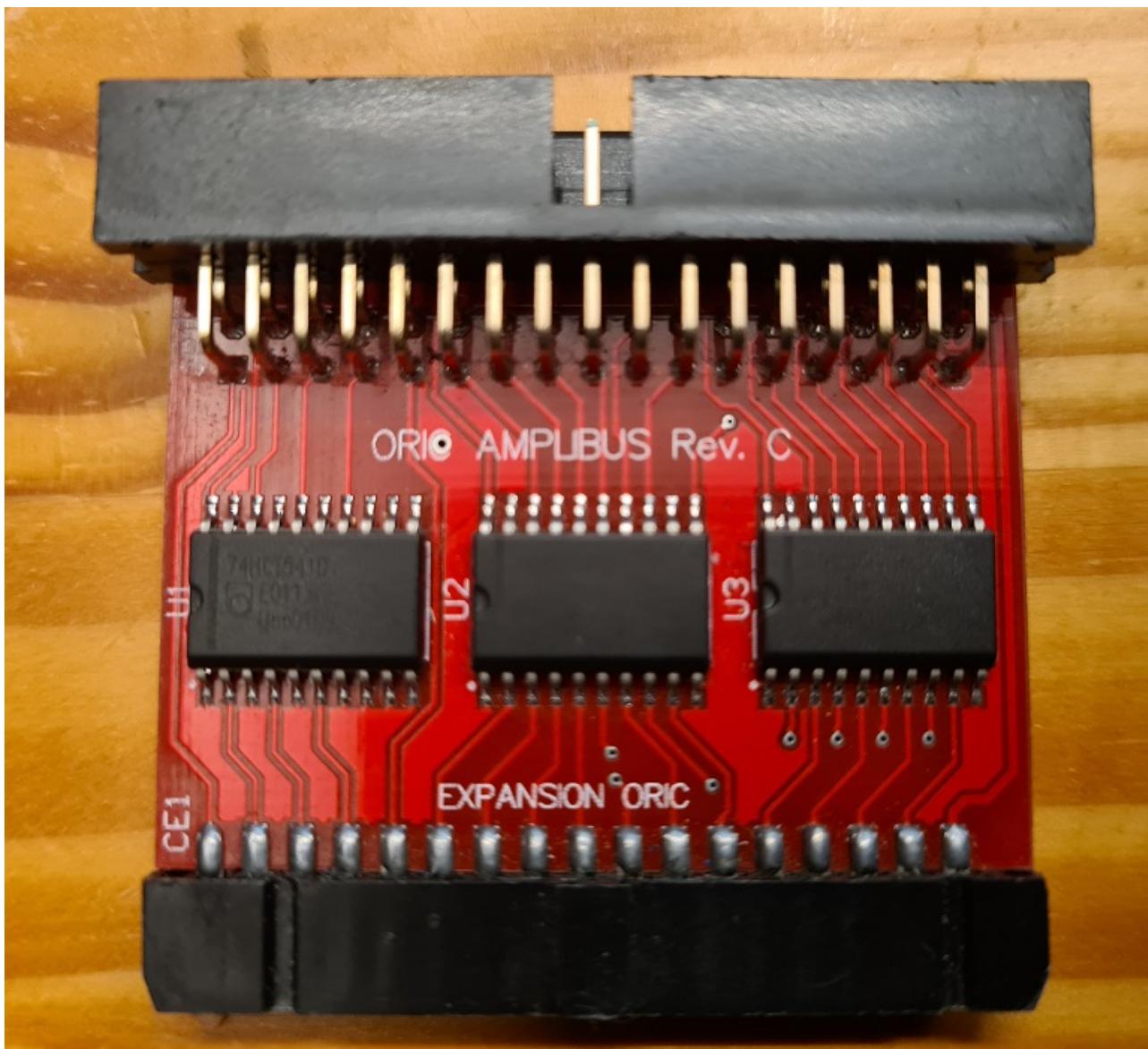
- 1) cut 4 pins on daughter card (ROMDIS, MAP, A14, A15)



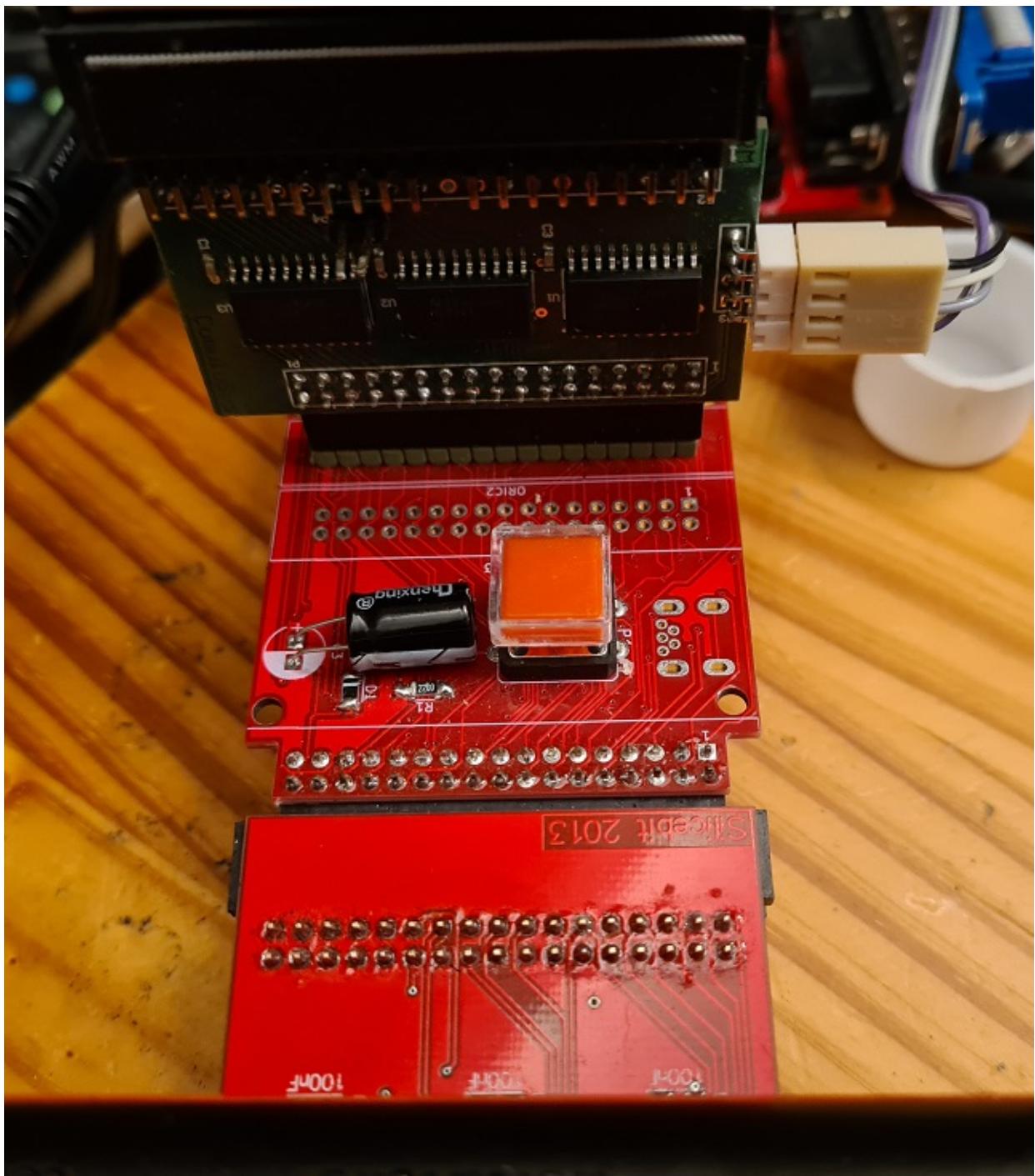
- 2) remove eprom from cumulus



3) add another amplibus before twilighte daughter board



4) Connect all cards to the oric



## Twilight board firmware compatibility

Only firmware 2 is available to use boot sector to start Microdisc disk.

## Hardware and firmware upgrade

### Firmware upgrade

There is only one firmware available. The version 2 is in development.

### First method : For those who have programmers and some hardware tool

But, when it will be released, you could update the firmware with :

- 1) a plcc extractor

- 2) altera software (Quartys v13)
- 3) a Jtag programmer
- 4) solder the jtag connector
- 5) get .pof file

**Second method : send the card to the author of the card (me)**

In that case, fimware upgrade will be done, and you could ask to upgrade to new board version to add (sometimes new functionnality)

# TROUBLE SHOOTING

## 'ls' displays garbage on screen

Insert your sdcard or your usb drive into your PC. You should have strange « file » on the sdcard : remove theses files.

## Impossible to mount a usb key or a sdcard

The sdcard must be in FAT32 format

## Screen garbage when i use bank

If you have screen garbage when you switched to ram bank before with « twil -w »

It means that ram bank are not initialized. See orixcfg section to fix it

## Pi zero always reboots

Check your PSU. If you have a 2A PSU and you have a pi zero, cumulus and TOM2 connected, you should reach the PSU limits. If you can't get another PSU, you can disable bluetooth of your pi zero, or you can also downclock from 1Ghz to 700mhz for example.

You can also use a 3A PSU. In the future, it will be possible to add another PSU on the board.

## When i start Orix, filesystem is unstable or displays usb controller not found

If you have pi zero connected, it could answer to the controller partial information or could hang the usb controller because controller does not understand usb data when it sends information to usb port.

You have to wait a bit. If you want to verify this, you can switch off the oric (and then the pi zero), switch on the oric with Orix, and type 'debug', if you have another value than #AA for ch376 check exists, it's the problem, if you do 'debug' another value will be displayed but not #AA. In fact, when pi zero boot, usb controller is unstable.

## « I/O Error » is displayed

You can reach this message in some case :

1. device (sdcard or usbdrive is missing)
2. after a launch of « df » command : There is an issue, the controller is in incorrect state after this command. It's a bug

You can usually fix it by launching « ls » twice. Because « ls » handles a reset order to the usb controller when it does not produce the correct answer. It means that if USB controller is not working well, ls displays the error message and will produce a reset command to the controller. If you launch ls again, it will work.

## The oric does not work : black screen

If you have a pi zero connected on usb port, unplug it. Boot the oric, and now insert pi zero into usb port

## Kernel panic

When kernel can't solve a « free memory kernel call» in a binary, it could produce a kernel panic. In that case, you need to do a reset. There is a bug in kernel 2021.1 which could produce this error. It will be corrected as soon as possible.

## A folder is displayed on my PC but not under my Oric

Sometimes sdcard or usbkey has bad format for the usb controller and it can reads some content. Format the usb key or sdcard and install again all files. Or try another usb key/sdcard

## I have strange behavior when un do csave or cload on basic ROM : It's always the same file event i cload another content

Sometimes sdcard or usbkey has bad format for the usb controller and it can reads some content. Format the usb key or sdcard and install again all files. Or try another usb key/sdcard

## **Garbage on basic screen**

When you start altered charset on screen when you start basic11 command (or funct+T), download again basic.tgz here :  
<http://repo.orix.oric.org/dists/official/tgz/6502/>

untar and unzip it on your device, it should fix this : the charset.chs file had been altered

## **Q&A**

### **I want to change the current directory**

See « cd » command

### **I want to see which bank are loaded into ROM and RAM**

See «bank» section

### **I want to read a .dsk file**

You can only extract files from a .dsk file (see « dsk-util »)

If you have a cumulus board, you can use « bootfd » and connect your cumulus on expansion board « see how to connect a cumulus section »

### **I can't type anything in basic rom (« basic11 » command)**

There is a firmware bug on some board which generate a false state for the third button of a joystick. The easier step to avoid this, is to connect a joystick to the left port on the board.

The issue can be fixed by upgrading firmware board (it needs to open the box and program the firmware with Jtag port)

# ANNEXES

## Firmware version

Version	Features	Known bugs
1	RAM/ROM switch, ROM programmation, joysticks, usb controller	N/A
2	Start all sedoric disks from cumulus	N/A

## Upgrade from older version

### Upgrade from v2022.1 to v2022.3

If your orix version is below v2022.1 version, please go to annexes part at the end of this document, before you try to upgrade to v2022.3

- Download <http://repo.orix.oric.org/dists/official/tgz/6502/sdcard.tgz>
- untar/gunzip sdcard.tgz (use 7zip under windows) on your device usb or sdcard : It could require some time to copy because there is a lot of small files (tap, hlp etc)
- you can start orix on real machine, and type :

```
/#cd usr  
/usr#cd share  
/usr/share#cd carts  
/usr/share/carts#cd 2022.3
```

If you want to usr usb drive for default device :

```
/usr/share/carts/2022.3#orixcfg -r -s 4 kernelus.r64
```

If you want to use sdcard for default device :

```
/usr/share/carts/2022.3#orixcfg -r -s 4 kernelsd.r64
```

- press ‘y’, and **wait until Orix reboots**

(Don’t switch off the Oric at this step)

### Upgrade from v2021.4 to v2022.1

If your card is below v2021.4 version, please go to annexes part at the end of this document, before you try to upgrade to v2022.1

- Download <http://repo.orix.oric.org/dists/official/tgz/6502/sdcard.tgz>
- untar/gunzip sdcard.tgz (use 7zip under windows) on your device usb or sdcard : It could require some time to copy because there is a lot of small files (tap, hlp etc)
- you can start orix on real machine, and type :

```
/#cd usr  
/usr#cd share  
/usr/share#cd carts  
/usr/share/carts#cd 2022.1
```

If you want to usr usb drive for default device :

```
/usr/share/carts/2022.1#orixcfg -r -s 4 kernelus.r64
```

If you want to use sdcard for default device :

```
/usr/share/carts/2022.1#orixcfg -r -s 4 kernelsd.r64
```

- press ‘y’, and **wait until Orix reboots**

(Don’t switch off the Oric at this step)

## Upgrade from v2021.3 to v2021.4

If your card is below v2021.3 version, please go to annexes part at the end of this document, before you try to upgrade to v2021.4

- Download <http://repo.orix.oric.org/dists/official/tgz/6502/sdcard.tgz>
- untar/gunzip sdcard.tgz (use 7zip under windows) on your device usb or sdcard : It could require some time to copy because there is a lot of small files (tap, hlp etc)
- you can start orix on real machine, and type :

```
#cd usr  
/usr#cd share  
/usr/share#cd carts  
/usr/share/carts#cd 2021.4
```

If you want to usr usb drive for default device :

```
/usr/share/carts/2021.4#orixcfg -r -s 4 kernelus.r64
```

If you want to use sdcard for default device :

```
/usr/share/carts/2021.4#orixcfg -r -s 4 kernelsd.r64
```

- press ‘y’, and **wait until Orix reboots**

(Don’t switch off the Oric at this step)

## Upgrade from v2021.2 to v2021.3

You need to unzip/untar orixcfg new version here : <http://repo.orix.oric.org/dists/2021.3/tgz/6502/orixcfg.tgz>

- Download <http://repo.orix.oric.org/dists/official/tgz/6502/sdcard.tgz> or <http://repo.orix.oric.org/dists/2021.3/tgz/6502/cardridge.tgz>
- untar/gunzip sdcard.tgz (use 7zip under windows) on your device usb or sdcard : It could require some time to copy because there is a lot of small files (tap, hlp etc)
- you can start orix on real machine, and type :

```
#cd usr  
/usr#cd share  
/usr/share#cd carts  
/usr/share/carts#cd 2021.3
```

If you want to usr usb drive for default device :

```
/usr/share/carts/2021.3#orixcfg -r -s 4 kernelus.r64
```

If you want to use sdcard for default device :

```
/usr/share/carts/2021.3#orixcfg -r -s 4 kernelsd.r64
```

- press ‘y’, and **wait until Orix reboots**

(Don’t switch off the Oric at this step)

## Upgrade from v2021.1 to v2021.2

If your card is below v2021.1 version, please go to annexes part at the end of this document, before you try to upgrade to v2021.2

- Download <http://repo.orix.oric.org/dists/official/tgz/6502/sdcard.tgz>
- untar/gunzip sdcard.tgz (use 7zip under windows) on your device usb or sdcard : It could require some time to copy because there is a lot of small files (tap, hlp etc)
- you can start orix on real machine, and type :

```
#cd usr  
/usr#cd share  
/usr/share#cd carts  
/usr/share/carts#cd 2021.2
```

If you want to use usb drive for default device :

```
/usr/share/carts/2021.2#orixcfg -r -s 4 kernelus.r64
```

If you want to use sdcard for default device :

```
/usr/share/carts/2021.2#orixcfg -r -s 4 kernelsd.r64
```

- press ‘y’, and **wait until Orix reboots**

(Don’t switch off the Oric at this step)

From 2020.4 to 2021.1

Download : <http://repo.orix.oric.org/dists/2021.1/tgz/6502/carts.zip>

Unzip it on your device (sdcard/usbkey)

- you can start orix on real machine, and type :

```
/#cd usr  
/usr#/cd share  
/usr/share#/cd carts  
/usr/share/carts#/cd 2021.1
```

If you want to use usb drive for default device :

```
/usr/share/carts/2021.1#orixcfg -r -s 4 kernelus.r64
```

If you want to use sdcard for default device :

```
/usr/share/carts/2021.1#orixcfg -r -s 4 kernelsd.r64
```

- press ‘y’, and **wait until Orix reboots**

(Don’t switch off the Oric at this step)

## Optionnal step for upgrade

Now bank displays all banks from 1 to 64. It means that you should have some strange bank signature for eeprom. Now an empty set is provided in */usr/share/carts/2021.3* folder. With Orixcfg you can initialize your set with this cart. Don’t use « -s 4 » flag for orixcfg when you want to load emptyset.