

# Turbo Chameleon 64 User Manual

Draft Version (Beta 9q)

01/12/21



# 1. Contents

|  |           |
|--|-----------|
| <b>1. Contents.....</b>                            | <b>2</b>  |
| <b>2. Getting started.....</b>                     | <b>5</b>  |
| 2.1. What's in the Box?.....                       | 6         |
| 2.1.1. Chameleon v1.....                           | 6         |
| 2.1.2. Chameleon v2.....                           | 7         |
| 2.2. Operation in standalone mode.....             | 8         |
| 2.2.1. Using the docking station.....              | 10        |
| 2.3. Operation in cartridge mode.....              | 12        |
| 2.3.1. Operation on a C64.....                     | 12        |
| 2.4. Sound output.....                             | 13        |
| <b>3. Updating the Chameleon.....</b>              | <b>14</b> |
| 3.1. Updating using update.prg.....                | 14        |
| 3.2. Updating via USB.....                         | 15        |
| 3.2.1. Preparations.....                           | 15        |
| 3.2.2. Using the flasher program.....              | 15        |
| 3.2.3. Using ChaCo.....                            | 15        |
| <b>4. First Aid - If something goes wrong.....</b> | <b>16</b> |
| <b>5. Overview.....</b>                            | <b>17</b> |
| 5.1. Flash memory organization.....                | 17        |
| 5.2. Using disk images.....                        | 18        |
| 5.2.1. Disk slot organization.....                 | 18        |
| 5.2.2. Mounting images.....                        | 18        |
| 5.2.3. Unmounting images.....                      | 18        |
| 5.3. Using cartridges.....                         | 19        |
| 5.3.1. Cartridge slot organization.....            | 19        |
| 5.3.2. Mounting cartridge images.....              | 20        |
| 5.3.3. some notes on specific cartridges.....      | 21        |
| 5.4. Using custom ROM images.....                  | 22        |
| 5.5. Using configuration files.....                | 22        |
| 5.5.1. Format details.....                         | 22        |
| 5.5.2. Example.....                                | 25        |
| <b>6. Chameleon Menu System.....</b>               | <b>26</b> |
| 6.1. Boot Loader.....                              | 26        |
| 6.1.1. Hotkeys during boot process.....            | 27        |
| 6.1.2. Limitations of the boot-loader.....         | 27        |
| 6.1.3. Example: boot.prg.....                      | 28        |
| 6.2. Keys.....                                     | 29        |
| 6.3. Main Menu.....                                | 30        |
| 6.4. Options.....                                  | 32        |
| 6.4.1. CPU Turbo.....                              | 32        |
| 6.4.2. Emulated drives.....                        | 33        |
| 6.4.3. Cartridges and Memory Expansions.....       | 34        |
| 6.4.4. Input devices.....                          | 35        |
| 6.4.5. VGA output.....                             | 35        |
| 6.4.6. Emulation settings.....                     | 36        |
| 6.4.7. RTC.....                                    | 36        |
| 6.4.8. Boot-loader settings.....                   | 36        |
| 6.4.9. Button settings.....                        | 37        |
| 6.4.10. File Browser and Menu settings.....        | 37        |
| 6.4.11. Load and Save the settings.....            | 37        |
| 6.4.12. Buttons.....                               | 38        |
| 6.5. File Browser.....                             | 40        |
| 6.5.1. Keys.....                                   | 40        |

## Contents

---

|  |           |
|--|-----------|
| 6.5.2. User / File menu.....                               | 42        |
| 6.5.3. File-Action menu.....                               | 43        |
| 6.5.4. Supported File types.....                           | 44        |
| 6.5.5. Plugins.....  | 45        |
| 6.5.6. Limitations of the file-browser.....                | 48        |
| 6.6. ML Monitor.....                                       | 49        |
| 6.6.1. Features.....                                       | 49        |
| 6.6.2. Number formats.....                                 | 49        |
| 6.6.3. Converting values.....                              | 50        |
| 6.6.4. Commands.....                                       | 50        |
| 6.6.5. Keyboard.....                                       | 52        |
| 6.7. Online Help.....                                      | 52        |
| 6.7.1. Keys.....   | 52        |
| 6.8. System Info.....                                      | 53        |
| <b>7. ChaCo, the Chameleon Control Software.....</b>       | <b>54</b> |
| 7.1. Preparations.....                                     | 54        |
| 7.1.1. Linux / macOS.....                                  | 54        |
| 7.2. Linux notes.....                                      | 54        |
| 7.3. macOS notes.....                                      | 54        |
| 7.4. Overview.....   | 55        |
| 7.4.1. Slot Management.....                                | 56        |
| 7.4.2. Memory Access.....                                  | 56        |
| 7.4.3. Debugging.....                                      | 56        |
| 7.4.4. Status display.....                                 | 56        |
| 7.5. Examples.....   | 57        |
| 7.5.1. Starting a core.....                                | 57        |
| 7.5.2. Flashing a Core.....                                | 58        |
| 7.5.3. Accessing memory.....                               | 59        |
| 7.6. Command-line options.....                             | 62        |
| 7.7. chacocmd.....   | 63        |
| 7.7.1. examples.....                                       | 63        |
| <b>8. Frequently asked questions.....</b>                  | <b>64</b> |
| <b>9. known Bugs and Limitations.....</b>                  | <b>64</b> |
| <b>10. Reporting Bugs.....</b>                             | <b>64</b> |
| <b>11. Developer information.....</b>                      | <b>65</b> |
| <b>12. Related projects.....</b>                           | <b>66</b> |
| <b>13. Appendix.....</b>                                   | <b>67</b> |
| 13.1. PS/2 Keyboard Key Assignment.....                    | 67        |
| 13.2. PS/2 Keyboard Joystick Assignment.....               | 68        |
| 13.3. Amiga Keyboard Key Assignment.....                   | 69        |
| 13.4. Amiga Keyboard Joystick Assignment.....              | 70        |
| 13.5. CDTV Remote Control Key Assignment.....              | 71        |
| 13.6. supported Cartridge types.....                       | 72        |
| 13.7. Opening the Cartridge.....                           | 73        |
| 13.7.1. Chameleon v1.....                                  | 73        |
| 13.7.2. Chameleon v2.....                                  | 74        |
| 13.8. Re-programming the USB micro-controller.....         | 75        |
| 13.8.1. Chameleon v1.....                                  | 75        |
| 13.8.2. Chameleon v2.....                                  | 77        |
| 13.8.3. Linux.....   | 80        |
| 13.9. configuring Quartus for JTAG.....                    | 81        |
| 13.10. Chameleon breakout-cable pinout (Chameleon v1)..... | 88        |
| 13.11. mini IEC cable pinout (Chameleon v2).....           | 90        |
| 13.12. Technical Data.....                                 | 91        |
| 13.12.1. Turbo Chameleon FPGA Core.....                    | 91        |
| <b>14. Credits.....</b>                                    | <b>92</b> |

|                            |           |
|----------------------------|-----------|
| <b>15. Fine print.....</b> | <b>93</b> |
|----------------------------|-----------|

## 2. Getting started

Thank you for purchasing an Individual Computers product. The Chameleon is one of the most complex projects that we've ever made for any computer, and it will take much more than this user manual to explain all the features and possibilities of this cartridge. Take your time to read all of this document.

You've been made aware a number of times that the current state of the firmware is beta, and we appreciate that you've bought the unit anyway. If you find any bugs, please report them in our forum:

<http://forum.icomp.de>

This is where the developers meet, and where your questions will be answered by the people who made the cartridge. We will also make important announcements on that list, so even if you don't have anything to report, please subscribe to that list.

Please download the latest version of the FPGA Core and ChaCo, the Chameleon Control software, from:

<http://wiki.icomp.de/wiki/Chameleon>

## 2.1. What's in the Box?

### 2.1.1. Chameleon v1



- upper left corner: mini USB cable (for data transfers, 180cm)
- middle bottom: Turbo Chameleon 64 :-)
- upper right corner: breakout cable with connectors for USB (power supply, 80cm), keyboard and mouse (both PS/2, 20cm), IEC (DIN, 20cm)
- a short user manual (which you do not need any more now that you have found this one)

A battery (CR-1220) required to make use of the RTC feature is **not** included and must be supplied by the user. See page 73, 13.7 Opening the Cartridge.

### 2.1.2. Chameleon v2

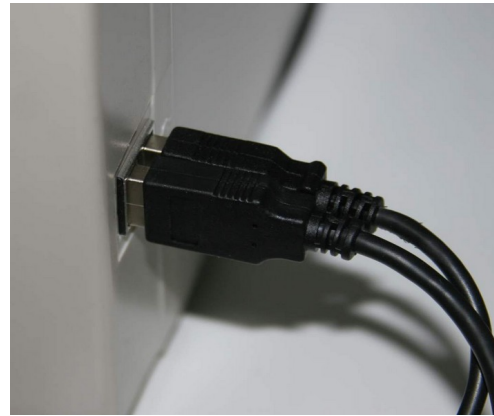
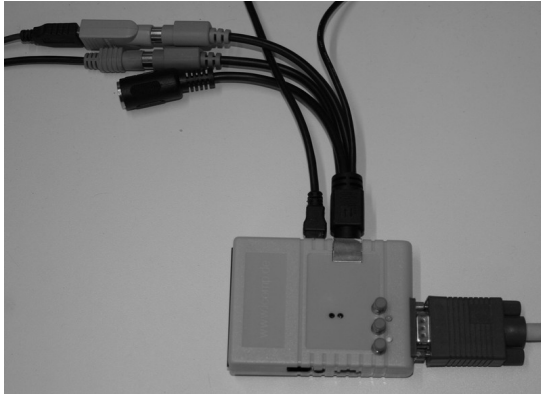


- Turbo Chameleon 64 :-)
- a short user manual (which you do not need any more now that you have found this one)

A battery (CR-1220) required to make use of the RTC feature is **not** included and must be supplied by the user. See page 73, 13.7 Opening the Cartridge.

Also not included is a mini USB cable, since most customers already own more of them than they need. Should this not be the case just buy one at your local consumer electronics store.

## 2.2. Operation in standalone mode



In stand-alone mode, you will need at least the breakout cable to connect to Chameleon, because it will provide power to the cartridge. With the second version of the Chameleon, please use the mini USB connector at the bottom/left for power:



Make sure that the power supply can deliver at least the 2.5W (5V and 500mA) that's specified for a USB connection. The Chameleon hardware will take about 2 Watts of power, so make sure that your keyboard and mouse don't overload the power supply! Most PCs don't have any trouble supplying Chameleon with power. Note that some PCs don't provide power without a data connection on a USB port. In that case, please use a separate power supply.

Connect your monitor to the VGA output. By default the Turbo Chameleon 64 FPGA core uses a standard 800x600 mode with a refresh rate of 72Hz. Some monitors don't support this high frequency. Please consult your monitor's manual if you have any trouble with Chameleon's VGA output.



Connect your keyboard to the purple PS/2 connector, and your mouse to the green PS/2 connector. The IEC connector can be left open. If you want to use ChaCo with the cartridge, you also need to connect the mini-USB cable to that PC. (see chapter 7 ChaCo, the Chameleon Control Software)

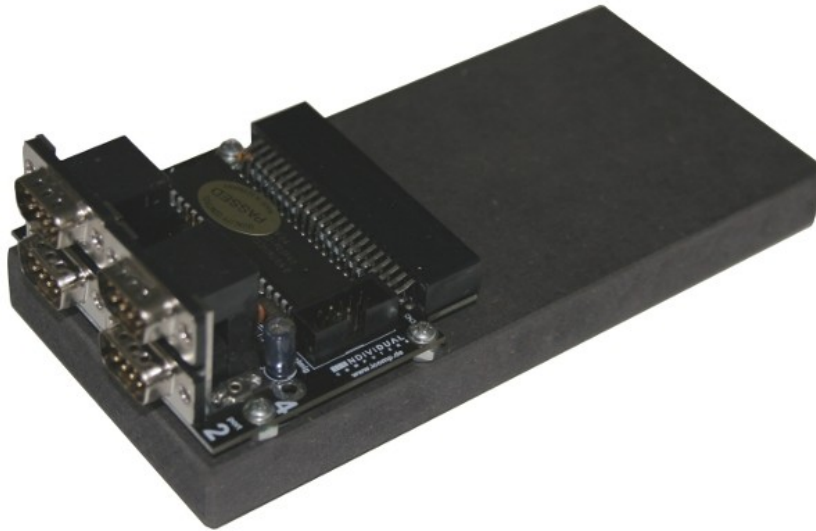
For the second version of the chameleon you need a mini IEC cable, which is available separately, to connect IEC-bus devices. It is not required if you only use the floppy drive emulation.



### 2.2.1. Using the docking station

The Chameleon docking station extends the available ports of your Chameleon by the following:

- four joystick ports
- one C64 keyboard connector
- one Amiga 500 keyboard connector



Both keyboard connectors are keyed, which means that they have a missing pin in order to help you finding the correct orientation of the connector. However, the A500 keyboard connector is not always keyed, so you have to take a close look at the colours. The black wire of the A500 keyboard is marked next to the connector. Wrong connection may destroy both, the docking station and the keyboard! Better double-check before you apply power to the unit.

All C64 keyboards that we have encountered have been keyed. There is no danger when plugging in the keyboard the wrong way round – you will just observe funny behaviour. Caution: This only applies to the Chameleon docking station! You may actually destroy a C64 if you connect the keyboard the wrong way round!

Please download and install Chameleon core Beta-7 or higher before you use the docking station. It's available from <http://wiki.icomp.de/wiki/Chameleon>. Older cores will not support the docking station, and the ports won't be of any use to you. For flashing, please disconnect all external hardware: Remove the mouse, remove the PS/2 keyboard and remove the SD card from the slot. Do not connect the docking station to Chameleon when you're updating the flash!

There is only one way that Chameleon fits the docking station: It's with the button side up. Slide the cartridge all the way in until the black connector is at least covered half by the Chameleon. Make sure that there's proper contact between the docking station and the Chameleon – the cartridge may not be tilted at all.



Connect the breakout cable and supply power to the whole setup through the USB connector of the breakout cable (first version of the chameleon) or the mini USB connector at the bottom/left (second version of the chameleon). Make sure that your power supply is strong enough! With the docking station, we recommend to use a 5V/1000mA power supply. If available, use a 5.2V/1000mA power supply, but **never** use any higher voltage than 5.2V. Only use regulated power supplies.

The pinout of the joystick ports follow the common “Atari standard”:

| Pin | Function                           |
|-----|------------------------------------|
| 1   | /Up                                |
| 2   | /Down                              |
| 3   | /Left                              |
| 4   | /Right                             |
| 5   | /Button 3 (POTY on C-64 and Amiga) |
| 6   | /Button 1                          |
| 7   | VCC (+5V)                          |
| 8   | GND                                |
| 9   | /Button 2 (POTX on C-64 and Amiga) |

Note: Unlike the POT lines of a real C-64, **Pin 5 and 9 must be pulled to GND instead of VCC**. For details see <http://wiki.icomp.de/wiki/DB9-Joystick>

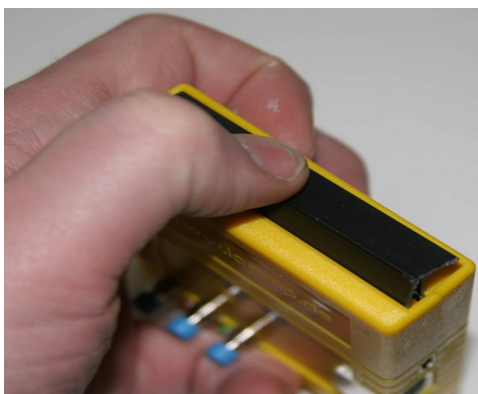
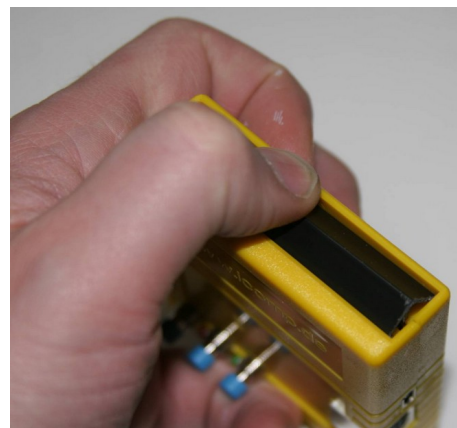
## 2.3. Operation in cartridge mode

WARNING: Chameleon is a C64 cartridge. **It does not work in the C128.** Don't even try, as this **might cause damage to the computer, the cartridge or both.** Don't experiment, it can not work and you risk damage to valuable equipment that's not covered by warranty.

WARNING: Also, Chameleon does not work in cartridge port expanders – neither as the only cartridge, nor together with other cartridges. It will only work if it's plugged directly to the C64. **Don't experiment with combinations of cartridges!** It cannot work, and you risk damage to valuable equipment that's not covered by warranty.

### 2.3.1. Operation on a C64

Remove the protective plastic from the Chameleon expansion connector. Just push it in and rotate it until it comes out – be gentle, there's no force required! The brown rubber is glued in place. Don't try to peel it out, it is required if you want to put the protection back in. The rubber will not conflict with the C64's expansion port.



Now insert the cartridge into the C64 expansion port with the buttons facing up. Connect your VGA monitor to the VGA output of Chameleon. At this point, we recommend to also connect your old 15kHz screen, so you can compare the output of Chameleon against the output of the original VIC chip.

Use of the breakout cable is optional on the C64. If you want to use a PS/2 keyboard or mouse, you'll need it (first version of the chameleon only).

If you want to use the ChaCo software with this setup, you also need to connect a mini-USB cable to your PC. (see chapter 7 ChaCo, the Chameleon Control Software)

### ***Note for SX-64 users***

- If for some reason the Chameleon fails to start the menu system, power-cycle the computer and hold "2" during power-on. This will make the menu use a fail safe configuration (skips booting from SD card and copying ROMs from C64)
- There seems to be a timing problem that makes copying ROMs from the C64 fail on some SX-64s, this will result in a crash when exiting the menu system. If this is the case, disable the "copy ROMs from C64" option (see chapter 6.4 Options), write options and power-cycle the computer. Optionally you can enable "boot from SD card" and put a SX-64 kernal (or whatever else you want to use) ROM image on it (see chapter 6.1 Boot Loader).

## ***2.4. Sound output***

Chameleon and C64 each have their own audio outputs, the Chameleon doesn't get the audio from the C64 and the C64 doesn't get the audio from the Chameleon so each must be connected separately.

The Chameleon has a stereo output (it can emulate 2 SIDs). You connect it to your speakers with an 3.5 mm stereo-audio plug similar what portable players and many laptops have. You need speakers with build-in amplifier (or separate amplifier as Chameleon doesn't have one).

### 3. Updating the Chameleon

Each release archive contains a combined ChaCo, core and software update. The two files needed to update manually (\*.rbf for the core and \*.bin for the menu and ROMs) can be found in the UPDATE directory. The CHAM64 directory contains files used by the menu system, such as the online help. Just copy the entire directory to the root of your SD card.

Updates are made available at <http://wiki.icomp.de/wiki/Chameleon> on a regular basis. Please make sure that you are running the latest firmware and are using the latest version of ChaCo before reporting problems.

#### 3.1. Updating using update.prg

Copy the UPDATE directory and UPDATE . PRG to the root of your SD card (or simply unpack the entire release archive to it), insert the SD card and power-cycle the Chameleon. (If you are updating from beta-7a or earlier, run UPDATE . PRG from the file-browser.) You can do this either in standalone mode or leave the Chameleon connected to your C-64.

In addition to the files contained in the release archive, the UPDATE directory may contain the following custom ROM files:

- "BASIC . ROM" - C64 basic ROM
- "KERNAL . ROM" - C64 kernal ROM
- "CHARGEN . ROM" - C64 character ROM
- "MMC64 . ROM" - MMC64 BIOS ROM
- "SLOT1 . ROM" - Cartridge slot 1 ROM (\*1)
- "SLOT2 . ROM" - Cartridge slot 2 ROM (\*1)
- "DRIVE . ROM" - 1541 drive ROM

\*1) Cartridge ROMs must be plain binaries (without load address), NOT .crt files. Files in proper format can be created by the ROM saver (see chapter 6.5.2 User / File menu ). Cartridges which are bigger than 64k can not be used in slot 1 and 2, see chapter 13.6 supported Cartridge types for details. To make cartridges readily available for use, the respective cartridge type must be configured in the settings first (see chapter 6.4 Options).

It is highly recommended to first test custom ROM files by loading them from SD card (see 6.1 Boot Loader) before flashing them permanently. Flashing broken ROM files may render the C64 unusable (you will have to fix it by updating the chameleon via USB in that case).

## 3.2. Updating via USB

### 3.2.1. Preparations

Before updating your Chameleon, please make sure that it has sufficient power. Any and all trouble that has been reported so far was related to power problems that were caused by external power draining. To be safe, please:

- disconnect the Chameleon docking station
- disconnect the PS2 keyboard
- disconnect the PS2 mouse
- you **MUST** remove the SD/MMC card from the card slot (else you will get an error message and can not update)

You may provide power via the USB breakout cable (first version of the Chameleon) or the mini USB connector at the bottom/left (second version of the Chameleon), or leave the Chameleon plugged into your C-64.

If you are using a USB charger as your power supply, please make sure it's a 1000mA type. Although Chameleon takes just over 400mA of power, many 500mA types are already at their limit with a Chameleon. If possible, use a 5.2V/1000mA type instead of 5V/1000mA. However, do **not** go over 5.2V for supply voltage!

### 3.2.2. Using the flasher program

Windows users just double-click the `flasher.exe` file in Windows explorer and follow the instructions on screen. In Windows 8 and above use right mouse click and choose "Run as Administrator" (on `flasher.exe`).

Linux users may run the respective flasher program from a terminal. Since the executable bit of the binary is not preserved in the archive, you will have to set it manually first like this:

```
$ sudo chmod +x ./flasher-linux64
```

After that run the flasher as root:

```
$ sudo ./flasher-linux64
```

In addition to the files contained in the release archive, the `UPDATE` directory may contain custom ROM files as described in Updating using `update.prg` above.

### 3.2.3. Using ChaCo

You may also update the Chameleon manually using ChaCo or `chacocmd`, you can find the required files in the `UPDATE` directory. (See chapter 7.5.2 Flashing a Core)



## 4. First Aid - If something goes wrong

First, don't panic :)

There is no way to “brick” the Chameleon by updating, installing an additional core, or wrong programming. The Chameleon might not start, but it can always be fixed by updating with a valid FPGA core image. In case updating using UPDATE.PRG failed (please report this), re-try updating via USB (see 3.2 Updating via USB above).

If you have trouble flashing, try again. Before a re-try:

- Double check that you have sufficient power. Maybe try another USB port.
- If the USB connection seems unstable and stops working randomly make sure to power the Chameleon from the same PC/Hub the USB data cable is connected to.
- Make sure to select the correct files (when updating manually). Note that the first hardware version (with breakout cable) requires a different core file than the second hardware version (without breakout cable).
- PLEASE POWER-CYCLE YOUR CHAMELEON. It may then give an error-blink on the red LED, but there's nothing to worry about. Updating will fix it.

If the Chameleon does not start correctly, please try the following:

- Disconnect all peripherals except the monitor from the Chameleon. In many cases it will now start properly again. To narrow down the problem, proceed as follows:
- Make sure that “shift lock” is not active. If that is the case, the bootloader will indicate it by showing “waiting for keys released”. If you are using a PS/2 keyboard and the problem persists, the keyboard might be defective.
- Check that at a connected joystick “auto fire” is disabled – a typical symptom for that is when the chameleon will leave the menu immediately and starts BASIC.

If C-64 programs do not work as expected anymore:

- Reset the configuration to its default values (“restore defaults” in the options menu). Many programs are incompatible with one or more expansions emulated by the Chameleon.
- Connect only one floppy drive to the system, including emulated and real drives. Many programs do not work when more than one drive is connected to the IEC bus.



## 5. Overview

Upon power-up, Chameleon will load the OS from its on board flash. It will then start the menu system which lets you perform various actions. (see chapter 6 Chameleon Menu System)

The outermost button (right button) is your reset button. The middle button is the freeze and menu button. The left button can be used to perform various actions such as change disks in the emulated floppy.

Press and hold the reset button for more than 0.7 seconds to re-launch the Chameleon core. This is (almost) equivalent to a fresh power-up. Press and hold the Freeze button for more than 0.7 seconds to enter the menu system.

| Button        | PS/2 Key | remote  | Function   |
|---------------|----------|---------|--|
| Left button   | F11      | GENLOCK | Multi purpose, see chapter 6.4 Options                 |
| Middle button | F12      | CD/TV   | Freeze, hold longer than 0.7s to enter menu            |
| Right button  | PrntScrn | POWER   | Reset, enter menu if "reset to menu" option is enabled |

### 5.1. Flash memory organization

There is a flash chip built into Chameleon that's purely for system purposes and hence can't be utilized by the user. The flash carries FPGA cores and their respective ROM files. The FPGA core is the true core of Chameleon's operations – it defines the hardware behaviour. If we fix a bug in the hardware (such as a VIC effect emulation or register functions of a freezer), we will provide the fix in the form of an FPGA core file.

The size of the on-board flash is 16 MByte, organized in 16 slots of 1 MByte each. Every slot can carry one FPGA core and the corresponding ROM file. If Chameleon is powered up, the FPGA is started from slot number 0.

If starting the FPGA was not successful, then the micro-controller will tell you so by sending out a blink code on the red LED of Chameleon. The blink code will tell you which slot was attempted to be used. Just count the number of blinks, subtract one and you'll know which slot of the flash chip is corrupted. If for example slot 0 is corrupted, then the LED will flash once, then pause for a second and start over. If the LED blinks four times, then slot number three is the one you should take a closer look at.

## 5.2. Using disk images

### 5.2.1. Disk slot organization

There are two emulated 1541 floppy drives of which each handles up to 4 different disk images which can be mounted via the file browser and then changed by pressing the left button (see chapter 6.4 Options).

### 5.2.2. Mounting images

Before you can mount a disk image, one of the emulated drives must be enabled, and to use more than one disk slot per drive, the left button must be configured accordingly (see chapter 6.4 Options).

To mount a D64 or G64 image press CBM+M on the selected image or RETURN on a PRG file inside an image (see chapter 6.5.1 Keys). You will then be prompted to select a drive and whether to start a new list (“first disk”) or to add the image to the existing list (“next disk”) for this drive. Starting a new list means that all mounted images for this drive will get unmounted and the selected image will become the first image in the list for the selected drive.

#### ***Mounting using swaplist files***

Swaplist files are simple text files that contain one file name per line. Pressing CBM+M or RETURN on such a file will mount up to 4 disk images into the respective drive slots. If not already done so, you will also be prompted to configure the left button for disk change.

Swaplist files are recognized by the file extensions .lst (introduced by sd2iec) and .vfl (introduced by VICE). Swaplists created by the mentioned devices/emulators are compatible with chameleon, with one limitation: file names should have **no** path component, only files in the same directory as the swaplist file will get recognized.

A swaplist that contains the currently mounted images can be saved using the User / File menu in the file-browser (press F4). Before doing this, change the current directory to the one containing the currently mounted images.

### 5.2.3. Unmounting images

To write changes made to a mounted image back to the D64 file on SD card you will either have to make a copy manually (see chapter 6.5.2 User / File menu ) or unmount it by pressing CBM+U (see chapter 6.5.1 Keys). Also the menu will check if changes to an image have been made when entering the menu system and when mounting images, and ask whether the image on SD card should get updated or not.

**WARNING: If you choose to ignore this the menu system will “forget” about the respective image being changed and not ask again. You will have to make sure to save the image manually if you want to keep these changes.**

Note: due to the current limits of the drive emulation, G64 images will not get written back automatically (as the disk layout is not exactly preserved, which would most likely result in a corrupted G64)

### **5.3. Using cartridges**

Please note that a real-world cartridge consists of two major building blocks: The hardware, and the software. By switching a slot to a specific cartridge type, you merely select the hardware, but you don't have the contained software yet. Real C64 cartridges bring their software typically on EPROMs, which are emulated in the RAM space of Chameleon. Just choosing a cartridge type for a slot therefore does not mean that you actually can already use the cartridge, because it would appear like an empty shell to the computer. Compared to a real-world cartridge, it can be compared to removing the EPROM chips from the cartridge and then plugging it into the computer, which will not work.

Note that the EPROM contents of the Retro Replay freezer and the MMC64 come with every menu system, which is why you can use these cartridges after power-up and without an SD-card. However, it would be a waste of memory space to keep all the EPROM contents of all supported cartridges in the menu system: They would not even fit in Chameleon's memory! This is why all other cartridges require the use of a .crt or .rom file containing a binary image of the cartridge EPROM. Please see chapter 5.3.2 Mounting cartridge images below for a detailed description of how to use cartridge image files.

#### **5.3.1. Cartridge slot organization**

There are 4 cartridge slots. Slot 1 and 2 can hold cartridges of max. 64k, slot 3 and 4 can hold max 1Mb. Slot 1 and 2 will get loaded from flash when the Chameleon boots, by default there is a PAL Retro Replay ROM in slot 1 and an NTSC Retro Replay ROM in slot 2. Additional/other cartridge ROMs can be loaded by the boot-loader (see chapter 6.1 Boot Loader) or by the CRT loader in the file-browser (see chapter 6.5 File Browser). Slot 3 is configured as expert cartridge by default, and slot 4 remains empty.

### 5.3.2. Mounting cartridge images

Cartridge images can be mounted by the boot loader (see chapter 6.1 Boot Loader) or the file browser. In the file browser simply press return on a ROM or CRT file. Unless configured to always use slot 4, you will be asked which cartridge slot to use and the cartridge will be started.

Note that ROM files must be in a Chameleon specific layout (which may be different from what you get from converting a CRT file to a flat binary using common tools such as cartconv), you can create such files using the ROM saver in the file browser (see chapter 6.5.2 User / File menu ). Also when using ROM files, you will have to set up the correct cartridge type manually.(see chapter 6.4 Options).



### **5.3.3. some notes on specific cartridges**

#### ***Expert Cartridge***

The original expert cartridge has a 3-position switch that is emulated by the left button on the Chameleon. The button configuration must be set to "cartridge" when using the expert emulation. A short press toggles the expert emulation between ON and OFF (green LED is lit when the cartridge is on). A long press puts the expert in programming mode PRG (green LED is flashing when the cartridge is in programming mode).

Please understand that this manual cannot cover the use of all cartridges in detail. Using this cartridge requires software on a disk (image) that is not included with Chameleon.

As the Expert Cartridge is RAM based, it's contents may change – you may save the current contents by using the ROM saver, see 6.5.2 User / File menu below.

#### ***Easyflash***

The Flash-ROM chip of the Easyflash is not emulated. Instead if the EAPI signature is found in the first ROM bank, the EAPI area will get replaced by one that properly updates the chameleon's RAM. That means actually flashing it using the regular programming tool will not work (because there is no flash chip) – but cartridges that incorporate the EAPI will.

You may save the current contents by using the ROM saver, see 6.5.2 User / File menu below.

#### ***Retro Replay***

The Flash-ROM chip of the Retro Replay is not emulated. To permanently replace the default ROM you may flash it as described in 3.1 Updating using update.prg above.

#### ***MMC64***

The Flash-ROM chip of the MMC64 is not emulated. To permanently replace the default ROM you may flash it as described in 3.1 Updating using update.prg above.

## 5.4. Using custom ROM images

You may use custom ROM files that replace the kernal, basic and character ROMs of the c64 or the ROM of the emulated 1541 floppy drives. To do this you may either place the respective ROM images in the CHAM64 directory and enable booting from SD card (see chapter 6.1 Boot Loader below) or load the respective images through the file browser (see chapter 6.5 File Browser).

Note that loading custom ROM images has several implications. For example loading a broken kernal will make the C64 crash.

Also note that the timing of the drive emulation is not perfect yet, which causes problems with some speed loaders whose timing is very much “on the edge”. One such known case is Jiffy DOS on PAL setups, which is even unreliable on some real C64s, so problems are expected.

## 5.5. Using configuration files

Some configuration options – the options that might need adjustment for individual programs – can be saved to and loaded from configuration files. These files can be created using the User / File menu in the file browser, and the browser can be configured to load these settings automatically when certain images are mounted (.d64, .g64, .t64, .prg, .p00, .lst).

To create a configuration file configure the Chameleon as needed first, and test the program you want to run with this configuration. Once you are satisfied and everything works as expected, navigate to the image that contains the program and use the User / File menu (press F4) to save a configuration file. If you highlight the program/image when pressing F4, the default name will be the name of the program/image with a .cfg extension.

To automatically load the corresponding configuration file, go to File Browser and Menu settings and set “Browser .cfg auto load” to “on”. For this to work the name of the configuration file must be equal to the name of the related image/program, with .cfg extension.

For Example:

superglazz.d64

superglazz.cfg

### 5.5.1. Format details

Note that usually .cfg files should be created by using the User / File menu, editing them manually is not recommended.

The general format is similar to .ini files, every line starts with a string indicating the option, followed by an equal sign and the value of the respective option. Eg:

string=value

The actual strings and valid values can be derived from the following table. Please note that all values, including the numeric hex values, **must** be written **exactly** like in the table:

| name       | value   |                 |
|------------|---|-----------------|
| reusize    | RAM Size of the REU expansion in KiB            |                 |
|            | 0x00000000                                      | none/disabled   |
|            | 0x00000080                                      | 128 KiB         |
|            | 0x00000100                                      | 256 KiB         |
|            | 0x00000200                                      | 512 KiB         |
|            | 0x00000400                                      | 1 MiB           |
|            | 0x00000800                                      | 2 MiB           |
|            | 0x00001000                                      | 4 MiB           |
|            | 0x00002000                                      | 8 MiB           |
|            | 0x00004000                                      | 16 MiB          |
| georamsize | RAM Size of the GEORAM expansion in KiB         |                 |
|            | 0x00000000                                      | none/disabled   |
|            | 0x00000200                                      | 512 KiB         |
|            | 0x00000400                                      | 1 MiB           |
|            | 0x00000800                                      | 2 MiB           |
|            | 0x00001000                                      | 4 MiB           |
| stereosid  | I/O Address used for the 2 <sup>nd</sup> SID    |                 |
|            | 0x00000000                                      | None (disabled) |
|            | 0x0000d420                                      | 0xd420          |
|            | 0x0000d500                                      | 0xd500          |
|            | 0x0000d700                                      | 0xd700          |
|            | 0x0000de00                                      | 0xde00          |
|            | 0x0000df00                                      | 0xdf00          |
| sid1type   | 1 <sup>st</sup> emulated SID type               |                 |
|            | 6581  | 6581            |
|            | 8580  | 8580            |
| sid2type   | 2 <sup>nd</sup> emulated SID type, see sid1type |                 |

## Overview

| name       | value  |          |
|------------|--|----------|
| mouseemu   | Enable or disable mouse emulation via PS/2       |          |
|            | auto   |          |
|            | off  |          |
| mouseport  | Joystick port the emulated mouse is connected to |          |
|            | 0x00000001                                       | Port #1  |
|            | 0x00000002                                       | Port #2  |
| turbo      | Is the turbo mode enabled?                       |          |
|            | off  |          |
|            | on   |          |
| turbolimit | Maximum turbo speed                              |          |
|            | off  | No limit |
|            | 200%   |          |
|            | 300%   |          |
|            | 400%   |          |
|            | 500%   |          |
|            | 600%   |          |
|            | 75%  |          |
|            | 50%  |          |
|            | 25%  |          |
| turbod030  | Is the turbo sensitive to \$d030?                |          |
|            | off  |          |
|            | on   |          |
| turboiec   | Is the turbo sensitive to IEC bus activity?      |          |
|            | on   |          |
|            | off  |          |



| name      | value                 |  |
|-----------|-----------------------|--|
| clockport | Clockport I/O mapping |  |
|           | off                   |  |
|           | auto                  |  |
|           | rr                    |  |
|           | mmc64                 |  |

### 5.5.2. Example

The default configuration saved into a .cfg file would look like this:

```
reusize=0x00000000
georamsize=0x00000000
stereosid=0x00000000
sid1type=6581
sid2type=6581
mouseemu=off
mouseport=0x00000001
turbo=off
turbolimit=off
turbod030=off
turboiec=on
clockport=auto
```

## 6. Chameleon Menu System

### 6.1. Boot Loader

The boot loader is executed once at start-up, it's job is to do some basic initialization:

- load the configuration data from flash
- set up default MMU mapping

If "copy ROMs from C64" is enabled:

- copy C64 kernal- and basic-ROM to Chameleon RAM

If "UPDATE . PRG" is present:

- load and execute "UPDATE . PRG" (Chameleon updater)

If "boot from SD-card" is enabled:

- load a custom menu binary "MENU . BIN" (if present)
- load a REU image "BOOT . REU" (if present)
- load a GEORAM image "BOOT . GE0" (if present)
- load a custom boot program "BOOT . PRG" (if present)
- load custom ROMs from "/CHAM64" directory (if present):
  - "CHAM64/BASIC . ROM" - C64 basic ROM
  - "CHAM64/KERNAL . ROM" - C64 kernal ROM
  - "CHAM64/CHARGEN . ROM" - C64 character ROM
  - "CHAM64/MMC64 . ROM" - MMC64 BIOS ROM
  - "CHAM64/SLOT1 . ROM" - Cartridge slot 1 ROM (\*1)
  - "CHAM64/SLOT2 . ROM" - Cartridge slot 2 ROM (\*1)
  - "CHAM64/SLOT3 . ROM" - Cartridge slot 3 ROM (\*1)
  - "CHAM64/SLOT4 . ROM" - Cartridge slot 4 ROM (\*1)
  - "CHAM64/DRIVE1 . ROM" - 1541 drive 1 ROM
  - "CHAM64/DRIVE2 . ROM" - 1541 drive 2 ROM

and finally:

- if a boot program was loaded, execute it. If not, execute the menu system.

\*1) Cartridge ROMs must be plain binaries (without load address), NOT .crt files. Files in proper format can be created by the ROM saver (see chapter 6.5.2 User / File menu ). Cartridges which are bigger than 64k can only be used in slot 3 and 4, see chapter 13.6 supported Cartridge types for details. To make cartridges readily available for use after loading, the respective cartridge type must be configured in the settings (see chapter 6.4 Options).

### 6.1.1. Hotkeys during boot process

At start-up the boot loader will check the following "magic keys":

| key              | remote |   |
|------------------|--------|---|
| Space            | 0      | Presse and hold down when loading core to see config-ROM messages (pause before starting the boot-loader). This shows the FPGA core version and current mode. |
| ← (Arrow left)   | ESCAPE | Hold down when loading core to skip boot loader (see NOTE)  |
| 1                |        | Hold to force default configuration.  |
| 2                |        | Hold to force fail safe configuration   |
| 3                |        | Hold to skip menu   |
| 4                |        | Hold to force fail safe configuration with 640x480 VGA mode.  |
| <i>Any other</i> |        | Hold to see boot-loader messages (pause before starting the menu application)   |

#### **NOTE**

When skipping the boot loader, a hard-coded default setup will be used (MMC64 + PAL Retro Replay) which is meant for testing and to eventually recover from a broken boot loader and/or menu system.

If you press the menu button to enter the menu after skipping the boot-loader, the result will be that the boot loader will run and initialize the Chameleon the same way as if was not skipped - meaning various settings may change or even a boot program may get executed.

Bottom line: Usually the boot loader should not be skipped. If you don't want the Chameleon to start up with the menu system, use the "skip menu at boot" option (see chapter 6.4 Options). This will then result in the Chameleon starting up using the settings used in the options menu, and further freezing/ restarting will work as expected.

### 6.1.2. Limitations of the boot-loader

- MMC, SD, SDHC cards supported (max. medium size: 32GB)
- File-system must be FAT12, FAT16 or FAT32
- no long file-names

The limits of the boot-loader do not apply to the file-browser, see chapter 6.5.6 Limitations of the file-browser below.

### 6.1.3. Example: boot.prg

A possible usecase for a boot.prg that starts automatically at powerup is starting a different core directly – for this a small BASIC program can be used:

```
10 slot = 6:rem index of slot to start
15 poke 53502,42:rem enable cfg mode
20 poke 53489,1:rem enable mmc64
25 poke 57105,5:rem mmc64 8mhz spi, write trigger, deselect
30 poke 57105,7:rem mmc64 8mhz spi, write trigger, select
35 poke 57104,64+12:rem cmd12
40 poke 57104,0
45 poke 57104,0
50 poke 57104,0
55 poke 57104,0
60 poke 57104,97: rem crc
65 poke 57105,5:rem mmc64 8mhz spi, write trigger, deselect
70 poke 57105,0:rem mmc64 off
75 poke 53502,16+slot:rem start core
```

## 6.2. Keys

The menu system can be controlled by the following keys / buttons:

| Key                         | PS/2 Key             | Remote    | Joy #2     | Function                   |
|-----------------------------|----------------------|-----------|------------|----------------------------|
| <- (arrow left)<br>RUN/STOP | ESCAPE               | ESCAPE    |            | exit menu/one level up     |
| Cursor<br>up/down           | Cursor<br>up/down    | PLAY/STOP | up/down    | navigate up/down           |
| Cursor<br>left/right        | Cursor<br>left/right | REW/FWD   | left/right | enter sub menu/toggle item |
| RETURN                      | Enter                | ENTER     | Fire       | enter sub menu/toggle item |
| F1                          | Page up              | 1         |            | Page up                    |
| F2                          | F2                   | 2         |            | Go to first menu item      |
| F7                          | Page<br>down         | 7         |            | Page down                  |
| F8                          | F8                   | 8         |            | Go to last menu item       |

### 6.3. Main Menu

This is the main user interface and provides access to all features of the Chameleon core.



| Hotkey         | Menu Option            | Function  |
|----------------|------------------------|---|
| F1             | Clear C64 Memory       | fill C64 memory with zeros and reset                        |
| F2             | Clear ALL Memory       | fill C64 memory, reu, georam with 0 and reset               |
| F3             | Reset to Basic         | reset with all cartridges and expansions disabled           |
| F5             | File Browser           | enter file browser (see chapter 6.5 File Browser)           |
| F6             | Monitor                | enter machine language monitor (see chapter 6.6 ML Monitor) |
| F7             | Reset with Cartridge   | reset with selected default cartridge enabled               |
| F8             | Launch Core            | start another FPGA core                                     |
| ← (Arrow left) | restart C64 (unfreeze) | restart currently running program (*1)                      |
|                | Options                | enter options menu  |
|                | Online Help            | Enter help system (see chapter 6.7 Online Help)             |
|                | System Info            | show some general info                                      |

---

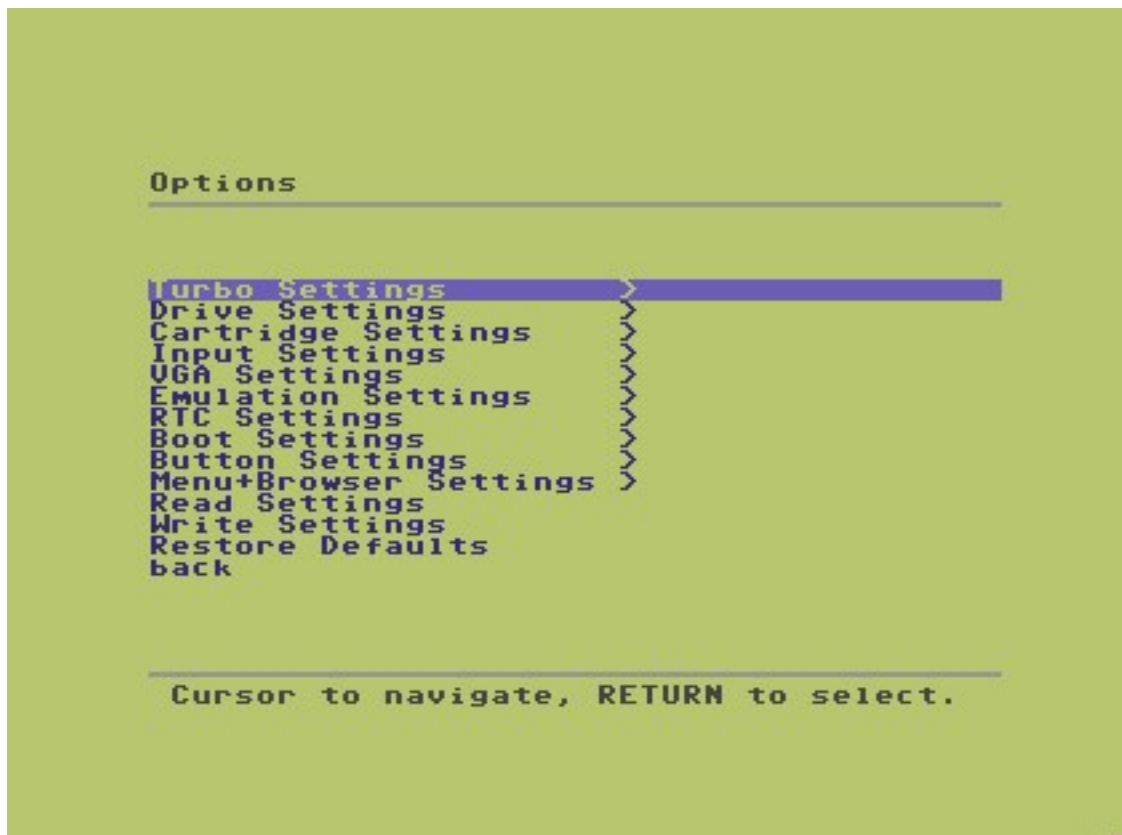
## Chameleon Menu System

---

| Hotkey | Menu Option | Function                          |
|--------|-------------|-----------------------------------|
|        |             |                                   |
| 1..4   |             | start with cartridge in slot 1..4 |
| A..P   |             | start FPGA core slot 1-16         |

\*1) Only if the menu was entered by pressing the menu button ("freeze"). Not all options can/will be changed when restarting, see chapter 6.4 Options NOTE.

## 6.4. Options



### 6.4.1. CPU Turbo

| Option              |  |
|---------------------|--|
| Turbo Mode          | Enable turbo mode.   |
| Turbo Speed Limit   | Set maximum speed if turbo is enabled.   |
| Turbo d030 bit      | Enable emulation of the VIC-IIe (C128) 2Mhz bit.   |
| Turbo IEC sensitive | When enabled, the turbo will automatically slow down on IEC accesses. It is recommended to always have this enabled. |

Only enable the turbo mode when you really need it – many programs contain timing critical parts that will only work correctly at the original speed.



### 6.4.2. Emulated drives

| Option                |  |
|-----------------------|--|
| Emu Drive 1 Device ID | Set device number for first emulated drive. This option can not be changed while a disk image is mounted.  |
| Drive 1 write protect | Enable write protection for first emulated drive.  |
| Emu Drive 2 Device ID | Set device number for second emulated drive. This option can not be changed while a disk image is mounted.   |
| Drive 2 write protect | Enable write protection for second emulated drive.   |
| Update mounted images | Set whether the menu should ask to update mounted image, or never or always update them automatically.   |
| IEC bus connection    | Choose whether the IEC bus of the emulated drive is connected internally or only to the breakout cable (use this if you want to use the Chameleon as a standalone drive emulator). |
| Real C64 IEC bus      | Enable / disable the C64's IEC bus. SX-64 users may use this to "disconnect" the internal drive.   |
| Reset emulated Drives | Send a reset to all emulated IEC devices.  |

Please keep in mind that many programs only work correctly when started from drive 8, and many (fast)loaders will only work with just one drive on the bus – so unless you know you will need a second drive, always leave it disabled. You should also not leave any real drives on the bus when they are powered off – disconnect them instead.

### 6.4.3. Cartridges and Memory Expansions

| Option                |  |
|-----------------------|--|
| MMC64                 | Enable MMC64 emulation.  |
| default Cartridge     | Set default cartridge slot used by "reset with cartridge" and with "browser exec setup" set to "custom"  |
| Cartridge Type Slot 1 | Cartridge type in slot 1. (see chapter 13.6 supported Cartridge types)   |
| Cartridge Type Slot 2 | Cartridge type in slot 2. (see chapter 13.6 supported Cartridge types)   |
| Cartridge Type Slot 3 | Cartridge type in slot 3. (see chapter 13.6 supported Cartridge types)   |
| Cartridge Type Slot 4 | Cartridge type in slot 4. (see chapter 13.6 supported Cartridge types)   |
| Cartridge mount slot  | Set whether the menu should ask for the cartridge slot to use or to always use slot 4 when loading a cartridge image file in the file browser.   |
| Clock Port            | Select clock port mapping. If "auto" is selected, then the clock-port will be mapped to de02 if RR is active, to df20 if MMC64 is active and RR is not, and disabled in all other cases. |
| RR-NET MAC Address    | Set default MAC Address for RR-NET.  |
| REU Size              | Enable REU and set emulated size.  |
| GeoRAM Size           | Enable GeoRAM and set emulated size.   |

These settings can not be changed if "browser exec setup" is set to "compatible" (because then the browser will disable them anyway). Also changes to these settings will not get applied when unfreezing (instead the respective registers will be restored by the freezer), so to apply them you will have to reset through the menu. See NOTE below.

Enable cartridges and/or memory expansions only when you really need them to avoid conflicts with the programs you want to run. Also keep in mind that not all emulated cartridges can work when one of the memory expansions is enabled – refer to the manual of the respective cartridge.

#### 6.4.4. Input devices

| Option          |  |
|-----------------|--|
| PS/2 Mouse      | Auto-detect or disable emulated mouse. |
| PS/2 Mouse port | Set port for emulated mouse.           |
| IR Receiver     | Toggle CDTV IR enable.                 |

Since some programs show erroneous behaviour when a mouse is connected, only activate it when you are actually using a program with mouse support.

#### 6.4.5. VGA output

| Option            |   |
|-------------------|---|
| VGA Mode          | Set display mode/resolution used for VGA output.  |
| VGA Buffers       | Selects number of VGA frame-buffers (Single, Double, Triple). When triple buffering is enabled, the current frame is blended with the previous frame to reduce frame-rate conversion artefacts (jerky scrolling, strange looking interlace pictures). |
| VGA Filter        | Selects VGA scaling filter (nearest neighbour, Scale2x, Alien)  |
| VGA Scanlines     | Select the amount of scanline shading to simulate the look of a CRT (off, 75% brightness, 50% brightness, 25% brightness)   |
| VGA VIC-II sync   | When enabled, the VGA output is synced to the VIC-II. When used with 50 or 60 Hz modes this enables perfect scrolling and interlace.  |
| VGA Mode Test     | Test selected mode and options. Press RETURN once to switch to selected mode, then RETURN again to confirm or any other to switch back to previous settings.  |
| VGA Debug Overlay | Toggle VGA debug overlay.   |

Note: due to the varying capabilities of VGA monitors it is hard to give general advice on what the “best” settings are. We recommend doing the following:

- start with all settings at their respective default (Mode: [800x600@72Hz](#), Buffers: Single, Filter: nearest neighbour, VIC-II sync: off)
- now try finding a 50Hz (PAL) or 60Hz (NTSC) mode that your monitor is capable of displaying.
  - If you found one, try enabling VIC-II sync
- if you can not use a 50 or 60 Hz mode, choose one of the available standard modes
  - choose suitable VGA Buffer setting (Double buffering removes tearing, Triple buffering adds frame blending)

### 6.4.6. Emulation settings

| Option               |   |
|----------------------|---|
| VIC-II type          | Choose VIC-II type. This setting has no effect in cartridge mode.   |
| VIC-II VGA Output    | Enable VIC-II output to VGA. Disabling this will allow to use the frame-buffer from within a running program. |
| SID Emulation        | Select type and mapping of emulated SID(s)  |
| SID Real Stereo Chip | Select type and mapping of real SID(s). This setting has no effect in standalone mode. (*)                    |
| SID Type (first)     | Toggle between 6581 (old) and 8580 (new) for first SID  |
| SID Type (second)    | Toggle between 6581 (old) and 8580 (new) for second SID   |
| CIA Type             | Toggle between 6526 (old) and 8521 (new) for both CIAs  |

(\*) The chameleon core always re-maps all SID I/O. That means accesses are always intercepted at the address configured for “SID Emulation”, and then – in cartridge mode – always routed to the addresses configured as “SID Real Stereo Chip”. That way you can use the real SIDs at a different address than what they are mapped to by the hardware in the C64, and change that address on the fly in the chameleon menu.

### 6.4.7. RTC

| Option      |  |
|-------------|--|
| RTC Time    | Set current Time (Hours:Minutes:Seconds).                                    |
| RTC Date    | Set current Date (Day:Month:Year).   |
| RTC Weekday | Day of the week. This item is informal only and can not be changed directly. |

### 6.4.8. Boot-loader settings

| Option             |  |
|--------------------|--|
| boot from SD card  | Enable loading various files at boot time (see chapter 6.1 Boot Loader). |
| boot skips menu    | If enabled the C64 starts with default cartridge.                        |
| copy ROMs from C64 | Copy basic- and kernal- ROM from C64 at boot time.                       |

### 6.4.9. Button settings

| Option             |  |
|--------------------|--|
| Left Button        | Select function for the leftmost button (see chapter 6.4.12 Buttons below).    |
| Reset goes to menu | If enabled, pressing reset goes to menu instead of resetting the emulated C64. |

### 6.4.10. File Browser and Menu settings

| Option                 |  |
|------------------------|--|
| Browser Exec Setup     | Execution setup that applies when starting programs and/ or cartridges via the file browser. "custom" will use the setup from config menu, "compatible" will disable most things and configure a bare stock C64. |
| Browser auto sort      | When enabled, always sort SD cards directories.  |
| Browser start-up       | Browser may start in 1 or 2 panel mode.  |
| Browser charset        | The character set used by default.   |
| Browser .cfg auto load | Enable automatic loading of options when mounting/loading Images.  |
| Menu colour scheme     | Select menu colour scheme.   |
| Menu input scheme      | Select input scheme (*)  |

(\*) When “CDTV remote” is selected as input scheme that has the following consequences:

- Some features will get deactivated because they are not usable without a keyboard and could make you end up in a situation where you can not go back into the main menu without a keyboard. (eg ML Monitor in the main menu)
- The “Mount” and “Unmount” features are called by F5 and F6

### 6.4.11. Load and Save the settings

| Option           |  |
|------------------|--|
| Read Settings    | Read configuration from flash.                     |
| Write Settings   | Write configuration to flash to make it permanent. |
| restore Defaults | Restore default configuration.                     |
| back             | Go back to main menu.                              |

## 6.4.12. Buttons

### ***Left***

The left button is the multi-purpose button of the Chameleon core and can be configured to be used as the following:

| Option       | Function   |
|--------------|--|
| Menu         | Start the menu system  |
| File-browser | Start the menu system and directly go to the File-browser  |
| Monitor      | Start the menu system and directly go to the ML Monitor  |
| Cartridge    | Short: cartridge on/off<br>Long: Emulated cartridge button (see chapter 5.3 Using cartridges)                          |
| Turbo        | Enable / Disable the CPU turbo   |
| Disk Drive 1 | Short button press: Cycle through mounted disk images on drive 1<br>Long button press: select first mounted disk image |
| Disk Drive 2 | Cycle through mounted disk images on drive 2<br>Long button press: select first mounted disk image                     |

### ***Middle***

The middle button is generally used as the “freezer” button. A “short” button press (shorter than 0.7 seconds) will activate the freezer of an activated cartridge (such as retro replay or the final cartridge 3). A “long” button press (longer than 0.7 seconds) will activate the menu freezer and go to the main menu of the menu system.

### ***Right***

The right button is the “reset” button. Generally a “short” button press (shorter than 0,7 seconds) will reset the C64, and a “long” button press (longer than 0,7 seconds) will reload the Chameleon core (equalling a “hard” reset, or power-cycle).

If “reset goes to menu” is enabled, then the menu system will be started on every reset (instead of resetting the C64).

When reset is pressed (shorter than 0,7 seconds) while the file browser is active (only) the emulated drives will be reset.

In some file browser plugins (eg the SID-player) pressing reset (shorter than 0,7 seconds) brings you back zu the file browser.

### **NOTE**

All options will generally, and always, get applied when leaving the menu. However there are some exceptions that apply when the menu was entered by "freezing" using the menu button and then "unfreezing" the running C64. This is needed so for example Chameleon-aware programs can also be interrupted.

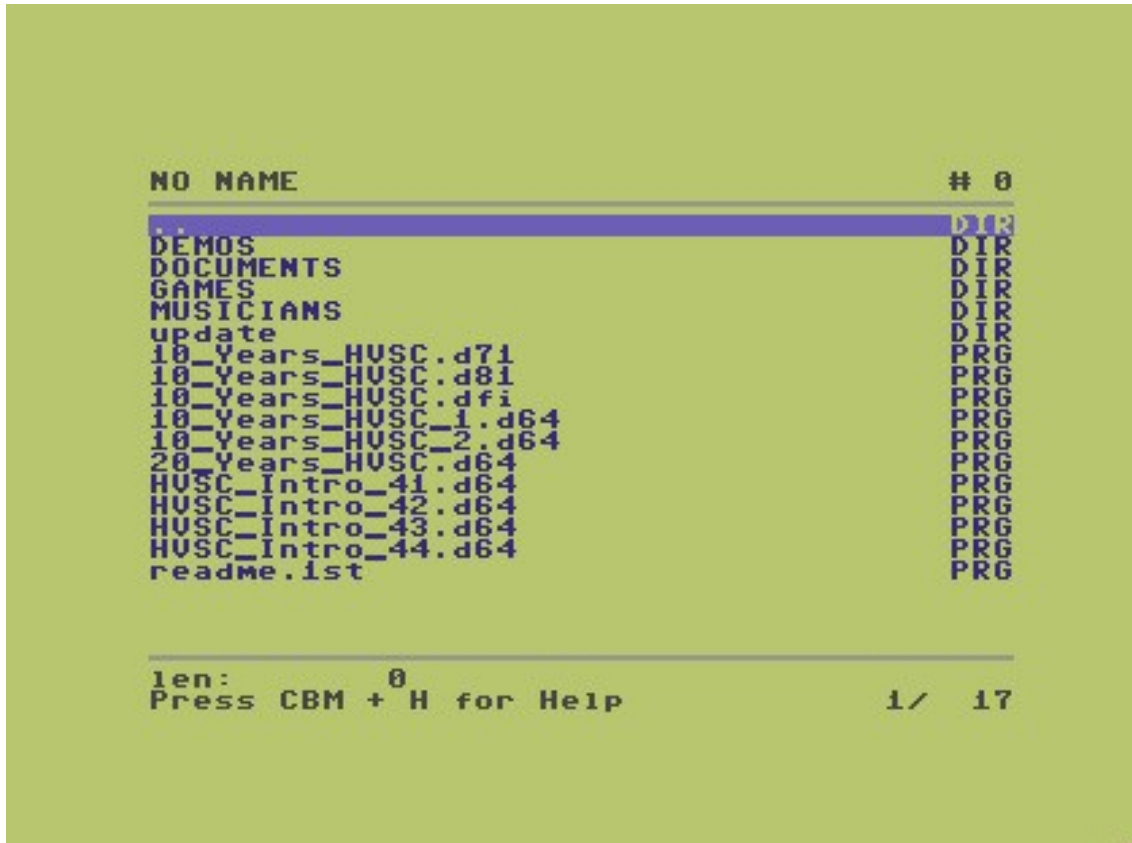
- Cartridge settings have no effect when unfreezing (would almost certainly result in a crash anyway, and this way interrupting a game cartridge is possible). You will have to reset through the menu to apply new cartridges settings.
- Changing GEORAM or REU settings will have no effect when unfreezing. (again, this would almost certainly result in a crash anyway).
- VIC-II settings have no effect when unfreezing. (this allows to interrupt Chameleon-aware programs that change them)
- Turbo settings will be copied from the register(s) to the options when entering the menu by freezing. (so they can be changed after freezing, and restarting does not alter them otherwise)

A Chameleon-aware program should generally not touch any of the above settings (or their respective registers), with the following exceptions (which are handled by the freezer and not altered when unfreezing):

- enabling config mode
- Register settings (CFGREG)
- Turbo settings (CFGTUR)
- Cartridge settings (CFGCRT)
- GEORAM and REU settings (CFGREU)
- VIC-II extension settings (upper 4 bits of CFGVIC)
- MMU settings (all MMU registers)
- all Cartridge registers, with the following precautions:
  - freezing during an ongoing SPI transfer (RTC, flash, SD-card) is impossible to handle, for this reason the result is completely undefined.
  - freezing while RR-NET is in use, and then entering a menu option that uses it may result in undefined behaviour when restarting.
  - freezing during a REU transfer may be problematic, although the menu itself does not use the REU registers.

## 6.5. File Browser

Press F5 in the main menu to enter the file browser:



### 6.5.1. Keys

| C64 Key                      | PS/2 Key          | Remote        | Joy #2         |   |
|------------------------------|-------------------|---------------|----------------|---|
| ← (Arrow left)<br>"RUN/STOP" | ESCAPE            | ESCAPE        |                | exit file browser   |
| Cursor<br>up/down            | Cursor up/down    | PLAY/<br>STOP | Up/<br>Down    | navigate up/down  |
| Cursor<br>left/right         | Cursor left/right | REW/<br>FWD   | Left/<br>Right | choose left or right tab  |
| F1 / F7                      | Page up/down      | 1 / 7         |                | page up / page down   |
| F2 / F8                      | F2 / F8           | 2 / 8         |                | start / end of list   |
| A - Z                        | A - Z             |               |                | Quickfind – type filename to jump to with<br>max 500ms delay between keypresses |
| RETURN                       | Enter             | ENTER         | Fire           | enter directory, open file  |
| F3                           | F3                | 3             |                | view file   |
| F4                           | F4                | 4             |                | user menu (see chapter below)   |

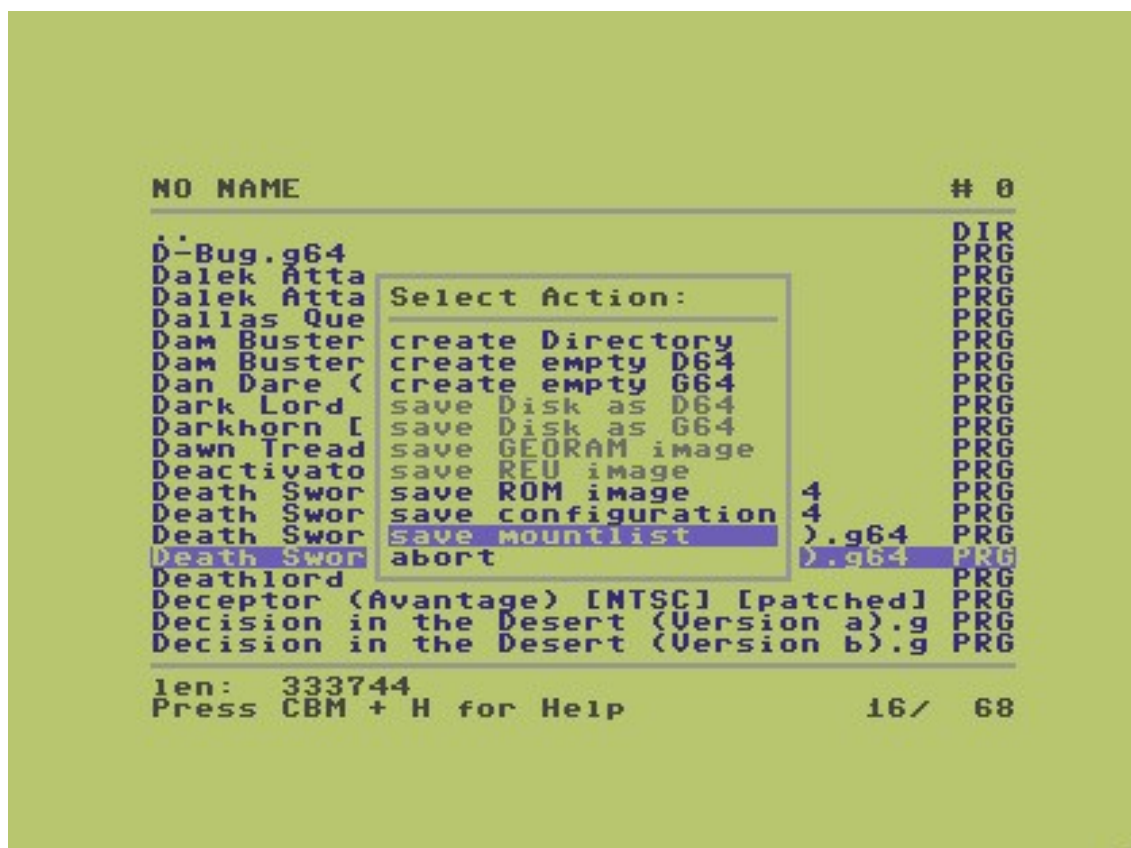


## Chameleon Menu System

| <b>C64 Key</b> | <b>PS/2 Key</b>         | <b>Remote</b>   | <b>Joy #2</b> |   |
|----------------|-------------------------|-----------------|---------------|---|
| F5             | F5                      | 5               |               | Copy file   |
| F6             | F6                      | 6               |               | copy image  |
| £ (Pound)      | F9                      |                 |               | select SD-card (device 0)   |
| 1..7, 8..0     | 1..7, 8..0              |                 |               | select IEC device 11-17,8-10  |
| + / -          | F10 / “-” (US) “ß” (DE) | VOL UP/<br>DOWN |               | choose device number  |
| @              | “[“ (US) “Û” (DE)       |                 |               | send DOS command (to IEC drive)   |
| CBM + H        | ALT + H                 |                 |               | show quick reference / help   |
| CBM + P        | ALT + P                 |                 |               | toggle panel size   |
| CBM + R        | ALT + R                 |                 |               | (re)read directory of active panel  |
| CBM + S        | ALT + S                 |                 |               | sort directory of active panel  |
| CBM + M        | ALT + M                 |                 |               | mount image file  |
| CBM + U        | ALT + U                 |                 |               | Unmount images  |
| CBM + L        | ALT + L                 |                 |               | load file   |
| CBM + E        | ALT + E                 |                 |               | load and execute file. note that this will NOT mount the image it is used in and just load the selected file. |
| CBM + D        | ALT + D                 |                 |               | delete file or directory (must be empty)  |
| CBM + N        | ALT + N                 |                 |               | rename file   |
| CBM + X        | ALT + X                 |                 |               | View file as hex dump   |
| CBM + T        | ALT + T                 |                 |               | View file as text   |
| space          |                         | 0               |               | File action menu (see below)  |
| ↑ (Arrow up)   | “\” (US) “#” (DE)       |                 |               | toggle upper-/lower-case display  |

### 6.5.2. User / File menu

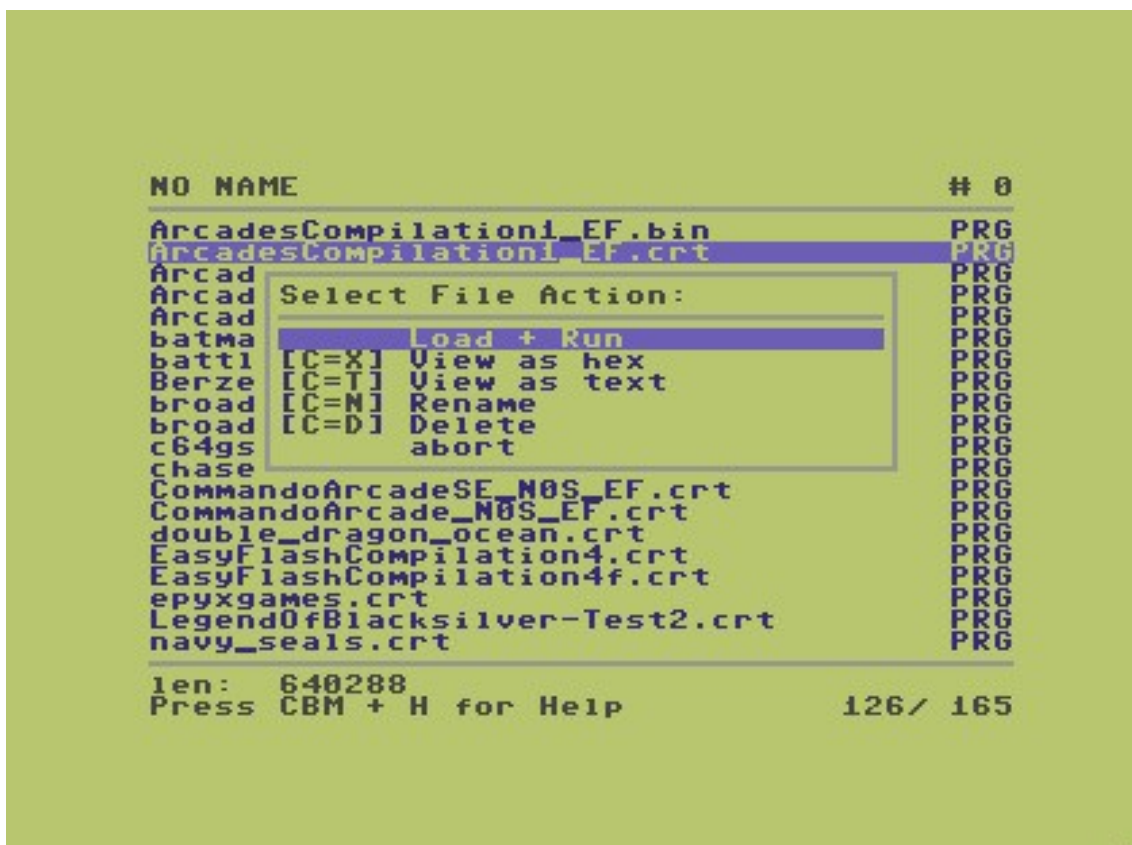
The user/file menu is brought up by pressing F4 any time in the file browser: (most features are only available on SD card, creating directories only works on CMD compatible IEC devices)



| Menu option        | Function   |
|--------------------|--|
| create Directory   | creates a new directory                                |
| create empty D64   | creates an empty D64 file                              |
| create empty G64   | creates an empty G64 file (*1)                         |
| save Disk as D64   | save contents of a mounted disk as D64 image (*2)      |
| save Disk as G64   | save contents of a mounted disk as G64 image (*3)      |
| save GEORAM image  | save current GEORAM content to a file (*4)             |
| save REU image     | save current REU content to a file (*4)                |
| save ROM image     | save content of various ROMs to a file                 |
| save configuration | writes configuration options into a .cfg file          |
| save mountlist     | creates a .lst file from currently mounted images (*5) |
| abort              | Abort menu   |

- \*1) Note that the created file will really be empty, as in a blank disk - it must be formatted before it can be used.
- \*2) Note that the limits of the D64 format apply. This item will not be available if no disk has been mounted.
- \*3) Note that current limits of the drive emulation apply. This item will not be available if no disk has been mounted.
- \*4) These items will not be available if GEORAM or REU are not enabled.
- \*5) This item is only available when two or more images are mounted. Please change the current directory to the one that contains those images before saving the file.

### 6.5.3. File-Action menu



This menu is “context sensitive” and shows possible operations depending on the currently selected file. It can be invoked at any time using **space**.

### 6.5.4. Supported File types

| Extension  | File type   |
|------------|---|
| .d64       | 1541 Disk image   |
| .t64       | C64S Tape image (read only)   |
| .g64       | 1541 GCR image  |
| .lst, .vfl | Swaplist file (see Mounting using swaplist files)   |
| .cfg       | Configfile  |
| .sid       | Playsid SID tune  |
| .mus, .str | Compute Gazette (stereo) Sidplayer tune   |
| .prg       | Plain program file (with load address)  |
| .p00       | Plain program file (PC64 format)  |
| .crt       | CCS64 Cartridge Image. Also see chapter 13.6 supported Cartridge types                                  |
| .reu       | REU image   |
| .geo       | GEORAM image  |
| .rom       | ROM images  |
| .koa       | Koala Paint image   |
| .rpm       | Runpaint image  |
| .cdu       | CDU Paint image   |
| .blp       | Blazing Paddles image   |
| .pmg       | Paint Magic image   |
| .art, .ocp | OCP Art Studio image (v1 and v2)  |
| .ddl       | Doodle image  |
| .hed       | Hi-Eddi image   |
| .hpi       | Hires Picture (same as Art Studio v1)   |
| .txt       | Ascii text file   |
| .wav       | Windows WAVE file. Supported formats are: 11khz mono, 11khz stereo, 22khz mono, 44khz mono, 48khz mono. |

### 6.5.5. Plugins

**default action: (RETURN)**

| Type       | Action   |
|------------|--|
| .d64       | enter image  |
| .t64       | enter image  |
| .g64       | mount G64 image and start with LOAD"*", 8, 1 (see chapter 5.2.2 Mounting images)                             |
| .lst, .vfl | Mount all images from the swaplist, start first image with LOAD"*", 8, 1 (see chapter 5.2.2 Mounting images) |
| .cfg       | load configuration   |
| .sid       | play sid file  |
| .mus, .str | Play sidplayer tune  |
| .wav       | Play .wav file.  |
| .prg       | load and execute. inside images also mount image.  |
| .p00       | load and execute   |
| .crt       | load and start cartridge image. (see chapter 13.6 supported Cartridge types)                                 |
| .reu       | load REU image   |
| .geo       | load GEORAM image  |
| .rom       | load ROM images  |
| .koa       | view Koala Paint image   |
| .rpm       | view Runpaint image  |
| .cdu       | view CDU Paint image   |
| .blp       | view Blazing Paddles image   |
| .pmg       | view Paint Magic image   |
| .art, .ocp | view OCP Art Studio image (v1 and v2)  |
| .ddl       | view Doodle image  |
| .hed       | view Hi-Eddi image   |
| .hpi       | view Hires Picture (same as Art Studio v1)   |
| .txt       | view text file   |
| other      | view hex file  |

**mount action: (CBM + M)**

| Type      | Action  |
|-----------|---|
| .d64      | mount disk image on emulated drive (see chapter 5.2.2 Mounting images)                    |
| .g64      | mount disk image on emulated drive (see chapter 5.2.2 Mounting images)                    |
| .lst, vfl | mount all disk images from swaplist on emulated drive (see chapter 5.2.2 Mounting images) |
| .rom      | load ROM images   |

**load action: (CBM + L)**

| Type      | Action  |
|-----------|---|
| .d64      | mount disk image on emulated drive (see chapter 5.2.2 Mounting images)                    |
| .g64      | mount disk image on emulated drive (see chapter 5.2.2 Mounting images)                    |
| .lst, vfl | mount all disk images from swaplist on emulated drive (see chapter 5.2.2 Mounting images) |
| .prg      | load prg file to C64 memory (*)   |
| .p00      | load prg file to C64 memory   |
| .reu      | load REU image  |
| .geo      | load GeoRAM image   |
| .rom      | load ROM images   |

\*) note that this will NOT mount the image it is located in and just load the selected file.

**execute action: (CBM + E)**

| Type      | Action  |
|-----------|---|
| .d64      | mount disk image on emulated drive and start with LOAD" ", 8, 1 (see chapter 5.2.2 Mounting images) |
| .g64      | mount disk image on emulated drive and start with LOAD" ", 8, 1 (see chapter 5.2.2 Mounting images) |
| .lst, vfl | mount all disk images from swaplist on emulated drive (see chapter 5.2.2 Mounting images)           |
| .prg      | load and run prg file (*)   |
| .p00      | load and run file.  |
| .crt      | load and start cartridge image. see chapter 13.6 supported Cartridge types                          |

\*) note that this will NOT mount the image it is located in and just load the selected file.

**view action: (F3)**

| Type       | Action   |
|------------|--|
| .koa       | view Koala Paint image   |
| .rpm       | view Runpaint image  |
| .cdu       | view CDU Paint image   |
| .blp       | view Blazing Paddles image   |
| .pmg       | view Paint Magic image   |
| .art, .ocp | view OCP Art Studio image (v1 and v2)  |
| .ddl       | view Doodle image  |
| .hed       | view Hi-Eddi image   |
| .hpi       | view Hires Picture (same as Art Studio v1)   |
| .txt       | view text file   |
| .sid       | play sid file  |
| .mus, .str | Play sidplayer tune  |
| .wav       | Play .wav file.  |
| other      | view file as hex dump. this is also the fallback used in some other cases when a certain action is not available. Use CBM+X and CBM+T to explicitly view a file as hex dump or text instead. |

**copy file: (F5)**

| Type | Action                     |
|------|----------------------------|
| any  | Copy file (to other panel) |

To copy a file, switch to two panel mode (CBM+P) and press F5 (on any file).

This currently works on SD card and IEC drives in both directions. Writing into g64 files or t64 and p00 containers is not supported.

**copy image: (F6)**

| Type | Action                                 |
|------|--|
| .d64 | write .d64 file to drive (other panel) |

To read an image from an IEC drive to a file, switch to two panel mode (CBM+P), then change to the drive panel and press F6 (on any file).

### **6.5.6. Limitations of the file-browser**

- MMC, SD, SDHC cards supported (max. medium size: 32GB)
- File-system must be FAT12, FAT16 or FAT32
- max. 1023 entries per directory
- max. 40 characters per file-name (including extension and the dot) are displayed
- max. 255 characters per file-name are handled before the name is replaced by its short 8.3 file-name.
- For speed reasons, file-types will only be determined by looking at the file extension (and not by reading their header)



## 6.6. ML Monitor

```

; -PC- ar xr yr sp 00 01 sr nv-bdizc
r 10e3 ff 00 05 f3 2f 37 a5 %10100101
d 1000 78
a 1000 78 sei
a 1001 a2 00 ldx #$00
a 1003 8a txa
a 1004 9d 00 04 sta $0400,x
a 1007 9d 00 d8 sta $d800,x
a 100a a9 a0 lda #$a0
a 100c 9d 00 05 sta $0500,x
a 100f 9d 00 06 sta $0600,x
a 1012 9d 00 07 sta $0700,x
a 1015 a9 00 lda #$00
a 1017 9d 00 d9 sta $d900,x
a 101a 9d 00 da sta $da00,x
a 101d 9d 00 db sta $db00,x
a 1020 e8 inx
a 1021 d0 e0 bne $1003
a 1023 ee 2b 10 inc $102b
a 1026 a2 00 ldx #$00
a 1028 8a txa
a 1029 18 clc
a 102a 69 01 adc #$01

```

### 6.6.1. Features

- illegal opcodes for assembling and disassembling
- The whole 32Mb memory can be accessed via bank switching.
- edit binary, character maps, sprite maps
- edit CPU registers and I/O registers to be set when you leave the monitor and then unfreeze

### 6.6.2. Number formats

By default all numbers are interpreted as hexadecimal, unless stated differently. A specific format can be specified using the prefixes:

| Prefix | Base |             |
|--------|------|-------------|
| &      | 10   | decimal     |
| \$     | 16   | hexadecimal |
| %      | 2    | binary      |

### 6.6.3. Converting values

To convert values just type them on a line and press return – it will then be shown in hexadecimal, binary and decimal notation.

### 6.6.4. Commands

**\***

Switches to bank 0 (C64 memory), and toggles between ROM and RAM mapping.

**@**

DOS wedge. A single @ shows the device number and status of the currently active device  
@\$ shows the directory of the active device. @<device> changes the active device, where 0 represents the SD-card. @<command> sends a regular CBM DOS command, where the SD-card device can be used with the following:

|    |                  |                                 |
|----|------------------|---------------------------------|
|    |                  |                                 |
| i  | initialize       | (re) initialize the SD-card     |
| cd | change directory | Change active working directory |
|    |                  |                                 |

**\$**

A single \$ shows the directory of the current device

**assemble** <addr>

enter assembly code

**bank** <addr>

set upper 9 bits of memory address.

**compare** <start> <limit> <dest>

Compare command, compare block from start to limit with block at dest

**disassemble** <addr>

show disassembly

**fill** <start> <limit> <byte1> [<byte2> ...]

Fill an address range with a byte pattern

**go** <addr>

Start program on given address. If parameter is omitted use freezer address.

**hunt** <start> <limit> <byte1> [<byte2> ...]

Search for byte(s) in an address range

**i** <addr>

show/edit memory as screencode

**io** <addr>

show/edit i/o area (this also includes the color RAM)

**j** <addr>

show/edit memory as PETSCII

**load** "<filename>" <addr>

load a PRG file to memory. If the address is omitted the load address found in the PRG file is used

**memory** <addr>

show/edit memory

**mchar** <addr>

Show/Edit chars

**msprite** <addr>

Show/Edit sprite

**registers**

Show/Edit CPU registers

**save** "<file name>" <start addr> <end addr+1> <new start>

Save a block of memory to a PRG file. If <new start> is given then this address will be used as load adress for the resulting PRG file.

**transfer** <start> <end> <dest>

Transfer command, copy block from start to end to dest

**x**

exit monitor (goes back to main menu)

### 6.6.5. Keyboard

Generally it is possible to edit and navigate like in the BASIC editor. Additionally there are some extra features:

| <b>Taste</b> | <b>Action</b>  |
|--------------|--|
| Cursor up    | At the top of the screen: scroll one line backwards  |
| Cursor down  | At the bottom of the screen: scroll one line forward |
| F1           | Move cursor to first line, scroll one page backwards |
| F7           | Move cursor to last line, scroll one page forward    |

For scrolling to work the output of a scrollable command must be present in the first or last line of the screen. Scrolling only works with a few commands: m, ms, i (forward and backwards) and d, mc (forwards only).

### 6.7. Online Help

To use the online help, please copy the CHAMHELP.HLP file into the CHAM64 directory of your SD card.

Please note that the help file is work in progress, it will get updated when time permits.

#### 6.7.1. Keys

The help system can be controlled by the following keys / buttons:

| <b>Key</b>                  | <b>PS/2 Key</b> | <b>Remote</b> | <b>Function</b>         |
|-----------------------------|-----------------|---------------|-------------------------|
| <- (arrow left)<br>RUN/STOP | ESCAPE          | ESCAPE        | exit to main menu       |
| RETURN                      | Enter           | ENTER         | Go to main screen       |
| F1                          | Page up         | 1             | Previous page           |
| F2                          | F2              | 2             | Go to last visited page |
| F7<br>SPACE                 | Page<br>down    | 7             | Next page               |

additionally, cross linked pages can be accessed by pressing the highlighted hotkeys.

## 6.8. System Info

The system info screen shows some basic information, such as the menu software version, the currently used FPGA slot, the RTC time and some details on the inserted SD card.



```
System Info
Menu built: Mar 10 2017      rev:997
Core built: Dec 16 2016      Beta-9g
Core Slot: 0      Mode: Cartridge
VIC-II Type: PAL  Sync Status: ok
Time: 17:34:17 Friday 10/ 3/2017
CP: ok  RR-Net MAC: 28:cd:4c:ff:fb:ff
Card inserted: yes  Write Protected: no
Card init: ok      Mode: SDHC
Card FS: FAT32      Cluster size: 4 kb
Free: 5248608 kb

(c)2011-2017 Individual Computers
http://wiki.icomp.de/wiki/Chameleon/

RETURN to exit.
```

## 7. ChaCo, the Chameleon Control Software

ChaCo is an application that uses WxWidgets for the GUI. This makes it look different from other (Windows) applications, but gives us the chance to compile it for Linux or for the Mac.

### 7.1. *Preparations*

Please notice that due to the way ChaCo accesses the USB port, it may need to run as administrator (windows) or root (linux) respectively.

#### 7.1.1. Linux / macOS

Because the executable bit for the binaries are not preserved in the release archive, you will have to set them manually like this:

```
$ sudo chmod +x ./Chaco ./chacocmd
```

If you want to be able to run Chaco or chacocmd as a regular user without using sudo and typing your password all the time, you can change it to “suid root” like this:

```
$ sudo chown root:root ./Chaco ./chacocmd
```

```
$ sudo chmod u+s ./Chaco ./chacocmd
```

### 7.2. *Linux notes*

We currently provide a dynamically linked 64 bit linux binary with the following dependencies:

- libc 2.6
- libusb v1.x
- wxwidgets 2.8 (and related render engine, eg GTK). Only needed for the GUI.

The source code for ChaCo, the flasher program, and many other related tools is available at our public SVN repository at <https://svn.icomp.de/svn/chaco/> so you can build it on your system.

### 7.3. *macOS notes*

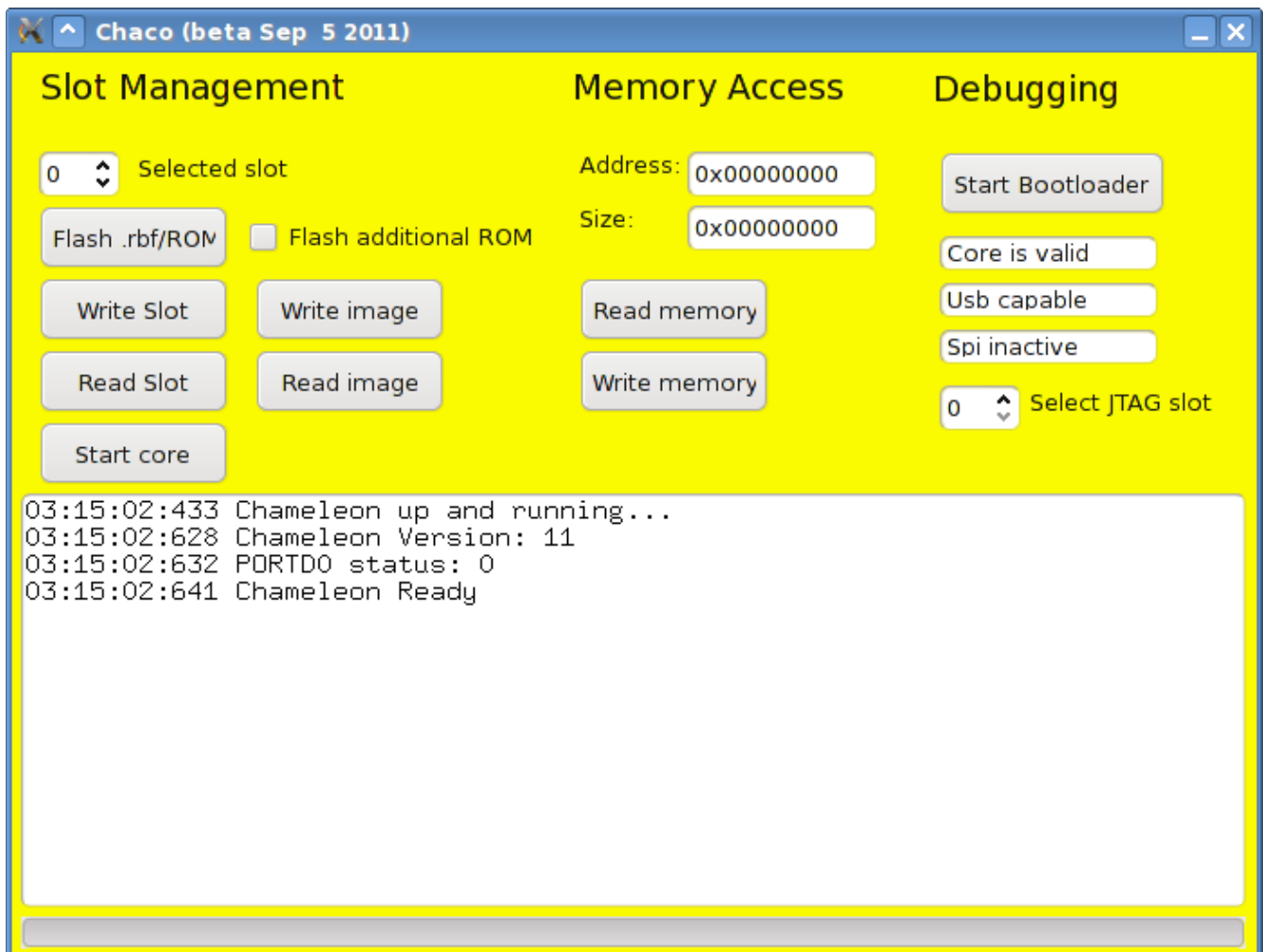
Unfortunately we can no more provide binaries for macOS anymore.

The source code for ChaCo, the flasher program, and many other related tools is available at our public SVN repository at <https://svn.icomp.de/svn/chaco/> so you can build it on your system.

For updating please use “update.prg” as described in 3.1 - Updating using update.prg

## 7.4. Overview

With a Chameleon connected, ChaCo will open with this screen:



### 7.4.1. Slot Management

**Selected slot:** Enter the number of the slot that you want to flash /read/write/start into this field.

**Flash .rbf/ROM:** This button starts the flash procedure of writing a core to a selected slot. (see chapter 7.5.2 Flashing a Core)

**Flash additional ROM: Tick this box if you want to add a rom file to the core file.** Some cores may not need a rom file, some other cores will *require* a rom file to work properly. Please read the documentation of the core you're about to flash. Remember that this box must be ticked *before* clicking the “Flash .rbf/ROM” button.

“Read Slot” and “Write Slot” will let you backup/restore one flash slot into or from a single file.

The buttons “Read image” and “Write image” will let you backup/restore the whole 16MByte flash into or from a single file.

**Start Core:** Click this button to launch the selected slot.

### 7.4.2. Memory Access

**Address:** address of memory block to read/write

**Size:** size of memory block

**Read memory, Write Memory:** These will let you access the memory of the running FPGA core (see chapter 7.5.3 Accessing memory).

### 7.4.3. Debugging

**Start Boot-loader:** Start boot-loader of the USB micro-controller to prepare it for reprogramming. WARNING: you usually will not need to do this. The Chameleon will have to be power cycled to resume normal operation.

**Select JTAG slot:** This is a setting for FPGA developers. If you launch the FPGA through the JTAG interface, the micro-controller will report this number to the core. It will behave as if the FPGA was launched from this slot.

### 7.4.4. Status display

The three fields “Core is valid”, “USB capable”, “SPI inactive” are status fields. You cannot enter anything here.

“Core is valid” is shown if the Chameleon is currently running a valid FPGA core. “Core is invalid” indicates invalid core data. (see chapter 7.5.1 Starting a core below)

“USB capable” means that the running core supports memory access using ChaCo (see chapter 7.5.3 Accessing memory below)

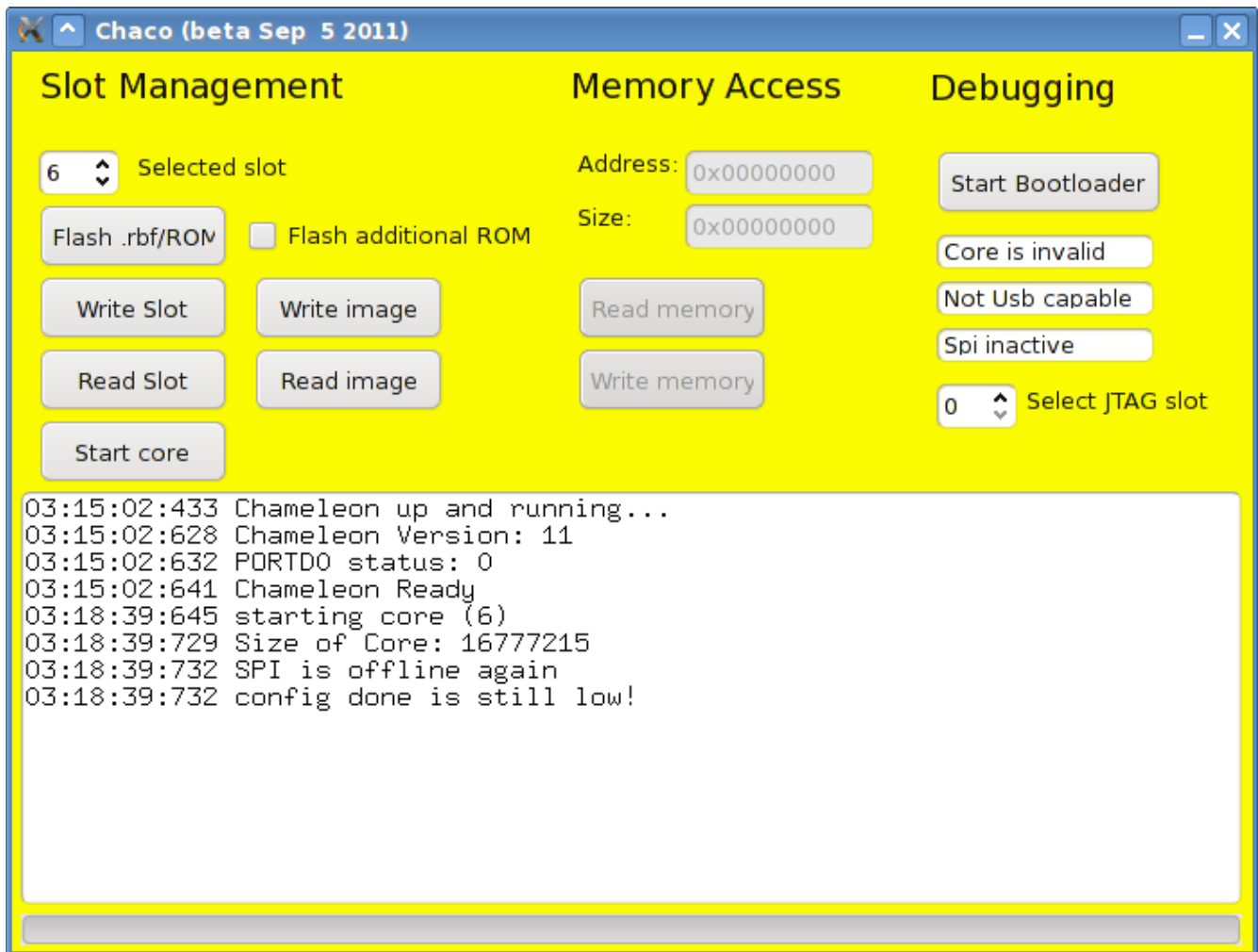
SPI active / SPI inactive shows whether the SPI bus is/was active. This is only useful for debugging ChaCo itself and will be removed later, it will show “inactive” most of the time.



## 7.5. Examples

### 7.5.1. Starting a core

Let's cause an error, so you can see the LED blink: Set “selected slot” to 6 and click on the “Start Core” button.



The error message in the window reflects what you should see on Chameleon: The red LED should blink seven times, then pause, then start over with blinking.

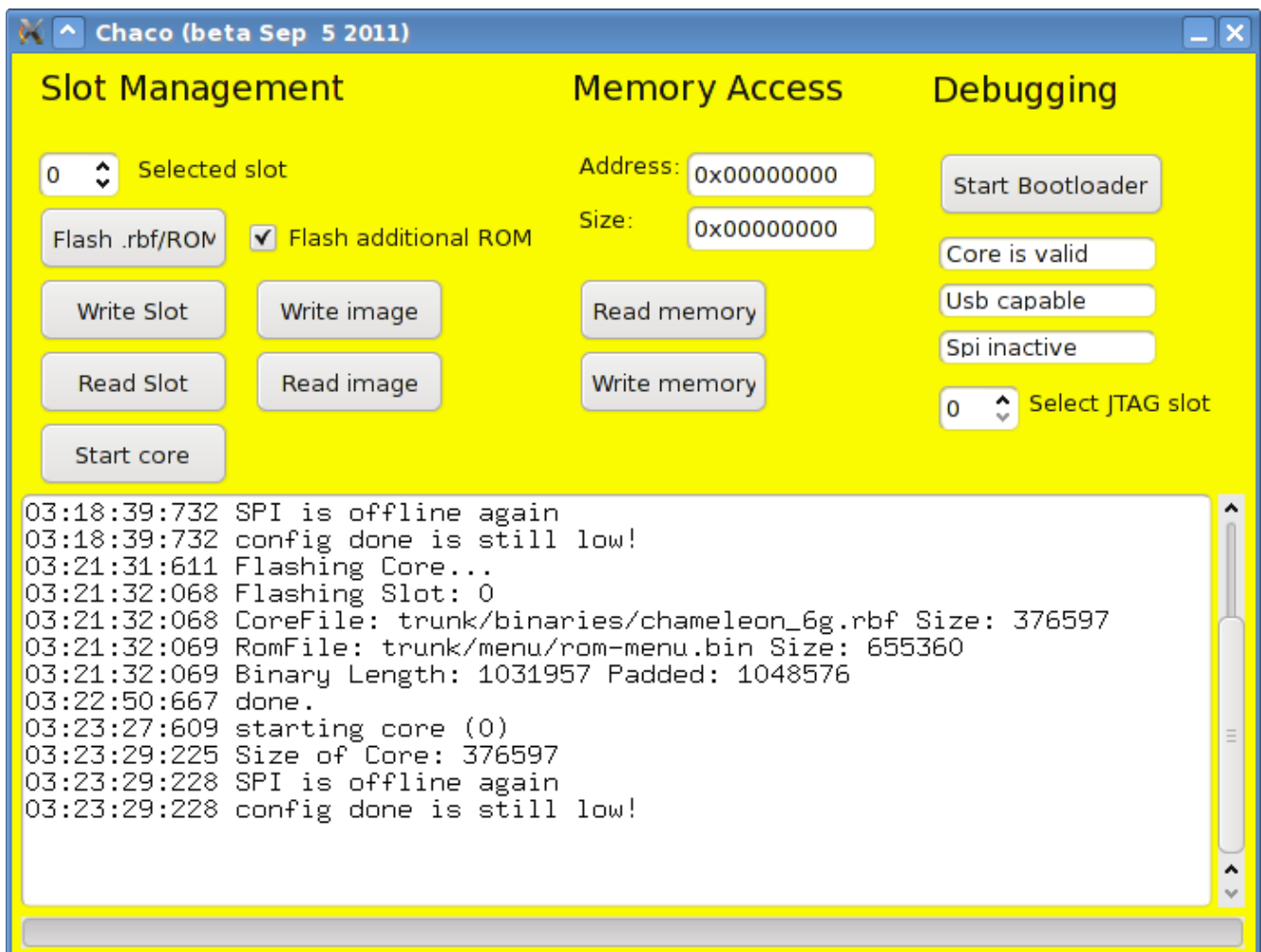
Now set “selected slot” back to 0 and click on the “Start Core” button. The Turbo Chameleon core should now start again.

## 7.5.2. Flashing a Core

First of all, remove the SD card from the Chameleon. (The internal flash memory and the SD card use the same SPI bus internally, which sometimes causes flashing to fail)

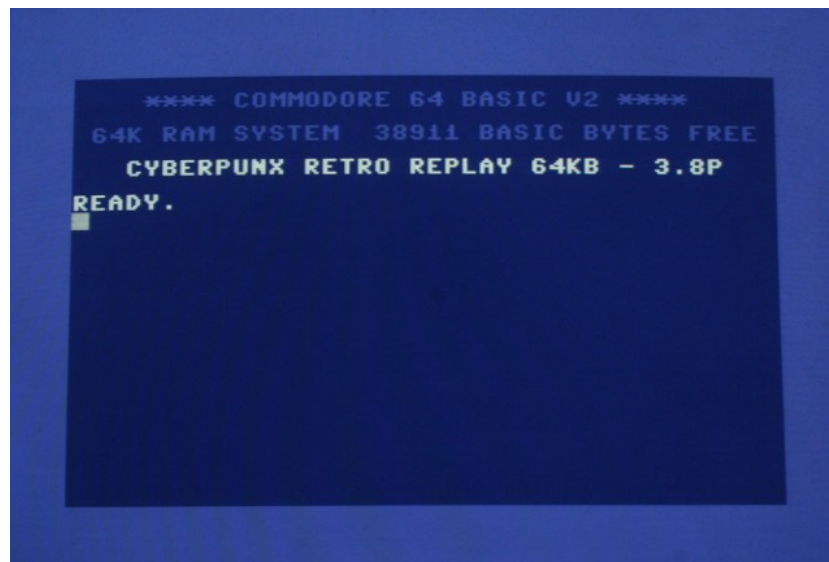
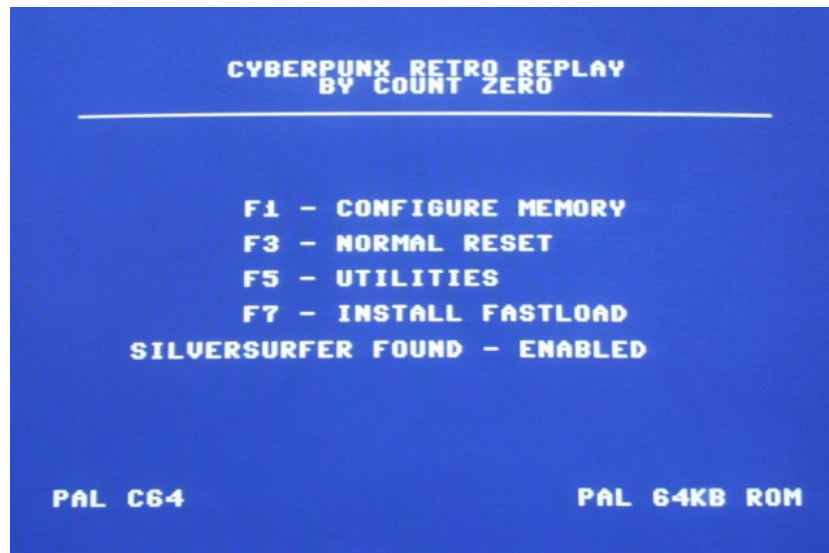
Now set the Selected slot number back to 0. Tick the box “**flash additional ROM**” and click on “**Flash .rbf/ROM**”. Two file requesters will open – please observe closely what each file requester is for! The first file you need to choose is the FPGA core, the second file is the ROM file that belongs to the core. These two files will be distributed with each Chameleon update. Make sure to select the right core file for the hardware (v2 is the hardware that does not require a breakout cable). After choosing the second file, the flash process will be started without further notice. Again, this is an engineering tool, so we have to give another word of **Caution**: The contents of the flash slot will be erased without any user interaction!

Now click on “**Start Core**” to test the core you've just flashed.

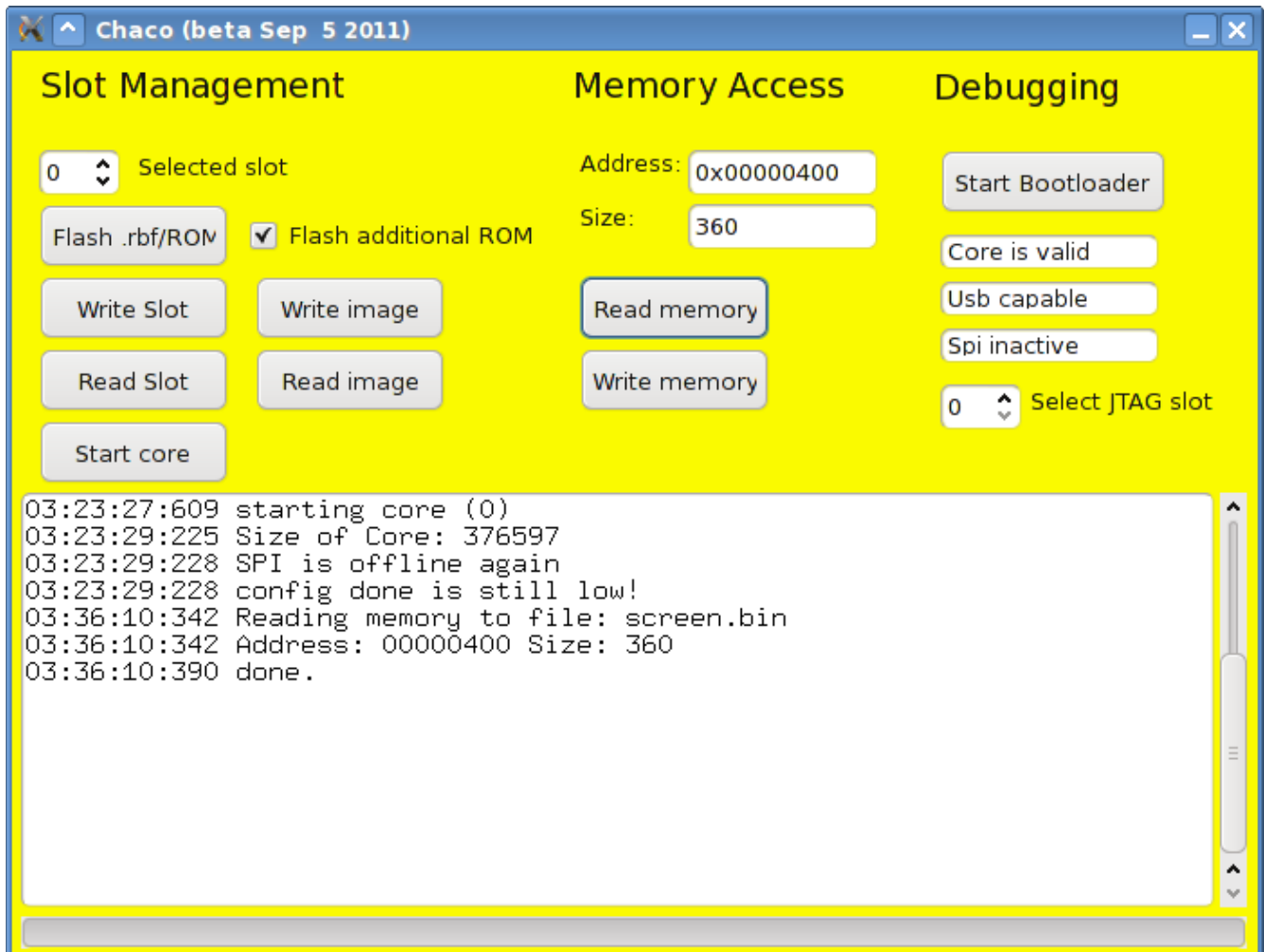


### 7.5.3. Accessing memory

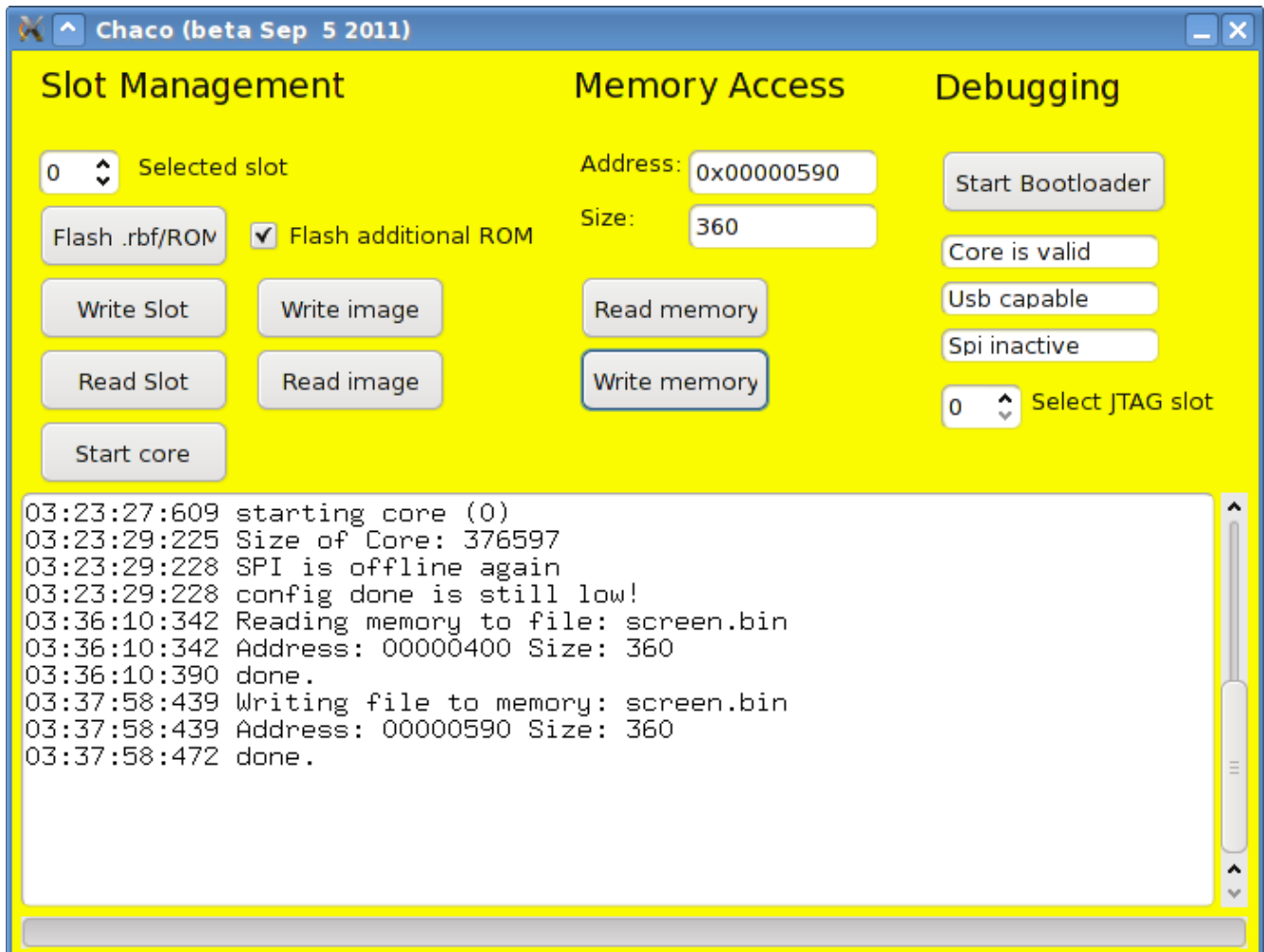
Now let's use ChaCo for some remote debugging! Power up the Chameleon and then select “reset with cartridge” (F7) followed by “INSTALL FASTLOAD” (again F7). The VGA screen should show something like this:



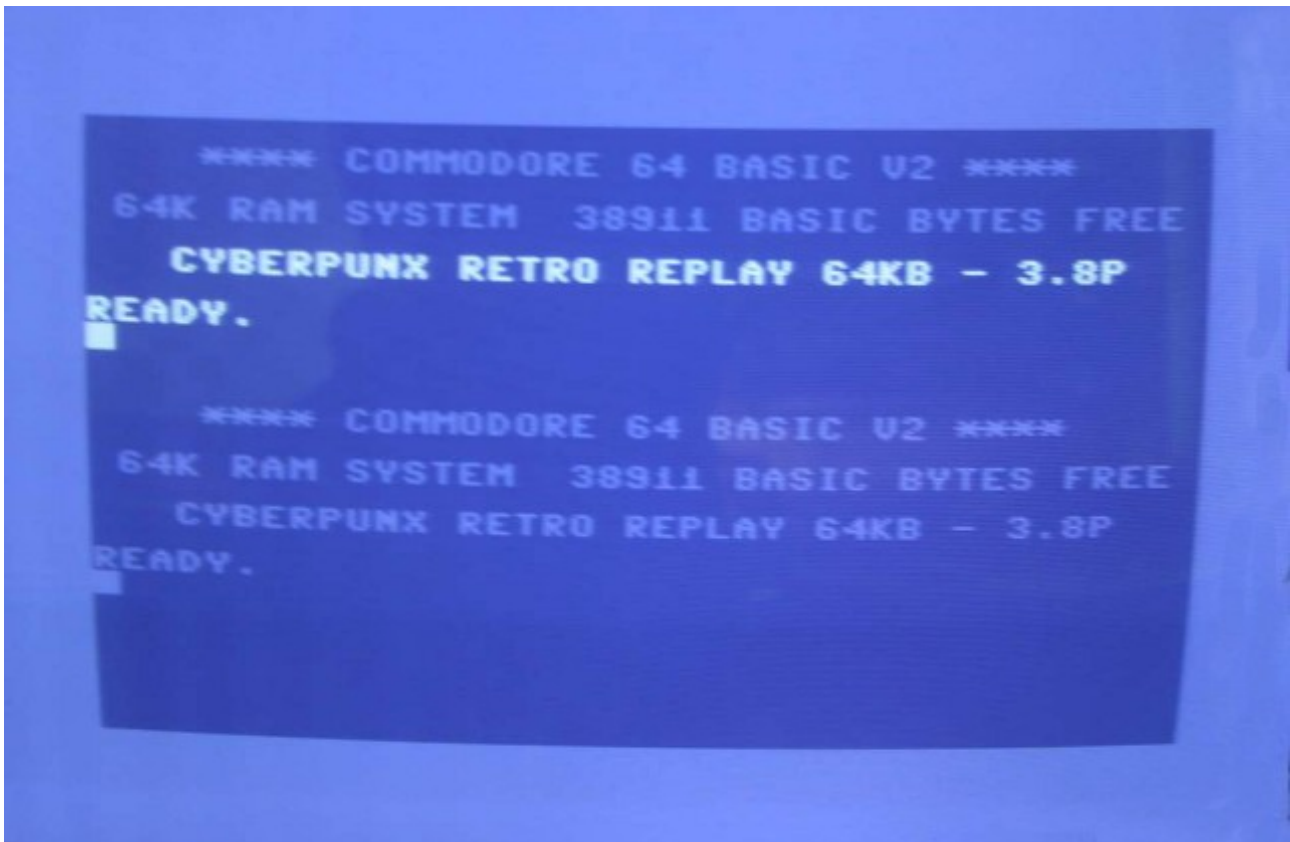
Now in ChaCo enter address 0x400 and size 360 into the fields for memory access. Then click on “**Read memory**”. ChaCo will read the memory of the C64 into a file that you can save to your PC's hard-disk. Let's call this file “screen.bin”.



In a second step, we'll write back this same file to the C64 memory, but to a different address. Let's use address 0x590. Then click on “**Write memory**” and choose the file you've just saved (screen.bin).



The resulting display on the Chameleon VGA screen should be like this:



What you've just done is to copy the first nine lines of the screen memory to lines ten and following of the C64 screen memory.

## 7.6. Command-line options

ChaCo recognizes these command-line options:

|           |                                     |
|-----------|-------------------------------------|
| --verbose | enable verbose messages             |
| --debug   | enable (perhaps lots) more messages |

## 7.7. *chacocmd*

*chacocmd* is a small command-line tool with essentially the same functionality as ChaCo, intended for use by developers in makefiles and scripts.

*chacocmd* recognizes these command-line options:

|   |                                  |
|---|----------------------------------|
| <code>-h --help</code>  | Show all options                 |
| <code>--verbose</code>  | enable verbose messages          |
| <code>--debug</code>  | enable debug messages            |
| <code>--noprogess</code>  | disable the progress indicator   |
| <code>--info</code>   | show flash info                  |
| <code>--status</code>   | show core status                 |
| <code>--addr &lt;num&gt;</code>                                       | set start address                |
| <code>--len &lt;num&gt;</code>  | set length                       |
| <code>--dumpmem</code>  | show memory dump                 |
| <code>--readmem &lt;name&gt;</code>                                   | read RAM to file                 |
| <code>--writemem &lt;name&gt;</code>                                  | write file to RAM                |
| <code>--dumpflash</code>  | show flash dump                  |
| <code>--readslot &lt;slot&gt; &lt;name&gt;</code>                     | read slot image to file          |
| <code>--readimage &lt;name&gt;</code>                                 | read full image to file          |
| <code>--flashrbf &lt;slot&gt; &lt;corename&gt; &lt;romname&gt;</code> | flash .rbf/.rom                  |
| <code>--flashslot &lt;slot&gt; &lt;name&gt;</code>                    | flash slot image                 |
| <code>--flashimage &lt;name&gt;</code>                                | flash full image                 |
| <code>--eraseslot &lt;slot&gt;</code>                                 | Erase slot                       |
| <code>--start &lt;slot&gt;</code>                                     | start core                       |
| <code>--jtagslot &lt;slot&gt;</code>                                  | set JTAG slot                    |
| <code>--bootloader</code>   | start microcontroller bootloader |

### 7.7.1. examples

Flash Chameleon core with menu rom to slot 0 and then start it:

```
$ chacocmd --flashrbf 0 chameleon_6g.rbf rom-menu.bin --start 0
```

write test.bin to address 0x1000 in c64 memory:

```
$ chacocmd --addr 0x1000 --writemem test.bin
```

## 8. Frequently asked questions

Since the Chameleon is under constant development, and this section did not really fit the nature of this manual, the FAQ was moved to the Individual Computers WIKI:

[http://wiki.icomp.de/wiki/Chameleon\\_FAQ](http://wiki.icomp.de/wiki/Chameleon_FAQ)

## 9. known Bugs and Limitations

Since the Chameleon is under constant development, this section of the manual has been removed to reduce maintenance. Please refer to the Individual Computers WIKI for up to date information on currently known bugs: [http://wiki.icomp.de/wiki/Chameleon\\_Bugs](http://wiki.icomp.de/wiki/Chameleon_Bugs)

[http://wiki.icomp.de/wiki/Chameleon\\_Core\\_Bugs](http://wiki.icomp.de/wiki/Chameleon_Core_Bugs) has a detailed list of known problems in the FPGA core, [http://wiki.icomp.de/wiki/Chameleon\\_Menu\\_Bugs](http://wiki.icomp.de/wiki/Chameleon_Menu_Bugs) is the equivalent for the Menu system and [http://wiki.icomp.de/wiki/ChaCo\\_Bugs](http://wiki.icomp.de/wiki/ChaCo_Bugs) collects bugs in ChaCo.

[http://wiki.icomp.de/wiki/Chameleon\\_SD\\_Card\\_Problems](http://wiki.icomp.de/wiki/Chameleon_SD_Card_Problems) lists a few problematic SD cards.

[http://wiki.icomp.de/wiki/Chameleon\\_Monitors](http://wiki.icomp.de/wiki/Chameleon_Monitors) lists some working (and non working) monitors.

## 10. Reporting Bugs

If you find any bugs, please report them in our forum (<http://forum.icomp.de>). Before doing so, please make sure you are running the latest version of the Turbo Chameleon FPGA Core, which can be downloaded at <http://wiki.icomp.de/wiki/Chameleon> .

To make it easier and more efficient for us to handle bug reports, please provide the following information with them:

- whether you are using the Chameleon in standalone or cartridge mode, and whether you are using PAL or NTSC
- if you are using cartridge mode, then it would be helpful to know which kind of C64 it is (board revision, type of CIAs, VIC, SID)
- a detailed description that enables us to reproduce the bug

Have a look at [http://wiki.icomp.de/wiki/Chameleon\\_Bugs](http://wiki.icomp.de/wiki/Chameleon_Bugs) for details.



## 11. Developer information

Information for developers is made available separately from this manual. Look at the turbo Chameleon 64 project page ( <http://www.syntiac.com/chameleon.html> ) for the most recent version of these documents:

- `chameleon_progmanual.pdf` – contains register description and programming information for the turbo Chameleon 64 FPGA core.
- `chameleon_coredef.pdf` – hardware overview for FPGA core developers
- `chameleon_schematics.pdf` – Chameleon board schematics for FPGA core developers
- `chameleon_fpga_pins.zip` - a pin file that you can add to the Quartus \*.qsf project settings file.
- `chameleon_v5_hwtest_src.zip` – the source code for the FPGA hardware test core
- `cdtv_remote.zip` - protocol description of the CDTV remote in open-office format.

We are working on making more example- and framework FPGA code available, please contact us if you have specific questions which are not yet answered in the above documents.

To get in touch with other Chameleon users and developers you might want to join our web forum (<http://forum.icomp.de>).

## 12. Related projects

This section of the manual has been removed to reduce maintenance. Please refer to the Individual Computers WIKI ( <http://wiki.icomp.de/wiki/Chameleon> ) for up to date information on related projects, such as other FPGA cores.

## 13. Appendix

### 13.1. PS/2 Keyboard Key Assignment

| PS/2 Key (US) | PS/2 Key (DE) | C64 Function                     |
|---------------|---------------|----------------------------------|
| ALT           | ALT           | CBM Key                          |
| ESCAPE        | ESCAPE        | RUN/STOP                         |
| F1            | F1            | F1                               |
| F2            | F2            | RShift + F2                      |
| F3            | F3            | F3                               |
| F4            | F4            | RShift + F4                      |
| F5            | F5            | F5                               |
| F6            | F6            | RShift + F5                      |
| F7            | F7            | F7                               |
| F8            | F8            | RShift + F7                      |
| F9            | F9            | £ (Pound)                        |
| F10           | F10           | +                                |
| F11           | F11           | Left cartridge button            |
| F12           | F12           | Middle (Freeze) cartridge button |
| Print Screen  | Print Screen  | Right (Reset) cartridge button   |
| PAUSE         | PAUSE         | RESTORE                          |
| ~             | ^ / °         | ⇐ (Arrow left)                   |
| —             | ß / ?         | —                                |
| = / +         | ' / `         | =                                |
| Backspace     | Backspace     | DEL/INST                         |
| Home          | Home          | HOME/CLR                         |
| Page Up       | Page Up       | F1                               |
| Page Down     | Page Down     | F7                               |
| [ / {         | Ü             | @                                |
| ] / }         | + / *         | *                                |
| \ /           | # / '         | ↑ (Arrow up)                     |
|               | Ö             | :                                |
|               | Ä             | ;                                |
|               | - / _         | /                                |

### 13.2. PS/2 Keyboard Joystick Assignment

| PS/2 Key  | C64 Function            |
|-----------|-------------------------|
| NUM-LOCK  | Select port 1 or port 2 |
| Numeric 0 | Joystick Fire Button    |
| Numeric 1 | Joystick Left + Down    |
| Numeric 2 | Joystick Down           |
| Numeric 3 | Joystick Right + Down   |
| Numeric 4 | Joystick Left           |
| Numeric 6 | Joystick Right          |
| Numeric 7 | Joystick Left + Up      |
| Numeric 8 | Joystick Up             |
| Numeric 9 | Joystick Right + Up     |

Take note that it is possible with the keyboard to press both "Joystick Left" and "Joystick Right" at the same time (same is true for up and down). There are a few games that crash when you do so. Don't blame Chameleon for the crash, but the programmer that wrote the fragile game code.

### 13.3. Amiga Keyboard Key Assignment

| Amiga Key (US) | Amiga Key (Euro) | C64 Function                     |
|----------------|------------------|----------------------------------|
| Amiga          | Amiga            | CBM Key                          |
| ESCAPE         | ESCAPE           | RUN/STOP                         |
| F1             | F1               | F1                               |
| F2             | F2               | RShift + F2                      |
| F3             | F3               | F3                               |
| F4             | F4               | RShift + F4                      |
| F5             | F5               | F5                               |
| F6             | F6               | RShift + F5                      |
| F7             | F7               | F7                               |
| F8             | F8               | RShift + F7                      |
| F9             | F9               | £ (Pound)                        |
| F10            | F10              | +                                |
| Numpad (       | Numpad } / ]     | Left cartridge button            |
| Numpad /       | Numpad /         | Middle (Freeze) cartridge button |
| Numpad *       | Numpad *         | Right (Reset) cartridge button   |
| HELP           | HELP             | RESTORE                          |
| ~ / `          | ~ / `            | ⇐ (Arrow left)                   |
| - / _          | ß / ?            | –                                |
| = / +          | ' / `            | =                                |
| Backspace      | Backspace        | DEL/INST                         |
| Del            | Del              | HOME/CLR                         |
| [ / {          | Ü                | @                                |
| ] / }          | + / *            | *                                |
| \ /            | \ /              | ↑ (Arrow up)                     |
| ; / :          | Ö                | :                                |
| # / @          | Ä                | ;                                |
| / / ?          | - / _            | /                                |
| , / <          | , / ;            | , / <                            |
| . / >          | . / :            | . / >                            |
| Numpad +       | Numpad +         | +                                |
| Numpad -       | Numpad -         | -                                |

**13.4. Amiga Keyboard Joystick Assignment**

| <b>Amiga Key (US)</b> | <b>Amiga Key (Euro)</b> | <b>C64 Function</b>     |
|-----------------------|-------------------------|-------------------------|
| Numlock               | Numpad { / [            | Select port 1 or port 2 |
| Numeric 0             | Numeric 0               | Joystick Fire Button    |
| Numeric 1             | Numeric 1               | Joystick Left + Down    |
| Numeric 2             | Numeric 2               | Joystick Down           |
| Numeric 3             | Numeric 3               | Joystick Right + Down   |
| Numeric 4             | Numeric 4               | Joystick Left           |
| Numeric 6             | Numeric 6               | Joystick Right          |
| Numeric 7             | Numeric 7               | Joystick Left + Up      |
| Numeric 8             | Numeric 8               | Joystick Up             |
| Numeric 9             | Numeric 9               | Joystick Right + Up     |

### 13.5. CDTV Remote Control Key Assignment

| Infrared CDTV Remote Key     | C64 Function                          | Menu Function<br>(in CDTV mode) |
|------------------------------|---------------------------------------|---------------------------------|
| 1                            | F1                                    | Page up                         |
| 2                            | RShift + F1                           | Home                            |
| 3                            | F3                                    | View file                       |
| 4                            | RShift + F3                           | User-Action menu                |
| 5                            | F5                                    | Mount                           |
| 6                            | RShift + F5                           | Unmount                         |
| 7                            | F7                                    | Page down                       |
| 8                            | RShift + F7                           | End                             |
| 9                            | RUN/STOP                              | exit/abort                      |
| 0                            | Space bar                             | File-Action menu                |
| ESCAPE                       | Left Arrow                            | exit/abort                      |
| ENTER                        | RETURN                                |                                 |
| REW                          | CRSR Left (RShift + CRSR Right)       | Left                            |
| PLAY/PAUSE                   | CRSR Up (RShift + CRSR Down)          | Up                              |
| FF                           | CRSR Right                            | Right                           |
| STOP                         | CRSR Down                             | Down                            |
| GENLOCK                      | Left Push Button                      |                                 |
| CD/TV                        | Middle Push Button                    | Freeze / Menu                   |
| POWER                        | Right Push Button                     | Reset                           |
| Vol Up                       | +                                     |                                 |
| Vol Down                     | -                                     |                                 |
| M/J Switch in MOUSE Position | Select Joystick 1                     |                                 |
| M/J Switch in JOY Position   | Select Joystick 2                     |                                 |
| A                            | Fire                                  |                                 |
| B                            | Auto Fire (8 Hz) or "Joystick Up" (*) |                                 |

(\*) Some C64 games use "Joystick Up" to jump. To make this a bit more comfortable, the action triggered by the B button can be adjusted as follows:

- Hold the B button and move the mouse/joystick slide switch to the opposite position. Now B works like "Joystick Up"
- Do the same again to switch back to auto fire

The remote control must have "visual contact" with the IR receiver of the Chameleon.

**13.6. supported Cartridge types**

| CRT ID |   |
|--------|---|
| 0      | generic (8k Game, 16k Game, Ultimax)    |
| 1      | Action Replay v5 (v4.2 and higher ROMs) |
| 2      | KCS Power Cartridge                     |
| 3      | Final Cartridge 3                       |
| 4      | Simons Basic                            |
| 5      | Ocean (*1)                              |
| 6      | Expert Cartridge                        |
| 7      | Funplay / Powerplay (*1)                |
| 8      | Supergames                              |
| 9      | Atomic Power / Nordic Power             |
| 10     | Epyx Fastload                           |
| 11     | Westermann                              |
| 15     | C64 Game System (System 3) (*1)         |
| 16     | Warpspeed                               |
| 17     | Dinamic (*1)                            |
| 18     | Zaxxon                                  |
| 19     | Magic Desk (*1)                         |
| 20     | Super Snapshot v5                       |
| 21     | Comal 80                                |
| 23     | ROSS                                    |
| 28     | Mikro Assembler                         |
| 31     | Stardos                                 |
| 32     | Easyflash (*1)                          |
| 34     | Capture                                 |
| 36     | Retro Replay / Nordic Replay            |
| 37     | MMC64 (*2)                              |
| 43     | Prophet64 (*1)                          |
| 51     | Mach5                                   |
| 53     | Pagefox                                 |
| 54     | Kingsoft (Business Basic, S'more)       |

\*1) These types can only be used in slot 3 and 4, because they are (or may be) larger than 64k.

\*2) MMC64 is handled separately and can not be loaded to normal cartridge slots.



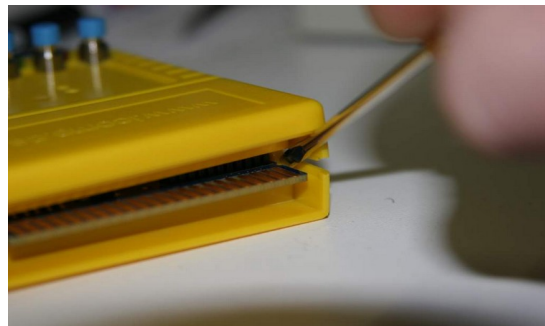
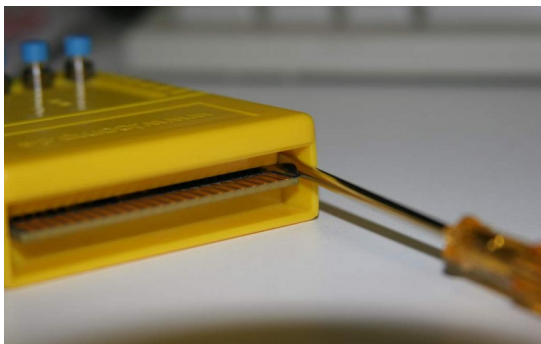
## 13.7. Opening the Cartridge

To install the RTC battery (CR-1220) or a RR-NET, you will have to open the Cartridge - it may look difficult, but if you know where to lever it open, it's really easy. All you need is a flat screwdriver. Please follow the steps given here, don't change the order of the steps, because you might break the case otherwise!

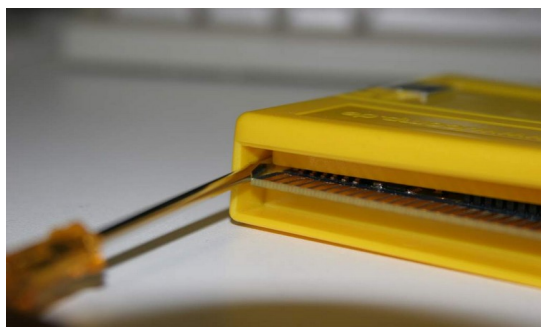
### 13.7.1. Chameleon v1

Use the flat screwdriver as a lever. Be sure not to touch the Chameleon board, only the case must be touched! There will be no warranty for scratched boards.

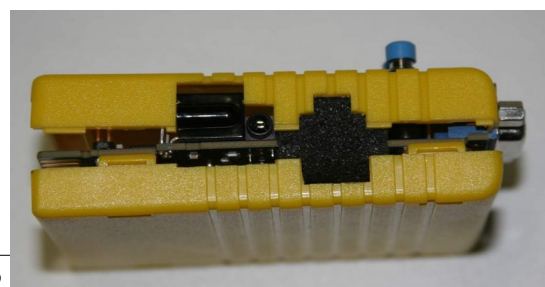
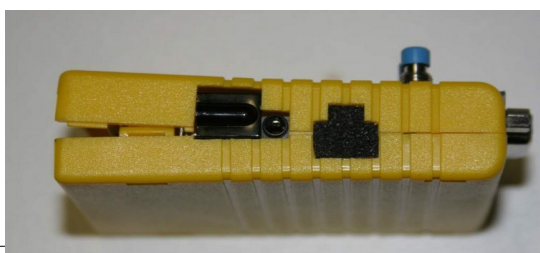
Use the zoom-function of your PDF viewer to exactly see where the lever needs to be placed.



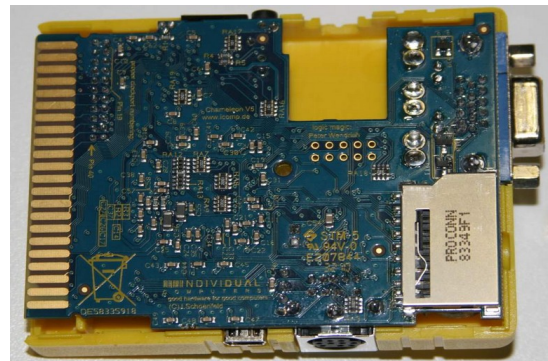
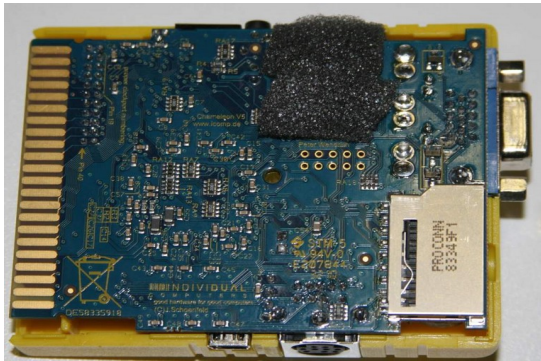
Next, use the screwdriver on the left side of the Chameleon case. Again, be sure to only lever the case, not the board!



With the opening in the Chameleon case for the mini-DIN connector, the left half of the case top is fairly weak. We therefore recommend to open the right side of the case first:



Remove the bottom by tilting it away; the left side (weakest side of the top) will now come off fairly easy, almost no force is required. Remove the foam from the case and keep it in a safe place. The foam on your cartridge may look different.



### 13.7.2. Chameleon v2

Opening the second version of the chameleon is much easier, since the new cases are held together by a screw instead of plastic clips. Simply unscrew that screw and proceed as described above.

Chameleons produced in the first few months of 2019 have a PZ1 screw. **Make sure to use a PZ1 (PoziDriv) screwdriver - using the wrong kind of screwdriver it is very easy to apply too much force and damage the head of the screw.**

Later produced units have a T08 Torx screw.



## 13.8. Re-programming the USB micro-controller

**WARNING:** It is very unlikely that this kind of update must be done in the field. It is only documented for completeness. It's the worst-case scenario – Please do not attempt to do any of the following unless being told by us to do so. In the unlikely event that you will have to update the micro-controller, you will receive an appropriate .hex file and detailed instructions directly from individual computers. The micro-controller is a core component which is not meant to be user programmable, thus refrain from experimenting with it.

**UPDATING THE MICRO-CONTROLLER WITHOUT BEING TOLD DO SO AND/ OR USING A .HEX FILE NOT ENDORSED BY INDIVIDUAL COMPUTERS VOIDS WARRANTY.**

Should the micro-controller require an update, you will need the “FLIP” tool from the Atmel website. FLIP is a Java application available for Windows and Linux. If you run under Linux, remember to run the tool as “root”, otherwise it will not get access to the USB port. You can easily find FLIP using Google – URL not given here, as links might change in the future. FLIP is the same for a large family of micro-controllers, the tool is the same for all of them.

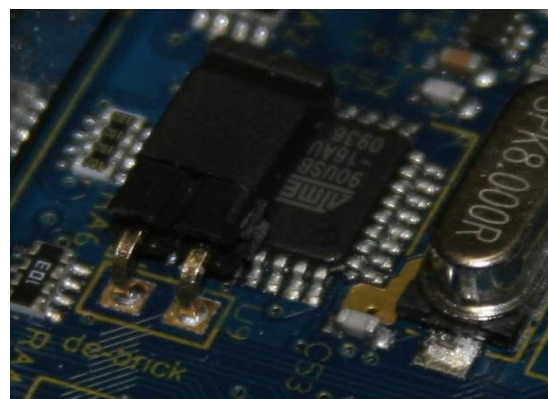
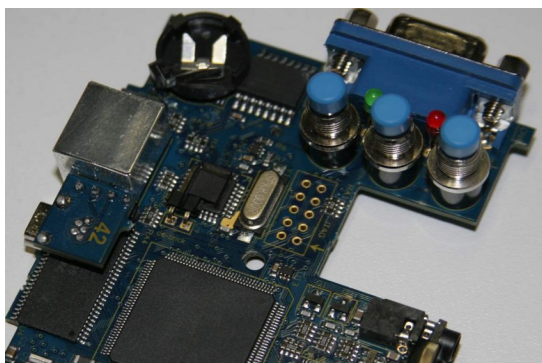
If you run under Windows, you will need a FLIP driver. New Windows versions (Vista, 64-bit versions) either require some expert settings, or they require a signed driver. Download the signed driver from our website and install it. Also, you may need to download and update Java. Even if Java does not prompt you to re-start the computer after an update, it may be required. Especially if FLIP exits with a null-pointer error message, just restart the computer, and it'll work.

Start FLIP – it will come up with an error message that a device was not found. This is expected behaviour. Click OK on that requester.

Now click on “settings” and select “preferences”. Within the preferences dialogue, select the rightmost tab “connect/closing”. Tick the box “Auto-connect”, then click on OK.

### 13.8.1. Chameleon v1

Now locate the de-brick jumper of Chameleon and close it:



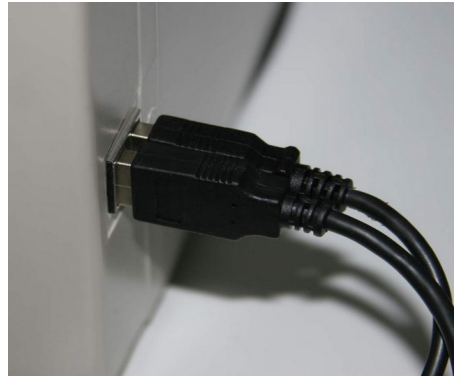
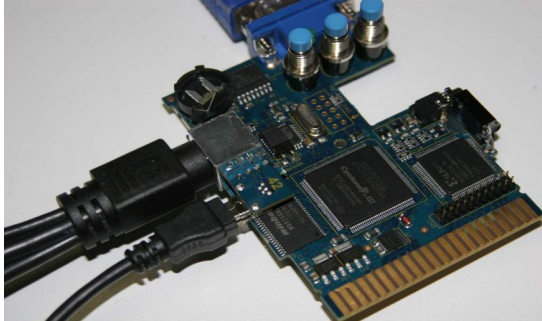


---

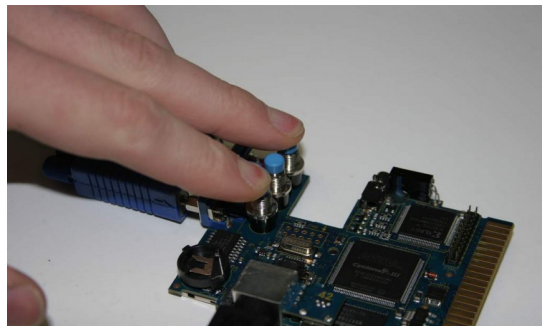
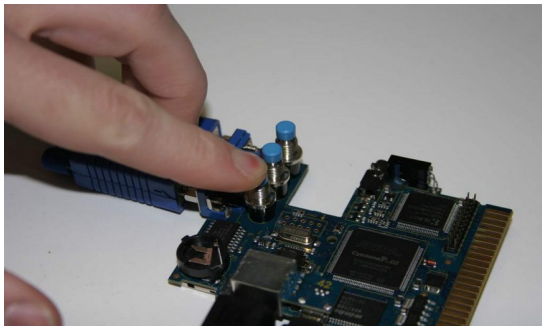
## Appendix

---

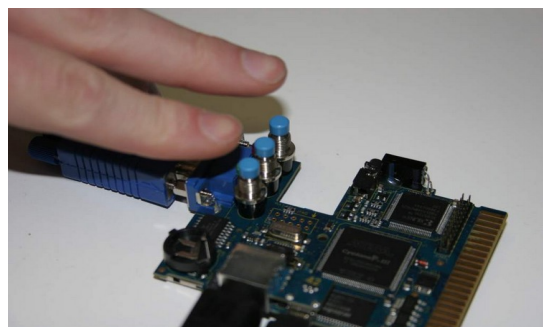
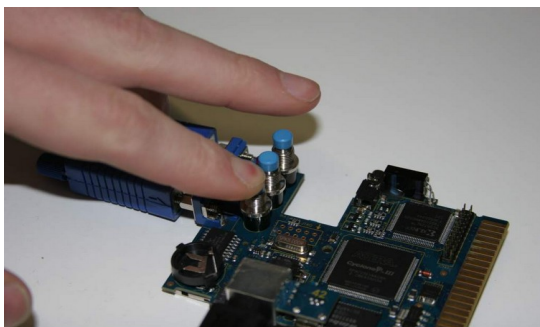
Now make two USB connections to your PC and switch on the PC. The USB connection of the breakout cable provides power to Chameleon, and the mini USB connection is your data connection:



Now place the micro-controller into de-brick mode with the following combination of button presses. Press and hold the left button, then press the right (reset) button:

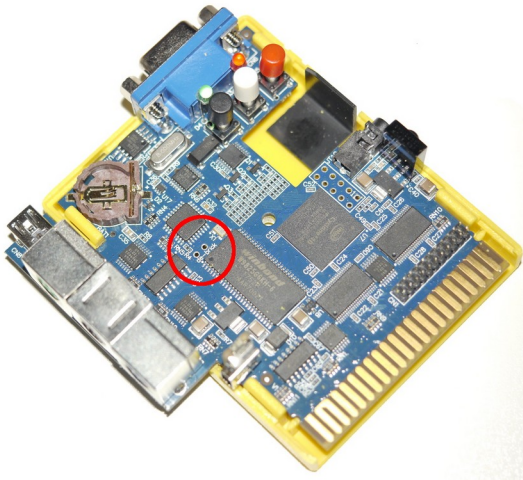


Then release the right (reset) button, but keep on holding the left button. As the last step, release the left button:

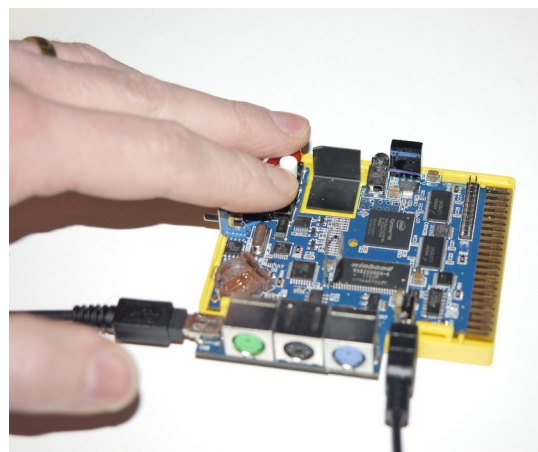
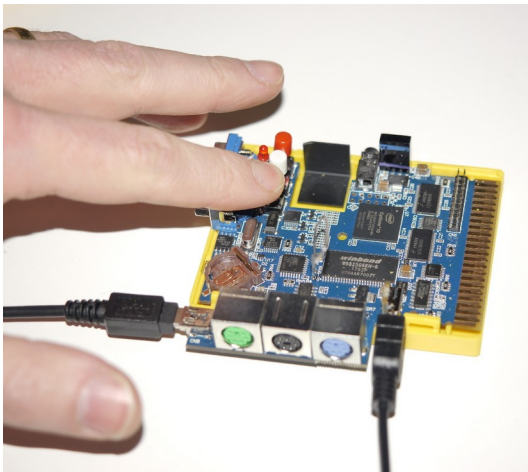


### 13.8.2. Chameleon v2

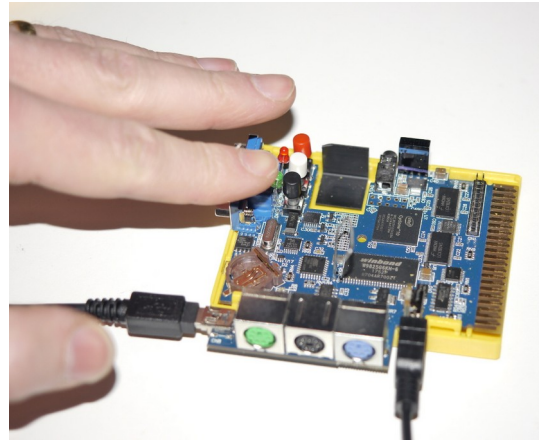
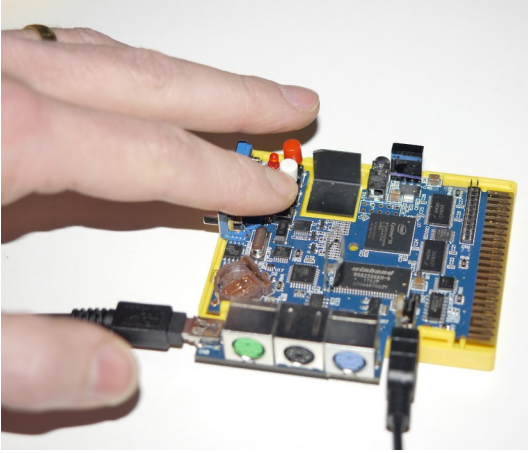
Since until now it wasn't necessary for customers to update the microcontroller, and it is unlikely to happen in the future, the de-brick jumper is not populated on the second version of the chameleon. In case of emergency the existing solder joints can be connected eg with a paperclip:



Now make two USB connections to your PC and switch on the PC. Then place the micro-controller into de-brick mode with the following combination of button presses. Press and hold the left button, then press the right (reset) button:



Then release the right (reset) button, but keep on holding the left button. As the last step, release the left button:

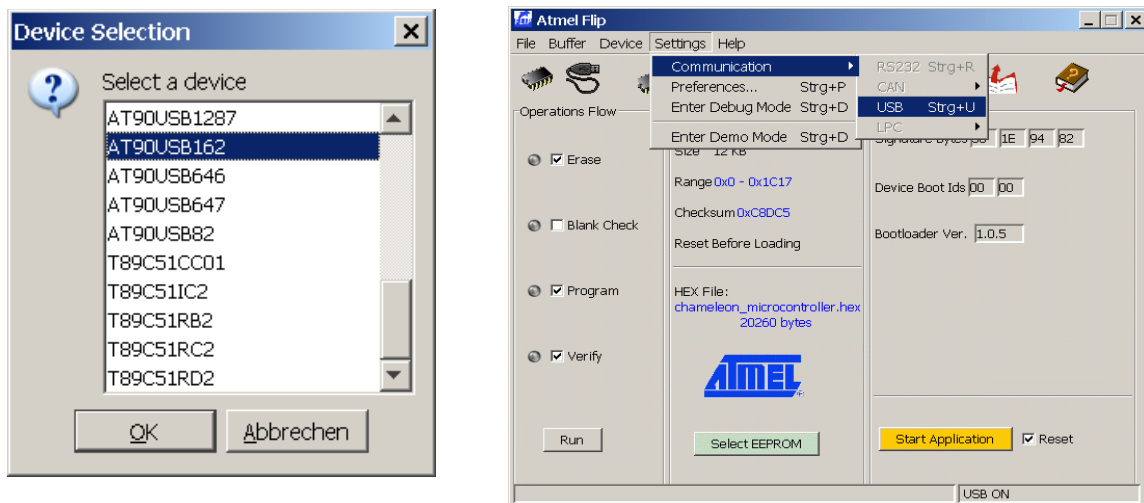


---

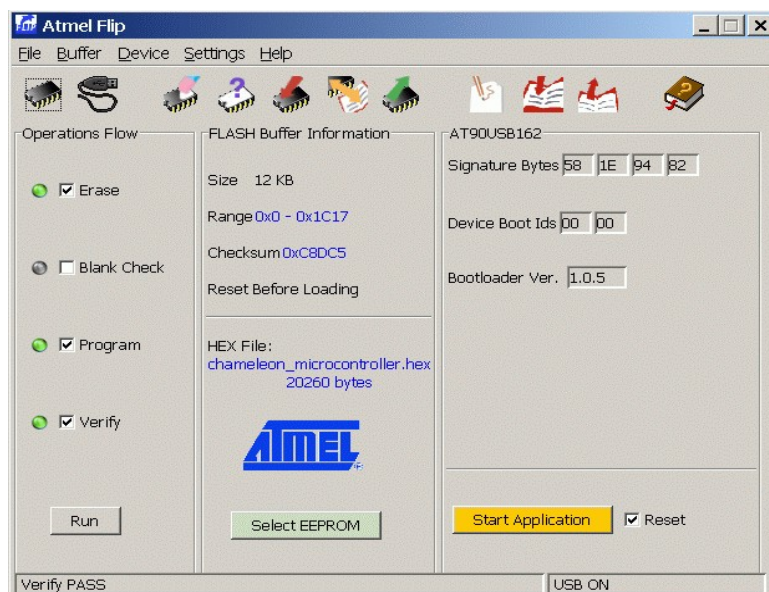
## Appendix

The micro-controller is now in de-brick mode and the FLIP tool will now find it.

First, select the processor type AT90USB162, then select Settings->Communication->USB:



Now load the micro-controller hex file by clicking on file->Load hex file. The final step is to click on the “Run” button within FLIP (at the bottom left of the window) and watch the program do its magic. After programming the micro controller, FLIP will indicate the successful steps with a green light next to the action tags:



Note: The checksum may be different for a possible update.

That's it – the micro-controller is now up to date, and you can leave de-brick mode by powering off and opening the de-brick jumper. Before you can make use of the new micro-controller software, you may need to update your version of ChaCo (the Chameleon Control software), and you may also have to update one of the flash slots. Please read the respective instructions for ChaCo for those steps.

### 13.8.3. Linux

In Linux "dfu-programmer" (<http://dfu-programmer.sourceforge.net/>) may be used:

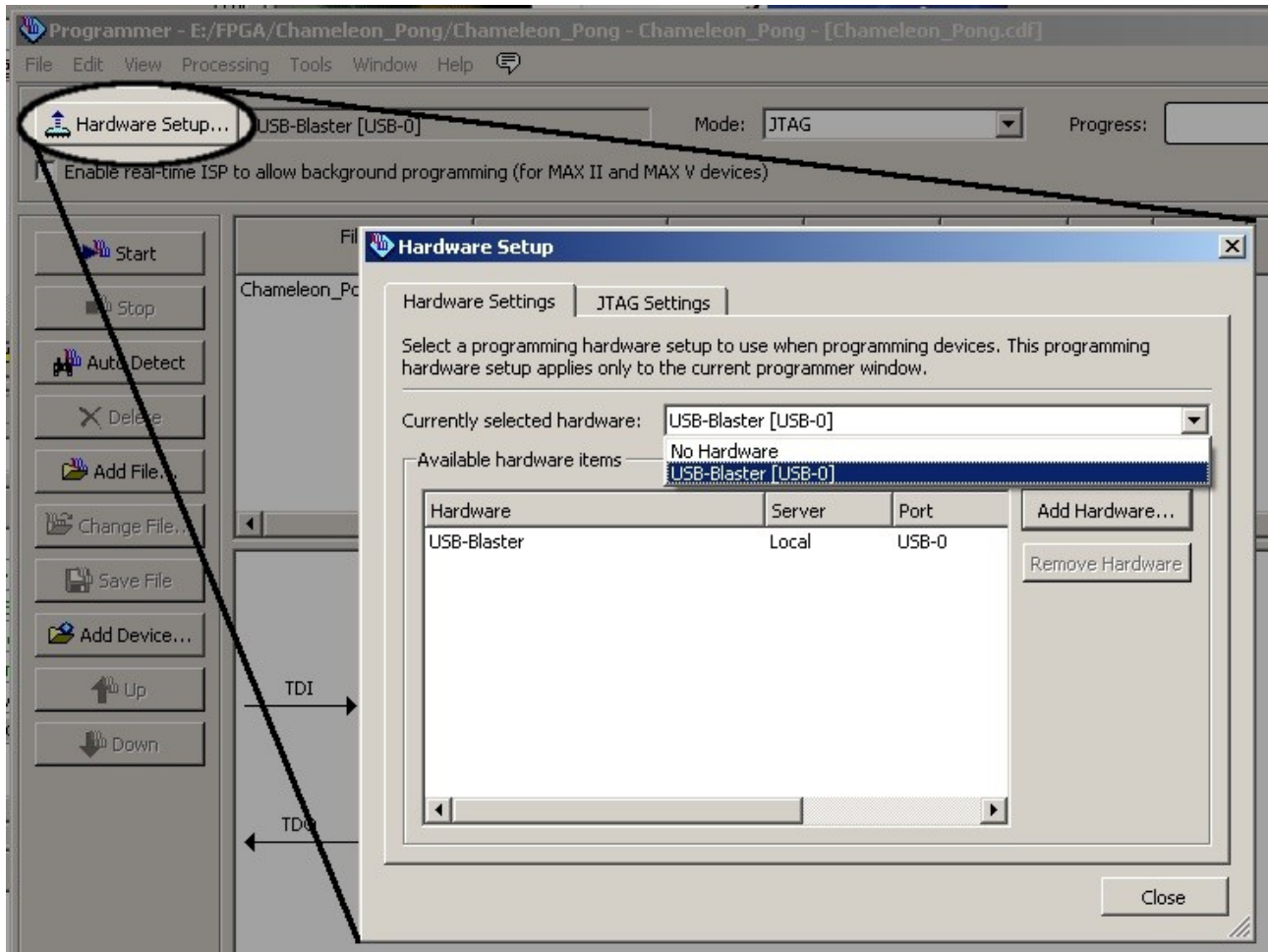
```
$ sudo chacocmd --bootloader
$ sudo dfu-programmer at90usb162 erase
$ sudo dfu-programmer at90usb162 flash
chameleon_microcontroller.hex
$ sudo dfu-programmer at90usb162 start
```



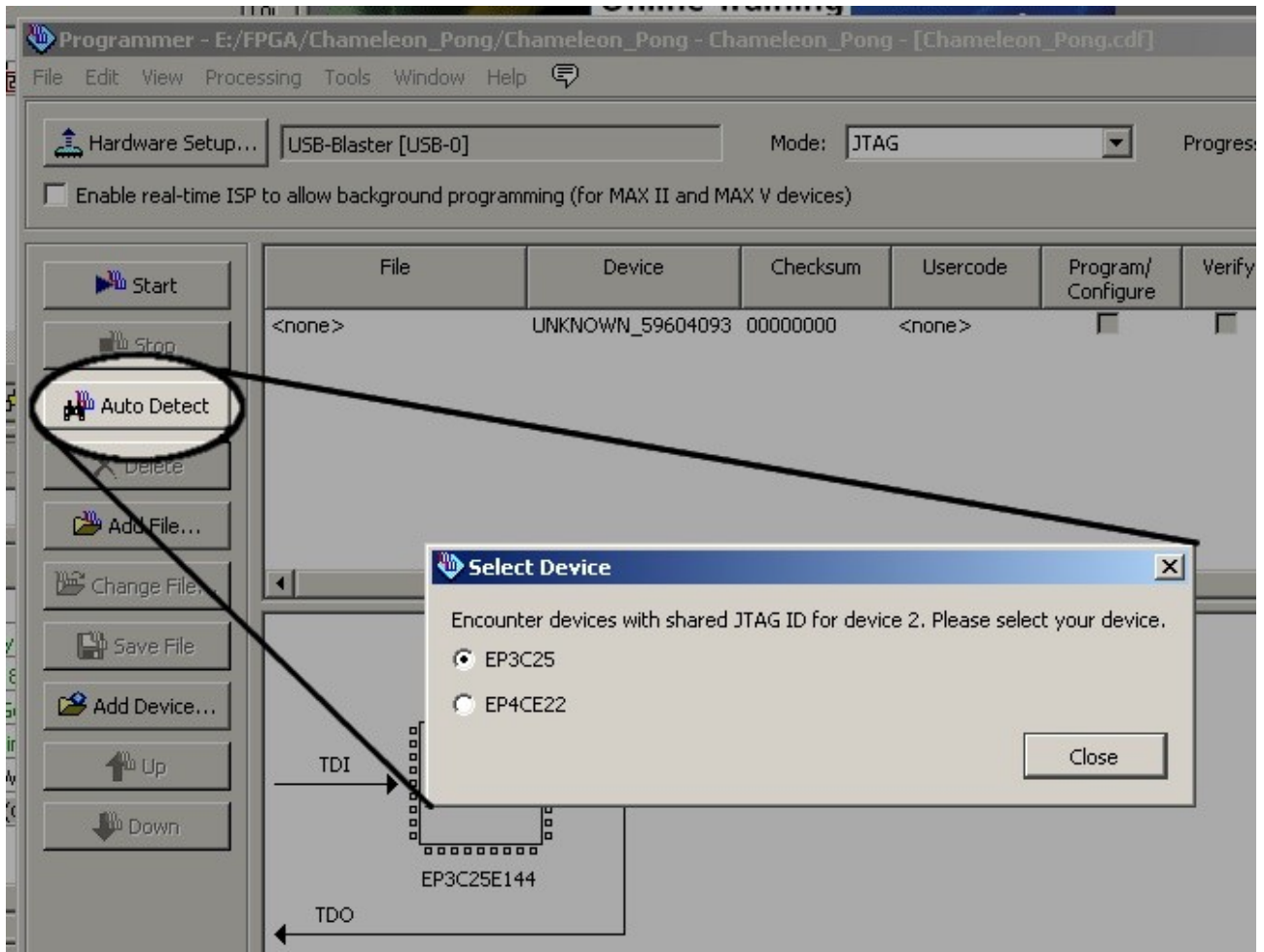
### 13.9. configuring Quartus for JTAG

Drivers are needed for the USB Blaster - they can be found in the Quartus II installation directory (quartus/drivers/usb-blaster).

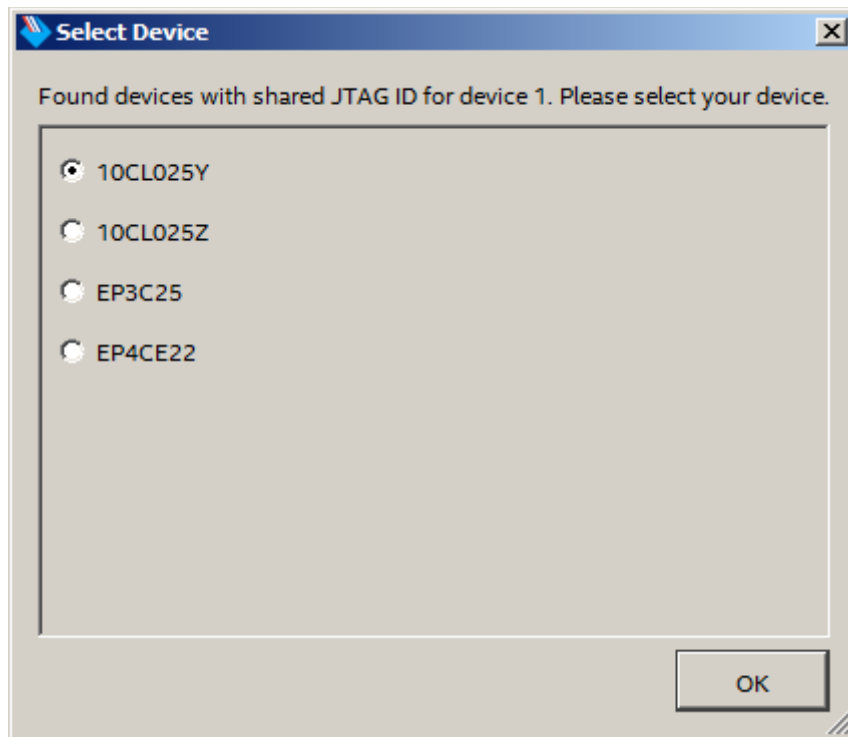
Click Hardware settings and select USB Blaster from the drop-down box.



The project's default settings won't work, because there's another device on the JTAG chain (the flash chip), so to solve that: Click the "Autodetect" button to reveal the second device. You'll also be asked whether the FPGA is an EP3C25 or EP4CE22. Pick the first.



When using the Chameleon v2, there will be only one JTAG device, select the 10CL025Y:

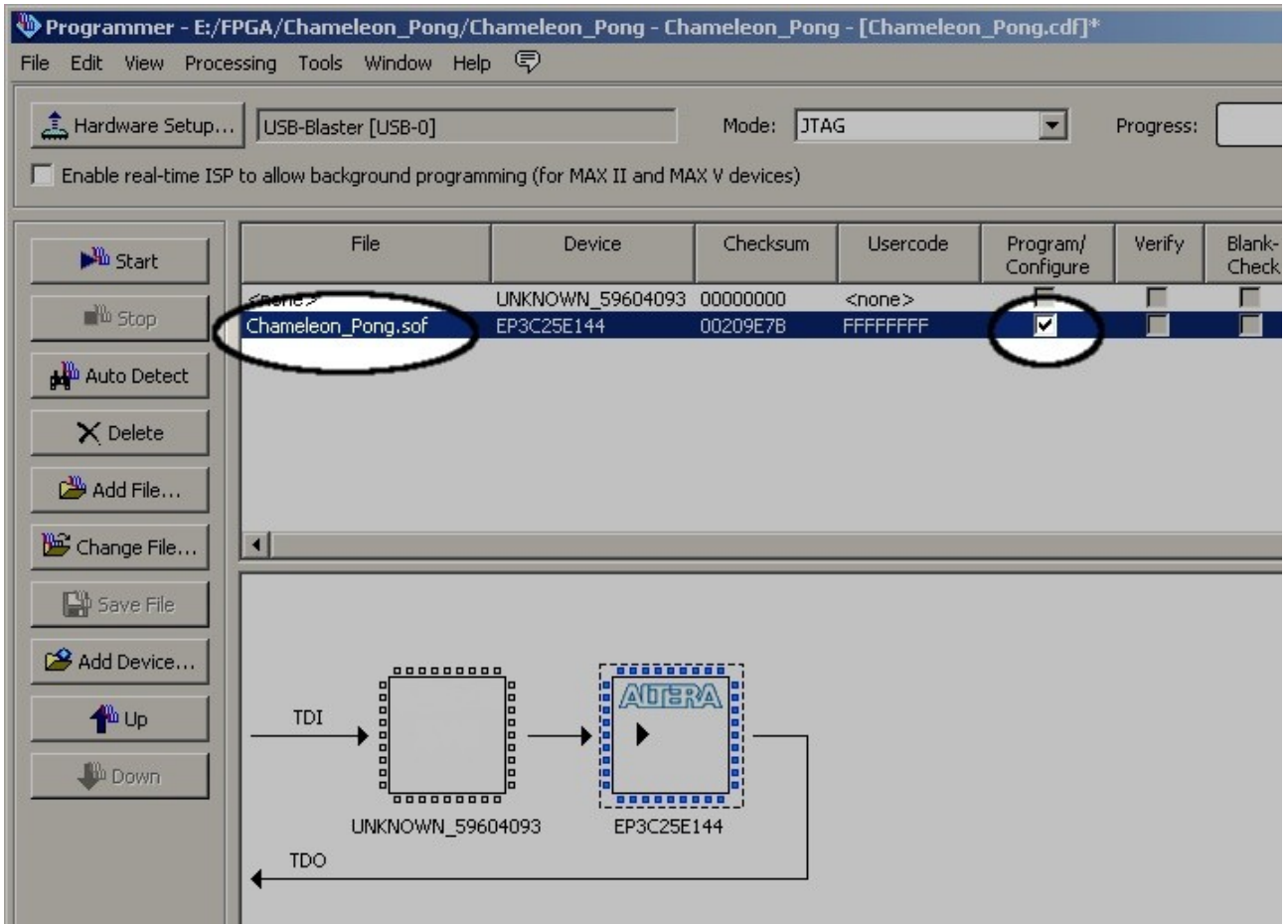


## Appendix

Now either click on the picture of the EP3C25 and click the Change File button, or double-click the "<none>" in the File column of the EP3C25's line in the top panel.

Pick the .sof file in the file dialog.

Check the "Program/Configure" checkbox in the EP3C25's row of the top panel.

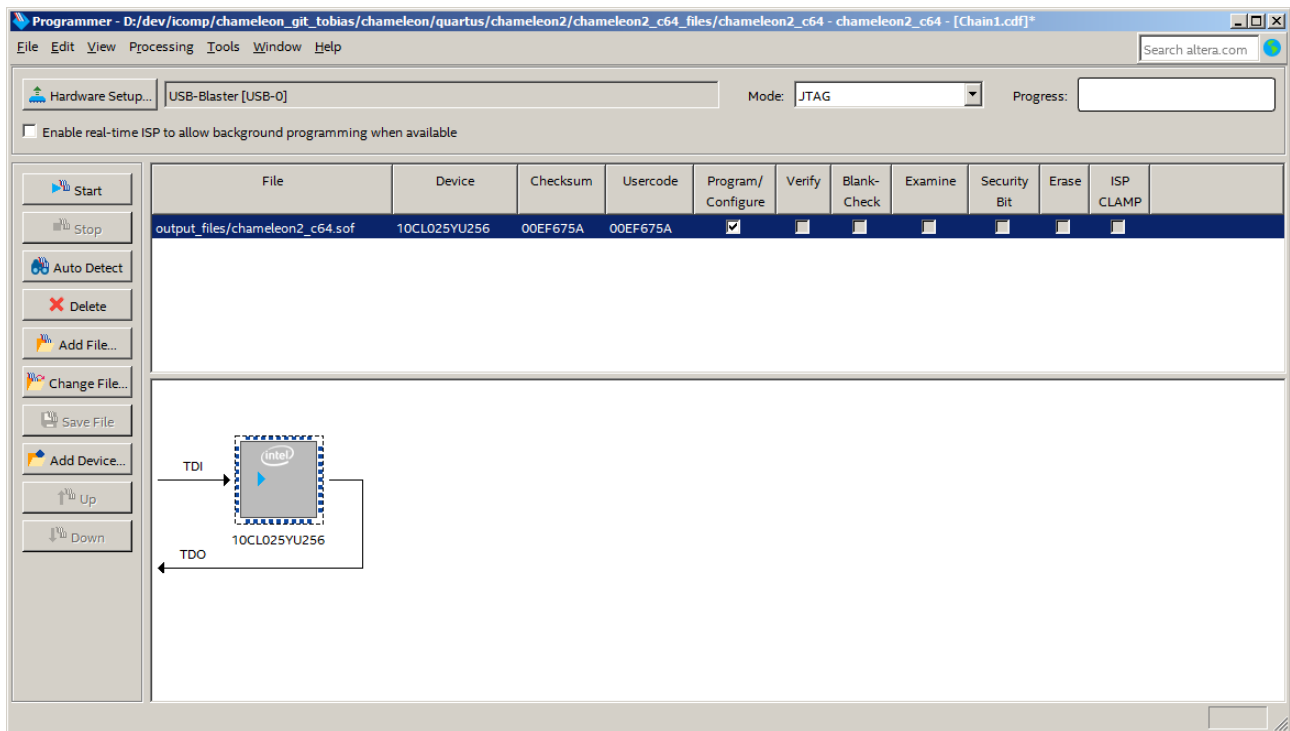


## Appendix

When using Chameleon v2 Hardware either click on the picture of the 10CL025V and click the Change File button, or double-click the "<none>" in the File column of the 10CL025V's line in the top panel.

Pick the .sof file in the file dialog.

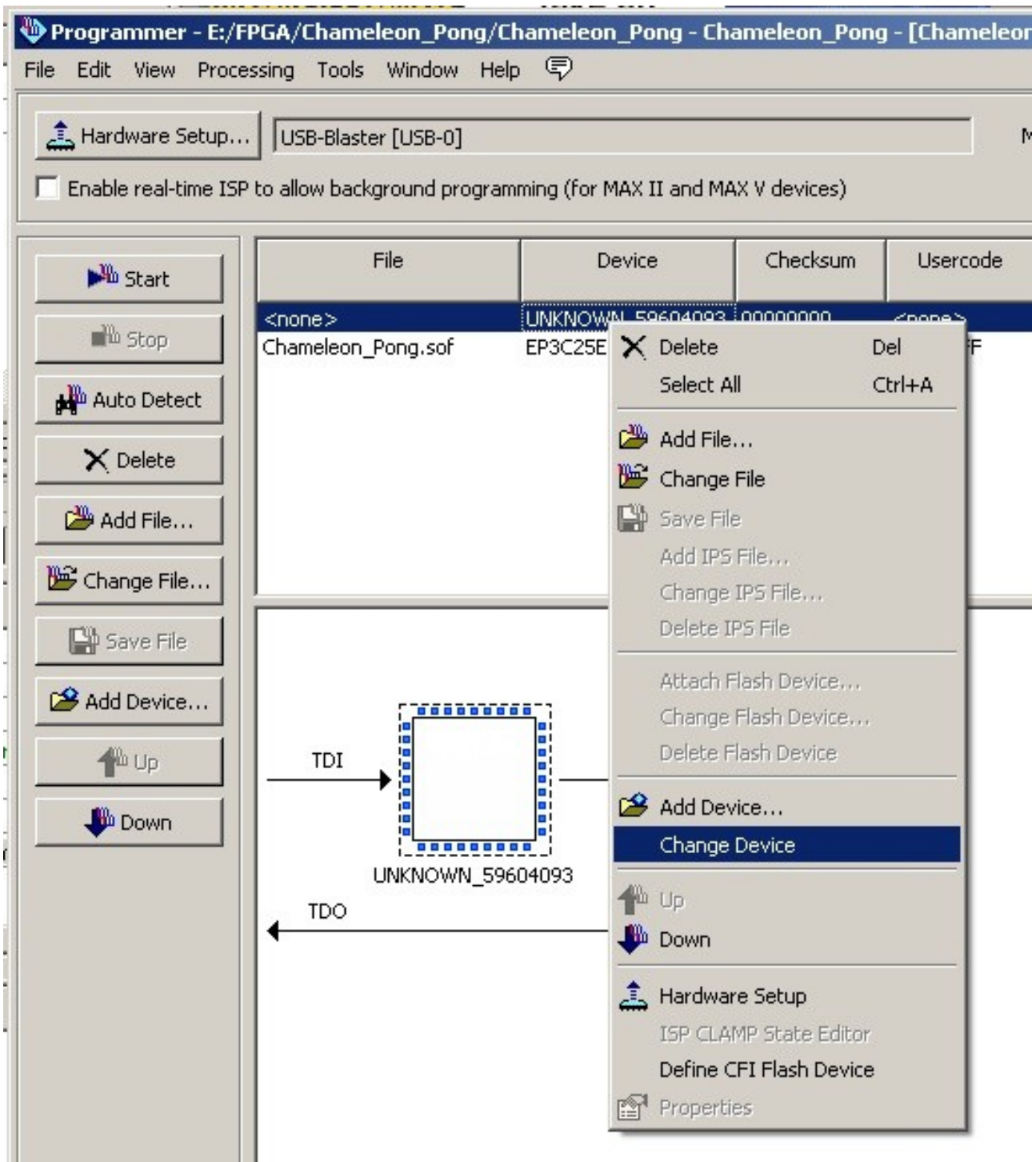
Check the "Program/Configure" checkbox in the 10CL025V's row of the top panel.



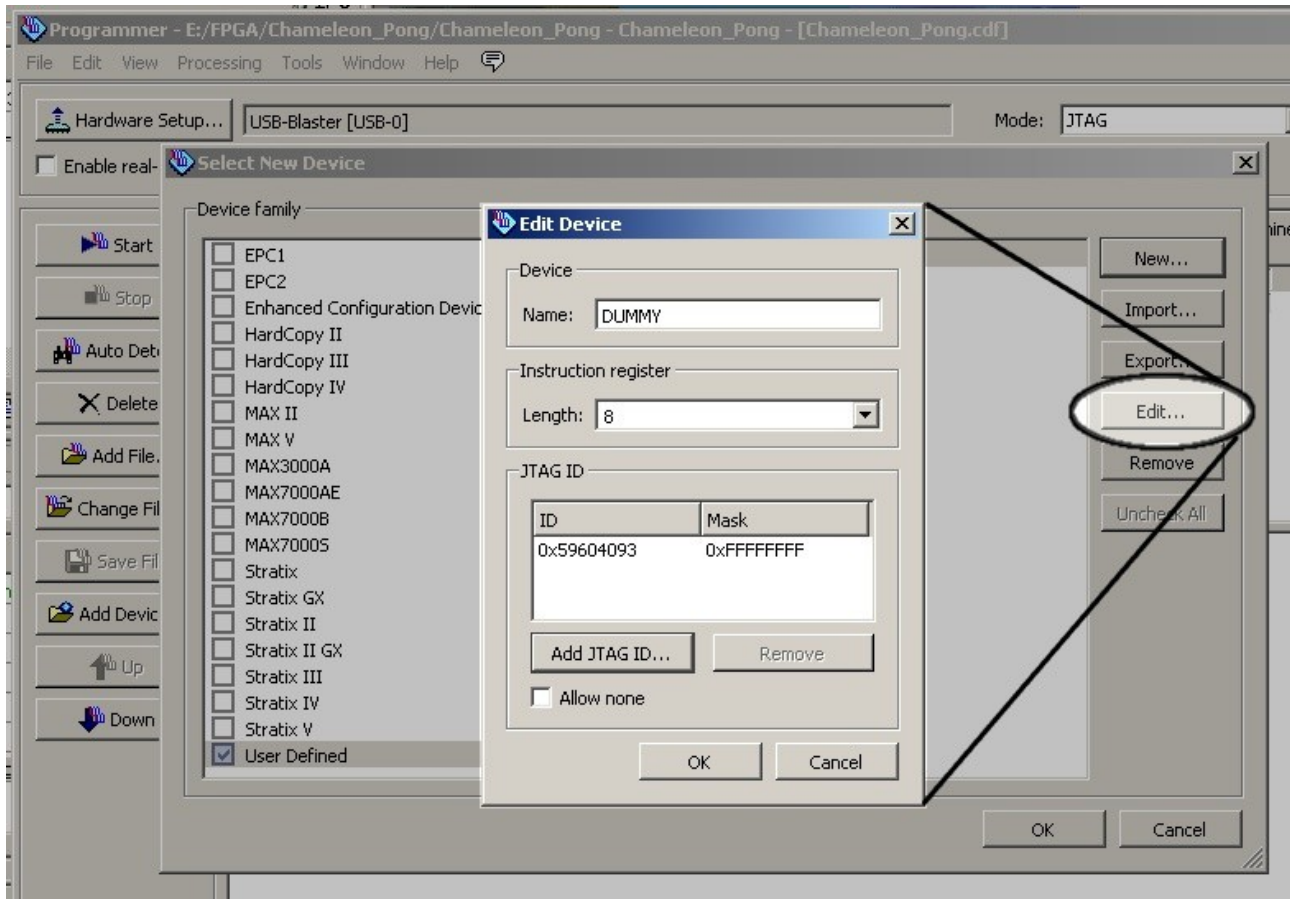
Finally, click Start, and the project should be uploaded to the FPGA.

When using the first version of the Chameleon hardware there's one more complication to take care of: When closing the programming window, Quartus will offer to save changes to the device chain, so you don't have to repeat this procedure each time you program the device. Unfortunately, saving will fail while the extra device is classed as "Unknown". This can be fixed just by giving the extra device a name.

In the top list, right-click on "Unknown\_59604093" and select "Change Device".



In the dialog that appears, the Unknown device should already be selected, so just click "Edit", and change the name to something else.



Quartus should now be able to save the device chain for future use.



### 13.10. Chameleon breakout-cable pinout (Chameleon v1)

CN1: Mini DIN 9 connector pinout:

shield: GND  
 1: IEC\_CLK  
 2: KDat  
 3: MDat  
 4: KClk  
 5: MClk  
 6: Vcc  
 7: IEC\_SRQ  
 8: IEC\_DAT  
 9: IEC\_ATN



Mating connector for RM158-DN508H-9. Pin numbering is taken from drawing.

CN2 and CN3 are coloured female Mini-DIN6 connectors:

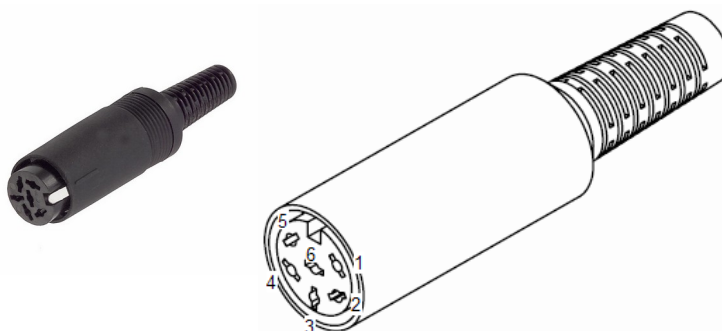
**purple CN2:** **green CN3:**

|             |             |
|-------------|-------------|
| shield: GND | shield: GND |
| 1: KDat     | 1: MDat     |
| 2: NC       | 2: NC       |
| 3: GND      | 3: GND      |
| 4: Vcc      | 4: Vcc      |
| 5: KClk     | 5: MClk     |
| 6: NC       | 6: NC       |



CN4 is a female hex DIN6 connector:

shield: GND  
 1: IEC\_SRQ  
 2: GND  
 3: IEC\_ATN  
 4: IEC\_CLK  
 5: IEC\_DAT  
 6: n.c.



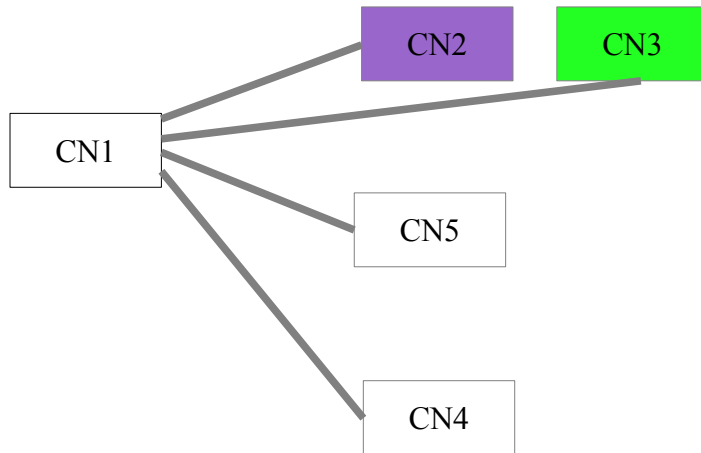
CN5 is a USB Type A connector:

shield: GND  
 1: Vcc  
 2: n.c.  
 3: n.c.  
 4: GND





## Wiring diagram



Connection CN1 to CN2 is a 4-core shielded AWG26 cable, length 200mm:

| CN1    |   | CN2              |
|--------|---|------------------|
| shield | - | shield and pin 3 |
| pin 2  | - | pin 1            |
| pin 4  | - | pin 5            |
| pin 6  | - | pin 4            |

Connection CN1 to CN3 is a 4-core shielded AWG26 cable, length 200mm:

| CN1    |   | CN3              |
|--------|---|------------------|
| shield | - | shield and pin 3 |
| pin 3  | - | pin 1            |
| pin 5  | - | pin 5            |
| pin 6  | - | pin 4            |

Connection CN1 to CN4 is a 4-core shielded AWG26 cable, length 200mm:

| CN1    |   | CN4              |
|--------|---|------------------|
| shield | - | shield and pin 2 |
| pin 1  | - | pin 4            |
| pin 7  | - | pin 1            |
| pin 8  | - | pin 5            |
| pin 9  | - | pin 3            |

Connection CN1 to CN5 is a single-core shielded AWG18 cable, length 800mm:

| CN1    |   | CN5              |
|--------|---|------------------|
| shield | - | shield and pin 4 |
| pin 6  | - | pin 1            |



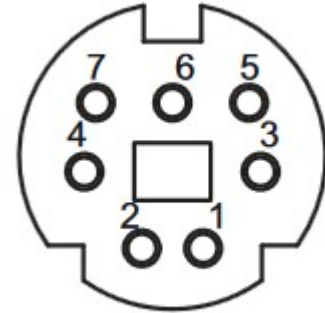
### 13.11. mini IEC cable pinout (Chameleon v2)

**CN1:** Mini DIN 7 male connector pinout:

shield: GND

- 1: GND
- 2: GND
- 3: IEC\_CLK
- 4: IEC\_ATN
- 5: IEC\_SRQ
- 6: Reset
- 7: IEC\_DAT

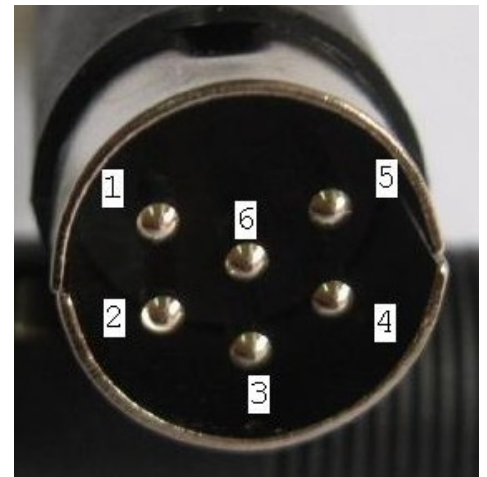
*Looking into female connector*



**CN2** is a male DIN6 connector, 240 degrees angle:

shield: GND

- 1: IEC\_SRQ
- 2: GND
- 3: IEC\_ATN
- 4: IEC\_CLK
- 5: IEC\_DAT
- 6: Reset



Connection CN1 to CN2 is a 5-core shielded AWG26 cable:

| CN1                  | CN2           | Signal Name |
|----------------------|---------------|-------------|
| Shield, Pin 1, Pin 2 | Shield, Pin 2 | GND         |
| Pin 3                | Pin 4         | IEC_CLK     |
| Pin 4                | Pin 3         | IEC_ATN     |
| Pin 5                | Pin 1         | IEC_SRQ     |
| Pin 6                | Pin 6         | Reset       |
| Pin 7                | Pin 5         | IEC_DAT     |

### **13.12. Technical Data**

- Altera EP3C25 / 10CL025V FPGA with 24624 LEs
- 16Mb Flash Memory, 32Mb RAM
- VGA connector
- MMC/SD card slot
- Connectors for PS/2 mouse, PS/2 keyboard and IEC bus via breakout cable
- Battery-backed real time clock (CR-1220 battery not included)
- IR receiver for CDTV remote control
- Connector for RR-NET
- USB link to a PC for debugging and data transfer

#### **13.12.1. Turbo Chameleon FPGA Core**

- Turbo function with full 6510 processor compatibility (incl. illegal opcodes)
- REU (memory expansion) with up to 16 MB RAM
- GEORAM (memory expansion) with up to 4 MB RAM
- Freezer (compatible with Retro Replay)
- VGA-output with 60 Hz refresh rate or more
- MMC/SD card slot with MMC64 compatibility
- 1541-emulation (2 drives)
- emulation of 1351 and Micromys on PS/2 mouse port

## 14. Credits

- Jens Schönfeld – Hardware design, production and documentation
- Peter Wendrich – Turbo Chameleon FPGA Core, programmers manual, first version of the ml monitor and help system
- Tobias Korbmacher – Menu System, ChaCo, user manual, testbench
- Stefan Seitz – USB Micro-controller code, first version of ChaCo
- Andreas Postler - case modifications, other manual steps in production
- Andreas Muttke - logo design
- Christian Vogelgsang – provided the chacocmd macOS port
- Alastair M. Robinson – contributed chapter 13.9 configuring Quartus for JTAG
- Geert De Prins – contributed a huge update to the help file

## 15. Fine print

The Chameleon is not designed, authorized or warranted to be suitable for use in life-support devices or systems or other critical operations. Inclusion of the product in such applications is understood to be fully at the customer's risk.

Names and other trademarks referred to within this manual are the property of their respective trademark holders. None of these trademark holders are affiliated with Individual Computers.

Windows® is a registered trademark of Microsoft Corporation in the United States and/or other countries.

Mac and macOS are trademarks of Apple Inc., registered in the United States and/or other countries.

For updates and further information visit <http://wiki.icomp.de/wiki/Chameleon>.

Join or forum at <http://forum.icomp.de>

Individual Computers Jens Schönfeld GmbH  
Im Zemmer 6  
52152 Woffelsbach  
Germany



good hardware for good computers.