

ULTIMATE - 64 *elite*
DOCUMENTATION

by Gideon Zweijtzer



INDEX:

2. Index
3. Getting Started, Installing
5. Real SID on Board revisions 1.1, 1.2 and 1.2a
6. Real SID on Board revisions 1.3 (U64 Elite) and 1.4
7. Multi Button
7. User Interface.
8. File browser
9. F2 - configuration menu
10. F5 - Tool / Command menu
11. Clock settings / RTC
12. Setting up the MultiSID configuration on the Ultimate 64
14. Hardware
15. LED Strip
17. LED Header
18. LED options
19. Example with two separate LED's / Example with duo LED
21. Audio/Video Connector
22. Build your own RGB Scart cable
23. SID-TAP header
24. HDMI port
25. USER PORT
27. Ethernet port
28. Network settings
29. USB ports
29. Data Streams
35. Ultimate SID player auto configuration

Getting Started

What do I need to get started?

Prerequisites

- Ultimate 64 motherboard
- Power adapter 12V DC (included)
- A case to put your Ultimate 64 in
- A C64 keyboard
- HDMI cable or standard C64 video cable
- USB Pen drive
- Kernal ROM (extract from your Commodore 64 or download online)
- Basic ROM (extract from your Commodore 64 or download online)
- Char ROM (extract from your Commodore 64 or download online)

Optional

- One or two SID chips
- Keyboard risers (for the C64C model case)

Installing

Mounting the Ultimate 64 in a Case

If you have a spare Commodore 64C case, or bought a new case (which is the C64C model as well), you will need keyboard risers. Some of these have loose metal keyboard risers already in them, others have them mounted with pop rivets to a metal shield. If you have the latter, it is recommended to drill the pop rivets to get the risers loose, otherwise they will interfere with the keyboard connector on the Ultimate 64.

A few C64C have the keyboard clipped onto the top of the case. For those you don't need keyboard risers.

If you buy a new case online, be aware that you will need a keyboard as well.

If you have an older case (the breadbin model), the keyboard is mounted in the top of the case, and therefore you don't need keyboard risers.

1. Mount the risers to your keyboard. The left can be farthest to the edge, but the right riser you have to mount a bit to the left of the right edge, so it won't interfere with the keyboard connector.
2. There are some places online where you can find risers that are supposed to fit perfectly with the Ultimate 64. Here is one model that should work:
<https://www.thingiverse.com/thing:3051450> You can 3D-print them yourself or ask someone to do it for you. These can be screwed directly onto the bottom of the case as well.
3. Set the keyboard aside for now.

4. Now gently slide the Ultimate 64 board in your case, and fit it to the right side of the case first. The joystick connectors should slide in easy.
5. Push it gently against the back of the case, and let the connectors on the back of the board fit through the holes.
6. Adjust the board so you can see the mounting holes on the bottom of the case through the Ultimate 64 board mounting holes.
7. Fasten the board with suitable screws. Don't tighten too much.
8. Put the keyboard in the case, and connect it to the connector to the right. It will only fit one way, as there are a key pin on the connector.
9. Attach the top of the case.

Accessories

If you have a 3D printer, or you know someone who has, you can find many nice accessories at thingiverse. Here are a few of them:

- [Ultimate64 Keyboard Mounting Brackets](#)
- [Ultimate64 & C64 Reloaded power connector cover](#)
- [Ultimate64 Multi-Button \(Power Button\) cap](#)
- [Ultimate64 board support clips](#)

Commercially available kits

If you do not have a 3D printer yourself, you may also want to take a look at these kits that are available from Corei64 by Thomas Christoph, or the Pixelwizard Retro Shop from Thomas Koch. Both offer professional kits that make mounting of the U64 easy. Both shops have put special pages together, just for the U64, where the items are listed:

- [All Ultimate64 accessories from Corei64](#)
- [All Ultimate64 accessories from Pixelwizard](#)
- [Installation kit for 64C Corei64](#)
- [Installation kit for 64C Pixelwizard](#)
- [Installation kit for breadbin Corei64](#)
- [New C64C cases in different colors Pixelwizard](#)

Other installation accessories, stickers and batches are available at <https://www.corei64.com/shop/> or at <https://shop.pixelwizard.eu/en>

Connecting your Ultimate 64

Connect your Ultimate 64 to the wall and your TV/monitor. If you start with HDMI, your TV/monitor have to be DVI capable over HDMI. You can change this to HDMI later.

Flashing ROM

Before you can use your Ultimate 64, you have to flash your Kernal and Basic ROM to the Ultimate 64. Power on and wait for picture, where it explains how to do.

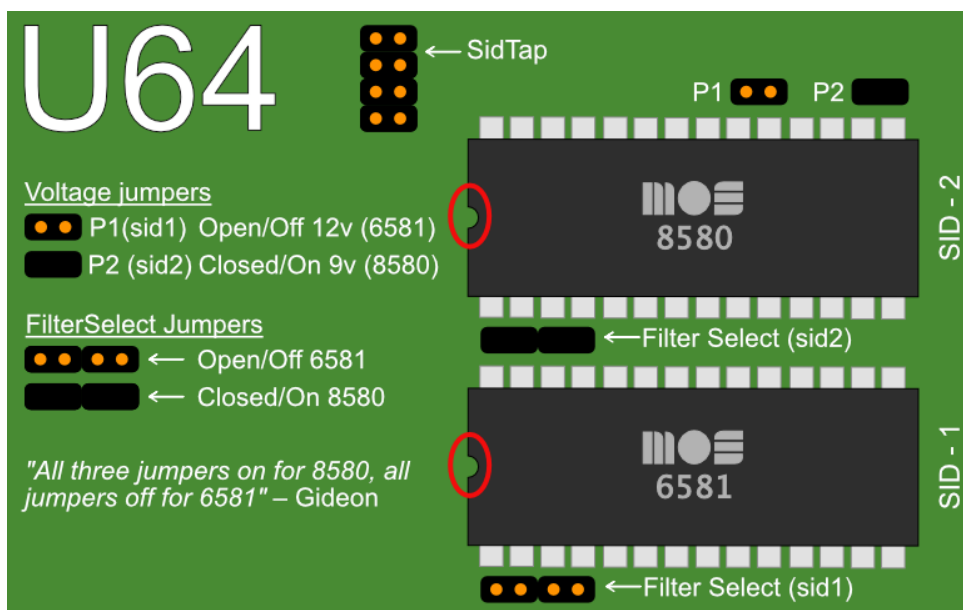
Basically you stick your USB drive in the back of the Ultimate 64 and push shortly on the power button to bring up the Menu.

The ROMs has to have the suffix `.bin` or `.rom`, as in `mykernel.bin`, `mybasic.bin`, `mychar.bin`.

1. Browse to your Kernal ROM image and press Enter.
2. Choose `Flash as Orig. Kernal ROM`.
3. Browse to your Basic ROM image and press Enter.
4. Choose `Flash as Orig. Basic ROM`.
5. Browse to your Character ROM image and press Enter.
6. Choose `Flash as Orig. Character ROM`.
7. Push `F5` and choose `Reboot C64`.

Real SID on Board revisions 1.1, 1.2 and 1.2a

If you're installing real SID chips, you have to set the jumpers accordingly. If you don't have or don't want to install SID chips you can use the built in 'UltiSID'.



Voltage Jumpers:
P1: SID 1 Voltage
P2: SID 2 Voltage
On/Closed: 9V (8580)
Off/Open: 12V (6581)

Filter Select Jumpers:
On/Closed: 8580
Off/Open: 6581

In Firmware revisions before V1.22, it is required to set your SID types in the Ultimate-II+ menu:

Short press on power button
F2
U64 specific settings
Set your types at
SID in socket 1
and
SID in socket 2

Setting the SID type in the configuration enables the bus access to the chip, and also allows the auto-configurator to choose the right chip when playing a SID tune with the provided player from Wilfred Bos.

Real SID on Board revisions 1.3 (U64 Elite) and 1.4

The newer board revisions are jumperless. This means that both the voltage as well as the filter caps are selected automatically, based on the inserted SID type. For this, firmware V1.22 or later is *required. Do not downgrade to earlier versions on jumperless boards!*

In order to make sure that 8580 chips are never given a higher voltage than they can handle, the socket gets disabled automatically whenever an 6581 chip (which needs 12V) is detected for the first time. When the popup appears “SID changed”, enter the configuration menu, and enable the socket in which the 6581 resides:

```
Short press on power button
F2
SID Sockets Configuration
Enable the sockets with options:
SID in socket 1
and
SID in socket 2
```

Once the chip in the socket is recognized as 6581 AND the socket is enabled, the 12V is applied.

Pull Down Resistor

Note that in the original schematics of the C64, a so called “pull-down” resistor of 1 k Ω is placed from the audio output pin to ground. This resistor is present on the 6581-based boards, but absent on the 8580-based boards. On the jumperless U64 boards this resistor is also enabled and disabled automatically. However, you may override this in the same configuration menu. The influence is limited.

Multi Button

The Ultimate-64 has one button on the side of the board, which is placed on the original position of the power button.

This button is called the MultiButton, because it handles several functions.

Function	Description
Power on	When the machine is off, press the button briefly to switch it on
Power off	Press the button for 4 seconds to switch off the machine.
Reset	Press the button for about 1 second to reset the machine.
Menu	Press the button briefly to enter the File Selection (Ultimate) menu.
Freeze	Hold the button and briefly tap on the RESTORE key to get into the freeze function of the selected utility

User Interface

The Ultimate-64 provides a user friendly menu driven interface, which can be accessed by pressing the multi button (power) briefly.

There are two modes in which this user interface can operate:

1. Freeze mode
2. Overlay mode

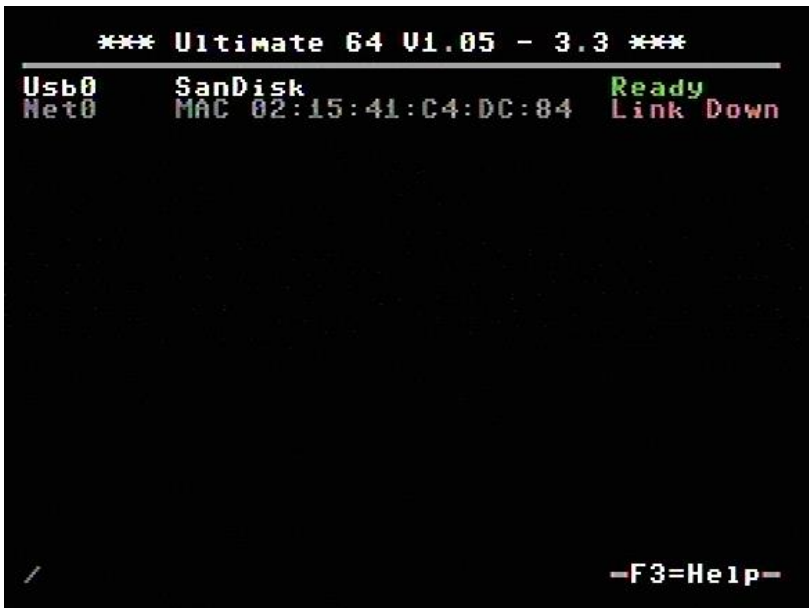
In Freeze mode, pressing the power button briefly will interrupt the currently running program on the C=64 and show the menu. Leaving the menu will resume the C=64 gracefully. This mode of operation works on both PAL and HDMI outputs.

In Overlay mode, pressing the power button briefly will show the menu in an overlaid area on the screen. The C=64 will simply continue running, but the keyboard is switched over to the Ultimate application, as long as this overlay screen is visible. Pressing run-stop in the file browser, or pressing the power button again briefly will make the overlay disappear.

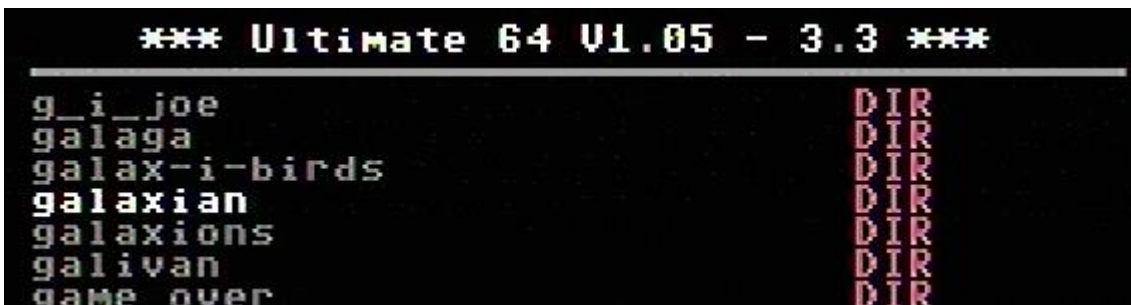
The user interface has three main functions:

1. File browser
2. Configuration of the Ultimate 64 (F2)
3. Tool menu (F5)

File browser



The menu starts with showing the available network interfaces, and the storage devices that are attached to the cartridge. By default, it will only show the built-in Ethernet port. When USB-sticks are inserted, one or more items will appear in this list. This screen is the 'root' of the file system.



Use the cursor keys to navigate through the file system and select your file (disk image) to mount. The highlighted line shows the current selection.

F2 - configuration menu



Use the UP/DOWN cursor keys to navigate and RIGHT to enter the desired configuration screen. Once inside a settings screen, the behavior of the keyboard is slightly different:

Key	Function
CRSR up/down	Move the cursor (highlighted line) up/down
CRSR left/right	Increase or decrease a setting, cycling through the available options.
DEL	Go one level up
Return / Space	For string fields: bring up a text entry box For enumerated fields: bring up a context menu with the available options
Run-stop	<i>Leave</i> the configuration menu and saves the new values.

Note that some values will only take effect after reset or a power cycle of your C=64. However, the main menu provides a way to restart the 1541 and the C=64 with the new settings.

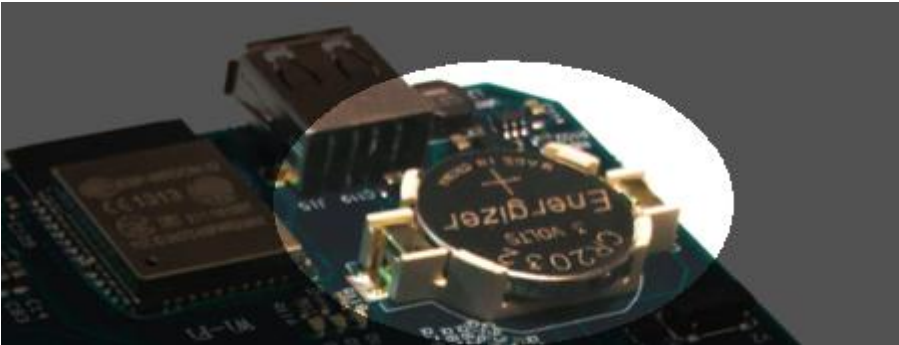
F5 - Tool / Command menu

```
*** Ultimate 64 V1.05 - 3.3 ***
g_i_joe DIR
galag
galax Save EDID to file
galax Create D64
galax Create G64
galiv Flush Printer/Eject Page
game_ Reset IEC and Printer
games UltiCopy 8
games UltiCopy 9
gaplu Reset C64
garfi Reboot C64
garfi Power OFF
gary_ Save REU Memory
gary_ Create Directory
gates Sample tape to TAP
gatew Capture save to TAP
gaunt Reset 1541 Drive A
gaunt Insert blank disk in drive A
gazza_2 DIR
gazzas_super_soccer DIR
gba_championship_basketball DIR
/usb0/c64 d64/g/ -F3=Help-
```

The purpose of the tool menu is to command the Ultimate application to do something that is not related to the selected entry in the file browser menu. For instance, creating a new disk image or directory is such an 'action' that can be performed using the tool menu. Some options might appear / disappear, depending on the state of the application, or based on whether the current file browser directory is writable. For instance, `Create Directory` is available only when the file browser is in a location where directories can be created.

More information about each of the available options can be found in the description of the Ultimate Application itself, with all of its features.

Clock settings / RTC



The U64 has an internal clock that is powered with a battery to keep track of the date & time.

See hardware RTC documentation

At this config screen date & time can be changed with the following options:

- Year
- Month
- Day
- Hours
- Minutes
- Seconds
- Correction
 - This corrects the deviation of the internal clock (RTC)

Setting up the MultiSID configuration on the Ultimate 64

This page explains how to use the visual SID address editor to set the UltiSID in MultiSID mode, giving you four or even eight SIDs.

In the following example, it is assumed that the settings are as follows:

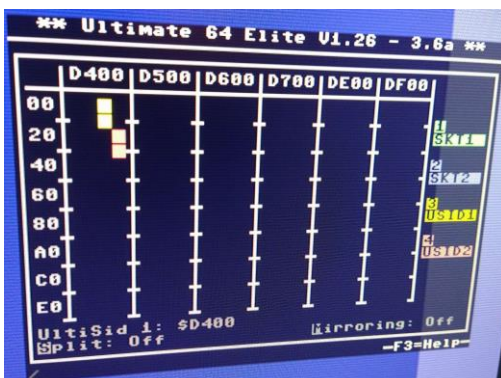
- SID Socket 1 Address: Unmapped
- SID Socket 2 Address: Unmapped
- UltiSID 1 Address: \$D400
- UltiSID 2 Address: \$D420
- UltiSID Range Split: Off
- Auto Address Mirroring: Enabled

First step is to enter the visual editor and note that UltiSID 1 is mapped at \$D400, UltiSID 2 is mapped at \$D420 and auto mirroring is on, just like the settings we started with. Address mirroring means that the appearance of the available SIDs are mirrored in the address space, such that they fill up the \$D400-\$D7FF range. Also pay attention that the lower line shows “Socket 1”, which means that the controls (cursor and split) now have effect on socket 1.

We want to change the UltiSID settings, so we press “3” on the keyboard; the highlighted number next to the “USID1” label.



Press “M” to turn off auto mirroring, to have a better view of the settings. Note the lower line says “UltiSid 1”, which we will modify.



When pressing “S”, the split function gets enabled. The line below shows the split is active on one address line, being A6. This means the UltiSIDs (both of them) get split in two, and the second part of it gets enabled with address line 6 set (thus \$D440). Note that this instance gets a “B”, indicating the second half of the split UltiSID1. The same goes for UltiSID2, which now appears at \$D460.



Similarly for a split on A7; the second half appears with an offset of \$80.



And last but not least, split on two address lines, causing the UltiSID to be split into four SIDs. In this case A7 and A8, such that the second part appears at \$D480 (A7=1), the third part at \$D500 (A8=1), and the fourth part at \$D580 (A7=1, A8=1). Also note the letters A, B, C and D.



Since the split settings are shared for both UltiSIDs, UltiSID2 also gets split in the same way. But because its base address was set at \$D420 (A5=1), the splits appear also at offsets where A5=1, thus \$D420, \$D4A0, \$D520, \$D5A0. In total 8 SIDs, thus 24 voices and 8 independent filters.

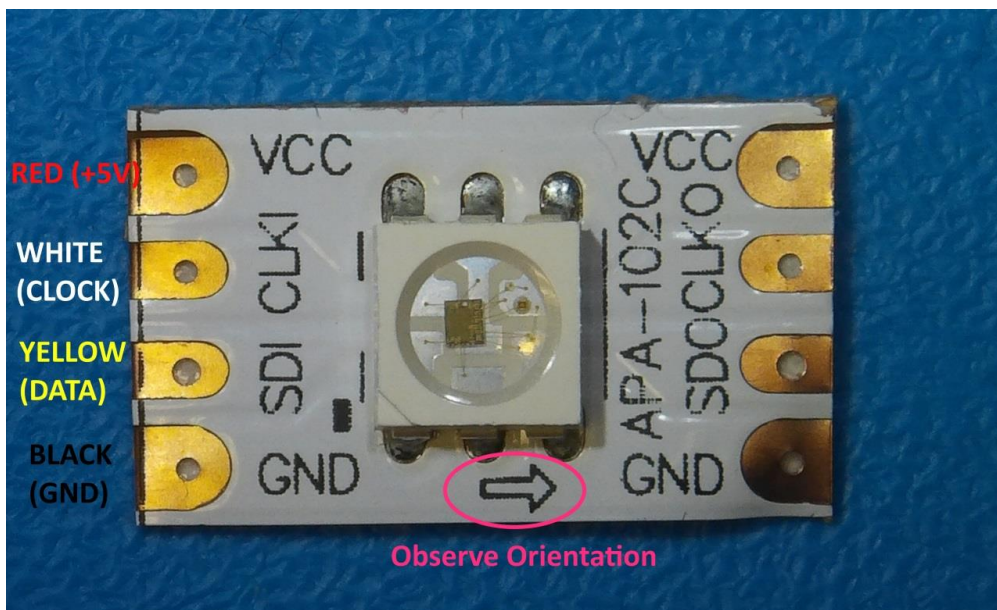
Hardware

- [Attaching a LED strip](#)
- [LED settings](#)
 - [Pinout](#)
 - [LED options](#)
 - [What does !\(option\) mean?](#)
 - [Example with two separate LED's](#)
 - [Example with duo LED](#)
 - [Technical information](#)
- [Audio/Video Connector Pinout](#)
 - [Pinout](#)
 - [Audio](#)
 - [Video](#)
 - [Build your own RGB Scart cable](#)
- [SID-TAP Header Pinout](#)
- [HDMI port](#)
 - [Resolution of the HDMI output](#)
 - [Frame delays of the digital HDMI port](#)
- [User Port Header Pinout](#)
 - [Attaching the User Port Adapter](#)
- [Ethernet port](#)
 - [Network settings](#)
 - [RR-net compatibility](#)
 - [The Ethernet port is used natively by the firmware](#)
 - [Ultimate-related BBS and IRC clients](#)
- [USB port](#)

LED Strip

Attaching a LED Strip

Using [firmware version 1.10 and later](#), it is possible to attach a programmable LED strip to your Ultimate 64 board. The supported LED type is APA102C, with is a 5V only device with built-in PWM controller for each color, red, green and blue. At the time of writing, the firmware supports up to 64 LEDs, of which 24 are currently used. On a strip with 60 LEDs (IP30) per meter, this corresponds to 40 cm of length, which is roughly the width of the machine. Alternative are SK9822 strips.



For an easy connection of the LED strip, it is recommended to connect four wires / leads to it, with a single pin socket, like the ones shown below. These wires can be obtained anywhere, and are sold -for example- as Raspberry PI jumper cables.



The following connections need to be made:

Pin	Name	Description	User Port
1	VCC (+5V)	Red wire	Pin 2
2	CLKI	White wire, Clock input	Pin 13
3		SDI	Yellow wire, Data input
4		GND (0V)	Black wire
			Pin P
			Pin 1, 12, or N

Note, the wire colors are just suggestions. Obviously it will work with other colors as well. But for the sake of not making mistakes, please at least use black and red for ground and +5V respectively.

Please refer to the drawing of the user port header, in order to see where to connect your wire leads. [USER PORT](#).

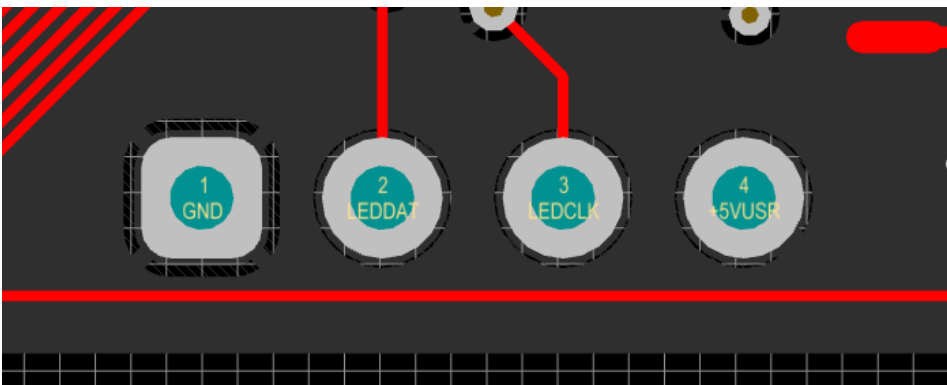
LED pinheader P8

Beginning with hardware revision 1.4 you'll find the new LEDstrip pin header P8 to the left under the User Port. This can be used to attach an APA102C LED strip and simultaneously use the User Port for other purposes.

The pin header is defined as follows:

P8 —————
Pin: 1 2 3 4

GND DAT CLK 5V



W A R N I N G ! Please always use extreme caution!

Please be extremely careful not to shorten the 5V to ground! It will burn a ferrite bead on your board! Also shorting 5V to pin 3 or any other FPGA pin will likely damage the FPGA!

LED Header



Pinout

Pin	Description
1	Power
2	Ground
3	Drive activity

With firmware 1.18 (and above) the pin assignment is not static any more.

LED options

```

*** Ultimate 64 V1.18 - 3.4c ***
-----
Vol Socket 2                0 dB
Vol Sampler L                0 dB
Vol Sampler R                0 dB
Vol Drive 1
Vol Drive 2
Vol Tape Read
Vol Tape Write
Pan UltiSID 1
Pan UltiSID 2
Pan Socket 1
Pan Socket 2
Pan Sampler L
Pan Sampler R
Pan Drive 1
Pan Drive 2
Pan Tape Read
Pan Tape Write
LED Select Top
LED Select Bot
Speaker Volume ($p)

On
Off
Drive A Pwr
DrvAPwr + DrvBPwr
Drive A Act
DrvAAct + DrvBAct
DrvAPwr ^ DrvAAct
USB Activity
Any Activity
!(DrvAAct)
!(DrvAAct+DrvBAct)
!(USB Act)
!(Any Act)
IRQ Line
!(IRQ Line)

a--F3=Help-

```

LED Options

Option	Description
On	LED always ON
Off	LED always OFF
Drive A Pwr	Power indicator of drive A
DrvAPwr + DrvBPwr	Power indicator of drive A and drive B combined
Drive A Act	LED ON when there is activity on drive A
DrvAAct + DrvBAct	LED ON, on activity of drive A and B combined
DrvAPwr ^ DrvAAct	Drive power LED XOR drive activity. This results in an inverse drive activity, with similar dimming as the power LED has.
USB Activity	LED ON, on USB activity
Any Activity	LED ON, on any activity (excluding tape)
!(DrvAAct)	Inverse of drive A activity, so the LED will turn OFF on drive activity instead of ON
!(DrvAAct+DrvBAct)	Inverse of drive A and B combined, so LED will turn OFF on activity instead of ON
!(USB Act)	Inverse of USB Activity, so the LED will turn OFF on USB activity instead of ON
!(Any Act)	Inverse of Any Activity, so the LED will turn OFF on any activity instead of ON
IRQ Line	LED is ON when IRQ line is high (inactive)
!(IRQ Line)	Inverse of IRQ Line, so the LED is ON when IRQ line is low (active)

With the introduction of firmware 1.18 the behaviour of the LED can be changed.

What does !(option) mean?

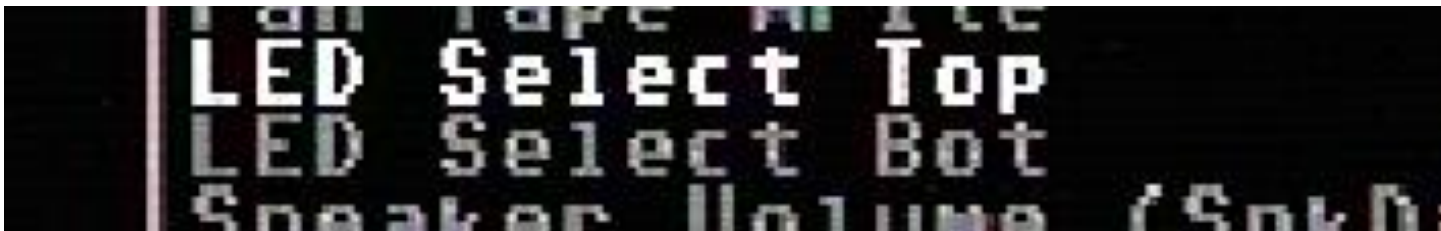
As we can see in the table above, there are setting options (functions) that are noted as !(function). !(function) means it is an inverse outcome of a function.

We take a closer look at the function **DrvAAct+DrvBAct**. This means when there is drive A or drive B activity (combined) the LED will turn ON. So **!(DrvAAct+DrvBAct)** will do the opposite (inverse) of the function **DrvAAct+DrvBAct**, so the LED will turn OFF when there is activity on drive A or drive B.

Example with two separate LED's

We are using a red and green LED, red will be the power LED (power) and green will be drive activity (drive). The power wire to the red led is connected to the Top pin (1) and the green led is connected to the Bottom pin (3), and of course the ground of both LED's is connected to the ground pin (2).

Because there are two separate LED's we want that the power is always ON, and the drive will only turn ON when there is activity.



So we use the following setting:

- Power => LED Select Top: **On**
- Drive => LED Select Bottom: **DrvAAct+DrvBAct**

To swap the color assignment, just swap the settings between Top and Bottom.

Example with duo LED

We are using a duo LED that has red and green combined in one package, red will be the power LED (power) and green will be drive activity (drive). The power wire to the red led is connected to the Top pin (1) and the green led is connected to the Bottom pin (3), and of course the ground of the LED is connected to the ground pin (2).

Because there is only “one” LED package we want that power will turn OFF when there is drive activity, or else the two colors will be mixed to yellow (red+green).



So we use the following setting:

- Power => LED Select Top: **!(DrvAAct+DrvBAct)**
- Drive => LED Select Bottom: **DrvAAct+DrvBAct**
- To swap the color assignment, just swap the settings between Top and Bottom.

Technical information

Power LED

The power pin is provided with a 260KHz PWM signal, the PWM signal makes the LED less bright. The U64 has an on-board 220 ohms series resistor to lower the current, the supplied voltage 3.3V.

Drive activity LED

PWM	Description
Off	Drive Off
1/3	Drive On
2/3	Drive On + Floppy Mounted
3/3	Drive On + Floppy Mounted + Motor running

PWM = pulse width of the pwm signal.

Audio/Video Connector



Note: The connector used is a full size 8-pin 262° DIN, also known as the ‘horse-shoe’ type. A 270° DIN plug does *NOT* fit without violence.

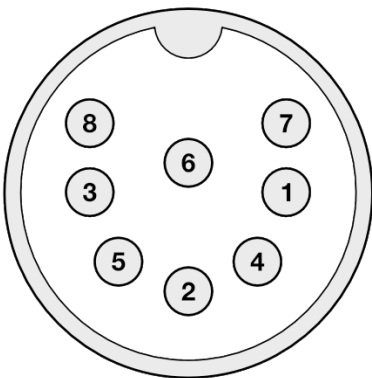
The 8-pin DIN (audio/video) connector can output several different video signals, for each type you need a special assembled cable.

Output signals

- Composite (CVBS) (PAL)
- S-Video (PAL)
- RGB (PAL)

The Composite and S-Video cables are the same as you use for the original Commodore 64.

Pinout



Pin	Modulated Color	RGB Mode
1	Luma (Y)	Green
2	Ground	Ground
3	Audio Out Left	Audio Out Left
4	Composite (Y+C)	Red
5	Audio Out Right	Audio Out Right
6	Chroma (C)	Blue

Pin	Modulated Color	RGB Mode
7	Not used	CSync
8	Not used	Fast Switching

Audio

The audio outputs have consumer electronic line levels (Line Out) and need to be connected to an external amplifier.

Video

The video signals on pin 1, 4 and 6 have a source impedance of 75 Ohms, with standard 1V-peak video levels. In the modulated color mode, Luma and Composite contain sync information (-0.3V), while Chroma does not. The video blanking of the VIC output has been corrected, and is now in accordance with specification.

The PAL output is non-interlaced, with a field rate of 50.2 Hz, and line rate of 15.639 kHz.

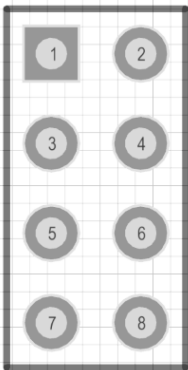
The Fast Switching output is a 1.5V fixed 'high' output signal (when terminated with 75 Ohms), that is used on SCART to enable RGB mode. The CSYNC output is a sync-only output that is needed on the SCART signal as a replacement of the composite video or luma signal. In RGB mode, there is *no* sync information in either R, G or B; i.e. no 'sync-on-green'.

Build your own RGB Scart cable

Pinout U64 & SCART

Pin	U64 video port	Pin	SCART
1	Green	11	RGB Green
2	Ground	4	and pin 5,9,13, 21(Shell)
3	Audio Out Left	6	Audio Input Left
4	Red	15	RGB Red
5	Audio Out Right	2	Audio Input Right
6	Blue	7	RGB Blue
7	Csync	20	Composite video input
8	Fast switching	16	RGB Selection, 1-3V = RGB

SID-TAP header



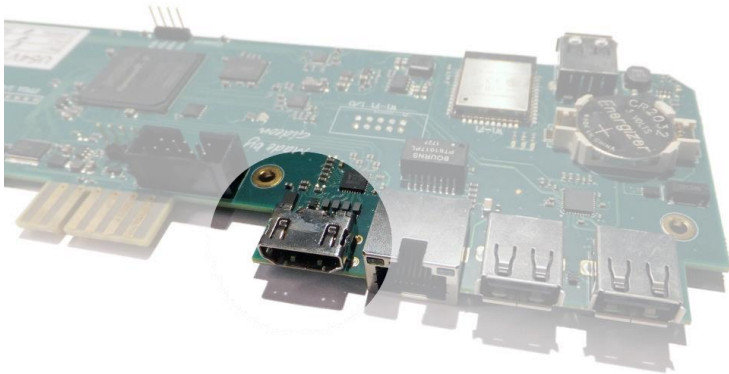
Function	Pin	Pin	Function
SID 2 OUT	1	2	Ground
Ground	3	4	SID 2 IN
Ground	5	6	SID 1 IN
SID 1 OUT	7	8	Ground

The SID-TAP header provides a direct (analog) output from the installed SID chips. This header is meant for the purists, who do not want the SID signals to be processed by the U64 board. It is possible to attach a 3.5mm jack plug directly to the SID OUT signals (left column of the header) and use it as a line-out source.

The SID inputs are not often used. However, if you like to use them, you can use the same trick, and attach a 3.5mm jack plug to the right side of the header. This connection provides a line-in connection to your SIDs. This has *not* been functionally verified!

Note that the arrow on the board points at pin 2. In other words, the image and table above are the top view of the header.

HDMI port



The HDMI port can be connected to a modern display (eg. television), so you can use the Ultimate-64 in the future.

Resolution of the HDMI output

The output resolution is 720 x 576 at 50 Hz (for the PAL version), also noted as 576p50. This is a standard resolution that all HDMI capable screens are supposed to support, according to the CEA-861 standard. However, to meet the exact timing of a C64, the timing parameters of this video mode are slightly modified. I have not come across screens that do not accept the generated HDMI signal.

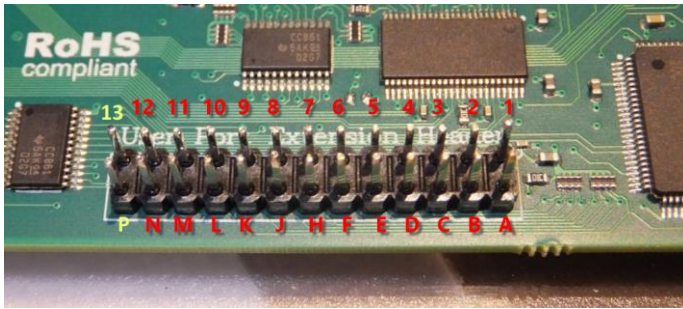
Frame delays of the digital HDMI port

None. There is no frame buffer, so there is no need to worry. Some screens, tho, especially non-gaming screens may introduce some latency. I observed this with a 4K screen that actually resampled the output to just 30Hz. I am pretty sure displays do not have this problem. Because there is no frame buffer, scrollers and such are perfectly smooth. If the TV renders the image at the speed it is sent over the HDMI link everything will be smooth. If HDMI is used for output, there is also output generated on the PAL output (display connector).

In the configuration menu you can change several setting regarding this port.

- U64 Specific Settings
 - HDMI Scan lines
- User Interface Settings
 - Interface Type (overlay mode)

USER PORT



User Port Extension Header

Pin	Name	Description	Special Purpose
1	GND	Ground	
2	+5V	+5 VDC (200 mA max)	
3	/RESET	Reset, will force a Cold Start. Also a reset output for devices.	
4	CNT1	Counter 1, from CIA #1	
5	SP1	Serial Port 1, from CIA #1	
6	CNT2	Counter 2, from CIA #2	
7	SP2	Serial Port 2, from CIA #2	
8	/PC2	Handshaking line, from CIA #2	
9	ATN	Serial Attention	
10	12VDC	12VDC (Max 500mA)	
11	12VDC	12VDC (Max 500mA)	
12	GND	Ground	
13	USRCTL0	FPGA Pin, <i>NOT</i> 5V tolerant!	LED Clock
A	GND	Ground (RS232: GND)	
B	/FLAG2	Flag 2 (RS232: RxD=Both B+C)	
C	PB0	Data 0 (RS232: RxD=Both B+C)	
D	PB1	Data 1 (RS232: RTS)	
E	PB2	Data 2 (RS232: DTR)	
F	PB3	Data 3 (RS232: RI)	
H	PB4	Data 4 (RS232: DCD)	
J	PB5	Data 5	
K	PB6	Data 6 (RS232: CTS)	
L	PB7	Data 7 (RS232: DSR)	
M	PA2	PA2 (RS232: TxD)	
N	GND	Ground (RS232: GND)	
P	USRCTL1	FPGA Pin, <i>NOT</i> 5V tolerant!	LED Data

Applies to: Ultimate 64

Note that the row 1-12 corresponds to the upper row of the User Port card edge fingers on a C64, and that the row A-N corresponds to the bottom row of the User Port card edge fingers.

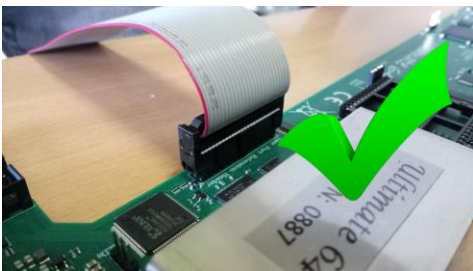
Pin 13 and P, the two left most pins on the header, are added for add features to the user port. One of the intended usages of these two pins is the generation of the “9 V~” on the user port extension board.

There is one other use of pin 13 and P, which is the control of a LED Strip. In this case, pin 13 the LED Clock and pin P is the LED Data, assuming a LED strip that is based on APA102C.

In the LED Strip configuration menu, the use of pind 13 and P can be chosen. When the LED Strip is set to ‘off’, the 9 V~ generation is enabled.

Attaching the User Port Adapter

The user port adapter accessory should be connected using a 26-pin ribbon cable. Note that there is only ONE correct way to connect it, and other ways may result in damage of the U64 itself or to the accessory.

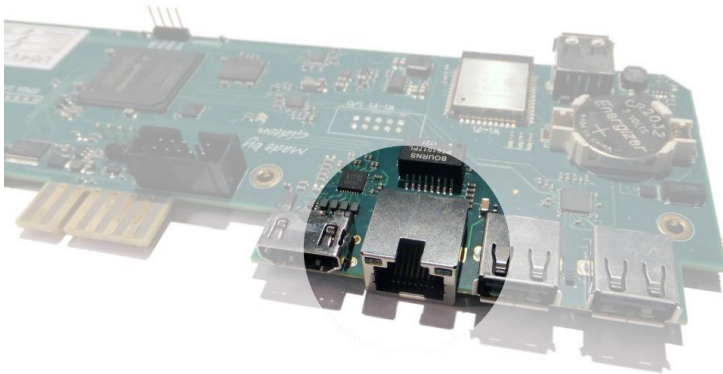


Pay attention to the orientation!! The red wire should line up with pin 1!!!

Please make sure the connector is not shifted a whole row, or to the left or to the right:



Ethernet port



The ethernet port can be connected to a switch to communicate with your Ultimate over your own local network.

The Ultimate 1541-II+ and Ultimate 64 have an integrated ethernet port.



Ultimate 1541-II and compatible usb ethernet adaptor

The Ultimate 1541-II needs a compatible USB Ethernet adaptor that is connected to the USB port of the Ultimate 1541-II. Any USB2LAN adaptor that uses an AX88772/-A/-B chip should work.

Please note that the compatible USB Ethernet adaptors will ONLY be stable on the Ultimate 1541-II if the firmware < 3.2. They will work fine on the Ultimate 1541-II+ and Ultimate 64.



If your Ultimate product has an ethernet port, you can see this when entering the Ultimate menu. The picture above shows a **Net0** port, this indicates ethernet is accessible.

Network settings

Within the network settings menu the ethernet port can be configured as needed, so it matches with your local network settings.

The Network settings menu is only visible if a ethernet port is available.

Menu name	Explanation	Options (bold default)
Use DHCP	Get IP settings from local DHCP Server on your network. Most home routers provide these settings by DHCP. If this is not the case please consult your internet/network provider.	Disabled Enabled
Static IP	Set custom IP settings, DHCP has to be disabled. <i>This field can be left empty if DHCP is enabled.</i>	Inputbox
Static Networkmask	Set custom networkmask, DHCP has to be disabled. <i>This field can be left empty if DHCP is enabled.</i>	Inputbox
Static Gateway	Set custom Gateway, DHCP has to be disabled. <i>This field can be left empty if DHCP is enabled.</i>	Inputbox
Hostname	Set unique network hostname to identify your Ultimate product.	Inputbox

RR-net compatibility

Some of you may be familiar with the RR-net solution that brings Ethernet to the C-64. Currently, the built-in Ethernet port does not provide RR-net compatibility.

The Ethernet port is used natively by the firmware

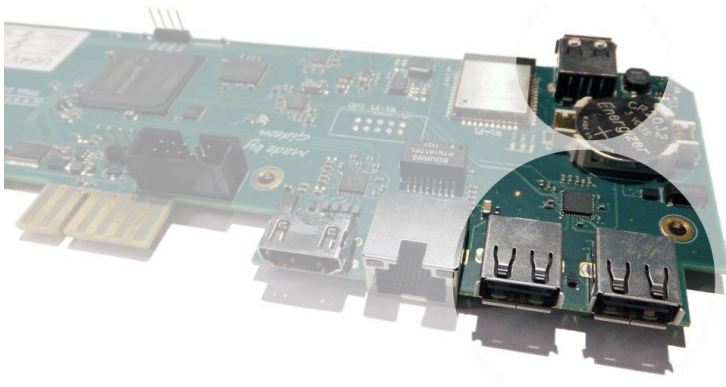
There is some primitive support for file-transfer using FTP (needs improvement), and it is possible to connect to the Ultimate using a VT-100 terminal program on the Telnet port (port 23). This gives the possibility to control the machine remotely, and swap disks without actually interrupting the program running on the C-64.

Ultimate-related BBS and IRC clients

Since RR-net compatibility is not yet provided, some specific Ultimate software has been developed for accessing BBS and IRC networks. [Ulimite Lib](#) is a library (written by **Scott Hutter**) that gives easy access to the full TCP/IP stack that the Ultimate provides. This library comes along with two sample applications: [Ulimite Chat](#) and [Ulimite Term](#), this one co-developed with **Francesco Sblendorio**.

Applies to: Ultimate 1541-II, Ultimate 1541-II+, Ultimate 64

USB ports



The Ultimate-64 supports most USB sticks and Flash card readers out of the box. It also supports USB 2.0 Hubs. It is recommended to use powered USB hubs only (with an external power supply).

USB 1.1 Hubs are not supported.

There are three USB 2.0 (**no USB 3!!**) ports available on the board; two on the back and one on the “internal” side of the board/case.

Usb0	External USB (next to NIC)
Usb1	External USB (edge of PCB)
Usb2	Internal USB

Data Streams

Introduction

Context

The Ultimate-64 is capable of streaming real time data over its built-in Ethernet port. Currently, there are two different types of streams that the machine can output:

- VIC Video Stream
- Audio Stream

This page describes these features in more detail.

Goal

This feature is intended for debugging, regression testing and analysis. However, there are more uses for it; such as making pixel-perfect recordings of a demo, or a digital audio recording without

conversion to analog first and sampling it on a PC. Or, even a real time share over the internet to a friend!

Networking Background

The network streams are based on UDP. This means that they are connection-less, and can be captured off the network by the addressed node. UDP does not guarantee delivery, so it may be possible that packets get lost. However, on a local network this usually does not happen.

Aside from its simplicity, UDP has another advantage: it can be used to send data to multiple receivers at once. This is called ‘multicast’. The Ultimate is capable of sending the datastreams in unicast mode (to one specific IP address), in multicast mode (to a multicast IP address), or in broadcast mode. When the stream is transmitted in multicast mode, the receiver needs to ‘join’ the multicast group. Note that this will only work when the switch supports IGMP snooping, otherwise the switch will broadcast the data over all of its network ports; thus treating the packets as broadcast packets.

When the switch supports IGMP snooping, it will ‘see’ the network packet that is generated by the client who wants to receive the stream when it joins the multicast group. Such a request is sent using the IGMP protocol. Once the switch ‘sees’ this packet, it will enable the transmission of the requested stream to the physical port on the switch that requested the data (the port the PC is connected to). No other ports will receive the data, so the network is not flooded.

Unicast Example

When your switch does not support IGMP snooping, the use of unicast is recommended. Especially when there are other users of your local network, that are behind WiFi links, it is not recommended to flood the network and get angry looks from everyone else in your household.

Unicast is the simplest form. Let’s say your PC is at 192.168.0.100; either statically configured or obtained by DHCP, and your Ultimate has the IP address 192.168.0.64; again either statically configured or obtained from the DHCP server. You can then simply instruct the ultimate to stream the data to IP address 192.168.0.100.

The receiving software will open a socket, listening to a specific UDP port number. It is necessary to send the stream to this specific UDP port number, otherwise the stream cannot be received.

Multicast Example

When using multicast, a specific range of IP addresses is used for transmission: 224.0.0.0 to 239.255.255.255. Some of these addresses are reserved for a specific type of communication, like IGMP, MDNS, and such.

Let’s say the Ultimate is configured to send the stream to 239.0.1.64. This is in the free range for multicast on the local network (LAN). The PC application starts by telling the network infrastructure and its own routing tables that it wants to listen to 239.0.1.64. From that moment on,

the stream is 'seen' by the PC and the application receives the data. Once the PC application is done, it 'leaves' the multicast group, and the PC ceases to receive the data.

Note that the IGMP switching is based on IP address. Therefore it is not possible to use the UDP port number to differentiate which stream to receive and which not. Use different multicast IP addresses if you like to receive more than one stream.

Starting and Stopping Streams

The available streams can be started and stopped in different ways. When the machine is turned on, it does not transmit any stream.

Using the menu

A stream can be started and stopped from the "action menu", which is brought up by pressing 'F5'. The start/stop options appear in that menu. When selected, the Ultimate will ask for the IP address to send the stream to. The default IP address is configured in the configuration screen "Data Streams". Note that changing the IP address in the configuration screen *does not* start or stop a stream; it is only the place where the last entered data is kept for later reuse.



The format of the destination is either an IP address, or a valid DNS name. The port number can be given as well, by adding it after a colon (:).



Examples:

Destination string	Meaning
192.168.0.119:11000	unicast address on the local network, port number 11000
myserver.com	unicast address, using DNS and default port number
myserver.com:4567	unicast address, using DNS and specific port number
239.0.1.64	multicast address, using default port number
239.0.2.77:64738	multicast address with port number specified

Using the TCP command interface

Using the TCP socket '64', the following commands have been added to start and stop streams.

- Enable output stream: FF2n
- Disable output stream: FF3n

whereby 'n' is the stream number (see below)

The generic command structure is:

- <command word, little endian>
- <command parameter length, little endian>
- <command parameters>.

The Disable command does not take any parameters. The enable command can have two parameters; neither of them are required.

- Parameter 1: Duration of the stream enable (0 = forever)
- Parameter 2: Destination (string).

The duration is given in system ticks, which come at an interval of 5 ms.

So to elaborate on the commands:

To enable stream 0 for one second (200 ticks), to the pre-configured destination address, the bytes sent over the TCP control port are:

```
20 FF 02 00 00 C8
```

To enable stream 0 indefinitely to '192.168.0.119', you'd send:

```
20 FF 0F 00 00 00 192.168.0.119 (the latter part being ASCII)
```

-> Command length = string length + 2 bytes.

To disable stream 0, you send:

```
30 FF 00 00 (length of parameters is 0)
```


Available streams

VIC Video Stream (ID 0)

The VIC Video Stream is the stream that presents the active part of the video output of the VIC. The ID of this stream is 0.

Each UDP datagram contains a header and pixel data. The header is as follows:

- Sequence number (16 bit LE)
- Frame number (16 bit LE)
- Line number (16 bit LE). If this is the last packet of the frame, bit 15 will be set.
- Pixels per line (16 bit LE), always 384
- Lines per packet (8 bit), always 4
- Bits per pixel (8 bit), always 4
- Encoding type (16 bit), always 0 for now. May be '1' later for RLE encoded data.

Example packet:

The following packet contains data of the third frame, line 100 - 103, starting with dark blue:

```
A1 00 02 00 64 00 80 01 04 04 00 00 66 66 66 66 66 66 66 ...  
seq #|frm #|line#|width|lp|bp|-enc-|data.....
```

So in total, the pixel data is preceded with 12 bytes of header data. The header is followed by pixel data, coded in 4-bit VIC colors, little endian, thus nibble 3..0 first. 4 lines of 384 pixels, equals 768 bytes of pixel data. Total UDP datagram = 780 bytes.

To capture an entire frame, you first wait for a packet with bit 15 set of the line number, then capture until you get another packet with bit 15 set of the line number. You should then have 68 packets of 4 lines (totaling 272 lines). Note that the video is cropped to 384 x 272, and aligned with the reference images of the VICE test suite. The actual output to the screen is 400 x 288 in PAL mode.

In NTSC mode, the output stream carries 384 x 240 pixels. The actual output to the screen is 400 x 240.

Audio Stream (ID 1)

The Audio stream is taken from the output of the audio mixer. Thus, the stream received over the network contains the same data as the data sent over HDMI and back to the audio codec for conversion to the analog signal that is available on the DIN connector. The ID of this stream is 1.

The format of the audio stream is simpler than the video stream. The only two bytes that are sent as a header in front of the raw audio data is the sequence number of the packet, since the stream was enabled. This allows for detection of missing packets.

The sequence number is followed by 192 stereo samples in 16-bit signed, little endian format; left and right interleaved, starting with the left channel. Thus, the total UDP packet size is 770 bytes: 2 header bytes, and 192 times 4 bytes per sample:

```
00 00 fe ff 02 00 ff ff 03 00 fd ff 00 00 ...
-seq- left--right-left--right-left--right-...

01 00 ff ff 03 00 fd ff 00 00 03 00 ff ff ...
-seq- left--right-left--right-left--right-...

02 00 fd ff 03 00 fd ff 00 00 fe ff 02 00 ...
-seq- left--right-left--right-left--right-...
```

The sample rate of the stream is close to 48000 Hz. It depends on the video mode. The actual sample rate for PAL is:

$(F_c * 16/9 * 15 / 77 / 32) = 47983 \text{ Hz}$ ($F_c = 4433618.75 \text{ Hz}$) (48kHz: -356 ppm)

For NTSC, the audio clock is derived as follows:

$(F_c * 16/7 * 15 / 80 / 32) = 47940 \text{ Hz}$ ($F_c = 3579545.45 \text{ Hz}$) (48kHz: -1243 ppm)

Viewing and recording

Windows

Thanks to Martijn Wieland (TSB), there is a viewer for the stream for Windows. This tool can be obtained here: [TSB U64 Streamer](#).

Linux/Mac

Jimmy (DusteDdk) made a Linux viewer, more info check: [u64view](#)

This version can also be compiled on Mac OS.

Python script

Also there is an example script in the 'python' directory of the ultimate repository to show how grabbing of video data is done. This example script can be seen here: [grab.py](#)

Ultimate SID player auto configuration

What this does, is to select the correct socket when the internal player is started. The autoconfig selects the socket based on what type of SID is found in the socket and for what type of SID the tune was written. If the type of SID is not found, it may revert to UltiSID. This means that it will look for a 6581 if the SID header indicates this. This may lead to confusing results if the SID type installed is 8580.

If the SID types are correct, it will set the address(es) correctly, as well as the panning:

- Single SID tunes will pan to center.
- 2SID (Stereo SID) will pan to L3/R3.
- 3SID will pan to L4/Center/R4.

So it is a best-effort algorithm that works best for single SID tunes and you have one 6581 and one 8580 installed.

The SID autoconfig temporarily overrules the user configured audio mixer settings with the automatic configuration!