
Picoraptor Video Encoder Manual

AIRBORNE INNOVATIONS LLC

info@airborneinnovations.com

720-515-3720

20 November 2020

<i>Date</i>	<i>Description</i>
01/23/17	Release
05/27/17	Added advanced network configuration, multicast RTP / multicast VLC configuration, advanced RTP configuration, oem ethernet switch option
11/07/17	Added documentation for serial port usage and serial to TCP virtual com port software, raspberry pi video test source, and confirmed working cameras list.
06/04/18	Link updates
07/24/18	Added serial port pin 1 diagram
01/16/19	Added cooling info, multicast serial configuration, android and Pixhawk client info, MicroSD reliability, serial debug information
11/20/20	Note: Default IP Address changed to 192.168.168.168 Updated most old references to 192.168.1.x to 192.168.168.x. Youtube link added

Table of Contents

1 Introduction.....	3
2 Picoraptor.....	3
2.1 Picoraptor hardware interface.....	5
2.1.1 Power.....	5
2.1.2 Ethernet.....	5
2.1.3 Serial.....	6
2.1.4 Serial Dev Cable.....	7
3 Quickstart guide.....	8
3.1 New Default IP Address.....	8
3.2 Videos / Youtube channel.....	8
3.3 PC Network Configuration (Wired).....	8
3.4 PC Network configuration (Wireless).....	9
3.5 Video streaming using VLC.....	10
4 Usage.....	11
4.1 Cooling.....	11
4.2 IP address.....	11
4.3 Configuration	11
4.4 Ffmpeg.....	11
4.5 Gstreamer.....	11
4.6 Android tablet interface.....	12
4.7 Pixhawk compatible clients.....	12
4.8 Hardware Decoders.....	12
4.8.1 Decoder configuration.....	12
4.8.2 HD-SDI decoder configuration.....	12
4.9 Serial port usage.....	13
4.9.1 Virtual Com Port Software.....	13
4.9.2 Multicast serial configuration.....	13
4.10 Using a Raspberry Pi as a video test source.....	13
4.11 Confirmed Working Cameras.....	13
4.11.1 Some notes about GoPros:.....	14
5 UDP / RTP streaming modes.....	14
5.1 Playing RTP video with VLC.....	15
5.1.1 Using RTSP to initiate the unicast or multicast RTP stream.....	15
5.1.2 RTP Unicast: SDP file to initiate the stream.....	15
5.1.3 RTP Multicast: SDP file to initiate the stream.....	15
5.2 RTP Multicast mode (one video source, multiple receivers).....	16
5.3 Using Multiple Simultaneous Encoders.....	16
5.4 Advanced Network Configuration.....	16
5.4.1 Changing IP address.....	16
5.4.2 Advanced network configuration changes (mac address).....	16
5.4.3 Default netconfig.txt file.....	16

5.5 Advanced RTP configuration.....	17
5.5.1 Default vidconfig.txt file.....	17
5.6 RTP Video + Hardware digital video output Decoder.....	18
6 Multiple Video Stream Configuration.....	18
6.1 Accessing Video Streams.....	19
6.2 Multistream Web Configuration API.....	20
7 Onboard Recording.....	21
7.1 MicroSD card compatibility.....	21
7.2 HTTP API to enable/disable recording.....	22
7.3 Commandline/startup script recording.....	22
8 FLIR Vue Compatibility.....	23
9 AerialPixels cables.....	23
10 Compact 5 Port Ethernet Switch.....	24
11 MicroSD Card reliability.....	25
12 Serial debugging / Linux console.....	25

1 Introduction

This document is a manual covering features and usage of the Airborne Innovations Picoraptor video encoder system.

Picoraptor is a HD video encoder that receives DVI video (720p/1080p), performs H.264 HD video compression, and outputs streaming HD video in RTSP/RTP and other formats.

The board footprint is 40x40mm. Picoraptor can also integrate well with our Airborne Innovations Picoradio OEM module to form a powerful, compact, and cost effective long range video and datalink system. Picoradio can even screw onto the bottom of Picoraptor to form a compact board stack (or mount Picoradio and Picoraptor separately), and Picoradio can provide power and communications for Picoraptor (as well as provide a long range bidirectional vehicle datalink).

Picoraptor (or Picocamera) can mate with our Picoradio board to form a miniature long range HD video link + command and control link (and with our optional SBUS module for manual control).

- Picoradio can be mounted in a stack with the Picoraptor board or can be mounted separately.
- Flat customizable cables from Aerialpixels mate very well with the system.
- Some customers use the FLIR Vue camera with FLIR's digital video module (which outputs in 720p).
- Multiple cameras can be connected to Picoradio for simultaneous video feed transmission.

2 Picoraptor

Picocamera is a low latency streaming IP encoder.

It mates well with Picoradio to form a long range capable combined IP camera+datalink system.

Specifications:

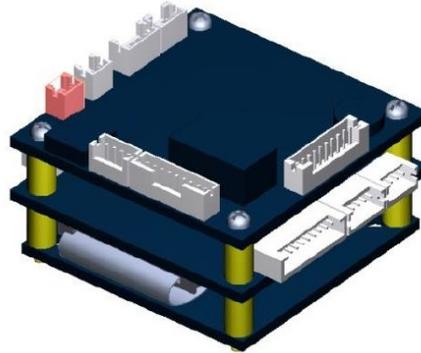
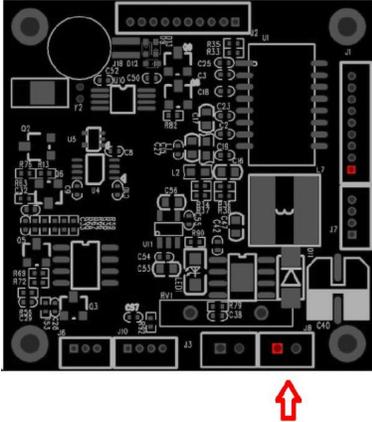
Weight:	~31g
Dimensions:	boards 40x40mm (approx 40x44 mm with connectors), 32mm high board stack with connectors
Interface:	DVI 1080p30 / 720p60 input (selectable output frame rate)
Power:	12V input

Output: Ethernet IP video stream, RTSP, RTP
Bitrate: Typical 3-5 Mbps @ 720P30, 5-8 Mbps @ 1080p30, up to 12 Mbps
Latency: Typically 100-200 Msec depending on mode and streaming method (excluding camera).

2.1 Picoraptor hardware interface

Note that the standoffs are M2, and standoff kits can be ordered from Amazon (brass or nylon standoffs are available).

2.1.1 Power

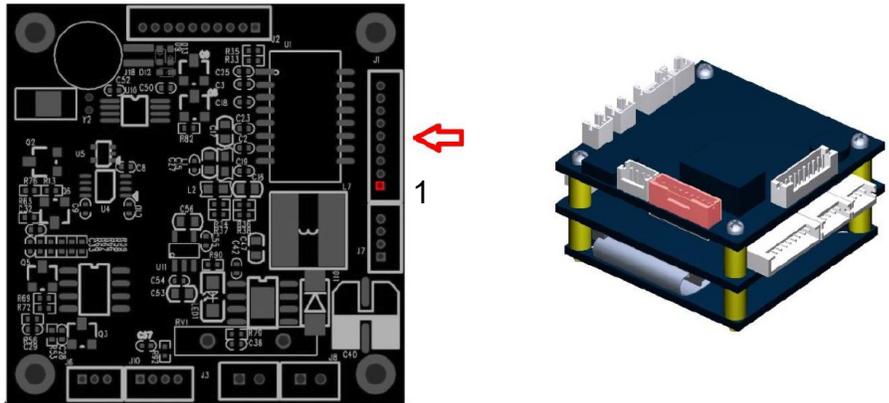


Mating connector: Molex 0873690200

Pin	Description
1	+12V In
2	GND

Note that there are two power connectors, which can be used to daisy chain power to multiple encoders. Please note carefully the power pin orientation. **The pinout is (unfortunately) opposite that of the power orientation of the Picoradio 12V aux power output.**

2.1.2 Ethernet

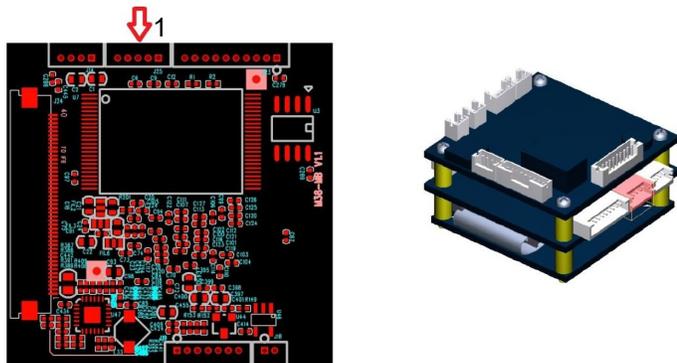


Mating part: Hirose DF13-9S-1.25C

Pre-crimped wires available: Digikey H4BBT-10110-B8-ND

Pin	Description
1	GND
2	Yellow LED out
3	Green LED out
4	48V POE
5	Ethernet Rx-
6	48V POE
7	Ethernet Rx+
8	Ethernet Tx-
9	Ethernet Tx+

2.1.3 Serial



A 3.3V level TTL serial port is available. A serial to ethernet bridge is available to transmit and receive serial communications over the ethernet stream.

Mating connector housing: Digikey WM1723-ND / Molex 0510210500, contacts we recommend using pre-crimped 6" leads:

Digikey 0500798000-06-W8-D-ND white

Digikey 0500798000-06-B8-D-ND black

Digikey 0500798000-06-L8-D-ND blue

Pin	Description
1	NC
2	Serial RX
3	GND
4	Serial TX
5	5V out @ 50 mA

2.1.4 Serial Dev Cable

You can make a serial dev cable using a USB to 3.3V TTL adapter, a short 0.1" header, and a 5 pin picoblade housing with crimped wires.

This cable can be used for configuration and debug, as well as testing serial passthrough connectivity.



Figure 1: USB Serial cable wired to 5 pin Picoblade serial connector

Qty	Part	Desc	Note
1	JBTek USB-TTL	USB-3.3V TTL cable	https://www.amazon.com/gp/product/B00QT7LQ88
1	PRT-00116	Header, 3 position	https://www.sparkfun.com/products/116
1	Molex 0510210500 / Digikey WM1723-ND	Picoblade 5 position housing	
1	Digikey 0500798000-06-W8-D-ND	pre-crimped 6" leads, white	

1	Digikey 0500798000-06-B8-D-ND	pre-crimped 6" leads, blue	
1	Digikey 0500798000-06-L8-D-ND	pre-crimped 6" leads, black	

3 Quickstart guide

3.1 New Default IP Address

Please Note:

As of ver 1.0.56 firmware, the default IP address is now 192.168.168.168, mostly for better default compatibility with Picoradio.

The below refers to the previous IP address / subnet of 192.168.1.x. Please change the default subnet of your PC etc. accordingly.

3.2 Videos / Youtube channel

Some videos are available on our youtube channel, for example:

Picoraptor / FLIR Duo integration

<https://www.youtube.com/watch?v=SPMpcr1McE0>

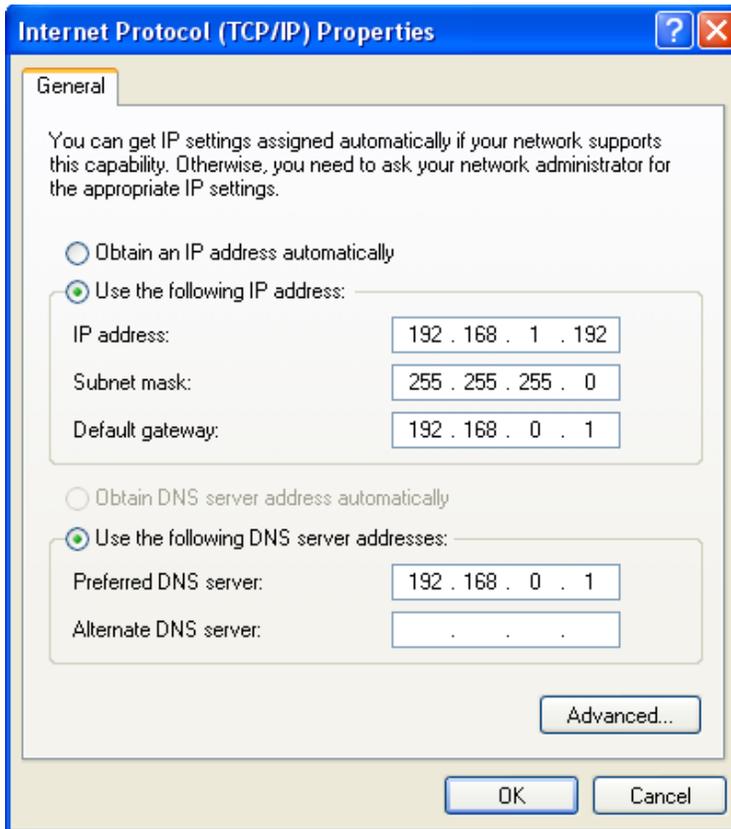
Please subscribe to our channel for more updates.

3.3 PC Network Configuration (Wired)

Disable any wireless network adapter the PC / laptop may have.

Configure the PC to have these network settings (example given for XP):

Internet Protocol (TCP/IP), properties



This sets up the system so that it can talk directly to the Picoraptor (gateway and DNS should not be necessary if direct wired).

At this point you should be able to ping the Picoraptor (address 192.168.168.168).

```
C:\>ping 192.168.1.168

Pinging 192.168.1.168 with 32 bytes of data:

Reply from 192.168.1.168: bytes=32 time<1ms TTL=64
Reply from 192.168.1.168: bytes=32 time<1ms TTL=64
Reply from 192.168.1.168: bytes=32 time<1ms TTL=64
```

You can also use ping 192.168.168.168 -t (on windows) to keep a continuous ping going (useful to know when the encoder is active after boot).

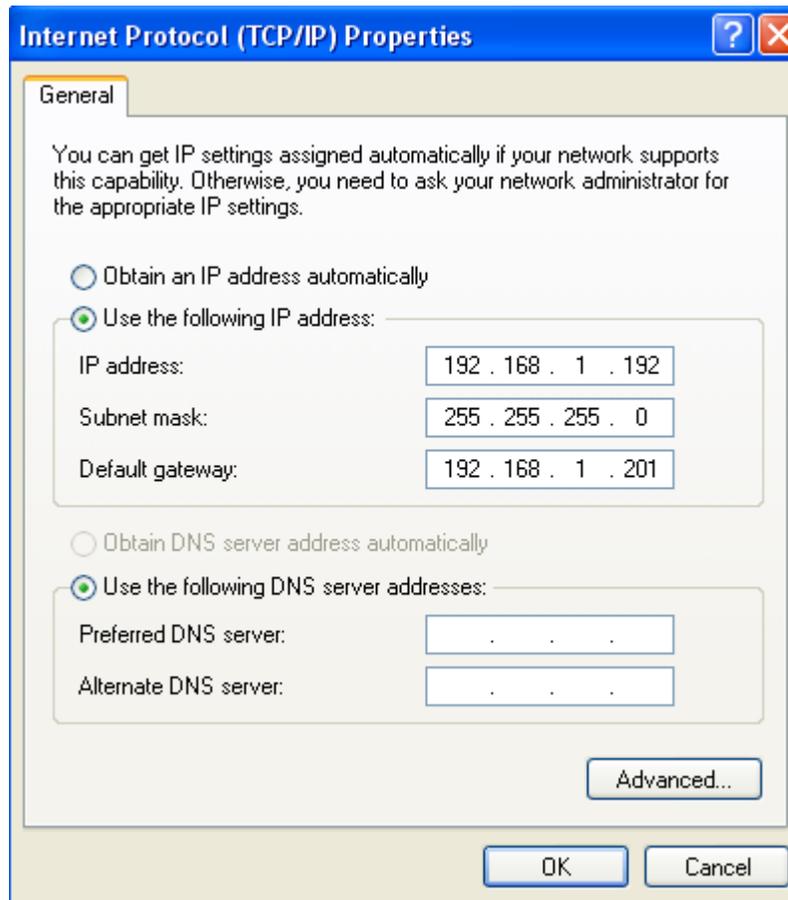
3.4 PC Network configuration (Wireless)

Set up the ground station. Plug in power and a UHF antenna of appropriate frequency. Plug a crossover (usually red) cable from the ethernet port of the ground station directly to the PC.

Disable any wireless network adaptor the PC / laptop may have.

Configure the PC to have these network settings (example given for XP):

Internet Protocol (TCP/IP), properties



This tells the system to use the ground station as the primary network gateway. Click on ok.

At this point you should be able to ping:

192.168.168.201 (the ground station radio)

192.168.168.210 (the airborne radio)

192.168.168.168 (the Picoraptor system through the radio link)

3.5 Video streaming using VLC

VLC is a popular freeware video client which may be used to stream and record video from the Picoraptor.

It is available for Windows / Mac / Linux / Android. / iOS

Download it from <http://videolan.org>

Once installed, you can enable the 'record' button by toggling on View/Advanced Controls. After it is enabled, the record button will toggle video recording. The destination directory depends on the platform, but is usually the Videos directory of the active user.

To stream video, select Media/Open Network Stream, and enter the following URL:

```
rtsp://192.168.168.168:8557/PSIA/Streaming/channels/2?videoCodecType=H.264
```

VLC can also be used to display video from the Picoraptor operating in Raw H.264 and MPEG2 modes.

You may also want to configure the network caching option to ~250-300 ms, possibly less if you are running wired instead of wireless, or possibly as much as 1000ms wireless if the link has dropouts.

Note that using VLC will not result in minimum latency over the wireless link. Our RaptorviewHD application addresses this issue (Linux client).

4 Usage

4.1 Cooling

Please note for extended use at high performance settings (1080p for example but not limited to this), we recommend some attention to cooling (particularly the CPU). Some airflow over the unit will make this task easy.

4.2 IP address

The default IP address is 192.168.168.168 (formerly 192.168.1.168 for v1.0.55 firmware and before).

4.3 Configuration

Configuration interface is

<http://192.168.168.168/config.html>

user: admin

pwd: 9999

4.4 Ffmpeg

You can use ffmpeg as a video client.

For ultra low latency, select 720max60 mode in the picocam/picoraptor settings for 60fps encoding. This may need a bit more bandwidth for motion detail (~5 Mbps)

Install ffmpeg from <https://ffmpeg.org/>

you can click through the windows builds and download one at <https://ffmpeg.zeranoe.com/builds/>

Once you install ffmpeg, create a script / batch file:

For example on windows:

ffmpeg_pico.bat:

```
cd ffmpegdir\bin [replace ffmpegdir with where you installed ffmpeg]
ffplay -fflags nobuffer -flags low_delay -framedrop -strict experimental
-rtsp_transport tcp
rtsp://192.168.168.168:8557/PSIA/Streaming/channels/2?videoCodecType=H.264
```

4.5 Gstreamer

For lowest latency you can use the Gstreamer media dev kit.

Under Windows, install for example:

<https://gstreamer.freedesktop.org/data/pkg/windows/1.9.2/>

gstreamer-1.0-x86-1.9.2.msi

Make sure you install all options (optional plugins).

Install Gstreamer.

Create a batch file called 'picocam.bat' in c:\gstreamer\1.0\x86\bin\

picoraptor.bat:

```
gst-launch-1.0 rtspsrc location=rtsp://192.168.168.168:8557/PSIA/Streaming/channels/2?videoCodecType=H.264 latency=50 ! queue ! rtph264depay ! avdec_h264 ! autovideosink
```

4.6 Android tablet interface

You may download the freeware Android client QtGStreamerHUD at:

<https://www.dropbox.com/s/iq65z4ab5b3vf8h/QtGStreamerHUD.apk?dl=0>

If you connect a router to a Picoradio or Microhard base station, (ideally running OpenWRT, such as TP-Link, VoCORE2, and, with patience, others), you can then connect the tablet to wifi and then to Picocamera or our Picoraptor encoder.

QtGStreamerHUD allows you to enter a custom Gstreamer pipeline. The pipeline to use is (change IP address accordingly):

```
rtspsrc location=rtsp://192.168.168.168:8557/PSIA/Streaming/channels/2?videoCodecType=H.264 latency=50 ! queue ! rtph264depay ! avdec_h264
```

QtGStreamerHUD also allows you to connect to a Pixhawk / APM stream to display a HUD.

Note to get proper colors you have to 'reverse' the color bytes in this interface (there is an option to do this in the QtGStreamerHUD interface).

4.7 Pixhawk compatible clients

We have had some mixed results with Pixhawk clients with built-in video support.

QtGStreamerHUD: Works, low latency on Android (720p mode, Samsung Galaxy Tab S2)

UGCS: Reported to work with RTSP mode but large latency

QgroundControl: Reported to work with RTSP (large latency)

4.8 Hardware Decoders

We offer a digital video decoder as well as an HD-SDI hardware decoder option.

These are available as enclosed or OEM units which decode the video stream and output video directly to digital video or HD-SDI.

4.8.1 Decoder configuration

The digital video decoder currently supports RTP video but not RTSP.

4.8.2 HD-SDI decoder configuration

The HD-SDI output module uses the digital video decoder with an HD-SDI converter added.

4.9 Serial port usage

The encoder can be used as a TTL serial port server. By default this server is active and is available on port 8111. Several TCP ports can be configured. The serial baud rate can be configured using the config web page. You can use this as an ethernet to serial bridge to control serial devices.

You can connect to the serial device with a telnet style TCP connection, for example:

```
telnet 192.168.168.168 8111
```

4.9.1 Virtual Com Port Software

You can use virtual com port software to create a virtual com port which behaves like a physical com port on your host operating system (Windows/Linux etc).

For example, under Windows you can use the free HW VSP3 software from HW-Group, available at:

https://www.hw-group.com/products/hw_vsp/index_en.html

Another option is COM2TCP, at

<http://astrogeeks.com/AstroGeeks/COM2TCP/index.html>

4.9.2 Multicast serial configuration

We now have a multicast serial configuration which can be used to receive multicast serial.

A usage case is to communicate with cameras inside a camera gimbal using serial data received over the ethernet stream. An advanced example is that we use this with our SBUS modules so that we can receive SBUS data at the vehicle using one of our SBUS2Serial modules, and also receive SBUS packet data at the encoder serial port. We have an SBUSPacket2Visca module which can then control camera zoom and focus inside the gimbal, from a stick command on the SBUS console (or on for example the Desert Rotor GCS).

4.10 Using a Raspberry Pi as a video test source

For in-house testing we frequently use a Raspberry Pi as a video test source.

You can download a raspberry pi image at:

<http://www.timschwartz.org/raspberry-pi-video-looper/>

It does come with a (comical) test video.

Instructions for installation and modifying the video are at:

<https://github.com/timatron/videolooper-raspbian>

With this looper setup the unit will automatically switch between 720p and 1080p video sources depending on what the Picoraptor is configured for.

4.11 Confirmed Working Cameras

We have tested a number of different video sources. Basically the requirement is that the camera be able to output 1080p30, 720p60, or 1080i60. However do not have tons of cameras to test.

Some cameras that provide only 1080p60 output may not work properly.

FLIR Duo (720p)

GoPro Hero 3 White: 720p60 / 1080p30 works

ReplayXD Prime X: 720p60 / 1080p30 works

Samsung NX1: 1080i60 mode working, 1080p30 not working

Other DSLRs: More reports of 1080i60 working but not 1080p30.

Raspberry Pi: 720p60 / 1080p30 works

Blackmagic Micro Cinema Camera: 720p60 / 1080p30 / 1080i60 works

Please feel free to send reports of your cameras so that we can add them to the list.

The Picoraptor must be configured for a particular input setting (i.e. 720p, 720p60 aka 720max60, 1080p30, or 1080i60), using its web based configuration page.

4.11.1 Some notes about GoPros:

The main annoyance with GoPros in our experience has been that they will turn off their video output very quickly (to save power) if they don't see anything connected. This resulted in us using a power-on sequence with them: Essentially power on the encoder, then power on the GoPro.

GoPros may also need to be configured to enable the video output.

We have had some reports of issues with recent GoPros (there are forum posts about issues with using them with other equipment as well).

5 UDP / RTP streaming modes

Support has been added for new MISB compatible RTP streaming modes. These are functional as of v.1.0.35.

To use these modes you must enable them in the video streaming configuration on the config.html page.

These are RTP UDP broadcast modes, available in unicast or multicast. These modes function well at HD video bandwidth (and work fine at high bandwidth, i.e. do not have the limitations of the other modes and can be over 2 Mbits/sec). The RTP packets have timestamps which can be correlated with KLV data. They can be accessed through RTSP or by using an SDP file (or just by tuning into the UDP stream). The RTSP stream for these modes can be accessed through rtsp://192.168.168.168:8559/rtp. The RTP UDP broadcast port is 18888.

For unicast, the data is broadcast to IP address 192.168.168.110 (this is currently fixed, can be changed in a text file on the SD card).

A multicast broadcast mode is also available, to address 239.255.43.43.

One advantage of these modes is that they are 'connectionless', i.e. always broadcasting, which can work better than RTSP over an RF link. If the link drops and resumes, video will resume. You can also start the receiving client before a video source is available, and when RTP packets start to arrive, video will be displayed.

Sample SDP file (unicast mode):

```
RTSP/1.0 200 OK
CSeq: 3
Date: Sat, Jan 01 2000 01:27:58 GMT
Content-Base: rtsp://192.168.168.168:8559/rtp/
Content-Type: application/sdp
Content-Length: 473

v=0
o=- 946689928100756 1 IN IP4 192.168.168.168
s=Session streamed by "MicroraptorHD"
i=/mnt/nand/in.264
t=0 0
a=tool:LIVE555 Streaming Media v2013.11.15
a=type:broadcast
a=control:*
a=source-filter: incl IN IP4 * 192.168.168.168
a=rtcp-unicast: reflection
a=range:npt=0-
a=x-qt-text-nam:Session streamed by "MicroraptorHD"
```

```

a=x-qt-text-inf:/mnt/nand/in.264
m=video 18888 RTP/AVP 96
c=IN IP4 192.168.168.110/255
b=AS:500
a=rtptime:96 H264/90000
a=control:track1

```

5.1 Playing RTP video with VLC

5.1.1 Using RTSP to initiate the unicast or multicast RTP stream

Configure laptop for IP address 192.168.168.110.

Enable the unicast or multicast RTP stream from the Picoraptor config page.

Start VLC. Open network stream, and enter URL: `rtsp://192.168.168.168:8559/rtsp` (optional) select 'show more options', then change caching to 300ms.

5.1.2 RTP Unicast: SDP file to initiate the stream

Create a text file (you can call it `RTPUnicast.sdp`).

Paste the following text:

```

v=0
o=- 968895205556561 1 IN IP4 192.168.168.168
s=Session streamed by "testH264VideoStreamer"
i=/mnt/nand/in.264
t=0 0
a=tool:LIVE555 Streaming Media v2013.11.15
a=type:broadcast
a=control:*
a=source-filter: incl IN IP4 * 192.168.168.168
a=rtcp-unicast: reflection
a=range:npt=0-
a=x-qt-text-nam:Session streamed by "testH264VideoStreamer"
a=x-qt-text-inf:/mnt/nand/in.264
m=video 18888 RTP/AVP 96
c=IN IP4 192.168.168.110/255
b=AS:500
a=rtptime:96 H264/90000
a=control:track1

```

Save the text file.

Open the text file with VLC (if the extension is `.sdp`, VLC will automatically start). Video should start playing.

5.1.3 RTP Multicast: SDP file to initiate the stream

Create a text file

Paste the following text:

```

v=0
o=- 946684819174959 1 IN IP4 192.168.168.168
s=Session streamed by "MicroraptorHD"
i=/mnt/nand/in.264
t=0 0
a=tool:LIVE555 Streaming Media v2013.11.15
a=type:broadcast
a=control:*
a=source-filter: incl IN IP4 * 192.168.168.168
a=rtcp-unicast: reflection
a=range:npt=0-
a=x-qt-text-nam:Session streamed by "MicroraptorHD"
a=x-qt-text-inf:/mnt/nand/in.264
m=video 18888 RTP/AVP 96
c=IN IP4 239.255.43.43/255
b=AS:500
a=rtptime:96 H264/90000
a=control:track1

```

Save the text file.

Open the text file with VLC (if the extension is .sdp, VLC will automatically start). Video should start playing.

5.2 RTP Multicast mode (one video source, multiple receivers)

In multicast mode, you can have one transmitter and multiple receiving stations. This works well with either multiple machines on the same local network, or can be used with the master/slave mode of the radio, for example in a configuration where an aircraft is transmitting video using a radio set in 'master' mode, and there are multiple connected receivers set to 'slave' mode, which are all receiving the video stream simultaneously, without duplicating bandwidth.

5.3 Using Multiple Simultaneous Encoders

It is possible to have multiple simultaneous encoders. For this to work properly, a few things must be set up.

The IP address of each needs to be changed to a unique value, for example with 4 encoders you would need to set each one up as for example 192.168.168.168, ~.1.169, ~.1.170, and ~.1.171.

In versions of firmware before 1.0.48, the firmware the encoders will all have the same MAC address by default. This needs to be changed because otherwise ethernet networking will not work properly (an ethernet switch will probably randomly switch between talking to two units and neither will function well). See the Advanced Network Configuration section for more information on how to change MAC address.

In versions of firmware 1.0.48 and later, units are given unique MAC addresses which depend on the IP address which is set, and there is no need to change MAC address of individual units..

You will also need to change the destination RTP ports and/or destination RTP IP addresses.

Essentially every unicast RTP source should have a separate destination IP address + RTP port combination. To be clear you can send multiple streams to one destination IP address, but they would need unique RTP destination ports.

And every multicast RTP port should have a unique destination multicast IP address + RTP port combination. In this case for clarity it may make more sense to use multiple destination multicast IP addresses.

5.4 Advanced Network Configuration

5.4.1 Changing IP address

The IP address can be changed within the same subnet by using the IP address field on the web configuration page.

Currently if the IP address is changed out of the subnet there is a bug which will result in the unit not being able to communicate on the network. This will be resolved in the next version. This can be fixed by communicating with the unit using the serial port instead of the network. Please get in touch if you have this issue.

5.4.2 Advanced network configuration changes (mac address)

To change the network configuration (including the subnet), you currently need to communicate with it via over telnet or potentially over serial.

Assuming you can successfully ping the unit, do the following:

```
telnet 192.168.168.168
```

```
user: root (no password)
```

```
cd /mnt/nand
```

```
vi netconfig.txt
```

Use the VI editor (use the cursor keys to scroll to the appropriate section, 'a' to start adding/editing, backspace to delete etc.

When changes are complete, use ESC then :wq

to save the file. Changes will take place after a reboot.

5.4.3 Default netconfig.txt file

```
LocalIP="192.168.168.168"  
NetMask="255.255.255.0"
```

```

Gateway="192.168.168.1"
DNS="0.0.0.0"
DHCPConfig=0
DHCPEnable=0
HTTPPort=80
Title="TI_IPNC"
MAC="de:ad:be:af:00:40"
PPPoEAccount=""
PPPoEPassword=""
PPPoEEnable=0

```

5.5 Advanced RTP configuration

There is also a similar configuration file which allows changing destination RTP ports and destination RTP IP addresses.

To change it:

```
telnet 192.168.168.168
```

```
user: root
```

```
cd /mnt/nand
```

```
vi vidconfig.txt
```

Use the VI editor (use the cursor keys to scroll to the appropriate section, 'a' to start adding/editing, backspace to delete etc.

When changes are complete, use ESC then :wq

to save the file. Changes will take place after a reboot.

5.5.1 Default vidconfig.txt file

```

# Simple config file rules:
#   - Comments begin with a hash
#   - Format is key = value
#   - Values can only be integers or strings, defined in airborne_config.c
#   Integers:
#       0 = FALSE
#       1 = TRUE
#   - Don't make this config file crazy big
#   - Don't make the key/value pairs crazy long
#   - Spaces and tabs are ignored, so these are equivalent:
#       bob=621
#       bob = 621
#       bob= 621
#
# Swap YC in the DM368
#   (code in ipnc_rdk/av_capture/framework/drv/usermod/src/drv_isif.c::miscConfig.ycInSwap)
# Default is 1 (true)
adv7611_dm368_yc_swap=1

# Swap YC on the ADV7611 chip itself, meaning write 0xA2 to register 4 of config area 98
#   (code in
ipnc_rdk/av_capture/framework/drv/usermod/src/imgs_ADV7611/drv_imgs_ADV7611.c::DRV_adv7611_imgsSe
tRegs())
# Default is 0 (false)
adv7611_on_chip_yc_swap=0

# Set H.264 encoder interlace mode
#   (code in ipnc_rdk/av_capture/framework/alg/src/alg_vidEnc.c::ALG_vidEncSetDynamicParams())
# 0 = ARF (default mode), 1 = SPF, 2 = MRCF
h264_interlace_mode=0

# Swap CB and CR on ADV7611 chip
#   This will turn on bit 0 of register 5 of config area 98

```

```
# (code in
ipnc_rdk/av_capture/framework/drv/usermod/src/imgs_ADV7611/drv_imgs_ADV7611.c::DRV_adv7611_imgsSe
tRegs())
# Default is 0 (false)
adv7611_cb_cr_swap=0

# Destination RTP port (unicast/multicast), Integer, usually 18888
rtp_port=18888

# Unicast destination IP address, String,          192.168.168.110
udp_unicast_ip=192.168.168.110

# Multicast broadcast IP address, String,          239.255.43.43
udp_multicast_ip=239.255.43.43

# Destination UDP port for non-RTP UDP modes, Integer, usually 4444
udp_uni_multi_port=4444
```

5.6 RTP Video + Hardware digital video output Decoder

RTP Video (multicast or unicast) can be used with our hardware decoder option to output digital video. At the moment this has somewhat higher latency than our software decoder.

This works with either unicast or multicast mode.

6 Multiple Video Stream Configuration

This section is a quick start guide to setting up dual video streams (for the same source) on the Picoraptor system.

Boot the system and connect to it direct wired through the external ethernet cable.

Connect to the AI configuration interface (<http://192.168.168.168/config.html>) and make sure that the camera configuration is set appropriately, and to a 720p frame size.

Using Internet Explorer (needed for ActiveX), log into the system's network camera configuration page:

<http://192.168.168.168>

user: admin

password: 9999

You will see the network video control panel.

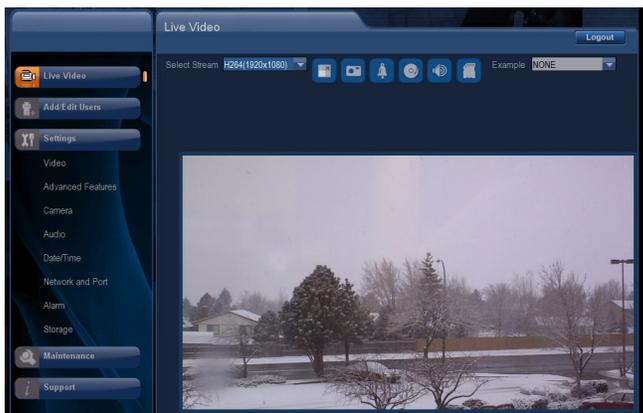


Figure 2: Network video interface live video screen

Set the output resolution to 720p

Click on the video tab.

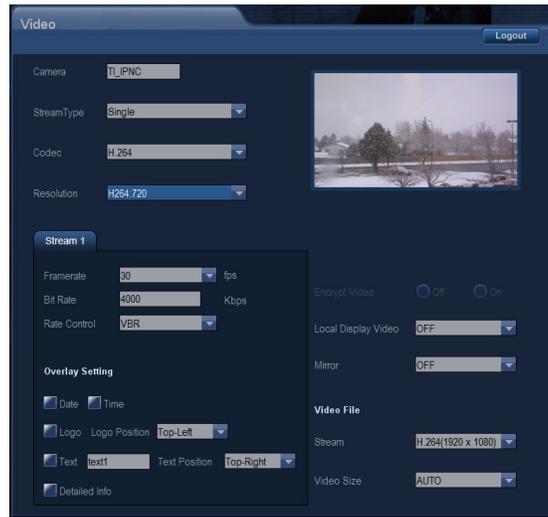


Figure 3: Video configuration tab

Select Resolution: H264:720

Click ok.

Wait for the system to reconfigure at 720p if needed.

Click on Video:StreamType, and change this to Dual.

Click on Codec and change the stream type to the desired dual stream configuration (here we will use dual H.264).

Select resolution: H264:720, H264:QVGA.

Note: Not all resolution combinations are functional with all imager sources.

You can configure the frame rates and bit rates for the two streams by clicking on Stream1 and Stream2.

Click Ok.

Wait for the system to reconfigure.

Click on the Live Video display.

Select each stream in the live video display, and/or enable both streams with the 'Display all streams' checkbox.

6.1 Accessing Video Streams

Video streams may be accessed at the following URLs (when configured with these streams enabled):

Primary H.264 Video Stream	rtsp://192.168.168.168:8557/PSIA/Streaming/channels/2?videoCodecType=H.264
Secondary H.264 Video Stream	rtsp://192.168.168.168:8556/PSIA/Streaming/channels/2?videoCodecType=H.264
Primary MPEG4 Video Stream	rtsp://192.168.168.168:8553/PSIA/Streaming/channels/1?videoCodecType=MPEG4
Secondary MPEG4 Video Stream	rtsp://192.168.168.168:8554/PSIA/Streaming/channels/1?videoCodecType=MPEG4
MJPEG Video Stream	rtsp://192.168.168.168:8555/PSIA/Streaming/channels/0?videoCodecType=MJPEG

These video streams may be displayed using VLC on any platform, including PC/MAC/iOS/Android.

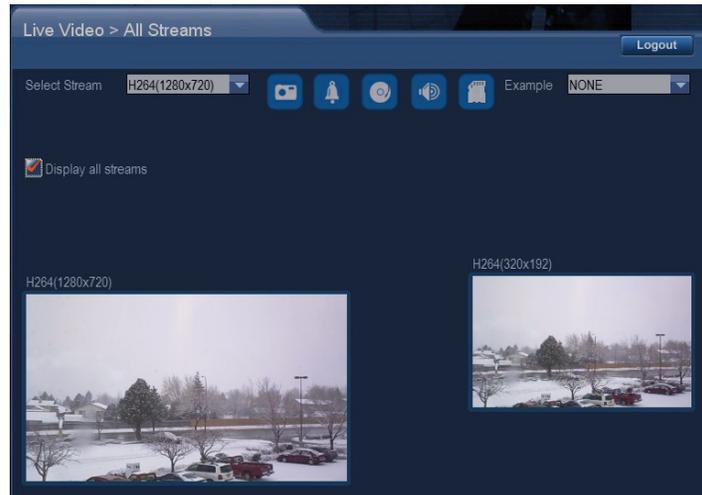
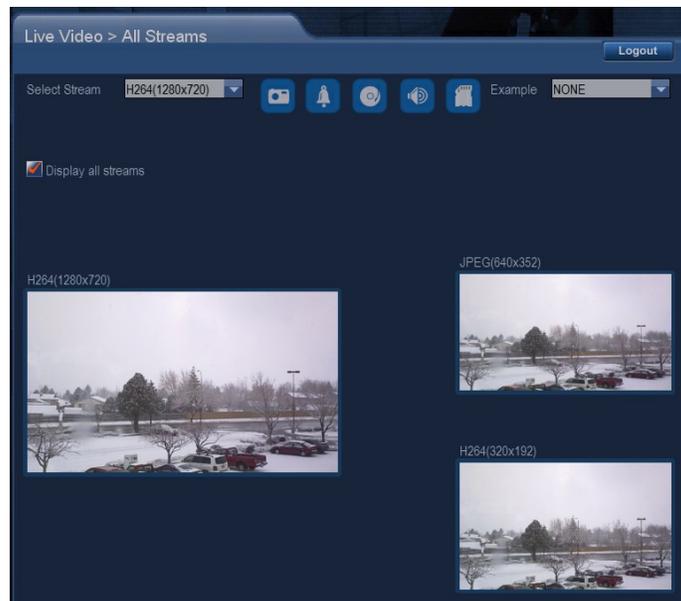


Figure 4: Multiple video stream display

Note it is also possible to set up a tri-stream configuration, with two H.264 streams and one MJPEG stream.



6.2 Multistream Web Configuration API

Example: Tri-Stream with codec set to "Dual MPEG4 + JPEG" and resolution of "MPEG4:720, MPEG4:QVGA, JPEG:VGA"

<http://192.168.168.168/vb.htm?videocodec=2&videocodeccombo=1&videocodecres=0>

```
Stream Type = videocodec
Codec Combo = videocodeccombo
Resolution = videocodecres
```

```
Stream Type = 0 = Single
Codec Combo = 0 = H.264
```

```

Resolution = 0 = 720
Resolution = 1 = D1
Resolution = 2 = SXVGA
Resolution = 3 = 1080
Resolution = 4 = 720MAX60
Codec Combo = 1 = MPEG4
Resolution = 0 = 720
Resolution = 1 = D1
Resolution = 2 = SXVGA
Resolution = 3 = 1080
Resolution = 4 = 720MAX60
Codec Combo = 2 = MegaPixel
Resolution = 0 = H264:2MP
Resolution = 1 = JPG:2MP
Resolution = 2 = H264:3MP
Resolution = 3 = JPG:3MP
Resolution = 4 = H264:5MP
Resolution = 5 = JPG:5MP
Stream Type = 1 = Dual
Codec Combo = 0 =      H.264 +  JPEG
Resolution = 0 = 720  +  VGA
Resolution = 1 = D1   +  D1
Resolution = 2 = 720  +  720
Resolution = 3 = 1080 +  QVGA
Resolution = 4 = 1080 +  D1
Codec Combo = 1 =      MPEG4 +  JPEG
Resolution = 0 = 720  +  VGA
Resolution = 1 = D1   +  D1
Resolution = 2 = 720  +  720
Resolution = 3 = 1080 +  QVGA
Resolution = 4 = 1080 +  D1
Codec Combo = 2 =      H.264 +  H.264 (Dual H.264)
Resolution = 0 = 720  +  QVGA
Resolution = 1 = D1   +  D1
Resolution = 2 = D1   +  QVGA
Resolution = 3 = 1080 +  QVGA
Resolution = 4 = 1080 +  D1
Codec Combo = 3 =      MPEG4 +  MPEG4 (Dual MPEG4)
Resolution = 0 = 720  +  QVGA
Resolution = 1 = D1   +  D1
Resolution = 2 = D1   +  QVGA
Resolution = 3 = 1080 +  QVGA
Resolution = 4 = 1080 +  D1
Codec Combo = 4 =      H264 +  MPEG4
Resolution = 0 = D1   +  D1
Resolution = 1 = 1080 +  D1
Stream Type = 2 = Tri-Stream
Codec Combo = 0 =      H264 +  H264 +  JPEG
Resolution = 0 = 720  +  QVGA +  VGA
Codec Combo = 1 =      MPEG4 +  MPEG4 +  JPEG
Resolution = 0 = 720  +  QVGA +  VGA

```

7 Onboard Recording

Note: We are finishing Picoraptor support for this feature now so it is not yet a mainline feature.

Picoraptor supports recording video onboard using a MicroSD card. Typically a large MicroSD card is desirable (we have tested up to 32GB).

Note that since we are currently using the MicroSD card to boot the system (for several reasons), some special preparations must be used to create a 32GB card using the system software.

7.1 MicroSD card compatibility

Please use a class 10 speed card.

32GB Cards that did not work:

Monster
Sandisk Ultra 32GB

Worked:

Emtec 32GB

Wintec 32GB Micro SDHC FileMate Mobile Professional SDHC Model 3FMUSD32GBC10-R

7.2 HTTP API to enable/disable recording

Record enable/disable can be performed from the ActiveX interface (<http://192.168.168.168> using Internet Explorer).

Or using the below API. Note that the user and password must be authenticated (default is admin/9999).

The recording enable/disable controls

Enable Recording24-7

[http://192.168.168.168/vb.htm?](http://192.168.168.168/vb.htm?rftpenable=0&ftpfileformat=0&sdrenable=1&sdfileformat=0&recordlocalstorage=0&schedulerepeatenable=0&schedulenumweeks=52&schedule=schedule=00107000000235900&schedule=01101000000235900&schedule=02102000000235900&schedule=03103000000235900&schedule=04104000000235900&schedule=05105000000235900&schedule=06106000000235900&scheduleinfiniteenable=1)

[rftpenable=0&ftpfileformat=0&sdrenable=1&sdfileformat=0&recordlocalstorage=0&schedulerepeatenable=0&schedulenumweeks=52&schedule=schedule=00107000000235900&schedule=01101000000235900&schedule=02102000000235900&schedule=03103000000235900&schedule=04104000000235900&schedule=05105000000235900&schedule=06106000000235900&scheduleinfiniteenable=1](http://192.168.168.168/vb.htm?paratest=reloadflag)
<http://192.168.168.168/vb.htm?paratest=reloadflag>

Disable Recording

[http://192.168.168.168/vb.htm?](http://192.168.168.168/vb.htm?rftpenable=0&ftpfileformat=0&sdrenable=0&sdfileformat=0&recordlocalstorage=0&schedulerepeatenable=0&schedulenumweeks=52&schedule=schedule=0000700000000000&schedule=0100700000000000&schedule=0200700000000000&schedule=0300700000000000&schedule=0400700000000000&schedule=0500700000000000&schedule=0600700000000000&scheduleinfiniteenable=1)

[rftpenable=0&ftpfileformat=0&sdrenable=0&sdfileformat=0&recordlocalstorage=0&schedulerepeatenable=0&schedulenumweeks=52&schedule=schedule=0000700000000000&schedule=0100700000000000&schedule=0200700000000000&schedule=0300700000000000&schedule=0400700000000000&schedule=0500700000000000&schedule=0600700000000000&scheduleinfiniteenable=1](http://192.168.168.168/vb.htm?paratest=reloadflag)
<http://192.168.168.168/vb.htm?paratest=reloadflag>

7.3 Commandline/startup script recording

You can also set up recording to happen from a script.

```
/opt/ipnc/Appro_avi_save filename interval msgid framerate audiocodec checkcard streamid abitrare
```

filename: filename including path

interval: length of the file

msgid: message id to communicate with encode stream

framerate: AVI frame rate

audiocodec: audio codec (0 no audio, 1 ulaw 2 aac)

checkcard: tell the save route to check card left space or not.

Streamid: which stream to save (0: stream2, 1: stream3)

abitrare: audio bit rate which AV encoder used.

You can put this into the /mnt/nand/airborne.sh startup script if you want it to record at powerup.

8 FLIR Vue Compatibility



Figure 5: FLIR Vue with output module

With a Picoradio and the SBUS module, some functions of the FLIR camera can be controlled over SBUS (using an SBUS to PWM converter).

9 AerialPixels cables

AerialPixels makes custom flat flex cables with customizable ends (and offers some right angle and reverse right angle connectors).

Order from:

<https://aerialpixels.com>

10 Compact 5 Port Ethernet Switch

Airborne Innovations can supply a compact 5 port ethernet switch with robust connectors. You can use it to connect up to 4 encoders to one Picoradio module.



Figure 6: 5 port OEM Ethernet switch module

Power input is 5-12V.

Dimensions: 50x45x10mm

Connectors:

Ethernet: Housing JST XHP-4, Digikey 455-2267-ND
 Contacts JST CONN TERM CRIMP XH 22-28AWG, SXH-001T-P0.6
 12" Pre-crimped wires with contacts: ASXHSXH22K305

Power: Housing JST XHP-2, Digikey 455-2266-ND
 Contacts and crimped wires- same as above.

11 MicroSD Card reliability

Note that in the past year we have begun shipping units with industrial MicroSD cards for boot reliability. We recommend booting from SD instead of from NAND, also for that reason. One unfortunate consequence is a slower boot time. This also makes it difficult to use SD recording features.

We have had some reports of MicroSD card failures with the commercial grade cards we were using. For that reason we recommend the following MicroSD card:

Apacer AP-MSD256ISI-IT (Digikey 1582-1117-ND)



12 Serial debugging / Linux console

A serial interface is available for debugging. This can be configured as a Linux serial terminal.

By default this interface will display some initial boot information but will NOT boot into the Linux serial terminal.

With recent versions of firmware this can be configured for a one time boot into the Linux serial terminal.

The basic procedure is:

Configure a USB toTTL serial interface (or other 3.3V TTL serial port such as the default serial port on a Raspberry Pi) to 115200 baud 8n1 settings.

We can recommend USB to TTL serial converters from Adafruit and Sparkfun.

For example:

<https://www.sparkfun.com/products/13830>

<https://www.sparkfun.com/products/12977>

Power up the encoder.

Wait for the message which says: Type 'uboot' to continue

Type the word 'uboot' to enter the serial commandline

type: aircon [enter]

type: boot [enter]

When the Linux console finishes booting it will ask for a user

type: root [enter]

and you will be in the Linux console.

ifconfig [enter] will let you look at the network status.