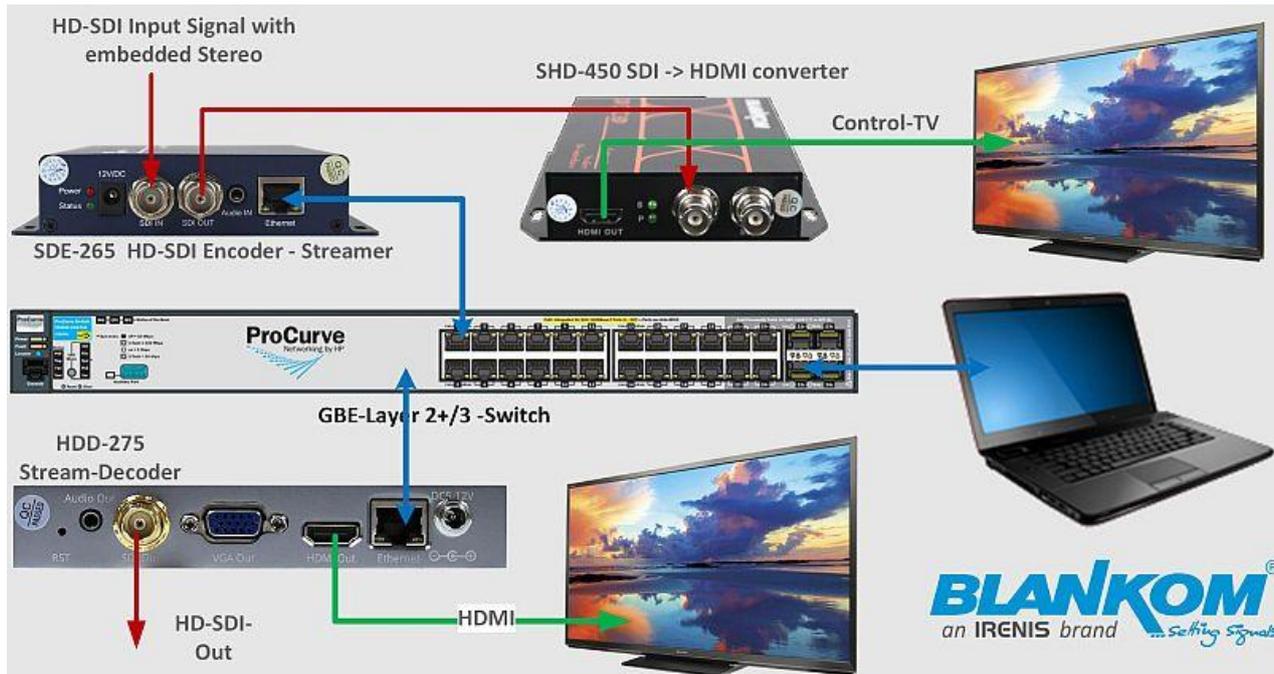


QuickStart DPE/VDE/HDE/SDE Encoder / Decoder pair – New FW with neu GUI 2026:



Example with SDI-IN: Setup scenario – devices can vary...

TOC

Starting with the Encoder IP:	2
The Network settings	5
On Screen Display	9
Recording to NFS.....	10
The different Input Frontends	14
Encoding Parameters:	15
About Preview functions:.....	17
TS/Multicast settings	24
Audio:.....	29
Other protocols HLS/RTMP:.....	33
RTSP/HTTP-streaming	34
ONVIF	34
Loki:.....	35
Others:	35
Annex 4Kp60 decoder/transcoder HDD-276 operation:	38

The only difference between the VDE, HDE and the SDE SoC Multiprotocol Video encoders is the Input – Frontend: VGA, HDMI or SDI and eventually the DPE = Display-Port encoder – depends on HW platform as well.

And therefore, the DPE/VGA/SDI Versions do not have an EDID entry in its Web-IF because it's not needed.

Both are equipped with loop-IN -> out connectors, so the Input signal can be cascaded or used for monitoring.

The HDD Decoder box is available as w/ HD-SDI output or as a 'light' Version w/o SDI.

We assume, that you are a skilled person which is familiar with the basic IT-network settings and understanding of streaming protocols. If not, you can get more information from our website- chapter Tutorials/Know-how:

<https://www.blankom.de/tutorials.html> (English) | <https://www.blankom.de/anleitungen-deutsch.html> German.

Different HW/SW combinations supporting different features and parameters:

Example: The W-FW supports recording to NFS/Samba network folders, A-Versions might not.

The encoders almost have 4 encoder engines in 1 chip but limitations are to be considered:

The Mainstream-Encoder-part can be used to support WebRTC and TRTC while the substreams do not and will be disabled. Main and 1st Substream supporting the full resolution encoding while 2nd and 3rd substream engines do 1 step below and another step below: E.g.: Main and sub will encode in FullHD, so 2nd 3rd supporting the output stream only as 720 but this depends from the HW version so the FW as well – please check the datasheets and HW/FW.

Starting with the Encoder IP:

Default Login Addresses Encoder: 192.168.1.168 | Decoder: 192.168.1.169 user/password: admin/admin

The password can be changed according to your own security policy. The username stays as it is because of its internal Linux OS to keep the system rights for its operation.

So, to access both units, your Laptop/PC should be set to the same IP-range as the decoders. For later stream checking's we recommend to use only the Network interface and disable another equipped – like WIFI – otherwise the computer might not be able to decide which interface to be used for the reception of a stream – in particular for Multicasts UDP/RTP.

If both Network-devices are ON, you can change/set the METRIC Values in your Network-devices as lower value has a higher priority: Example for Ethernet: Change from Auto to manual and set the Metric to 10, while the WIFI should be set to a higher value e.g., 100.

New features with the ne GUI version (1.xyA A=Ambarella / 2.yxW W= Hisilicon):

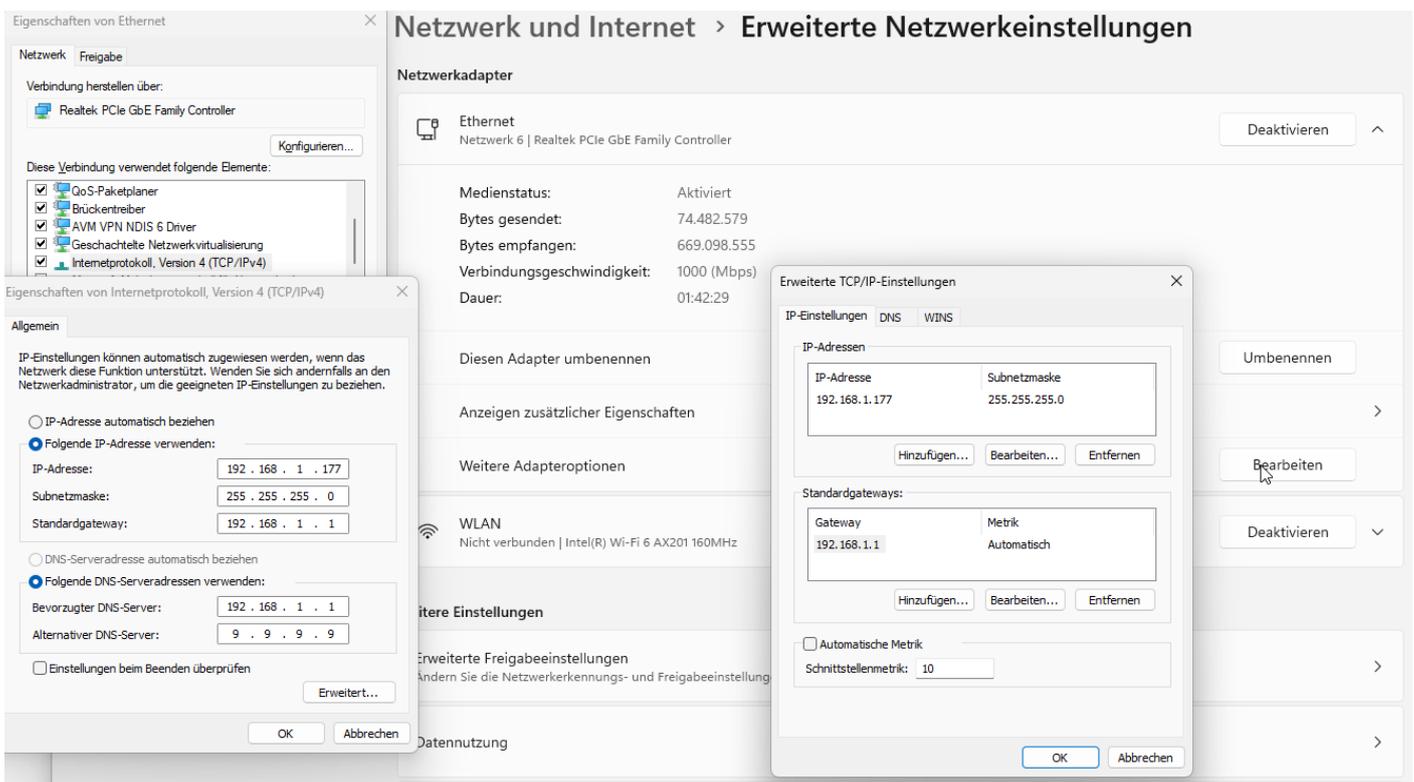
Network related: IPv6* support, mDNS, HTTPs by certificate,

NFS or SAMBA-network file systems mounting for mp4/TS Recording*

Stream-Targets: WebRTC and TRTC, 3x RTMPs – Pushing*, Multi SRT streaming as 'Listener' (No Rendezvous-mode) (Rem: VLC supports only 'calling the SRT-IP from the Listener (encoder-streamer))

**depends on HW and FW*

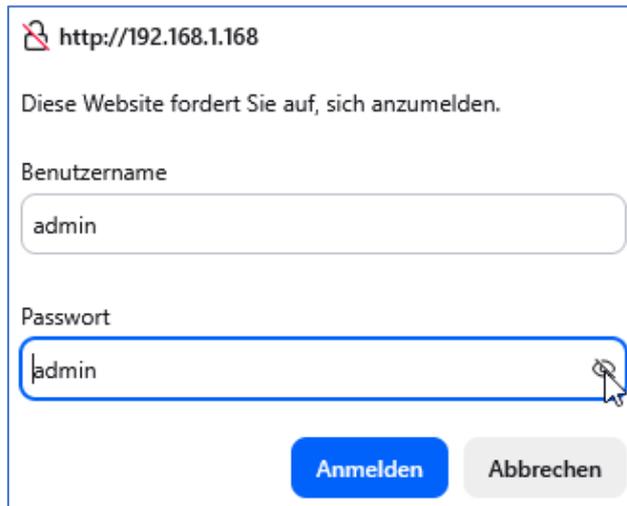
Example in Windows 11: Give your PC a manual IPv4 address/GW/NM/DNS and it's Metric:



Recommended browsers: Mozilla, EDGE, Chrome or Opera... or any other ... but it might have an impact depending on the individual browser settings.

HINT: All of our De- and Encoder of this product family are equipped with a **RESET** switch accessible through a tiny hole at the front or rear site. If you lost your changed IP-Address or password, you can set the unit back to default by pressing the RST switch by a needle for at least 10-15 seconds (Green LED and NW-LEDs will go off) -> release the button and it will reboot with factory defaults.

After Login,



http://192.168.1.168

Diese Website fordert Sie auf, sich anzumelden.

Benutzername

admin

Passwort

admin

Anmelden Abbrechen

You'll get following Status-screens:

Encoder:



Nicht sicher http://192.168.1.168/indexE.html

BLANKOM
H.265
HEVC

Encoder System Platform
Version: 2.15W

Status

Encoder

OSD

System

Status

Device Status

Uptime: 0000-00-00 01:25:28

Device Time: 2026-02-12 10:39:10 (Sync Time To Device)

CPU Usage: 3%

CPU Junction Temperature: 39°C

Memory Usage: 132.0M/496.8M

Input Resolution: 1920x1080i@50

Received Video Frames: 64280

Lost Video Frames: 2

Audio Samplerate: 48.0 kHz

Received Audio Frames: 120434

Network Sent Packets: 572

Network Dropped Packets: 0

Recording Status

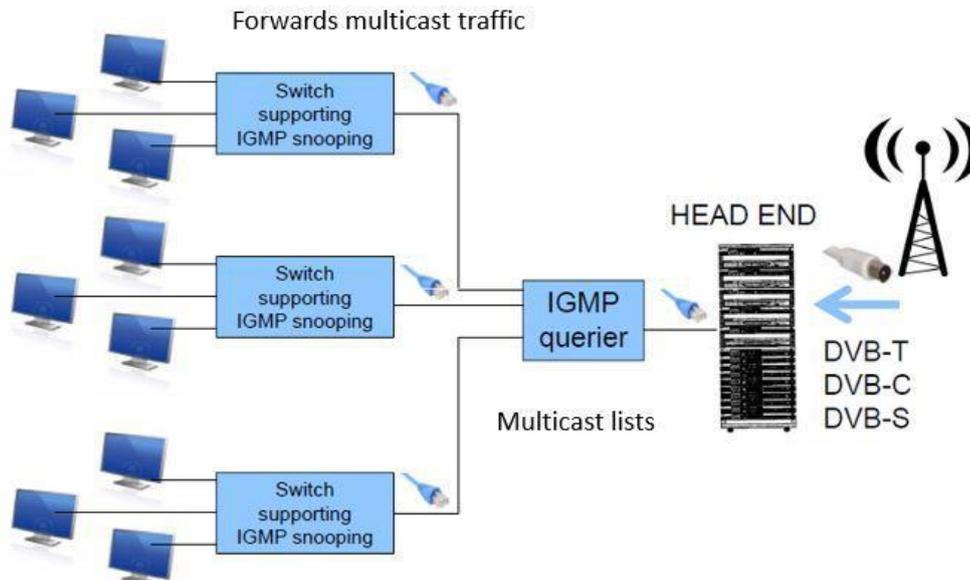
Some info upfront:

In General: The default encoding parameter settings are almost fine to start with. Unless you know what you are doing, keep them ;-)

As Encoder **bitrate** settings we recommend to set it to 5000-8000 kb/s depending on your needed picture quality or even more... you need to find the balance..

If you use **multicast streaming** (UDP/RTP) for IPTV hospitality environments, you should assure, that your Ethernet Switch supports **IGMP** to avoid flooding your network with those streams and overload other devices network ports in your system.

Please do not use the same UDP or RTP IP addresses (224.0.0.0 ... 239.255.255.255) in your network as IGMP manage these addresses in your Ethernet Switch on demand (Query / Snooping):



Decoder:

The screenshot shows the web interface of a BLANKOM H.265 Video Decoder (Model: HDD-275, FW Version: 1.70.0). The interface is accessed via a browser at http://192.168.1.169. The main content area displays the following information:

- Status**
- System status**
 - Device Name: decoderboy
 - Device Time: 2026-02-12 10:40:36 (Sync time to device)
 - Runtime: 0000-00-00 00:48:06
 - CPU usage: 43%
 - CPU Junction Temperature: 51°C
 - MEM usage: 32MB/253MB
 - Net status: internet
 - HDMI format: 1080I50
 - Channel number: 1
- Screen Layout**
 - Layout: Disable
- Channel1**
 - URL: srt://192.168.1.169:9000?mode=listener

Rem.: Both already with a connected input signal

The Network settings:

Recommendation: Use a static IP address is almost better than DHCP. IPv6 is now supported but we recommend to concentrate on the IPv4 and let the IPv6 in Automatic mode.

Please change the DNS values to your local needs. If you have a Router in between or a Bridge with own DHCP-Address ranges, please assure that you set it to non DHCP ranges reserved for static IP's and avoid IP address-conflicts.

NTP is almost helpful for time synchronisations and e.g. overlay the time/date information in the encoded Video stream. But you'll need a local NTP (some Internet routers are advertising it or grab it from the Internet).

The Port-addresses usually can be kept as they are as defaults or changed if you know what you are doing.

HOSTNAME

HOSTNAME:

mDNS:

mDNS URL:

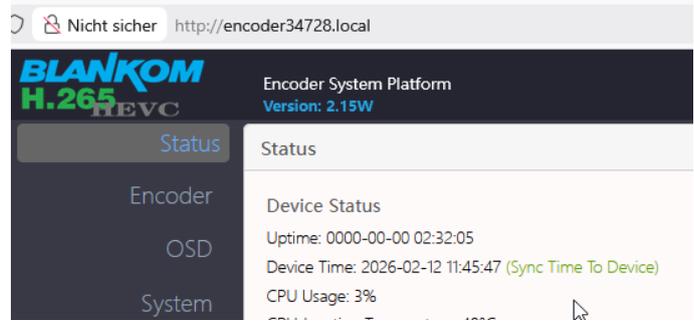
Apply

The new GUI/FW supports mDNS.

The Hostname can be changed.

If you need to use it, you can enable it and the device is accessible through its URL:

Copy it and paste to your browser: Example: encoder34728.local



Needless to say: All changes need to APPLY it.

Comparison: The **HDD-decoder** (as a couple encoder -> decoder) settings are nearly similar:

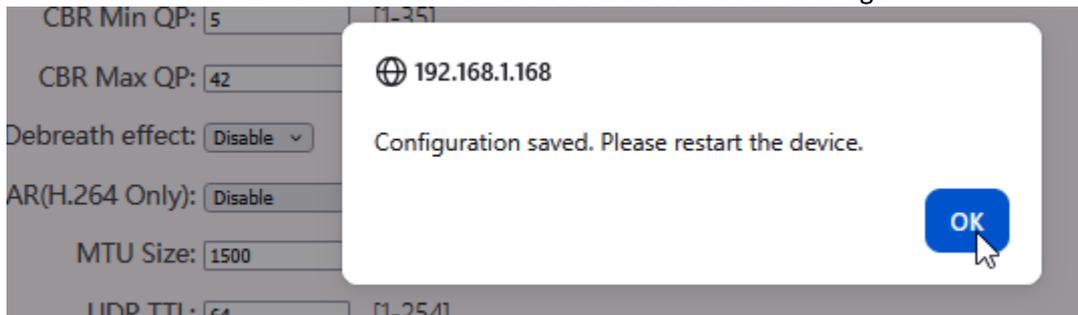
But actually, no mDNS supported and the MTU settings are located here. Rem.: If you pass the encoder-streams through VLANs in a GbE Switch, you might need to increase the MTU values to avoid packet loss i.e., in UDP streams which you even not will recognize except in the Stream-Receiver as picture/sound disturbances or Makro-blockings.

The encoder supports MTU adjustments it in the Submenu Encoder -> Advanced:

MTU Size: [500-1500]
 UDP TTL: [1-254]
 UDP Socket Buffer Size: (0-20971520)

HINT: Also some Network-Switch vendors are using proprietary functions and non open public features... which might cause such distortions by packet droppings.

Just a small remark about the APPLY button and its confirmation message

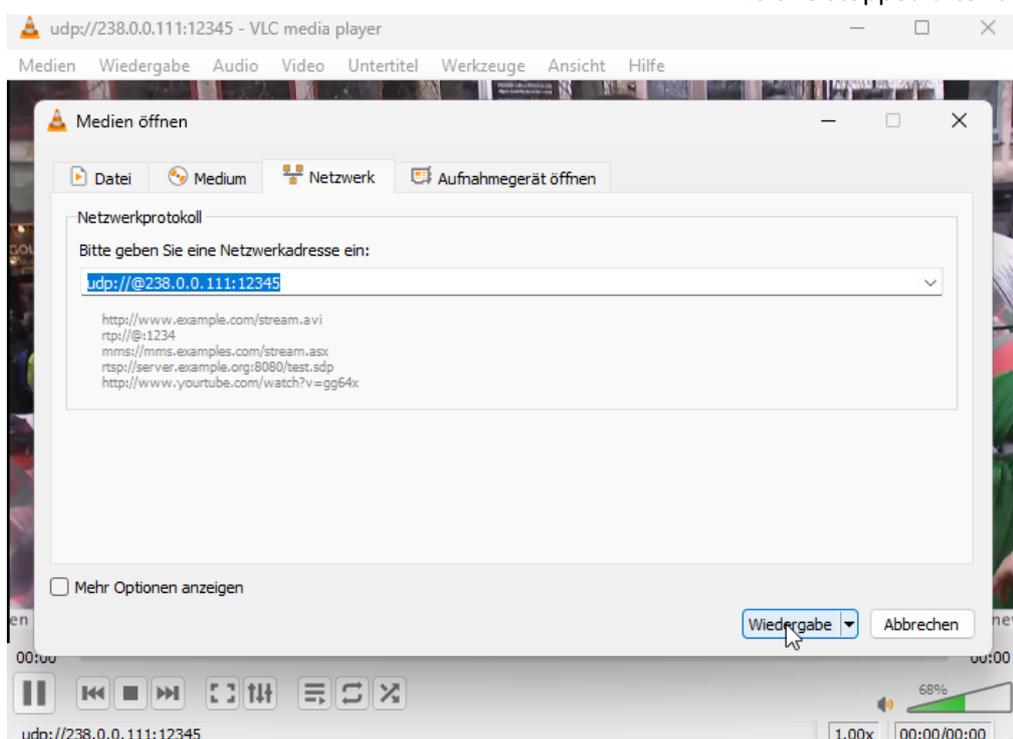


This mostly **does not mean to reboot** the units (En- Decoder) but to re-adjust your receiver unit and restart it because of essential parameters has been changed by you.

Example: If you change the codec from h.264 to h.265/HEVC, your receiver might stop decoding:

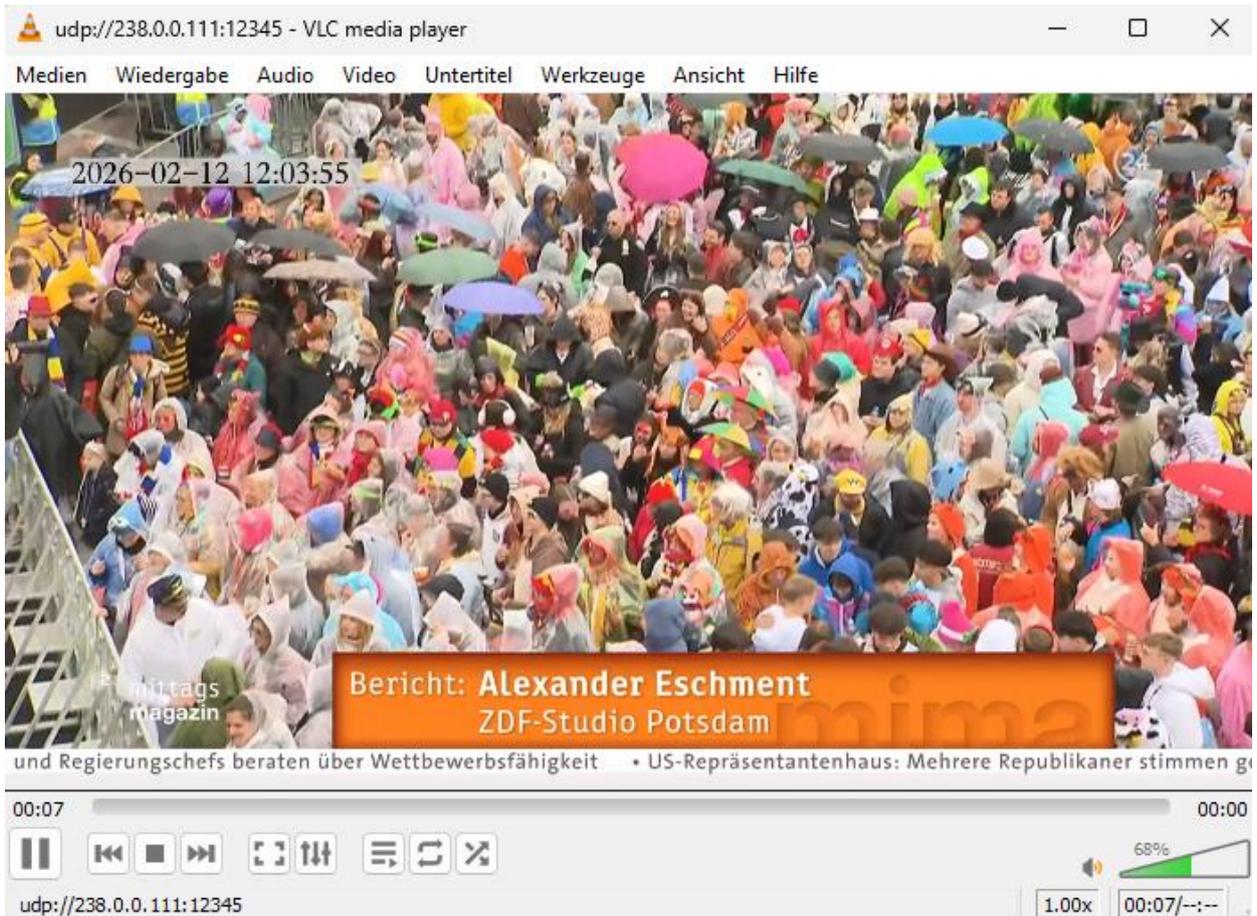


the codec. Restart it:



This one stopped after changed

And e'voila:



A hint: If you do not have a valid input signal, the encoder is streaming a Test-picture:



Some decoders or IPTV receivers might see this as a pumping stream. To avoid this and the stream does not contain a real 'still-picture' but even small changes in it, you can insert an overlay to the Video by inserting a running clock info (see above) by entering the

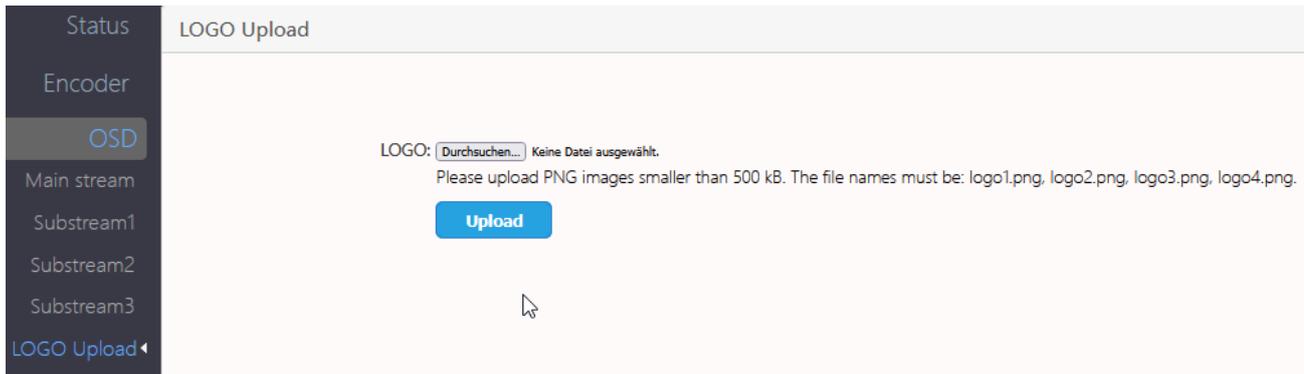
On Screen Display OSD -menu and set the 'Time' info to it:

Up to 4 OSD's can be added as Logos or Text and choosing the regions X/Y colours and transparency (ALPHA):

A lower ALPHA increases the transparency:

For inserting Logos as PNG there are some rules -like size max each 500kB - to consider for the picture to show it correctly.

The Logos or advertisement banners can be uploaded in its submenu:



Recording to NFS

The new GUI also supports the recording to a NFS-server (v4) or a SAMBA fileserver mounted to it. Because of SAMBA is a way more complicated the version is important and should harmonize while Windows-Server SMB is mostly not the same issue as with a Linux-Server, we assume, that your IT/Network knowledge should be professional or you need to consult your IT techies.

A simple example to establish a NFS server is the Linux installation here a RaspberryPI5 with the actual Raspbian OS:

Open a terminal and enter:

```

sudo apt-get update
sudo apt-get install nfs-kernel-server
sudo mkdir /home/public
sudo chmod ugo+rwx /home/public  makes it open for everyone
sudo nano /etc/exports
add this line:
/home/public *(rw, sync, no_subtree_check, all_squash)
Save it and reload:
sudo exportfs -ra
restart the server:
sudo service nfs-kernel-server restart
check the services running:
sudo rpcinfo -p
should show many times: portmapper, status, nfs, nlockmgr und mountd,
Add the IP and mounting folder and 'mount' it.
Then you see the following if the mounting was successful. If not ... search for the problem.

```

So, we need to enter into the Recording Settings: <IP-Address>: /home/public/

And mount it. If the mounting fails, you'll get a message. Remark:

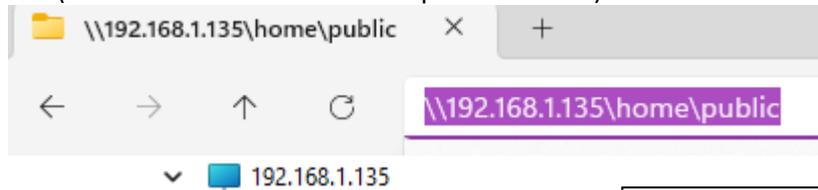
Mounting an NFS from a Windows11 PC has a different format by opening a CMD/PS as admin of course:

```
PS C:\Users\Blank> New-PSDrive -Name K -PSProvider FileSystem -Root "\\192.168.1.135\home\public" -Persist
```

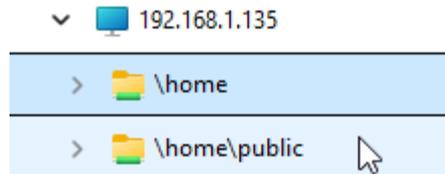
Name	Used (GB)	Free (GB)	Provider	Root	CurrentLocation
K	17,39	216,77	FileSystem	\\192.168.1.135\home\public	

```
PS C:\Users\Blank> |
```

Here as Virtual HD named K (should be a free and not occupied character):



-> Network:



You must enter the address/path once -> than it pops up

Unmounting from Windows11:

```
PS C:\Users\Blank> Remove-PSDrive -Name K
```

Back to the Encoder:

Encoder System Platform
Version: 2.15W

Video Recording Settings

Recording Status

Recording Format: NFS

Current Addr: 192.168.1.135:/home/public/

Mount Status: Unmounted (Mount Device)

Recording Settings

Storage Type:

Example Address: 192.168.1.100:/nfsroot/

Storage Address:

Apply

Encoder System Platform
Version: 2.15W

BLANKOM H.265 HEVC

- Status
- Encoder
- OSD
- System**
- Network
- Record
- Password
- Transport
- Upgrade and Backup
- Reboot
- Schedule Reboot
- Factory Reset

Video Recording Settings

Recording Status

Recording Format: NFS
 Current Addr: 192.168.1.135:/home/public/
 Mount Status: Mounted (Unmount Device)
 Total Disk Space: 239786 MByte
 Free Disk Space: 218957 MByte
 Recording Status: Not Recording (Start Recording) (Delete All Video Files)
 Recording Files: [Files List](#)

Recording Settings

Storage Type:

Example Address: 192.168.1.100:/nfsroot/
 Storage Address:

[Apply](#)

After recording something... Remark: **Of course, it needs an input signal...!!!** Otherwise recording will not start. You'll see the files you have done and can be accessed by your Windows-PC or directly in the Linux-Server or play it from Browser (downloading takes some time) ... So, you do not need an expensive NVR anymore.

Encoder System Platform
Version: 2.15W

BLANKOM H.265 HEVC

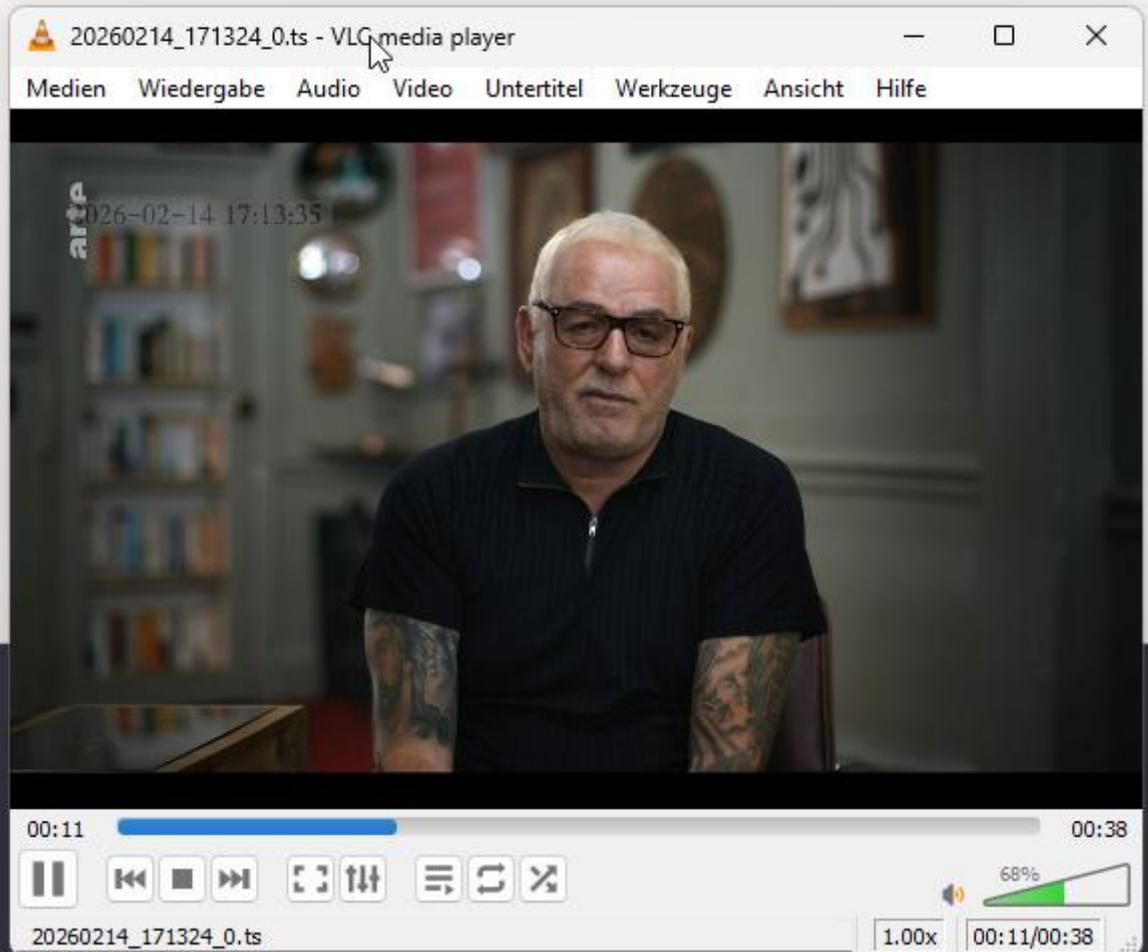
- Status**
- Encoder
- OSD
- System

File list

File name	Time	File size	Action
20260214_171105_0.mp4	2026-02-14 17:11:49	39.97M	Delete
20260214_171324_0.ts	2026-02-14 17:14:02	26.71M	Delete

You can download and play it:

File name	Time
20260214_171105_0.mp4	2026-02-14 17:11:49
20260214_171324_0.ts	2026-02-14 17:14:02



The different Input Frontends

have naturally different encoding parameters. Example for VGA (VDE-265):

Encoder System Platform
Version: 2.15W

Status | **Video Input** | Input Brightness Adjustment | Video Output

Encoder

- Main stream
- Substream1
- Substream2
- Substream3
- Video**
- Audio
- Advanced
- OSD
- System

◆ Rotation is available only when the input format is YUV420.
◆ Video Crop (Origin: Top-Left).

Video Input Format:

Flip / Mirror:

Video Crop:

Video Crop Start X (pixels): [0,1920]

Video Crop Start Y (pixels): [0,1080]

Video Crop Width (pixels): [400,1920]

Video Crop Height (pixels): [400,1080]

Apply

HDMI-Encoder – actual Ambarella Version:

Encoder System Platform
Version: 1.57A

Status | **Video Input**

Encoder

- Main Stream
- Substream1
- Substream2
- Substream3
- Video**
- Audio
- Advanced
- OSD
- System

◆ Rotation is available only when the input format is YUV420.

Video Rotation:

Flip / Mirror:

Video Crop:

Video Crop X Coordinate: [0,1920]

Video Crop Y Coordinate: [0,1080]

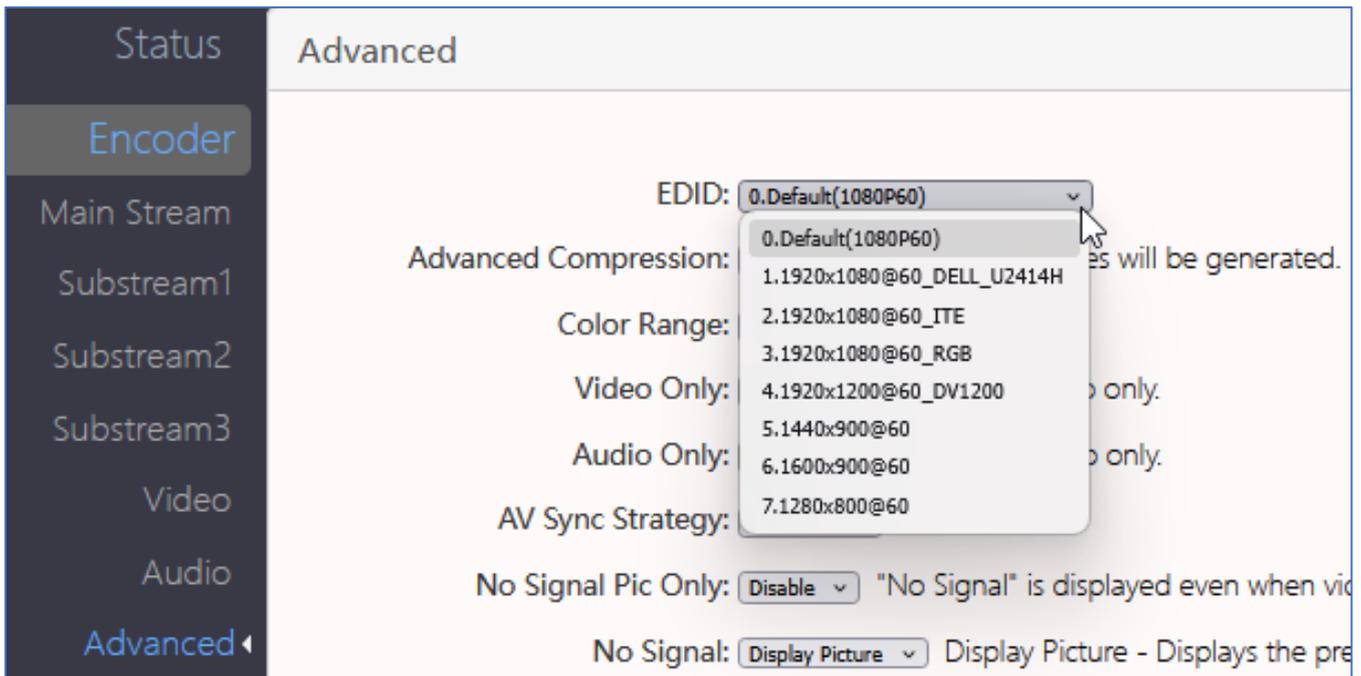
Video Crop Width (pixels): [0,1920]

Video Crop Height (pixels): [0,1080]

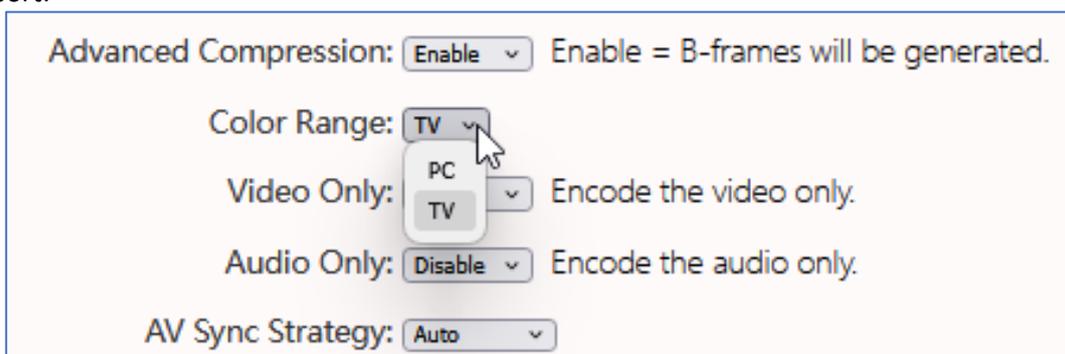
Monochrome:

Apply

Compared with the VGA and SDI-Versions it has an **extra entry** for the HDMI Settings:



The **EDID** is necessary to simulate different HDMI Input connections to its sources and the **B-Frame** support:



If **B-Frames** are supported by the Chipset* /Codec and should be used, it increases the picture quality if necessary but might increase latency and bitrate of the stream. Make yourself familiar with the different codec architectures please.

Encoding Parameters:

Of course, such a Video-Encoder has a lot of parameters to be adjusted to serve worldwide requirements.

Example: US related countries were working on 60Hz (30 or 60 FPS) or its history is different (NTSC instead of PAL). Interlaced or progressive modes are also to be considered.

E.g.: Full HD can have 1920x1080 i50 (Interlaced) as an Input or like the German public broadcasters sending 720p50 (progressive) via Satellite while others are transmitting 1920x1080i50.

Our SoC encoders are taking the interlaced input signals but combine the even and odd half-picture into a Fullscreen progressive output: 1080i50 will be 1080p25 (or can even done a p50). The default settings are FPS 30 and GOP 30 – which is related to US 60 Hz!!! Usually the IBBP- or IP-frame components in a codec is also used for different approaches – so you should also investigate a bit in codec basics?

So, your **first action** when configuring the encoder with its menu **should be**:

Check the input signal value you are getting from the STATUS screen:

Be patient, it takes some time to update itself.

Change the encoding parameters in MAIN/Secondary etc...:

Pay attention to the hints in the web-menu.

Of course, you can modify the input to a different output frame rate but imagine what happens if you have an input of interlaced 50 and output is set to 30 (progressive by chip-default). The encoder must reply/add or delete pictures/frames... and therefore the quality will decrease. **Recommended: Set the GOP to 2x FPS.**

About Preview functions:

You can get all/most actual running parameters in Realtime from the STATUS window:

BLANKOM H.265 HEVC
Encoder System Platform
Version: 1.57A

Status

- Device Status
- Uptime: 0000-00-00 01:46:46
- Device Time: 2026-02-17 12:26:47 (Sync Time To Device)
- CPU Usage: 19%
- Memory Usage: 238.3M/512.0M
- Codec Usage: 86%
- Input Resolution: 1920x1080i@50
- Video Status: Normal
- Received Video Frames: 9091
- Audio Samplerate: 48.0 kHz
- Audio Status: Normal
- Received Audio Frames: 8422
- Network Sent Packets: 178
- Network Dropped Packets: 0
- Preview (Low Latency)

The video preview shows a man and a woman sitting at a table in a studio setting. A red glowing joint is overlaid on the video. A red banner at the bottom of the video contains the text: "EILMELDUNG der Verfassungsschutz Niedersachsen hat demnach 'rechtsextremistische Bestrebungen' bei der AfD".

Here you can check with the web-Interface directly the input signal by clicking the Preview-link (green). Upper corner right side of the embedded Video -> you can close it...

Another STATUS-Page preview function is directly from after the encoder parts Main/Secondary etc:

Main Stream

Video Codec: H.264

Video Resolution: 1920x1080@50

Bitrate (kbps): 8000

Real-time FPS: 50.10

TS URL: <http://192.168.1.168/0.ts>

HLS TS URL: Disabled

HLS MP4 URL: Disabled

MP4 URL: <http://192.168.1.168/0.mp4>

FLV URL: <http://192.168.1.168/0.flv>

RTSP URL: <rtsp://192.168.1.168/0>

RTMP URL: Disabled

RTMP(S) Push URL0: Disabled

RTMP(S) Push URL1: Disabled

RTMP(S) Push URL2: Disabled

Multicast URL: <udp://@238.0.0.1:1234>

SRT Listener URL: <srt://192.168.1.168:9000>

SRT Caller URL(Connected):<srt://192.168.1.169:9002>

HLS Push URL:Disabled

TRTC Room: Disabled

Icecast Push URL:Disabled

Preview (HTML5)

Apart from all information you are getting here, you can also copy and paste those URL's to your video-Player like PC-VLC, Potplayer and FFPLAY as well as IPTV STB's, our IP-decoder HDD series TV's with integrated IPTV clients...

But for the internal Preview player the FLV URL must have been enabled.

Because FLV is an outdated Adobe thing, it works best with h.264 codec but h.265-hevc is also supported but no control possible

Main Stream

Video Codec: H.264

Video Resolution: 1920x1080

Bitrate (kbps): 8000

Real-time FPS: 50.10

TS URL: http://192.168.1.168/0.ts

HLS TS URL: Disabled

HLS MP4 URL: Disabled

MP4 URL: http://192.168.1.168/0.mp4

FLV URL: http://192.168.1.168/0.flv

RTSP URL: rtsp://192.168.1.168/0

RTMP URL: Disabled

RTMP(S) Push URL0: Disabled

RTMP(S) Push URL1: Disabled

RTMP(S) Push URL2: Disabled

Multicast URL: udp://@238.0.0.1:1234

SRT Listener URL: srt://192.168.1.168:9000

SRT Caller URL(Connected): srt://192.168.1.169:9002

HLS Push URL: Disabled

TRTC Room: Disabled

Icecast Push URL: Disabled

Preview (HTML5)

Fullscreen Browser-Preview can be used with mp4 http address streaming enabled:

Main Stream

Video Codec: H.264

Video Resolution: 1920x1080@50

Bitrate (kbps): 8000

Real-time FPS: 50.10

TS URL: http://192.168.1.168/0.ts

HLS TS URL: Disabled

HLS MP4 URL: Disabled

MP4 URL: <http://192.168.1.168/0.mp4>

FLV URL: http://192.168.1.168/0.flv

Opens a new browser window and plays

Remark:

-----> back to Encoder parameters:

It is essential that you know:

- Your input values (i.e., resolution and framerate, maybe audio level and its format PCM/Bitstream)
- If you want to transcode that or keep the input to be streamed simply
- Your target streaming clients and which protocols are supported

Please note, that there are particular differences between such (System on Chip) SoC based encoders and FPGA based professional encoders and that PC's as receivers like using VLC are almost failure tolerant against packet loss and other problems.

This manual cannot explain in details about the different codecs their many different parameters for different stream clients – note: Also, the different IPTV receivers like STBs have their different problems, bugs and features which might not match.

We are skipping here the setups of the 1/3 Sub-streams which can be configured for parallel output streaming with other encoding parameters – except Audio – this is common for all 4 (or 2 in some encoder types).

The ideal AV-encoding setup is to first start with following the input parameters from the STATUS Page:

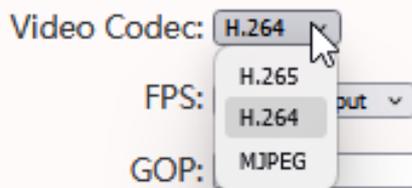
Codec Usage: 86%
 Input Resolution: 1920x1080i@50
 Video Status: Normal
 Received Video Frames: 15272
 Audio Samplerate: 48.0 kHz
 Audio Status: Normal
 Received Audio Frames: 19673

The input is interlaced (here HDMI from a SAT STB which its output setting is fixed to 1080i50 nevertheless the TV service it receives might be 720p50...) it should be constant/fixed at the input because frequently changings in the HDMI/SDI input might cause longer interruptions in the stream and other side effects. So, we have 1920x1080i50 and an Audio SR of 48KHz -> we should set the encoder parameters accordingly.

Note the hints from the Encoding web-menu:

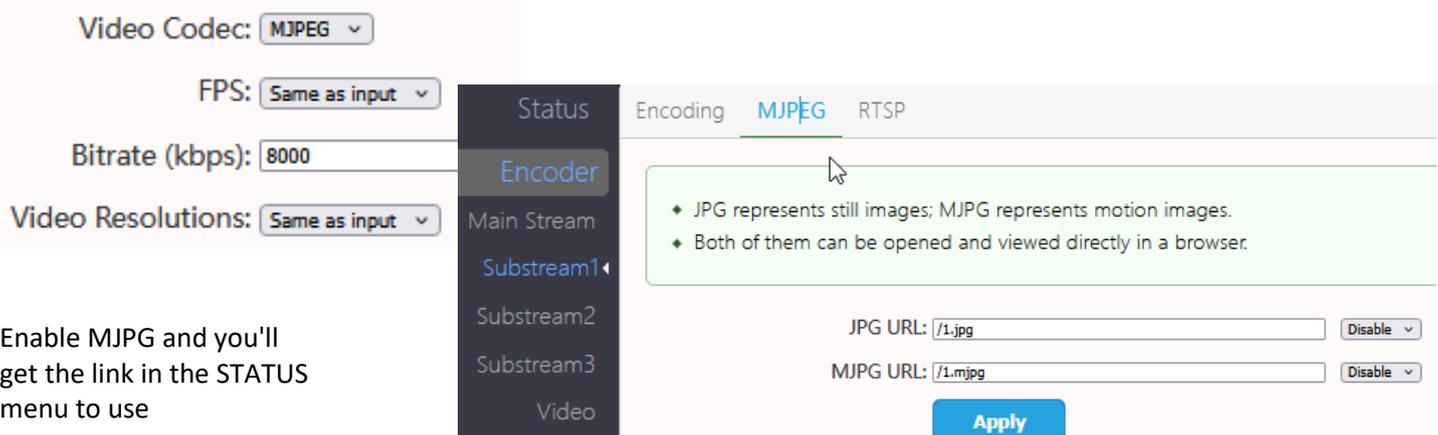
- H.265 offers better quality than H.264. Use it whenever your **decoder supports** it. Group of Pictures, keyframe interval = GOP / FPS (in seconds)
- In VBR mode, the bitrate is limited by the Quality Range. Selecting "Better...Best" or "Best" can result in very high actual bitrates, ignoring the target bitrate. Only "Lower...Best" keeps the target bitrate meaningful.
- For DVB systems, "Strong CBR" is required. Minimum: 5000 kbit/s for 1080p; Recommended: 8000kbit/s.
- Key-Frame Interval Settings:
 If the input frame rate is "N" (e.g., 60, 50, 30, 25), set the key-frame interval to twice the input frame rate (2xN).
- In Auto mode, the interval is set to half the input frame rate when N>30; otherwise, it is set to N.

Some IPTV receiver might only support h.264 – we recommend to start with this and test it:

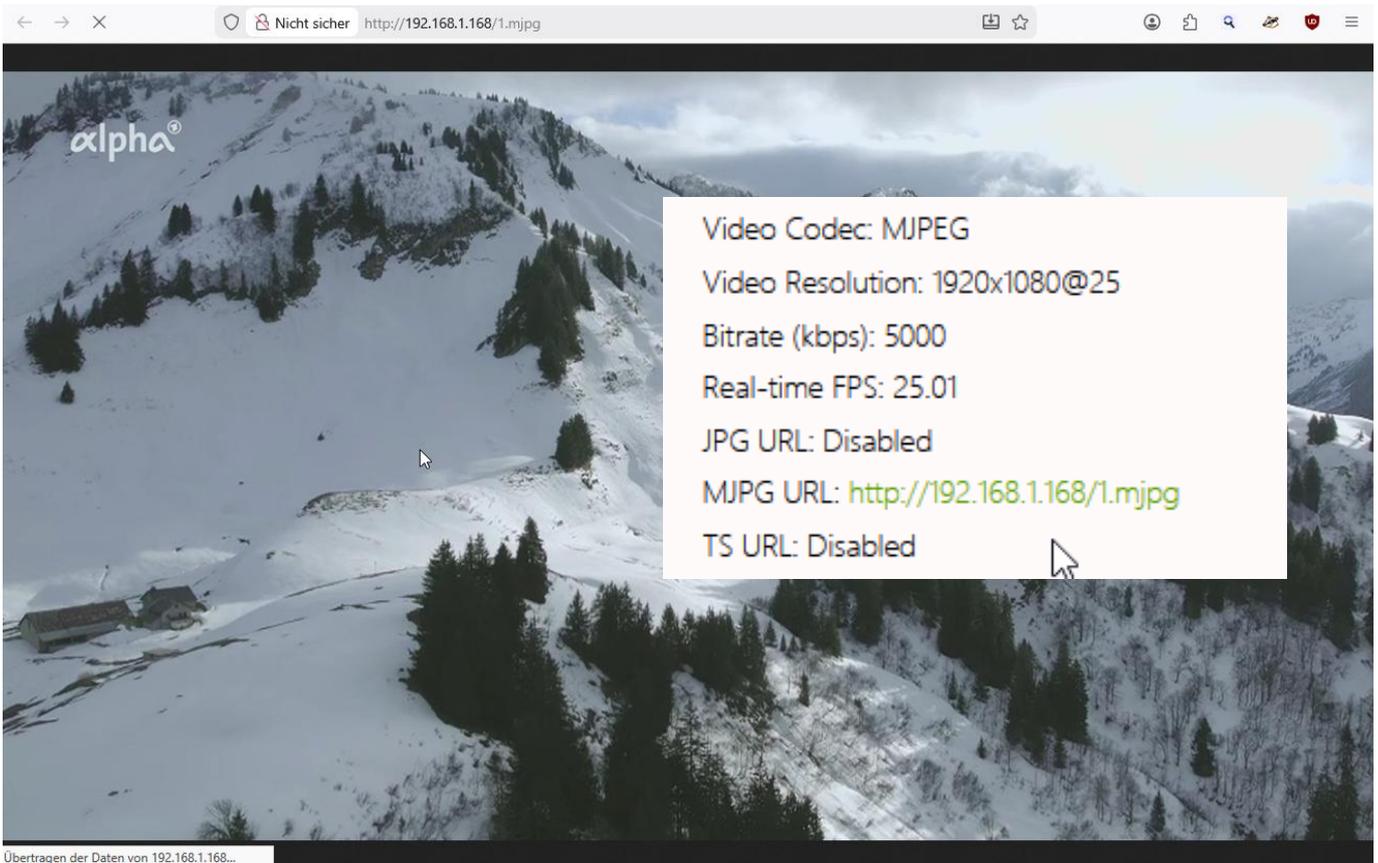


MJPEG is for direct MJPEG streaming support and if

used, the streaming menu changes:



Enable MJPEG and you'll get the link in the STATUS menu to use



But there is no AUDIO support for MJPG...

Please **NOTE**: If you set one of the encoder parts (Main-substream1...3) to MJPG, the other streams will be switched off. If you accidentally enabled MJPG, and have returned, please re-enable your former streaming protocols.

Video Codec:

FPS:

GOP:

Bitrate (kbps):

Same as Input does the same to the output e.g.: i50 will be p50 out p25 will be p25 out... interlaced will become progressive always.
 AUTO means: input i50 = out p25 (best choice)
 CUSTOM: you can adjust by yourself but be careful...

Video Codec:

FPS:

Custom FPS: [5-60]

GOP: [5-300] GOP

Bitrate (kbps): [32-32000]

Changing to 30, the stream will get 30

Video Codec: H.264
 Video Resolution: 1920x1080@30
 Bitrate (kbps): 8000
 Real-time FPS: 29.91

for US-Norm TV's 😊 ...

Image Quality Levels: **Lower → Best**

Video Resolutions: **Best**

H.264 Profile: **Better → Best**

Bitrate Mode: **Mid → Best**

Lower → Best

Lowest → Best

We recommend to use that default but you can play with the settings. **Rem:** increasing the quality will increase the latency...

Encoding TS/Multicast RTSP RTMP(S) SRT WebRTC TRTC HLS **MP4/FLV/Other**

- ♦ MP4 streams can be played directly in browser. FLV streams support H.265 and H.264, but players such as VLC only decode H.264.
- ♦ Icecast is audio-only and supports MP3 streams.

MP4 URL: **Disable**

FLV URL: **Enable**

Icecast Push URL: **Disable**

Apply

This is true: VLC does not accept the old Adobe Flash streaming with HEVC codec – only h.264 if you want to check it:

The screenshot shows the VLC media player interface. On the left, the 'Aktuelle Medieninformationen' (Current Media Information) window is open, displaying details for Stream 0 (H264 - MPEG-4 AVC) and Stream 1 (MPEG AAC Audio). The video stream is playing in the main window, showing a rocky landscape. An error dialog box titled 'Fehler' (Error) is displayed in the foreground, stating 'Unidentifizierter Codec: VLC konnte den Audio- oder Videocodec nicht identifizieren' (Unidentified Codec: VLC could not identify the audio or video codec). The error dialog has 'Löschen' (Delete) and 'Schließen' (Close) buttons.

Encoder System Platform
Version: 1.57A

Encoding TS/Multicast RT (S)

- ◆ H.265 offers better quality
- ◆ Group of Pictures, keyframe
- ◆ In VBR mode, the bitrate is
- ◆ "Lower~Best" keeps the tar
- ◆ For DVB systems, "Strong C
- ◆ Key-Frame Interval Setting:
- ◆ If the input frame rate is "N
- ◆ In Auto mode, the interval

Video Codec: Same as input

FPS: 1920x1080, 1600x1200, 1600x900, 1440x900, 1366x768, 1280x1024, 1280x800, 1280x720, 1024x768, 1024x576, 1024x544, 960x1080, 960x540, 850x480, 800x600, 720x576, 720x540, 720x480, 720x404, 704x576, 640x480, 640x360, 608x448, 544x480, 480x480, 480x384, 480x360, 480x320

GOP: Same as input

Bitrate (kbps): Same as input

Image Quality Levels: Same as input

Video Resolutions: Same as input

As you can see, the encoder chip supports format changings...

But see below, now the OSD overlay needs new parameters as well.

Please also have an eye on the correct picture settings: 16:9 input -> to keep that E.g.: if you set it to 720x576 its 4:3 the picture 16:9 will be egg-head formatted

Therefore, we recommend to use input follow mode.

Aktuelle Medieninformationen

Medien Metadaten Codec Statistiken

Informationen über den Aufbau des Mediums oder des Streams. Muxers, Audio- und Videocodes, Untertitel werden angezeigt.

- Stream 0
 - Originale ID: 100
 - Codec: H264 - MPEG-4 AVC (part 10) (h264)
 - Typ: Video
 - Videoauflösung: 480x272
 - Pufferabmessungen: 480x272
 - Bildwiederholrate: 50
 - Decodiertes Format: DX11
 - Ausrichtung: Oben links
 - Grundfarben: ITU-R BT.709
 - Farübertragungsfunktion: ITU-R BT.709
 - Farbraum: ITU-R BT.709 Bereich
 - Farbsättigungslage: Links
- Stream 1
 - Originale ID: 200
 - Codec: MPEG Audio layer 1/2 (mpga)
 - Typ: Audio
 - Decodiertes Format: 32 bits float LE (f32l)
 - Dekodierte Kanäle: Stereo
 - Dekodierte Abtastrate: 48000 Hz
 - Dekodierte Bits.pro Sample: 32

Ort: udp://@238.0.0.1:1234

udp://238.0.0.1:1234 - VLC media player

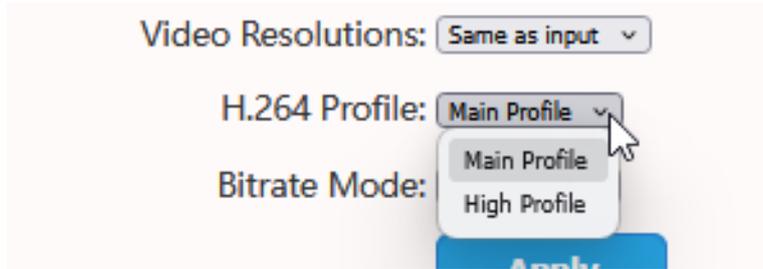
Medien Wiedergabe Audio Video Untertitel Werkzeuge Ansicht Hilfe

alpha 2026-02-17 16:15:51

00:32 00:00

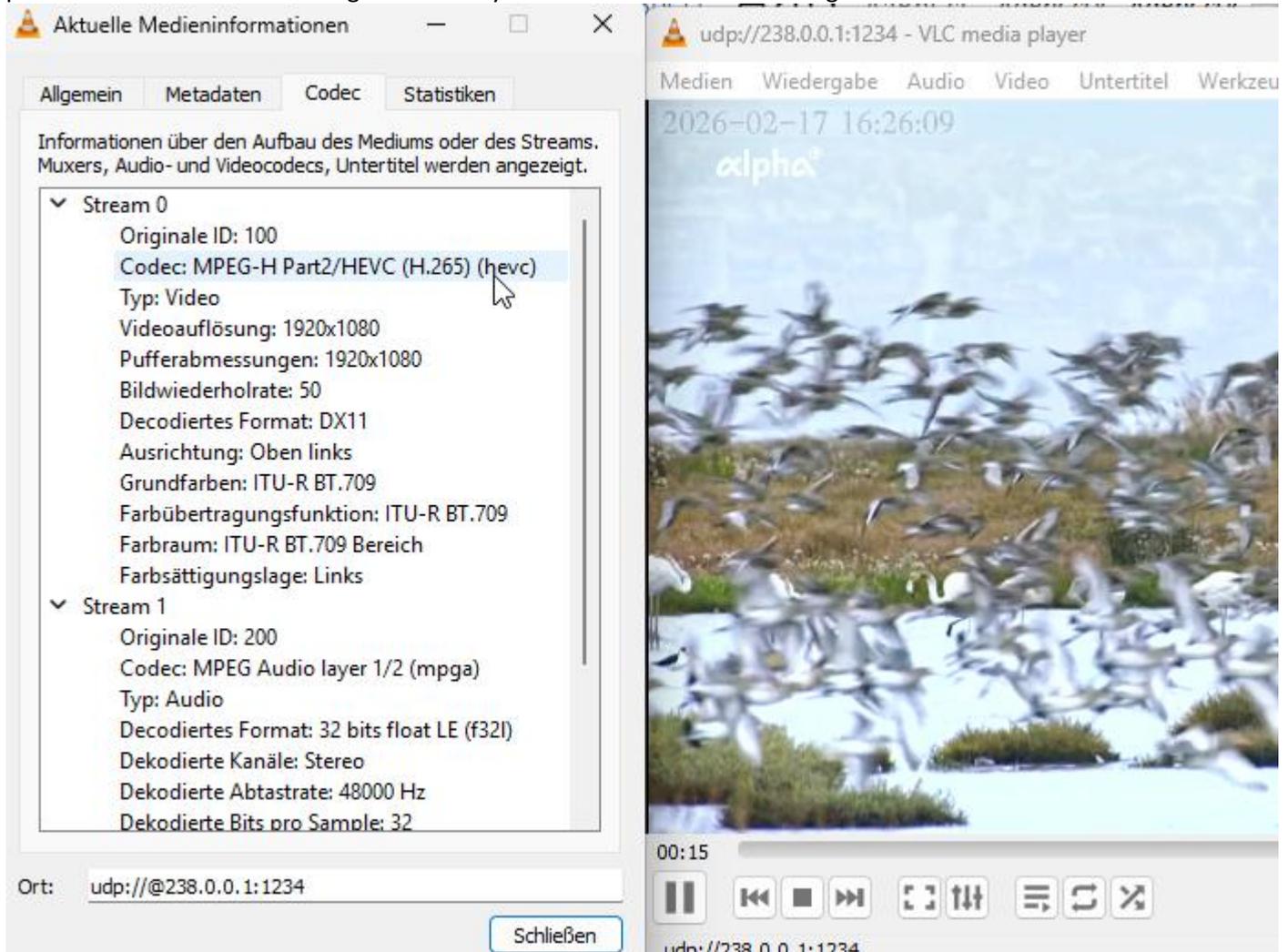
71%

1.00x 00:32/---



The Ambarella chip supports only Main and High

profile of the h.264 codec. Using h.265 is only 1 Profile so it cannot be changed:



A nice overview for that profile stuff can be read here:

[H.264 profiles and levels | Inside & Outside MediaCoder](#)

High Efficiency Video Coding (HEVC), also known as H.265 and MPEG-H Part 2, is a proprietary video compression standard designed as part of the MPEG-H project as a successor to the widely used Advanced Video Coding (AVC, H.264, or MPEG-4 Part 10).

See also: [ITU MyWorspace: ITU-T H.265 \(V3\) \(04/2015\) - High efficiency video coding](#)
[High Efficiency Video Coding - Wikipedia](#)

Bitrate mode:

Video Resolutions:

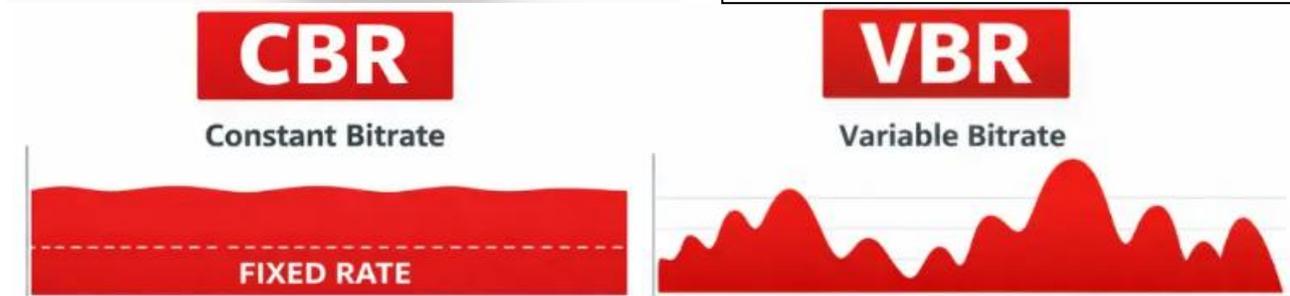
Bitrate Mode:

Bitrate Stable:

- 1(Most Stable)
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10(Least Stable)

CBR is optimal for live streaming because it offers reliability in quality and bandwidth usage. VBR is optimal for on-demand content because it is more economical in storage space while maintaining better quality per size ratio. But increasing latency.

CBR = stable, predictable upload → best for live streaming and platforms that expect a steady bitrate (including Dacast live ingest). VBR = bitrate flexes by scene complexity → best for VOD / uploads where encoding time is OK and you want better quality-per-MB. We have 10 different stable-Modes you can chose from... to match your requirements... trial and error...



Comes from [CBR vs VBR: Which Bitrate is Best for Low-Latency Streaming?](#)

STRONG CBR is mostly useful for DVB-receivers IP SPTS to Multiplexer-Modulators and inserts also a so called Zero-packet into the stream which is PID 8191dec in the DVB-tables. In combination with this, the PCR-setting might be also adjusted for DVB – IP receivers which per-DVB-Norm are for headends and declares only UDP/RTP Multicast-Streams as Inputs for the DVB-Multiplexer-Modulators:

TS/Multicast settings

The most important for IPTV Systems/receivers:

Status Encoding TS/Multicast RTSP RTMP(S) SRT WebRTC TRTC HLS MP4/FLV/Other

Encoder

Main Stream

Substream1

Substream2

Substream3

Video

Audio

Advanced

- Multicast address can also be entered in the broadcast unicast address field. The system will automatically detect it.
- If VLC cannot play multicast, please check whether the computer has multiple network cards and try switching to another one or disabled the others.
- Multicast SAP name is visible on supported devices (e.g., VLC playlist → Local Network).

TS URL:

Multicast IP:

Multicast Port:

Multicast SAP Name:

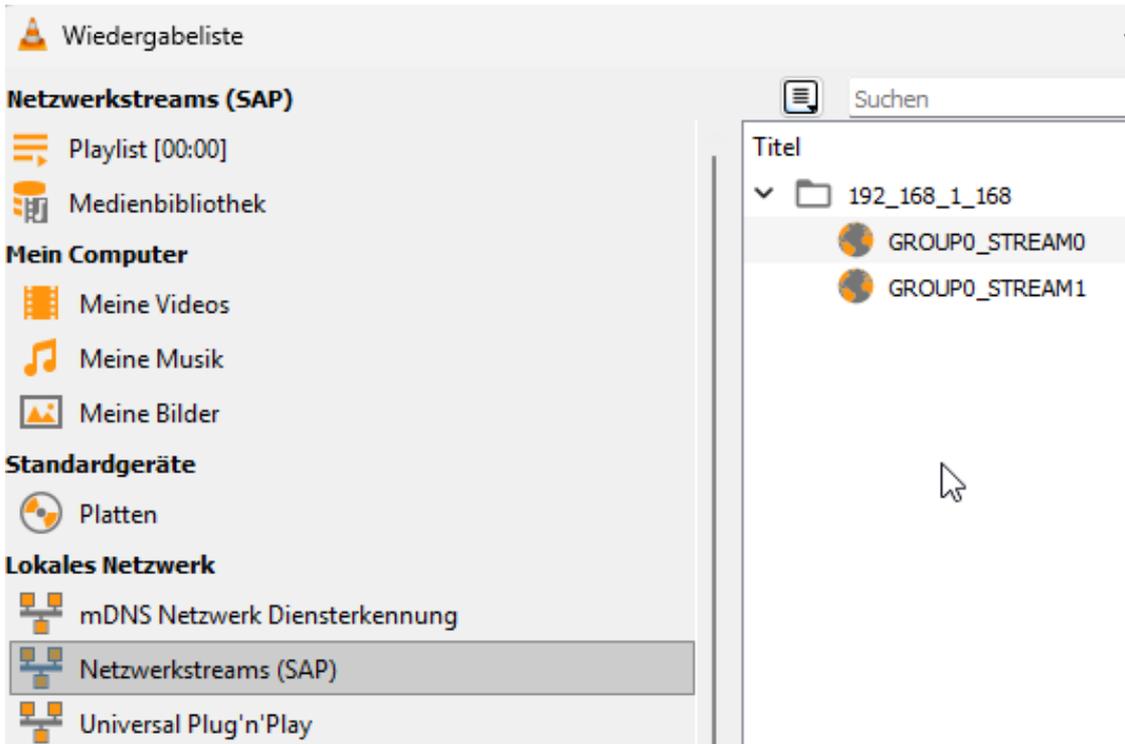
Please consider IGMP for your managing network Switches (Layer3) if you stream multicasts.

The IP addresses should be always different in a common network!!!, Port numbers should be considered according to IANA: [IPv4 Multicast Address Space](#) and [50 Common Ports You Should Know - GeeksforGeeks](#)

SAP/SDP is the 'Session Announcement Protocol' which broadcasts the network multicast streams and is very helpful to grab them.

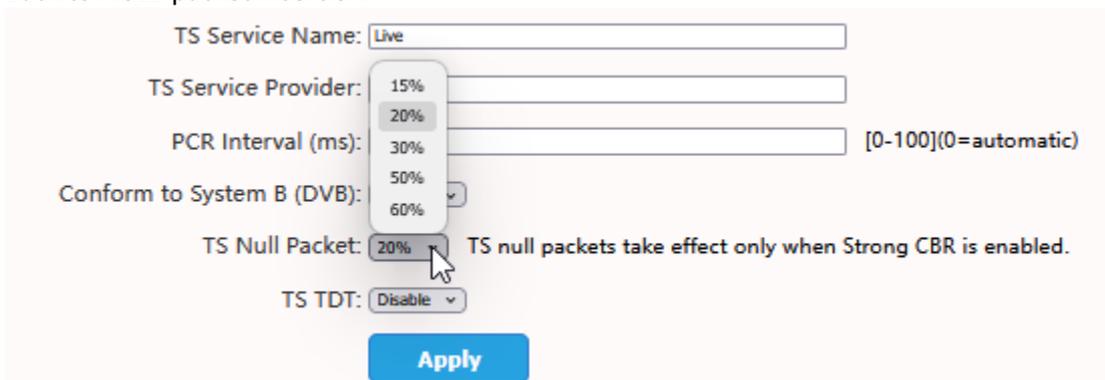
[Session Announcement Protocol - Wikipedia](#)

Example by VLC (and you can create a mu-file from it ... but that's not the topic here):



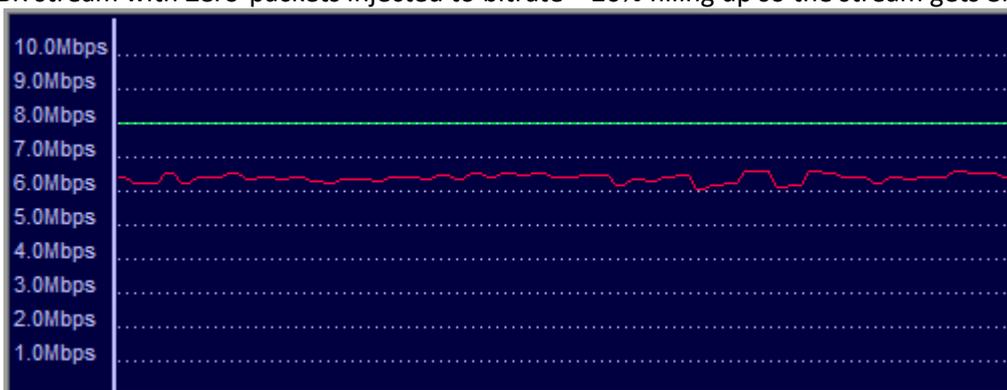
You can directly click and it opens the stream...

Back to NULL-packet insertion:



20% should be used

Example for a CBR stream with Zero-packets injected to bitrate + 20% filling up so the stream gets 8Mb/s:



While the content is nearly CBR but fluctuations are normal for moving pictures

Important as well as the PCR interval should be considered for DVB IP receivers: max 40ms are recommended for DVB-IP-TS:



So, it might be worth to play with PCR settings if you face problems with a DVB-IP-receiver.

**Depending on model/type -----Technical changes are subject to change w/o notifications*

System B DVB according to ITU Annex-B is the derivate for US-CATV units and can be enabled if needed for an EDGE QAM modulator in DVB-C Annex-B. The rest of the world (except Korea) is using the 'normal DVB-C Annex A/C standard.

You should have knowledge about the DVB-Tables to change the Streaming parameters...:

- ◆ In most cases, TS advanced settings do not need to be adjusted.
- ◆ For DVB systems, set the TS null packet rate to 20%. The device will insert null packets to maintain a stable bitrate. This applies only in "Strong CBR" mode.

TS Video PID: [16-8190]
 TS Audio PID: [16-8190]
 TS ID: [1-65535]
 TS Original Network ID: [1-65535]
 TS PMT PID Start: [16-7936]
 TS PID Start: [32-3840]
 TS Tables Version: [0-31]
 TS Service ID: [1-65535]
 TS Service Name:
 TS Service Provider:
 PCR Interval (ms): [0-100](0=automatic)
 Conform to System B (DVB):
 TS Null Packet: TS null packets take effect only when Strong CBR is enabled.
 TS TDT:

The parts which are necessary to have a valid table framework for Multicasts and also Unicasts containing so called PID's. The table version should not be changed unless you know what you are doing.

PAT Program Associating Table is a must have as well as

SDT = Service descriptor Table

Video PID

and Audio PIDs (Audios can be several in a stream if multilanguage and /or Dolby is send)

Those basic tables containing essential information about the Stream and its components like TS-ID, ONID, links to the Audio/Video and their codecs in use and so on.

DVB: PID 0, 17, 18 EIT, and others are reserved and cannot be changed...

Comparing with a DekTec analyser makes it visible for you (set to decimal PID-numbers instead Hex):

In the SDT above you see the values from the encoder: ONID, TS-ID, Service-ID, types, name of the broadcast (live...). The PCR PID comes along with the Video PID and as you can see above, with Strong CBR and 20% zero packets inserted, it is in Auto-Mode with 2ms – which is quite perfect.

Set the encoder to VBR-mode:

Video Codec:

FPS:

GOP:

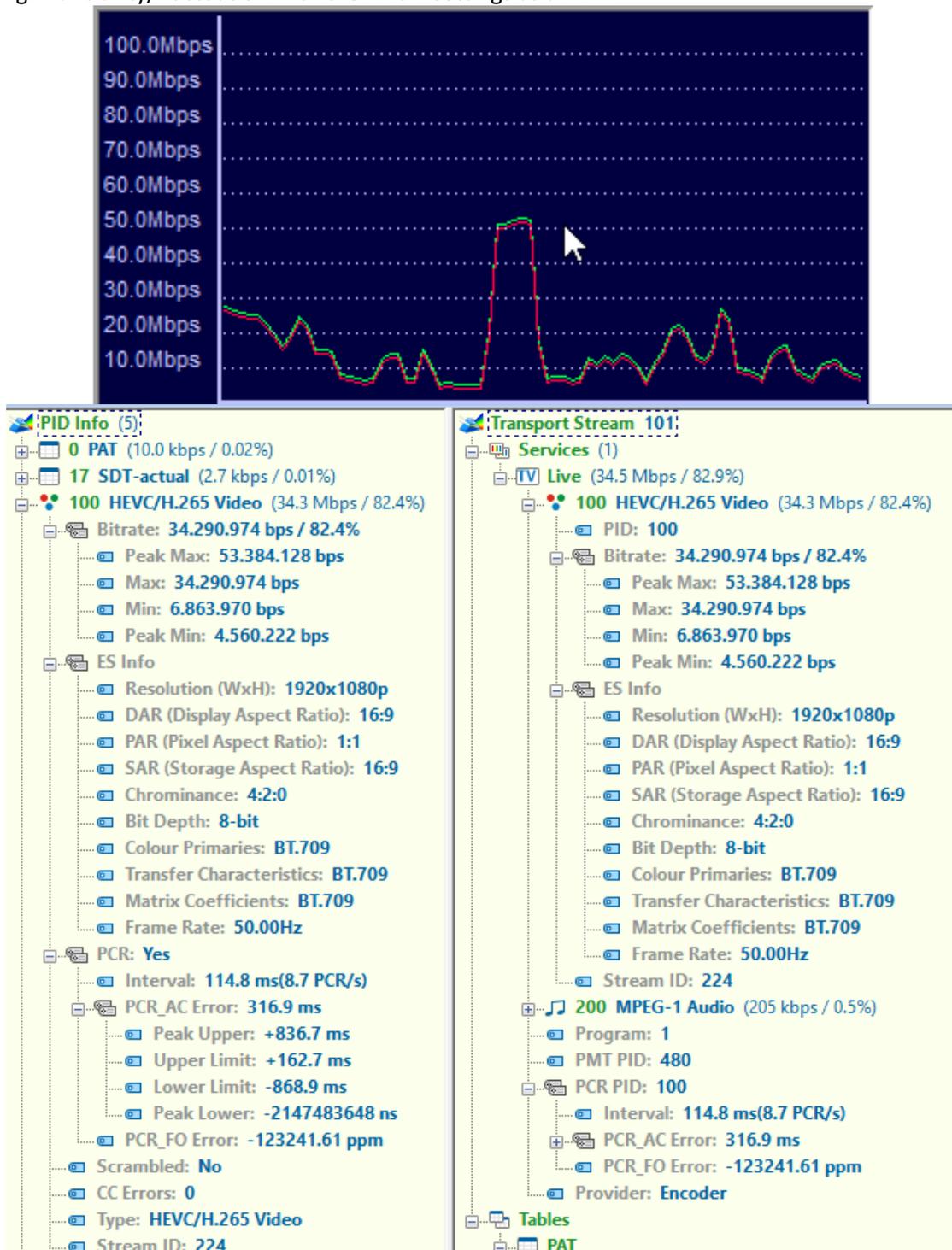
Bitrate (kbps):

Image Quality Levels:

Video Resolutions:

Bitrate Mode:

Shows the high variability/fluctuation with the 'BEST' settings as a VBR:



The TS – part of this Menu

TS URL: Enable ▾

Multicast IP: Enable ▾

is for the unicast Transport-Stream as HTTP-Mode which can be inside a UDP while the handling is based on http / TCP mode:

Real-time FPS: 50.10

TS URL:

HLS TS URL: Disabled

Audio:

Before we come to the other UNICAST streams, we come to the **common** AUDIO settings because the encoder sets it for all Main and Secondary streams:

Status

Encoder

Main Stream

Substream1

Substream2

Substream3

Video

Audio

Advanced

OSD

System

Audio ONVIF Audio

◆ Do not set the Digital Volume Gain too high, as this may cause distortion or introduce noise. In most cases, a gain of 0 is sufficient. Adjust the front-end audio volume to control the overall sound level.

Audio Input: ▾

Sampling Rate: ▾

Audio Codec: ▾

Audio Channel: ▾

Audio Bitrate (bps): [64000~320000]

Digital Volume Gain (dB): [-20~10]

Denoise: ▾

Denoise (dB): [1~100]

G711 Over PTE: ▾

Apply

Most encoder types have external 3.5mm Stereo jacks for analogue audio inserting instead using SDI embedded or HDMI. VGA or DP encoders always have external inputs. For SDI and HDMI we recommend to use the Digital in like as the default settings – see above.

Audio Input: ▾

Sampling Rate: ▾

Audio Codec: ▾

Sampling Rate: ▾

Audio Codec: ▾

Audio Channel: ▾

We recommend to follow the input (HDMI/SDI).

Audio Codec: **MP2** ▼

Audio Channel: AAC

Audio Bitrate (bps): AAC++ [64000~320000]

Digital Volume Gain (dB): MP2 [-20~10]

Denoise: AC3

Opus

Different embedded Codec can be used. For DVB its almost either MPEG1Layer2 Stereo (MP2) or AC3 while AAC will work, but the extended AAC + and ++ need to be checked by the stream client in use. Every codec have different audio bitrate settings to be considered:

Audio Codec: **AAC++** ▼

Audio Channel: **L+R** ▼

Audio Bitrate (bps): **256000** [14000~64000]

This needs to be harmonized when changing it. See above, 256000 is not in the AAC++ range. But you'll get a hint:

Samplng Rate: Same as inp

Audio Codec: AAC++ ▼

Audio Channel: L+R ▼

Audio Bitrate (bps): 256000

Digital Volume Gain (dB): 0

Denoise: Enable ▼

🌐 192.168.1.168

Audio Bitrate must be an integer between 14000 and 64000.

Weitere Aufforderungen von 192.168.1.168 verbieten

OK

Volume and Denoise:

Digital Volume Gain (dB): **0** [-20~10]

Denoise: **Enable** ▼

Denoise (dB): **15** [1~100]

G711 Over PTE: **Disable** ▼

Apply

As the web-content shows: The digital Volume increasing might cause audio distortions. You should balance this with your input Audio Volume coming form your input source (HDMI/analog/SDI). Use the denoise function if needed. G711 Over PTE (which is our proprietary multicast protocol should not be mentioned here.

As well as ONVIF support – this is a special case for NVR's which are using this to monitor and control Cameras – and so our encoders can work like a streaming camera device. More in our www.blankom.de websites-tutorials and so...

Audio [ONVIF Audio](#)

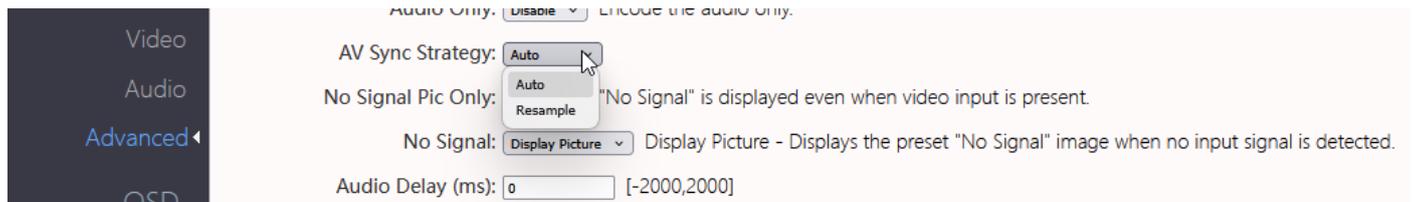
◆ When connecting to an NVR, if the RTSP stream uses G711 encoding, please select the "Enable and resample with 8 kHz" option.

G711 Over RTSP: **Disable** ▼

G711: **G711A** ▼

Apply

The Audio Lipsync should be set to automatic in the ADVANCED Menu:



But if set to **resample** it might glitch over the time... and maybe manually corrected by the user -2000...+2000 ms.

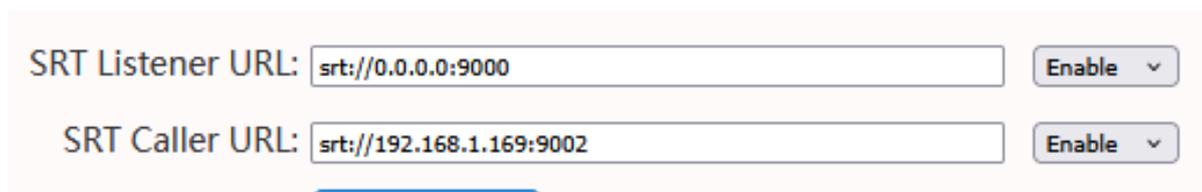
Using SRT Listener/Caller modes:

Rem: VLC only supports the caller mode – so it calls the encoder stream (listener) on demand by entering the simple **srt://<IP-Address>:9000** (for the main-encoder part as set to default) when enabled.

The listener mode works vice versa: In the encoder you need to setup the target streaming address e.g. a Decoder-IP address:port number in the encoder web-menu:

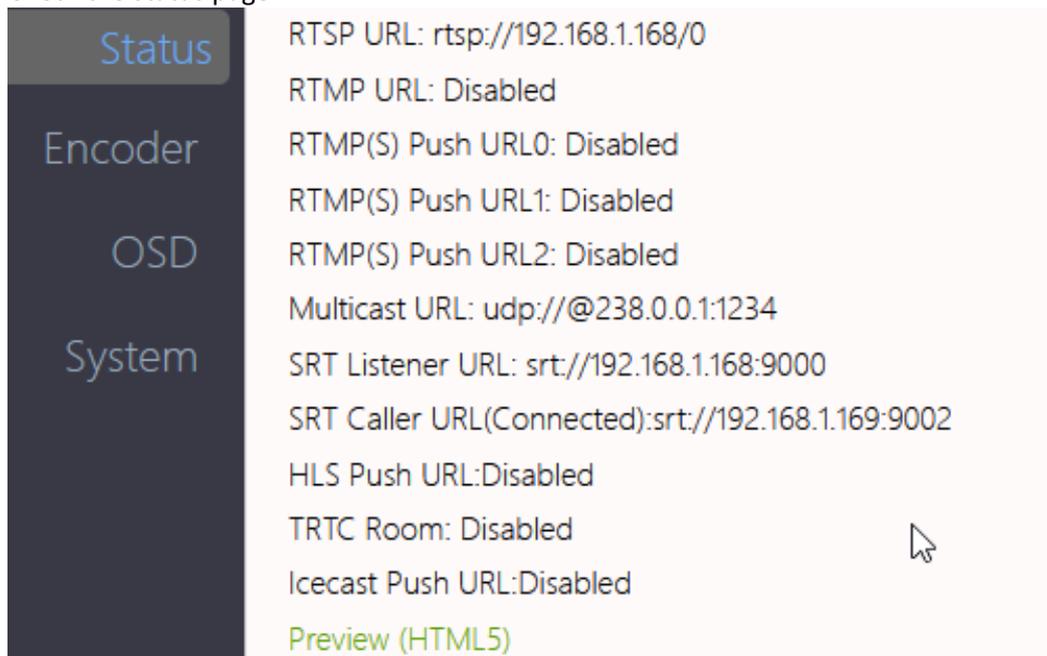
The target like the HDD-Decoder or other stream receivers like our Protocol changing device BIG-1050/1100X need to assign the port-number and declare it as 'waiting-for a stream pushed by the sender according to FFMPEG commands:

Encoder settings – push mode: Encoder IP-Address is 192.168.1.168, Decoder is 192.168.1.169

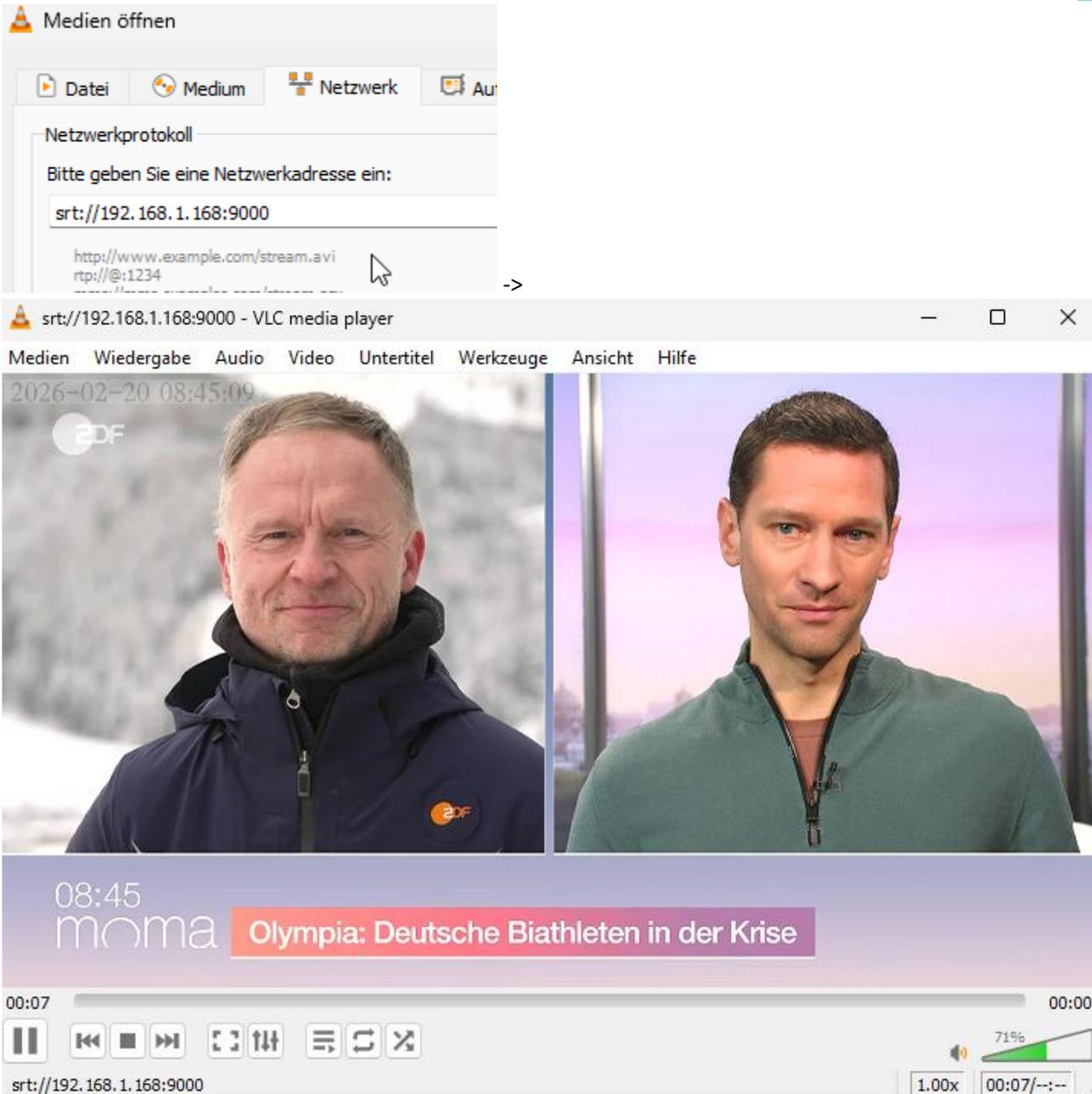


Encoder as Listener needs only these 0.0.0.0* as IP address because it automatic uses its own one as the target from the Caller (the receiver-decoder). The port-address can be chosen and should not interfere with other network port settings in use.

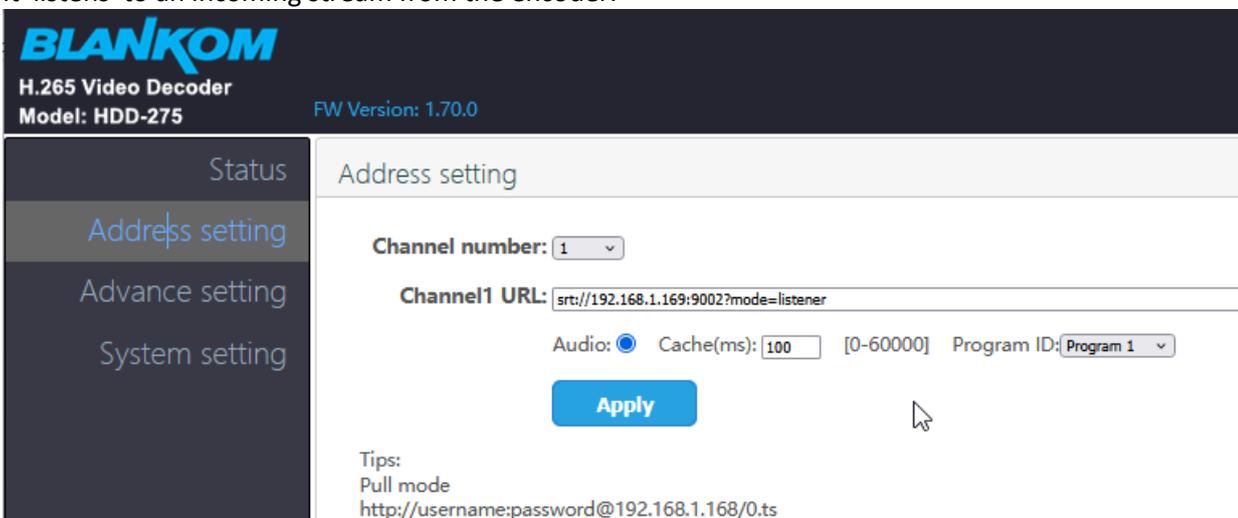
Check the status page:



Pull the Stream by a receiver like VLC (**caller** from the listener (encoder)):



As mentioned before, VLC only supports this mode while the encoder can do both. Here the Push-mode: It 'listens' to an incoming stream from the encoder:



And the encoder STATUS changes from not-connected to connected if the handshake was successful:

SRT Listener URL: srt://192.168.1.168:9000
 SRT Caller URL(Connected):srt://192.168.1.169:9002
 HLS Push URL:Disabled

Decoder settings IP=192.168.1.169: Follow the hints given in the web-menu:

Status Encoding TS/Multicast RTSP RTMP(S) **SRT** WebRTC TRTC HLS MP4/FLV/Other

Encoder

Main Stream

Substream1

Substream2

Substream3

Video

Audio

Advanced

OSD

System

- In SRT Listener mode, a local port is opened for players to pull the stream.
 Example: By default, the stream is available at `srt://0.0.0.0:9000`. Play it in VLC using `srt://device-ip:9000`.
 Example (with a streamid): `srt://0.0.0.0:9000?streamid=channel100`, then play it in VLC using `srt://device-ip:9000?streamid=channel100`.
- Example with a passphrase (minimum 10 characters): `srt://0.0.0.0:9000?passphrase=0123456789`
 Example: Configure the SRT stream with a streamid and passphrase. `srt://0.0.0.0:9000?streamid=channel100&passphrase=0123456789`
- In SRT Caller mode, the stream is pushed to an SRT server.
 Example: `srt://live.example.com:9000?streamid=channel100&latency=120&maxbw=8000000`
- The device supports the SRT parameters listed below. Refer to the SRT protocol specification for detailed definitions.
 - streamid** - Stream ID: A unique identifier used by the receiver to distinguish and access the SRT stream.
 - passphrase** - Encryption Passphrase: The password used to generate the AES encryption key for the SRT stream.
 - pbkeylen** - Encryption Key Length: Length of the AES encryption key in bytes (16 / 24 / 32).
 - latency** - Latency: Target delay for smoothing playback, in milliseconds (20–8000 ms).
 - maxbw** - Maximum Bandwidth: Maximum allowed stream bandwidth in bytes per second.
 - rcvbuf** - Receive Buffer: Size of the SRT socket receive buffer in bytes.
 - sndbuf** - Send Buffer: Size of the SRT socket send buffer in bytes.
 - ipttl** - IP Time-To-Live: The number of hops that SRT packets are allowed to traverse.
 - conntimeo** - Connection Timeout: Maximum time (in milliseconds) to wait for establishing an SRT connection before aborting.

SRT Listener URL: Disable ▾

SRT Caller URL: Enable ▾

Apply

Both can work in Caller or Listener mode. You always have to consider the view-direction from the Source and from the Destination (Receiver) as both cannot be caller at the same time or Listener at the same time. If the Streamer is set to Listener, the Receiver need to be the caller and vice versa.

Other protocols HLS/RTMP:

As HLS and also RTMP(s) are supporting also push and pull modes, mostly used for streaming to a Video-Platform in the internet (Youtube/Vimeo, ...) those settings /tokens/certificates/user:passwords need to be clarified with the Media-server access before starting to configure:

- PUSH URL 0–2 are streaming addresses. They can push to RTMP(S) / RTSP servers.
- This device has a built-in RTMP server that supports H.264 streaming.
 To view the stream in VLC, open the URL: `http://device-IP/live/stream name` to play the video.

RTMP URL: Disable ▾

RTMP(S)/RTSP PUSH URL0: Disable ▾

RTMP(S)/RTSP PUSH URL1: Disable ▾

RTMP(S)/RTSP PUSH URL2: Disable ▾

- ◆ When HLS TS or MP4 is enabled, the device buffers segments in memory according to the duration and count set in Advanced settings. Setting both to 3 helps reduce RAM usage.
- ◆ The HLS publish URL sends the stream to servers such as YouTube. Paste the URL in this field.

HLS TS URL: Disable ▾

HLS MP4 URL: Disable ▾

HLS PUSH URL: Disable ▾

Maybe worth to mention, that HLS is one of the most Latency adding protocol because of its architecture. In Advanced settings those can be adjusted for HLS:

HLS Segment Duration (s): [3-20]

HLS Segment Quantity: [3-20]

RTSP/HTTP-streaming

are the oldest existing streaming protocol while both are handling by TCP and RTSP can use RTP as a on top if UDP inside. We do not explain those in detail here but you can have a look to our Website: <https://www.blankom.de/tutorials.html> and <https://www.blankom.de/anleitungen-deutsch.html> and you might find the right documents for those....

- ◆ Enable - The actual transport (UDP or TCP) is chosen by the client. For example, select "RTP over RTSP (TCP)" in VLC to use TCP.
- ◆ Enable (TCP only) - Forces the device to use TCP.
- ◆ UDP may occasionally drop packets on some networks. Switch to TCP if all clients support it.
- ◆ Enable (Multicast) - RTSP multicast uses 4 consecutive ports starting from the configured RTSP Multicast Port.

RTSP URL: Enable ▾

For RTSP you should chose which encoder part to be used for it in Advanced Settings:

RTSP Default Stream: Main Stream ▾

vMix Compatible: Disable

TS Over RTSP: Main Stream

Multicast Type: Substream1

Substream2

Substream3

Also, the Transportstream-type content can be assigned:

TS Over RTSP: TS ▾

Multicast Type: ES

TS

as TS or Elementary Stream.

ONVIF

Is supported as version 2.8 and maybe higher. Please check the internet about what's best to use like a NVR or the free ODM (Onvif Device Manager freeware for Windows):

<https://www.blankom.de/assets/downloads/odm-v2.2.250.msi>
or 'Milestone' as a ONVIF Camera Software tool.

Audio
 Advanced ◀
 OSD
 System

Onvif Service: ▾
 Onvif Location:
 Logger Service: ▾
 Loki Server Address: ▾
 Loki Server Port: [1-65500]

Location can be set... as it will be shown by ODM...
 And it can be assigned a password for it:

Onvif Password: ▾
 No HTTP Password: ▾

as well as this can be done also for HTTP and RTSP to 'secure' the streams because the dedicated receiver needs to know this to open the stream:

TS/RTSP Password: ▾
 Onvif Password: ▾
 No HTTP Password: ▾

If enabling, the encoder user:password are taking effect also for the streams (default: admin:admin)

Loki:

If you need to log all actions of the encoder, you can install a Loki-Server in your net and address it. See picture above.

Others:

MTU Size: [500-8000]
 NAT Address:
 Ethernet Speed: ▾
 UDP TTL: [1-254]
 UDP Socket Buffer Size: (0-20971520)

can be adjusted

This can do a correction if you have e.g., a 16:9 input and do a 4:3 output resolution (Downscaling) to correct it:

PAR: ▾ (DAR = SAR x PAR)

- Disable
- 3:4 (16:9 → 4:3)
- 4:3 (4:3 → 16:9)
- 16:15 (720x576 → 4:3)
- 16:9 (720x576 → 16:9)
- 64:45 (720x576 → 16:9)
- 8:9 (720x480 → 4:3)
- 32:27 (720x480 → 16:9)
- 9:16 (16:9 → 1:1)
- 3:4 (4:3 → 1:1)
- 2:1 (8:9 → 16:9)

Self-explaining:

Kick All on Input Change: When enabled, all connected players will be disconnected if the audio or video input changes.

No Video Auto Restart: If no video is detected, the encoder will automatically reboot after 3 minutes.

No Audio Auto Restart: If no audio is detected, the encoder will automatically reboot after 3 minutes.

Screen Freeze Auto Reboot: If a screen freeze is detected, the encoder will automatically reboot after 1 minute.

Deblocking Enable:

Deblocking Alpha: [-6~6]

Deblocking Beta: [-6~6]

Lock All Configurations:

Color Range:

Video Only: Encode the video only.

Audio Only: Encode the audio only.

Manipulation of the Picture: Rotation, Flip/Mirror and crop it

Status

Encoder

Main Stream

Substream1

Substream2

Substream3

Video

Audio

Advanced

OSD

System

Video Input

◆ Rotation is available only when the input format is YUV420.

Video Rotation:

Flip / Mirror:

Video Crop:

Video Crop X Coordinate: [0,1920]

Video Crop Y Coordinate: [0,1080]

Video Crop Width (pixels): [0,1920]

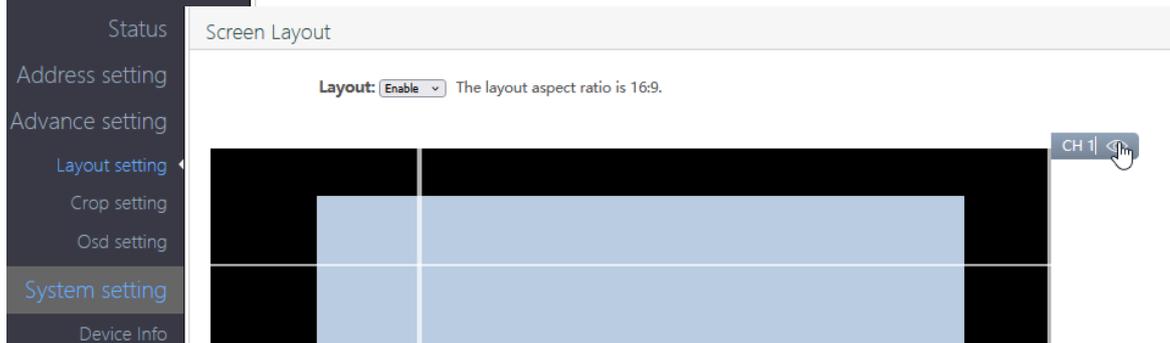
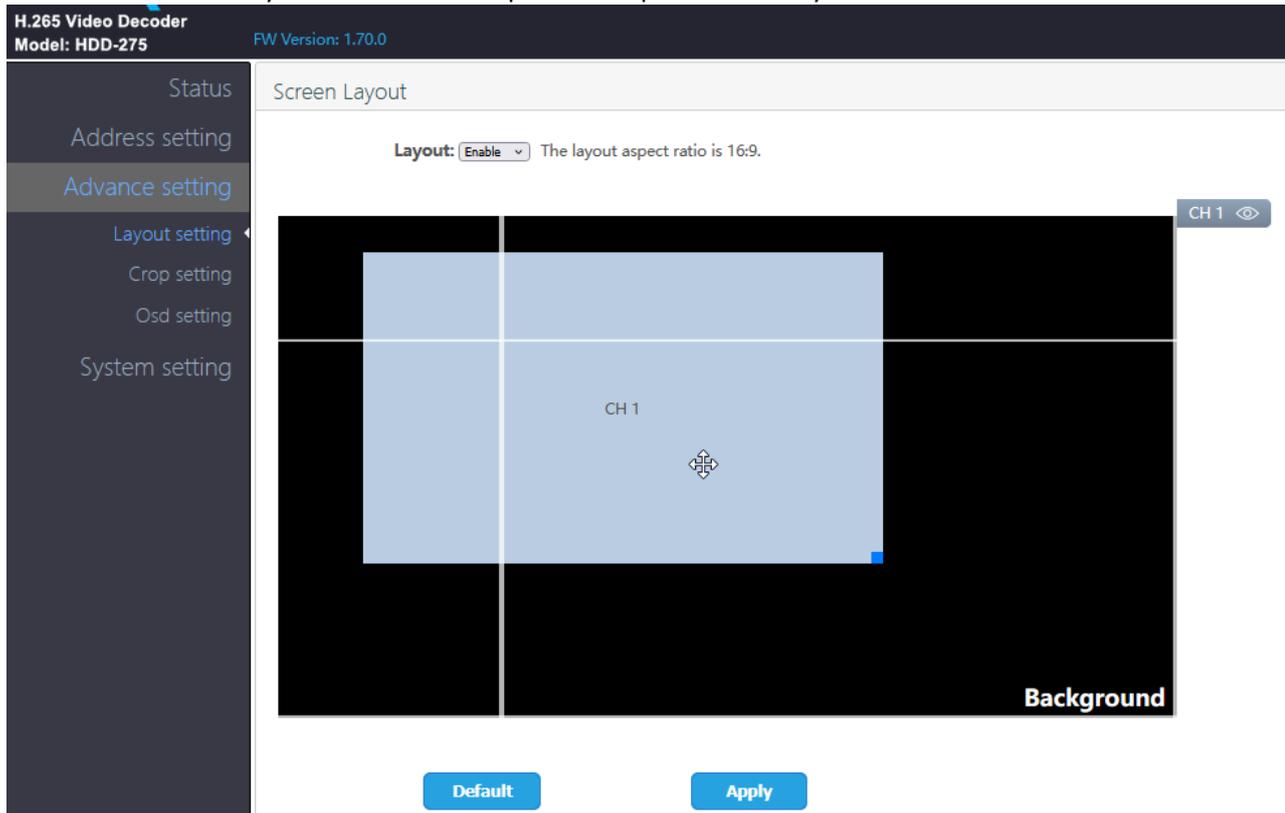
Video Crop Height (pixels): [0,1080]

Monochrome:

Apply

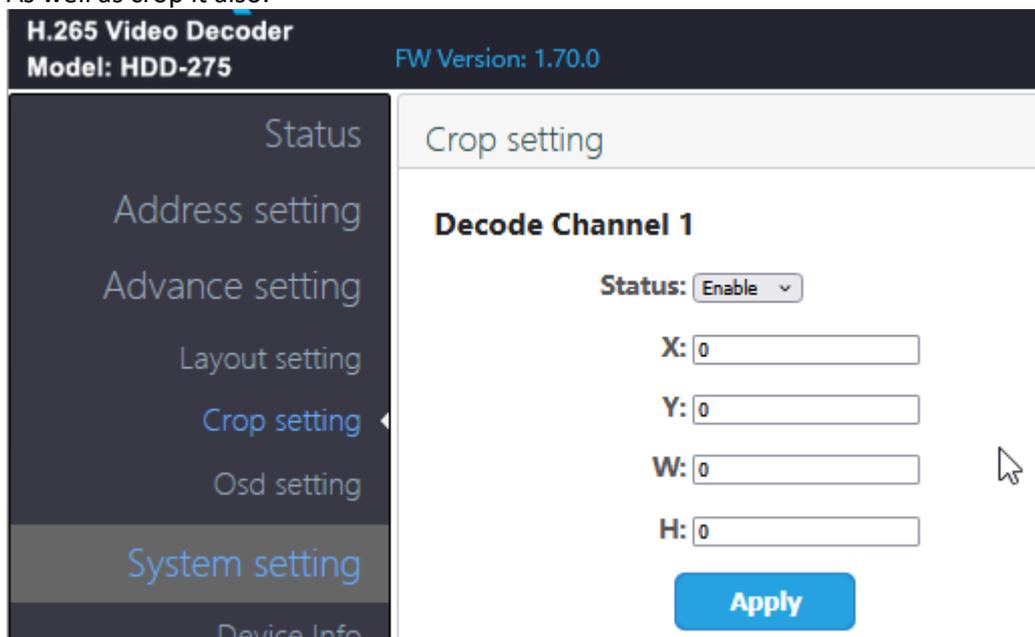
If you have more questions, do not hesitate to contact us by sending an email to: contact@blankom.de

In the HDD Decoders you can also set the picture output individually:



Here you can also adjust which Input stream should be placed where...if you use the 4 Inputs and Mosaic to the output.

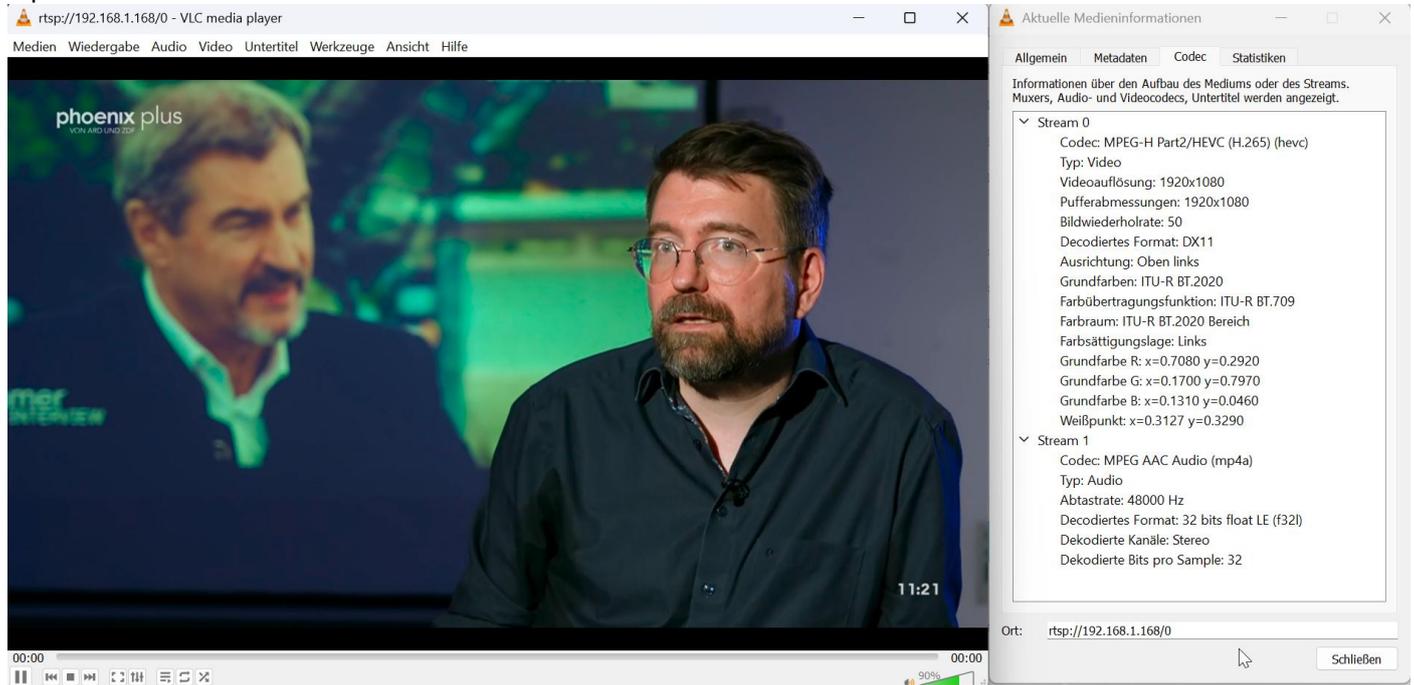
As well as crop it also:



Annex 4Kp60 decoder/transcoder HDD-276 operation:

This unit supports decoding up to 4 input streams to a 4-Mosaic 4K picture on the HDMI-Out and Transcoding to stream to an IPTV network output:

Input from encoder:



Set to transcode it:

Status

Input stream address

Transcoding setting

Advance setting

System setting

Transcoding setting

Encoding Type: H.264

FPS: [5-60]

GOP: [5-300]

Bitrate(kbit): [32-100000]

Encoded Size: 1920x1080

H.264 Level: BASELINE

TS URL: Enable

HLS URL: Disable

FLV URL: Enable

RTSP URL: Enable

RTMP(S)/RTSP PUSH URL: Disable

Multicast IP: Enable

Multicast Port: [1-65535]

Multicast Type: UDP

SRT URL Port: Disable

SRT PUSH URL: Disable

SRT pwd: Disable

Apply

<p>Status</p> <p>Input stream address</p> <p>Transcoding setting</p> <p>Advance setting</p> <p>System setting</p>	<p>Status</p>
	<p>System status</p> <p>Systemtime: 2026-02-19 18:41:27 (Synchronize time to the device)</p> <p>Runtime: 0000-00-00 00:21:47</p> <p>CPU usage: 45%</p> <p>Memory usage: 198MB/975MB</p> <p>net status: local</p> <p>Output format: 1080P60</p> <p>Channel number: 1</p> <hr/> <p>Decode channel1</p> <p>Decode address: rtsp://192.168.1.168/0</p> <p>Decode status: normal</p> <p>Decode type: H265</p> <p>Decode size: 1920x1080</p> <p>Frame rate(fps): 25</p> <p>Bitrate(kbit): 12251</p> <hr/> <p>Transcode channel1</p> <p>Encode type: H.264</p> <p>Encode size: 1920x1080</p> <p>Bitrate(kbit): 3200</p> <p>TS URL: http://192.168.1.169/0.ts</p> <p>FLV URL: http://192.168.1.169/0.flv</p> <p>RTSP URL: rtsp://192.168.1.169/0</p> <p>Multicast URL: udp://@238.0.0.1:1234</p>

Decoder output stream is now:

The screenshot shows the VLC media player interface. The main window displays a video stream with the text "destens drei Werke" and "WISO". The "Aktuelle Medieninformationen" (Current Media Information) window is open, showing the following details:

- Stream 0:**
 - Codec: H264 - MPEG-4 AVC (part 10) (h264)
 - Typ: Video
 - Videoauflösung: 1920x1080
 - Pufferabmessungen: 1920x1088
 - Bildwiederholrate: 29.970030
 - Decodiertes Format: DX11
 - Ausrichtung: Oben links
- Stream 1:**
 - Codec: MPEG AAC Audio (mp4a)
 - Typ: Audio
 - Abtastrate: 48000 Hz
 - Decodiertes Format: 32 bits float LE (f32l)
 - Dekodierte Kanäle: Stereo
 - Dekodierte Bits pro Sample: 32

The location is listed as "Ort: rtsp://192.168.1.169/0".