NEW

SHDE-4000

4Kp60 Encoder & IPTV Streamer with 12G SDI & HDMI Input

h.265 and h.264 compatible 4Kp60 UHD encoder & IP streamer combined with RECORDING function

4K H.265 HDMI 12G SDI Video Encoder – Parallel operation of SDI & HDMI input at the same time, maximum encoding 1x 4K60 and 1x 1080p60 in parallel. **Multiprotocol-Support**:
Unicast: SRT / HTTP / HLS / FLV / RTSP / RTMP / RTMPS,
Multicast: UDP/RTP Stream Protocols and ONVIF. The Encoder works perfectly with online live broadcast platform's, such as Vimeo, YouTube, Facebook, Ustream, Twitter, etc…

- **12G SDI-BNC and HDMI compatible input for encoding and recording to USB 3.0 in NTFS/exFAT/FAT32 format, with SAMBA server mounting built-in for network connection to the recorded files**
- **Stereo Audio embedded or external Input (3.5mm stereo) / out**
- **HD Resolution 2160p60, 1080p, 1080i, 720p…**
- **GbE IP output: RTSP, RTMP(s), UDP/RTP, HTTP, HLS, FLV, MJPG, SRT**
- **Distribution of Video Camera U(HD) and other sources content over LAN, WAN or internet.**
- **2 simultaneously and independent Live stream broadcast encoder engines to multiple destinations (Main + Secondary)**
- **Video-over IP applications, Digital Signage, NVR, Hotel Info-channel**
- **IPTV/OTT applications, Video conferencing, Camera streaming**
- **IPTV on LAN applications, Corporate IPTV for Broadcastings**
- **UHD, HD and SD video encoding and downscaling**
- **Corresponding product: BLANKOM IPTV-STB 6800+ (UHD)**
- **Motion-JPEG encoding and SRT streaming protocol**

BLANKOM SHDE-4000 encoder serves the distribution of SD/HD and UHD TV/video content through IP networks in digital quality. The live video can be received by Video media server, TV sets with IPTV Set-Top Boxes, on PC's and tablets with VLC Player.

BLANKOM SHDE-4000

IPTV encoder is designed for TV signal distribution in excellent quality over LAN and INTERNET.

The h.264 and h.265 (HEVC) compatible compression technology features low-latency and low bit rates for IPTV/OTT systems. The high-efficient encoding chips saves bandwidth cost through all its resolution range.

Distribution of SD up to UHD TV channels through the IPTV/OTT network using state-of-art IP technology from almost any kind of video input.

Excellent video and audio quality.

High reliability. No regular service and maintenance need during operation.

info@blankom.de www.blankom.de

Rear Side view

- Dual Power Supplies (PSU's) with IEC connectors, inbuilt Fuse, Power-switch.
- GbE LAN connector RJ45
- HDMI and 12G SDI Inputs, Additional external 2 pairs of Stereo Inputs by 3.5mm Jacks.

Application: Parallel Encoding and streaming from HDMI and SDI

Available also in a boxed version *upon request:*

Applications:

- IPTV (Wowza, nginx, Media Server, ReStreamer2, etc.)
- Online Live Broadcast (OBS, Vimeo, YouTube, Facebook, Ustream, Twitter, etc.)
- NVR (Network Video Recorder), ONVIF Software cathexisvideo.com, … https://www.onvif.org/
- Recording System / https://medevel.com/10-cctv-open-source-solutions/
- Digital Signage
- Teaching / Campus Broadcast
- Hotel TV system
- Video Conference

Product features

- User friendly English Web based interface, default IP is 192.168.1.168.
- Multiprotocol Support SRT / HTTP / HTTPS / HLS / FLV / RTSP / RTMP / RTMPS / UDP/RTP Unicast/Multicast) TS Stream Protocols and ONVIF.
- Support output of Multi RTMP(s) (Main stream & Sub stream) to different Media Streaming Server.
- 4K: 4096x2160@60fps / 3840x2160@60 / 2560x1440@144fps etc., input and encoding.
- Max. Colour key support 4:2:2 10 bit.
- 1 click recording: Support USB 3.0 in NTFS/exFAT/FAT32 format, with SAMBA server built-in, and video-files splitting adjustment with easy access through SAMBA network reviewing and operation.
- Text and image inserting as Message/Logo Overlays for Main-& Sub-streams.
- Output Stream bitrates adjustable.
- Additional External Analog audio input & output. Audio digital gain control and adjustment
- Remote control via router port forwarding

Specifications:

Interfaces	
Video Input	HDMI & 12G SDI input, Max support 4096x2160@60fps / 3840x2160@60 / 2560x1440@144fps,etc.
Audio Input	HDMI & SDI embed audio & External Analog audio input
Ethernet	1000Mbps, RJ-45
Video Encoding	
Video Format	H.264/AVC High/Main/Baseline ProfileH.265/HEVC main profile, MJPEG/JPEG baseline
Video Resolution	Max support 4096x2160@60fps
Video Bitrate	0.1...100Mbps
Video FPS	5-144FPS
Bitrate Control	VBR/CBR/Strong CBR
Streaming Protocols	Unicast SRT / HTTP / HTTPS / HLS / FLV / RTSP / RTMP / RTMPS Multicast UDP/RTP TS Streams
Audio Encoding	
Audio Format	AAC/AAC+/AAC++/MP3/MP2/AC3 G711
Sample Rates	44.1kHz/48kHz
Bitrate	12...640kbps
System	
Control Method	English Web Interface- management – Telnet access can be enabled
Firmware	Ethernet software upgrade, default IP 192.168.1.168, default user/pass= admin/admin
Working environment	
Operating temperature	-10° to 70°C
Storage temperature	–20° to 80° C
Relative Humidity	5% to 90% non-condensing
Miscellaneous	
Dimension (L× W× H)	19'' 1RU 480x 270x40 mm with integrated dual PSU or optional boxed 150x110x45mm
Approx. weight	2400g (boxed 500g w/o PSU)
Power adapter	2x IEC plug AC input 100-240V 50/60Hz internal DC 12V 2A

BLANKOM®

Anmerkung:

Alle von uns veröffentlichten Betriebsanleitungen richten sich an den Antennen- und IT-Fachmann, der über grundlegende Kenntnisse der Empfangs-, Netzwerk- und Anlagentechnik verfügt. Die Einhaltung aller relevanten Vorschriften und Richtlinien für den Aufbau und Betrieb von solchen Anlagen obliegt dem Installateur und/oder dem Betreiber. Insbesondere sind die in den jeweiligen Ländern geltenden Vorschriften und Richtlinien für die Inbetriebnahme speziell für den Stromanschluß und alle mit den Produkten in Zusammenhang stehenden und geltenden Normen und Gesetze einzuhalten.

Remark:

All operating instructions published by us are intended for the antenna and IT specialist who has basic knowledge of reception, network and system technology. Compliance with all relevant regulations and guidelines for the installation and operation of such systems is the responsibility of the installer and/or the operator. In particular, the regulations and guidelines applicable in the respective countries for commissioning, especially for the power connection, and all standards and laws related to the p must be complied with.

Annotation:

Tous les modes d'emploi que nous publions sont destinés aux professionnels de l'antenne et de l'informatique qui ont des connaissances de base en matière de réception, de mise en réseau et de technologie des équipements. Le respect de toutes les réglementations et directives pertinentes pour l'installation et l'exploitation de ces systèmes relève de la responsabilité de l'installateur et/ou de l'exploitant. En particulier, il convient de respecter les réglementations et directives applicables dans les pays respectifs pour la mise en service, notamment pour le raccordement électrique, ainsi que to normes et lois relatives aux produits.

Annotazione:

Tutte le istruzioni per l'uso da noi pubblicate sono destinate al professionista dell'antenna e dell'informatica che ha una conoscenza di base della tecnologia di ricezione, di rete e delle apparecchiature. Il rispetto di tutti i regolamenti e le linee guida pertinenti per l'installazione e il funzionamento di tali sistemi è responsabilità dell'installatore e/o dell'operatore. In particolare, devono essere rispettati i regolamenti e le linee guida applicabili nei rispettivi paesi per la messa in funzione, soprattutto per il collegamento alla rete elettrica e tutte le norme e le leggi relative ai prodotti.

Anotación:

Todas las instrucciones de uso publicadas por nosotros se dirigen al profesional de la antena y de la informática que tiene conocimientos básicos de recepción, de redes y de tecnología de equipos. El cumplimiento de todos los reglamentos y directrices pertinentes para la instalación y el funcionamiento de dichos sistemas es responsabilidad del instalador y/o del operador. En particular, deben cumplirse los reglamentos y directrices aplicables en los respectivos países para la puesta en marcha, especialmente para la conexión de la energía y todas las normas y leyes relacionadas con los productos.

Anotação:

Todas as instruções de operação publicadas por nós são destinadas ao profissional de antena e TI que possui conhecimentos básicos de recepção, rede e tecnologia de equipamentos. O cumprimento de todos os regulamentos e diretrizes relevantes para a instalação e operação de tais sistemas é de responsabilidade do instalador e/ou do operador. Em particular, os regulamentos e diretrizes aplicáveis nos respectivos países para comissionamento, especialmente para a conexão de energia e todas as normas e leis relacionadas aos produtos devem ser obedecidas.

We do not enter into warranty cases if the device has been misused or wrong handled. This is in particular sometimes the case (often with boxed versions), when connecting wrong AC or DC Voltages to the unit.

TOC

Quickstart:

The connectors are self-explaining:

Front:

USB3.0-port for the Recording – device you like to plug-in
A Manual start-stop Record-Button and several blue Status LEDs: Power, LAN, SDI and HDMI connected, and 2x PSU-status LED's in red.

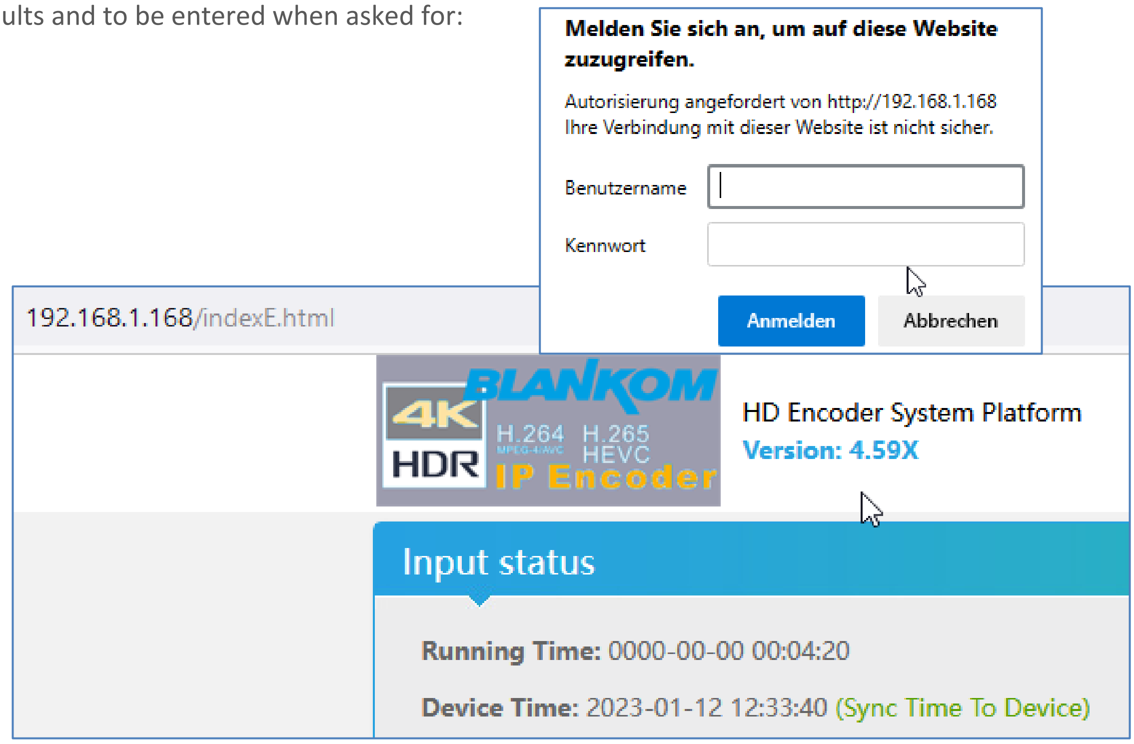
Rear-Panel:

- 2x IEC Power cord connectors 100-240 V AC with Fuses and ON/OFF switches.
- A 'RESET' hole to reset the unit to factory default – just in case you forgot the IP address or other problems occurred.
- HDMI-Connector: Supports up to 4k 2160p60 Input / HDR as well.
- 12G SDI Input BNC Connector
- 2x Audio Stereo external Input to mix into a Video stream encoding (either – or, not adding)
- Gigabit-Ethernet RJ45 connector for the Web-Interface connection / Stream outputs
- Sticker with default data: MAC, Sr.Nr, def. IP, user/pass for that

We recommend to connect the RJ45 GbEthernet port by a DSTP CAT-Cable to a Layer3 Switch with autosensing 100/1000 BaseT and supporting the IGMP v2 or V3 protocol if you like to use/design multicast stream outputs.

Starting the Web-Interface:

After entering the default IP (make sure your computer/browser is configured to be in the same network: 192.168.1.xxx – otherwise you won't be connected. ***admin***/***admin*** as *user*/*password* are defaults and to be entered when asked for:



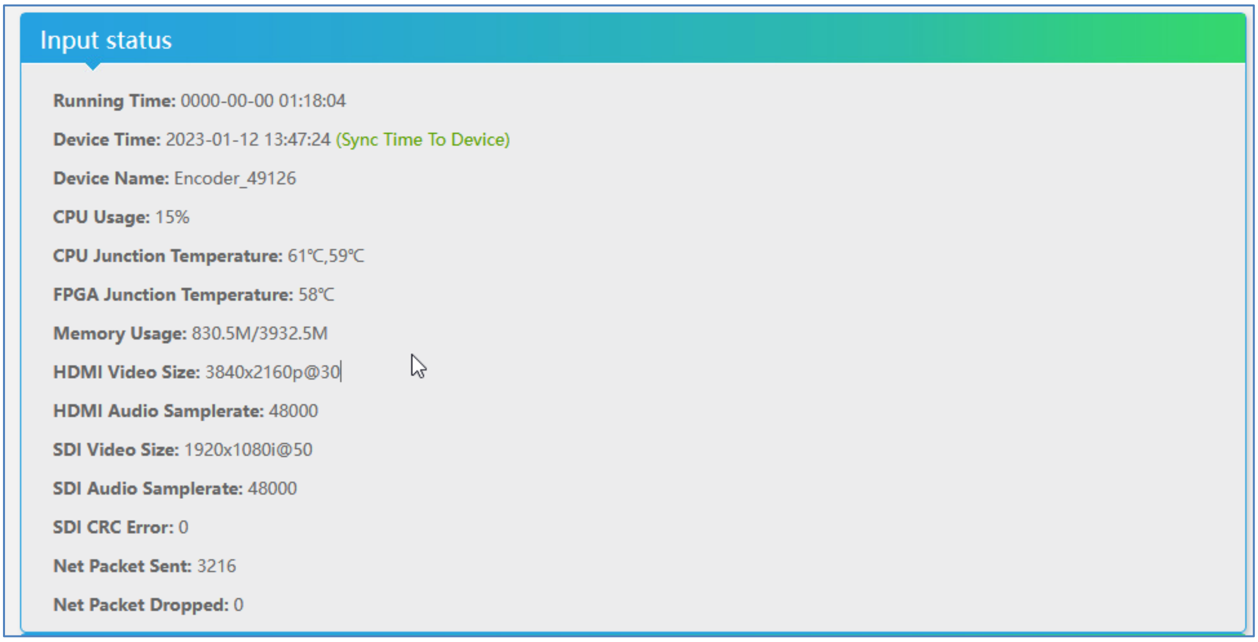
In case you are asking for problem solutions and/or never Firmware releases: Always give us the current installed Version shown in the status window here Version 4.59x

Status-Window and config- Menu parts:

The different menus are accessible through the bottom located parts:

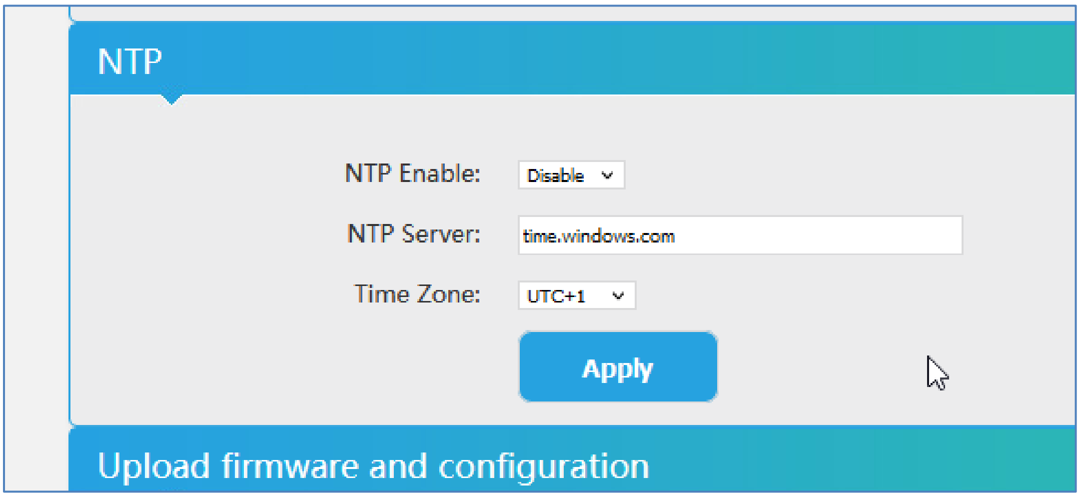
STATUS – NETWORK – Main Stream – Substream – System settings can be switched…

The INPUT Status Frame:



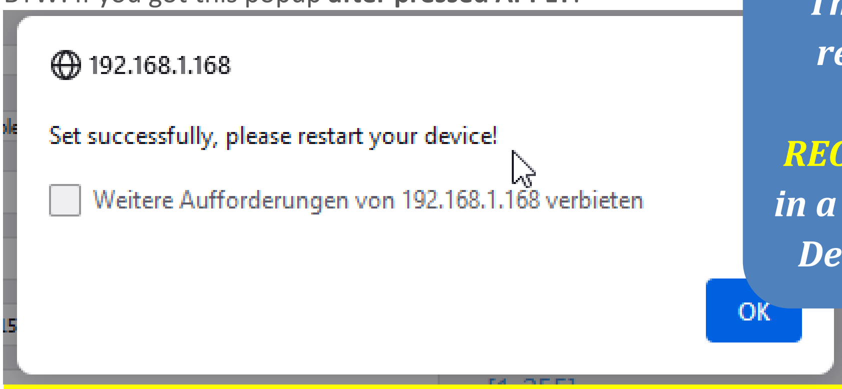
The Input status gives information about the HDMI, SDI and other values. You can press on (Sync time to device) if you have configured the right time zone in the Systems setting scrolling down: (Europe almost is UTC+1 or in Summertime: UTC+2)

NTP:



Always press Apply after you done the changings.

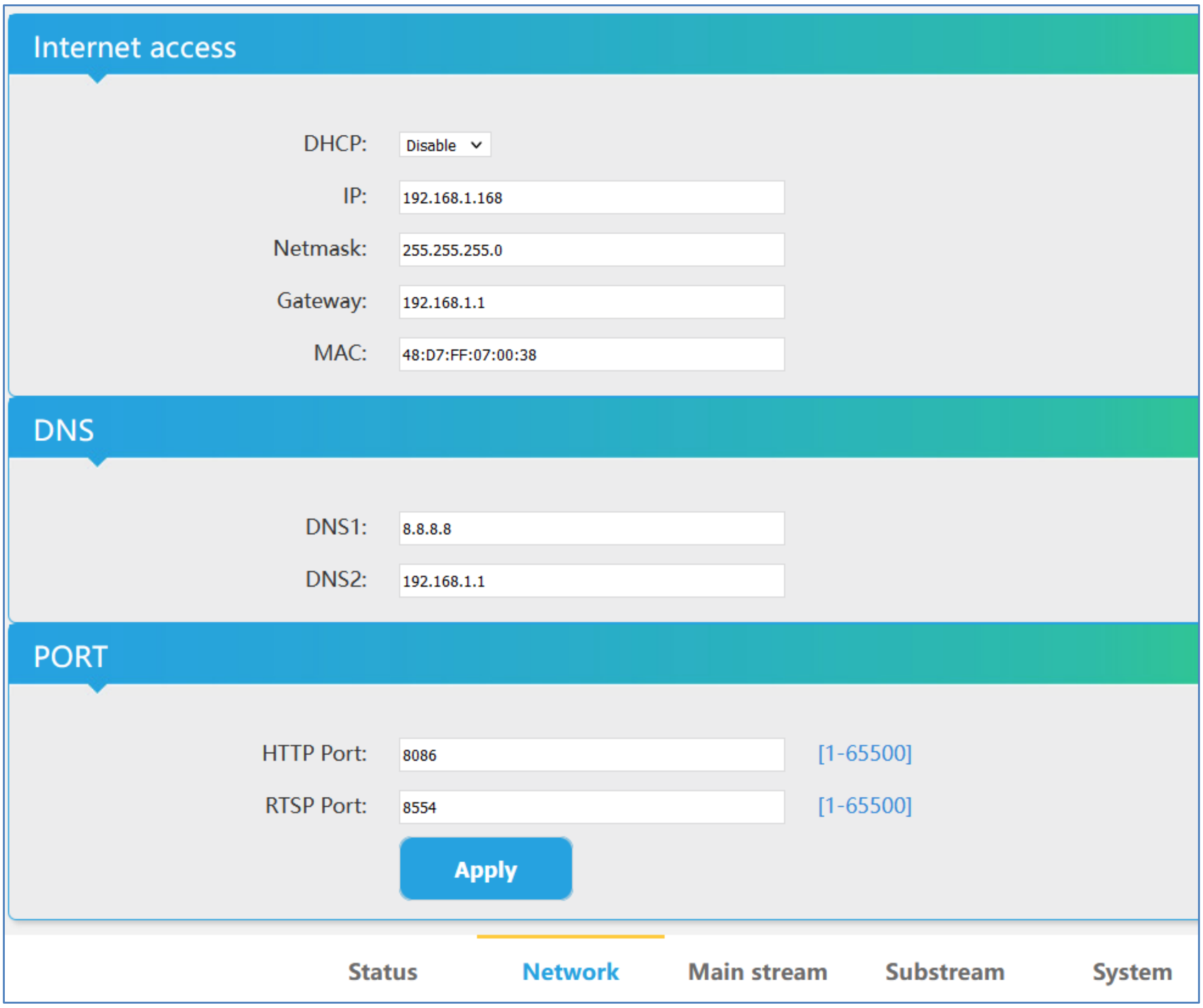
BTW: If you got this popup **after pressed APPLY**:



This does NOT mean to restart the ENCODER-Streamer but your RECEIVER-Device like VLC in a PC or an IPTV STB or a Decoder – Box from us…

Because it almost needs to re-sync on the new values given by you now…

In the
Network section
you can adjust the Encoder to your local Network:



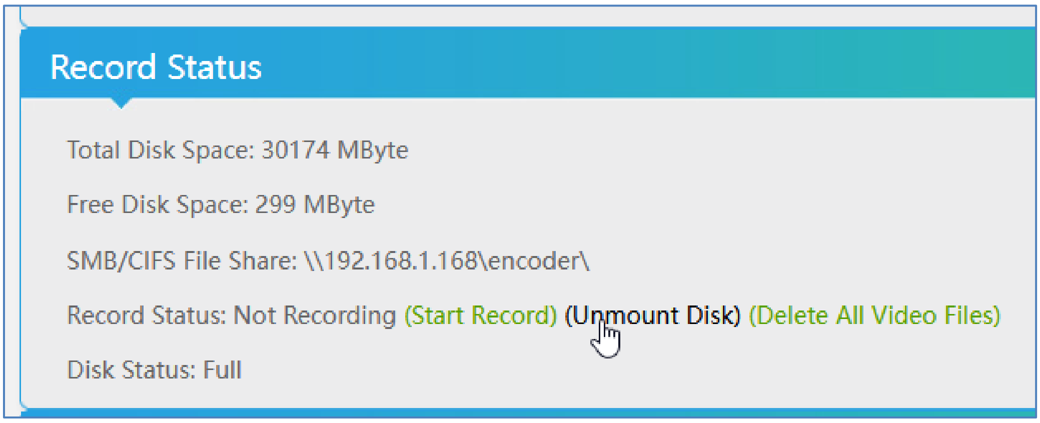
If you choose DHCP, you should check in your DHCP-Server/Router which address has been deployed to the Encoder. We recommend to use a fixed static address and NM, GW settings accordingly.
The **DNS** is needed for NTP settings or/and streaming **to RTMP(s) Media-Servers** which often give you their URI instead of their IP Addresses from the Internet. Make sure these are available.
Example: 8.8.8.8 is a Google driven international Internet DNS-Server while 192.168.1.1. is your local DNS from a Router in your LAN.
If changing the **MAC**, make sure you do not have the same in your internal network – which could cause problems…
The Ports are basically the right ones for the stream outputs by HTTP or RTSP and normally do not need to be changed.

Back to

STATUS-Page:

The recording

part only shows up after you have connected a valid formatted USB-Pen drive or external HDD. If no USB is connected it will disappear.

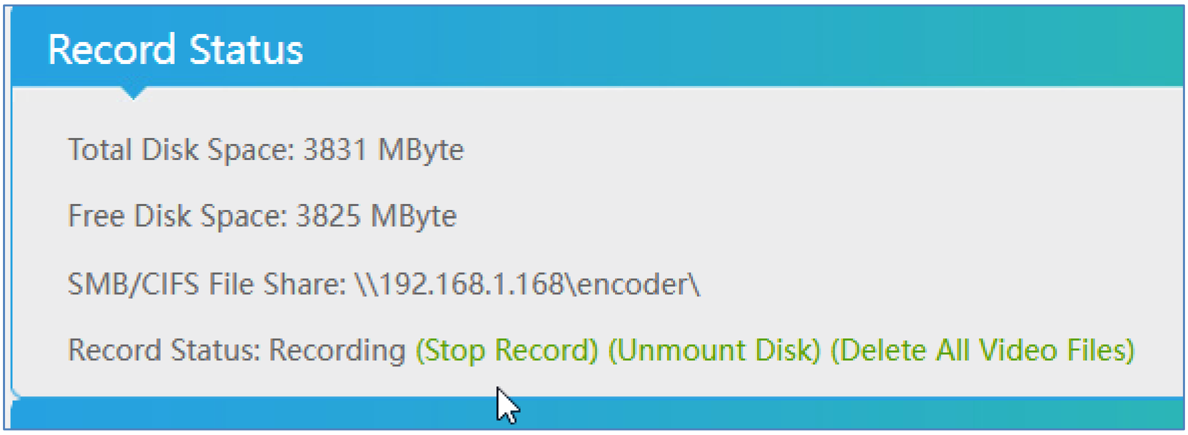


Of course, it shows relevant information and you can start recording (as well with the front-panel-button.

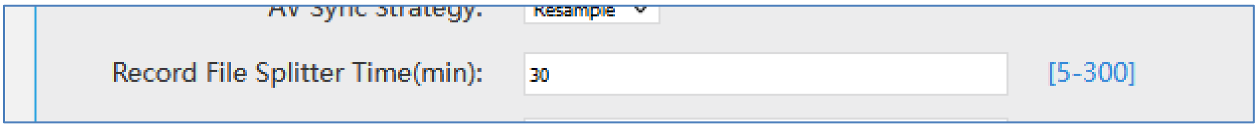
Unmount disk before you want to remove it, please!

We recommend to use USB 3.0 at least because of the recording speed needed by UHD content.

Pressing Start or the button at the front:

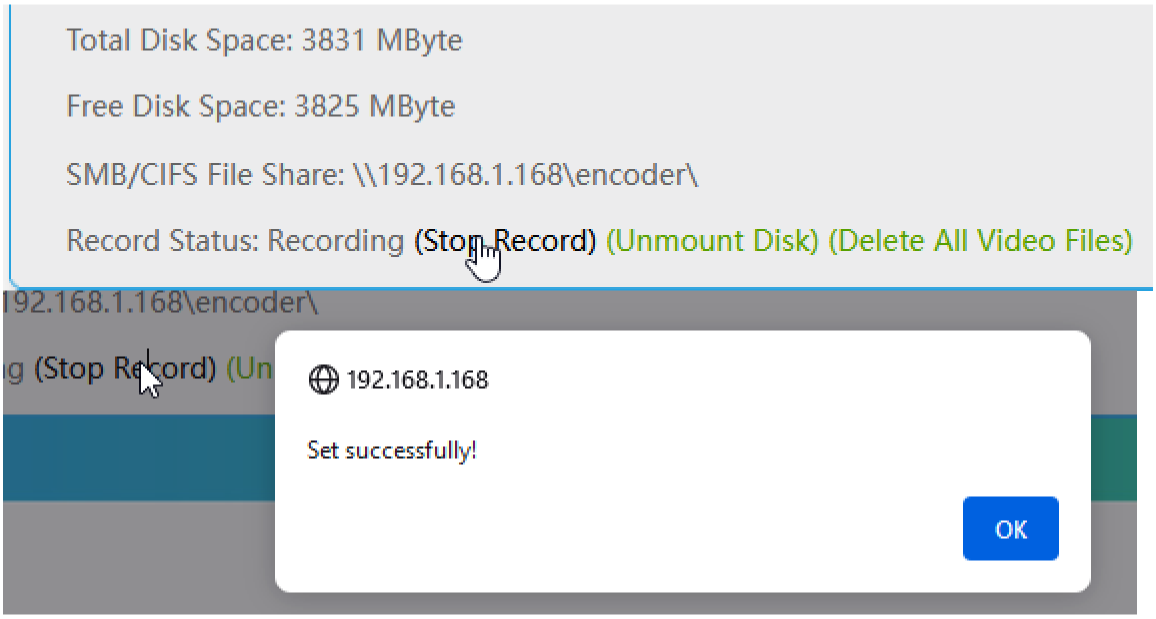


Shows you the actual status.
The TS records will be splitter into files of 30 minutes as defaults -if too big for the USB-used Filesystem its recommended... but you can adjust that in the Advanced System Settings:



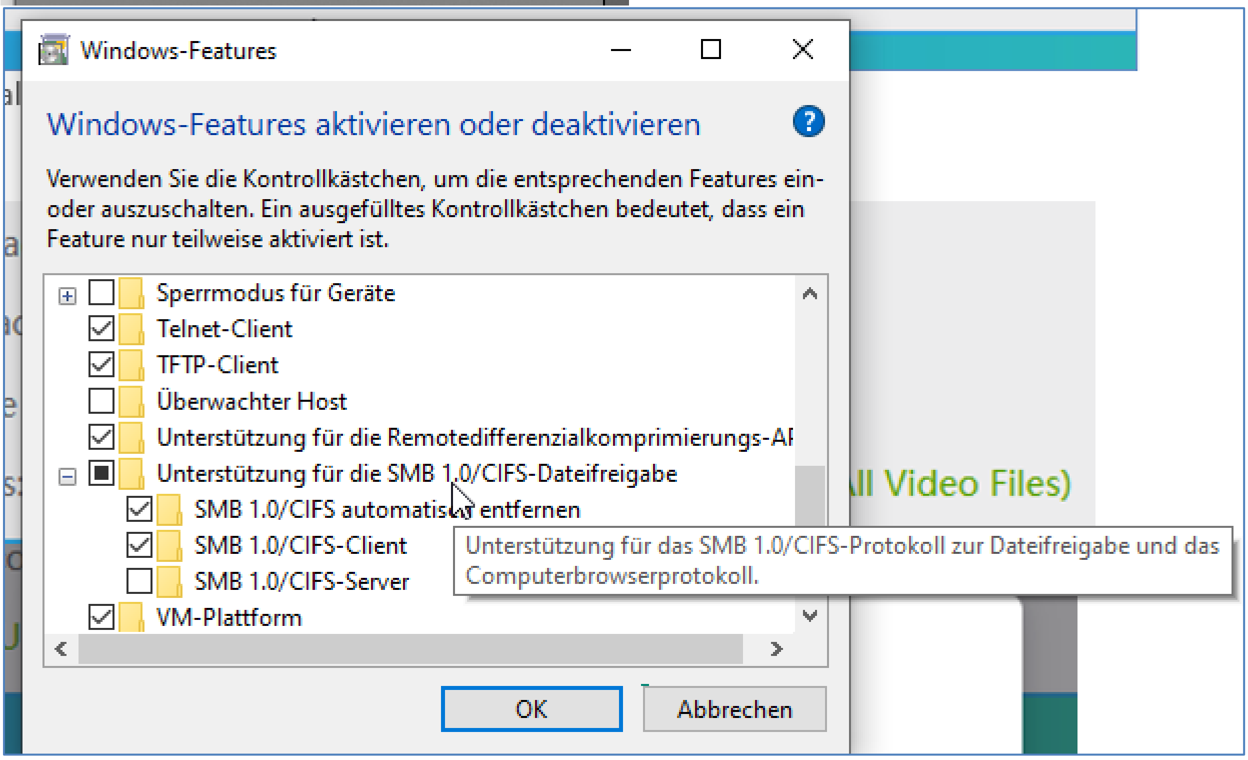
Natively, you cannot store files larger than 4 GiB on a FAT file system. The 4 GiB barrier is a hard limit of FAT: the file system uses a 32-bit field to store the file size in bytes, and 2^32 bytes = 4 GiB (actually, the real limit is 4 GiB minus one byte, or 4 294 967 295 bytes, because you can have files of zero length).
So, you cannot copy a file that is larger than 4 GiB to any plain FAT volume.
exFAT solves this by using a 64-bit field to store the file size but that doesn't really help you as it requires a reformat of the partition. However, if you split the file into multiple files and recombine them later, that will allow you to transfer all of the data, just not as a single file (so you'll likely need to recombine the file before it is useful).

We stop the recording now:

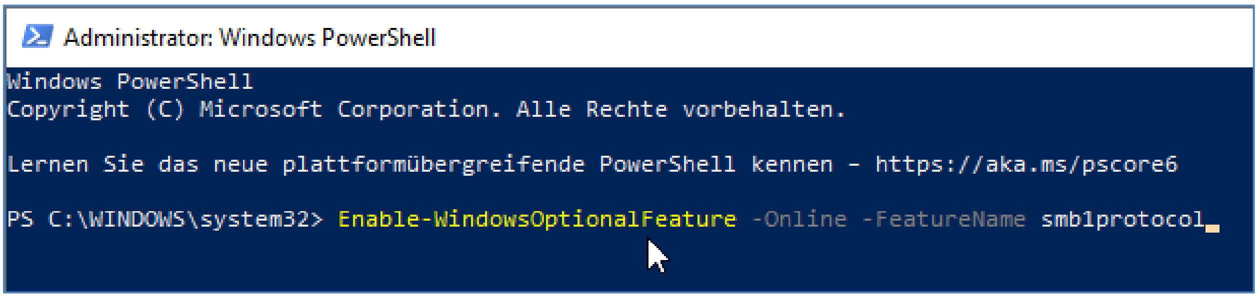


Please confirm that message.

To check or access the USB-PEN by your Windows computer you need to enable **SMB1.0 support** (Windows has disabled that due to a security reason):

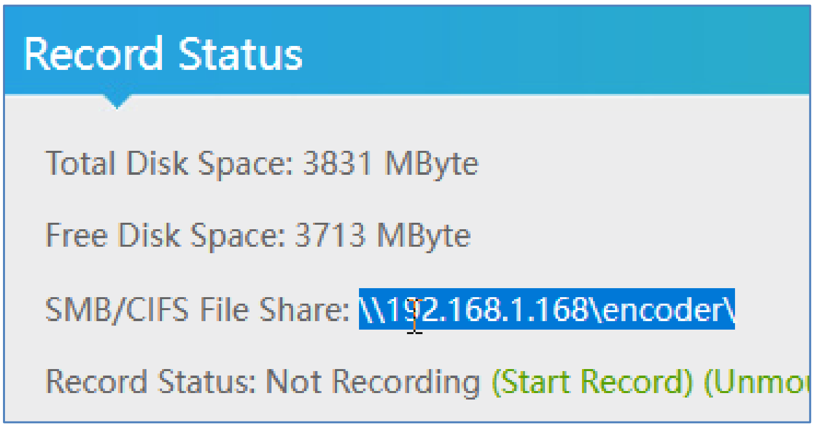


This can be enabled also by **Powershell admin console** and need a reboot of course:
Enable-WindowsOptionalFeature -Online -FeatureName smb1protocol

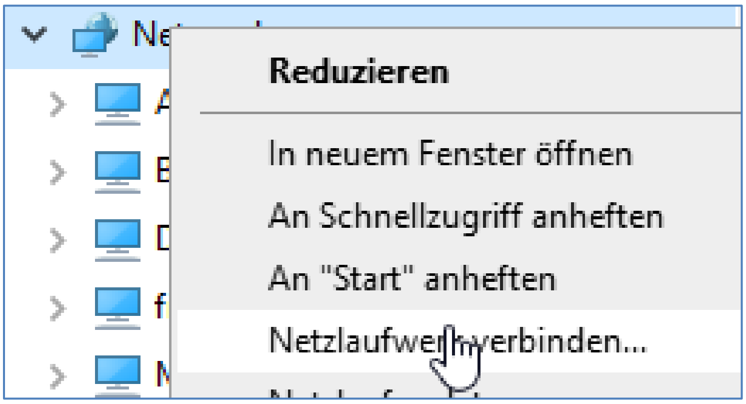


For Linux OS, please google.....

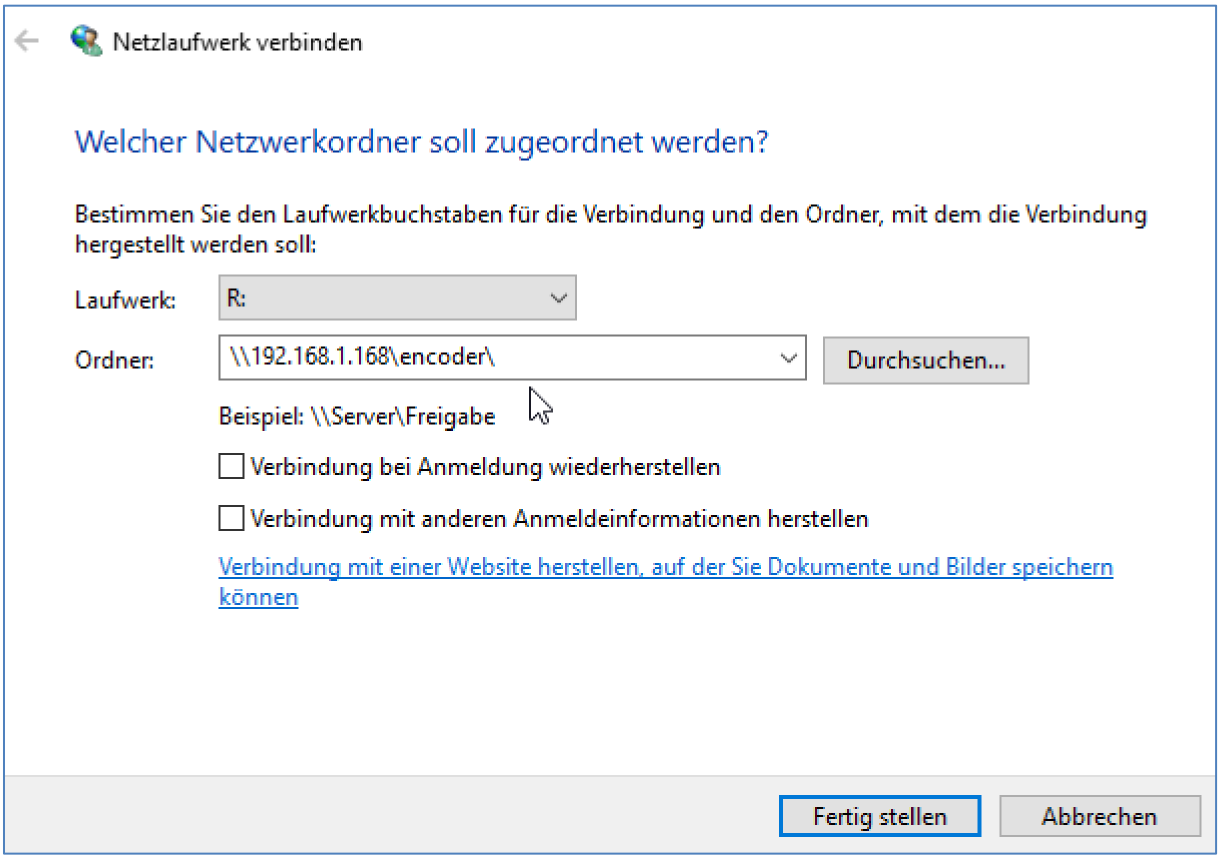
So, if you want to access that USB-Mount:



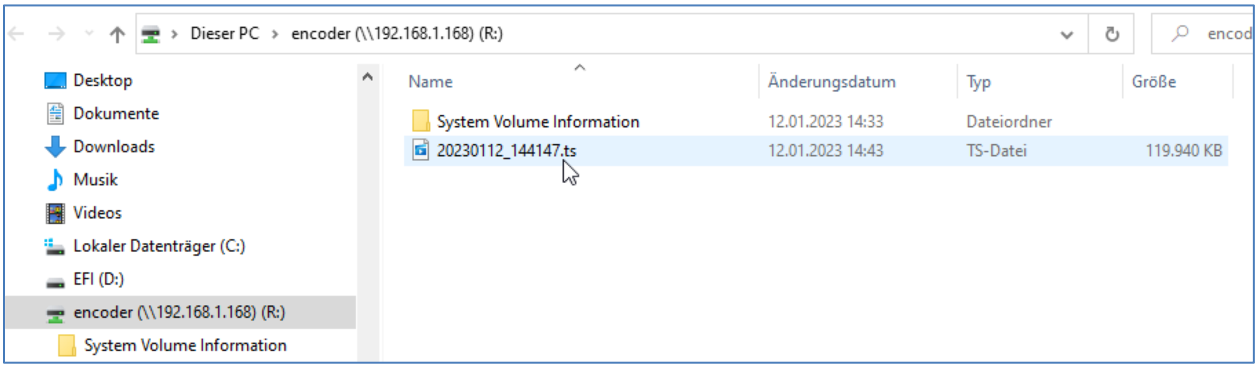
copy that and enter that in your network:



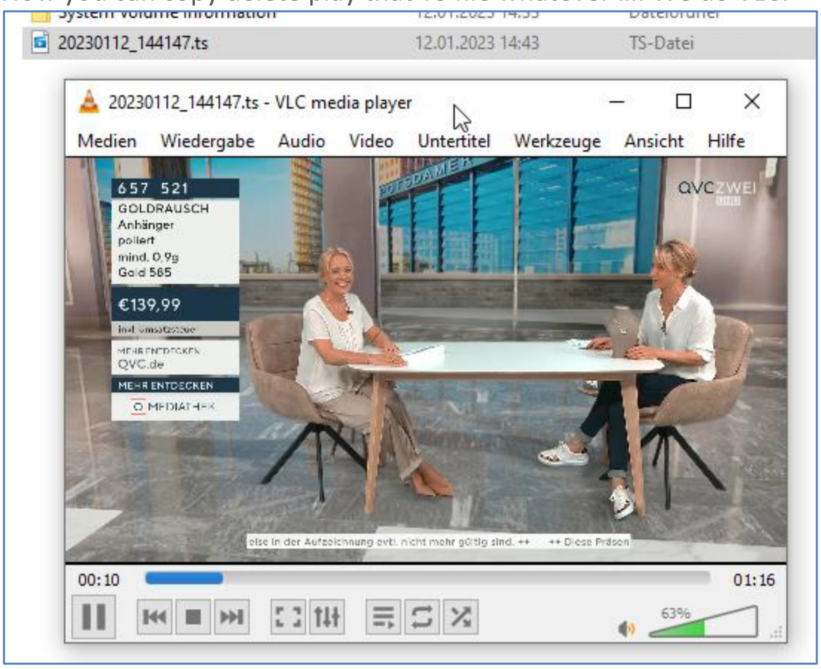
than assign a volume and here we go:



E' Voila:

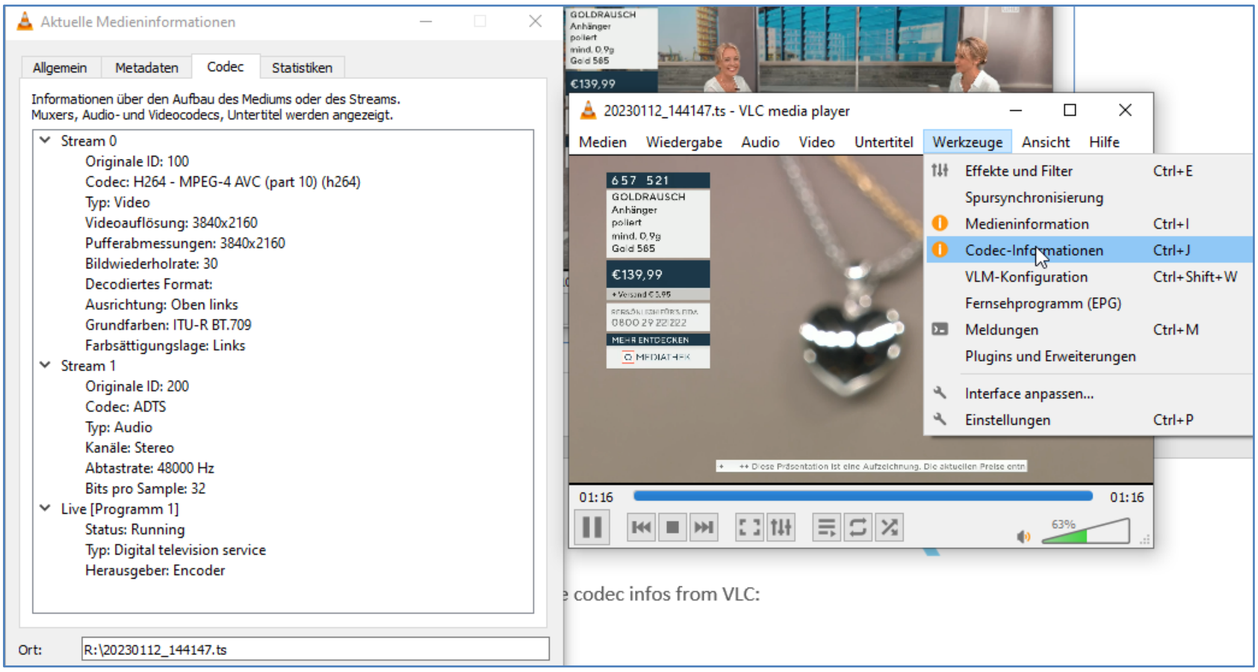


Now you can copy delete play that TS file whatever …. We do VLC:

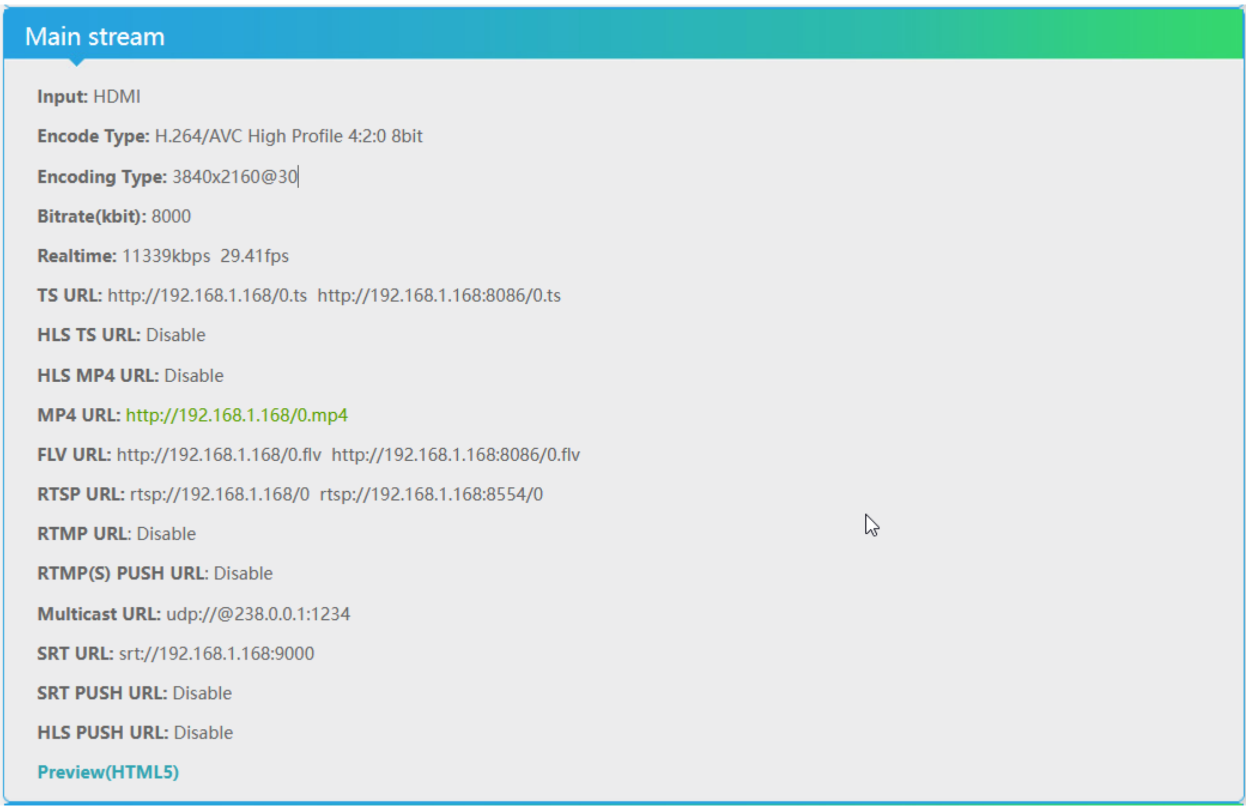


Isn’t that nice?

Get some codec info’s from VLC:

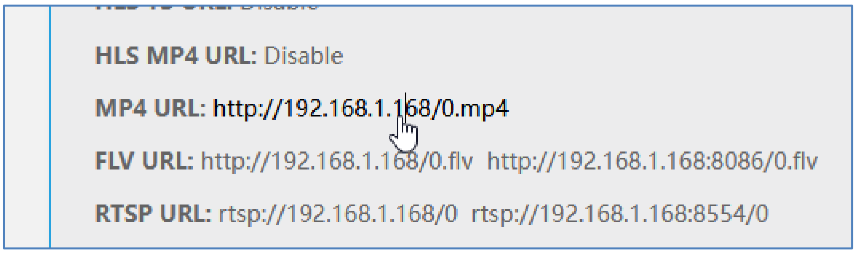


Again: The status-page …

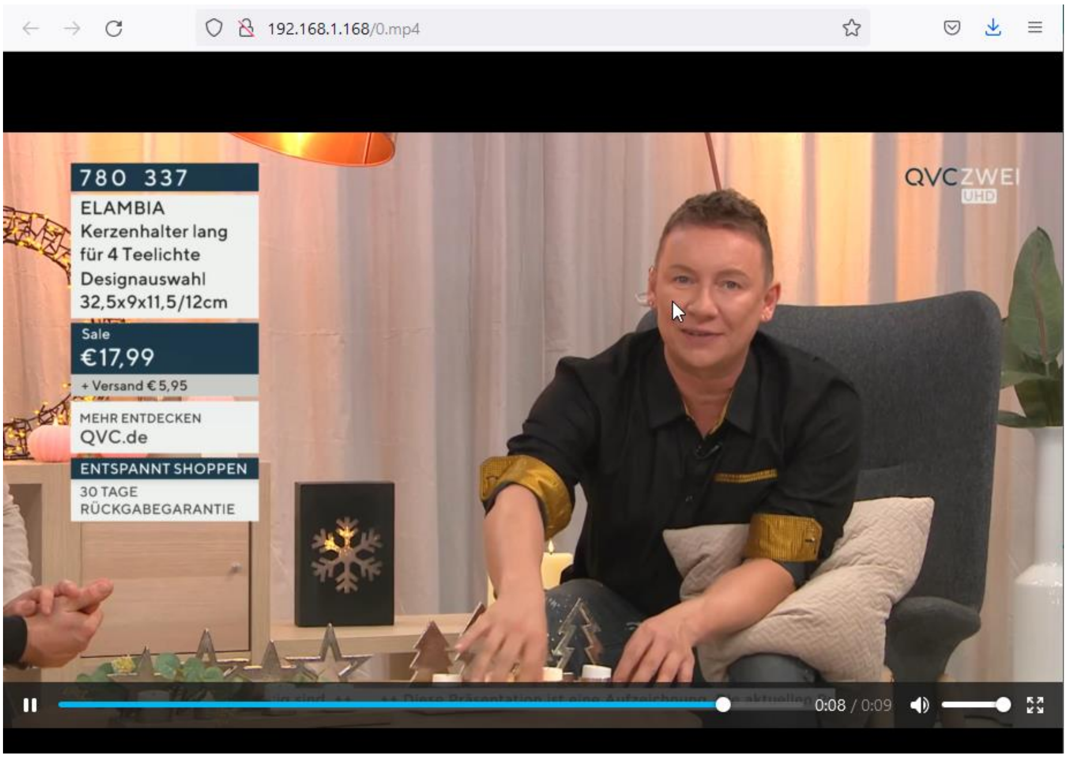


Gives you all stream information you have configured for encoding and Streaming. You can simply use your mouse to mark copy and paste:

Clicking the mp4-link:

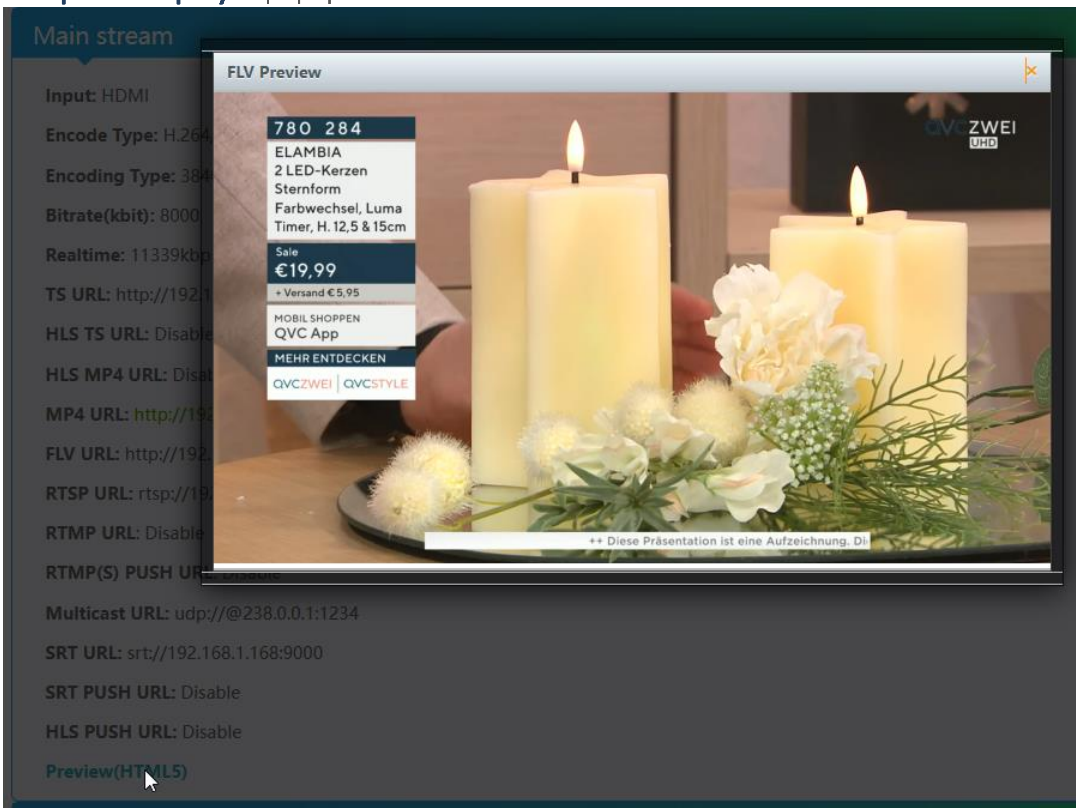


works only with h.264 codec in modern browsers:



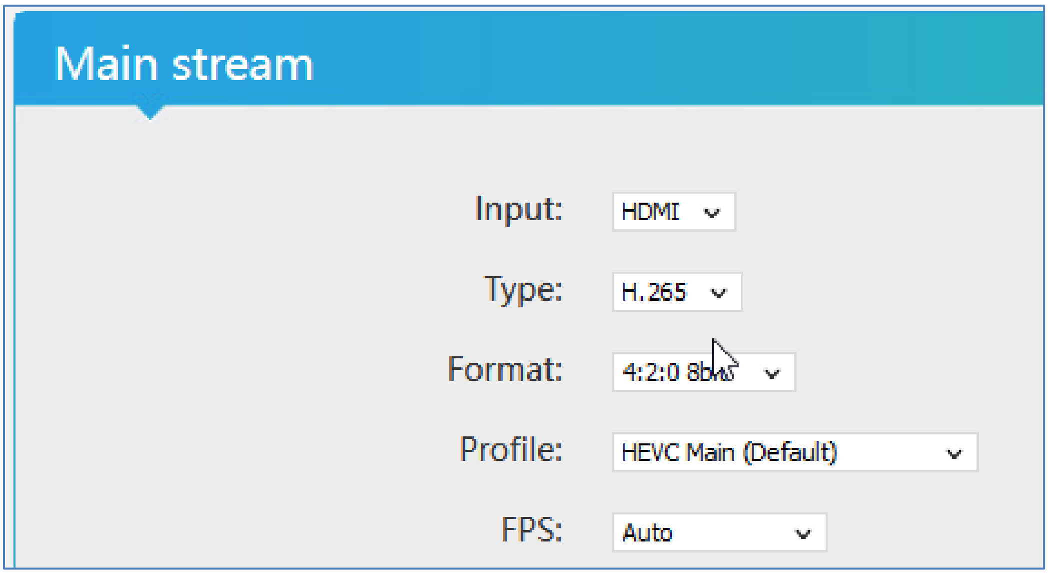
Or you can use:

The preview player popup:

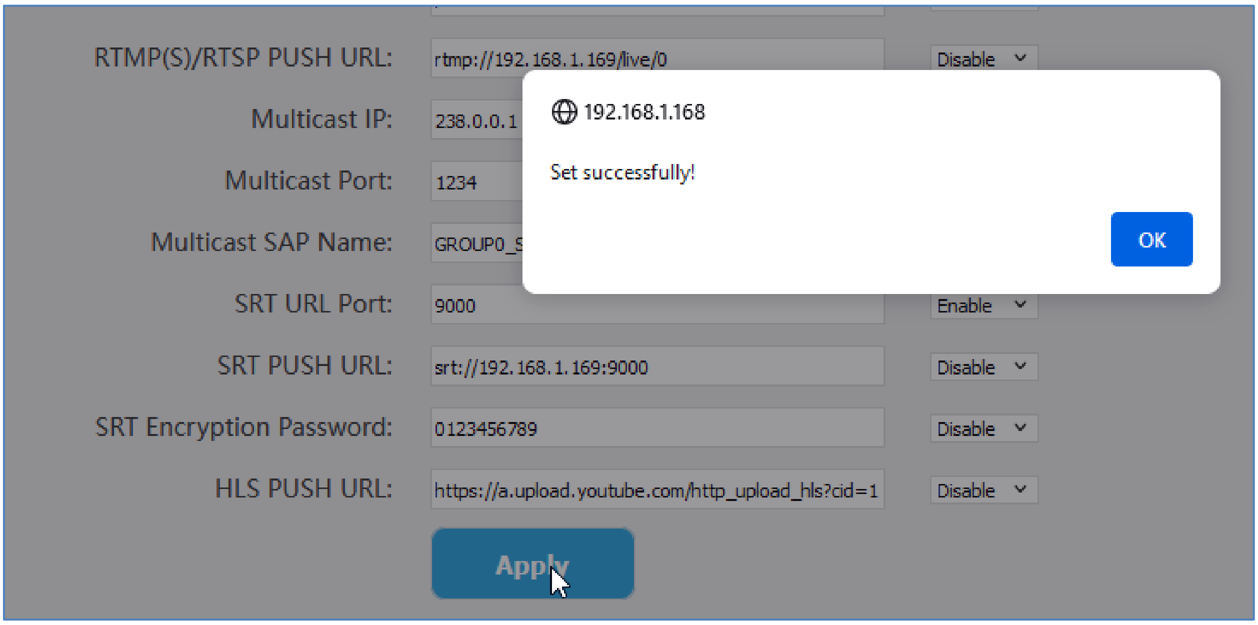


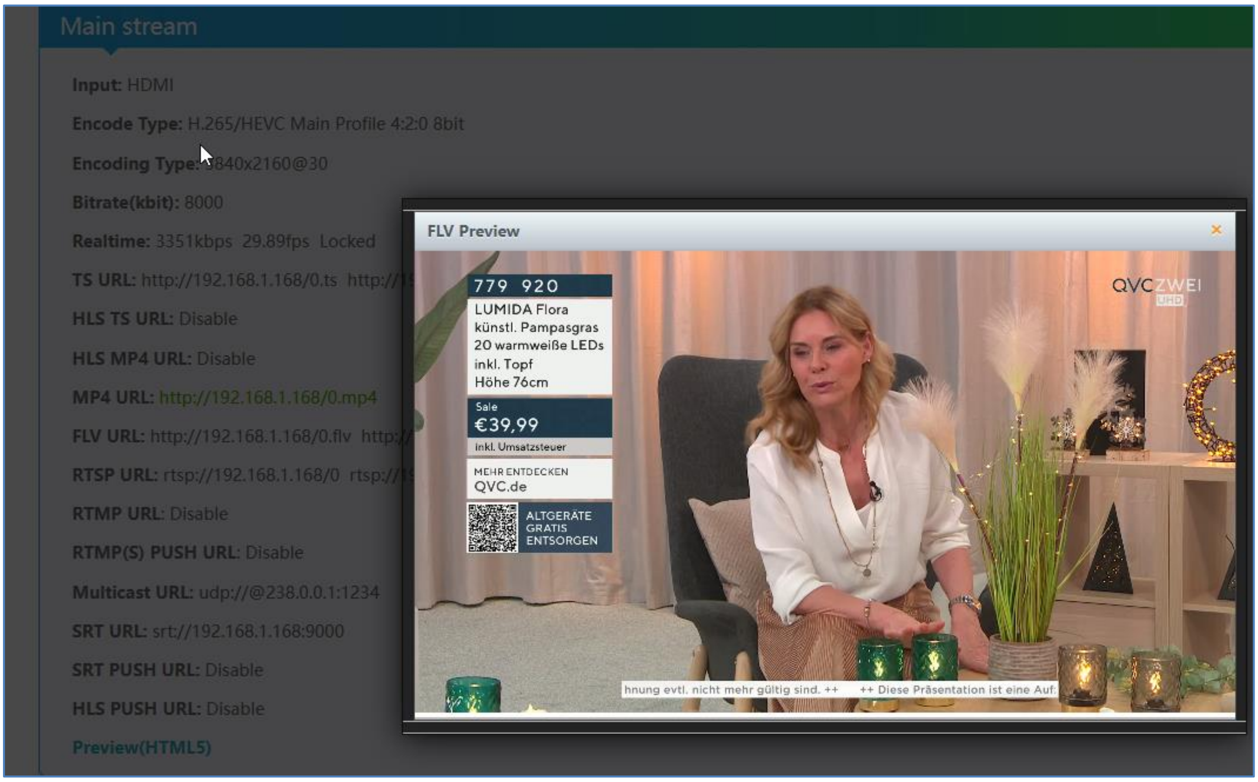
HEVC Preview works also with FLV-Stream enabled but you do not have the play stop pause functions:

In the Main-Settings:



➔ Not to forget APPLY!



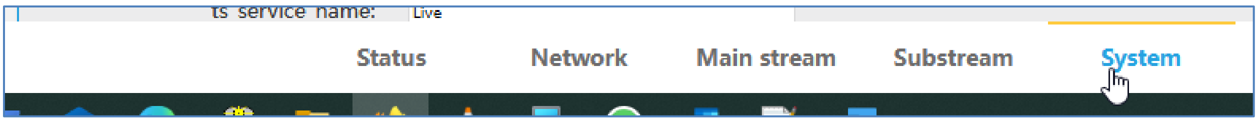


The same works in the Secondary encoder part of course.

Now we want to explain the **COMMON ENCODER Values** which are located in the system settings menus.
The

SYSTEM-ADVANCED

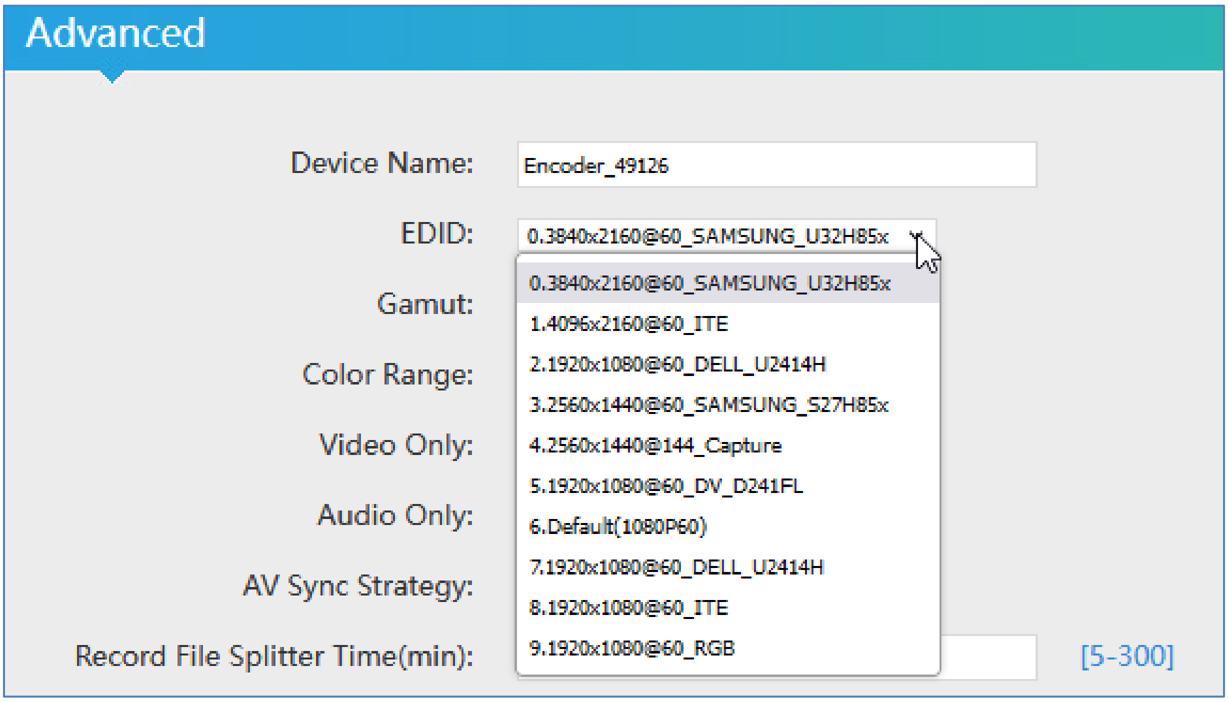
Settings:



On Top:

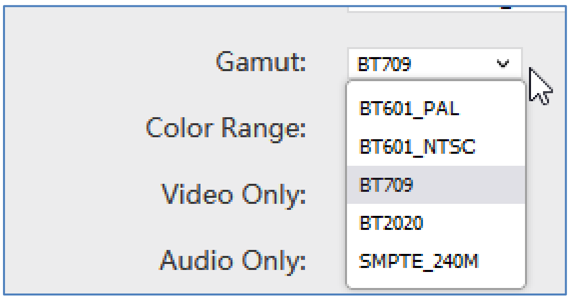
That is self-explaining, isn't it?

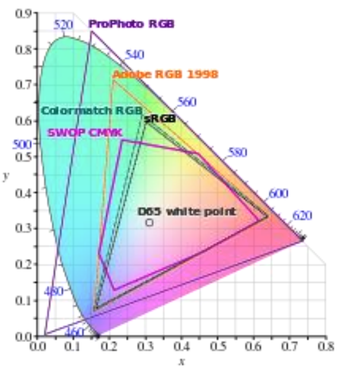
You can adjust many parameters for your encoding processes here:



First of all, the **EDID** the unit gives out depending on what and how you want to process your inputs.

The **GAMUT**:

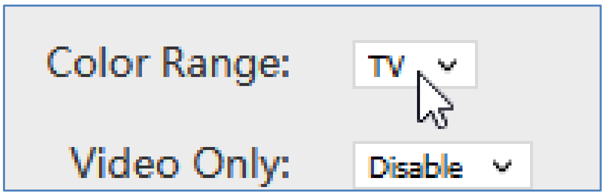




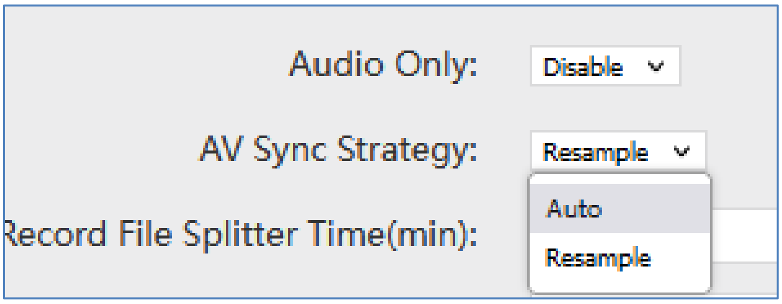
See https://en.wikipedia.org/wiki/Gamut

Video or Audio only – need no explanation…

But Colour range TV or PC might be of interest:

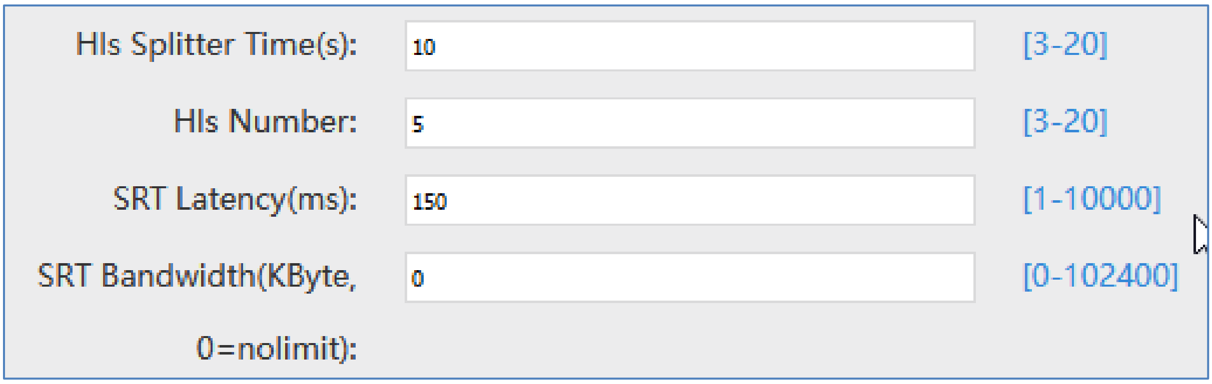


and



if you recognise lip-sync problems try this…

HLS (greetings from Apple …) is one of the streaming protocols with high latency. Check our website chapter: **Tutorials**, there are some documents regarding latencies…

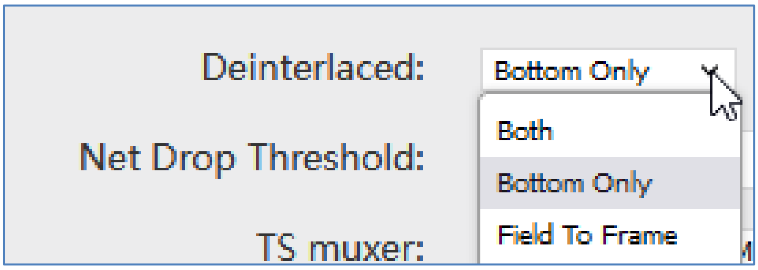


You can try and error with these HLS values as well as

SRT

is the most used streaming protocol meanwhile, reducing network-based latencies…

We released an own SRT document – also download from our web-Tutorials…



If you face jerky images in fast moving pictures horizontal left right like in F1 racings, please check BOTTOM only here – That reduces such artefacts in the encoder engine.

More details:

https://www.srtalliance.org

What is an SRT?

Secure Reliable Transport (SRT) is an Open-source software protocol and technology stack designed for live video streaming over the public internet.

SRT provides connection and control, reliable transmission similar to TCP, however, it does so at the application layer, using UDP protocol as an underlying transport layer. It supports packet recovery while maintaining low latency (default: 120 ms). SRT also supports encryption using AES.

Source: https://en.wikipedia.org/wiki/Secure_Reliable_Transport

Note: SRT works only in pairs: The stream receiver must support SRT reception.

Video Encoders are widely used in video transmission field, and SRT supported by our video encoder & decoder. Our Encoder & Decoder work perfectly for Haivision Play, Larix Broadcaster, etc.

srt-live-server(SLS)-for our Video Encoder

Our Video Encoder supports SLS for SRT.

Introduction

SRT-live-Server (SLS) is an open source live streaming server for low latency based on

Secure Reliable Transport (SRT). Normally, the latency of transport by SLS is less than 1 second via the internet.

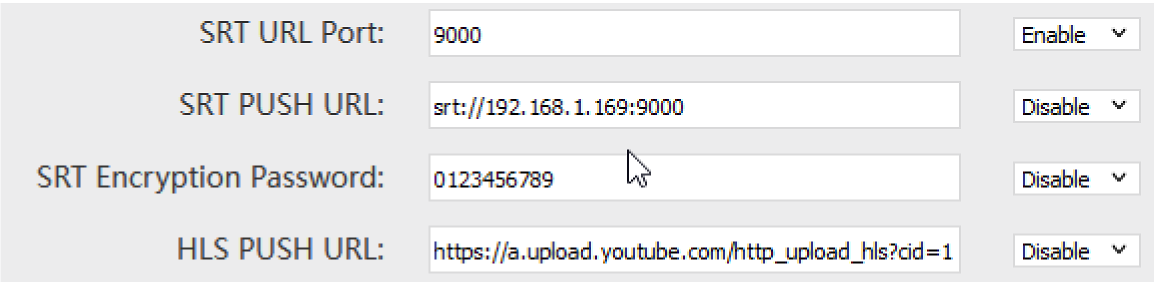
Requirements

Please install the SRT first, refer to SRT (https://github.com/Haivision/srt) for system environment basics. SLS can only run-on OS based on Linux, such as mac, centos or ubuntu etc.

Source: https://github.com/Edward-Wu/srt-live-server

Put the following URL to send to your docker container:

srt://your.server.ip:1935?streamid=input/live/yourstreamname



fill in:

For P2P, select SRT PUSH and enter the destination IP Address and Port.

Youtube supports HLS, so you can use this with your own account data to push it uploading the stream to Youtube.

SRT network-Latency can be adjusted in SYSTEM *Firmware Version depending… :*

SRT Latency(ms):	150	[1-10000]
SRT Bandwidth(KByte, 0=nolimit):	0	[0-102400]

It is a faster transport protocol for lower latency over public networks…

Don't ask me about that:

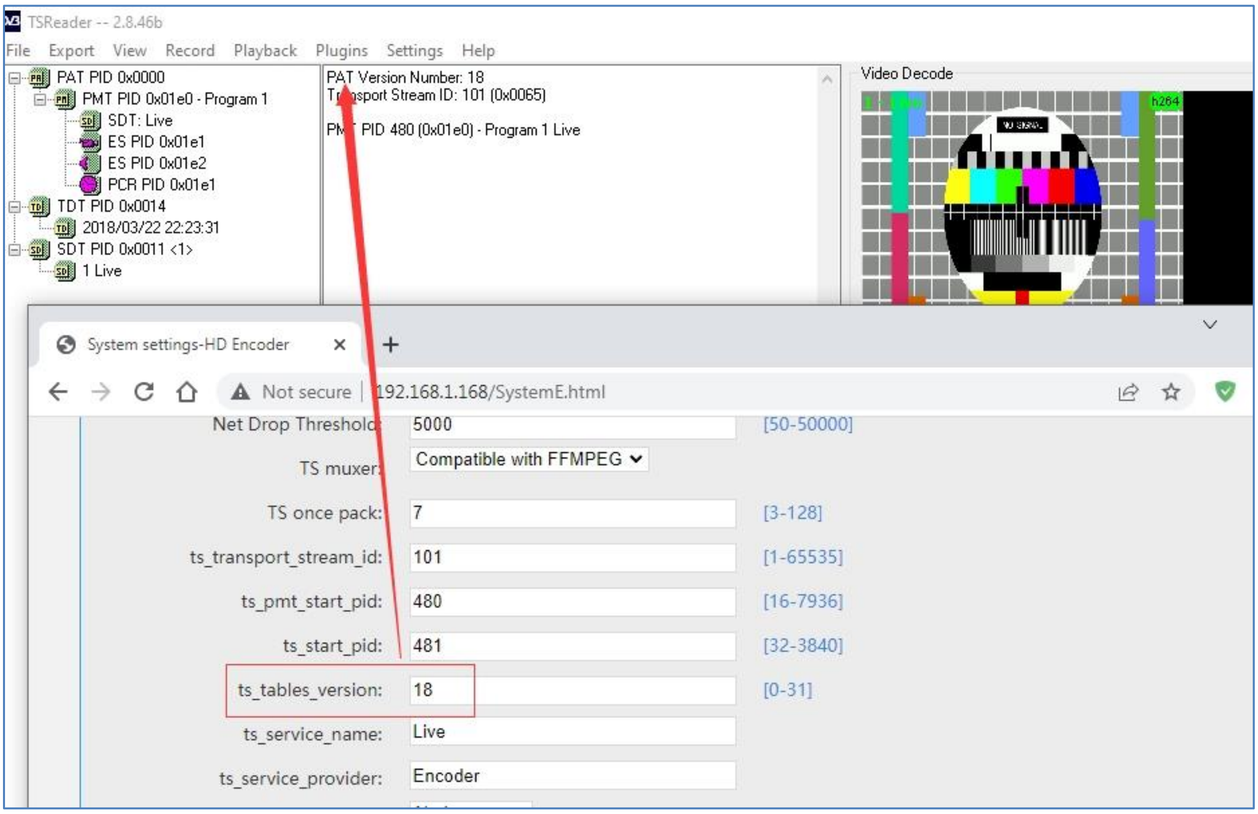
Net Drop Threshold:	5000	[50-50000]

TS muxer:	Compatible with FFMPEG	
TS once pack:	7	[3-128]
TS TDT:	Enable	
ts_transport_stream_id:	101	[1-65535]
ts_pmt_start_pid:	480	[16-7936]
ts_start_pid:	481	[32-3840]
ts_tables_version:	6	[0-31]
ts_service_id:	1	[1-65535]
ts_service_name:	Live	
ts_service_provider:	Encoder	
ts_system_b:	Disable	
TS Empty Packet:	No Insert	

This part is almost related to DVB-conformity of the streaming TS and DVB-Tables in it:

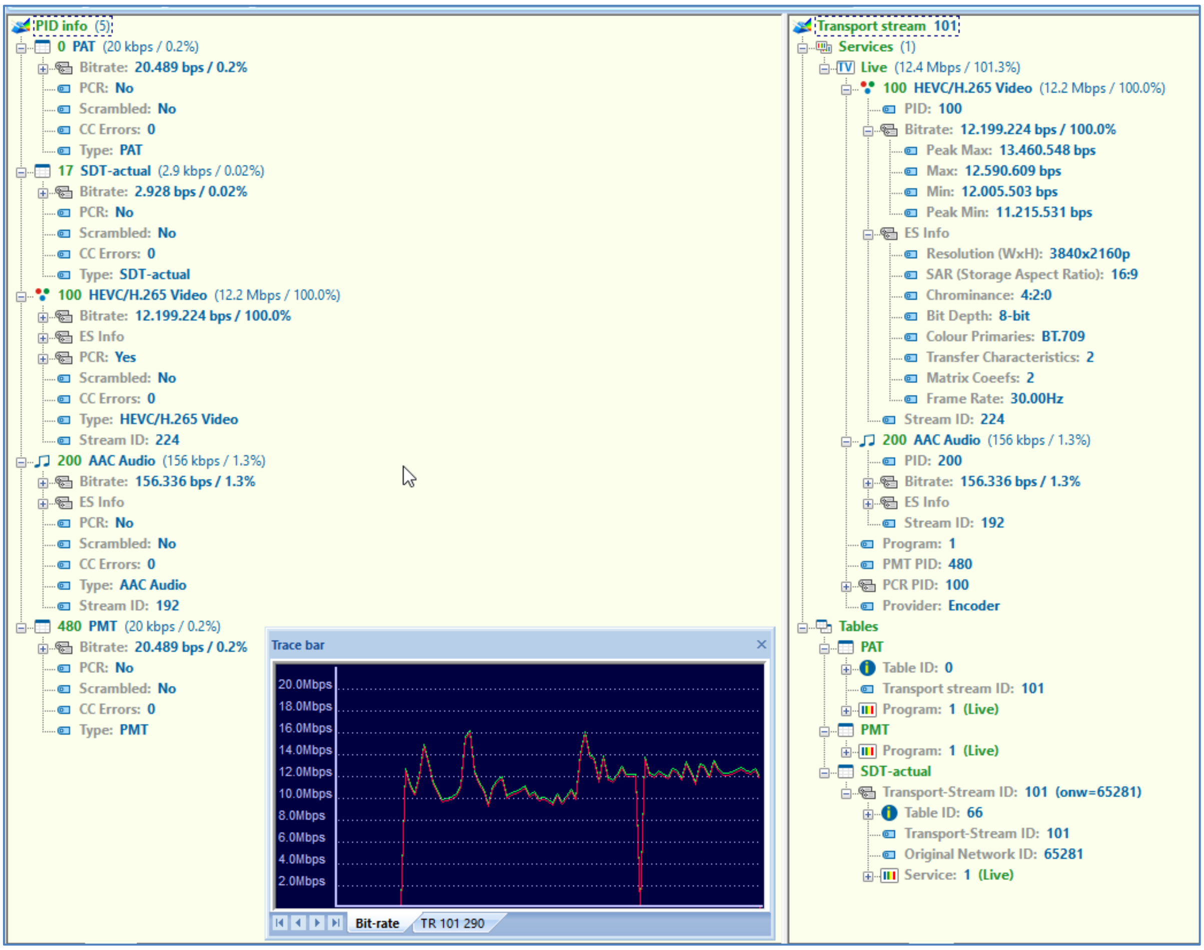
- TSMuxer can be VLC compatible (maybe there are some issues – who knows…)
- Or FFMPEG – which we should prefer.

TS Tables Version is related to the PAT (See MPEG-DVB):

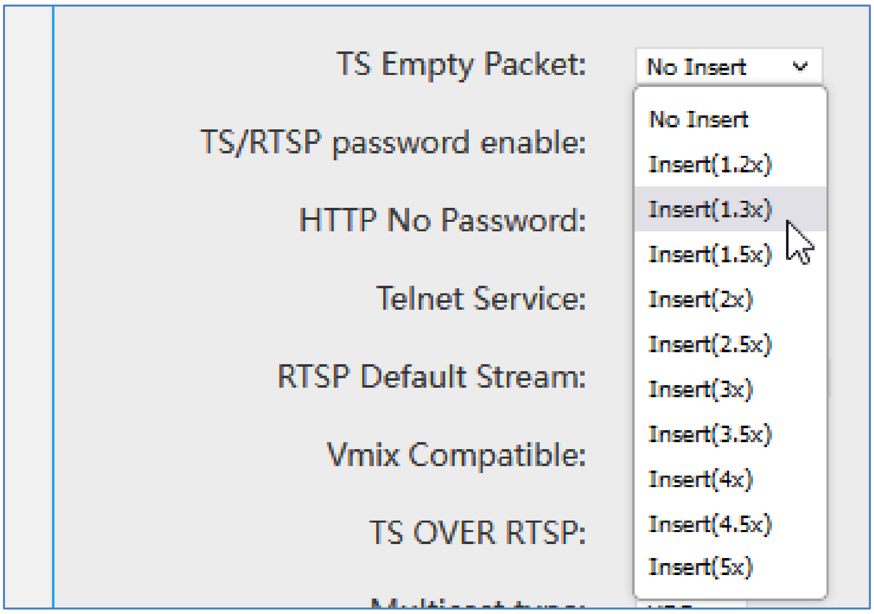


TS= TransportStream, TDT is the Time and Date table which can be obtained by enabling the NTP feature of the box… but it does not fully support TDT/TOT so it comes w/o the offset Table and no country – region settings. TS-ID is necessary in the tablework … check DVB.ORG please, we cannot explain all DVB-Tables here in a little manual. Check our Tutorial Web pages please.
PMT= Program Map Table which is essential for the TV Service you are streaming out.
PIDs are packet values assigning the Table constructs.
TS Empty packets insertion is if you need to add Zero-Packets in the PID 8191decimal (not Hex) to fill up the TS stream with zeros to produce an almost Constant bitrate STREAM (CBR) not to mismatch with CBR as an encoding procedure.
A Video-Stream is usually built as a VBR with variable bitrates because pictures and Audio are fluctuating… also, here not to mess up with the encoding bitrate processing: VBR means a different process in this:
Example: variable Bitrate Streaming VBR:

And with the related DVB-PIDs (left) and TS-table content right:

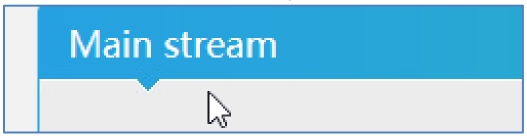


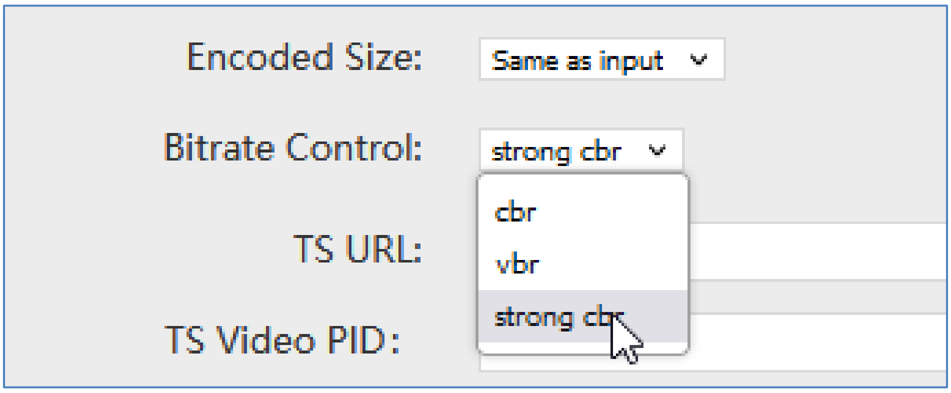
So, to produce a zero-packet added TS: (Some stupid and cheap IP to DVB-QAM Modulators might need that – checked with the STN-Holland 2x4 QAM-Modules) use factor 1.3 here



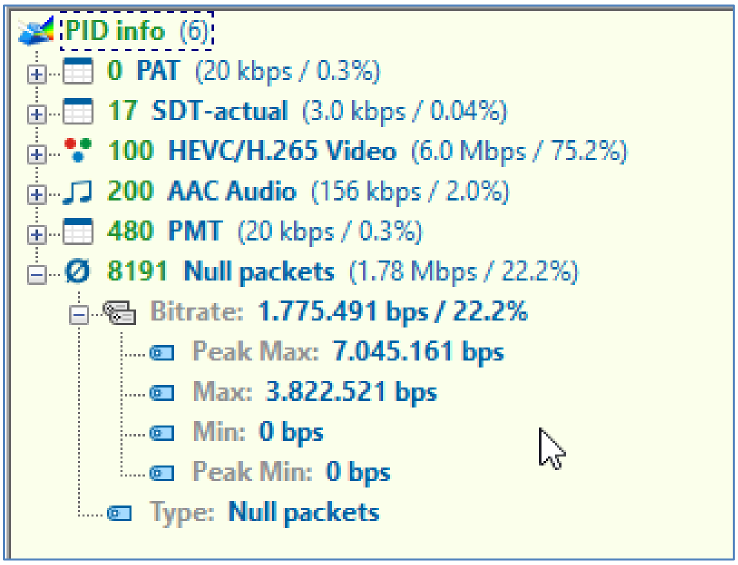
and change the mode to CBR here:

In the main or secondary stream menu:

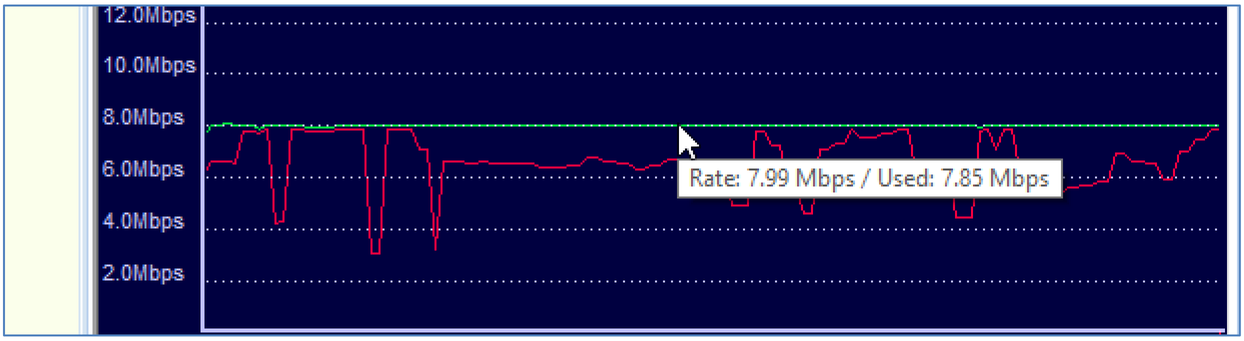




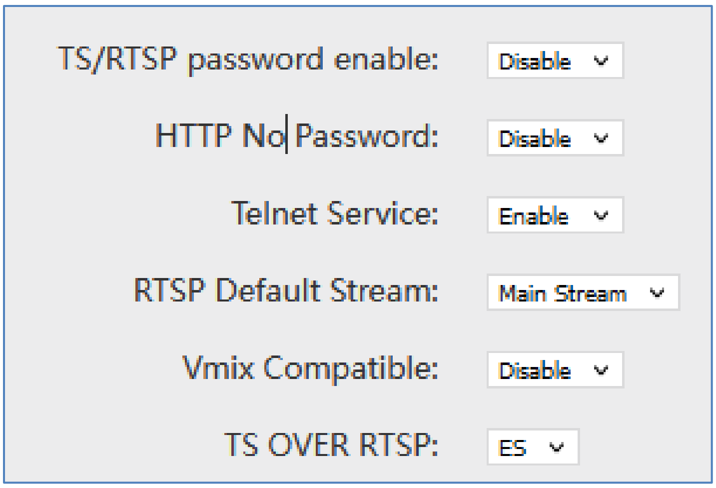
That inserts the PID8191dec:



see also:

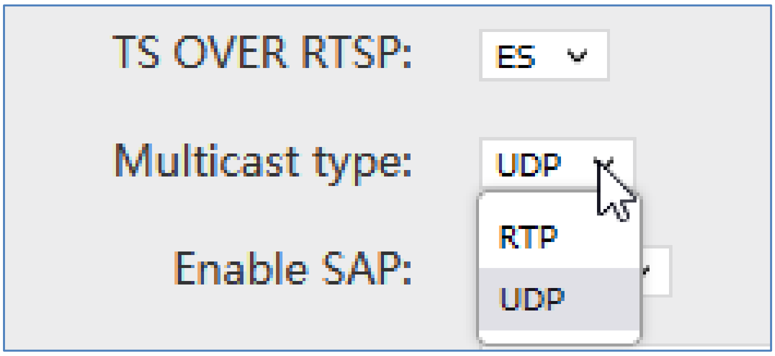


Back to SYSTEM SETTINGS:



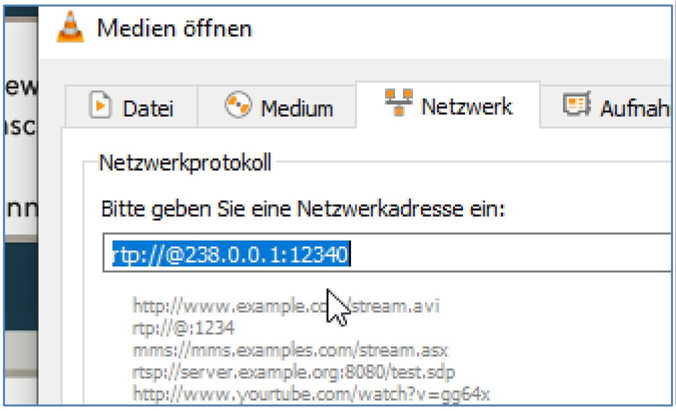
If you want to secure the different streams with a user/password you can enable that here. The user/password couple is the same like the encoder has (default: admin/admin).

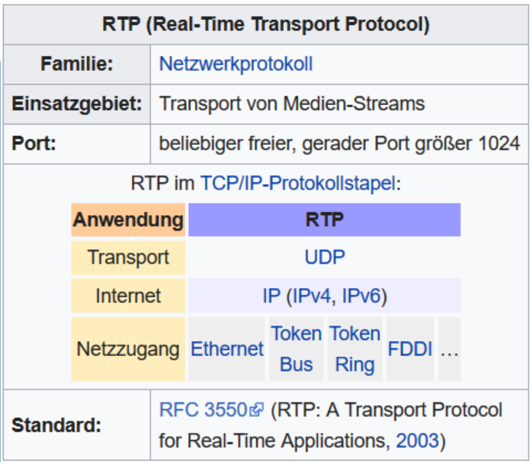
A telnet access can be also enabled and the RTSP default stream.
VMIX is a Mediaserver – Internet stream Service like youtube... to enable that service you need access to a VMIX server: https://www.vmix.com/
If you face problems try this 'compatible settings'.



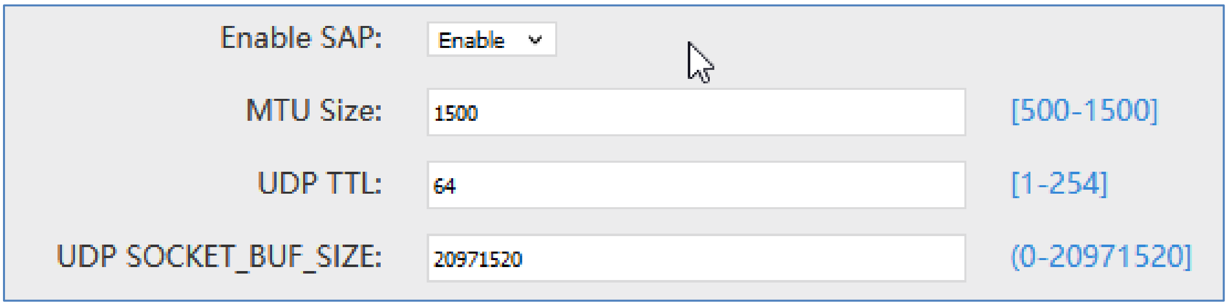
The TS over RTSP can be chosen as TS or ES (ES=Elementary Stream)
The Multicast type can be (for all multicasts in Main and secondary): UDP or RTP.
For RTP there are some rules:
The Port number should be higher than 5000 and an EVEN value eg.: 5004
Because RTP sends a 2nd stream with some timing and packet CrC information on Portnumber+1 which is odd of course.
If you do RTP, please check this for VLC it must be entered:

in VLC:





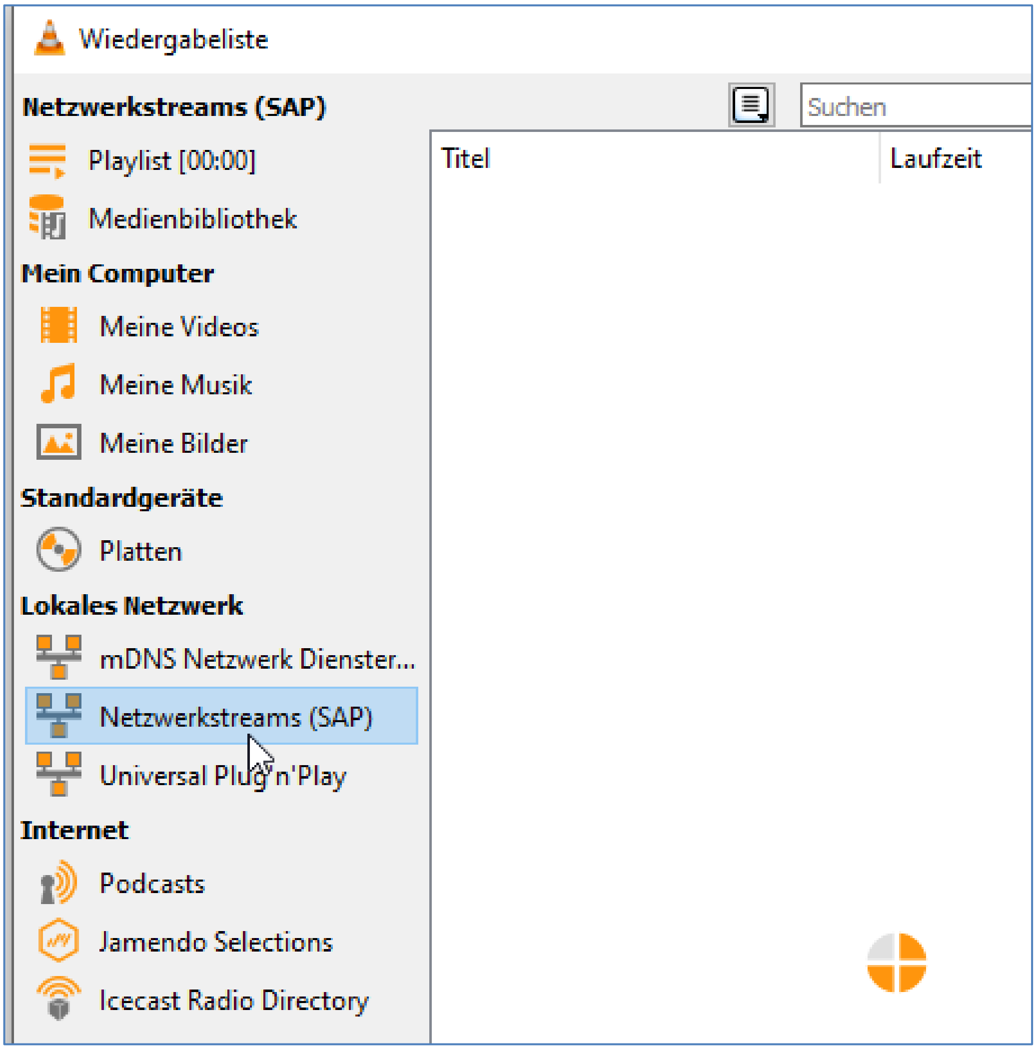
RTP (Real-Time Transport Protocol)	
Familie:	Netzwerkprotokoll
Einsatzgebiet:	Transport von Medien-Streams
Port:	beliebiger freier, gerader Port größer 1024
RTP im TCP/IP-Protokollstapel:	
Anwendung	RTP
Transport	UDP
Internet	IP (IPv4, IPv6)
Netzzugang	Ethernet, Token Bus, Token Ring, FDDI ...
Standard:	RFC 3550 (RTP: A Transport Protocol for Real-Time Applications, 2003)



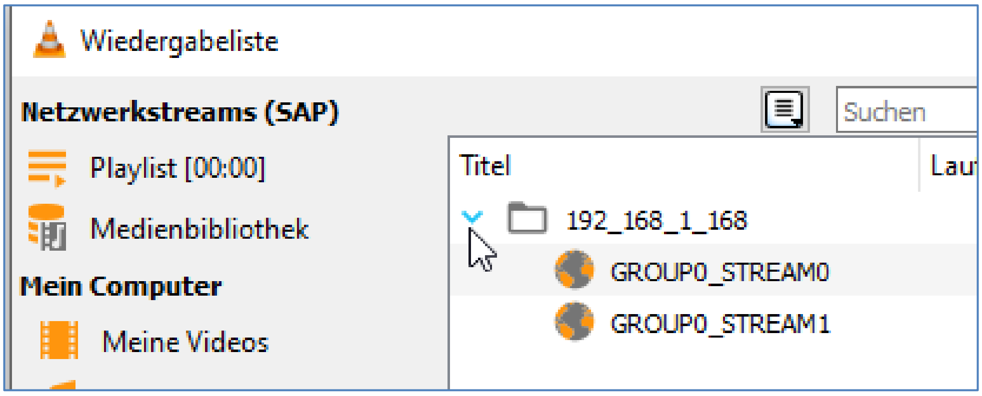
SAP is a broadcast Protocol named SAP/SDP.
The **Session Announcement Protocol (SAP)** is an experimental protocol for advertising multicast session information. SAP typically uses Session Description Protocol (SDP) as the format for Real-time Transport Protocol (RTP) session descriptions. Announcement data is sent using IP multicast and the User Datagram Protocol (UDP).
Under SAP, senders periodically transmit SDP descriptions to a well-known multicast address and port number (9875).[1] A listening application constructs a guide of all advertised multicast sessions.
SAP was published by the IETF as RFC 2974.[2]

Note about SAP:
IPv4 global scope sessions use multicast addresses in the range 224.2.128.0 - 224.2.255.255 with SAP Announcements being sent to 224.2.127.254 Port 9875 (note that 224.2.127.255 is used by the obsolete SAPv0 and MUST NOT be used).
IPv4 administrative scope sessions using administratively scoped IP multicast. The multicast address to be used for announcements is the highest multicast address in the relevant administrative scope zone. For example, if the scope range is 239.16.32.0 - 239.16.33.255, then 239.16.33.255 is used for SAP Announcements.

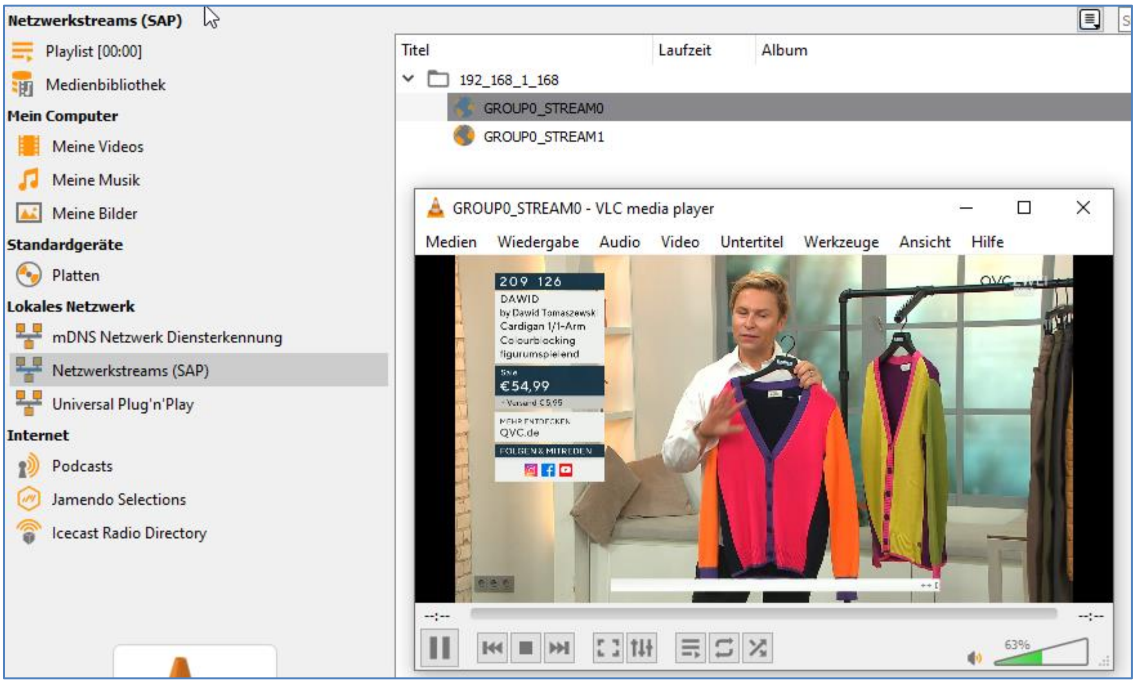
You can start a search by VLC :

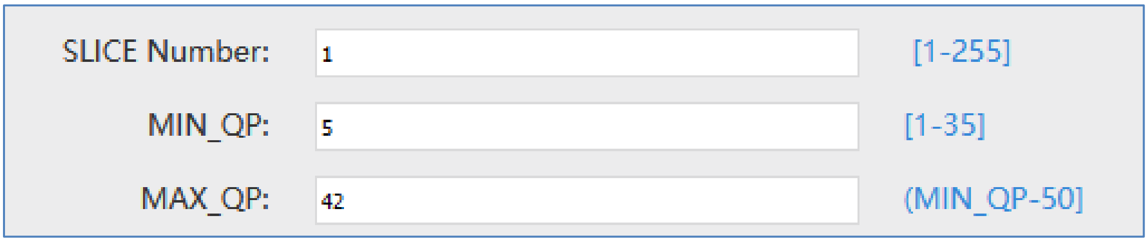


after a while it will find the multicast from the Encoder:



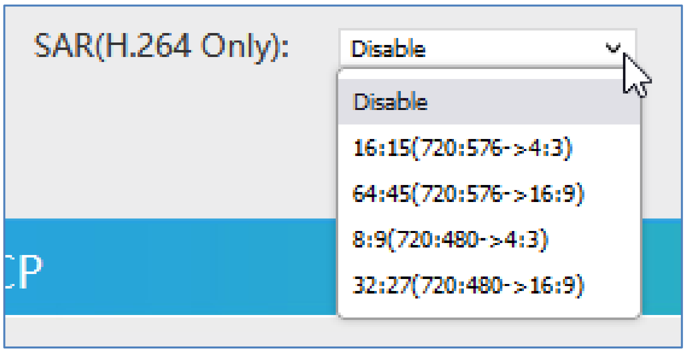
open the group of the IP and you'll see both Main and Secondary Stream entries.
By double-click you can open it:



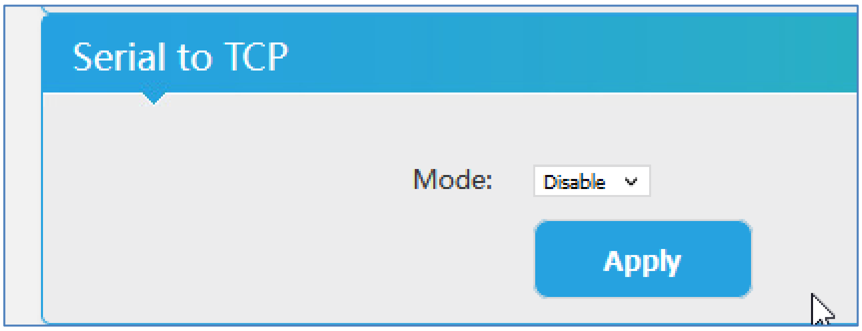


Slices can be somehow selected…. It has something to do with the Codecs:
One of the characteristics of the H.264/AVC standard is the possibility of dividing an image into regions called slices, each of which contains a sequence of macroblocks and can be decoded independently of other slices. These macroblocks are processed in a scan order, normally left to right, beginning at the top. A frame can be composed of a single slice, or multiple slices for parallel processing and error-resilience, because errors in a slice only propagate within that slice.
We recommend to keep the Slice-Number and Max-MIN_QP values as they are.
Please change them only if you know what you are doing.

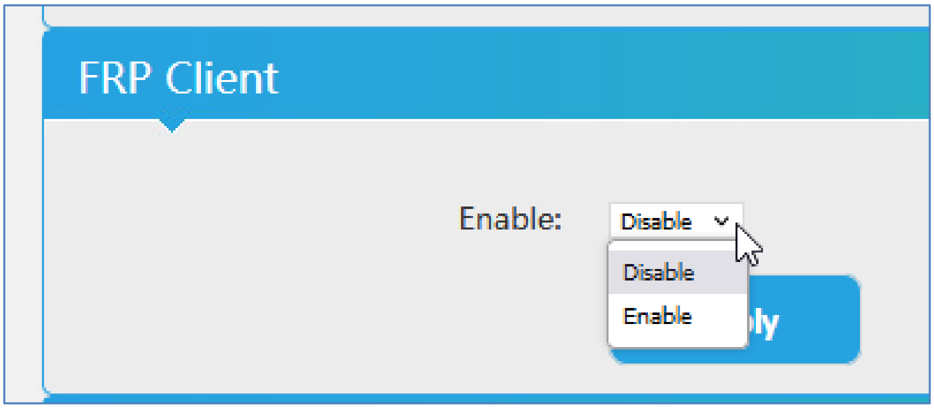
SAR-Settings:



If you do downscale or use a source which has older formats like PAL = 720x576 and that's 4:3 almost…
You can correct the EGG-Head-effect here to get correct output stream resolutions and picture format settings.



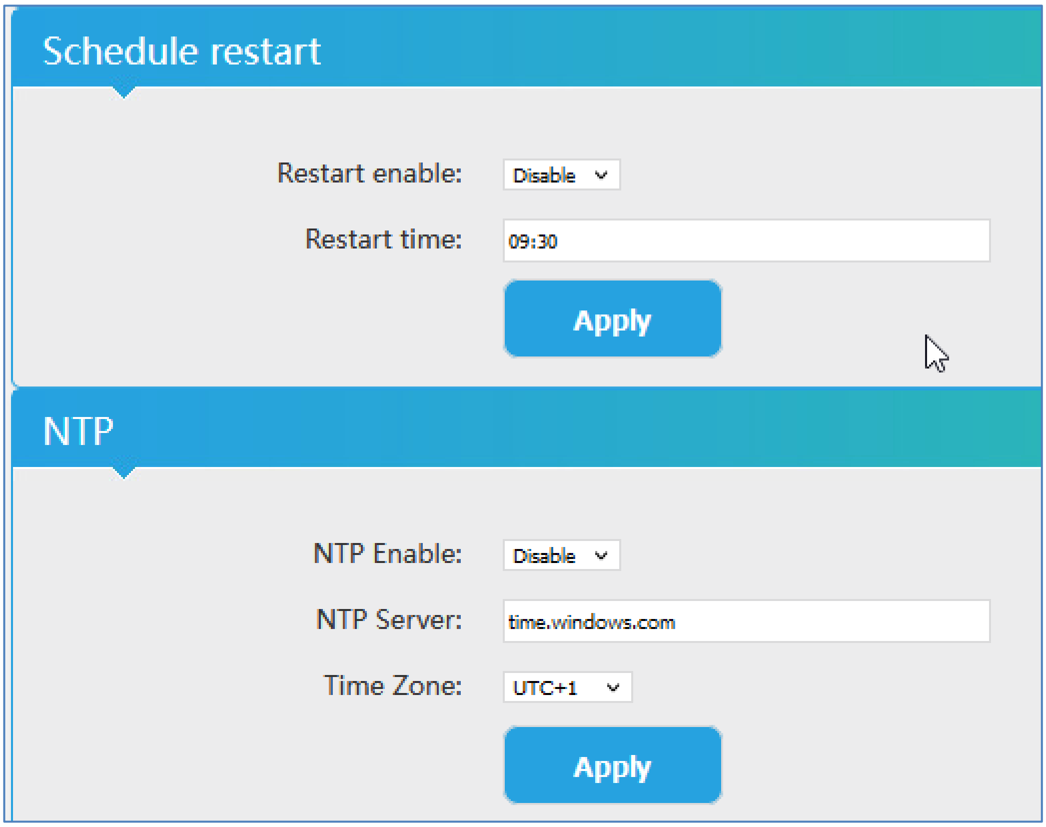
is a special protocol where serial connection (RS232 or…) are simulated over TCP. You need a tool for that in your computer
Some links:
https://sourceforge.net/projects/serialtoip/
See also a short extract about Rsync and Serial over TCP at the end of this manual.



Reverse Proxy Tools – frp

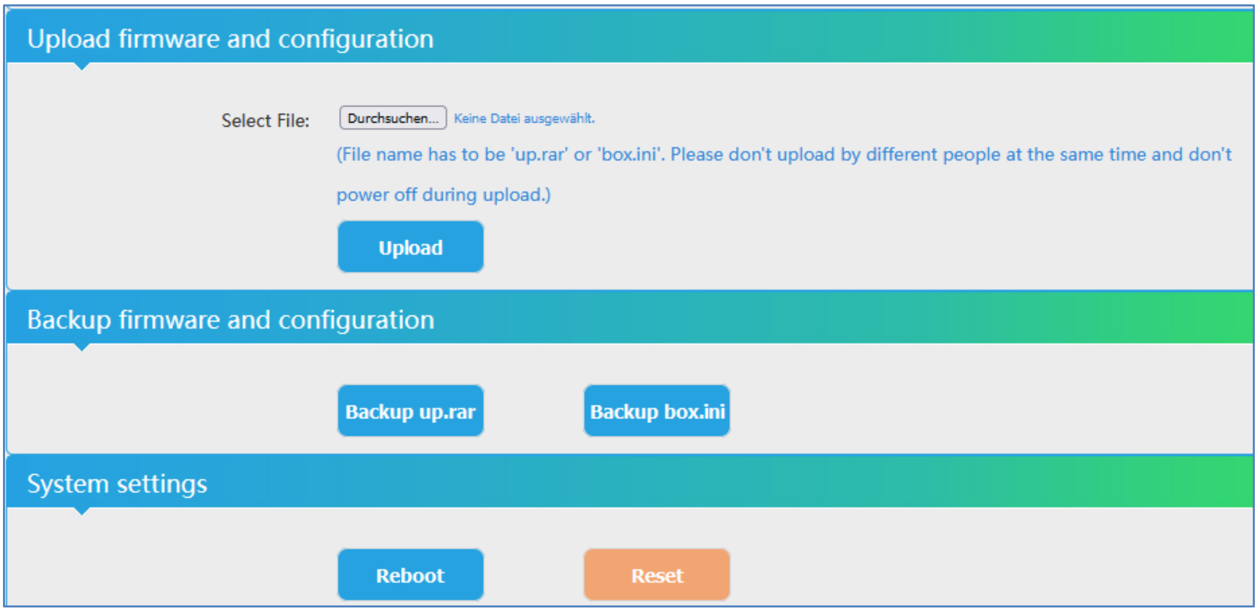
FRP is a fast reverse proxy to help you expose a local server behind a NAT or firewall to the internet. As of now, it supports tcp & udp, as well as http and https protocols, where requests can be forwarded to internal services by domain name.

If you like to set a restart for the encoder to maybe re-sync to the sources because they might be fluctuating in your TV net 😉:

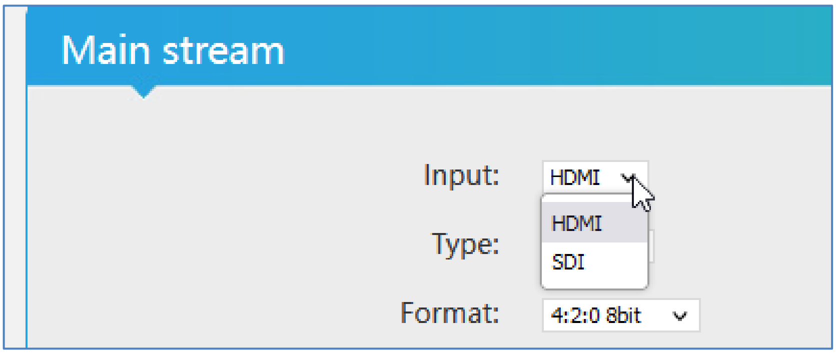


But you **should enable the NTP support**, so that the encoder know which time and date actually is.

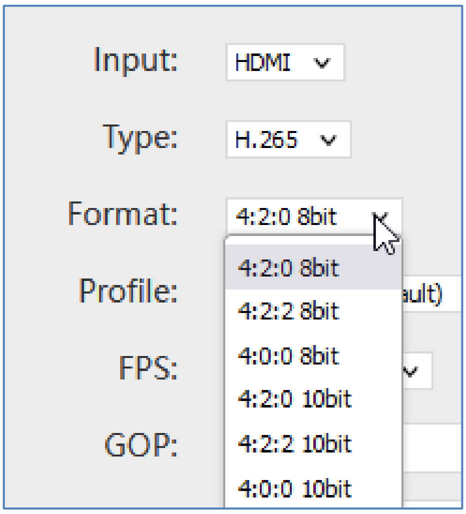
Nothing to say about these functions:



Back to the **MAIN and Secondary** Encodings:
There are many options for the encoding codecs, bitrates, pictures ...

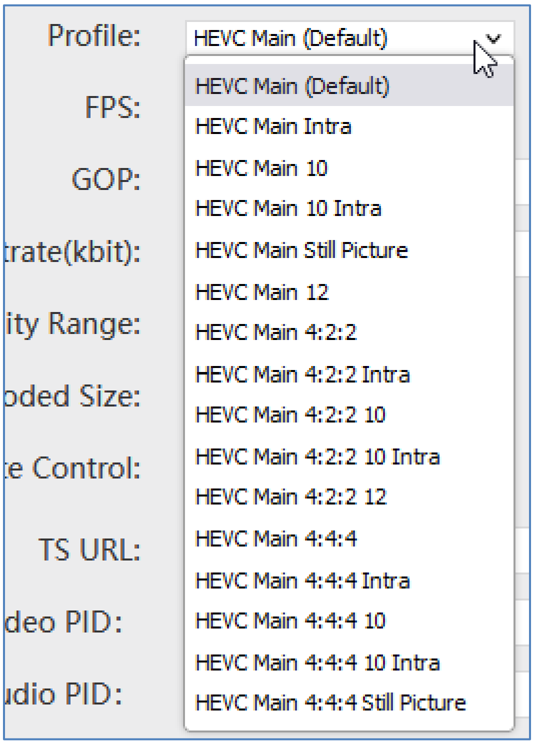


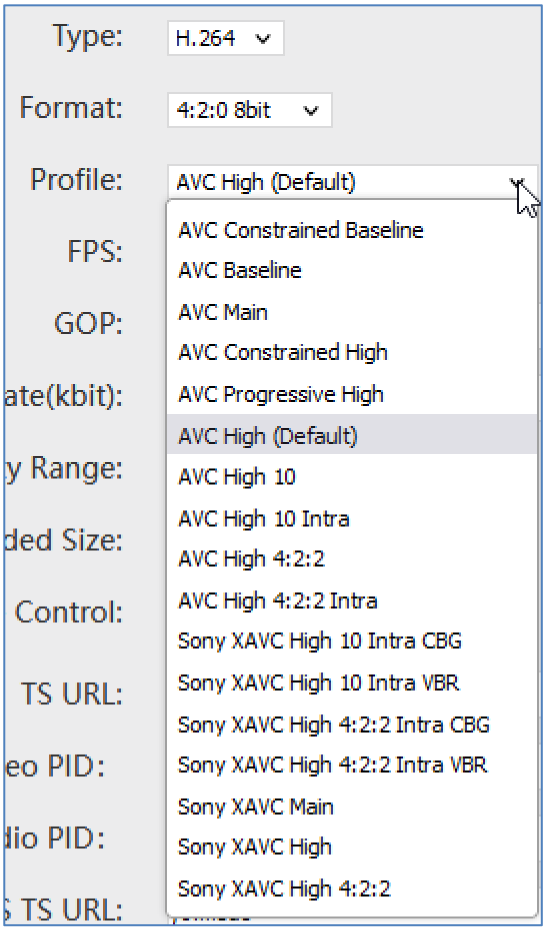
The Type is the codec to choose and even the **format and profiles**



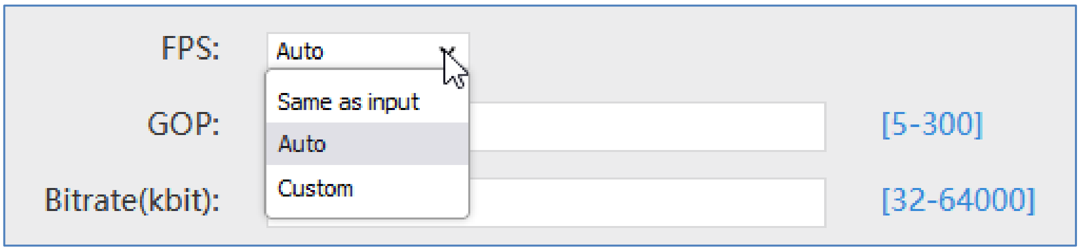
has a lot of options for your encoding process. Higher qualities need heavier computing time = increases latency.

And the profiles:

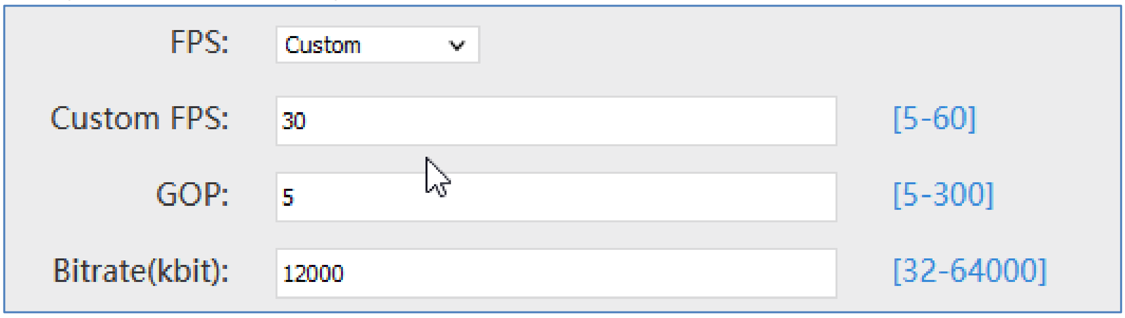




Depending on the chosen CODEC: h.264 AVC or h.265 HEVC and what actual modes the Inputs are serving. Because it does not make sense to increase your encoding profile higher than your input serves you – sounds logical ?

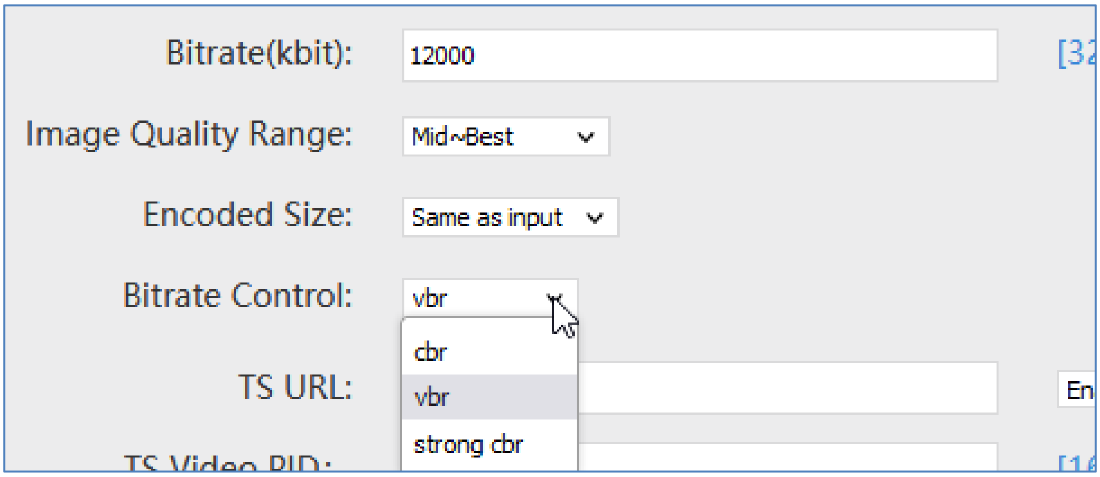


The Frames per Second should be chosen also in remembering the latency issues. If you change them the Latency will increase naturally.



We recommend to stay in the range of the usual best fitting 'FACTORS': If you re-encode an 'i50' picture than it dows not make sense to change it to i60 or p30 by the custom FPS settings. The factor should be either half or same. For the Group of Pictures GOP ist similar:
If you have a p50 input, a GOP of 30 is not very good fitting. But 25 or 5 …

The Bitrate:



Should be not too low and not too high.

It makes no sense to squeeze your bitrate as low as possible because you will lose a lot of picture quality by compression (HEVC and h.254 are working with compression algorithms …).
Too high settings don't make sense as well. You will flood your network or occupies too much bandwidth. Usual Bandwidth in DVB Streams are: 12-14 Mb/s for 720p50 (German public broadcasters) with proper quality. UHD-streams can reach 20-25 Mb/s or more (if HDR is used or 3D).

Encoded Size – of course you can downscale… Upscale: Try and error 😉…

The **bitrate control** means the encoding processes in Variable Bitrate (VBR or CBR).
CBR tries to encode a nearly constant output stream while VBR only encodes what comes in goes out and is the fastest method (Latency).

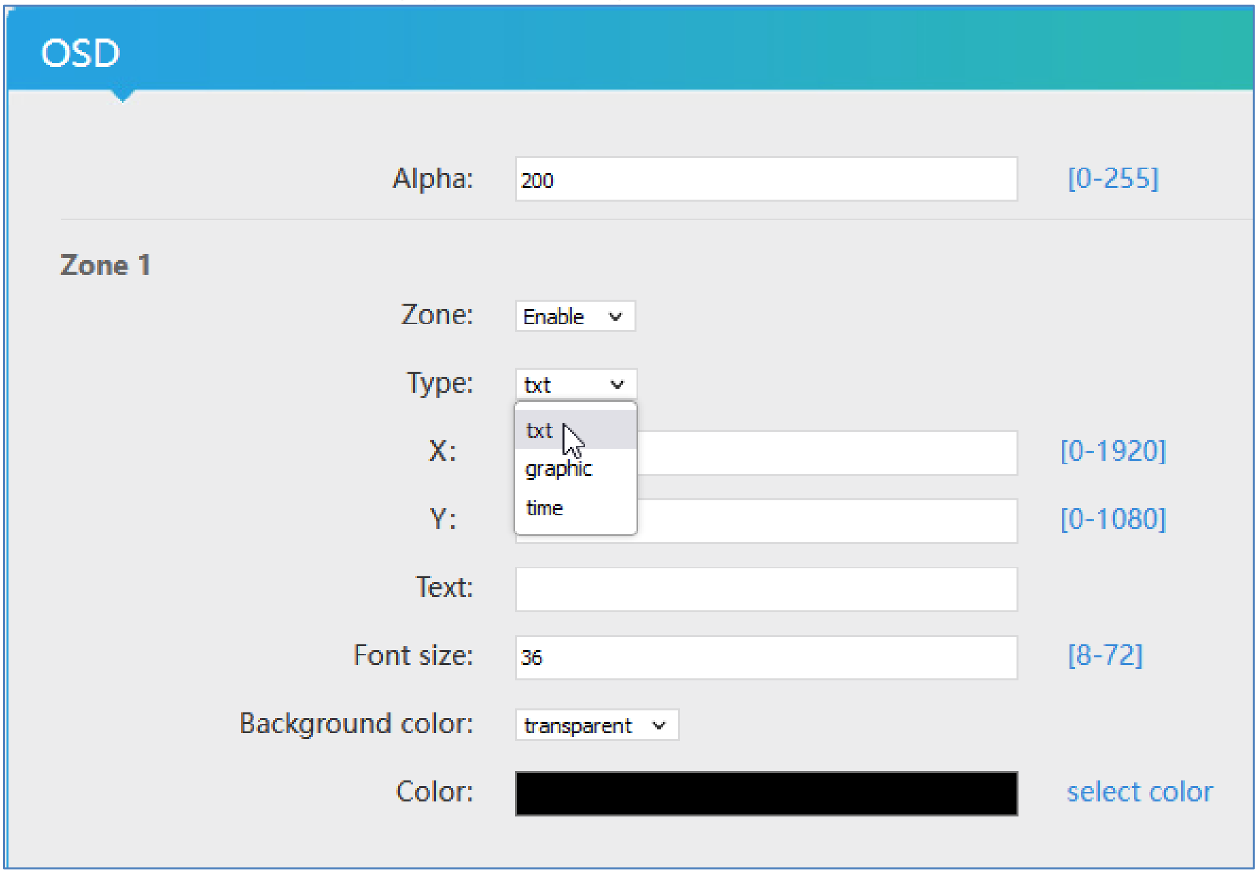
Strong CBR is recommended when you want to add Zero-Packets to the Stream (see somewhere before: already explained …)
Both have separate adjustments for their AUDIO-Codecs:

The streaming Protocols: It is a big big world…. So, we do not explain every single protocol here but you can check our stunning pages:
https://www.blankom.net/tutorials

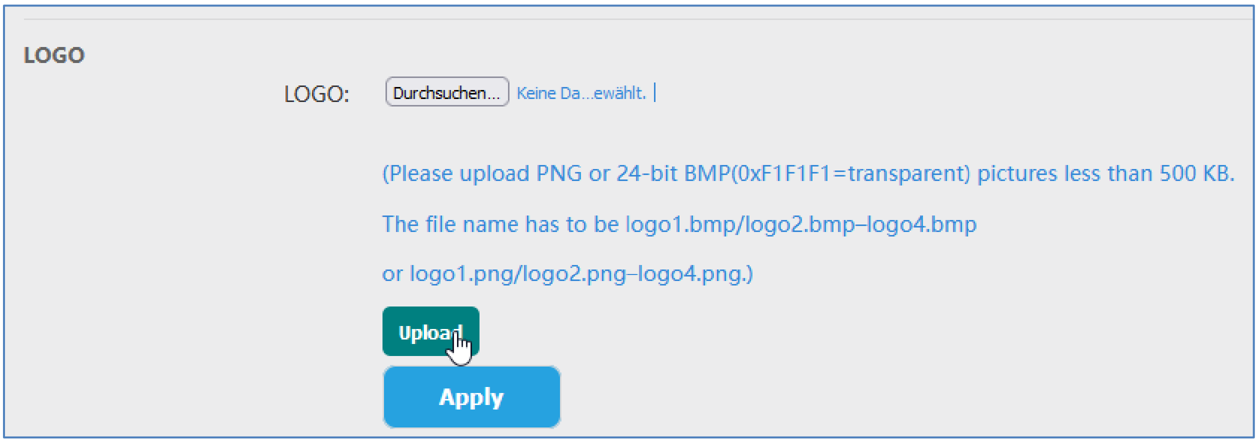
TS URL:	/0.ts	Enable
TS Video PID:	100	[16-8190]
TS Audio PID:	200	[16-8190]
HLS TS URL:	/0.m3u8	Disable
HLS MP4 URL:	/0_mp4.m3u8	Disable
MP4 URL:	/0.mp4	Enable
FLV URL:	/0.flv	Enable
RTSP URL:	/0	Enable
RTMP URL:	/0	Disable
RTMP(S)/RTSP PUSH URL:	rtmp://192.168.1.169/live/0	Disable
Multicast IP:	238.0.0.1	Enable
Multicast Port:	12340	[1-65535]
Multicast SAP Name:	GROUP0_STREAM0	
SRT URL Port:	9000	Enable
SRT PUSH URL:	srt://192.168.1.169:9000	Disable
SRT Encryption Password:	0123456789	Disable
HLS PUSH URL:	https://a.upload.youtube.com/http_upload_hls?cid=1	Disable

OSD: On Screen Display

In the past, old TV sets has a Menu which based on OSD-Overlays pressed into the running TV picture. Now we say to our text and or Logo picture inserting:

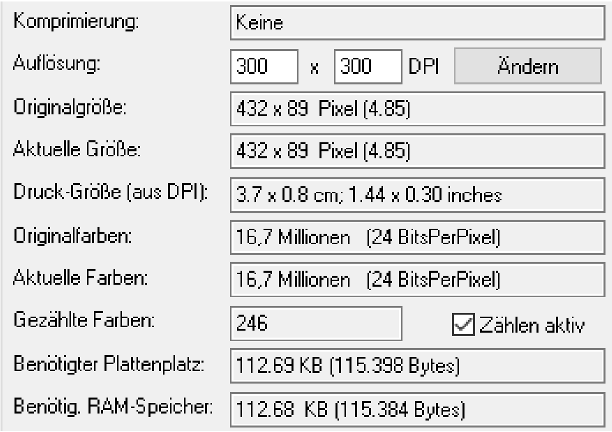


The logo can be as uploaded to the encoder system as bmp (with a particular background-colour if you like it transparent) or transparent PNG's:



Of course, there are some limitations 😉 in sizes ....and the names should be like described above.

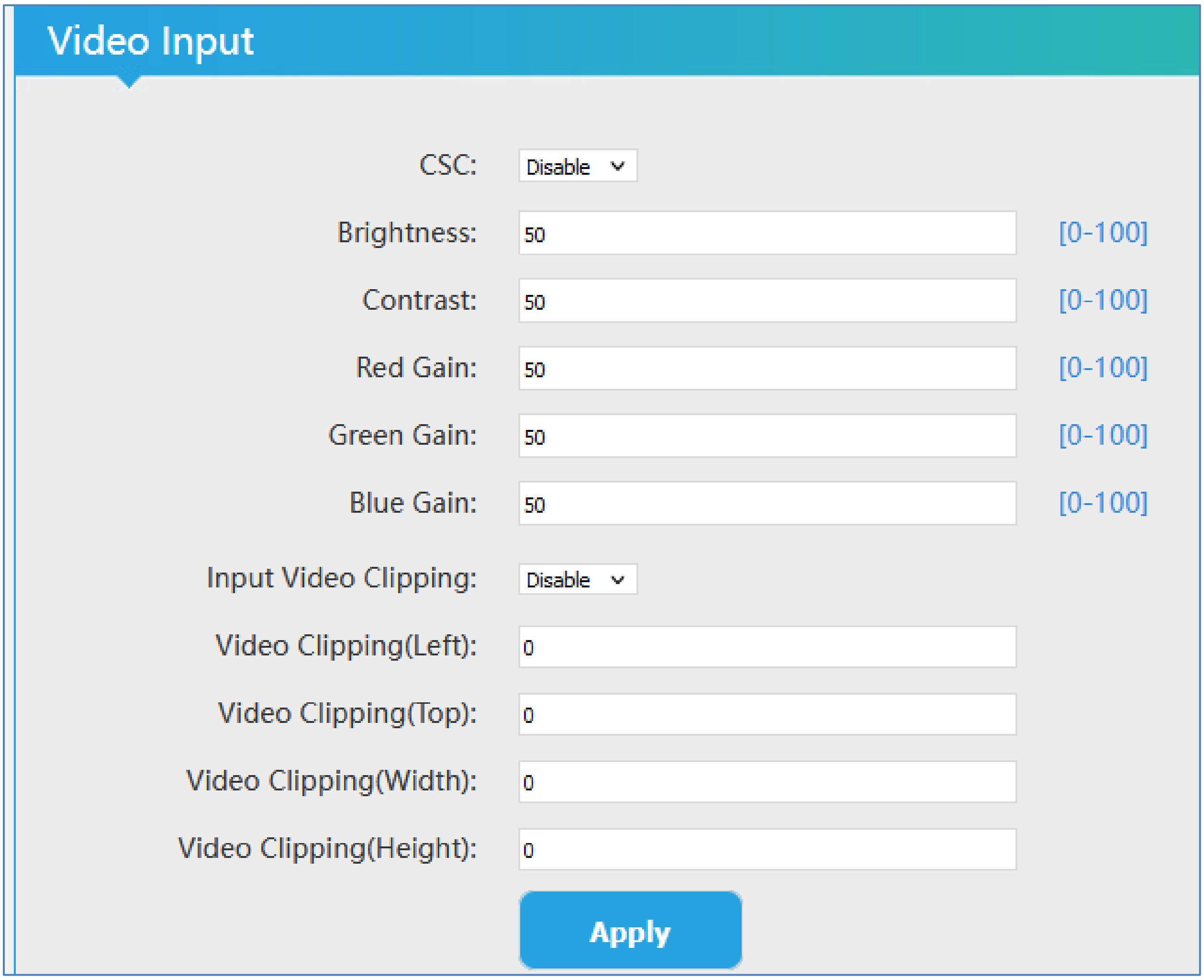
Example: The bitmap BMP:



The light grey background colour is: 0xf1f1f1 and will appear in the TV screen as Transparent.

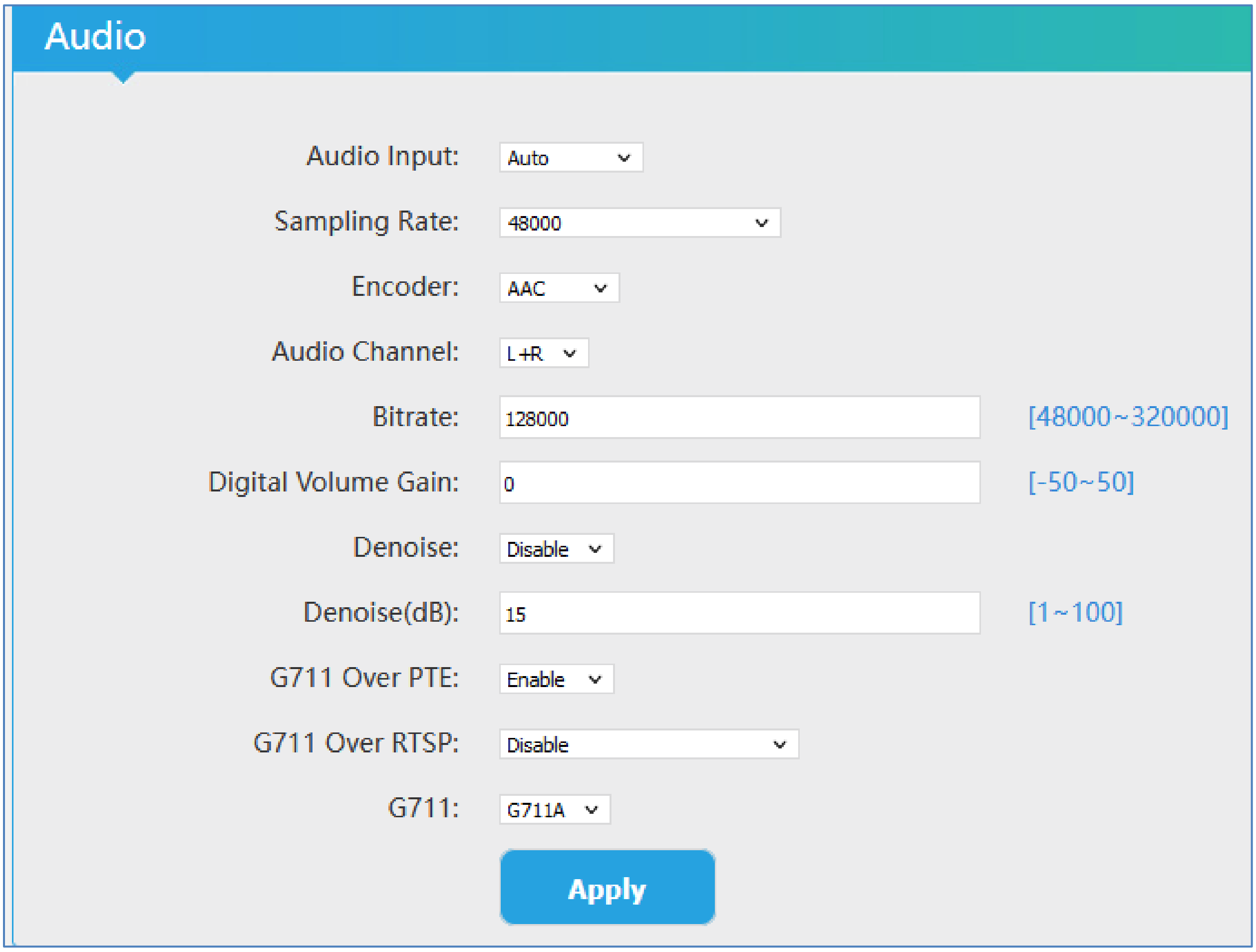
You can use GIMP or any other graphic software to change your logos background accordingly. PNG has a transparency option – BMP doesn't.

You can also use the encoder for improving the pictures with CSC settings or clip them if you need to get rid of some parts from the video source...



Be careful:

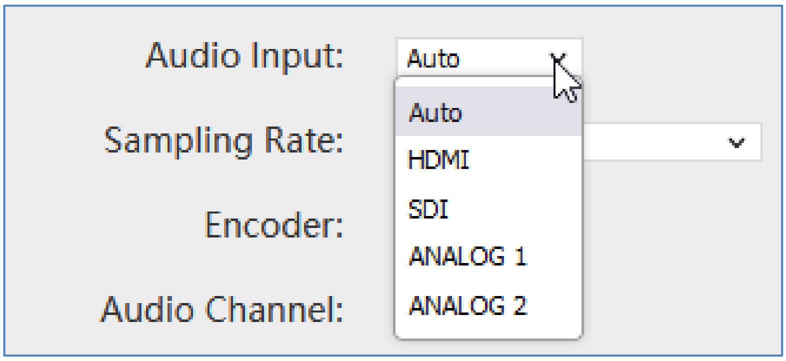
The Audio encoding parts are independend for both Main and Secondary Streams and can be chosen:



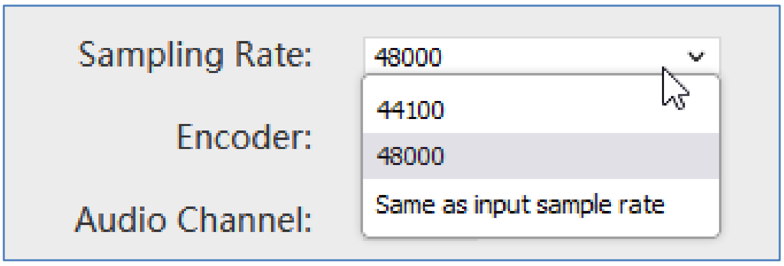
G711 is an Audio Mode which is in use by RTSP and ONVIF … maybe more 😉. You probably do not need to change it or use it anyway…

https://en.wikipedia.org/wiki/G.711

G.711 is a narrowband audio codec originally designed for use in telephony that provides toll-quality audio at 64 kbit/s. G.711 passes audio signals in the range of 300–3400 Hz and samples them at the rate of 8,000 samples per second, with the tolerance on that rate of 50 parts per million (ppm). Non-uniform (logarithmic) quantization with 8 bits is used to represent each sample, resulting in a 64 kbit/s bit rate. There are two slightly different versions: µ-law, which is used primarily in North America and Japan, and A-law, which is in use in most other countries outside North America.

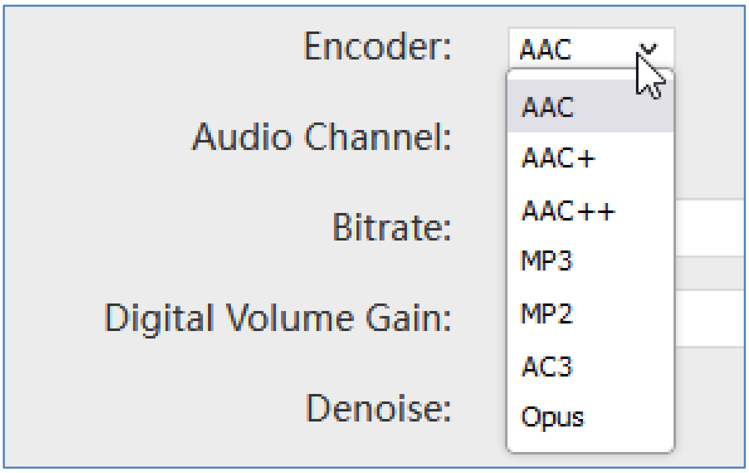


see the sources you can chose for this Main or Secondary channel…



self-explaining …

These are some supported **audio codecs** to choose from:



Example: HDMI-encoder with ONVIF

ONVIF worked with Genetec VMS like:

I would like to clarify the following:

The Encoder sent to Genetec HQ for integration is a
[redacted](HDMI)
This has been tested in our lab and works properly.

The Encoders used in the CESAC project are
[redacted] (DB15 Analog input)
This is the one we have not been able to stream video from.

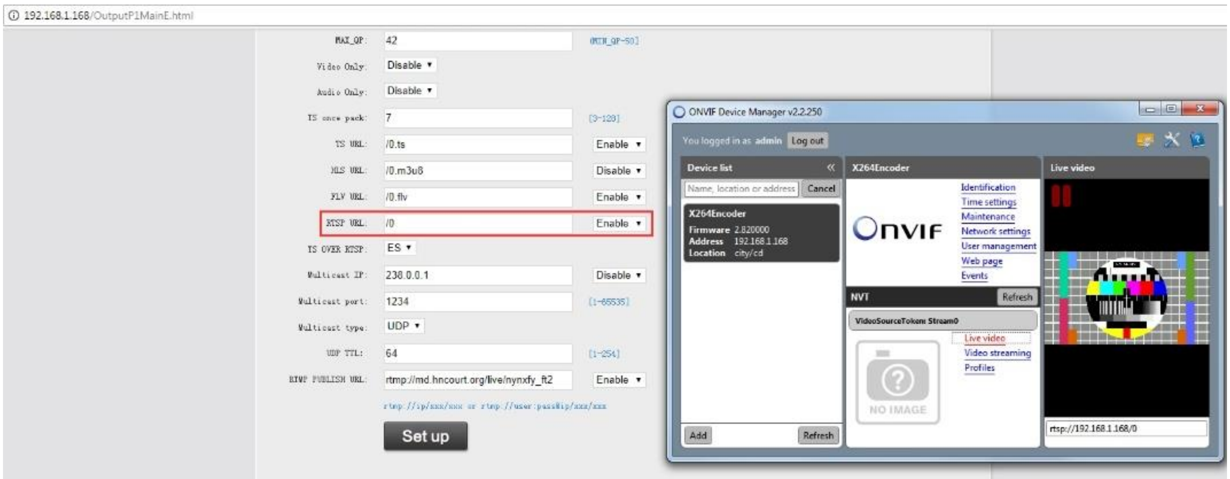
Rabindranath Parra
Professional Services Manager - Latin and South America

Genetec

Video Surveillance | Access Control | License Plate Recognition

P: +1-514-332-4000 X 6781 | M: +52 1 (443) 273-2460 | rparra@genetec.com
2280, Alfred-Nobel Blvd, suite 400, Montreal, QC, H4S 2A4, Canada

Built to evolve: www.genetec.com/dna



Note: We recommend to choose not all streams and protocols at the same time.
RTMP(s) is almost not working with h.265 HEVC-Codec because Adobe hasn't included this Codec into its list of valid codecs yet. Maybe Adobe and also Apple will be more open in the future or when you read this, they have integrated it.
For AC3 Stereo, RTSP do not support it, so, when you enable AC3, RTSP will probably use the G711A.

If you choose AC3, you can't disable the G711A audio for ONVIF:

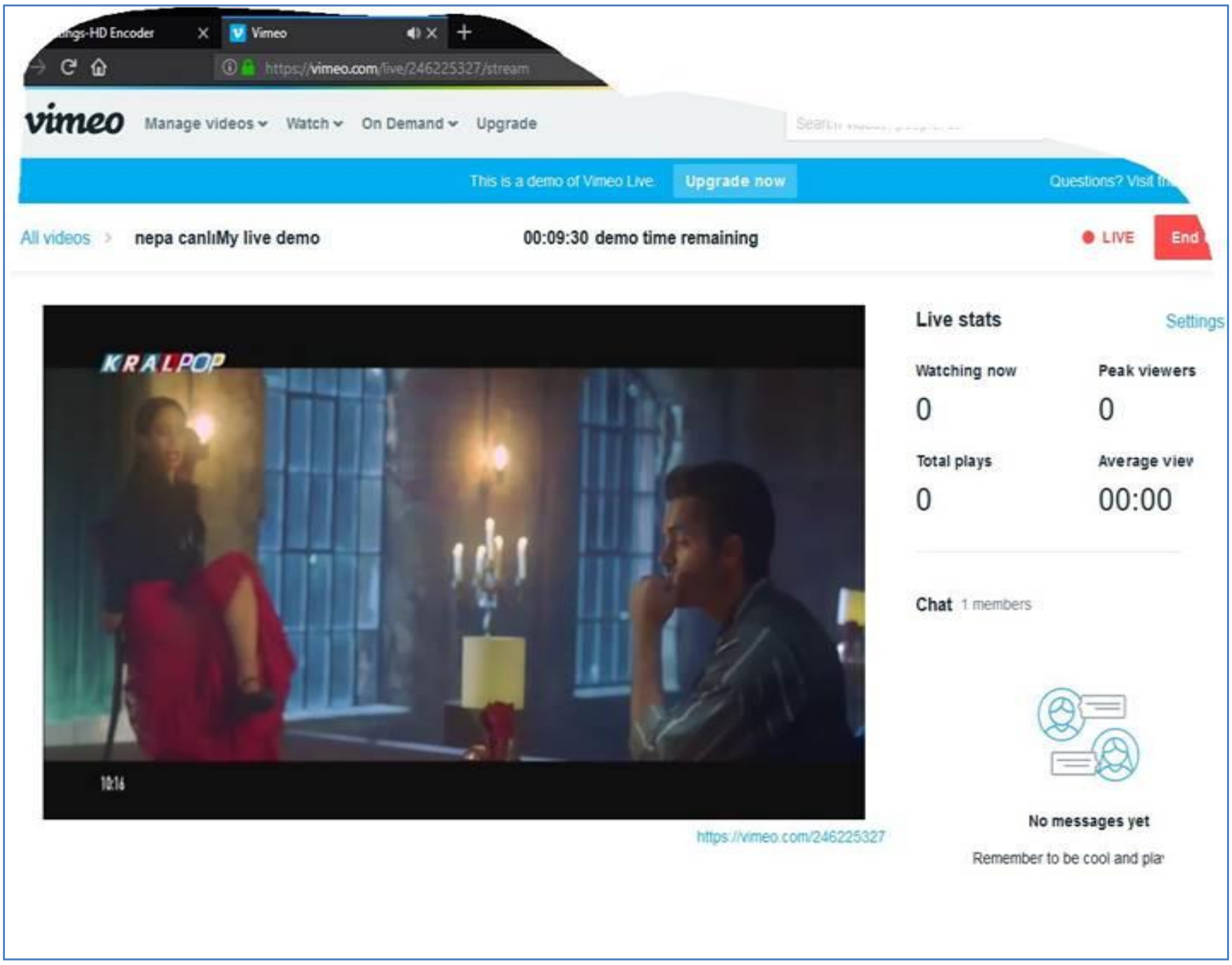
Appendix Stream to VIMEO:

Example for streaming to VIMEO Live by RTMP:

Main stream

```
Encoding Type:1920x1080@25

Bitrate(kbit):1800

TS URL: http://192.168.2.168/0.ts    http://192.168.2.168:8080/0.ts

HLS URL:Disable

FLV URL:Disable

RTSP URL:Disable

RTMP PUBLISH URL(Connected):rtmp://rtmp.cloud.vimeo.com/live?token=45dfd48b-9e8b-49bf-8539-90aa29aaf7a2
/dcfe39e7-5912-4388-8507-347daac833f8

Multicast URL:Disable
```

VIMEO gives the user an RTMP –address with a live token at the end. No username/password is necessary because they handover individual stream-keys which simply needs to be inserted as
rtmp://rtmp.cloud.vimeo.com/live?token=***********/streamkey**
Then you can control it by checking the ***vimeo*** live portal of your stream:

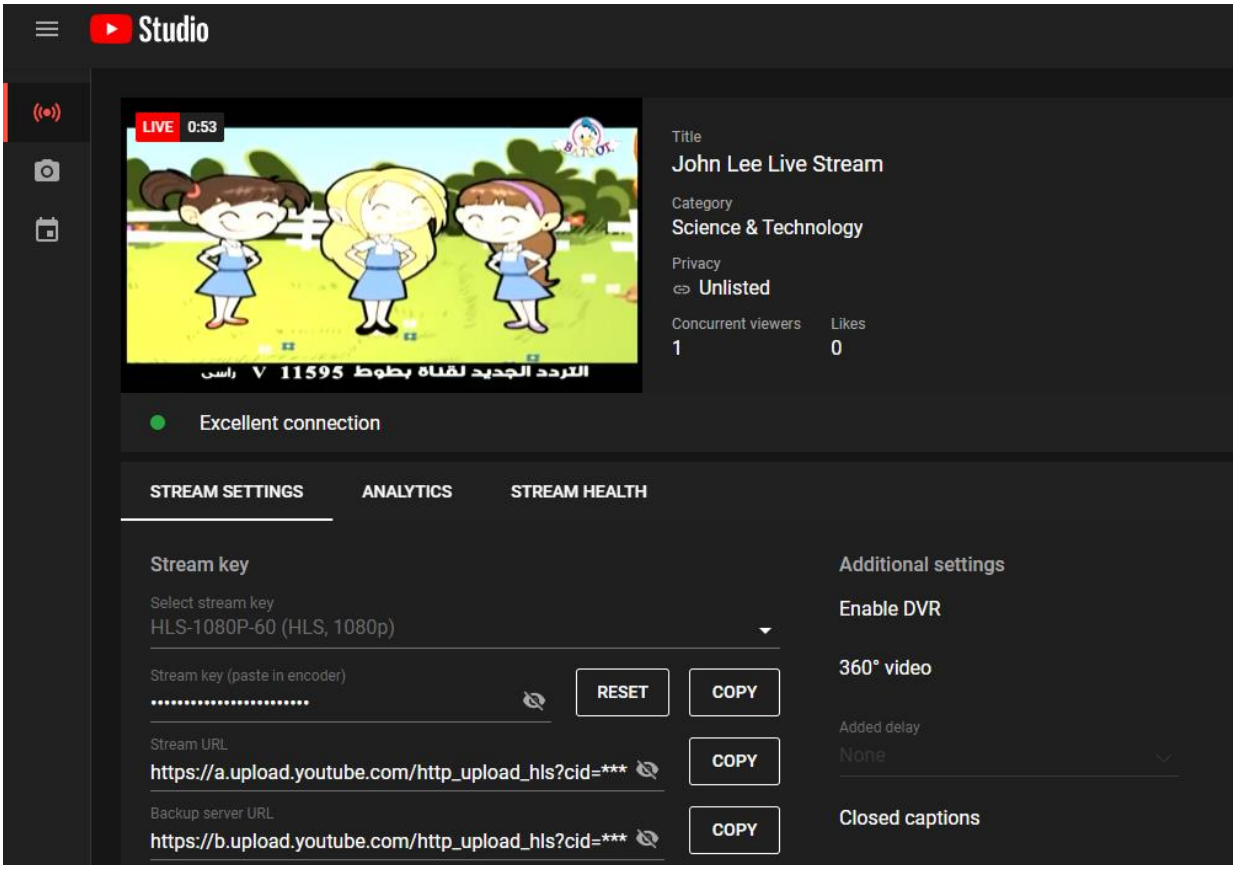


How to stream h265 encoded video to YouTube using HLS?

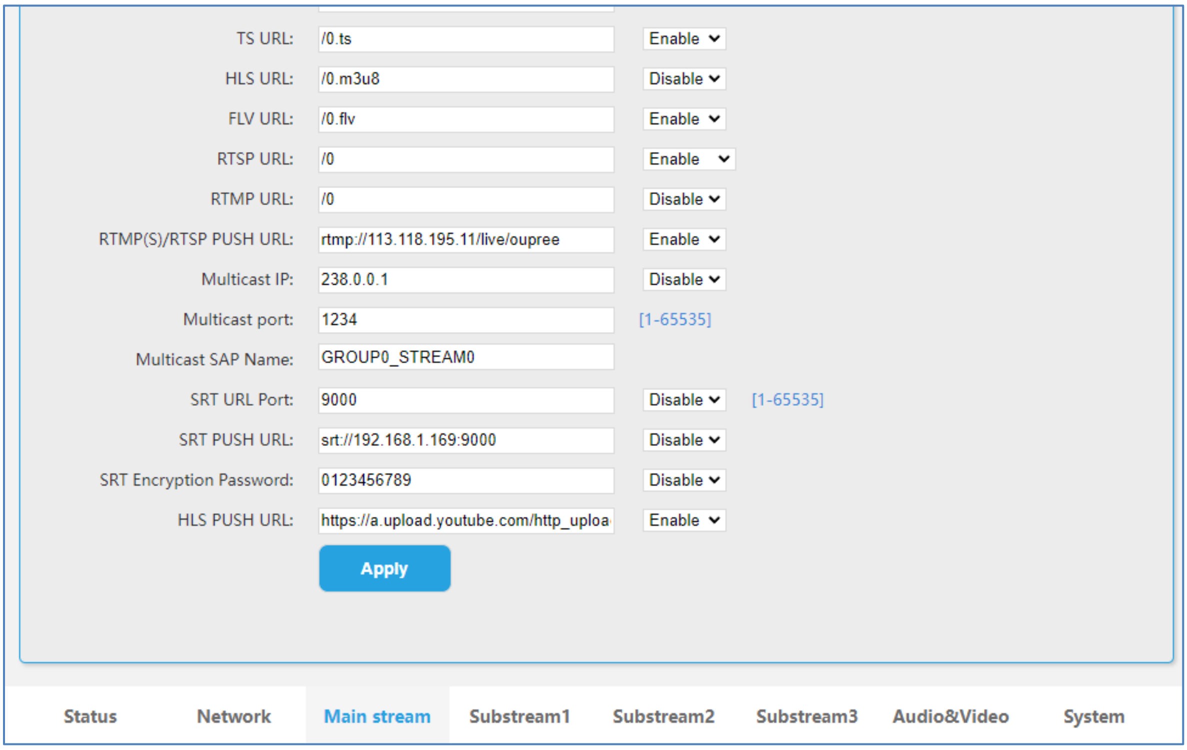
Do you want to **stream H.265 to YouTube**? Yes, **our H265 Video Encoder** supports HLS push to YouTube since the year 2021.

How to setup?

Step 1, get the **hls** stream **url** from YouTube, if you can't find the HLS stream settings, maybe you need read here: https://support.google.com/youtube/answer/10349430,



Step 2, copy the YouTube https-HLS streamURL and paste it to our **H265 Video Encoder,**

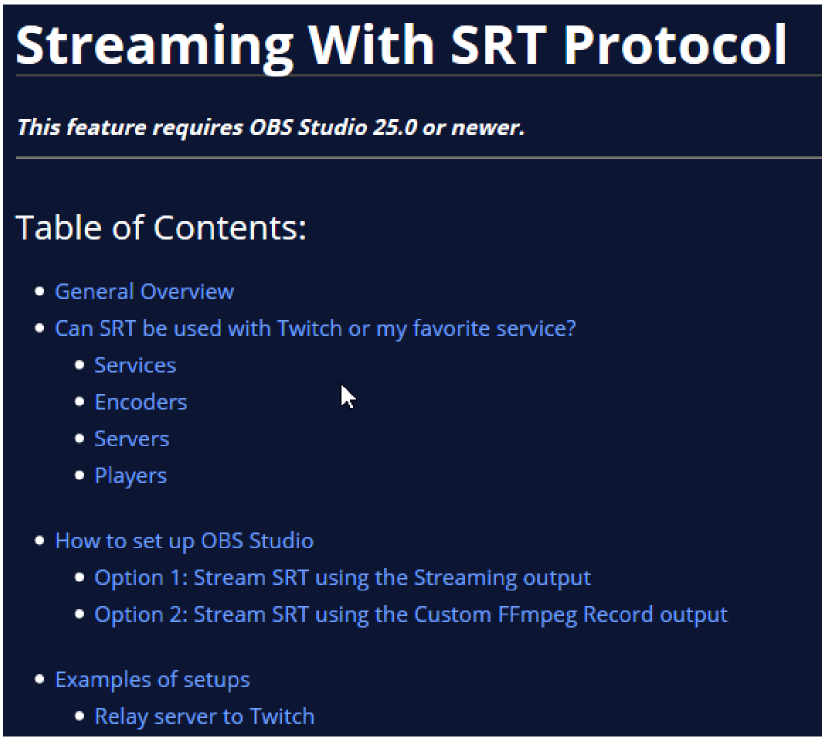


Addon Appendix about the firmware-file:

Do not modify any value where you are not 100% knowing what you are doing.
Because the firmware is specific for every model, HDMI or SDI with or w/o h.264 h.265 do not mix up them because the filename of the firmware is always up.rar .
Please use the backup only for saving a firmware for accidently cases where it might be necessary to go back a version.
Danger: If manipulating the up.rar firmware compressed file, do not use a modern 64bit rar packer because that will not work. 7Zip 32/64 will also not work.
So, we recommend not to pack that wrong because it can brick the unit and makes it non accessible any more.
Please use the backup only for saving a firmware for accidently cases where it might be necessary to go back a version. Like backing up before updating a unit…
If you accidently shoot it up, try the RST-RESET Button (5-10 sec. pressing until LED = off) to reload factory defaults.

Some more useful links regarding SRT:

A Media server to handle SRT and more: The Open Broadcaster Software
https://obsproject.com/
https://obsproject.com/wiki/Streaming-With-SRT-Protocol:



https://github.com/obsproject/obs-studio
https://github.com/haivision/srt

https://www.blankom.net/assets/downloads/How-to-connect-our-Video-Encoder-to-OBS.pdf

Another nice Project supporting SRT is RESTREAMER2:

https://datarhei.com/

Video Encoder & Decoder SRT settings as couple:

For HDMI/VGA&CVBS/SDI Decoder-Support h264 & h265, decoder SRT playing the URL as, here the encoder works as caller (SRT push URL) and listener (SRT URL port):

srt://ip:port *# encoder as Listener, decoder get srt from encoder, here 'ip' is the Encoder IP.*

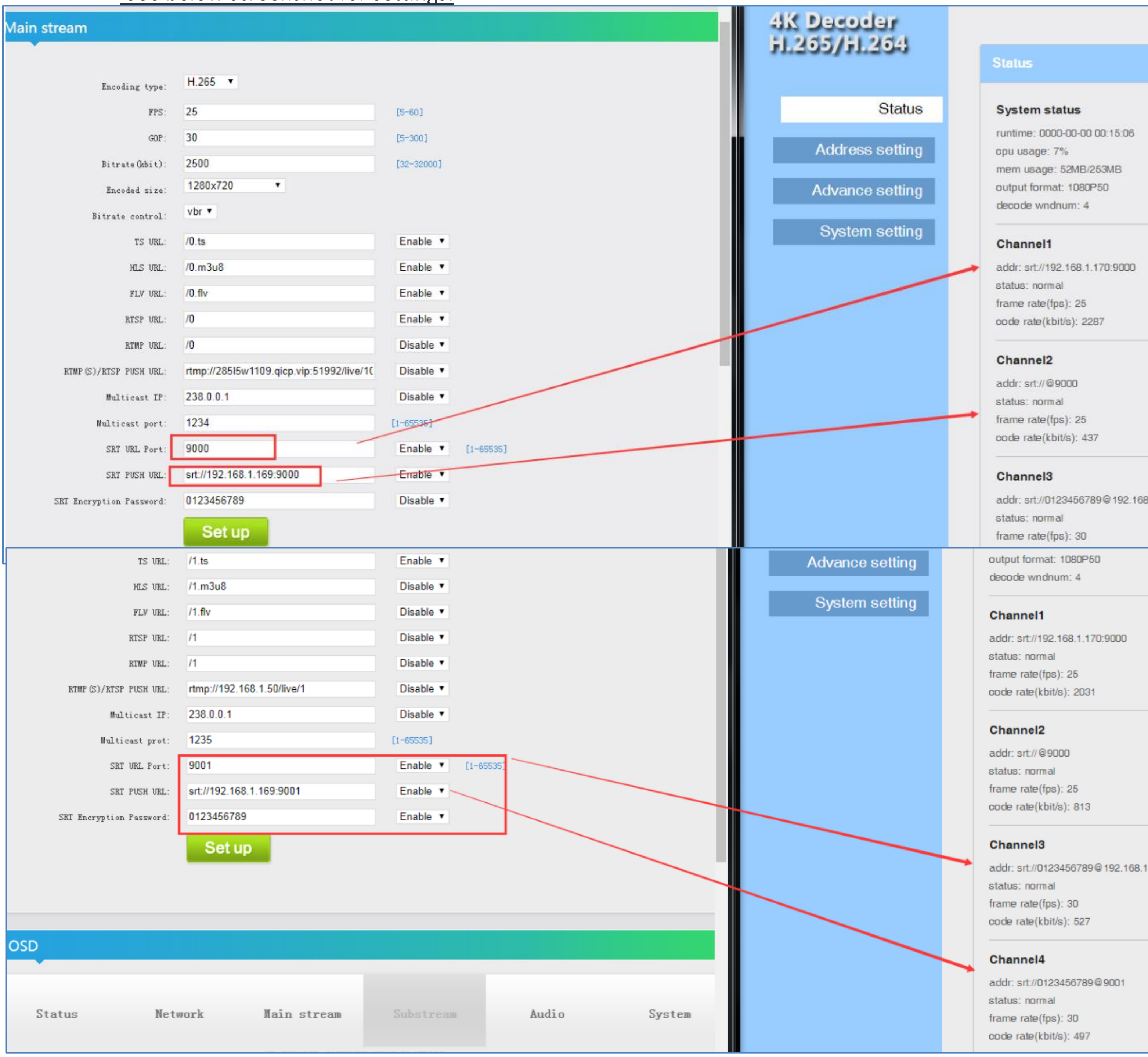
srt://port or srt://@port *# encoder mode as caller, push SRT to the decoder, (encoder SRT push URL as srt://decoder ip:port)*

With passphrase/Encryption, decoder SRT play URl:

srt://passpharese@ip:port *# encoder as Listener, decoder get SRT stream from encoder, here 'IP' is the Encoder IP.*

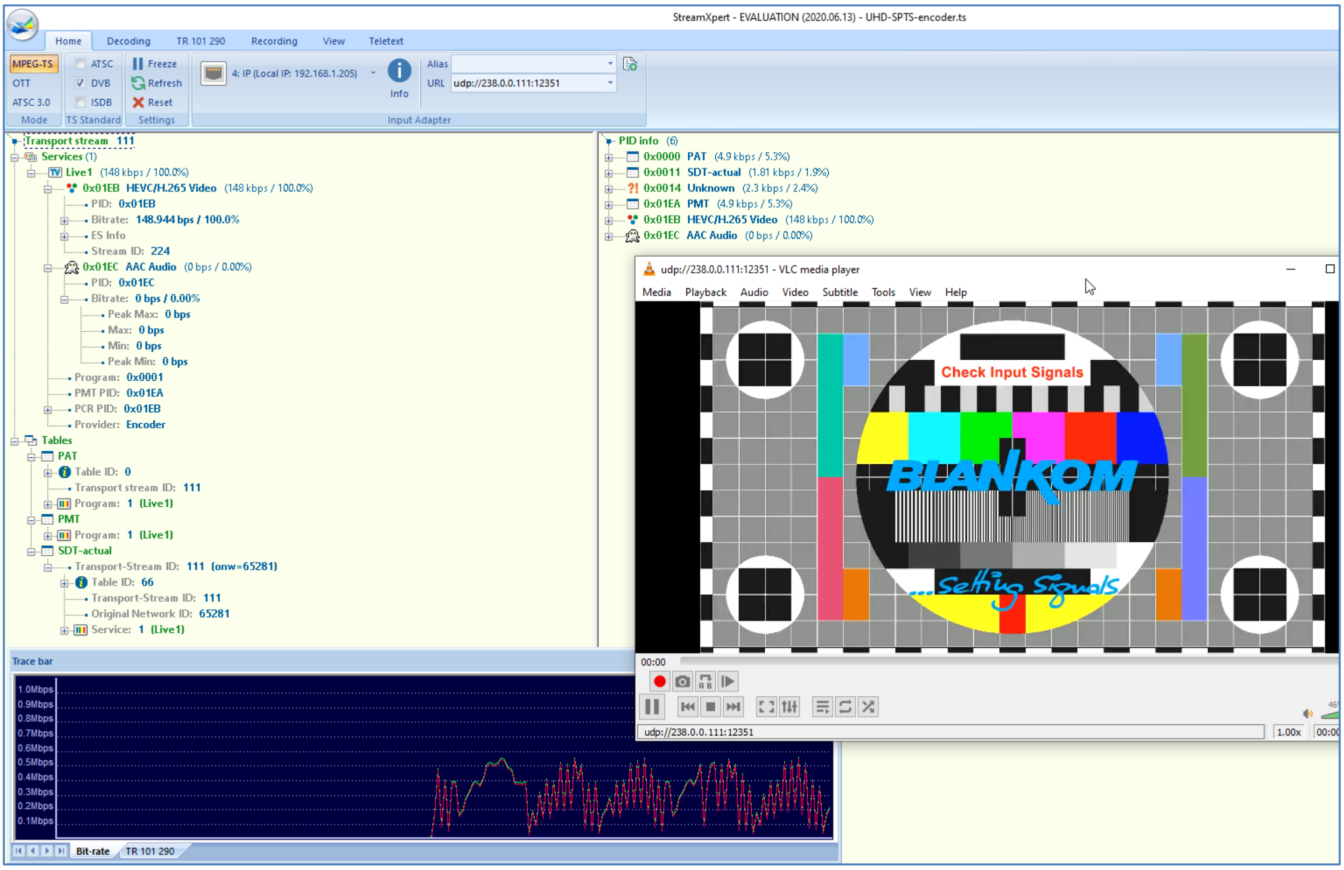
srt://passphrase@port *# encoder mode as caller, push srt to the decoder.*

See below screenshot for settings:



 All technical data are subject to change w/o further notice ....

If no signal has been detected at the Input connector, the Test-picture will appear and the Stream output may 'pump' because the encoder checks the input signal periodically – and in this periods', the output stream might fluctuate like:



Addendum: Serial to TCP

```
works in combination with an integrated remserial-1.4 function:
Remserial
---------
The remserial program acts as a communications bridge between a TCP/IP network port and a Linux
device such as a serial port.  Any character-oriented Linux /dev device will work.

The program can also use pseudo-ttys as the device.  A pseudo-tty is like a serial port in that
it has a /dev entry that can be opened by a program that expects a serial port device, except
that instead of belonging to a physical serial device, the data can be intercepted by another
program.
The remserial program uses this to connect a network port to the "master" (programming) side of
the pseudo-tty allowing the device driver (slave) side to be used by some program expecting a
serial port.  See example3 below for details.

The program can operate as a server accepting network connections from other machines, or as a
client, connecting to remote machine that is running the remserial program or some other program
that accepts a raw network connection.  The network connection passes data as-is, there is no
control protocol over the network socket.
Multiple copies of the program can run on the same computer at the same time assuming each is
using a different network port and device.

Some examples:
1) Give access to a RS232 device over a network.
The computer with the serial port connected to the device (such as a data aquisition device) runs
the remserial program:

        remserial -d -p 23000 -s "9600 raw" /dev/ttyS0 &

This starts the program in daemon mode so that it runs in the background, it waits for onnections
on port 23000 and sets up the serial port /dev/ttyS0 at 9600 baud.  Network connections to port
23000 from any machine can then read and write to the device attached to the serial port.
This can be started from /etc/rc.local or as an entry in /etc/inittab or set up as a system
service with a file in /etc/rc.init/.

2) Connect an RS232 device to a specified server.
```

The computer with the serial port connected to the device (such as a data acquisition device) runs the remserial program:

```
        remserial -d -r server-name -p 23000 -s "9600 raw" /dev/ttyS0 &
```

This would be used with case number 1 above creating an end-to-end serial port connection. What goes in the serial port on one machine would come out the serial port of the other machine. The ports could be running at different baud rates or other serial port settings.

3) Connect a Linux program that needs a serial port to a remote serial port.
Some programs are written to communicate directly with a serial port such as some data acquisition programs. The remserial program can use pseudo-ttys to fool the program into thinking that it is talking to a real serial port on the local machine:

```
        remserial -d -r server-name -p 23000 -l /dev/remserial1 /dev/ptmx &
```

This creates a file called /dev/remserial1 which can be used by the data acquisition application as its serial port. Any data sent or received is passed to the remote server-name on port 23000 where a computer configured in case number 1 above passes it to a real serial port.

The remserial program uses the special pseudo-tty master device /dev/ptmx (see man ptmx) which creates a slave device that looks like a normal serial port named /dev/pts/something. Unfortunately, the actual device name created isn't consistent, so the remserial program creates a symbol link from the device name specified with the -l option to the /dev/pts/ name that was created allowing the other application to be configured with a consistent device name.

4) Server farm console control.
Assuming multiple Linux servers (such as web servers) are set up to have a serial port as their console instead of a monitor/keyboard, their serial ports could be connected to a control server using a multi-port serial board.
On the control server, a copy of remserial is run for each server:

```
        remserial -d -p 23000 -s "115200 raw" /dev/ttyS0 &
        remserial -d -p 23001 -s "115200 raw" /dev/ttyS1 &
        remserial -d -p 23002 -s "115200 raw" /dev/ttyS2 &
        remserial -d -p 23003 -s "115200 raw" /dev/ttyS3 &
        etc.
```

From any computer on the local network, use a telnet program to connect to the control server on the appropriate port:

```
        telnet control-server-name 23002
```

This would connect through the associated serial port to the desired server's console. This example would then give the user console access to the 3rd server.

Careful scripting such as using the Linux "expect" program could allow batches of commands to be run on each server.

Other Linux program useful with remserial

- nc - The netcat program is similar to remserial except that it creates connections between network ports and command line standard input and output.

  For example, with case number 1 above, the following command run on another computer will send the contents of the named file out the serial port used by the remserial program:

```
  nc server-name 23000 <file-name
```

  Similarily, the following command will store incoming serial data in a file until the program is manually interrupted:

```
  nc server-name 23000 >file-name
```

- telnet - The telnet program is normally used to log into a remote computer, but when used with network ports other than number 23, it operates in a raw data mode.

  For example, with case number 1 above, the following command will allow the user of the telnet program to see incoming serial port data and type data on the keyboard to send to the serial port:

```
  telnet server-name 23000
```

  This is ideal for controlling the device connected to the serial port if it has some sort of command line interface usable over the serial port.

remserial Usage:

```
remserial [-r machinename] [-p netport] [-s "stty params"] device

-r machinename        The remote machine name to connect to.  If not
                      specified, then this is the server side.
-p netport            Specifiy IP port# (default 23000)
-s "stty params"      If serial port, specify stty parameters, see man stty
-d                    Run as daemon programs
-x debuglevel         Set debug level, 0 is default, 1,2 give more info
-l linkname           If the device is /dev/ptmx, creates a symbolic link
                        to the corresponding slave pseudo-tty so that another
```

```
                        application has a static device name to use.
-m max-connections      Maximum number of simultaneous client connections to allow
device                  Character oriented device node such as /dev/ttyS0.
```

Some useful links: or just go to www.blankom.de -> tutorials / know how
https://www.blankom.net/assets/downloads/TCP-IP-Unicast-IGMP-Dialog_und_snooping.pdf
https://www.blankom.net/assets/downloads/Whitepaper-streaming-protocols.pdf
https://www.blankom.net/assets/downloads/SRT_Protocol_Technical_Overview-blankom2.pdf
https://www.blankom.net/assets/downloads/Difference_IGMP-Dialog_and_snooping-DE_EN.pdf
https://www.blankom.net/assets/downloads/mpegts-introduction.pdf

Version History:

Topics	Author	Version	Date
Initial	RR	V1.0	17.01.2023