

# COBALT<sup>®</sup>

## How to Stream HDR Content to YouTube Using the Cobalt 9992-ENC

### Introduction

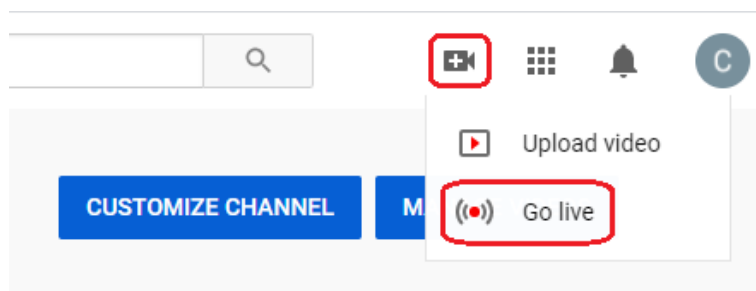
YouTube recently announced support for HDR content. The Cobalt 9992-ENC is capable of transmitting live HDR content to YouTube, both in HD and 4K resolutions. While HDR is available for both H.264 (AVC) and H.265 (HEVC), YouTube elected to support it only for HEVC.

In order to stream live HDR to YouTube, you need the following:

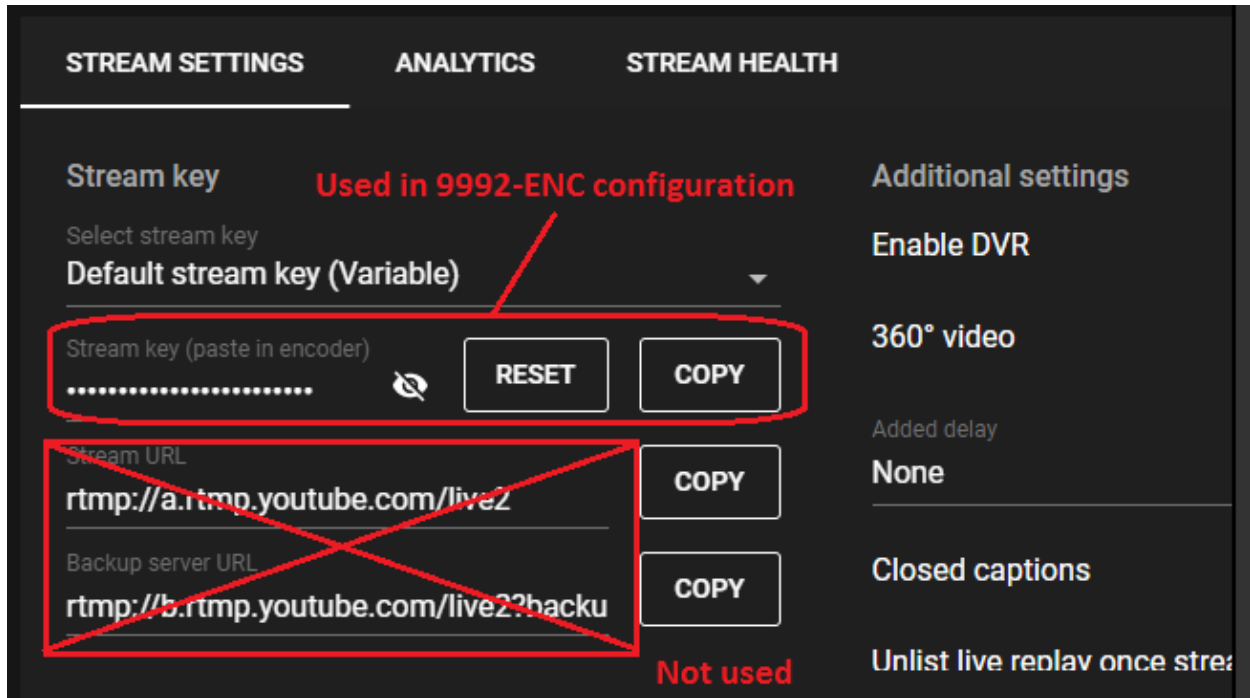
1. A 9992-ENC with appropriate licensing to work in HEVC mode. This can be:
  - a. Model [9992-ENC-4K-HEVC](#).  
or
  - b. Model [9992-ENC](#) with the optional **+HEVC** license.
2. An HDR video source in PQ10 format. If you have an HDR source in some other format, or if you have an SDR source that want to convert to HDR (by using inverse tone mapping), you can use the Cobalt [9904-UDX-4K](#) with the SDR/HDR Conversion Options (**+HDR-TCHCLR-4K**, **+HDR-TCHCLR**).
3. A YouTube account enabled for live streaming.

### YouTube Configuration

Prior to starting the encoder, you need to create a live broadcast at YouTube. Go to your channel, click on the indicated icon at the top of the page, and select **Go Live**:



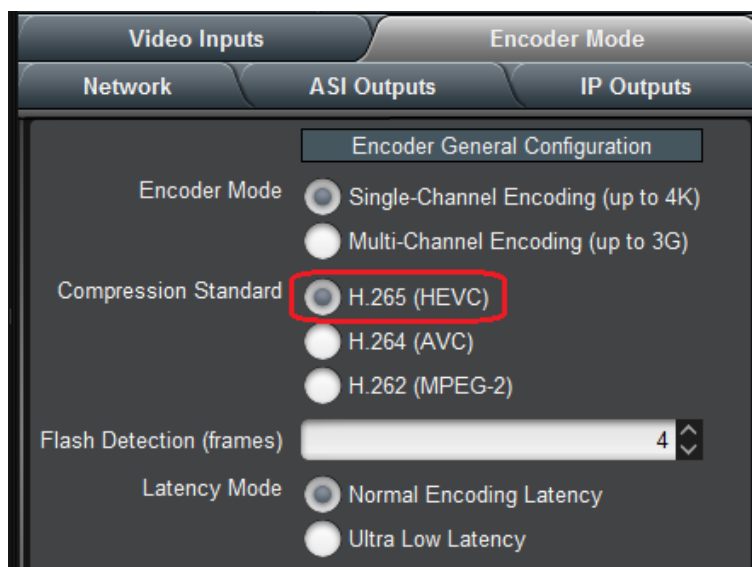
Once you select this, the following page opens:



The most important part is the **Stream Key**, which is required for the 9992-ENC configuration. Also note that the **Stream URL** part is not used – we will be streaming HEVC content, which is not supported by RTMP. The streaming will be through HLS, and the URL is different.

## 9992-ENC Configuration

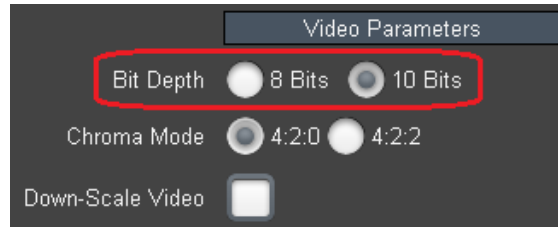
While the 9992-ENC supports HDR for both AVC and HEVC, YouTube only supports it for HEVC. Therefore, the first step is to configure the 9992-ENC for HEVC support. This is done in the **Encoder Mode** tab, as shown below.



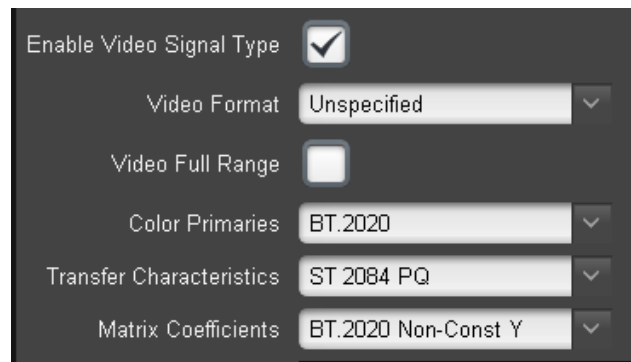
The other settings in this tab can be chosen to suit your needs.

### Encoder Video Configuration

HDR operation requires 10-bit encoding mode. This is selected in the encoder **Basic** tab, under **Video Parameters**:



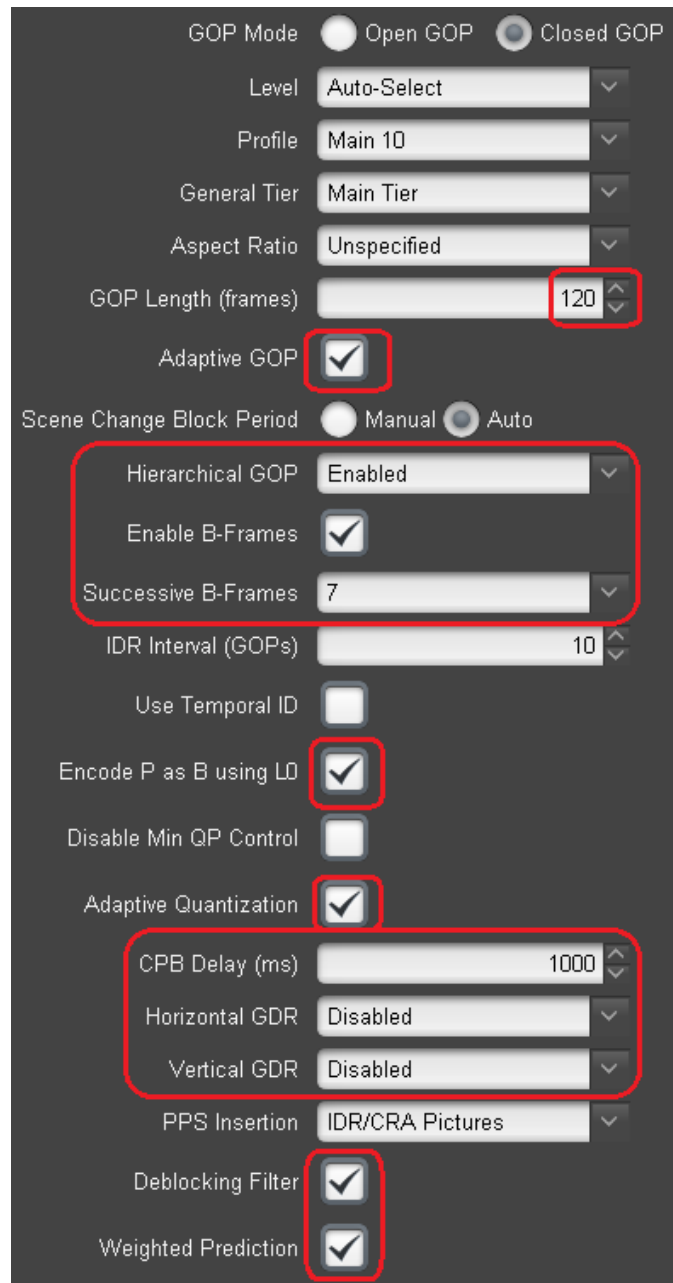
YouTube requires appropriate video signal metadata. This can be done in the encoder **Advanced** tab. When you check the **Enable Video Signal Type** box, the other parameters will become available. Here are the appropriate settings:



However, make sure that the **Color Primaries** match the actual signal.

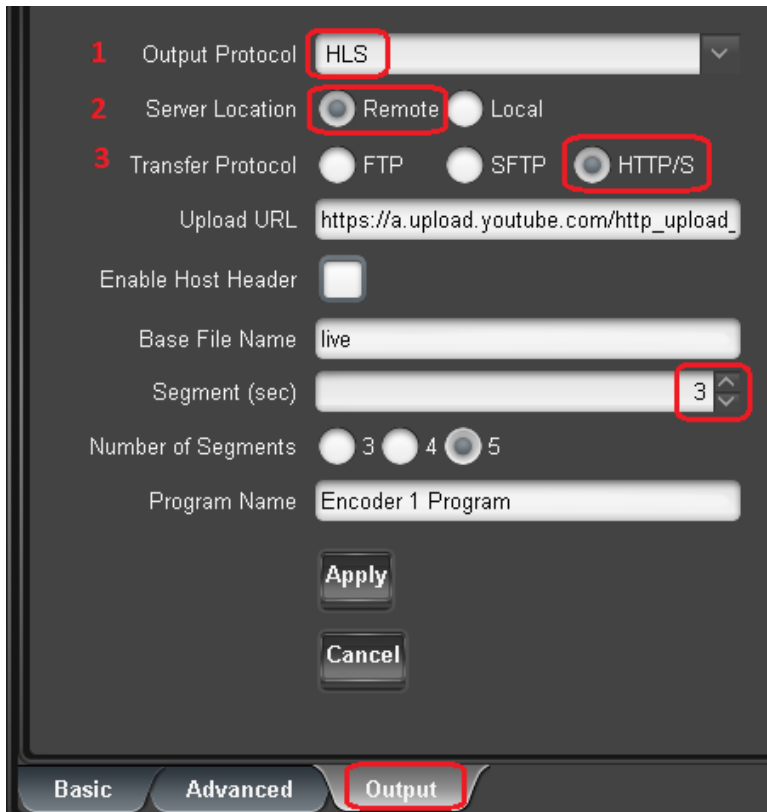
### Encoder Quality Optimization

These settings are not required for interoperability with YouTube, but are recommended to get the best quality out of your 9992-ENC. They are found in the encoder **Advanced** tab and are illustrated below.



### Encoder Output Configuration

Since HEVC is not supported by RTMP, the stream has to be supplied to YouTube as an HLS stream. The standard way to send HLS streams to a remote server is through HTTP Push. These selections are made in the encoder **Output** tab, illustrated below:



The steps are:

1. In the **Output** tab, select **HLS** in the Output Protocol drop down (step 1 above).
2. For **Server Location**, select **Remote** (step 2 above).
3. For **Transfer Protocol**, select **HTTP/S** (step 3 above).
4. In the **Upload URL** field, enter the following:  
**https://a.upload.youtube.com/http\_upload\_hls?cid=xxxx-xxxx-xxxx-xxxx-xxxx&copy=0&file=**  
Replace **xxxx-xxxx-xxxx-xxxx-xxxx** with your YouTube stream key, which you obtained from the YouTube web site (see YouTube Configuration above).
5. Leave the **Enable Host** Header box unchecked. The **Base File Name** field can also be left at the default value of “live” (YouTube does not care what this is, but a value must be supplied).
6. Choose a suitable segment duration (in the **Segment (sec)** field). YouTube prefers a value between 2 and 4 seconds. Larger values are more robust to varying network conditions.
7. Any of the choices for the **Number of Segments** will work. YouTube prefers 5.
8. YouTube does not use the **Program Name**. This field can be left untouched.

Once you have all of this configured, start the encoder. Note that it may take YouTube up to 30 seconds to start showing the video.