

Using SCTE for Inputs and Outputs

This article will explain how you can use SCTE for your projects. PLAYDECK supports SCTE-35, which are exclusively used in Streams and SCTE-104, which are exclusively used in SDI Device signals.

Scroll down for SCTE-104. We also recommend to follow-up with this article on sending SCTE-35 to Streaming Server.

In this article:

- Using SCTE-35 in Streams
- Using SCTE-104 in SDI

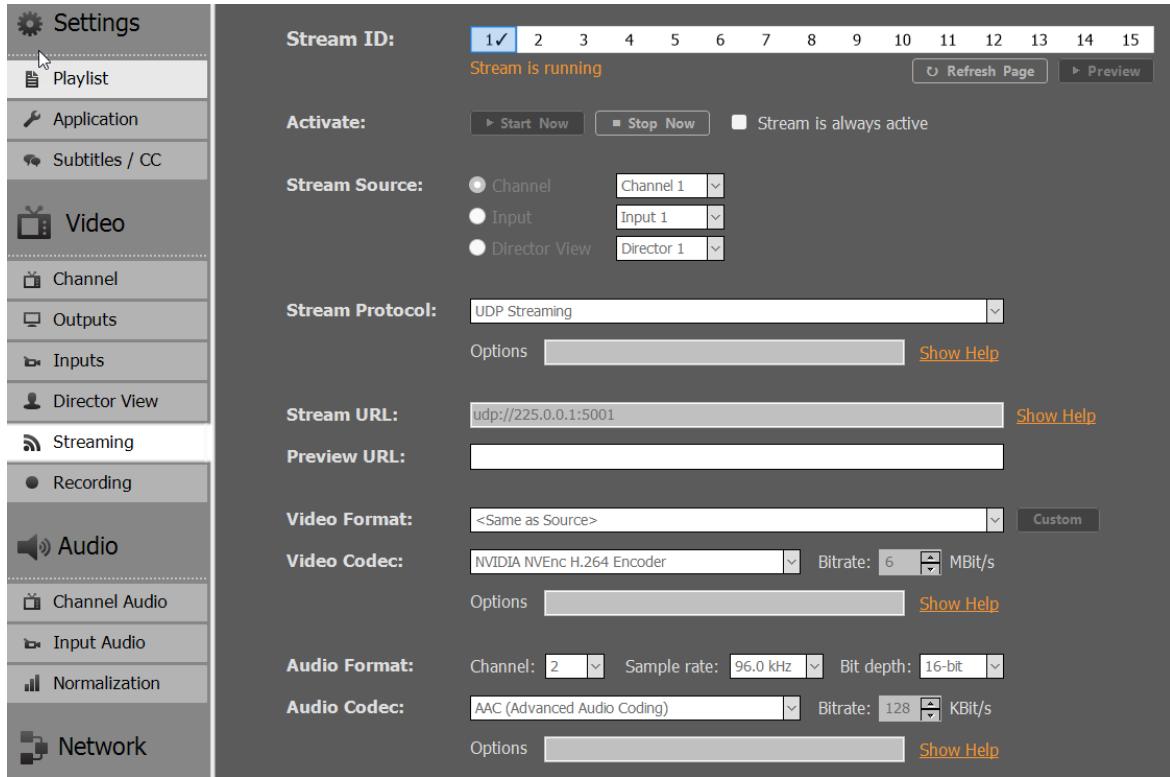
Using SCTE-35 in Streams

SCTE-35 Facts

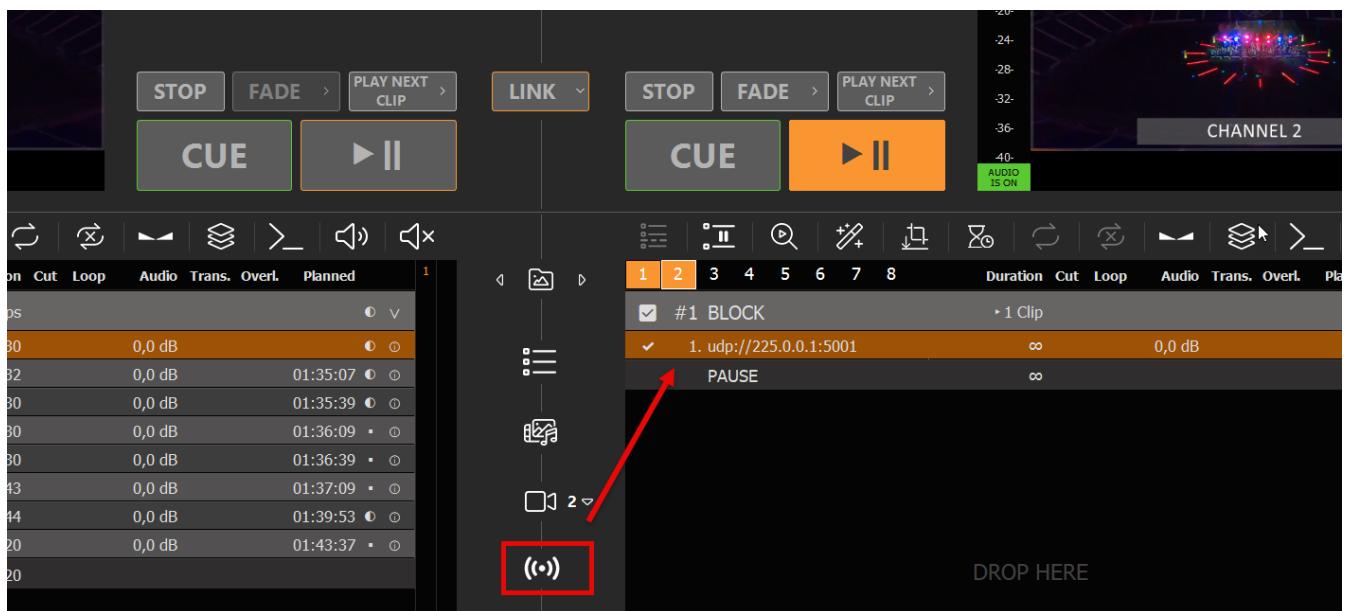
- Supported for Stream Inputs as well as Stream Outputs. Tested Stream Formats are UDP, DVB and SRT. The selection of the Video Codec does not influence SCTE transport. SCTE transport is always active and does not need to be enabled somewhere.
- PLAYDECK can detect and display incoming SCTE-35 Marker. All incoming and outgoing Marker will be written to an Event Log.
- PLAYDECK can forward incoming SCTE-35 Marker from Stream Input to Stream Output. Cross-Forwarding between UDP, DVB and SRT is supported, e.g. receiving SRT and forwarding to UDP.

SCTE-35 Testdrive

Let us dive into those topics a bit. As a test run we setup our own UDP signal loop. We stream Channel 1 to “`udp://225.0.0.1:5001`” via UDP:

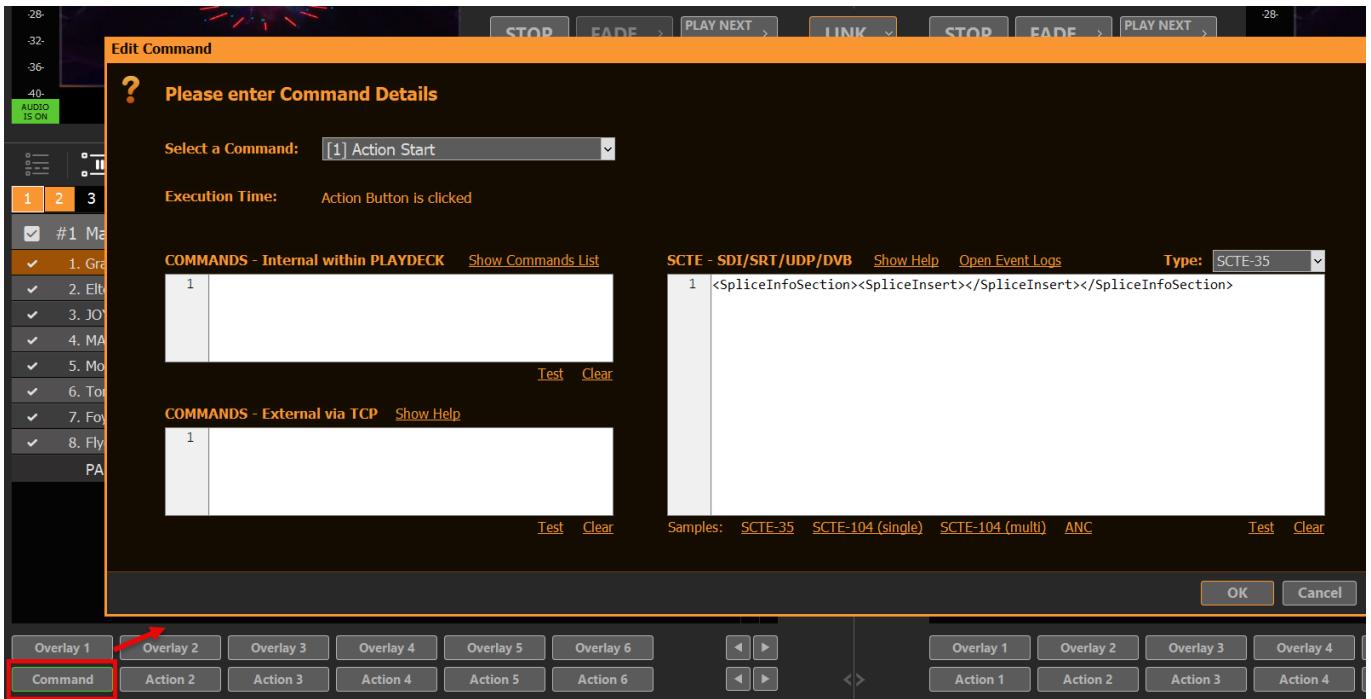


And play the Stream in Channel 2:

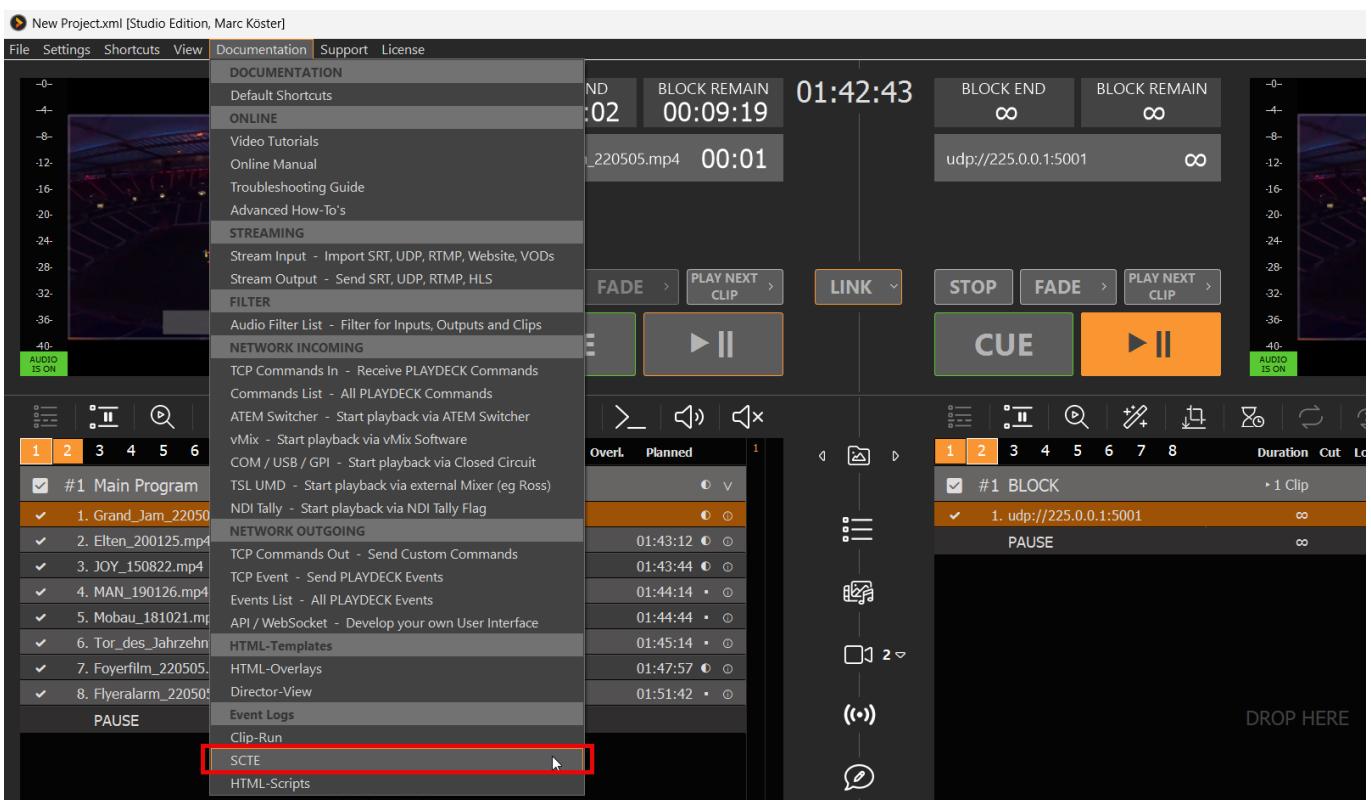


Now we only need a way to send a SCTE-35 marker in Channel 1. So we setup a new Command Action and use the smallest detectable SCTE-35 Marker (of course you can use any other):

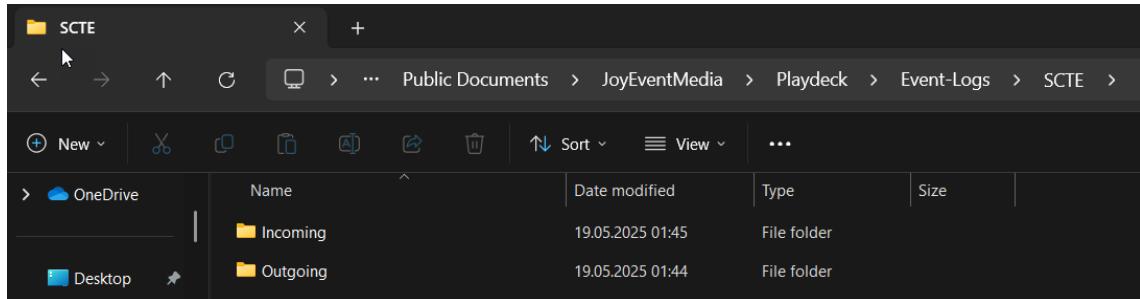
```
<SpliceInfoSection><SpliceInsert></SpliceInsert></SpliceInfoSection>
```



Make sure Channel 1 and Channel 2 are playing, then hit the Action Button. Then open the SCTE Event Log over the Main Menu > Documentation > SCTE:



You will see the outgoing Marker of your Action Button Click on Channel 1, and the detected incoming Marker on Channel 2:



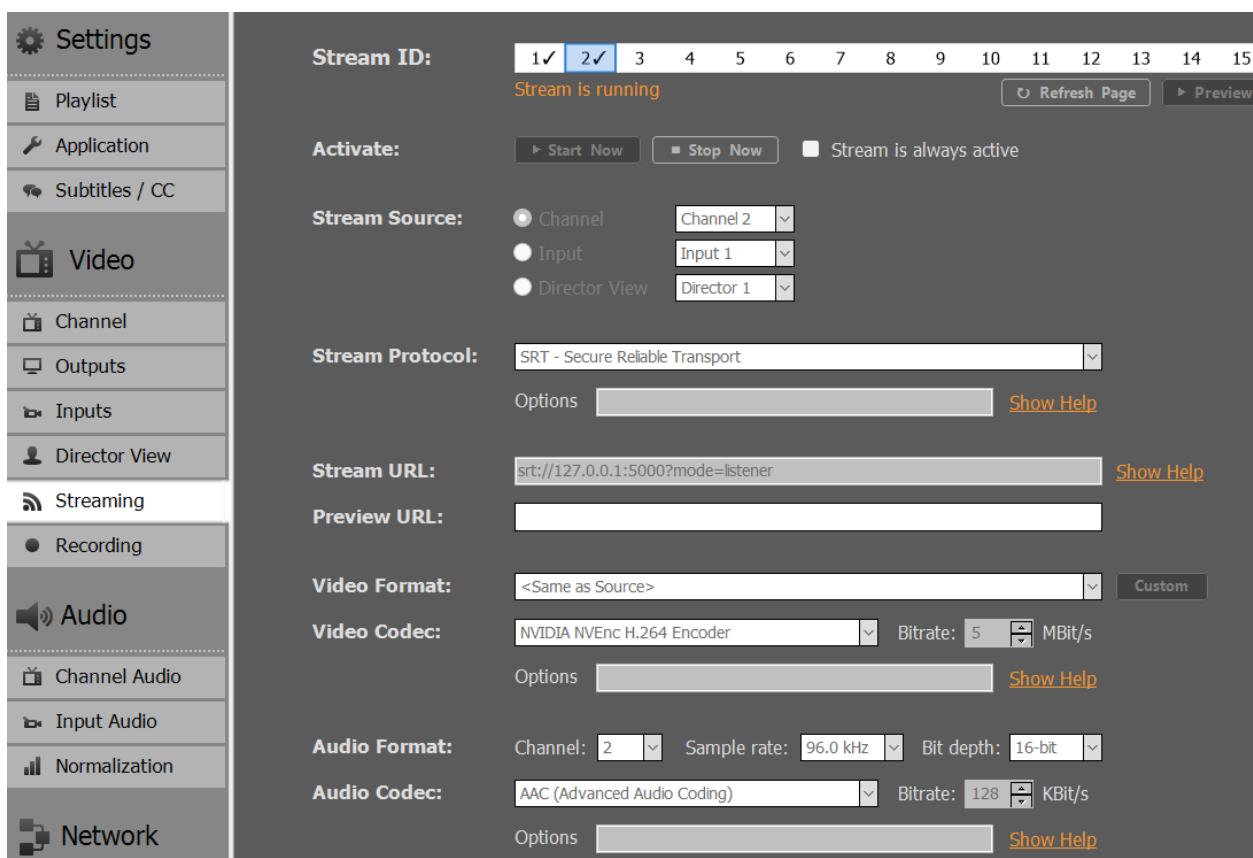
```

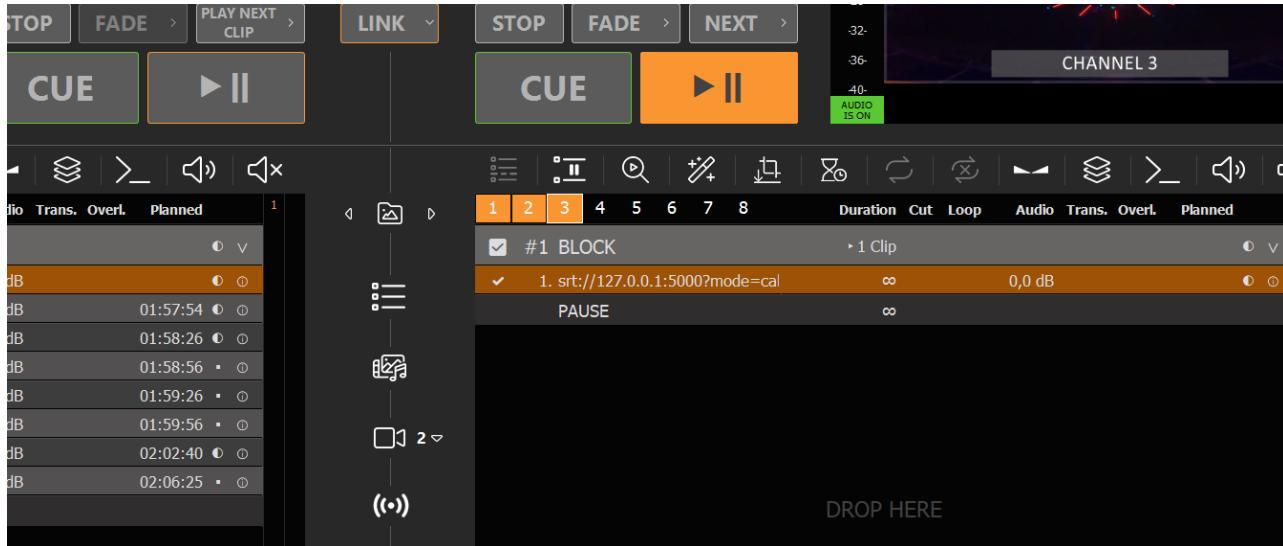
19.05.2025 01:44:58 - Ch 1 SCTE-35 - <SpliceInfoSection><SpliceInsert></SpliceInsert></SpliceInfoSection>
19.05.2025 01:45:02 - Ch 2 SCTE-35 - <SpliceInfoSection protocolVersion='0' ptsAdjustment='0' tier='0'>      <
SpliceInsert spliceEventId='0' spliceEventCancelIndicator='0' outOfNetworkIndicator='0' program_splice_flag='0'
spliceImmediateFlag='0' uniqueProgramId='0' availNum='0' availsExpected='0'/>  </SpliceInfoSection>

```

We even go one step further and setup ANOTHER Stream from Channel 2 to Channel 3 to test the SCTE Forwarding and Cross-over Detection (UDP to SRT). We send Channel 2 to "srt://127.0.0.1:5000?mode=listener" and receive in Channel 3 at "srt://127.0.0.1:5000?mode=caller".

Note: If the IP 127.0.0.1 doesn't work, please use your real local IP, e.g. 192.168.178.42. See Main Menu > License > About to check your Local IP address.





If we now press our Action Button to send the SCTE-35 Marker, we detect it also in Channel 3:

```

19.05.2025 01:45:02 - Ch 2 SCTE-35 - <SpliceInfoSection protocolVersion='0' ptsAdjustment='0' tier='0'>      <
SpliceInsert spliceEventId='0' spliceEventCancelIndicator='0' outOfNetworkIndicator='0' program_splice_flag='0'
spliceImmediateFlag='0' uniqueProgramId='0' availNum='0' availExpected='0' />  </SpliceInfoSection>
19.05.2025 01:54:46 - Ch 2 SCTE-35 - <SpliceInfoSection protocolVersion='0' ptsAdjustment='0' tier='0'>      <
SpliceInsert spliceEventId='0' spliceEventCancelIndicator='0' outOfNetworkIndicator='0' program_splice_flag='0'
spliceImmediateFlag='0' uniqueProgramId='0' availNum='0' availExpected='0' />  </SpliceInfoSection>
19.05.2025 01:54:48 - Ch 3 SCTE-35 - <SpliceInfoSection protocolVersion='0' ptsAdjustment='0' tier='0'>      <
SpliceInsert spliceEventId='0' spliceEventCancelIndicator='0' outOfNetworkIndicator='0' program_splice_flag='0'
spliceImmediateFlag='0' uniqueProgramId='0' availNum='0' availExpected='0' />  </SpliceInfoSection>

```

Using SCTE-104 in SDI

SCTE-104 Facts

- Supported for SDI Device Inputs as well as SDI Device Outputs, e.g. Blackmagic DeckLink. SCTE transport is always active and does not need to be enabled somewhere.
- PLAYDECK can detect and display incoming SCTE-104 Marker. All incoming and outgoing Marker will be written to an Event Log.
- PLAYDECK can forward incoming SCTE-104 Marker from SDI Input to SDI Output.

SCTE-104 Testdrive

Let us dive into those topics a bit. As a test run we setup our own SDI signal loop. We use our DeckLink Duo 2 and an SDI Loop Cable:

Outputs

Input ID: 1✓ 2 3 4 5 6 7 8 9 10 11 12

Input Name: INPUT 1 [Update to Playlist >](#)

Crop/Aspect: ➡ 0 ⬇ 0 ⬆ 0 ⬅ 0 Letterbox / Pillarbox [Update >](#)

Time shifting: Active Delay: 0 0 7 HH:MM:SS

Device Input: Device: DeckLink Duo 2 (2)

Line: SDI Video & SDI Audio

Format: <Auto/Variable>

Audio: <No Audio>

Background: <Black> Image: [Select Image](#)

Options: [Show Help](#)

Desktop Input: Monitor: NVIDIA GeForce RTX 3080 - 3840x1600@144,00 - PRIMARY

Audio: <No Audio>

And play the SDI Input in Channel 2:

CHANNEL 2

AUDIO IS ON

PAUSE

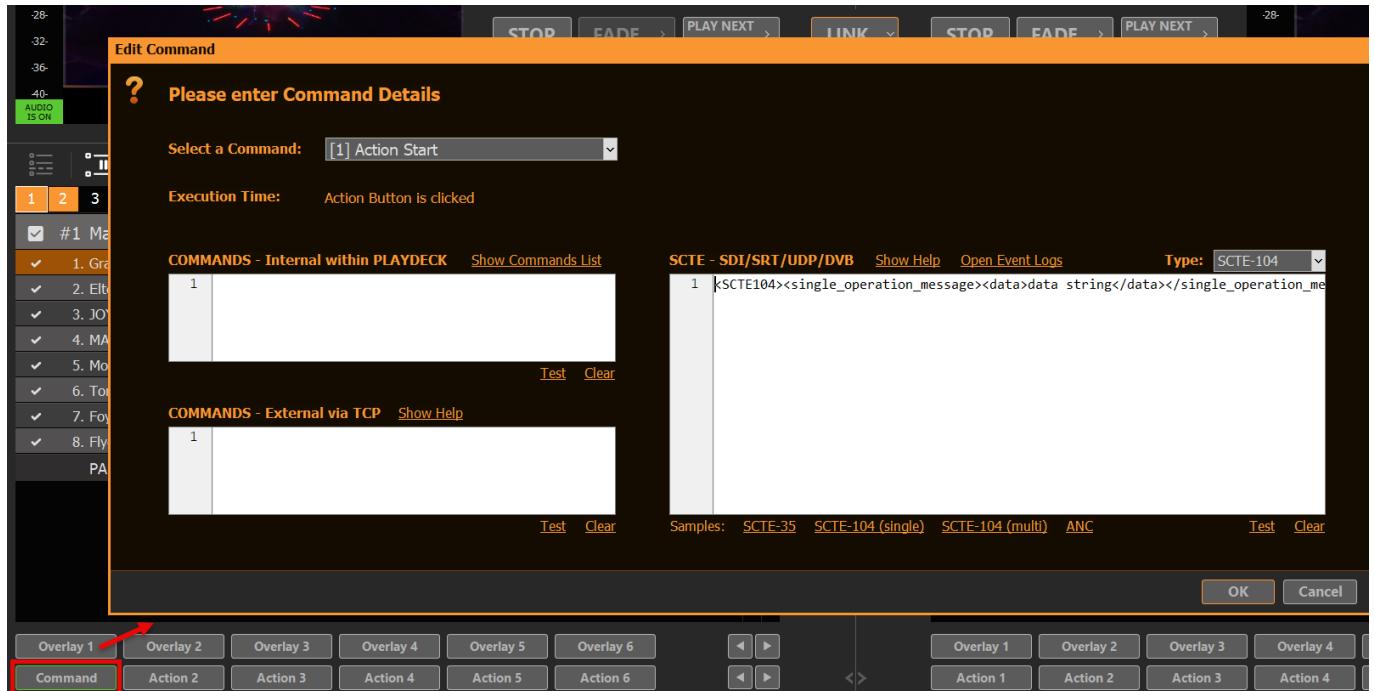
1. INPUT 1

PAUSE

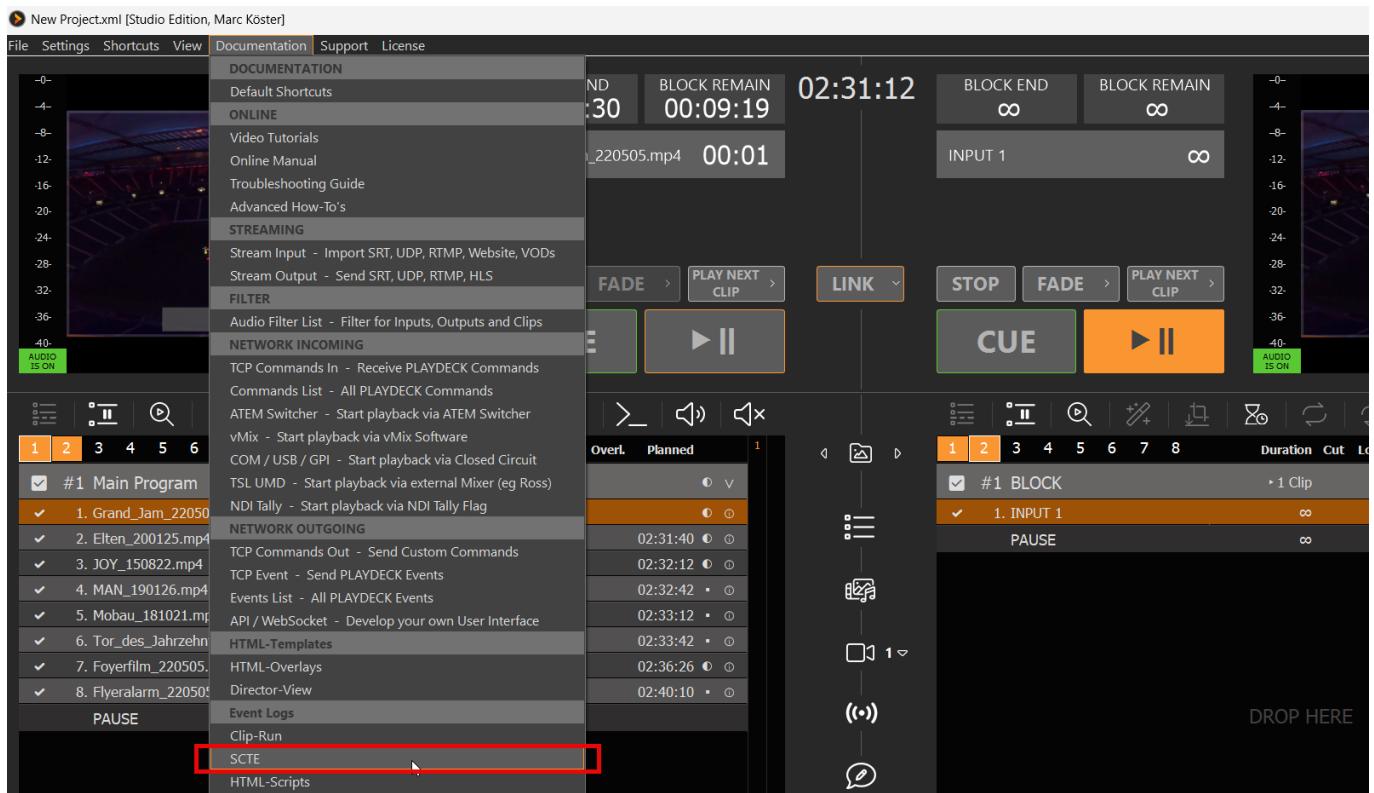
Now we only need a way to send a SCTE-104 marker in Channel 1. So we setup a new Command Action and use the smallest detectable SCTE-104 Marker (of course

you can use any other):

```
<SCTE104><single_operation_message><data>data
string</data></single_operation_message></SCTE104>
```



Make sure Channel 1 and Channel 2 are playing, then hit the Action Button. Then open the SCTE Event Log over the Main Menu > Documentation > SCTE:



You will see the outgoing Marker of your Action Button Click on Channel 1, and the detected incoming Marker on Channel 2:

File Explorer showing the directory structure and log files:

```

SCTE
  + Playdeck PLA
    + Incoming
    + Outgoing
  + This PC
  19.05.2025 02:29:49 - Ch 1 SCTE-104 - <SCTE104><single_operation_message><data>data
  string</data></single_operation_message></SCTE104>
  19.05.2025 02:29:49 - Ch 2 SCTE-104 - <single_operation_message> <opID>0</opID> <messageSize>
  20</messageSize> <result>0</result> <result_extension>0</result_extension> <protocol_version>
  0</protocol_version> <AS_index>0</AS_index> <message_number>0</message_number> <DPI_PID_index>
  0</DPI_PID_index> <data> <general_response_data/> </data> </single_operation_message>

```

We even go one step further and setup ANOTHER SDI Loop from Channel 2 to Channel 3 to test the SCTE Forwarding:

Screenshot of the Playdeck software interface showing the configuration for Channel 2 and Channel 3.

Channel ID: 1✓ 2✓ 3 4 5 6 7 8 Output is running

Output Scaler: Position (0, 0) / (0, 0) Pixel X/Y

Scale type: Original Size (1920 / 1080 Pixel X/Y) Lock X/Y

Device Output: Device: DeckLink Duo 2 (3)
Line: SDI
Keying: <None> / Straight Alpha
Options: Show Help

Desktop Output: Monitor: <Window Mode>
Audio: <No Audio>

Input ID: 1✓ 2✓ 3 4 5 6 7 8 9 10 11 12 Input is running

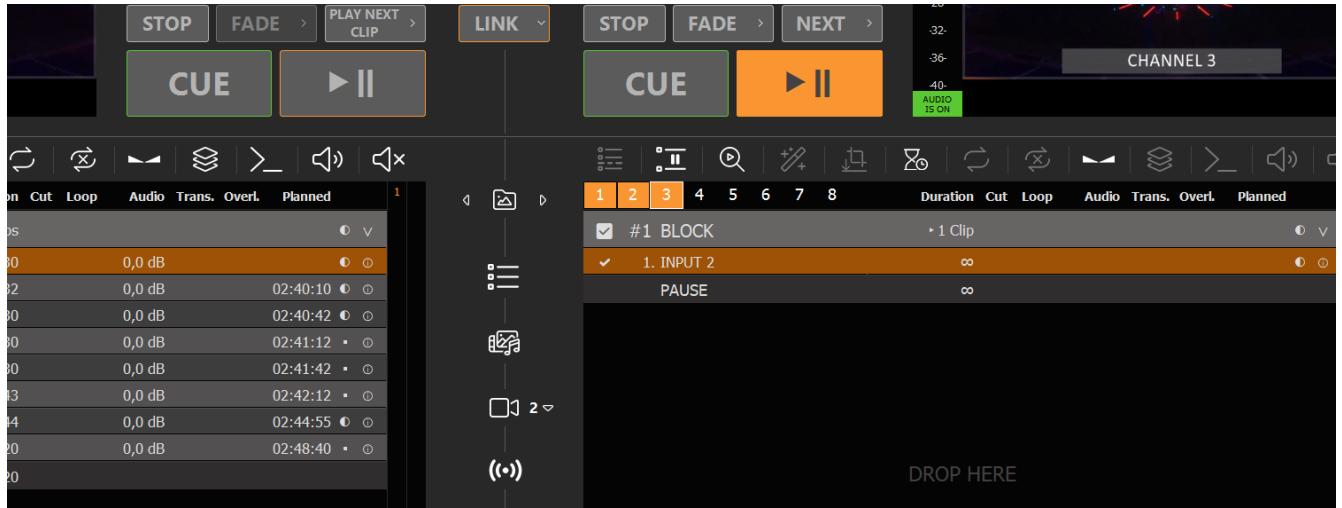
Input Name: INPUT 2 Update to Playlist

Crop/Aspect: 0 0 0 0 Letterbox / Pillarbox Update

Time shifting: Active Delay: 0 10 0 HH:MM:SS

Device Input: Device: DeckLink Duo 2 (4)
Line: SDI Video & SDI Audio
Format: <Auto/Variable>
Audio: <Same as Video Device>
Background: <Black> Image:
Options: Show Help

Desktop Input: Monitor: NVIDIA GeForce RTX 3080 - 3840x1600@144,00 - PRIMARY
Audio: <No Audio>



If we now press our Action Button to send the SCTE-104 Marker, we detect it also in Channel 3:

```

19.05.2025 02:29:49 - Ch 2 SCTE-104 - <single_operation_message>    <opID>0</opID>      <messageSize>
20</messageSize>  <result>0</result>    <result_extension>0</result_extension>    <protocol_version>
0</protocol_version>  <AS_index>0</AS_index>    <message_number>0</message_number>    <DPI_PID_index>
0</DPI_PID_index>  <data>        <general_response_data/>  </data>  </single_operation_message>
19.05.2025 02:40:28 - Ch 2 SCTE-104 - <single_operation_message>    <opID>0</opID>      <messageSize>
20</messageSize>  <result>0</result>    <result_extension>0</result_extension>    <protocol_version>
0</protocol_version>  <AS_index>0</AS_index>    <message_number>0</message_number>    <DPI_PID_index>
0</DPI_PID_index>  <data>        <general_response_data/>  </data>  </single_operation_message>
19.05.2025 02:40:28 - Ch 3 SCTE-104 - <single_operation_message>    <opID>0</opID>      <messageSize>
20</messageSize>  <result>0</result>    <result_extension>0</result_extension>    <protocol_version>
0</protocol_version>  <AS_index>0</AS_index>    <message_number>0</message_number>    <DPI_PID_index>
0</DPI_PID_index>  <data>        <general_response_data/>  </data>  </single_operation_message>

```