ActiveSQX Troubleshooting

1 Contents

2 Introduction

This document describes the support-oriented functionality that has been implemented within the ActiveSXQ card and the process that Datapath will follow, using these features, to swiftly investigate and remedy any problems that may be found when deploying the ActiveSQX.

The document describes the steps that should be taken when an IP stream source does not work with the ActiveSQX card. An IP stream may pass through a diverse set of hardware and software components and there are an equally large number of reasons why the stream may fail to work. The document covers the potential issues in order from most common to least common. The common issues could be diagnosed by the end-user or fist-line support without too much trouble however some of the less common issues will need to be escalated to Datapath for further investigation and the processes involved in that are also described.

3 Common issues

When a window is first opened to display an IP stream, "Connecting" will appear in the window. This indicates that the ActiveSQX card is attempting to establish a connection to the IP stream source. This may take anywhere from a fraction of a second to half a minute depending on the quality of the network and source.

In case of networking issues an error maybe displayed in the window immediately or the software may time out after approximatively 30 seconds and then display an error message in the window.

The different types of networking issues are described in section 3.1. Assuming the network connection can be established correctly, various issues with decoding such as trying to decode a stream which uses an encoding format or specific feature of that format which the ActiveSQX does not support may be encountered. These are described in section 3.2.

3.1 Networking connection issues

If there are issues with the network one of the following errors may appear:

3.1.1 Connection Failed

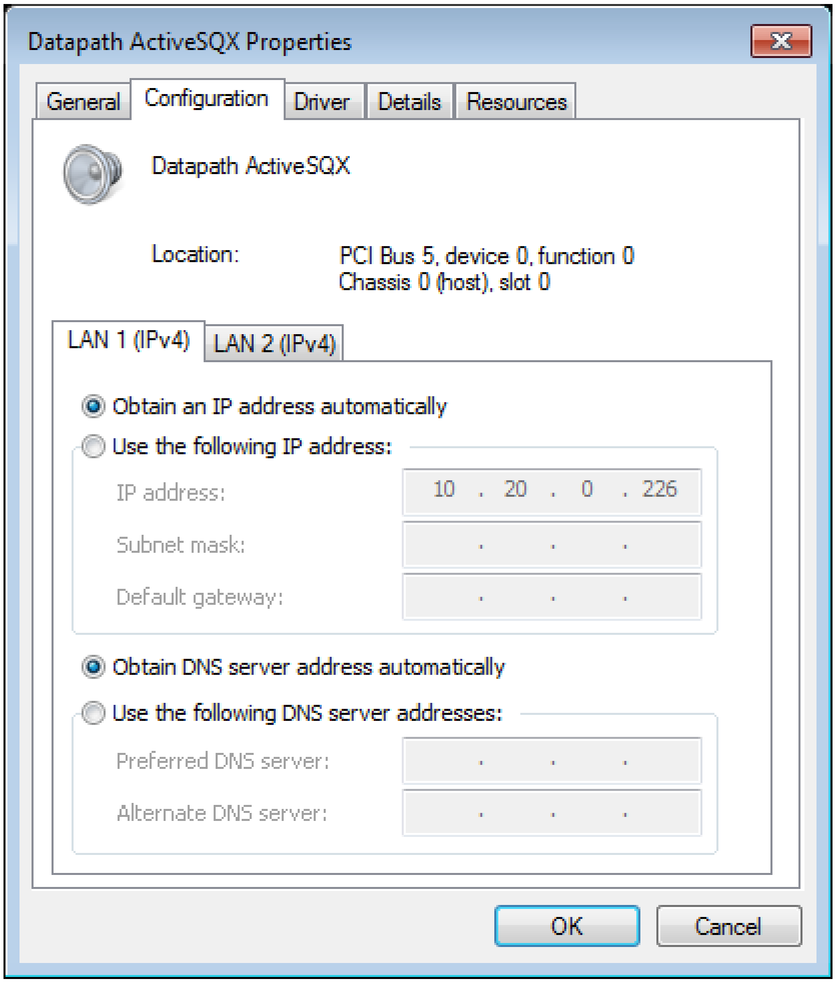
This means the ActiveSQX was not able to establish a connection with the source IP address and port. In some cases it may be that the source has rejected the connection due to invalid username and password (refer to section 3.1.4) or that the source IP has reached its maximum number of supported connections.

3.1.1.1 Check physical connection

Make sure that an Ethernet cable is connected to the ActiveSQX and is on a network which has physical access to the source's IP address.

3.1.1.2 Check IP address

If DHCP is being used make sure that the ActiveSQX has obtained an IP address, check this by opening device manager and double clicking on the ActiveSQX card under “Sound, video and game controllers”. Go to the “Configuration” tab and look for an IP address in the greyed out IP address field as shown in Figure 1. If the “Configuration” tab is not present then the ActiveSQX has failed to establish a connection with the host machine. If this is the case try to shut down and power back on, if the tab still does not appear then either the card is faulty or has been placed in an unsupported host machine.



Datapath ActiveSQX

Location: PCI Bus 5, device 0, function 0
Chassis 0 (host), slot 0

● Obtain an IP address automatically
○ Use the following IP address:
IP address: 10 . 20 . 0 . 226
Subnet mask: . . .
Default gateway: . . .

● Obtain DNS server address automatically
○ Use the following DNS server addresses:
Preferred DNS server: . . .
Alternate DNS server: . . .

Figure 1 ActiveSQX device manager configuration

The IP address field will only update when the dialog is first opened, so if an Ethernet cable is subsequently plugged into the ActiveSQX the dialog must be closed and re-opened for it to refresh. If an IP address does not show up in the dialog then verify that the DHCP server is running correctly. If necessary use a packet sniffer such as Wireshark on another machine connected to the same network to verify the DHCP packets are being transmitted. If the DHCP server is not functioning correctly then try assigning a static IP address instead. However if a static IP address is assigned make sure not to pick an address within the range reserved for DHCP to avoid IP address conflicts.

3.1.1.3 Ping IP addresses

Once the ActiveSQX has an IP address verify that it responds to ping requests from another machine on the network by typing "ping x.x.x.x" using a Command Prompt, replacing x.x.x.x with the IP address of the ActiveSQX. Also verify that the source responds to ping requests to its IP address. Provided the ping requests succeed verify that the port number entered for the source address is correct. Also verify that the stream is accessible from another machine in the same network using a software media player such as VLC.

3.1.2 DNS Error

If a domain name, rather than an IP address, is entered for the IP source and it has failed to resolve then this error will be displayed. Check that the domain name is correct and that it can be accessed using the ping command. Otherwise it may be because the ActiveSQX is unable to connect to the domain name server. The address of 1 or 2 DNS servers may be explicitly specified through the ActiveSQX properties page in device manager if the DNS cannot be established automatically through DHCP.

3.1.3 Invalid path

This error occurs when connection to the IP source was established but the source rejected the path given. Check the path in Wall Control by opening the "IP-Camera Configuration" dialog and selecting the "Camera Models" tab. Select the relevant camera, click on "Modify…" and verify the path is correct.

3.1.4 Not Authorized

This error indicates that the IP source has rejected the connection. This may be due to an invalid username or password. Check the username and password in Wall Control by opening the "IP-Camera Configuration" dialog and selecting the "Cameras" tab. Select the relevant camera, click on "Modify…", select the "UserName" radio button and enter the login credentials.

3.1.5 Stream stopped

This error indicates that no video frames have been decoded in the last second. This may be due to a temporary network glitch such as a cable being unplugged and may recover provided the network connection is re-established before the connection times out. This may also occur if the maximum decoding capacity of the ActiveSQX is exceeded, please refer to the datasheet for the limits.

3.1.6 Connection Terminated

This error will occur in situations where the connection is lost for an extended period of time (i.e. about 30 seconds). This may be due to a number of different reasons:

- A cable being disconnected somewhere between the source and the ActiveSQX

- The source may have stopped streaming if it was powered off for example
- If the source was multicast it may be due to a misconfiguration in the IGMP network settings on the switches (see section 4.3 for more details)

3.1.7 End of stream

This is not actually an error, it simply indicates that the source IP stream has ended. Typically this message will only be displayed if the source was a video file which had a fixed length rather than a never-ending live stream.

3.1.8 Invalid Connection

This error indicates a generic network error has occurred. Further investigation is required as described in section 4.

3.2 Decoding issues

If there are issues decoding the source IP stream then one of the following errors may be displayed:

3.2.1 Unsupported Stream

This error usually indicates that the source IP stream encoding format is not supported or it may also indicate an unsupported transport protocol. Please refer to the ActiveSQX datasheet for a list of supported formats. In some cases it may be possible to address the issue by configuring the source IP stream to a compatible format. If the source's encoding format is unknown it can be checked using a software media player such as VLC.

3.2.2 Invalid Stream

This error indicates that a generic stream error has occurred. Further investigation is required as described in section 4.

3.3 Stuttering and video corruption

In some cases a number of green tinted frames may be seen when first connecting which will then correct itself after some time. In the case of H264 encoded video this is due to not having received an I-Frame from the IP source when the connection was first made. The IP source should send an I-Frame on connection but not all IP sources do so. Unfortunately there is nothing that the ActiveSQX can do about this.

Another common issue with IP streams is to see video stutter and/or decoding artefacts. Since the decoding of frames often relies on having successfully decoded the frames before it, IP streams are very sensitive to any disruption and this often manifests itself in video corruption.

Decoding artefacts are usually caused by network packet loss, to confirm this is the case it is best to try and make the route from the source to the ActiveSQX as short as possible. If physical access to the network is available try to connect the source and the ActiveSQX directly into the same switch and disconnect all other unnecessary ports to see if the problems persist. The switch or cabling may also be faulty, try using several different switches and cables if possible. If that still fails the source could be faulty, try using different IP sources. If physical access to the network is not available then try measuring packet loss using the logging techniques described in section 4.

Stuttering video may also be caused by network packet loss but is usually also associated with decoding artefacts. If video stutter is present without artefacts then it may be due to packets taking different amounts of time to be transmitted from the source IP stream to the ActiveSQX. The ActiveSQX will by default cache packets for 200 milliseconds before decoding them, this will enable smooth playback even though packets are not received at a smooth rate, provided the difference in transit time between the slowest and fastest packet does not exceed 200 ms. This value is configurable in Wall Control by opening the "IP-Camera Configuration" dialog, selecting the camera, clicking "Modify…", then "Advanced" and changing the Caching value. Try increasing the value until the stuttering is eliminated. However be aware that increasing this value will add a delay between the live source and the displayed video. If the problem cannot be solved by changing the Caching value then the problem may lie with the IP source. Try decoding the same source with a software media player such as VLC to see if the same problem is exhibited or not.

4 Advanced troubleshooting

In cases where the problem cannot be identified from section 3 it will be necessary to make use of the built-in logging functionality of the ActiveSQX. By default logging is disabled for performance, durability and security reasons but is invaluable for debugging complicated issues.

Logging should be enabled and then the issue should be reproduced by opening the problematic IP stream (or performing whatever steps are necessary). The log files can then be analyzed to determine the cause of the problem or sent back to Datapath for further analysis if necessary.

4.1 Enabling logging

Logging can be enabled using the cmd183.exe command line tool from a Command Prompt. As a starting point SQX logging warnings and errors with GStreamer INFO level logging and pipeline dump should be enabled using the following command line:

cmd183.exe setlog file –s 2 –g 4 –p

These logs will not contain any sensitive information and are fairly light. For more details on the different logging types and how to use the cmd183 tool see section 5.1.

4.2 Retrieving logs

Logs are retrieved using the cmd183.exe command line tool as well. First create a folder on the host machine where the logs should be stored. Then run "cmd183.exe getlogs <path>" where <path> is substituted for the location of the folder created. Once the logs are successfully retrieved they will be cleared off the ActiveSQX card(s). All log files have a date and time appended to the filename as well as an ID such that each client window has its own unique log file. The contents of the folder can then be zipped and sent back to Datapath for further analysis. Depending on the outcome Datapath may request further logging with different options to be done in order to narrow-down the problem.

4.3 Detecting packet loss and jitter

If the network appears to be suffering from packet loss and/or jitter because the video is displaying decoding artefacts then enable the following logging:

```
cmd183.exe setlog file –g rtpsource:5
```

Open the IP stream and leave it running for a minute or so then retrieve the logs using the getlogs command. In sqx-client-child_xxx.log look for lines that look like this:

```
rtp_source_get_new_rb: fraction 0, lost 1, extseq 120132, jitter 98
```

fraction indicates the number of packets lost in the range of 0 to 256, to convert to a percentage divide by 2.56. **lost** indicates the number of packets lost since the last report. Unfortunately even small amounts of packet loss can result in large decoding artefacts. Take for example an H264 encoded video transmitted using RTP packets at a bit rate of 8 Mbit/s at 30 frames per second with a GOP size of 30. Many IP sources will encode entire frames into a single H264 NAL unit, so each frame will thus be on average roughly 34 KB. The maximum transmission size of a packet is typically around 1.5 KB, so a frame must be split into around 23 RTP packets. Due to the way the RTP protocol works, if any one of those 23 packets goes missing all of them must be discarded. So a single packet lost can in fact result in 23 packets being discarded in this example. As soon as one frame goes missing, all subsequent frames will suffer from decoding artefacts until the next I-frame arrives which could be as much as 29 frames in this example. A mere 0.1% packet loss could result in up to 66% frames being corrupt (0.1 * 23 * 29). So in order to achieve smooth and artefact-free video playback it is important to minimize packet loss as much as possible, ideally eliminate it completely.

jitter is the variation in the delay of received packets in the stream in clock rate units. It is measured by comparing the interval when RTP packets were sent to the interval at which they were received. For instance, if packet #1 and packet #2 leave 50 milliseconds apart and arrive 60 milliseconds apart, then the jitter is 10 milliseconds. The lower the jitter value the better, high jitter values indicate packets are not arriving at regular intervals and more caching is needed to achieve smooth playback. Jitter is generally less of a problem compared to packet loss since it can be compensated for by the ActiveSQX by increasing the caching.

4.4 IGMP Multicast issues

If a multicast IP stream cuts out after a certain amount of time, anywhere from a few minutes to hours or even days, then it may be due to a configuration issue with the switches being used to transmit the multicast packets. If that is the case a "Connection Terminated" error will appear in the window, we can verify that the RTP packets are no longer being received by enabling the following logging:

cmd183.exe setlog file –s 2 –g rtpsource:5 –n igmp

Open the IP stream and leave it running until the stream drops out, then retrieve the logs using the getlogs command. In sqx-client-child_xxx.log look for the lines "No frames transferred for more than 1 second" and "Treating end of stream as timeout", in between those the number of RTP packets being received can be seen. If that number has not increased in between the last frame being decoded and the timeout then it means that no new RTP packets are coming in and could point to a switch filtering the packets out.

Assuming the packets have stopped, we can look at the IGMP packets to see if it is behaving as expected. Open sqx-client_xxx.pcap with Wireshark and look for "Membership Query, general" packets coming from the switch. Make sure that every query is responded by a "Membership Report" within the Max Resp Time indicated by the Query packet for the multicast address connected to. If another device responds with a report before the ActiveSQX then the ActiveSQX will not send a response as well (this is part of the IGMP protocol standard). In some cases the ActiveSQX may receive a report from another device that it shouldn't, this indicates in issue with the network configuration.

5 Appendices

5.1 Logging types

Log files can either be stored on the ActiveSQX or sent live to the host machine as they are generated and viewed using DebugView or a kernel debugger. Writing to files on the ActiveSQX tends to be faster and doesn't interfere as much with the decoding process but the ActiveSQX has limited space for logging (about 1.4 GB). Conversely viewing the logs in DebugView isn't limited by the ActiveSQX's disk space but can severely impact performance if very verbose logging is enabled.

There are various different types of logging that can be enabled:

- SQX: log messages originating from Datapath's proprietary code.
- GStreamer: log messages originating from open source GStreamer code.
- GStreamer pipeline: graphical representation of the GStreamer pipeline.
- Network packet capture: raw network packet captures that can be viewed in Wireshark.

Depending on the nature of the problem any number of these logs can be enabled at once bearing in mind the limitations of disk space and performance impact. The best approach is to start with a fairly light level of logging and look for errors and warnings within them. Then as the source of the problem becomes more apparent increased logging levels can be activated.

5.1.1 SQX Logging

SQX logging comes in 4 levels: 0 to 3. 0 represents no logging, 1 is errors only, 2 is errors and warnings and 3 is all logging. Log level 2 is a good level to start with as it should not fill the disk too quickly. Example: *cmd183.exe setlog file –s 2*

5.1.2 GStreamer logging

GStreamer logging is more complicated, it includes 10 log levels 0 to 9. In addition to the log level, GStreamer has a number of different debug categories for different components within the GStreamer architecture. It is possible for example to enable debug level 9 specifically for the rtpsource category while leaving all other categories disabled. For more information refer to the GST_DEBUG variable under http://gstreamer.freedesktop.org/data/doc/gstreamer/head/gstreamer/html/gst-running.html. Level 4 across all categories is generally a good starting point as it strikes a balance between giving enough information and not filling up the disk space too quickly. Example: *cmd183.exe setlog file –g 4* or *cmd183.exe setlog file –g rtpsource:5*

5.1.3 Pipeline logging

GStreamer also has the ability to output the pipeline to a .dot file which once processed by graphviz gives a graphical representation of the pipeline which gives a good indication of what encoding formats and transport protocols are being used by a given IP source. Example: *cmd183.exe setlog file -p*

5.1.4 Packet logging

If the problem cannot be diagnosed with the log files alone then it may be necessary to capture raw network packets and analyze them in Wireshark. For intermittent problems it may also give a convenient way to reproduce rarely occurring issues. However it should be noted that if all packets are captured it may fill up the disk space very quickly. To help alleviate this issue and provide a level of privacy to customers it is possible to filter packets captured based on given criteria. Tcpdump is used for the packet capture and the filter syntax is described in detail here http://www.tcpdump.org/manpages/pcap-filter.7.html. Example: *cmd183.exe setlog file –n "src 10.20.0.1"* or *cmd183.exe setlog file –n " "* for no filtering.

Promiscuous mode is disabled to avoid capturing general network traffic not destined to the ActiveSQX. However since the packet capture operates at a low level it may capture potentially sensitive information depending on what information is being sent across the network if not explicitly filtered out. It is also possible to replay video from the packet captures and customers should be made aware of this before asking them to send packet captures back to Datapath. For these reasons packet capture should generally be used as a last resort for debugging issues.