Understanding and optimizing the latency of Aquilon Presentation Systems

Analog Way Worldwide
Europe & Africa: +33 (0)1 81 89 08 60
Middle East: +971 4 25 33 818
The Americas: +1 212 269 1902
Asia Pacific: +852 6882 7140
www.analogway.com

Introduction

Live presentation systems such as Aquilon series are just one link in a chain where each element adds its own latency (cameras, led wall processors, extenders…). Therefore, latency is one of the key characteristics for these systems. The lower it is, the better.

Note that the Aquilon series has been optimized to offer extremely low latency, in most cases between 1 and 2 frames, which is ideal for demanding live events.

However, the latency of a complex system is not easy to estimate as it may vary depending on several parameters:

- The hardware architecture
- The type of content to process
- The type of destination: Screen, Auxiliary Screen or Multiviewers.
- The way algorithms are implemented

This document aims at offering a better understanding of:

- The Aquilon architecture,
- What generates the latency,
- How to compute the latency of the Aquilon series,
- How to configure your Aquilon system and your setup.

Analog Way Worldwide
Europe & Africa: +33 (0)1 81 89 08 60
Middle East: +971 4 25 33 818
The Americas: +1 212 269 1902
Asia Pacific: +852 6882 7140
www.analogway.com

Contents

Analog Way Worldwide
Europe & Africa: +33 (0)1 81 89 08 60
Middle East: +971 4 25 33 818
The Americas: +1 212 269 1902
Asia Pacific: +852 6882 7140
www.analogway.com

Figures

1. Hardware architecture and latency

The hardware architecture of the Aquilon series is highly modular so it fits the needs and requirements of the customers in terms of inputs, outputs, layers and images capacities. This is clearly understandable when configuring an Aquilon C or C+. You can select the number and type of I/O cards, the number of VPUs (Video Processing Unit) and IPUs (Image Processing Unit)

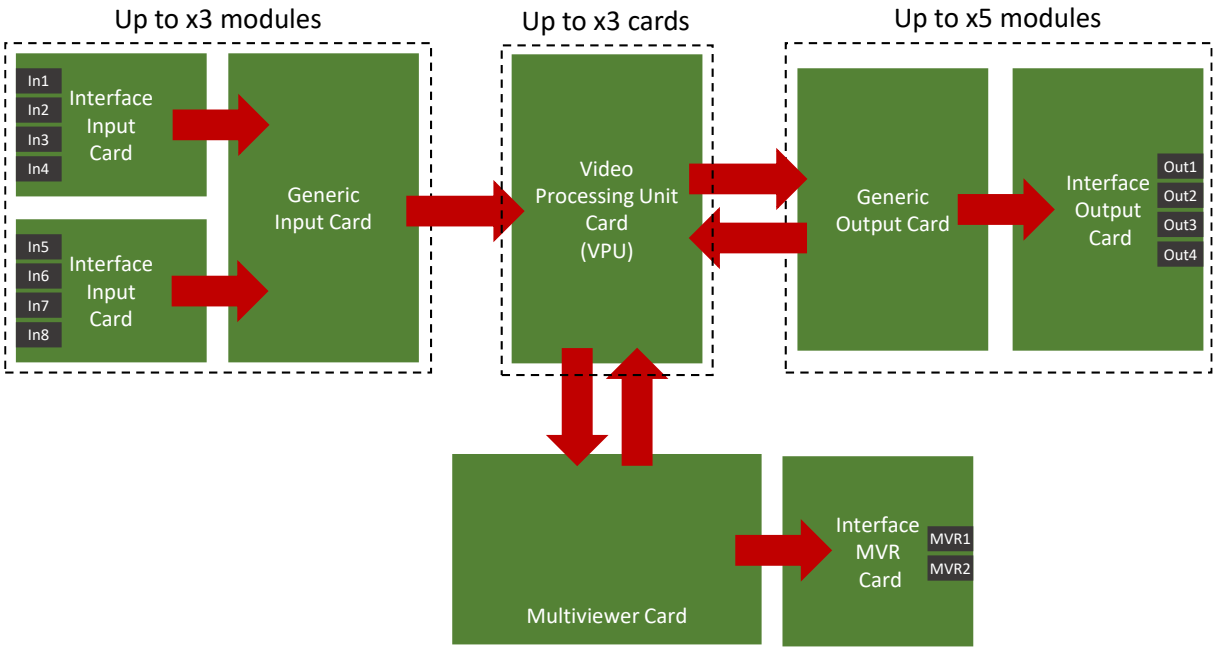

Figure 1: Simplified block diagram of the Aquilon architecture

Each of these board types contains processing algorithm. Some of them may add delay to the input signal as they require one whole frame or more to work on. For Aquilon series, the latency is mainly created in:

- Generic Input Cards
- VPUs
- Generic Output Cards
- Multiviewer Card

Note: a system with a monolithic architecture may have a slightly lower latency as all the processing is achieved in the same component and memory. However, this kind of architecture cannot be envisioned for large scale systems as it may induce high production costs.

2. Algorithms, processing and latency

This chapter deals with the different algorithms and treatments that may induce latency in the Aquilon architecture due to their implementation.

2.1. Deinterlacing

The deinterlacer transforms an interlaced video signal made of alternated frames with even and odd lines into a progressive signal, with each frame containing all the lines.

A good deinterlacer is characterized by its capacity to reduce the artefacts of image reconstruction such as comb effect. Unfortunately, the better the image quality, the worse the latency.
For live presentation switchers, the main difficulty is to find the best compromise between the image reconstruction quality and the latency.

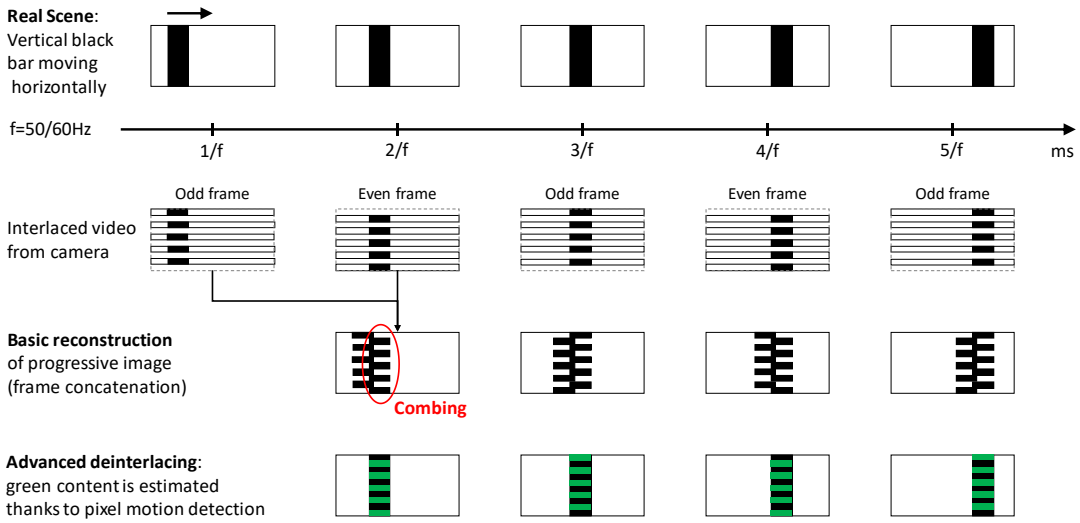

Figure 2: Basic explanation of deinterlacing

The deinterlacer of the Aquilon series, based on pixel motion detection, adds **no latency**, still offering unrivaled results in its category.

Note: the deinterlacer of LiveCore series has one standard mode with no latency and one optimized mode adding a 1-frame delay. Aquilon now always works in optimized mode without adding latency.

2.2. Frame Rate Conversion (FRC)

The Aquilon series architecture requires the system to work with a single Internal Rate. Therefore:

- Any input with a rate different or out of phase from the Internal Rate needs to be rate-converted/synchronized with the Internal Rate.
- Any output with a rate different from the Internal Rate also needs to be converted to the set rate.

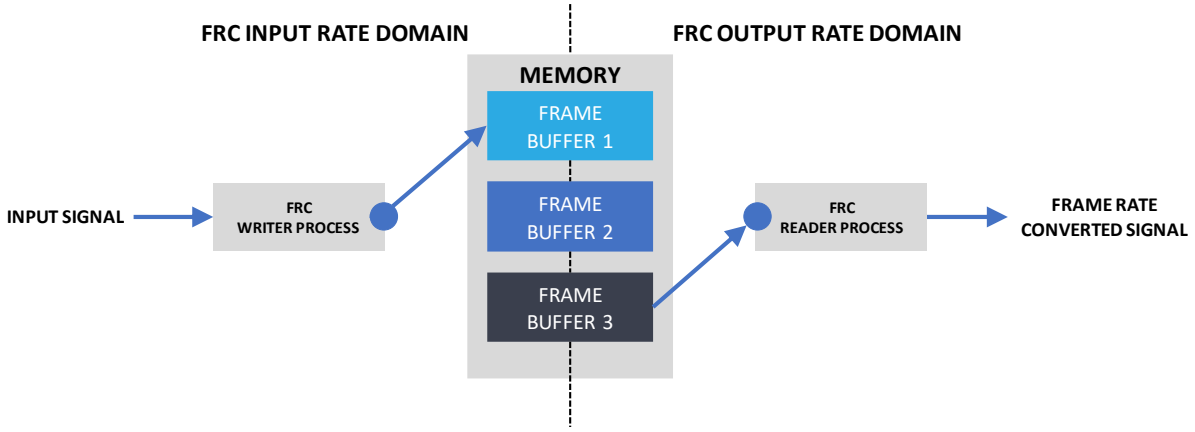

Figure 3: Basic approach of a frame rate converter

The FRC of the Aquilon Series is based on frame insertion and deletion. A pool of buffers is used to write and read the input frames:

- A buffer can only be read if it is ready. Once read, it is freed.
- A buffer can be written if it is free. Once written it is indicated as ready.

Consequently:

- If the input rate of the FRC is higher than its output rate (meaning writing images is faster than reading), a frame may be regularly lost as no free buffer is available.
- If the input rate of the FRC is slower than its output rate (meaning writing images is slower than reading), a frame may be regularly duplicated as no ready buffer is available. The reader process must read the previous frame waiting for the next one

As soon as an FRC is running, a time-varying delay is added and may reach **up to one frame**.
To disable the FRC at the input level, only use inputs frame-locked with the Internal Rate

To disable the FRC at the output level, only use output formats clocked the Internal Rate

Note: Inputs or outputs enabling an FRC may have their content stutter as there may be frame insertion or removal.

2.3.Scaling

The scaling algorithm used by the Aquilon series requires to buffer a frame to be able to process it in a layer. Therefore, a **1-frame delay** is systematically added to any source used in a layer whether it is a mixer layer or a split layer.

2.4.Grouping/Ungrouping

Grouping inputs is useful when working with a content delivered on several inputs.
Some usual cases are a 4K source content sent over Quad 3G-SDI or an extra-wide content delivered by a media server over several DP outputs. Grouping allows to handle the content as if it was a single physical input.
Ungrouping outputs allows to split a screen on several outputs. For example, a 4K screen is split in four 2K outputs.

Both grouping and ungrouping have the advantage to optimize the processing pipeline and the resources in term of layers.

When using the input/output groups, a **1-frame delay** is systematically added to ensure all the data are ready when grouping/ungrouping the stream. The reason is the Group Reader process is much faster than the writer processes.
When each input of the group is fully written in the memory at its appropriate location, the memory buffer is considered as ready and can be read as a single "input".

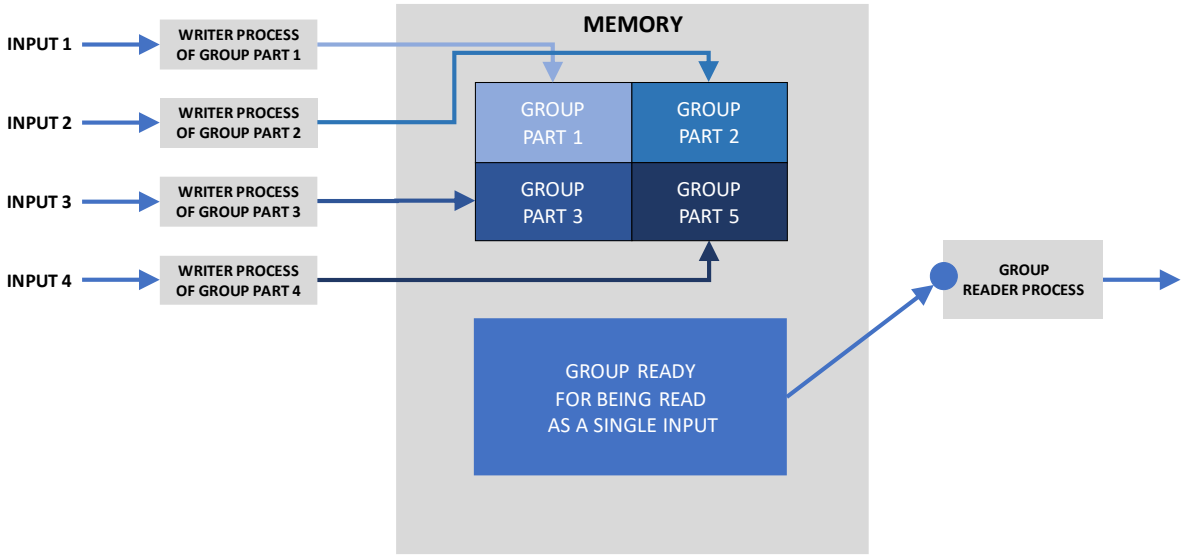

Figure 4: 2x2 Input grouping concept

2.5.Output rotation

When rotating an output, it requires a large amount of memory with non-consecutive accesses. Implementing a zero-latency rotation algorithm would be over-expensive and unreasonable.
The rotation algorithm implemented in the Aquilon series adds a **1-frame delay**.

Note: as a comparison, LiveCore series' output rotation generates a 2-frame delay.

2.6.DPH104 formatting

To be able to slice the 4K content it receives into four HD signals, the DPH104 requires a specific data format. This formatting operation is achieved by the Aquilon system. It adds a **1-frame delay** in the data flow.

2.7.Synchronization buffering

In a modular system, the data composing a video content may be generated by different cards with different timings. To ensure that all the data are consistent and ready for mixing, a buffering stage may be required, inducing a **1-frame delay**.

2.8.Conclusion

There are many processing treatments that adds latency. Thankfully, many of them are not used in usual applications and even if needed, most of them can be performed simultaneously. Please remember that **the latency of Aquilon series is between 1 and 2 frames in most cases**.

3. Computing the latency of Aquilon Systems

Depending on the type of sources and outputs, the data path is not the same. Therefore, this chapter details all these cases giving flow charts. The global delay is computed by adding the latency generated at each step of the path.

3.1.Input displayed in a Screen

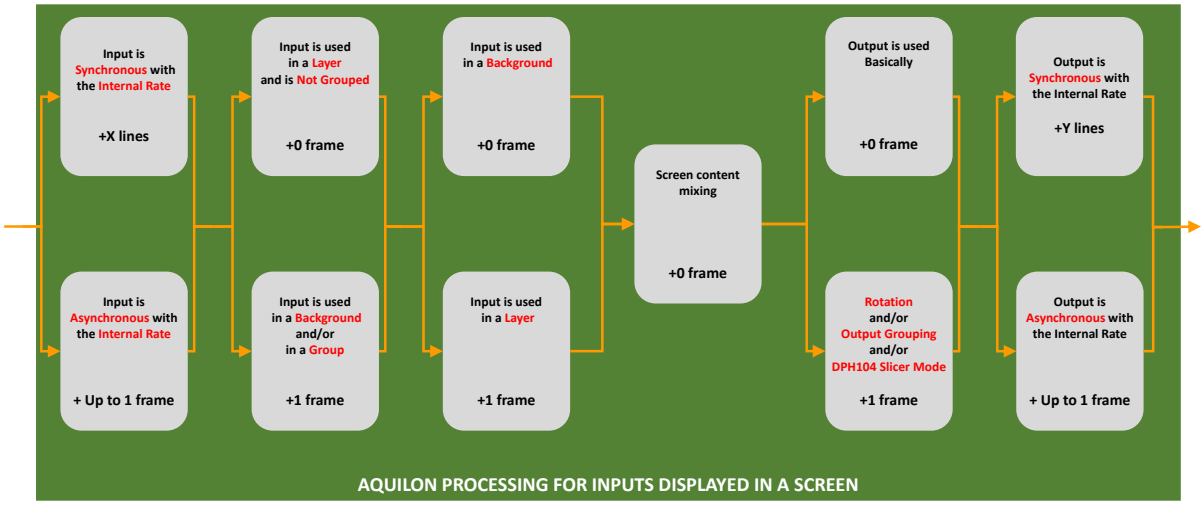

Figure 5: Latency map of an input displayed in a Screen

3.2.Input displayed in an Auxiliary Screen

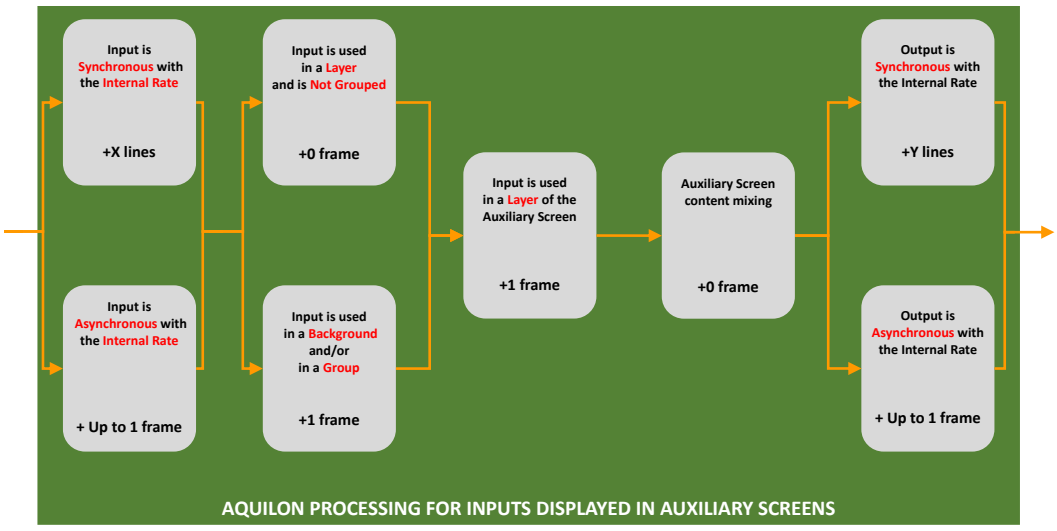

Figure 6: Latency map of an input displayed in an Auxiliary Screen

3.3.Input displayed in a Multiviewer

All the contents reaching the Multiviewer processing are buffered for synchronization.

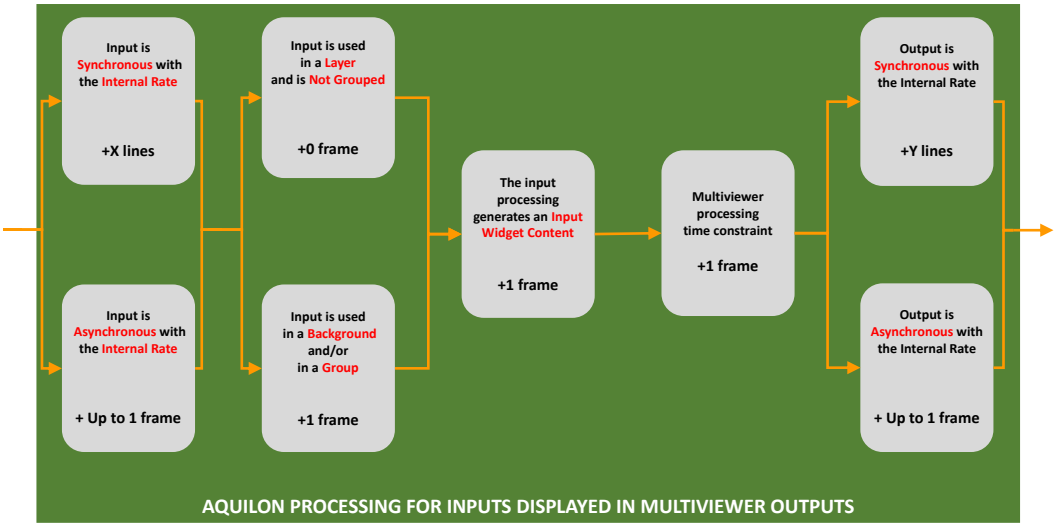

Figure 7: Latency map of an input displayed in a Multiviewer

3.4.Program/Preview displayed in a Multiviewer

All the contents reaching the Multiviewer processing are buffered for synchronization.

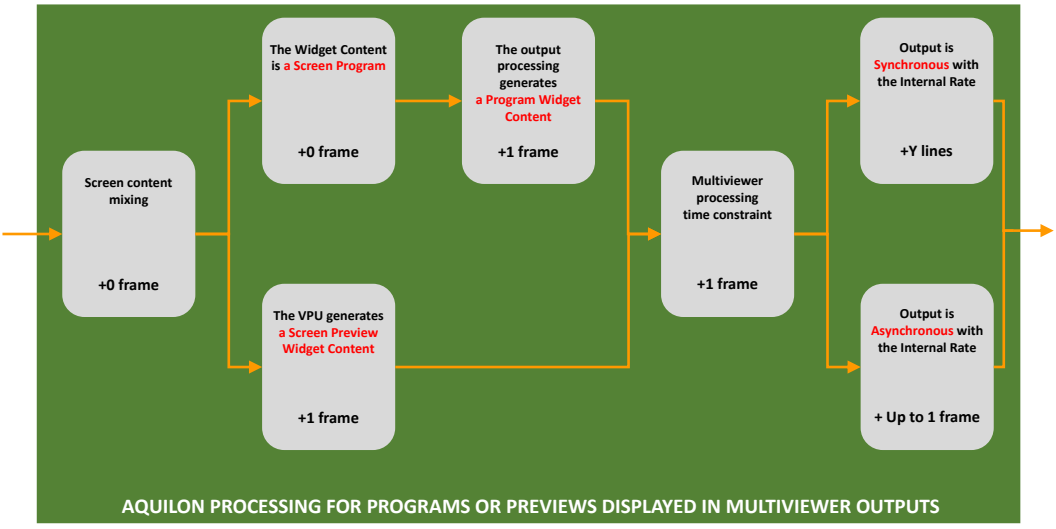

Figure 8: Latency map of a Program/Preview content displayed in a Multiviewer

3.5.Auxiliary Screen displayed in a Multiviewer

All the contents reaching the Multiviewer processing are buffered for synchronization.

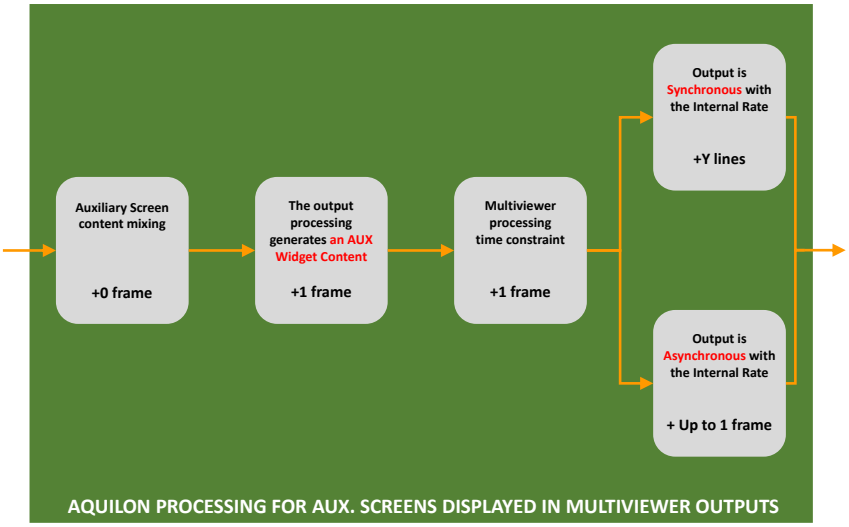

Figure 9: Latency map of an Auxiliary Screen displayed in Multiviewer

4. Optimizing the latency of a Aquilon system

4.1.Input Frame Rate Conversion

To avoid the latency due to the frame rate conversion, only use live inputs that are synchronous with the Internal Rate. To do so, the Internal Rate must be configured as framelocked to an external source. This reference source can be:

- A black burst or tri-level sync signal plugged to the Framelock input. This signal is also used to synchronize the other sources and display systems.
- A live input for which the latency must be as low as possible, typically a camera. Ideally, all the other input signals are synchronized with this reference source (genlocked for example).

This can be easily achieved in the Preconfig>System page of the Web RCS:

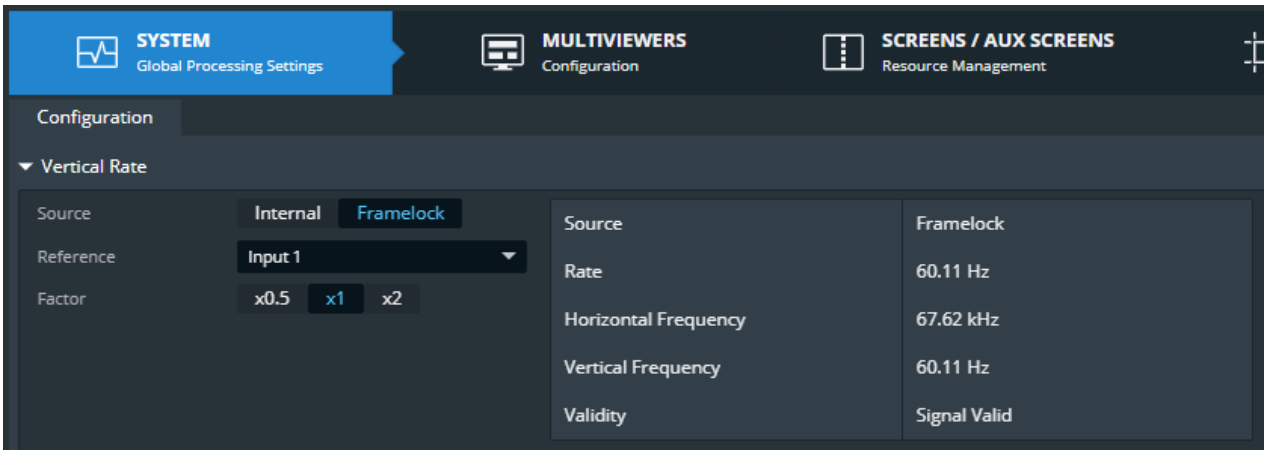

Figure 10: Example of the Internal Rate framelocked to input 1

4.2.Output Frame rate Conversion

The frequency of the outputs must be identical to the Internal Rate so there is no output frame rate conversion. Therefore, the output rate can be selected either as following the Internal Rate or as a custom rate identical to the Internal Rate.

Note:

- By default, the rate of all outputs is the Internal Rate.
- he outputs with the same rate value have the same phase so they can be used for Hard/Soft-edge purpose.

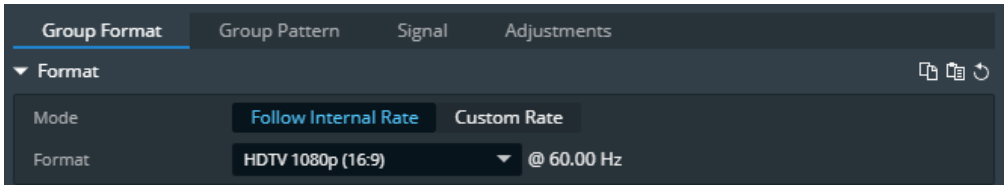

Figure 11: Example of an output with its rate following the Internal Rate

5. Application

5.1.Setup

5.1.1. Diagram

This setup includes different types of sources and displays so the latency of the switcher is evaluated in different cases.

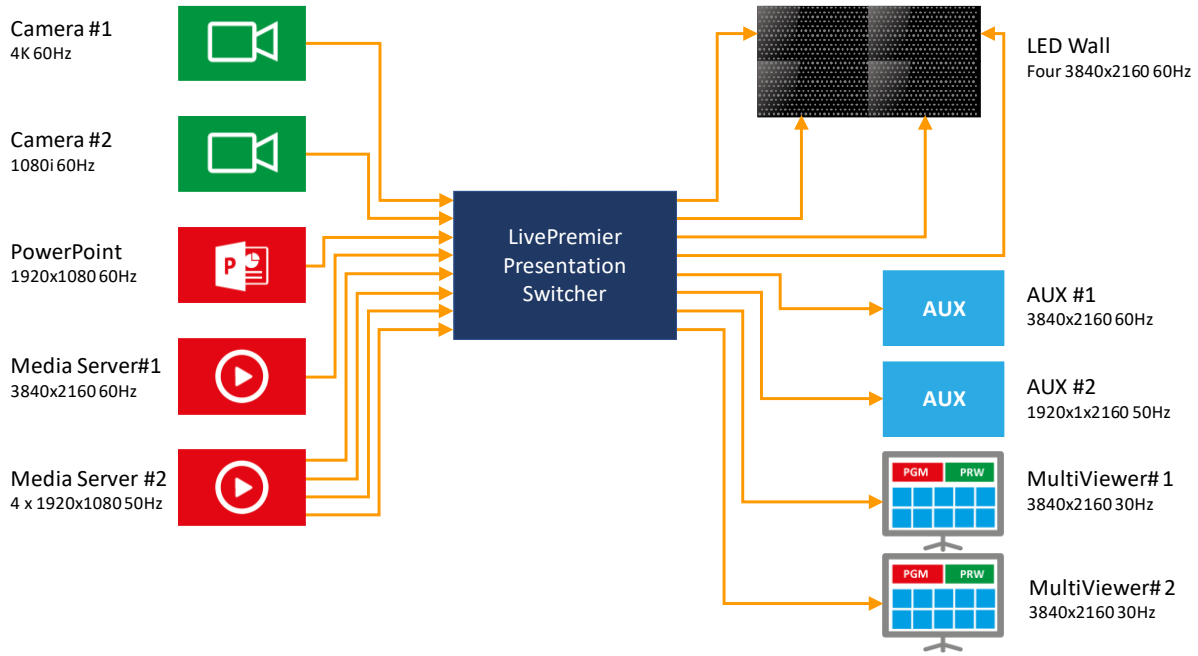

Figure 12: Setup Diagram

Sources:

- **Camera #1** delivers a progressive 4K 60Hz signal
- **Camera #2** delivers an interlaced 1080i 60Hz signal
- **PPT PC** is a computer with a PowerPoint presentation delivering a 1920x1080 60Hz signal
- **Media Server #1** delivers a video over one 3840x2160 60Hz computer signal used for video content
- **Media Server #2** delivers a video split over four 1920x1080 60Hz computer signals used for background content

Displays:

- The **Main Display** is an LED wall fed with four 1920x1080 60Hz signals
- The **Auxiliary Screen #1** is a 3840x21600Hz display.
- The **Auxiliary Screen #2** is a 1920x1080 display only supporting 50Hz.
- The **Multiviewers #1 and #2** are two 3840x2160@30Hz displays. Two multiviewers at 30Hz are chosen to optimize the number and size of widgets:
    - Multiviewer #1 is used to monitor the Program and Preview of the Main Display
    - Multiviewer#2 is used to monitor the inputs and the auxiliary screens.

5.1.2. Configuration

In this use case, all the sources are at 60Hz and so are most of the displays. Therefore, the internal rate of the Aquilon presentation switcher must be 60Hz. As there are two camera sources that are critical from the latency point of view, the best choices for the internal rate can be:

- To **use one of the two cameras as a reference for the framelock** and to ensure the other camera is synchronized.
- Or **to use an external sync generator as a reference** for both the cameras and the internal rate, plugging it to the framelock input of the switcher.

The output formats are all generated using the internal rate of the unit, except for AUX#2 that is 50Hz clocked.

5.2.Latency computing

Note:

- In this section, the latency of the Aquilon switcher is computed following the red path in the diagrams.
- The latency of the cameras and displays is not mentioned here and must be added to get the overall latency of the setup.

5.2.1. Main Display

5.2.1.1. Latency for Camera #1 and #2

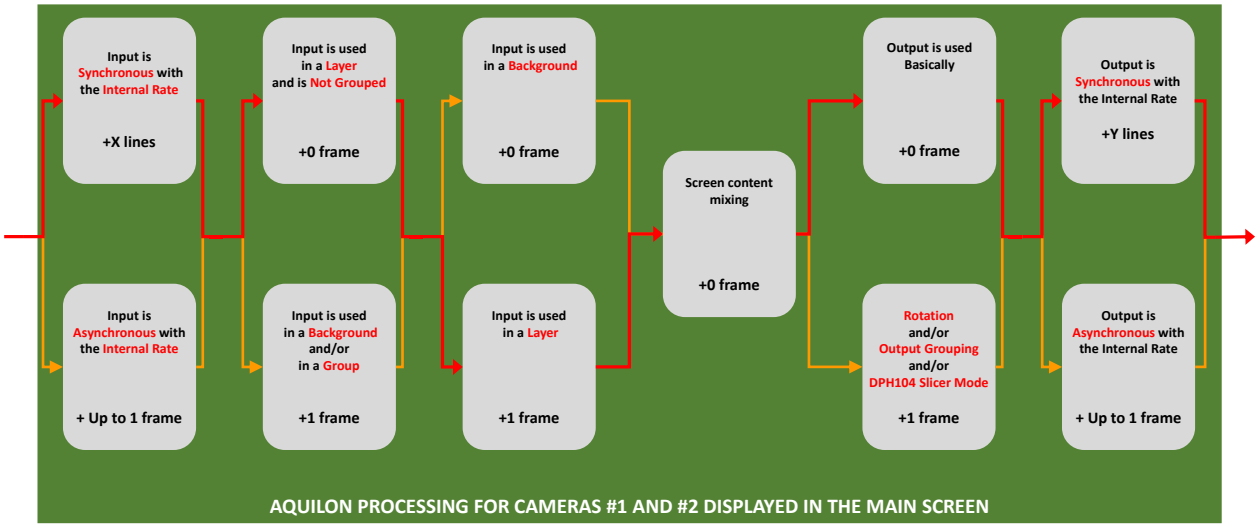

Figure 13: Data flow of Cameras #1 and #2 displayed in Main Screen

Latency = 1 Frame + (X+Y) Lines

5.2.1.1. Latency for PPT PC and Media Server #1

As the other sources are not synchronized the Frame Rate Conversion is activated.

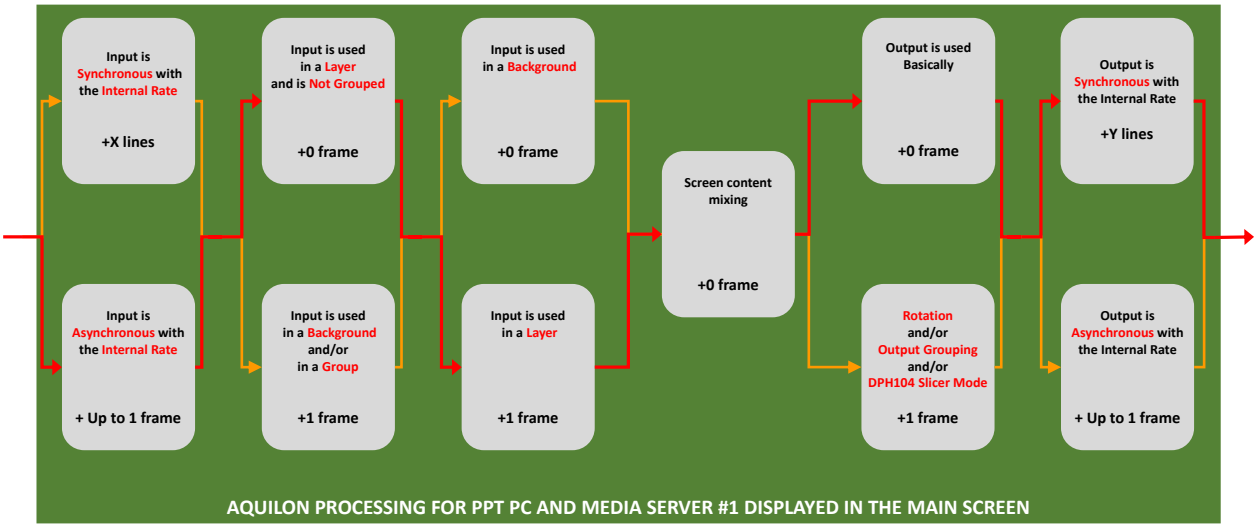

Figure 14: Data flow of PPT PC and Media Server #1 displayed in Main Screen

1 Frame + Y Lines < Latency < 2 Frames + Y Lines

5.2.1.1. Latency for Media Server #2

As Media Server #2 is not synchronized with the Internal rate and is used as a background, the path in the latency map is:

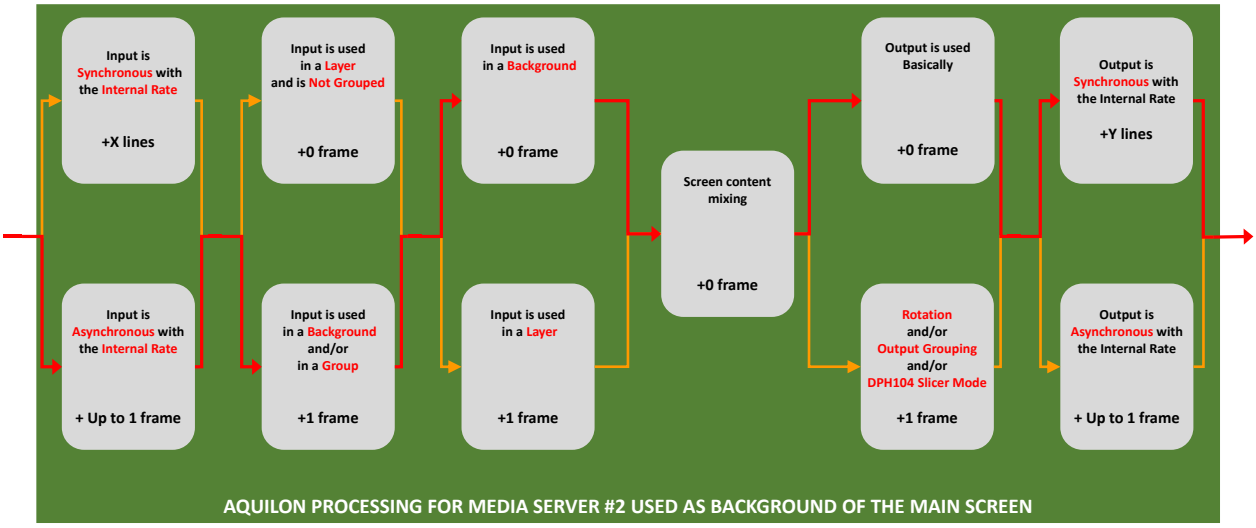

Figure 15: Data flow of Media Server #2 displayed in Main Screen

1 Frame + Y Lines < Latency < 2 Frames + Y Lines

5.2.1.2. CONCLUSION

The latency for the Main Screen never exceeds 2 frames and a few lines.

5.2.1. Latency of Auxiliary Screen #1

5.2.1.1. Latency for Camera #1 and #2

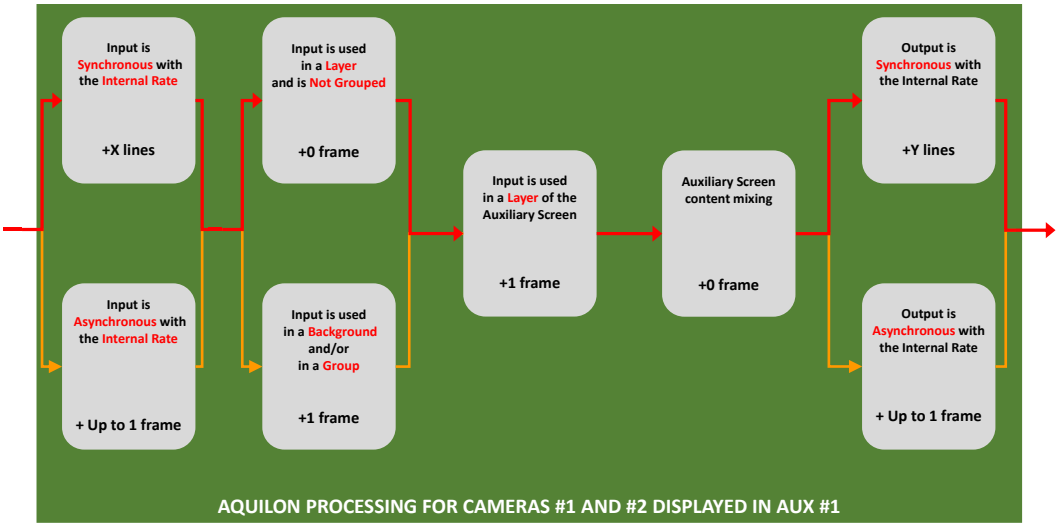

Figure 16: Data flow of Cameras #1 and #2 displayed in Auxiliary Screen #1

Latency = 1 Frame + (X+Y) Lines

5.2.1.2. Latency for PPT PC and Media Server #1

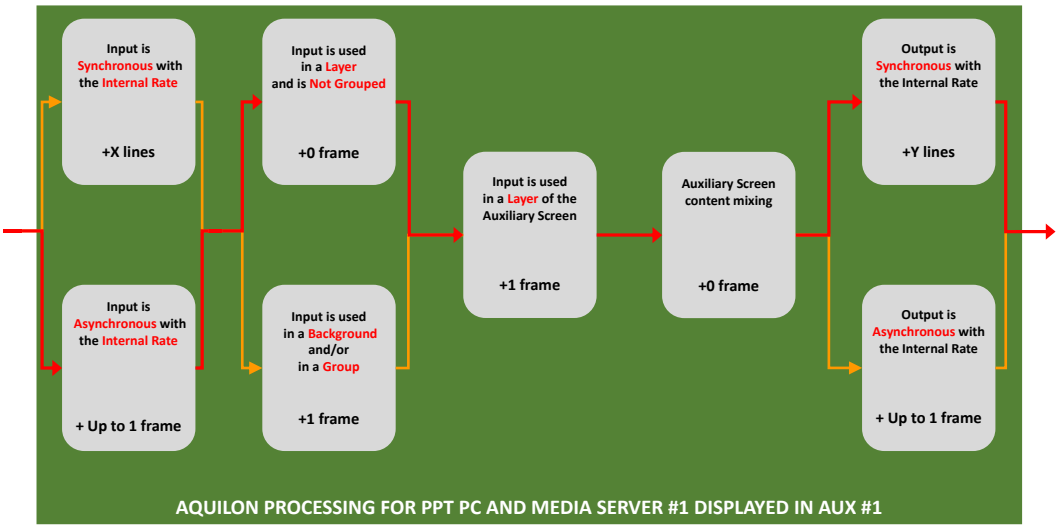

Figure 17: Data flow of PPT PC and Media Server #1 displayed in Auxiliary Screen #1

1 Frame + Y Lines < Latency < 2 Frames + Y Lines

5.2.1.3. Latency for Media Server #2

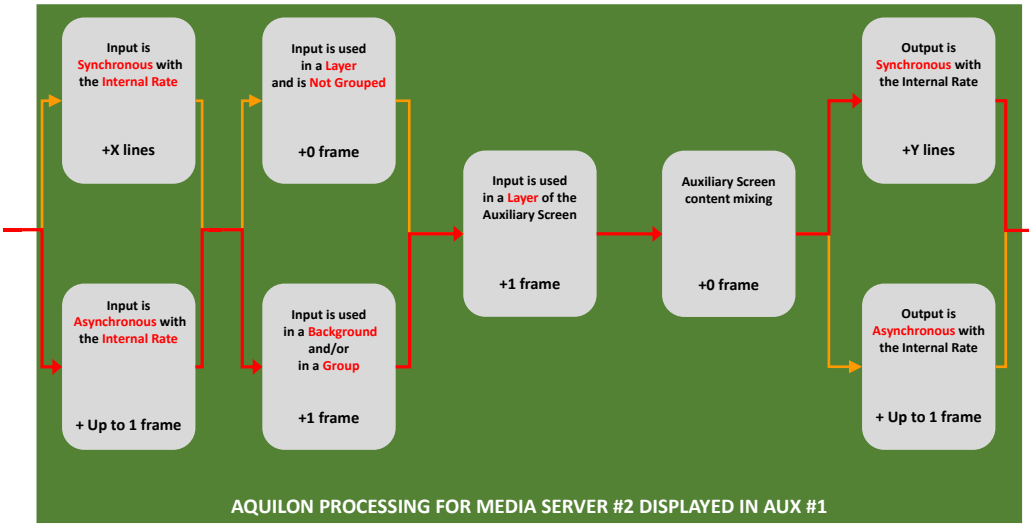

Figure 18: Data flow of Media Server #2 displayed in Auxiliary Screen #1

2 Frame + Y Lines < Latency < 3 Frames + Y Lines

5.2.2. Latency of Auxiliary Screen #2

5.2.2.1. Latency for Camera #1 and #2

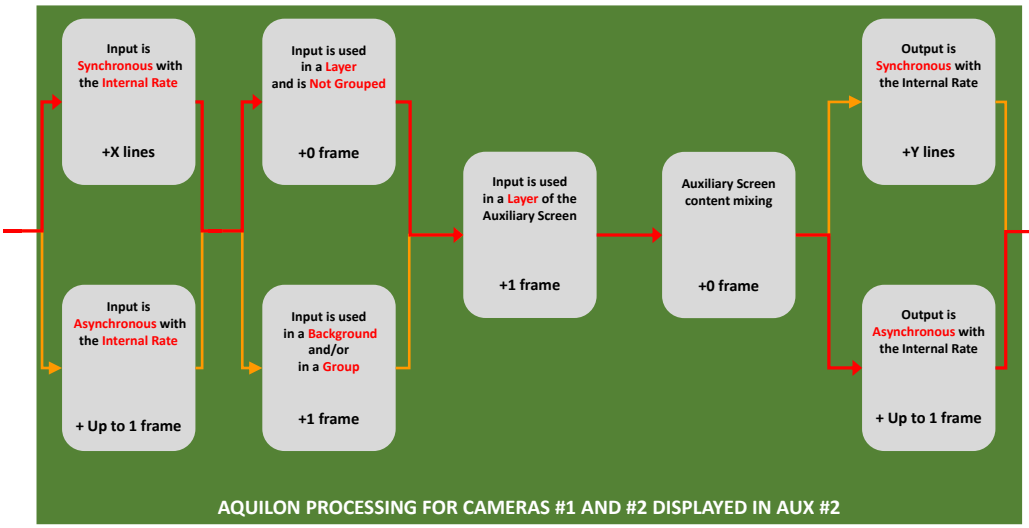

Figure 19: Data flow of Cameras #1 and #2 displayed in Auxiliary Screen #2

1 Frame + X Lines < Latency < 2 Frames + X Lines

5.2.2.2. Latency for PPT PC and Media Server #1

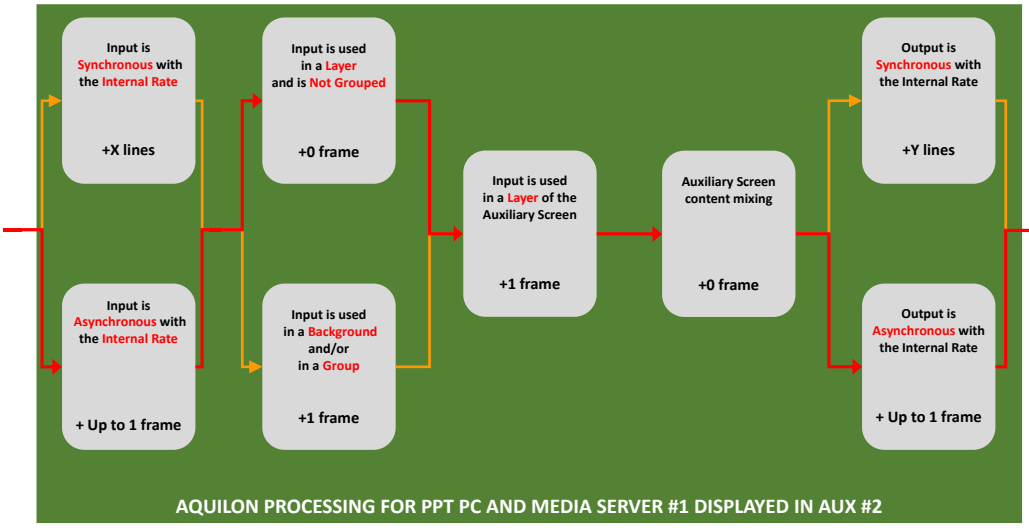

Figure 20: Data flow of PPT PC and Media Server #1 displayed in Auxiliary Screen #2

1 Frame < Latency < 3 Frames

5.2.2.3. Latency for Media Server #2

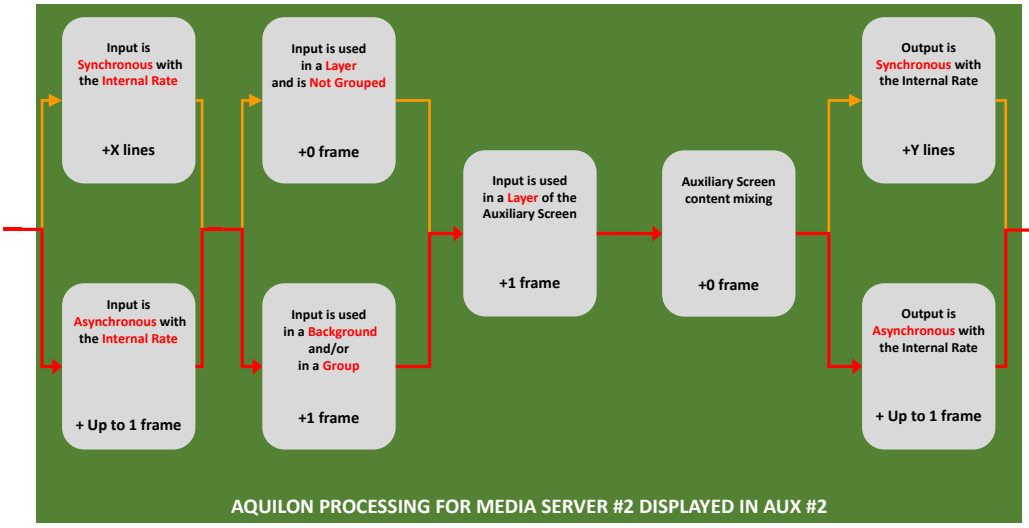

Figure 21: Data flow of Media Server #2 displayed in Auxiliary Screen #2

2 Frames < Latency < 4 Frames

5.2.1. Latency of Multiviewer #1

5.2.1.1. Latency for Cameras #1 and #2

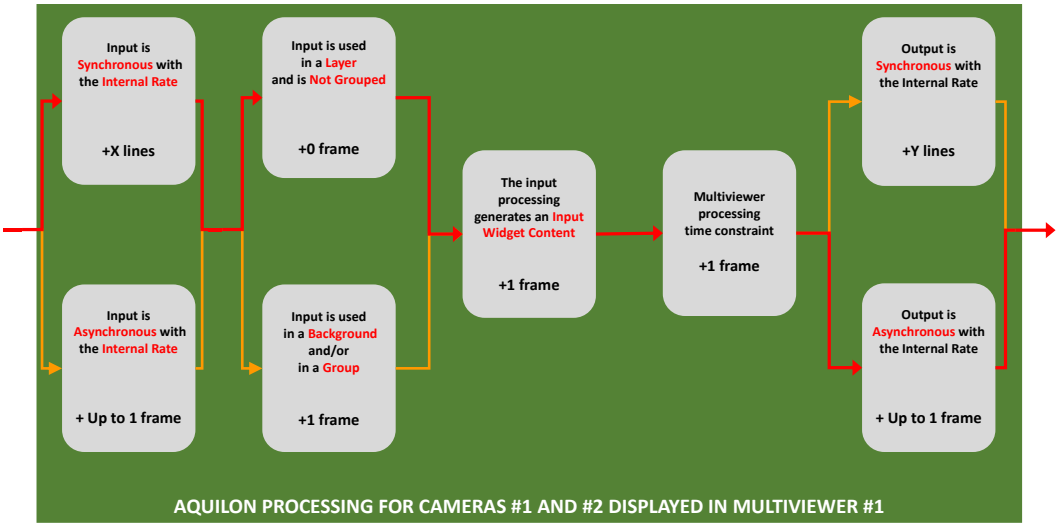

Figure 22: Data flow of Cameras #1 and #2 displayed in Multiviewer #1

2 Frames + X Lines < Latency < 3 Frames

5.2.1.2. Latency for PPT PC and Media Server #1

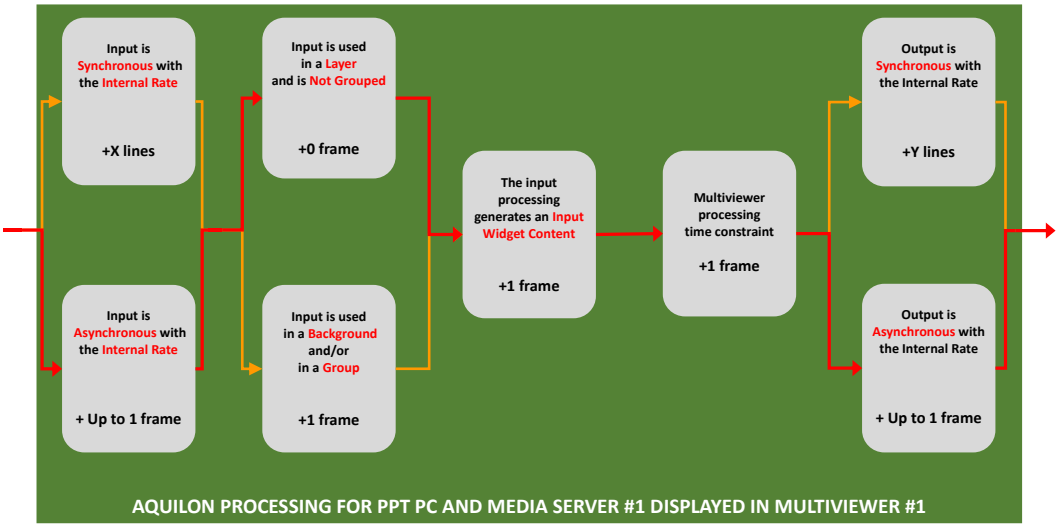

Figure 23: Data flow of PPT PC and Media Server #1 displayed in Multiviewer #1

2 Frames < Latency < 4 Frames

5.2.1.3. Latency for Media Server #2

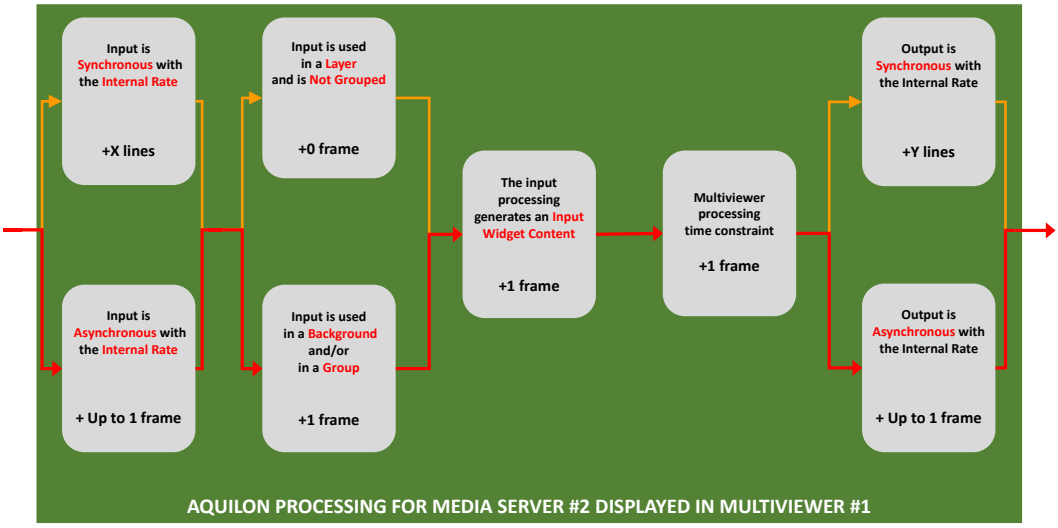

Figure 24: Data flow of Media Server #2 displayed in Multiviewer #1

3 Frames < Latency < 5 Frames

5.2.2. Latency of Multiviewer #2

5.2.2.1. Latency of Auxiliary Screen #1

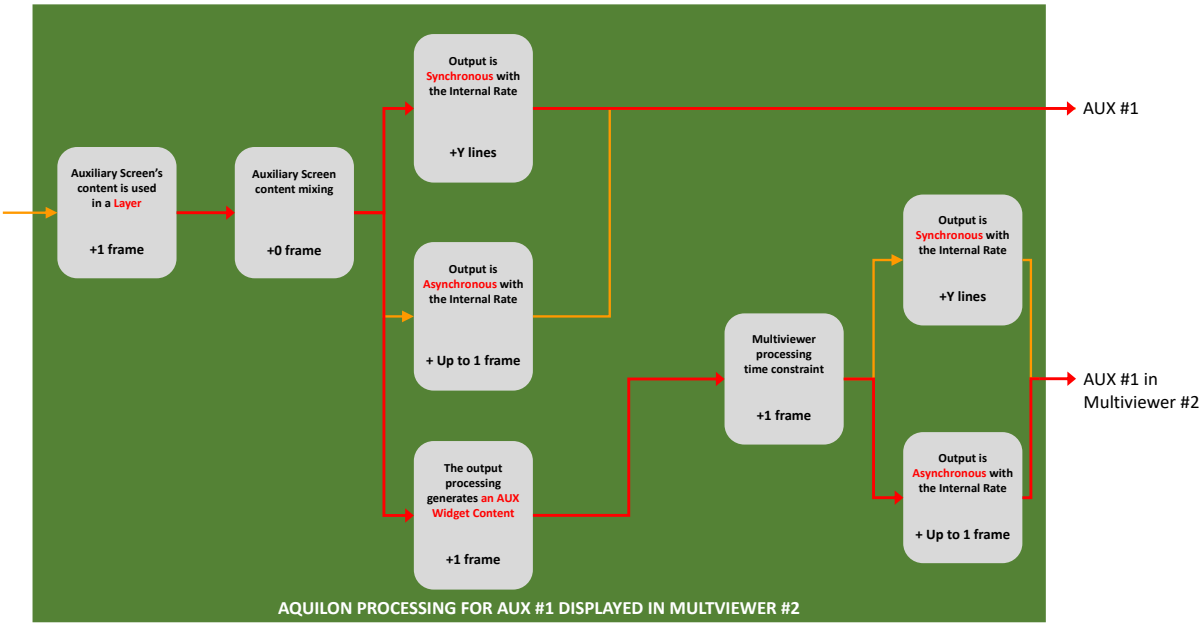

Figure 25: Data flow of Auxiliary Screen #1 displayed in Multiviewer #2

2 Frames < Latency referred to Aux #1 < 3 Frames

5.2.2.2. Latency of Main Screen Program and Preview

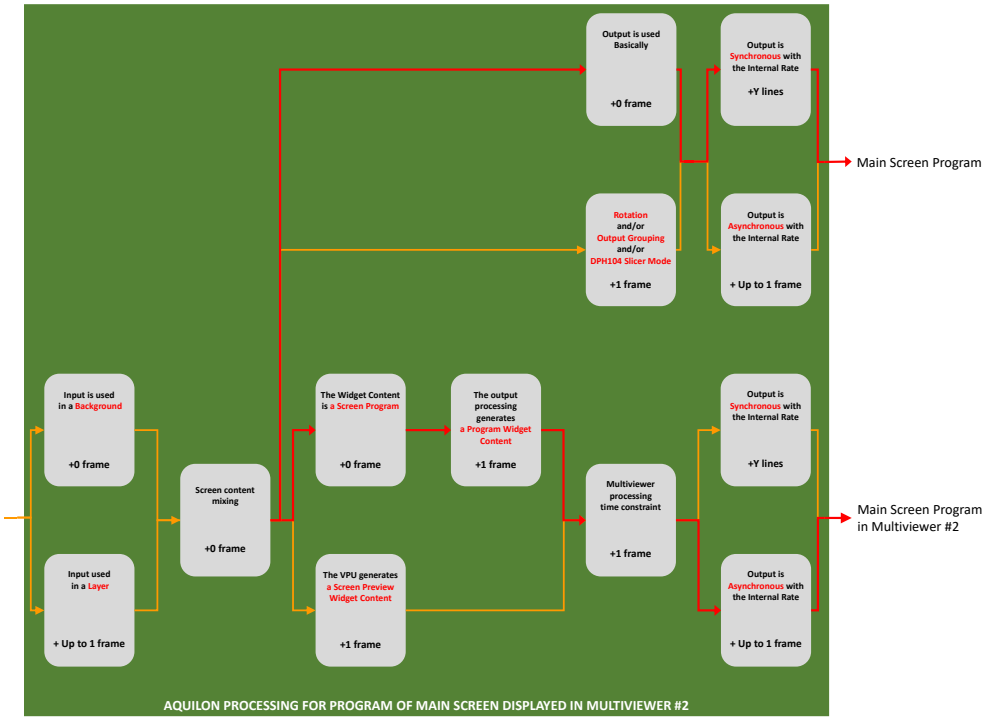

Figure 26: Data flow of Main Screen Program displayed in Multiviewer #2

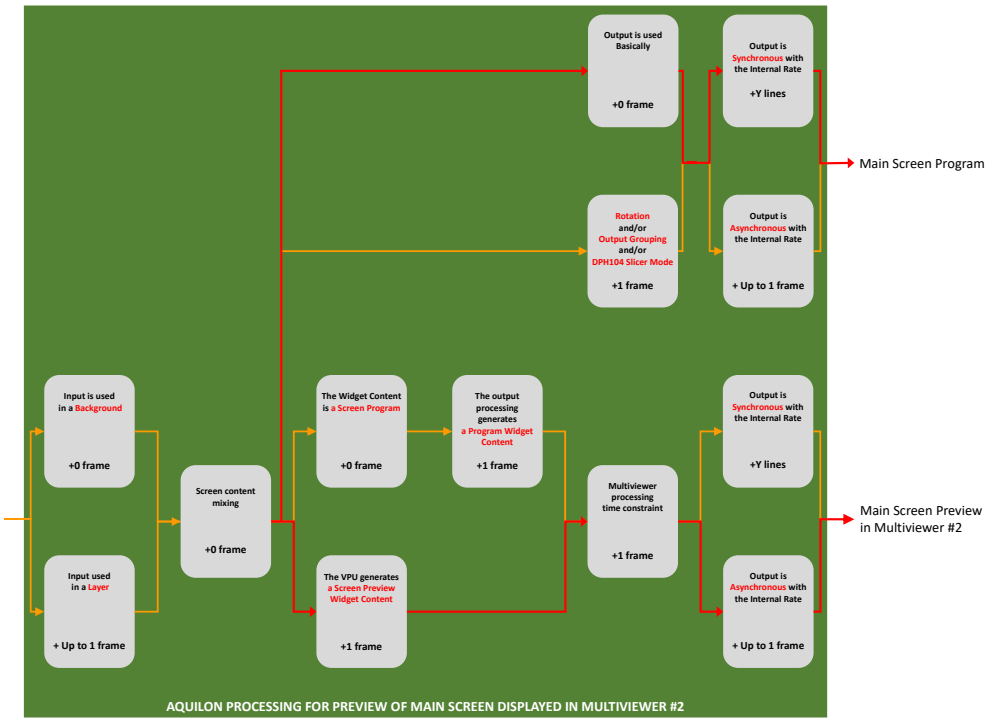

Figure 27: Data flow of Main Screen Preview displayed in Multiviewer #2

2 Frames < Latency referred to Main Screen < 3 Frames