

## IFF SDK on Jetson platform

This setup demonstrates capabilities of MRTech IFF SDK on one of the supported platforms.

### Hardware

- NVIDIA Jetson Xavier NX developer kit
- XIMEA MX089CG-SY-X2G2-FF camera with FireFly cable
- Laptop with NVIDIA GPU as a receiving station

### Software

- L4T operating system running on Jetson
- example IFF SDK application implementing RTSP streaming server with minimal code:

```
iff_initialize(base_cfg);  
void* handle = iff_create_chain(chain_cfg, error_handler);  
iff_chain_exec_cmd(handle, "streaming/on", "");
```

- demo RTSP streaming client application

IFF SDK supports Linux running on ARM and Intel CPUs and Windows operating systems.

IFF SDK can be customized or extended per customer request.

Integration with 3rd party software is also possible using import and export adapters.

### Features

- Acquisition from XIMEA, Basler, MIPI or other cameras.
- Various image processing modules, including, but not limited to black level subtraction, (auto-) white balancing, auto-exposure, demosaicing, gamma correction, H264/H265 encoding.
- RTSP/RTP streaming server and client.
- Reliable and proven production-ready code with deep technical support offered.
- Processing acceleration using NVIDIA GPU.
- JSON-based pipeline description language.
- Simple C SDK interface allowing for low-code solutions in simple cases and providing extensive capabilities for complex applications.
- Runtime control of various acquisition and processing parameters.

### Example operation performance

#### High dynamic range mode:

- 4K (3840x2160) ROI, 50 FPS, 12-bit acquisition
- 11x11 demosaic
- Hybrid Log-Gamma correction
- 8-bit H264 10 Mbit/s RTSP streaming
- glass-to-glass latency: 100-120 ms inc. 50 ms processing time

#### Low latency mode:

- 1080p (1920x1080) ROI, 2x2 camera skipping, 60 FPS, 8-bit acquisition
- 5x5 demosaic
- BT.709 gamma correction
- 8-bit H264 10 Mbit/s RTSP streaming
- glass-to-glass latency: 50-60 ms inc. 12 ms processing time