

Extended Lab System Goals and Objectives (MINIMUM, TARGET, STRETCH)

Please read the following general description of the Extended Lab SYSTEM GOALS and OBJECTIVES for Pairs or Groups. Note that you must verify and analyze your system so that you can accurately estimate the real-time accuracy and how predictable your response is for each test run.

Time-Lapse Image Acquisition MINIMUM GOALS and OBJECTIVES (Pairs and Groups)

System solution must acquire time-lapse images using a standard definition or VGA resolution digital camera that interfaces via USB, the Jetson camera port, or ethernet. Here are some detailed requirements for the time-lapse acquisition:

1. Resolution must be at least VGA (640x480) or [Standard Definition TV](#) (720x480 or 640x480) as documented [here](#).
2. You must acquire INDIVIDUAL frames from your camera - NOT an MPEG encoded stream. The frames must be acquired at 1 Hz for verification purposes (other rates are fine, but 1 Hz must be a supported mode).
3. Your camera must either be commanded to take an image at a 1 Hz rate or must continuously encode frames at a rate that is at least 24 frames/sec (24 Hz) or better which your frame acquisition service samples at 1 Hz ideally based on an interval timer like the PIT, a POSIX clock and notification, or use of accurate sleep mechanism. Frame rates for cameras can vary quite a bit and for NTSC can be at a rate of 59.94 interlaced fields per second or 29.97 frames/sec - please refer to documentation on your camera and [frame rates](#).
4. Image file format must be PPM "P3" or "P6" RGB format as documented for [Netpbm format](#) - if you camera encodes into a different format, you will need to convert. E.g. if your camera encodes into JPEG, MJPEG, TIF, etc., you must either use a library function in v4l2 or write your own conversion function to store and offload PPM format frames in ASCII or binary.
5. Your frame acquisition and time-stamping should be accurate so that there is no "observable" error in 1800 frames (or 1800 seconds of continuous operation over a wall-clock time period of 30 minutes).
6. Your time-stamps must be embedded in the PPM header of EVERY FRAME as a comment field (using "#" prefaced line in your header).
7. Your HOST or TARGET platform "uname -a" output must be embedded in the PPM header of EVERY FRAME as a comment field (using "#" prefaced line in your header) to clearly identify the system on which your test was run.

Time-Lapse Image Acquisition VERIFICATION MINIMUM GOALS and OBJECTIVES (Pairs and Groups)

1. You must verify that your image acquisition is accurate enough so that you can observe an EXTERNAL WALL CLOCK for a period of 30 minutes with high accuracy, or EXACTLY 1800 frames and that you capture the minutes display in a range of 0 to 30 or shifted anywhere within a start between 0 and 59 and stop anywhere at start+30 as displayed on the external clock and captured in your frames. So, for example, I should see 6:06 PM in the first frame and 6:36 PM in the last. Since your observability is 1 second, there is a small probability that you would see 6:06 PM and 6:35 PM if you acquisition started at 6:06 PM and 0.5 seconds and ended at 6:35 PM and 59.5 seconds. To ensure that you are accurate to +/- 1 frame or 1 second over the 30 minute period, you should count the initial frames that show the starting time and minute (e.g. 6:06 PM) and the number of frames that show the final stop time and minute (e.g. 6:36 PM or 6:35 PM). Your acquisition should be accurate so that you have no more than 1 second of error in the observation of the external clock over the 1800 second time-lapse acquisition - the same goes for your time-stamps in the headers - the difference should be 1800 seconds +/- 1 second or less. [Please see this](#)

- [diagram for more detailed information](#). You must compute your average frame jitter and determine if you have accumulated latency that builds up, or if you are running fast and by how much.
2. You must ALSO observe some physical process like the melting of a cube of ICE alongside your EXTERNAL CLOCK so that it proves this is a true time-lapse sequence when you encode it.
 3. You must encode your final video which includes the EXTERNAL CLOCK and the observed physical process into an MPEG2 or MPEG4 program stream using [ffmpeg](#) and include this on your final Canvas files along with your PPM frames.
 4. Include one more time-lapse video of any physical process indoors or outdoors that is of interest to you, but do not include the raw PPM frames for this second video, just the MPEG on your Canvas files.

Time-Lapse Image Acquisition TARGET GOALS (For Pairs and Groups)

Please choose to add one or more of the following features to your project and re-run your verification:

1. Compression of frames on your target so you can store more than 1800 frames and so that you have less to transfer over Ethernet.
2. Continuous download of frames over Ethernet so that you can run indefinitely and never run out of space on your flash file system which should maintain only the last 1800 frames.
3. Any other method you can dream up to capture and save at least 3x the minimum number of frames through compression and download or other means.
4. After adding either or both of the above features for compression and download, re-verify your frame jitter and accumulated latency as described above and describe and account for any differences you see.

Time-Lapse Image Acquisition STRETCH-GOALS (For Pairs and Groups)

Please choose to add ONE of the following features to your project:

1. Run at a higher frame rate of 10 Hz with continuous download, this time for 3 minutes, which will produce up to 1800 frames and repeat the jitter and accumulated latency verification at this higher rate.
2. Run at full frame rate and scan your RAW YUV or RGB image for foreground changes and acquire only images where you detect change that exceeds more than 1% of the total pixel value summed. Time-stamp each image when you trigger and observe a physical process with change. Stop when hit the same 1800 frames used in the verification. Attempt to repeat the EXTERNAL CLOCK verification and see if you can get the trigger to detect the change of display on the CLOCK you are observing with any sort of reliability.
3. Any other test you can dream up that runs at a higher rate, adds an element of real-time image processing, or otherwise significantly increases CPU loading significantly higher than 70% and allows you to verify if this causes more jitter and accumulated latency.

Time-Lapse Image Acquisition TARGET-GOAL EXTRA REQUIREMENTS (For GROUPS ONLY)

Please choose to add one of the following image processing features to your project:

1. Motion detection to trigger time-lapse acquisition.
2. Image processing features such as background elimination, sharpening of the image, detection of a target and computation of its centroid, segmentation of the image, statistics such as a spectral histogram computed for the image, etc.
3. Computation of difference images to highlight pixels that are changing, location of object centroids, etc.
4. Any other frame by frame image processing you dream up that can run at 1 Hz and faster 10 Hz rate as your STRETCH GOAL.