



IDT technical Note: 0011 (2014/09/16)

“SDK: acquire and store images in backup and streaming mode (Os camera)”

Backup mode

In backup mode, the camera records images in the DDR and, when the images are acquired, it transfers the data to the SSD. There is no limit in the frame rate, but the number of frames must fit in the camera DDR.

The procedure is shown below.

- Configure the camera with the required parameters. Set the XSP_PROP parameter to XS_PR_SSD_BACKUP. Make sure that the XSP_FRAMES parameter value is lower or equal to the maximum number of frames that fit in the camera DDR.
- Before recording images for the first time, make sure you have a clean SSD. Call the XsEraseDisk routine. If the SSD contains images that you don't want to erase, make sure that you don't overwrite them in the acquisition. To do that, check the "Read the images from the SSD" section and calculate the record address from the addresses of the previous acquisitions.
- To start recording, call the XsMemoryStartGrab. Then trigger the camera and wait for the callback or check the camera status.

```
XSULONG32 nStartAddLo, nStartAddHi, nFrames;
XSULONG32 nFrmSize, nMemLo, nMemHi;

// erase the SSD if doesn't contain useful images
XsEraseDisk(hCamera);

// Configure the camera
XsSetParameter(hCamera, &xsCfg, XSP_PROP, XS_PR_SSD_BACKUP);
XsRefreshCameraSettings( hCamera, &xsCfg );

// se the correct address
nStartAddLo = 0;
nStartAddHi = 0;

// calculate the max number of frames
XsGetParameter(hCamera, &xsCfg, XSP_FRAME_SIZE, &nFrmSize);
XsGetCameraInfo(m_hCam, XSI_MEMORY, &nMemLo, &nMemHi);

// divide the DDR size by the frame size
// the frame size changes if the ROI changes
nFrames =
(XSULONG32) (((XSINT64)nMemLo+(XSINT64) (nMemHi<<32))/nFrmSize);

// start recording
XsMemoryStartGrab(hCamera, nStartAddLo, nStartAddHi, nFrames, 0,
    fcallback, XS_CF_DONE, this);
```



Streaming mode

In streaming mode, the camera records images in the DDR and simultaneously stores them in the SSD. The maximum frame rate is limited by the SSD write speed. In this mode the user may acquire a number of images larger than the DDR size.

- Configure the camera with the required parameters. Set the XSP_PROP parameter to XS_PR_SSD_STREAMING. To calculate what is the number of frames you can acquire for a given frame rate, read the XSP_SSD_MAX_FRMS parameter. Make sure that the XSP_FRAMES parameter value is lower or equal to that value, otherwise you may receive an SSD write overrun error.
- Another important parameter that should be read is the XSP_SSD_STRM_PER parameter, that is, the inverse of the "streaming fps". The streaming fps is the value of the frame rate that corresponds to the write to disk speed.
 - If the frame rate is slower than the "streaming fps", the SSD is "faster" than the DDR, then the camera can record any number of frames without overrun. In this case the maximum number of frames is limited by the SSD size.
 - If the frame rate is faster than the "streaming fps", the SSD is "slower" than the DDR, then the camera has a limited number of frames that can be acquired before overrun (XSP_SSD_MAX_FRMS). This number is calculated assuming that the record mode is normal. If the record mode is circular, the overrun may occur if the user waits too long before triggering the camera.
- Before recording images for the first time, make sure you have a clean SSD. Call the XsEraseDisk routine. If the SSD contains images that you don't want to erase, make sure that you don't overwrite them in the acquisition. To do that, check the "Read the images from the SSD" section and calculate the record address from the addresses of the previous acquisitions.
- To start recording, call the XsMemoryStartGrab. Then trigger the camera and wait for the callback or check the camera status.

```
XSULONG32 nStartAddLo, nStartAddHi, nFrames;
XSULONG32 nFrmSize, nMemLo, nMemHi;

// nFrames is the number of images you want to acquire
nFrames = ...

// erase the SSD if doesn't contain useful images
XsEraseDisk(hCamera);

// Configure the camera
XsSetParameter(hCamera, &xsCfg, XSP_PROP, XS_PR_SSD_STREAMING);
XsRefreshCameraSettings( hCamera, &xsCfg );

// se the correct address
nStartAddLo = nStartAddHi = 0;
// calculate the max number of frames
XsGetParameter(hCamera, &xsCfg, XSP_SSD_MAX_FRMS, &nMaxFrames);

// make sure that yu record less than Max frames
if(nFrames>nMaxFrames) nFrames = nMaxFrames;

// start recording
XsMemoryStartGrab(hCamera, nStartAddLo, nStartAddHi, nFrames, 0,
    fcallback, XS_CF_DONE, this);
```



Read images from the SSD

The images stored in the camera SSD cannot be directly read, but they have to be copied to the camera DDR. The DDR acts as a temporary buffer from which the images may be read for playback or download.

The procedure is shown below:

- Read the list of acquisitions that are already stored in the SSD. the routine is **XsReadCameraSettingsArray**. The nOption parameter need to be set to 1.
- The routine returns an array of XS_SETTINGS structure. The parameters of each acquisition may be retrieved by the structure with the **XsGetParameter** routine. Important parameters are XSP_STARTADDRLO, XSP_STARTADDRHI (read the address in the SSD address space), XSP_FRAME_SIZE (the size of a frame), XSP_FRAMES (the total number of acquired frames). Those parameters may be used to calculate the total size of the acquisition in the camera SSD.
- Select the acquisition that will be downloaded by sending the corresponding parameters to the camera (**XsRefreshCameraSettings**).
- Call the **XsMemoryReadFromDisk**. The DDR address may be set to 0 and the SSD address is set to the start address of the corresponding acquisition. Then the frames may be copied in groups of N images by setting the start and the stop index. The number of frames that will be copied from the SSD to the DDR may be the total number of frames if the acquisition fits in the camera DDR, or a multiple of 256 if it doesn't.

```
XS_SETTINGS aXCfg[1024] = {0};

XSULONG32 nAddLo, nAddHi, nCount, nFrames;
XSULONG32 nFrmSize, nMemLo, nMemHi, nDDRFrms, nStart, nStop;

// read the array of configurations
nCount = sizeof(aXCfg)/sizeof(XS_SETTINGS);
XsReadCameraSettingsArray(hCamera, 1, &aXCfg[0], &nCount);

// select the configuration number 2 (just an example!!!)
XsRefreshCameraSettings( hCamera, &aXCfg[2] );

// read address
XsGetParameter(hCamera, &aXCfg[2], XSP_STARTADDRLO, &nAddLo);
XsGetParameter(hCamera, &aXCfg[2], XSP_STARTADDRHI, &nAddHi);
XsGetParameter(hCamera, &aXCfg[2], XSP_FRAMES, &nFrames);

// calculate the max number of frames
XsGetParameter(hCamera, &aXCfg[2], XSP_FRAME_SIZE, &nFrmSize);
XsGetCameraInfo(hCamera, XSI_MEMORY, &nMemLo, &nMemHi);

// divide the DDR size by the frame size
// and find the DDR room
nDDRFrms =
(XSULONG32) (((XSINT64) nMemLo + (XSINT64) (nMemHi << 32)) / nFrmSize);

// read the frames
nStart = 0;
if (nFrames < nDDRFrms) nStop = nFrames - 1;
else nStop = 255;
```



```
while( nStop<=(nFrames-1) )
{
    XsMemoryReadFromDisk(hCamera,0,0,nAddLo, nAddHi, nStart,
                        nStop, pfnCallback, this);
    nStart += 256;
    nStop += 256;
    if(nStop>(nFrames-1)) nStop = nFrames-1;
}
```