



celepixel

芯仑科技 SDK 使用说明

芯仑科技（上海）有限公司

目录

1	CeleX-5 SDK 使用步骤.....	3
1.1	安装 MIPI 转 USB3.0 驱动.....	3
1.1.1	Windows.....	3
1.1.2	Linux.....	4
1.2	运行 CeleX-5 Demo GUI	5
1.2.1	Windows.....	5
1.2.2	Linux.....	5
1.3	编译 CeleX-5 SDK 的 Source Code.....	6
1.3.1	Windows.....	6
1.3.2	Linux.....	6
1.4	编译 CeleX-5 Demo GUI 的 Source Code.....	7
1.5	编译 CeleX-5 Driver 的 Source Code	9
1.5.1	Windows.....	9
1.5.2	Linux.....	9
1.6	采集 FPN 文件	10
2	CeleX-5 Demo Kit GUI 的功能.....	11
2.1	Sensor 的 Fixed 模式介绍.....	11
2.2	Fixed Mode 与 Loop Mode 切换.....	13
2.3	录制 Sensor 数据功能.....	14
2.4	播放录制的 Bin 文件功能	15
2.5	采集 FPN 功能	17
2.6	图像翻转功能.....	20
2.7	Bin 转视频功能	21
2.8	Bin 转 CSV 文件功能	22
2.8.1	Event Off-Pixel Timestamp 模式的 CSV 格式	22
2.8.2	Event In-Pixel Timestamp 模式的 CSV 格式	22
2.8.3	Event Intensity 模式的 CSV 格式	23
2.9	Configurations.....	24
2.10	Advanced Settings.....	26

1 CeleX-5 SDK 使用步骤

1.1 安装 MIPI 转 USB3.0 驱动

1.1.1 Windows

把 CeleX-5 Sensor 通过 USB 线连接到 PC 上，双击 **zadig-2.4.exe** 弹出如下图 1-1 所示的界面，选择 Options → List All Devices (图 1-2)，然后选择设备 FX3 (图 1-3)，点击 Install Driver 或 Reinstall Driver (图 1-4) 安装驱动，安装成功后会弹出图 1-5 所示的界面。

备注：数据线必须连接 PC 上的 USB3.0 端口。

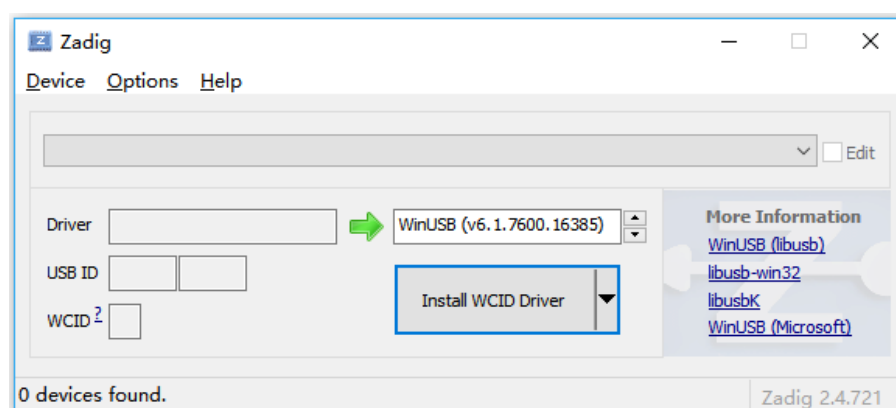


图 1-1

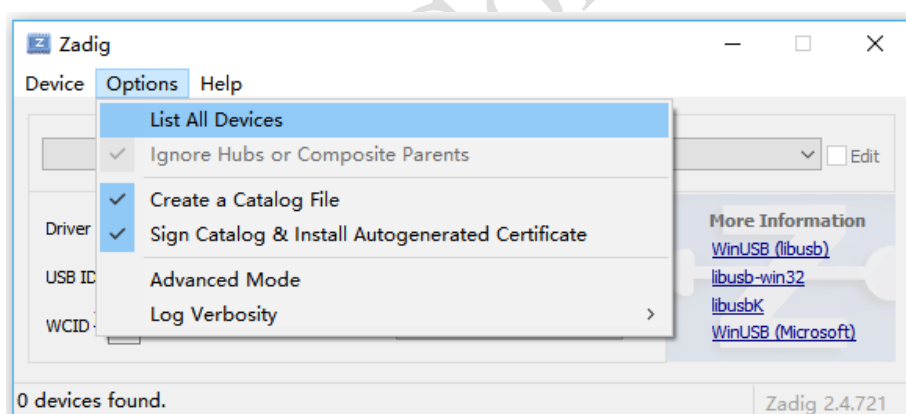


图 1-2

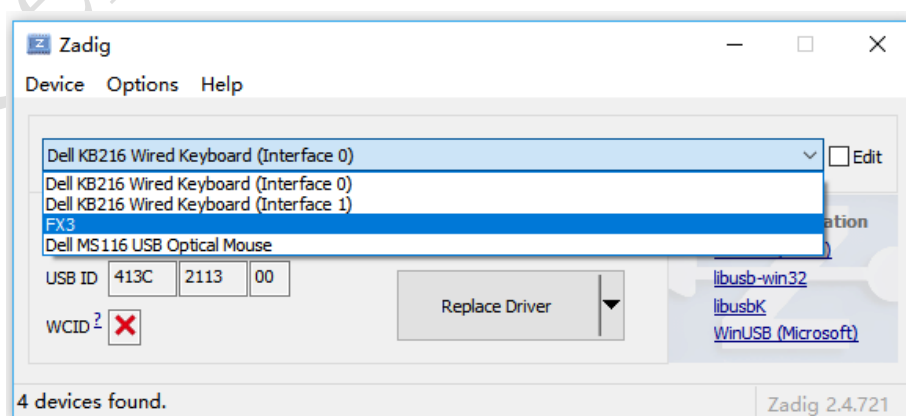


图 1-3

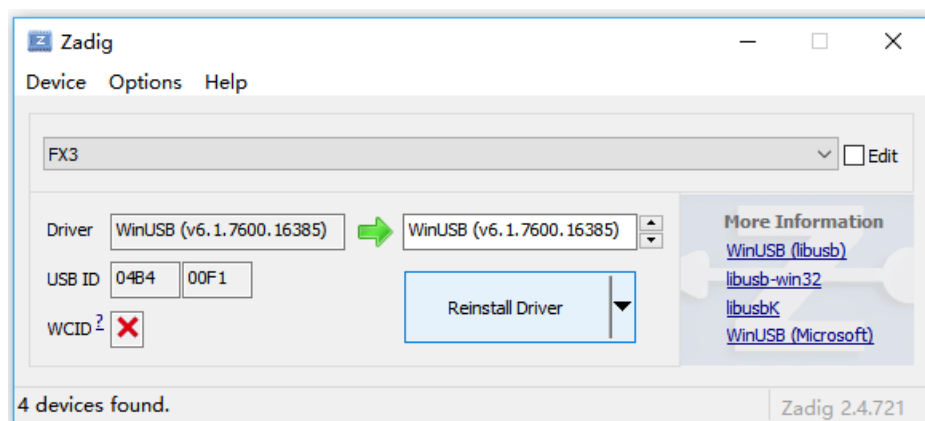


图 1-4

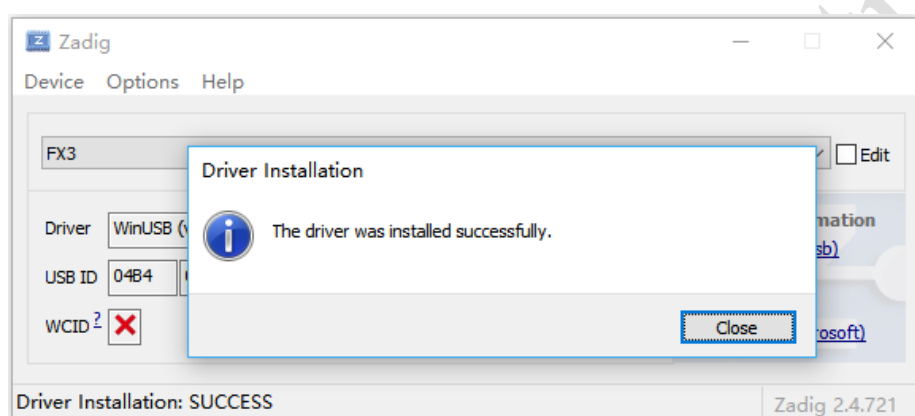


图 1-5

1.1.2 Linux

Linux 下安装 CeleX-5 Sensor 所需的驱动,将发布包 *Drivers/Linux/*目录下的压缩包解压,如图 1-6 所示,运行命令“*sudo sh install.sh*”即可进行安装,如图 1-7。

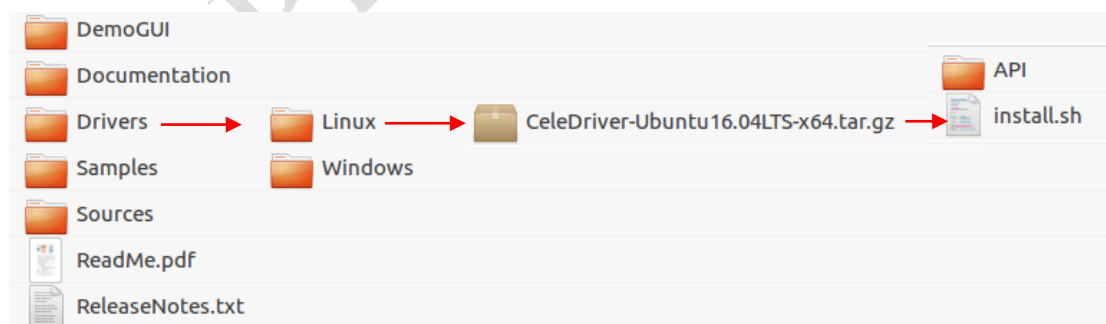


图 1-6

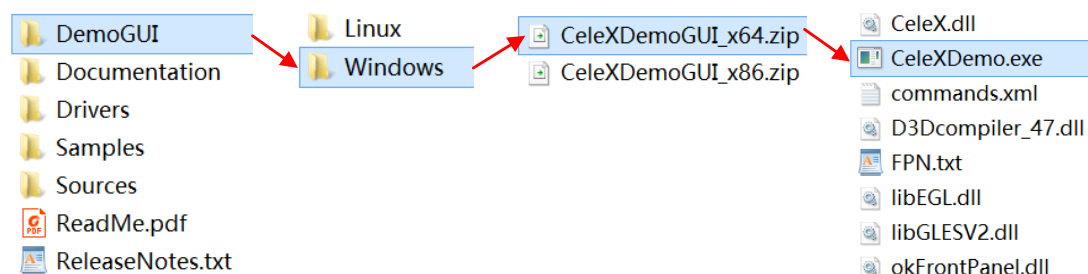
```
sudo sh ./install.sh
```

图 1-7

1.2 运行 CeleX-5 Demo GUI

1.2.1 Windows

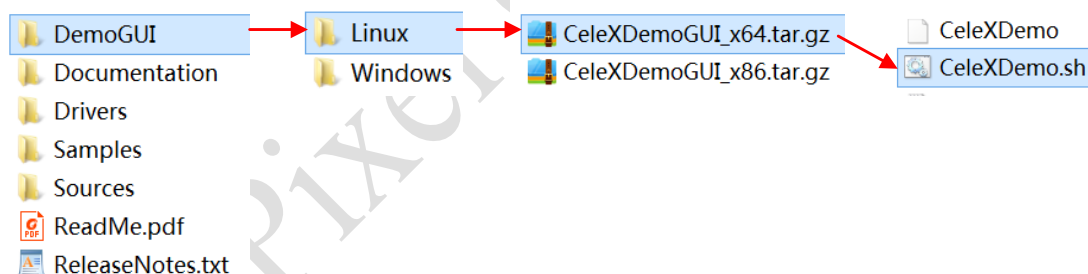
安装驱动程序后，用户可以从发布文件夹打开 Demo GUI，双击“CeleXDemo.exe”即可正常打开 CeleX Demo GUI。打开后的界面如图 2-2 所示（第 2 章）。



备注：如果 CeleXDemo.exe 无法打开，且 Windows 消息框显示缺失某些 dll 文件，这可能是由于缺少 Visual C ++支持包所造成的。可以在 *Drivers/Windows* 文件夹下安装“vc_redist.x86.exe”并再次尝试，则 CeleX Demo 应该可以正常工作。

1.2.2 Linux

安装驱动程序后，用户可以从以下文件夹打开 Demo GUI，打开终端，进入 CeleXDemo.sh 所在的目录，输入命令“*sudo sh CeleXDemo.sh*”，即可打开 CeleX-5 Demo GUI，打开后的界面如图 2-2 所示（第 2 章）。



```
$ sudo sh CeleXDemo.sh
```

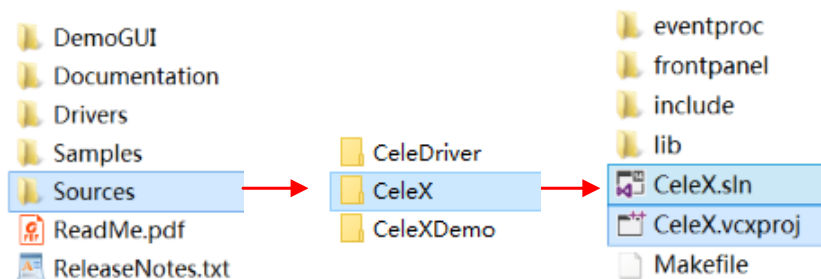
备注：这里要用 root 权限打开 Demo，因为我们需要对 usb driver 进行读写操作，没有 root 权限可能会造成打开 usb 设备失败的问题。

1.3 编译 CeleX-5 SDK 的 Source Code

本 SDK 中会使用 OpenCV 库（版本为 3.3.0），所以在编译源码之前请先安装 OpenCV 库并配置好其编译环境。

1.3.1 Windows

在 Window 平台上，我们提供了 VS2015 的工程直接编译该源码，可以按照以下图示进入 SDK 的 Source Code 目录：

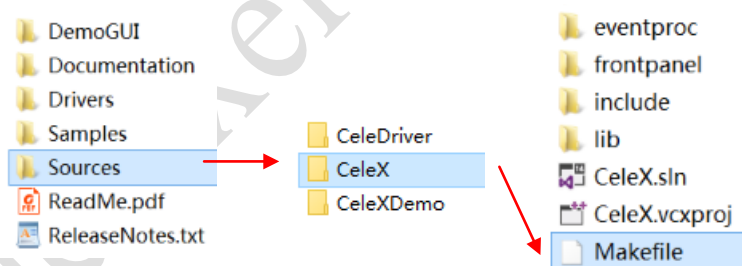


备注：

- (1) 需要修改工程属性中关于 OpenCV 的 Include 和 Lib 的路径的设置，否则会因为找不到 OpenCV 的头文件和库而编译失败。
- (2) 编译完成后，会在工程所在的目录下自动创建一个 build/Windows 目录，编译生成的库文件（CeleX.dll 和 CeleX.lib）会被自动导入到该目录下。

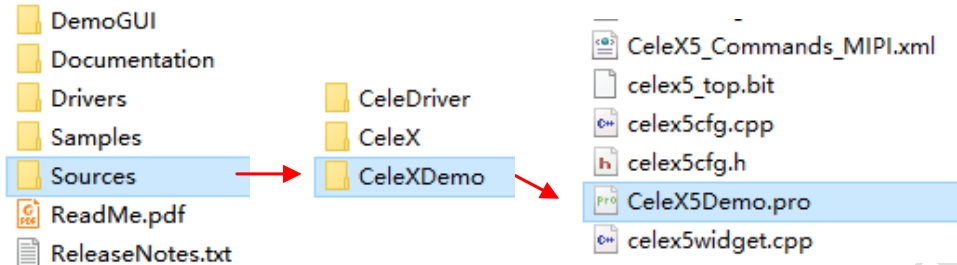
1.3.2 Linux

在 Linux 平台上，我们提供一个 Makefile 编译该代码，库文件（libCeleX.so）将生成在当前目录下。



1.4 编译 CeleX-5 Demo GUI 的 Source Code

由于本 Demo 是用 Qt 开发的，所以在编译该代码之前要先安装 Qt（本 Demo 使用的 Qt 版本为：qt-opensource-windows-x86-msvc2015_64-5.6.3.exe）。由于 Qt 也是跨平台的，所以 Windows 和 Linux 平台上，都可以用 Qt Creator 打开 *CeleXDemo.pro* 即可编译。



需要注意的是，由于本 Demo 中也用到了 OpenCV 的一些接口，所以需要修改一下 *CeleXDemo.pro* 文件中关于 OpenCV 的路径设置，如下所示：

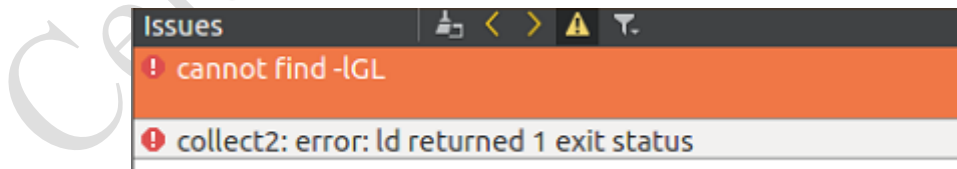
```
win32 {
    INCLUDEPATH += D:/opencv/build/include \
                  D:/Program Files/opencv/build/include/opencv \
                  D:/Program Files/opencv/build/include/opencv2
}
else {
    INCLUDEPATH += /usr/local/include \
                  /usr/local/include/opencv \
                  /usr/local/include/opencv2

    LIBS += /usr/local/lib/libopencv_highgui.so \
            /usr/local/lib/libopencv_core.so \
            /usr/local/lib/libopencv_imgproc.so \
            /usr/local/lib/libopencv_videoio.so
}
```

备注：Linux 下可能会遇到的编译错误

(1) OpenGL 错误



如果在编译的过程中，遇到以下错误，则需要安装 OpenGL 库（Qt 依赖 OpenGL 库），否则跳过该步骤。在终端上输入命令：sudo apt-get install libgl1-mesa-dev



(2) OpenCV 版本不兼容的问题（本 SDK 中使用的 OpenCV 版本为 3.3.0）

如果在编译的过程中，遇到该问题，则需要先编译一下 API 库（详见 1.3 章节），然后把编译好的 libCeleX.so 文件，替换掉 ../Sources/CeleXDemo/lib/Linux/x64/ 目录下的 libCeleX.so，再重新编译即可解决该问题。



Sources > CeleXDemo > lib > Linux > x64	
名称	修改日期
 libCeleDriver.so	2019/3/15 18:16
 libCeleX.so	2019/3/15 18:16

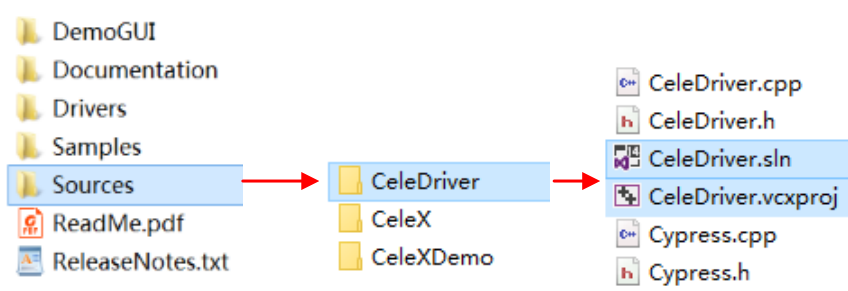
CelePixel Confidential

1.5 编译 CeleX-5 Driver 的 Source Code

CeleX-5 Driver 用于获取 USB 端的数据，SDK 再通过 CeleX-5 Driver 来获取数据进行后续处理。

1.5.1 Windows

在 Windows 平台上，我们提供了 VS2015 的工程直接编译该源码，可以按照以下图示进入 Driver 的 Source Code 目录：

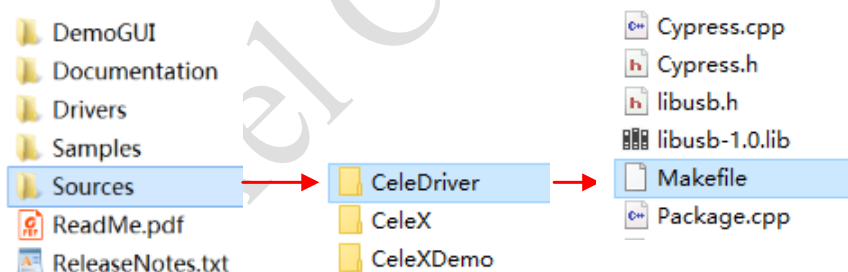


备注：

- (1) 编译完成后，会在工程所在的目录下自动创建一个 build/Windows 目录，编译生成的库文件（CeleDriver.dll 和 CeleDriver.lib）会被自动导入到该目录下。

1.5.2 Linux

在 Linux 平台上，我们提供一个 Makefile 编译该代码，库文件（libCeleDriver.so）将生成在当前目录下。



1.6 采集 FPN 文件

固定模式噪声 (FPN, Fixed Pattern Noise) 是数字图像传感器上的特定噪声模式的术语, 在较长的曝光镜头中经常可见, 其中特定像素易于在一般背景噪声之上提供较亮的强度。如果不从图像中减去 FPN, 则图像可能显示出高水平的背景噪声, 因此变得粗糙。为了解决该问题, 我们需要为 CeleX-5 Sensor 生成一个 FPN 文件, 具体的操作步骤请参考 [2.5 章节](#)。

CelePixel Confidential

2 CeleX-5 Demo Kit GUI 的功能

打开 CeleXDemo.exe，当没有连接 Sensor 设备时，界面如图 2-1 所示；当有 Sensor 设备连接时，界面如图 2-2 所示。

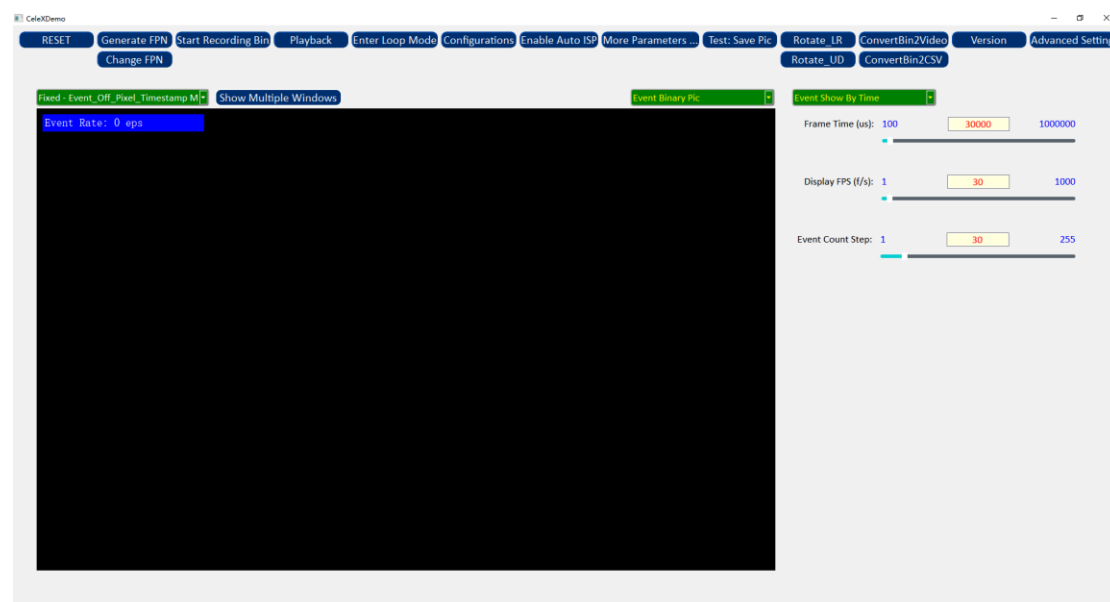


图 2-1

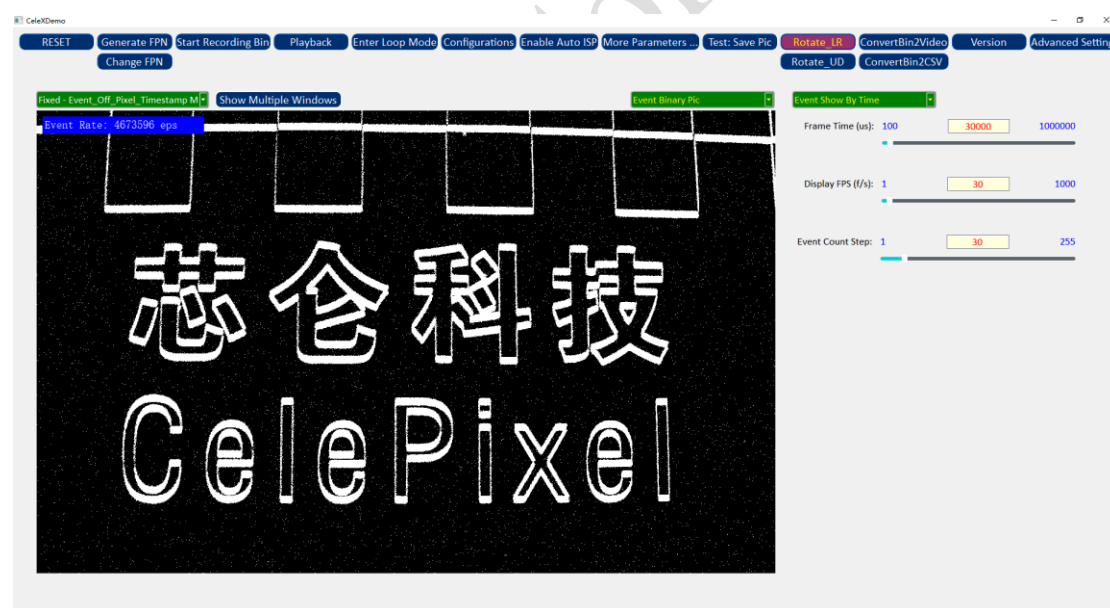


图 2-2

2.1 Sensor 的 Fixed 模式介绍

本 SDK 提供了 Sensor 的 5 种 Fixed 模式，用户可以通过 Demo GUI 在这 5 种模式种任意切换，下图给出了每个模式的名称以及对应的图像。



Event Off-Pixel Timestamp Mode



Event In-Pixel Timestamp Mode



Event Intensity Mode



Full-Picture Mode



Optical-Flow Mode

2.2 Fixed Mode 与 Loop Mode 切换

在 Fixed Mode 中，点击图 2-2-1 所示的红色框按钮“**Enter Loop Mode**”，可以进入 Loop Mode，其图像显示如图 2-2-3 所示。其中 Loop A 为第一个 loop，其模式为 Full-Picture 模式，Loop B 为第二个 loop，其模式为 Event 模式，Loop C 为第三个 loop，其模式为 Optical-Flow 模式。

在 Loop Mode 中，点击图 2-2-2 所示的红色框按钮“**Enter Fixed Mode**”，即可切换至 Fixed 模式（默认为 Event 模式）。



图 2-2-1



图 2-2-2

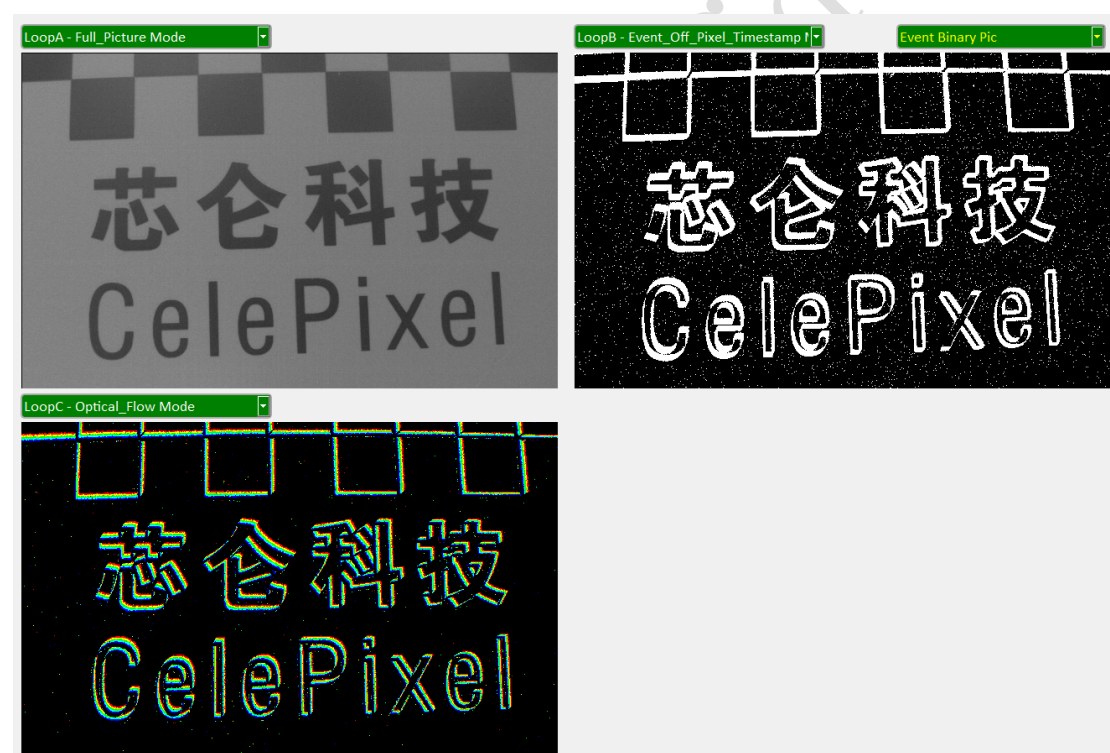


图 2-2-3 Sensor 工作在 Loop 模式

2.3 录制 Sensor 数据功能

点击图 2-3-1 中 “**Start Recording Bin**” 按钮即可开始录制 bin 数据，开始录制数据后，按钮上的文字会变成图 2-3-2 所示的 “**Stop Recording Bin**”，再点击“**Stop Recording Bin**”按钮即停止录制 bin 数据。录制的 bin 文件就存在 CeleXDemo.exe 的同目录下，以 MipiData_YYYYMMDD_HHMMSSSSS_SensorMode_ClockRate.bin 的形式命名，如下所示：

```

MipiData_20190418_140135451_E_100M.bin
MipiData_20190418_140139275_EO_70M.bin
MipiData_20190418_140144355_EI_70M.bin
MipiData_20190418_140148054_F_100M.bin
MipiData_20190418_140151731_FO1_100M.bin
MipiData_20190418_140253421_Loop_100M.bin
    
```

模式的缩写介绍：

- (1) **E** 表示录制的是 *Event Off Pixel Timestamp* 模式下的数据
- (2) **EO** 表示录制的是 *Event In Pixel Timestamp* 模式下的数据
- (3) **EI** 表示录制的是 *Event Intensity* 模式下的数据
- (4) **F** 表示录制的是 *Full-Picture* 模式下的数据
- (5) **FO1** 表示录制的是 *Optical-Flow* 模式下的数据
- (6) **Loop** 表示录制的是 *Loop* 模式下的数据

工作频率缩写介绍：

100M 表示 Sensor 的工作频率为 100MHz。

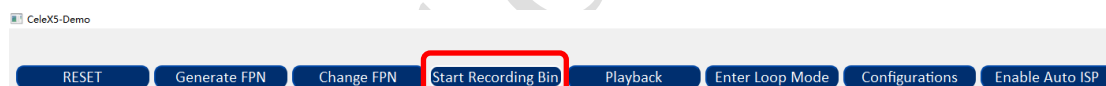


图 2-3-1



图 2-3-2

2.4 播放录制的 Bin 文件功能

点击“**Playback**”按钮（如图 2-4-1 所示），选择一个 bin 文件，开始播放后，按钮上的文字会变成“**Exit Playback**”（如图 2-4-2 所示），那点击“**Exit Playback**”按钮即会停止播放 bin 文件并退出 playback 的界面，回到实时显示的界面。其中播放的内容，跟录制 Bin 数据时的 Sensor 模式有关。

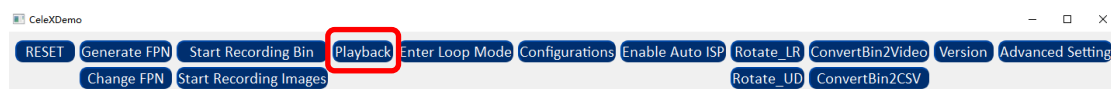


图 2-4-1

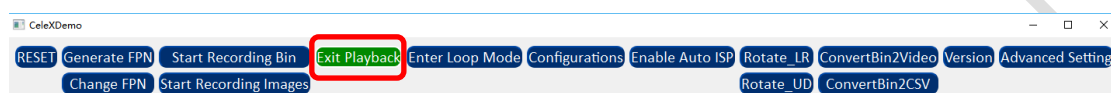
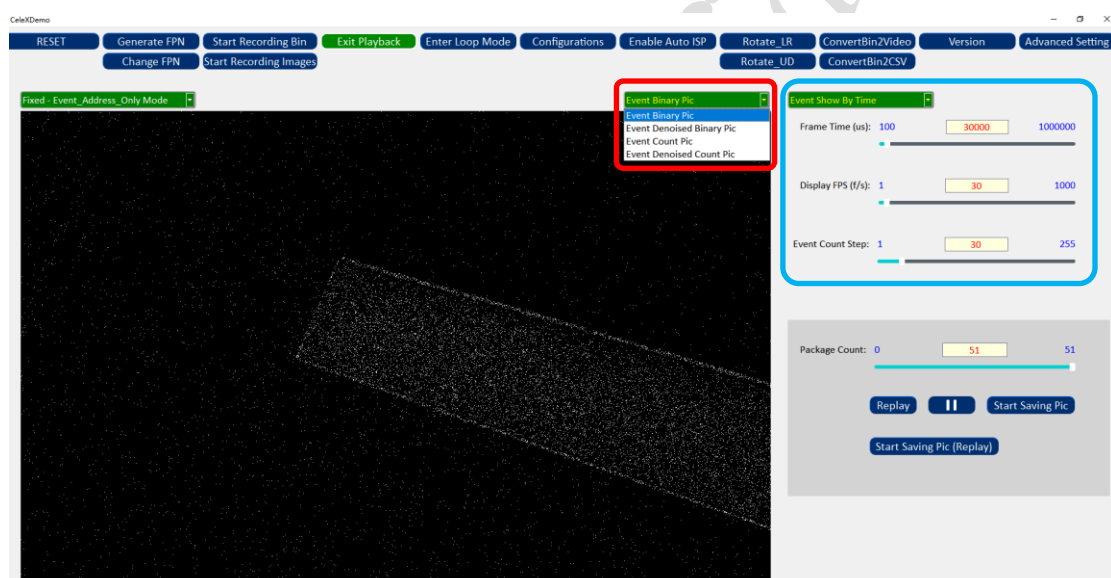


图 2-4-2

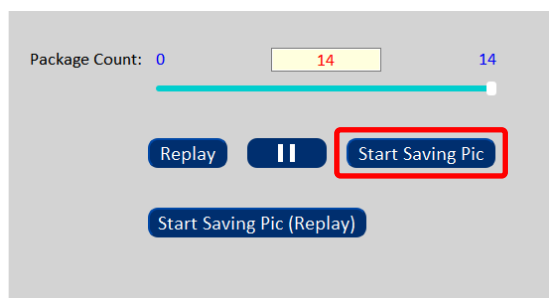
用户可以选择该 Bin 文件的各种 Pic 模式进行显示，也可以选择显示方式以及设置建帧时长（Frame Time）或是刷新频率（Display FPS）等参数。



用户可以选择该 Bin 文件的各种 Pic 模式进行显示(上图中红色框所示)，也可以修改建帧时长（Frame Time）或是刷新频率（Display FPS）等参数（上图中蓝色框所示）。

Playback 时还支持“**Replay**”、“**Play/Pause**”以及保存图片，其中“**Start Saving Pic**”表示从当前播放位置开始保存图片，“**Start Saving Pic (Replay)**”表示从头开始保存图片。

例如点击“**Start Saving Pic**”按钮，则开始保存图片；开始保存图片后，按钮上的文字会变成“**Stop Saving Pic**”，再点击“**Stop Saving Pic**”按钮，即可停止保存图片。



CelePixel Confidential

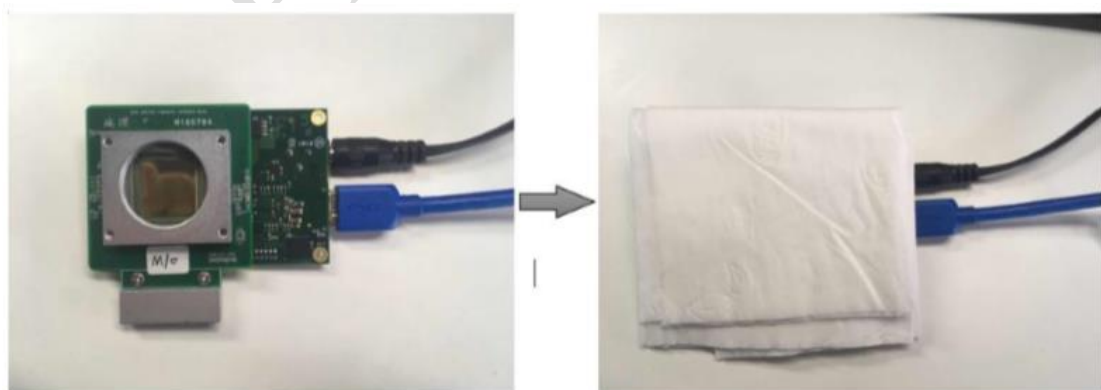
2.5 采集 FPN 功能

固定模式噪声 (FPN, Fixed Pattern Noise) 是数字图像传感器上的特定噪声模式的术语, 在较长的曝光镜头中经常可见, 其中特定像素易于在一般背景噪声之上提供较亮的强度。如果不从图像中减去 FPN, 则图像可能显示出高水平的背景噪声, 因此变得粗糙。为了解决该问题, 我们需要为 CeleX-5 Sensor 生成一个 FPN 文件, 具体的操作步骤如下:

- (1) 将 Sensor 的工作模式切换至 Full-Picture 模式。



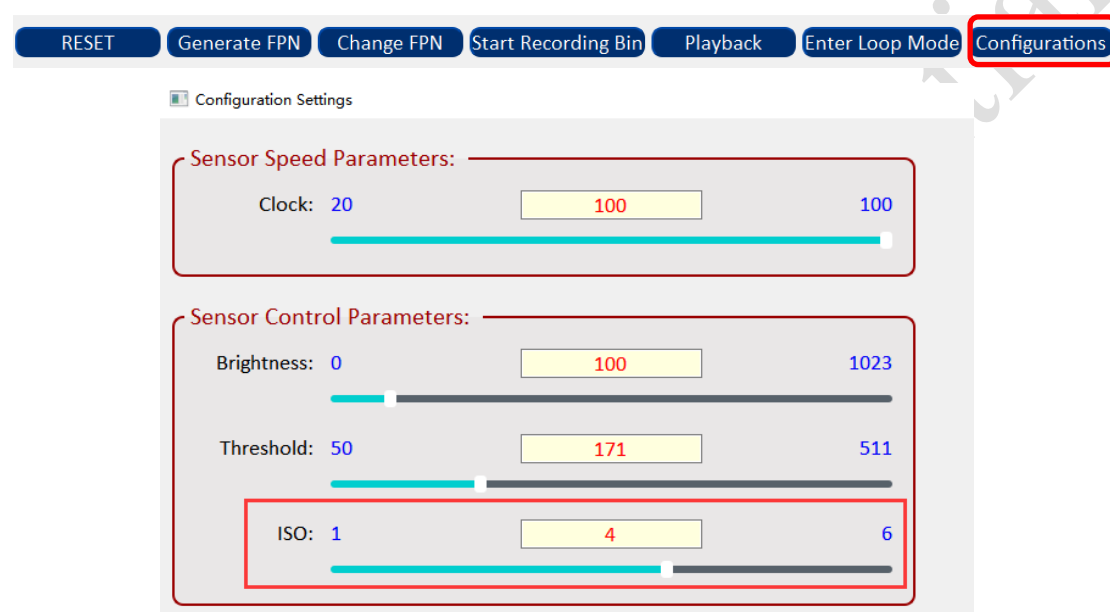
- (2) 由于 FPN 生成操作必须在光照均匀的环境下进行, 所以我们可以取下一光学镜头并用一张白纸 (薄纸或 A4 打印纸) 覆盖裸露的 Sensor 来实现这种情况。确保纸张均匀地完全覆盖传感器, 并且纸张保持静止。**备注: 如果你是在阳光下而不是 LED 灯下操作, 效果会更好。**



- (3) 执行 FPN 生成操作之前, 请检查图像屏幕, 确保其显示正常, 不要太暗或太亮。只需在裸露的 Sensor 上放置更多或更少的纸张, 或者在 GUI 窗口上打开或关闭“亮度”滑块, 即可更改照明。**备注: 下图中的第 3 幅图就是正常亮度的图。**



- (4) 点击 GUI 窗口中的“**Generate FPN**”按钮，当你在指定目录下看到 FPN_3.txt 文件时表明 FPN 文件成功生成了。备注：不同的 ISO 档位对应了不同的 FPN 文件，ISO 一共有六档，默认是第三档，对应了 FPN_3.txt 文件。用户可以通过调整 Configuration 设置中的 ISO 档位来获取更亮或更暗的图像。



- (5) 生成相应 ISO 设置下的 FPN 文件后，我们通过点击“**Change FPN**”按钮可以选择切换到相应的 FPN。备注：如果按照步骤生成了 FPN 文件，但是切换后图像清晰度没有提高，检查确认当前 ISO 与 FPN 是否对应；检查选择的 FPN 路径是否包含中文路径。

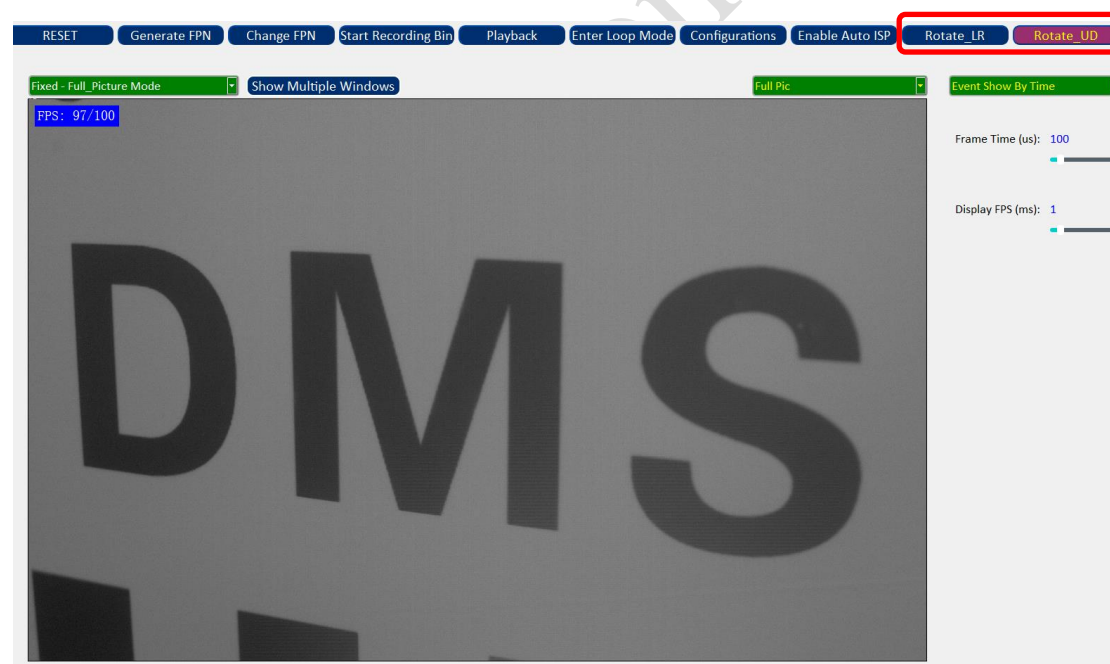
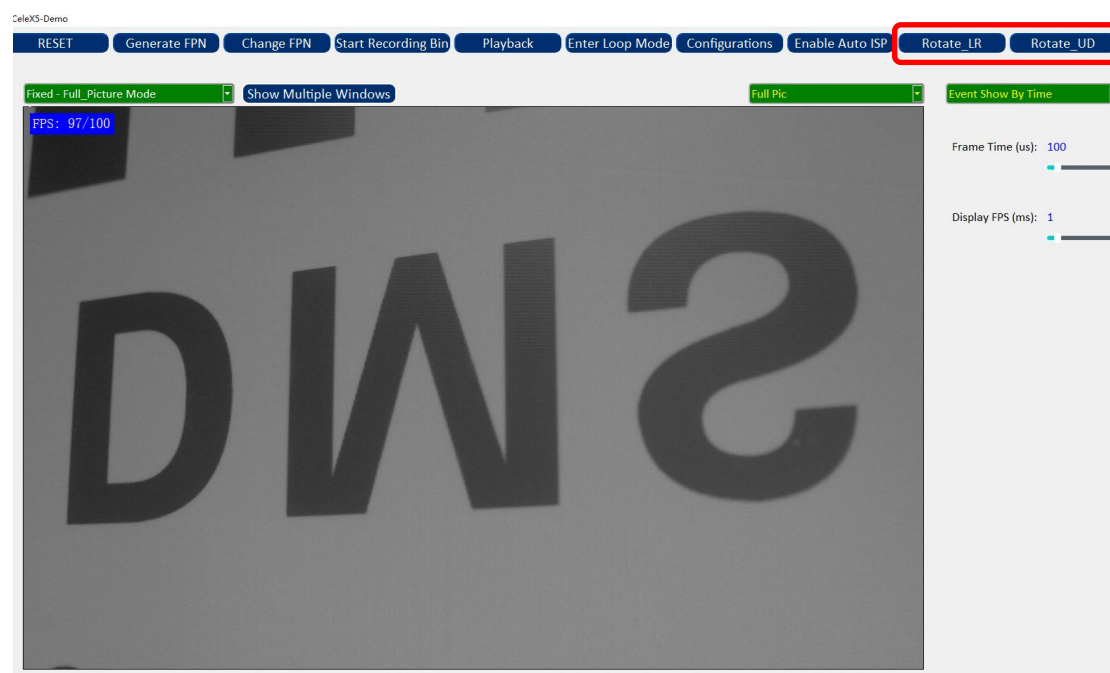




CelePixel Confidential

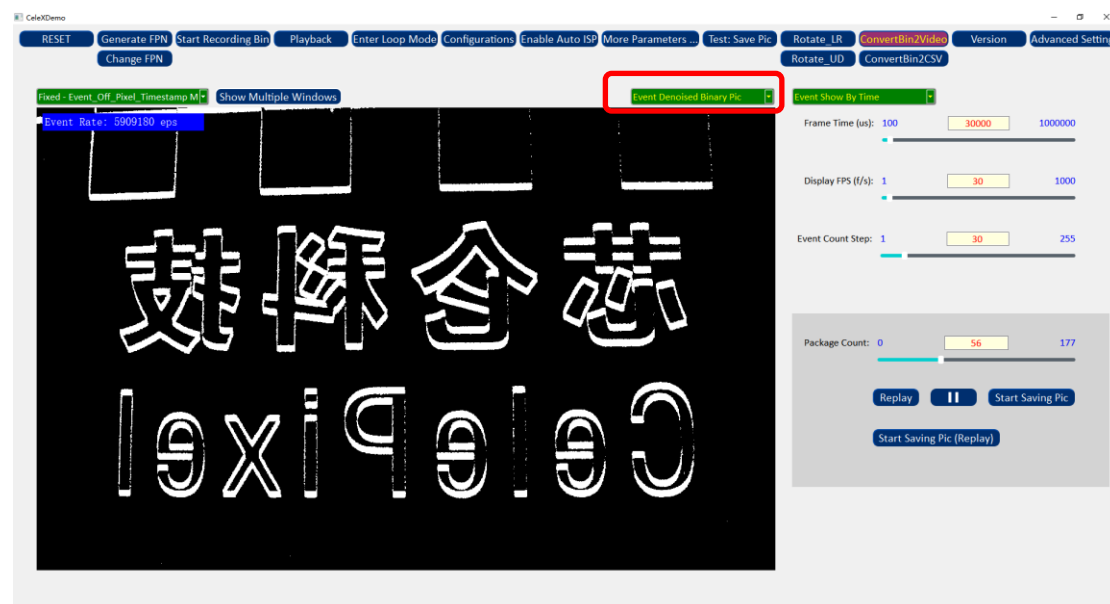
2.6 图像翻转功能

通过点击“*Rotate_LR*”和“*Rotate_UD*”按钮对图像进行左右或者上下翻转。



2.7 Bin 转视频功能

点击“**ConvertBin2Video**”按钮，可以将录制的 Bin 文件转换出与该文件同名的视频文件。Windows 下生成.mkv 格式的视频文件，Linux 下生成.mp4 格式的视频文件。选择不同的图片格式，可以转换出该 Bin 相应的图片格式视频。例如：选择去噪的图片格式，可以将 Event Off Pixel Timestamp 模式的 Bin 文件转换成去噪后的图像视频。



2.8 Bin 转 CSV 文件功能

点击 “*ConvertBin2CSV*” 按钮，可以将录制好的 Bin 文件转换成 CSV 文件。

2.8.1 Event Off-Pixel Timestamp 模式的 CSV 格式

对于 *Event Off-Pixel Timestamp 模式*，转出来的 CSV 文件格式如下表所示，A 列为行地址（范围：0~799），B 列为列地址（范围：0~1179），C 列为 Off-Pixel 时间戳（范围：0~1499，时间精度：20 μ s）。

A	B	C
458	849	0
459	425	0
460	705	0
463	1080	0
465	181	0
465	706	0
466	1166	0
468	234	0
470	926	0
471	1166	0
474	844	1
477	150	1

备注：默认情况下，上表的 C 列的时间戳每隔 30ms（即 Event Frame Time）会重置一次。如果在 *Advanced Settings* 配置界面中，选择了保存递增的时间戳，那么 C 列的时间戳就不会重置，而是递增的。

Other Parameters:

Resolution Parameter: 0 255

Whether to save a increasing timestamp when Bin converts to CSV

☒ open

2.8.2 Event In-Pixel Timestamp 模式的 CSV 格式

对于 *Event In-Pixel Timestamp 模式*，转出来的 CSV 文件格式如下表所示，A 列为行地址（范围：0~799），B 列为列地址（范围：0~1179），C 列为 In-Pixel 时间戳（范围：0~4095，时间精度：3.5 μ s），D 列为 In-Pixel 时间戳的 Ramp 周期号，E 列为 Off-Pixel 时间戳（范围：0~2141，时间精度：14 μ s）。

A	B	C	D	E
782	455	2105	1	0
782	890	2396	1	0
782	1063	2743	1	0
783	833	2902	1	0
783	1243	3327	1	0
784	10	1913	1	1
784	120	2161	1	1
784	200	2428	1	1
784	655	2820	1	1
785	356	2433	1	1

备注：默认情况下，上表的 E 列的时间戳每隔 30ms（即 Event Frame Time）会重置一次。如果在 *Advanced Settings* 配置界面中，那么转出来的 CSV 文件格式如下表所示，A 列为行地址，B 列为列地址，C 列为递增的 In-Pixel 时间戳（Incremental In-Pixel Timestamp = In-Pixel Timestamp + RampNo*4096），D 列为递增的 Off-Pixel 时间戳。

A	B	C	D
341	1153	4236	14433
342	718	4215	14434
342	1096	4113	14434
345	67	4117	14434
345	639	4096	14434
345	843	4107	14434
346	922	4159	14435
346	1098	4096	14435
351	457	4204	14435
351	782	4096	14435

2.8.3 Event Intensity 模式的 CSV 格式

对于 *Event Intensity 模式*，上表的转出来的 CSV 文件格式如下表所示，A 列为行地址（范围：0~799），B 列为列地址（范围：0~1179），C 列为灰度值（范围：0~4095），D 列为 Polarity（范围：+1, -1, 0），E 为每个 event 的输出时间戳（范围：0~2141，时间精度：14μs）。

A	B	C	D	E
306	644	721	1	0
307	201	0	0	0
307	802	432	1	0
308	16	238	1	1
308	884	127	1	1
309	171	304	1	1
309	884	800	1	1
309	1248	378	1	1
310	627	662	1	2

备注：默认情况下，E 列的时间戳每隔 30ms（即 Event Frame Time）会重置一次。如果在 *Advanced Settings* 配置界面中，选择了保存递增的时间戳，那么 E 列的时间戳就不会重置，而是递增的。

2.9 Configurations

点击“**Configurations**”按钮（如图 2-9-1 所示），打开配置界面，可以对 Sensor 进行一些配置。打开后的配置界面，如图 2-9-2 所示。

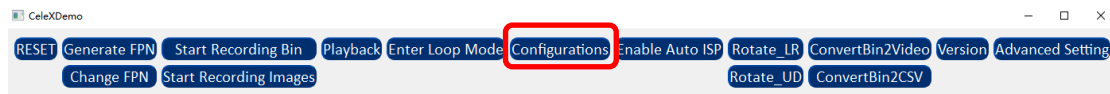


图 2-9-1

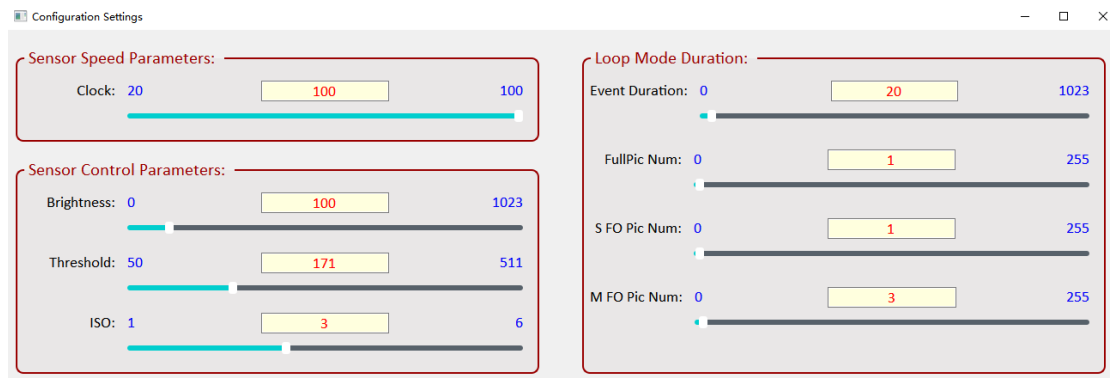


图 2-9-2

上图所示的所有参数都是控制 Sensor 的配置参数（即硬件参数），所以只有在实时显示时，修改这些参数才能起到修改 Sensor 输出数据的作用，Playback 时不需要调节这一组参数，下面将会介绍每个控制参数的意义。

1. Sensor Speed Parameters:

Clock: 调节 Sensor 的工作频率，默认值为 100MHz，该值越大，Sensor 的检测速度越快，对所有模式都有效。

2. Sensor Control Parameters:

Brightness: 调节图像的亮度，该值越大，图像会越暗。**备注：**仅对 Full-Picture 和 Event Intensity 模式有效。

Threshold: 调节触发 Event 的阈值，同样的条件下，该值越大，触发的 Event 的数量越小。**备注：**对 Event Off Pixel Timestamp Mode, Event In Pixel Timestamp Mode, Event Intensity 以及 Optical-Flow 模式都有效。

ISO: 调节图像对比度和动态范围，该值越大，对比度越低动态范围越高。如果该值被修改了，需要重新采集 FPN（如何采集 FPN，请参考 2.5 章节），如果每个 ISO 值对应的 FPN 文件事先都采集好了，那么本 SDK 会自动切换并使用新的 ISO 值对应的 FPN 文件。**备注：**仅对 Full-Picture 和 Event Intensity 模式有效。

3. Loop Mode Duration:

Event Duration: Loop 模式中，Sensor 每次在 Event Off Pixel Timestamp 模式持续的时间，模式值为 20，表示 Sensor 在该模式下会持续 20ms 时间，然后自动切换到下一个模式。

FullPic Num: Loop 模式中, Sensor 每次在 *Full-Picture* 模式持续的时间, 默认值为 1, 表示 Sensor 在该模式下会持续“产生 1 个图像帧”的时间, 然后自动切换到下一个模式。

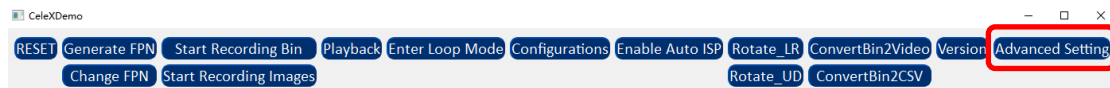
S FO Pic Num: Loop 模式中, Sensor 每次在 *Optical-Flow* 模式持续的时间, 默认值为 1, 表示 Sensor 在该模式下会持续“产生 1 个图像帧”的时间, 然后自动切换到下一个模式。

M FO Pic Num: Loop 模式中, Sensor 每次在 *Multi Read Optical-Flow* 模式持续的时间, 默认值为 3, 表示 Sensor 在该模式下会持续“产生 3 个图像帧”的时间, 然后自动切换到下一个模式。

CelePixel Confidential

2.10 Advanced Settings

点击“**Advanced Settings**”按钮，可以进行更多高级设置。在进行 Bin 文件录制操作时，可以选择关闭画面显示以保证数据的完整性（由于解析数据和显示图像非常耗时，可能导致数据的丢失）。用户也可以根据自己的需要，通过“**BinFile Time Duration**”来设置录制单个 Bin 文件的时长（单位为分钟），一旦 Bin 文件到达设定的时长，文件会自动保存并新建下一个新文件。



在进行 Bin 文件回放时，用户可以进行保存图片操作。通过设置“**SavePic Count Interval**”参数可以设置保存图片的间隔（如果间隔设置为 0，每一帧图像都会被保存；如果间隔设置成 2，则每隔 2 张会保存一张）。

此外，还开放了一个分辨率的设置。用户可以通过修改“**Resolution Parameter**”来关闭掉一些行的显示。该功能的详细介绍可见 [CeleX5_SDK_Reference](#) 文档。

