www.51camera.com.cn 51Camera微博公众号 51Camera微信公众号

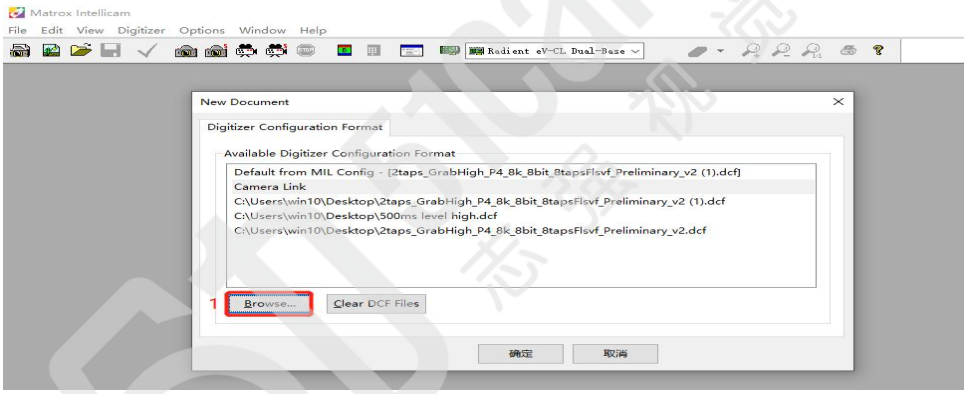
RAD EV 5M SF 采集卡

如何实现行帧触发返回最后不足一帧图像以及滤波

一：相机如何实现行帧触发高电平连续采集

1:单击  以启动新的 DCF。

2：点击 Browser 加载 DCF 配置文件（如下图）。



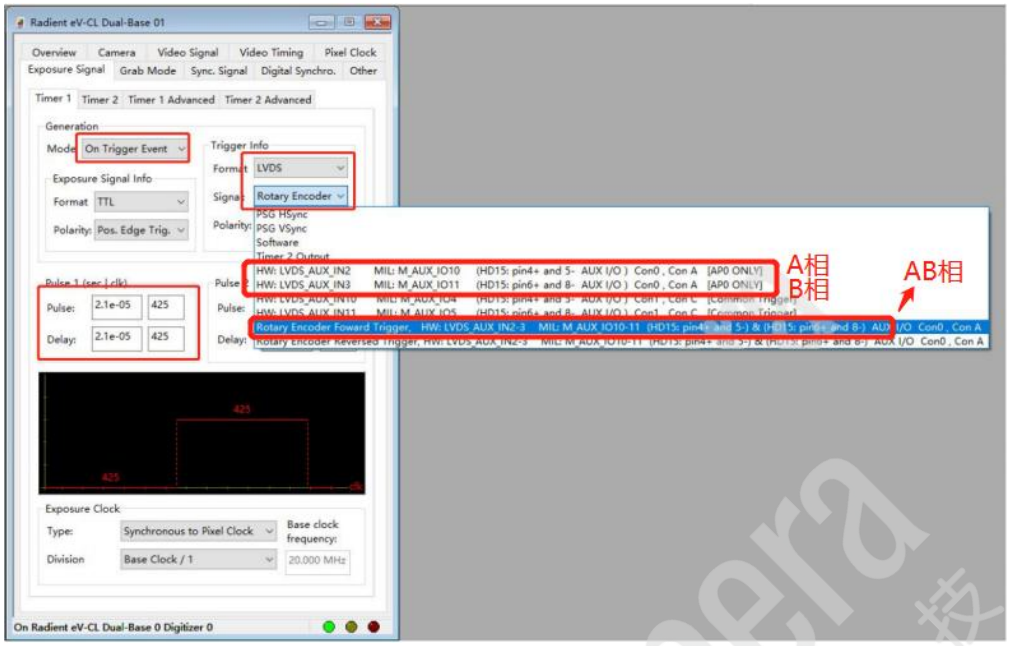
该 DCF 文详见如下网盘链接：https://pan.baidu.com/s/1waXuxBj3QvXHXljJjx1-FA 提取码：ZQSJ

3：DCF 文件进行如下设置

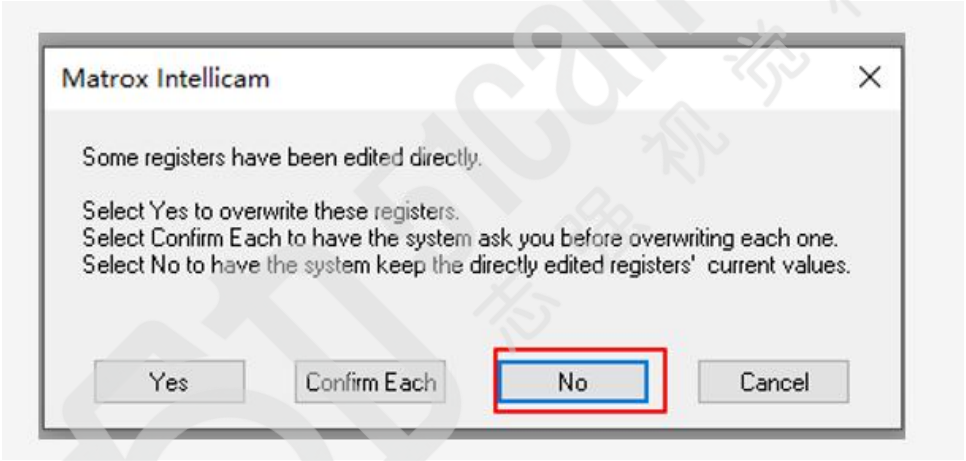
（1）Timer1 进行如下设置，Mode 可以设置不同的触发模式，如下图设置为 On Trigger Event 触发。

Pulse 1 （sec|clk）此处时间之合要大于相机的单行时间，设置的时间越大则帧率越低。

Trigger info 设置为编码器行触发。



（并且无论修改什么参数都会有如下弹出框，必须点击 No 该 DCF 才能继续正常使用）



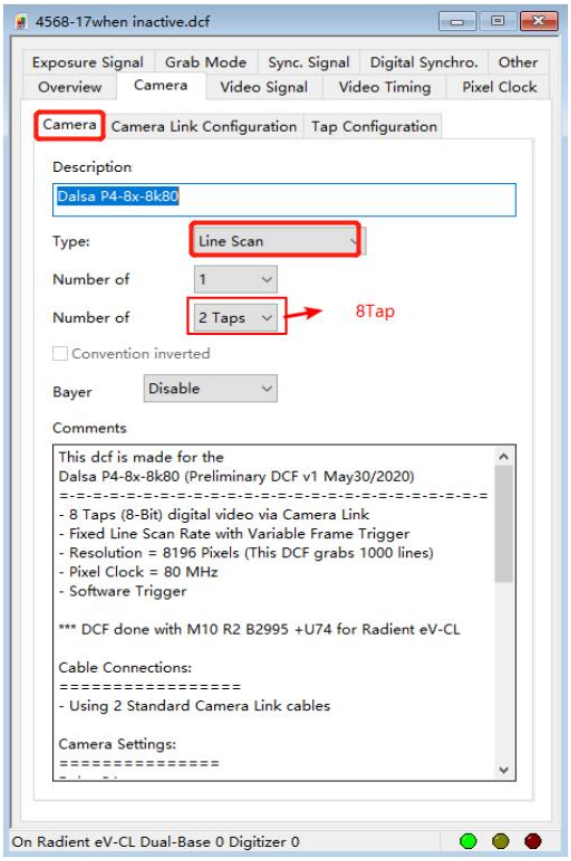
（2）Camera 参数进行如下设置：

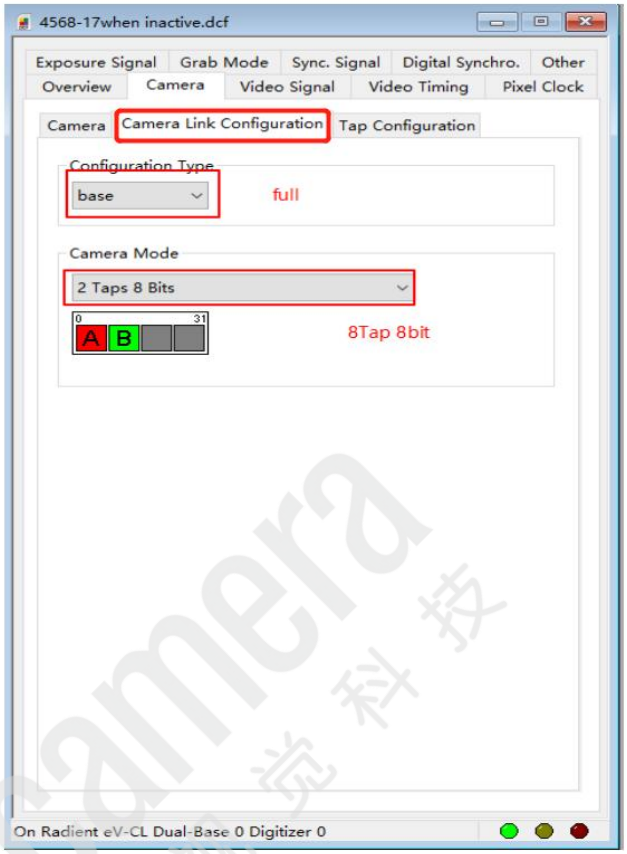
Type 可以选择相机采集方式，如下图设置为 Line Scan 模式。

Number of 可以选择 Taps 结构，

Configuration Type 可设置采集卡格式，选择 full 模式。

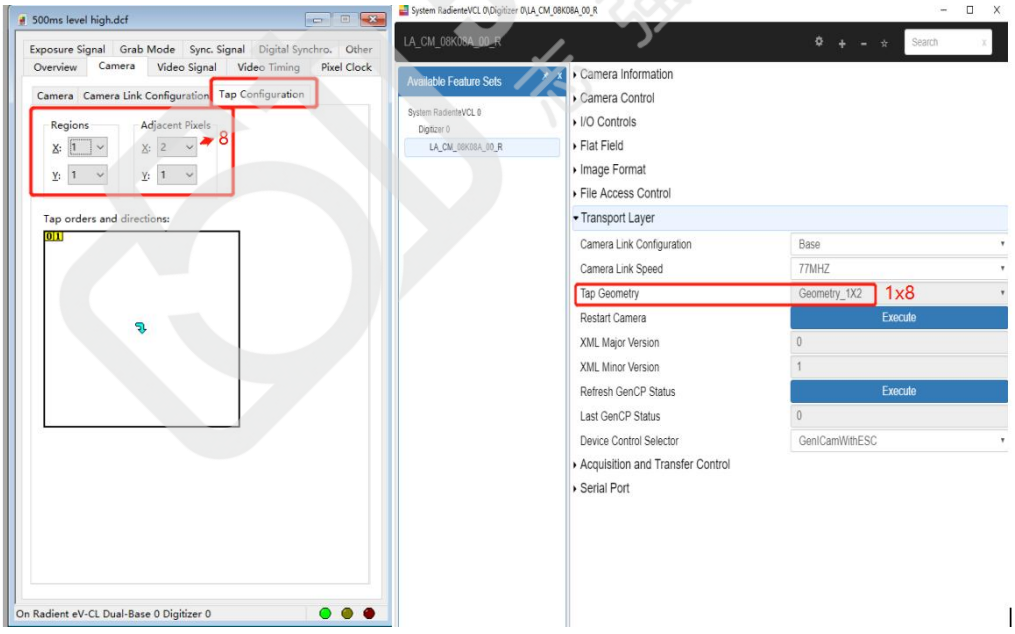
Camera Mode 可以选择相机模式，full 模式下请选择 8Taps 8Bits。





Tap Configuration 可配置采集通道，此处设置需跟相机端 Tap Geometry 设置一致（如下图）。

注：full 模式请根据红色字体修改参数。



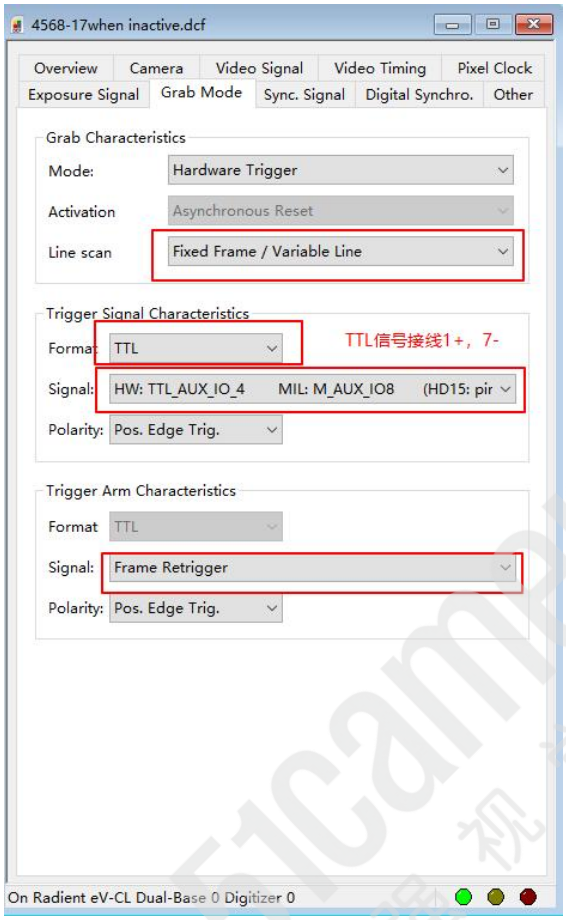
（3）帧触发参数进行如下设置。

Mode 可设置触发模式，此处可设置参数有，Continuous、Software Trigger、以及 Hardware Trigger，如下图设置为 Hardware Trigger。

Line scan 设置为 Fixed Frame/Variable Line。

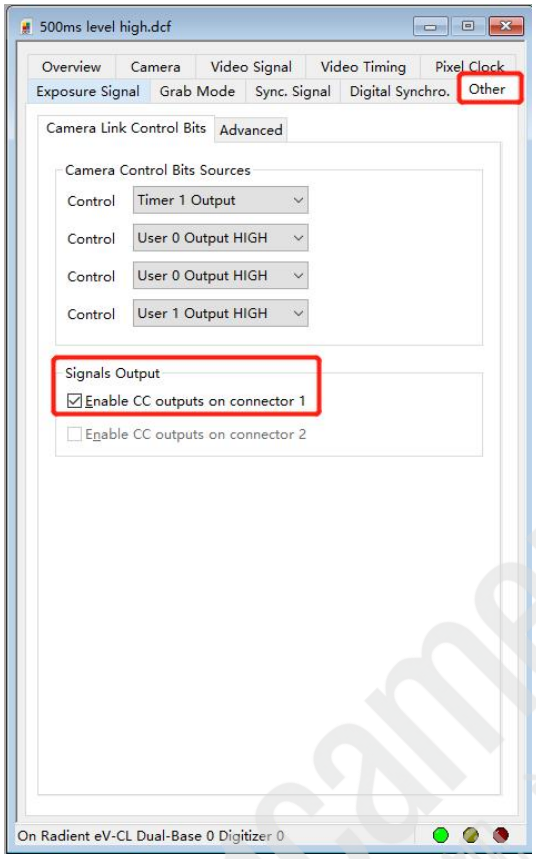
如果您使用的是 1+7-则 Format 设置为 TTL，Signal 设置为 pin1+7-（如下图）。

Signal 设置为 Frame Retrigger。



（4）Other 参数进行如下设置：

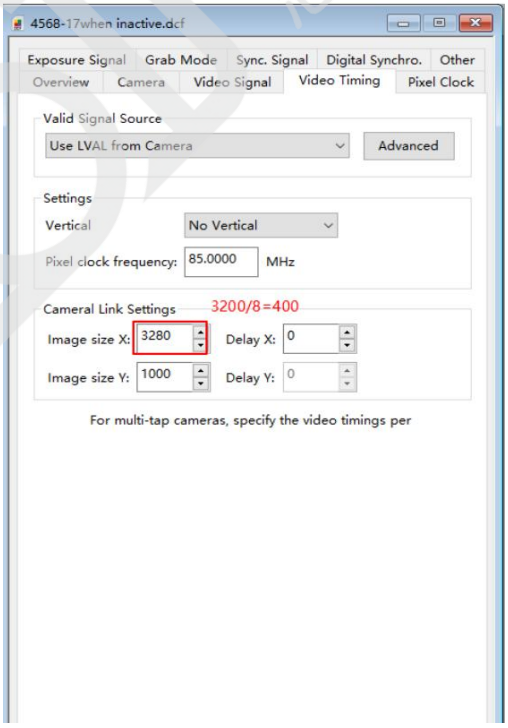
Signals Output 下属参数勾选 Enable CC outputs on connector1（如下图）。



（5）Video Timing 参数进行如下设置

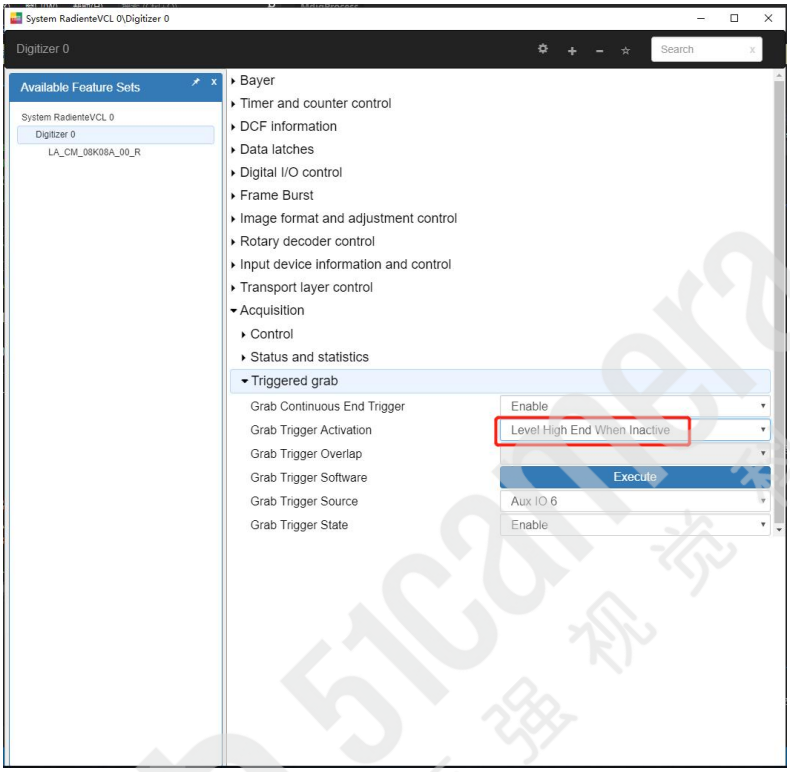
Pixel clock Frequency 可设置像素时钟频率，如下设置为 85Mhz。

Cameral Link Settings 可设置行宽以及行高，如下图为 3k 相机则 X 方向设置相机横向分辨率 3200/8Tap=400

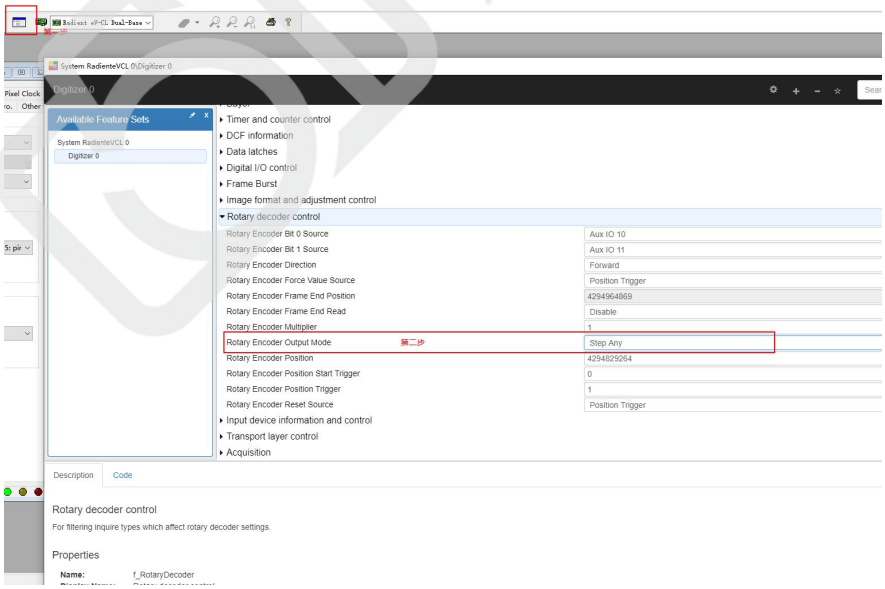


4：点击  打开 Feature Browser 点击 Digitizer 0 将触发信号设置为 Level High End When Inactive 触发。

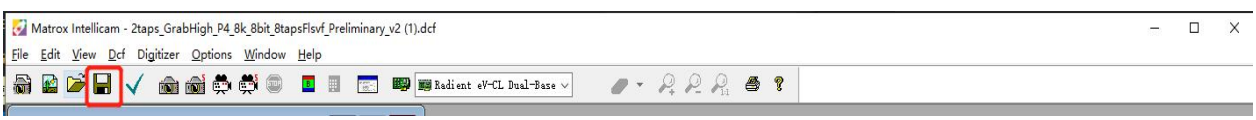
（注：每次 DCF 设置完毕 Feature Browser 都会自动关闭，需再次打开  Feature Browser 重新点击 Digitizer 0 将触发信号设置为 Level High End When Inactive 触发 ）（如下图）。



5：Rotary Encoder Output Mode 设置为 Step Any

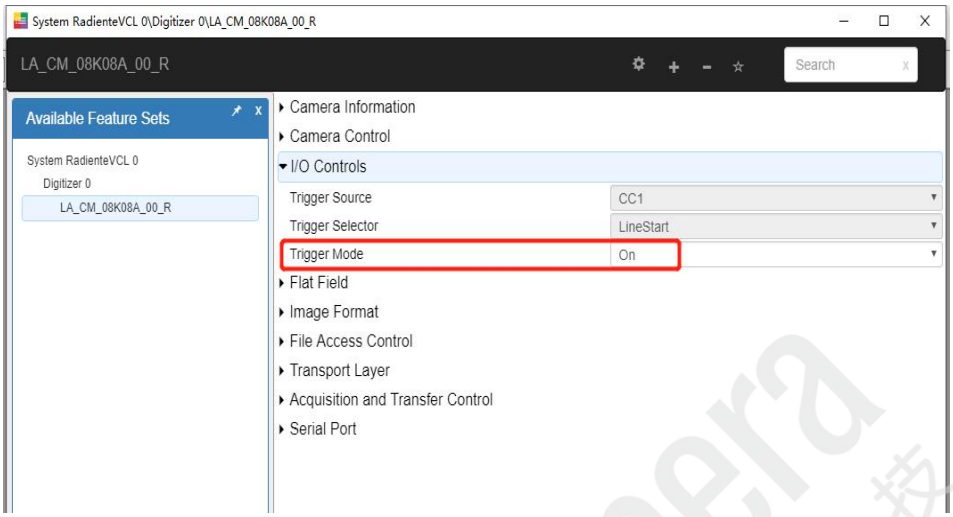


设置完成后点击 save 进行保存（如下图）。



6：相机端设置

Trigger Mode 设置为 ON（如下图）



二：如何设置返回最后不足一帧的图像

如何使用滤波参数滤除杂波首先更新补丁：

链接：*https://pan.baidu.com/s/19QMXuRJNiRlsMmOMhtxC3w* **提取码：ZQSJ**

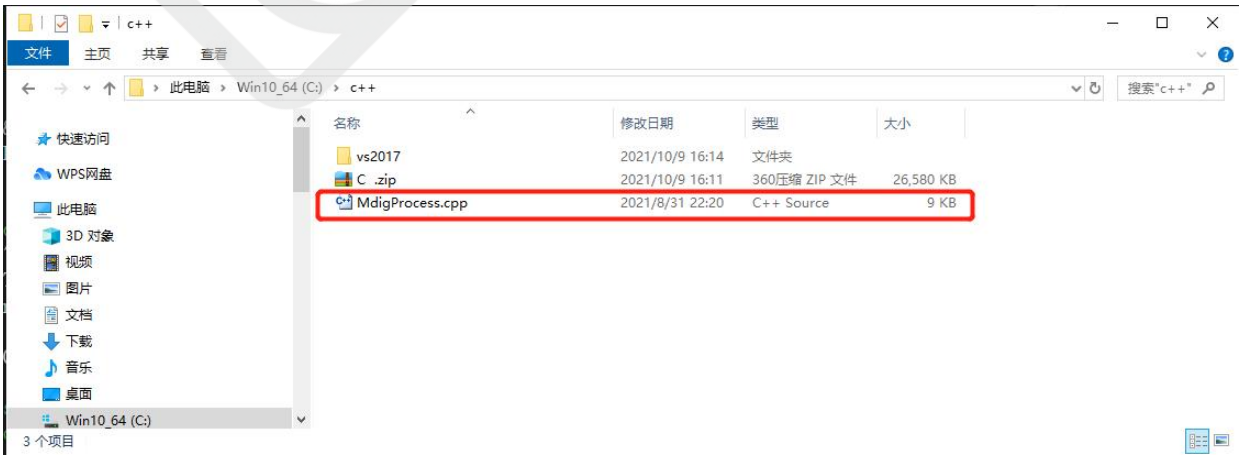
在代码里面添加如下指令：

MdigControl(MilDigitizer, M_IO_DEBOUNCE_TIME + M_AUX_IO8, 1000);（更新补丁之后才可以用，这个是设置 IO8 的滤波为 1us）

7：使用如下连接 C++工程进行采集

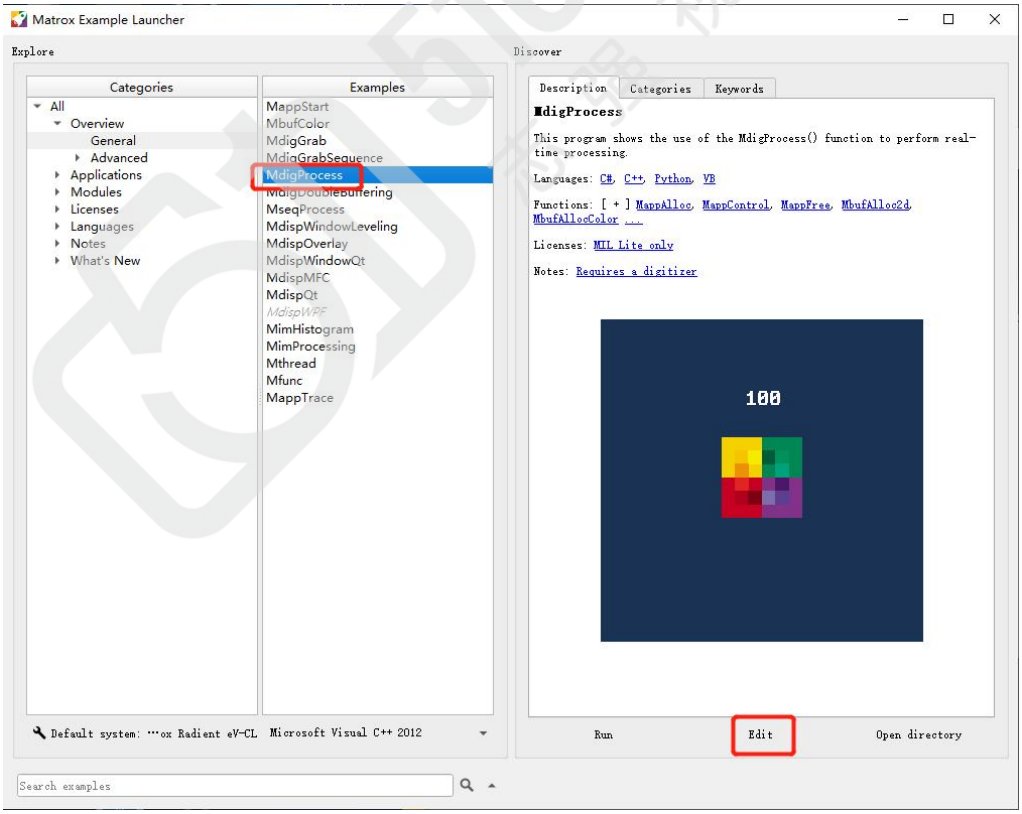
链接：*https://pan.baidu.com/s/1BLXppXX502JDdUfv4MWWug* **提取码：ZKFC。**

解压完成后用记事本打开 MdigProcess.CPP（如下图）。



打开 驱动软件。点击 Matrox Example Launcher（如下图）。

点击 MdigProcess，点击 Edit 打开此进程（如下图）。



将用记事本打开的 MdigProcess.CPP 代码全部复制到 Matrox Example Launcher 下的 MdigProcess 进程的 MdigProcess.cpp 中，将 Matrox Example Launcher 下的 MdigProcess 进程的 MdigProcess.cpp 中的代码全部替换。

需要在拼接的程序里面需要修改一下 M_GRAB_TRIGGER_SOURCE 为 M_AUX_IO8

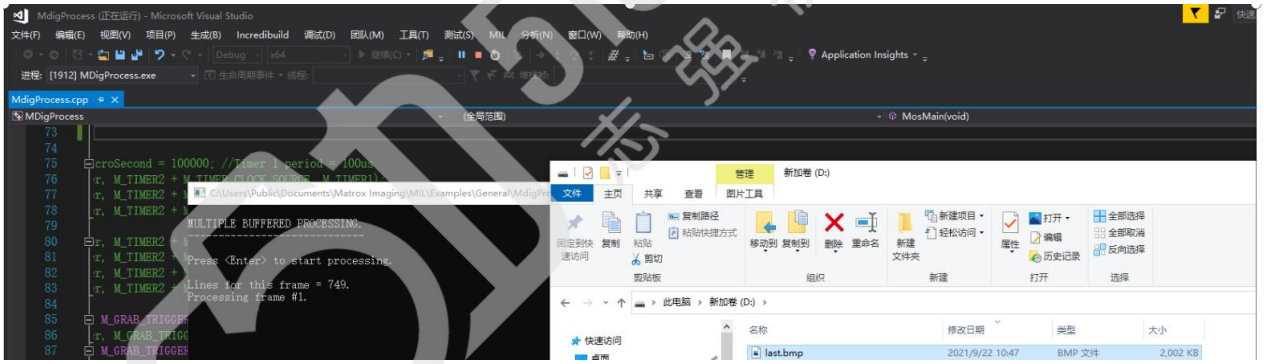
MdigControl(MilDigitizer, M_ROTARY_ENCODER_OUTPUT_MODE, M_STEP_ANY);

MdigControl(MilDigitizer, M_GRAB_TRIGGER_ACTIVATION, M_LEVEL_HIGH_END_WHEN_INACTIVE);

```
    MdigInquire(MilDigitizer, M_SIZE_Y, M_NULL),
    0 + M_UNSIGNED,
    M_IMAGE + M_GRAB + M_PROC + M_DISP,
    &MilImageDisp);
MbufClear(MilImageDisp, M_COLOR_BLACK);

//MIL_DOUBLE TiPeriodInMicroSecond = 100000; //Timer 1 period = 100us
//MdigControl(MilDigitizer, M_TIMER2 + M_TIMER_CLOCK_SOURCE, M_TIMER1);
//MdigControl(MilDigitizer, M_TIMER2 + M_TIMER_DURATION, TiPeriodInMicroSecond * NB_LINES_AFTER_SOFT_TRG);
//MdigControl(MilDigitizer, M_TIMER2 + M_TIMER_DELAY, 1);

//MdigControl(MilDigitizer, M_TIMER2 + M_TIMER_TRIGGER_ACTIVATION, M_EDGE_RISING); // Specifies that a timer trigger will be generated upon a low-to-high signal transition.
//MdigControl(MilDigitizer, M_TIMER2 + M_TIMER_OUTPUT_INVERTER, M_DISABLE);
//MdigControl(MilDigitizer, M_TIMER2 + M_TIMER_TRIGGER_SOURCE, M_SOFTWARE);   // M_SOFTWARE   M_CONTINUOUS  Specifies to use the output of the default rotary decoder (set with M_ROTARY_ENCODER_OUTPUT_MODE ) as the trigger source.
//MdigControl(MilDigitizer, M_TIMER2 + M_TIMER_STATE, M_ENABLE); // Specifies that the timer is enabled.

MdigControl(MilDigitizer, M_GRAB_TRIGGER_SOURCE, M_AUX_IO8); //M_AUX_IO8 M_TIMER2
//MdigControl(MilDigitizer, M_GRAB_TRIGGER_SOURCE, M_TIMER2); //M_AUX_IO8 M_TIMER2
MdigControl(MilDigitizer, M_GRAB_TRIGGER_ACTIVATION, M_LEVEL_HIGH_END_WHEN_INACTIVE); // M_LEVEL_HIGH_END_WHEN_INACTIVE);
//MdigControl(MilDigitizer,M_GRAB_TRIGGER_ACTIVATION, M_LEVEL_HIGH);
MdigControl(MilDigitizer, M_GRAB_TRIGGER_STATE, M_ENABLE);


/* Display the image buffer. */
//MdispSelect(MilDisplay, MilImageDisp);

/* Print a message. */
MosPrintf(MIL_TEXT("\nMULTIPLE BUFFERED PROCESSING.\n"));
MosPrintf(MIL_TEXT("----------------------------\n\n"));
MosPrintf(MIL_TEXT("Press <Enter> to start processing.\n\n"));

///* Grab continuously on the display and wait for a key press. */
//MdigGrabContinuous(MilDigitizer, MilImageDisp);
//MosGetch();

///* Halt continuous grab. */
//MdigHalt(MilDigitizer);
```

具体代码请参考该工程中 MdigProcess.cpp 文件。使用该程序进行采集当高电平结束时会返回最后一张不后一张不足一帧的图片的有效行数，然后对该图片进行有效行提取即可得到最后一张不足一帧的有效图片。可根据实际开发需要参考 MdigProcess.cpp 进行修改。

例如下图，是将内部行频设置为 10K，高电平持续时间设置为 0.075S，从图可以看出反馈得到的有效采集行数为 749 行，有一行的误差是因为信号的精准度的原因。再从保存的最后一张图片中提取有效行数。



三：返回最后不足一帧的图像及图像拼接功能的代码

如您需要自己做拼接，请使用标题“二、如何设置返回最后不足一帧的图像” 中提到的代码即可。如需要我们提供拼接请自行下载代码

链接：https://pan.baidu.com/s/1trjCbMe9nSlZqVit4EsgXw

提取码：parb

联系我们： 北京志强视觉科技发展有限公司
电话：+86 (010) 80482120
传真：+86 (010) 80483130
邮箱：51camera@51camera.com.cn
网址：www.51camera.com.cn