# TOF模块

tof\_noAlgo\_handler.c

// byte3 | byte2 | byte1 | byte0

// unused|mode|res|fps

cur\_all\_config =sonytof\_default\_allconfig //setting mode(m0~m1) of deal ,vga,fps

#ifdef \_USE\_SONY\_TOF\_DATAPATH

tof\_flash\_calib\_data\_check()；获取flashdata

SONY\_TOF\_DATA\_PATH\_DDR DDR\_Send ；//tof参数全局变量，bss段，

#ifndef USE\_SONYTOF\_DYNAMIC\_LOAD

tof\_init\_shave\_cfg();//动态调整使用shave核数量

sonytog\_init\_all\_lut(sonytof\_cfg);

#endif

#endif

tof\_noAlgo\_cameraCommunicationInit(void)// 摄像头GPIO配置

#ifdef TOF\_EEPROM\_READ\_ENABLE

//used eeprom load flash p

#endif

// xxxIIC初始化. TOF camera used iic load setting,and mipi out frame.

tofxxxiicDevices

//xxxinitializeSensors tof mipi instance 对象初始化. camera

tof\_init shave\_cfg();//dynamic load shave

while(fpKeepRuning)//external thread loop

{

frameSpec camFrameSpec = {0}//infonation for frame camera

// setting global frame buffer sptr

// setting laser value

// FRAME\_PUMP\_START , mipiread camera。

while(fpKeepRuning) inside thread loop

{

SonyTof\_CheckAllConfig();//

}

}

//’tof\_postproc.co

RawToIQHandler(SonyTofModeDetail\_t \* xxx,input,output)

{

if(rawToIQMode == 2) // single vga/qvga

{

} else if(rawToIQMode == 1) // dual vga/qvga

{

}

// what is single or dual vga/qual -> single or dual frequency

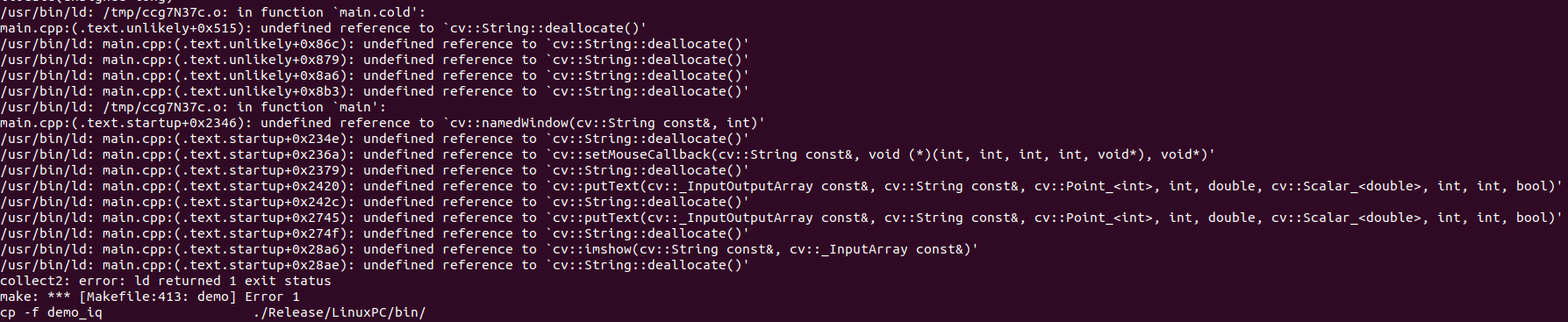
}

Carmen 258 初始化，

遇到问题：

问题1:  
 ./bin/demo\_iq: undefined symbol: \_ZN2cv3MatC1Ev

问题2：



最终解决方案，在物理机上重新安装18.04 然后解决。

TOF代码调试

- 帧率

- 分辨率

- 曝光参数

- IQ模式数据多大。

- RAW数据多大。

- depth 深度数据多大。

- 新的对象申请实例子。

- 各个参数的含义。

- 函数结构。

### tofflash数据

tof\_calib\_data // calibration parameter for tof camera,1k size

tof\_sn\_data // sn numer for camera

typedef enum frameTypes

{

YUV422i, // interleaved 8 bit

YUV420p, // planar 8 bit

YUV422p,// planar 8 bit

YUV400P,// 8-bit greyscale

RGBA8888,// RGB interleave stored in 32 bit word

RGB888,//planar 8 bit RGB data

LUT2 , // 1 bit per pixel,Lookuo table(used for graphics layers)

LUT4 , // 2 bit per pixel,Lookuo table(used for graphics layers)

LUT16 , // 4 bit per pixel,Lookuo table(used for graphics layers)

RAW16 , // save any raw type(8,10,12bit) on 16 bits

RAW8,

YUV444i,

NV12,

NONE

}

typedef struc frameSpecs

{

frameType type;

unsigned int height;

unsigned int width;

unsigned int stride;//defined as distance in bytes from pix(y,x) to pix(y+1,x)

unsigned int bytesPP;//bytes per pixel (for LUT types set this to 1)

}frameSpec;

typedef struc frameElements

{

frameSpec spec;

unsigned char\* p1 : // Pointer to first image plane

unsigned char\* p2 : // Pointer to second image plane

unsigned char\* p3 : // Pointer to third image plane

}

typedef struct FramePumpBuffer

{

frameBuffer fbMem；//memory frame buffer

frameBuffer fb;//normal frame buffer

uint64\_t timestamp;

struct timespec timestamp;

uint32\_t frameCount;

uint32\_t exposureNs;

float gain;

uint32\_t cam\_id;

}

struc FramePumbBuffer fpbtof\_noAlgo={0};//global varable