PRO2/TITAN开发说明书

通信协议

```
1.APP - PRO service: http

2.PRO service - camera servie:fifo
/data/ins_fifo_to_server : client to server
/data/ins_fifo_to_client : server to client
/data/ins_fifo_to_client_a : server to client
/data/ins_fifo_to_server_father : client to server father process
/data/ins_fifo_to_client_father :server father process to client

3.camera service - module: usb

```

客户端接口

camera._reset

```
request:
{
“name”:"camera._reset"
}
respone:
{
"name":"camera._reset",
"state":"done",
"error":{"code":int, "description":string}
}
```

camera._queryState

```
request:
{
"name":"camera._queryState"
}
respone:
{
```

```
"name":"camera._queryState",
"state":"done"
"results":{
  "state":string,
  "version":string,
  "moduleVersion":string,
  "origin":{
      "width":int,
      "height":int,
      "framerate":int,
      "bitrate":int,
      "mime":string,
      "saveOrigin":bool
      },
    "preview":{
        "width":int,
        "height":int,
        "framerate":int,
        "bitrate":int,
        "mime":string
        },
    "live":{
        "width":int,
        "height":int,
        "framerate":int,
        "bitrate":int,
        "mime":string,
        "url":string,
        "timePast":int,
        "timeLeft":int,
        "liveOnHdmi":bool,
        "liveRecording":bool
        },
    "record":{
        "width":int,
        "height":int,
        "framerate":int,
        "bitrate":int,
        "mime":string,
        "url":string,
        "timePast":int,
        "timeLeft":int
        },
     "audio":{
         "mime":string,
         "sampleFormat":string,
         "channelLayout":string,
         "samplerate":int,
         "bitrate":int
         }
    }
```

```
}

tips:

1.state
idle/record/preview/live/pic_shoot/pic_process/storage_speed_test/
gyro_calibration/calibration/stitching_box/blc_calibration
qrscan/stop_rec/stop_live/audio_capture

2.mime
video mime:h264/h265
audio mime:aac

3.record
如果只录原始流的时候，record obj就只有url

4.duration:单位秒

5.liveOnHdmi:如果是直播到hdmi那么，live中的width/height/url参数都没有

6.liveRecording:直播过程中是否存原始流
```

camera._setOffset

```
request:
{
"name": "camera._setOffset",
"parameters":{
    "offset_pano_4_3":string,
    "offset_pano_16_9":string,
    "offset_3d_left":string,
    "offset_3d_right":string
    "factory_setting":bool
 }
}
respone:
{
"name":"camera._setOffset",
"state":"done",
"error":{"code":int, "description":string}
}
```

camera._setGyroCalibrationResult

```
request:
{
"name": "camera._setGyroCalibrationResult",
```

```
"parameters":{"result":string}
}
respone:
{
"name":"camera._setGyroCalibrationResult",
"state":"done",
"error":{"code":int, "description":string}
}
```

camera._getOffset

```
request:
{
"name": "camera._getOffset"
}
respone:
{
"name":"camera._getOffset",
"state":"done",
"results":{
    "offset_pano_4_3":string,
    "offset_pano_16_9":string,
    "offset_3d_left":string,
    "offset_3d_right":string
    }
}
```

camera._setOptions

```
１.单个设置
request:
{
"name": "camera._setOptions",
"parameters":{“property”:string, "value":int}
}
2.单个设置：如果property有多个参数的话，value就是一个对象
request:
{
"name": "camera._setOptions",
"parameters":{“property”:string, "value"{xxx}}
}
3.批量设置：
request:
{
"name": "camera._setOptions",
"parameters":[{“property”:string, "value":int}] //数组
```

```
}
单个设置响应：
respone:
{
"name":"camera._setOptions",
“state”:"done",
"error":{"code":int, "description":string}
}
批量设置响应：
respone:
{
"name":"camera._setOptions",
“state”:"done",  //如果state不为done,每一个的错误通过result数组返回
"error":{"code":int, "description":string},
results":{"detail":[{"property":string,"code":int, "description":string}]
}
tips:
"aaa_mode": 0:手动 1：自动 2：独立 3：快门优先 4：iso优先
“wb”:0-14
"iso_value":0-9
"shutter_value":0-43
"long_shutter":1-60(s)
"brightness":-255:255
"contrast":0:255
"saturation":0:255
"hue":-15:15
"sharpness":0:6
"ev_bias":-255:255
"ae_meter":0:2
"iso_cap":int
"stabilization":1-eanble 0-disable
"flicker":0-pal 1-ntsc
"logMode": "value":{"mode":int, "effect":int}
"fanless": 0/1        //0:启动风扇 1:关闭风扇
"panoAudio":0/1   //0:拼接流不存声音 1:拼接流存非全景声 2:拼接流存全景声
"logo":0/1    //0:不打logo 1:打logo
"audio_gain":1-127
"video_fragment": 0/1 //0:视频不分段 1：视频分段
“depthMap”: string //深度图数据
```

camera._getOptions

```
request:
{
"name": "camera._getOptions",
"parameters":{“property”:string}
}
respone:
{
"name":"camera._getOptions",
```

```
“state”:"done",
"results":{“value”:string}
}
说明:
可以获取的property:
1.moduleHwVersion
2.supportedFunctions:
"liveFormat":["hls","rtsp","rtmp"],
"projection":["flat","cube"],
"gammaCurve":bool,
"pmode":bool,
"p8k5fps":bool
"audioGain":bool
“depthMap”:bool 是否支持深度图功能
“blc”:bool 是否支持blc矫正
3.depthMap:bool 是否有深度图信息
4.blc_state:int
0:blc没有矫正过
1:blc矫正过
```

camera._update_gamma_curve

```
request:
{
"name": "camera._update_gamma_curve",
"parameters":{"data":string}
}
respone:
{
"name":"camera._update_gamma_curve",
“state”:string,
}
说明:
1.gamma曲线的数据base64编码
2.重置曲线:data为空:data为空
```

camera._startPreview

```
request:
{
"name": "camera._startPreview",
"parameters":{
    "origin":
{"mime":string,"width":int,"height":int,"framerate":int,"bitrate":int,"log
Mode":int},
    "stiching":
{"mode":string,"mime":string,"width":int,"height":int,"framerate":int,"bit
rate":int},
```

```
       },
    “stabilization”:bool,
    “imageProperty”:{
    "aaa_mode":int,
    “wb”:int,
    "iso_value":int,
    "shutter_value":int,
    "brightness":int,
    "contrast":int,
    "saturation":int,
    "hue":int,
    "sharpness":int,
    "ev_bias":int,
    "ae_meter":int,
    "dig_effect":int,
    "flicker":int
    }
}
respone:
{
"name":"camera._startPreview",
“state”:"done"
"results":{“_previewUrl”:string }
}
tips:
mime:h264/h265
mode:pano/3d_top_left/3d_top_right
bitrate:kbits/s
logMode: 0-关闭log模式 1-开启log模式
```

camera._restartPreview

```
request:
{
"name": "camera._restartPreview",
"parameters":{
 "origin":
{"mime":string,"width":int,"height":int,"framerate":int,"bitrate":int,"log
Mode":int},
 "stiching":
{"mode":string,"mime":string,"width":int,"height":int,"framerate":int,"bit
rate":int},
       },
 “stabilization”:bool,
 “imageProperty”:{
 "aaa_mode":int,
 “wb”:int,
 "iso_value":int,
 "shutter_value":int,
```

```
 "brightness":int,
 "contrast":int,
 "saturation":int,
 "hue":int,
 "sharpness":int,
 "ev_bias":int,
 "ae_meter":int,
 "dig_effect":int,
 "flicker":int
 }
}
respone:
{
"name":"camera._restartPreview",
“state”:"done"
"results":{“_previewUrl”:string }
}
tips:
mime:h264/h265
mode:pano/3d_top_left/3d_top_right
bitrate:kbits/s
logMode: 0-关闭log模式 1-开启log模式
```

camerad._queryLeftInfo

```
request:
{
 "name":"camera._queryLeftInfo",
 "param": {"name":"camera._takePicture","parameters":{"delay":0,"origin":{"height":3000,"mime":"jpeg","saveOrigin":true,"storage_loc":0,"width":4000},"stiching":{"algorithm":"opticalFlow","height":7680,"mime":"jpeg","mode":"3d_top_left","width":7680}}}
}
param:需要查询的挡位参数

respone:
{
  "name":"camera._queryLeftInfo",
  "state":"done",
  "left":int
}
```

camera._shutdown

```
{
```

```
 "name":"camera._shutdown"
}
```

camera._stopPreview

```
request:
{
"name":"camera._stopPreview"
}
respone:
{
"name":"camera._stopPreview",
“state”:"done",
"error":{"code":int, "description":string}
}
```

camera._preview_finish_

```
indication:
{
"name":"camera._preview_finish_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}
```

camera._startRecording

```
{
request:
{
"name": "camera._startRecording",
"parameters":{
"origin":
{"mime":string,"width":int,"height":int,"framerate":int,"bitrate":int,"log
Mode":int,"hdr":bool,"saveOrigin":bool,"liveUrl":string,
"storage_loc":int, //0:存在大卡 1: 存在6张小卡
"hdr":bool
},
"stiching":
{"mode":sting,"mime":string,"width":int,"height":int,"framerate":int,"bitr
ate":int,"map":string},
"audio":
{"mime":string,"sampleFormat":string,"channelLayout":string,samplerate:int
```

```
,bitrate:int}
       },
"timelapse":{"enable":bool, "interval":int},
"duration":int,
"fileOverride":bool,
"storagePath":string,
“stabilization”:bool,
"storageSpeedTest":bool,
}}
respone:
{
"name":"camera._startRecording",
“state”:"done",
"error":{"code":int, "description":string}
}
tips:
video mime:h264/h265
audio mime:aac
mode:pano/3d_top_left/3d_top_right
sampleFormat:s16/s32
channelLayout:mono/stereo
bitrate:kbits/s
map:cube/flat, default is flat
interval:ms
storageSpeedTest:表明录像的目的是测速卡速
logMode: 0-关闭log模式 1-开启log模式
liveUrl:原始流推流，存在liveUrl的时候，saveOrigin必须为false，否则不生效
当liveUrl = "rtmp://127.0.0.1/live"的时候，六路流的直播直播地址为：
rtmp://127.0.0.1/live/originN  (N=[0,1,2,3,4,5])
```

camera._stopRecording

```
request:
{
"name":"camera._stopRecording"
}
respone:
{
"name":"camera._stopRecording",
“state”:"done",
"error":{"code":int, "description":string}
}
```

camera._record_finish_

```
indication:
{
"name":"camera._record_finish_",
```

```
"parameters":{
 "state":string,
 "error":{"code":int, "description":string}
}
}
tips:
state is "done", if record finished successfully,
otherwise, state is "error", and error will be described in error obj
```

camera._startLive

```
request:
{
"name": "camera._startLive",
"parameters":{
"origin":
{"mime":string,"width":int,"height":int,"framerate":int,"bitrate":int，“logMode”:int, “saveOrigin”:bool,},
"stiching":
{"mode":sting,"mime":string,"width":int,"height":int,"framerate":int,"bitrate":int, map:string, "_liveUrl":string, “liveOnHdmi":bool, "fileSave":bool},
"audio":
{"mime":string,"sampleFormat":string,"channelLayout":string,samplerate:int,bitrate:int}
       },
"autoConnect":{"enable":bool, "interval":int, "count":int}
"storagePath":string,
“stabilization”:bool
}
respone:
{
"name":"camera._startLive",
“state”:"done",
"results":{"_liveUrl":string}
"error":{"code":int, "description":string}
}
tips:
video mime:h264/h265
audio mime:aac
mode:pano/3d_top_left/3d_top_right
sampleFormat:s16/s32
channelLayout:mono/stereo
bitrate:kbits/s
map:cube/flat, default is flat
logMode: 0-关闭log模式 1-开启log模式
liveOnHdmi:如果hdmi直播的话，stiching里的 width/height/framerate/bitrate/_liveurl就不用填了
saveOrigin：直播的时候是否存原始流
```

```
fileSave：直播的时候是否存拼接流
autoConnect:直播重连参数，enable：是否重连， interval：重连间隔（单位ms），count：重连次数
```

camera._stopLive

```
request:
{
"name":"camera._stopLive"
}
respone:
{
"name":"camera._stopLive",
“state”:"done",
"error":{"code":int, "description":string}
}
```

camera._live_rec_finish_

```
indication:
{
"name":"camera._live_rec_finish_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}
```

camera._live_finish_

```
indication:
{
"name":"camera._live_finish_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}
```

camera._takePicture

```
request:
{
  "name": "camera._takePicture",
  "parameters":{
```

```
    "origin":{
      "mime":string,
      "width":int,
      "height":int,
      "saveOrigin":bool
      "storage_loc":int //0:存在大卡 1：存在6张小卡
    },
    "stiching":{
      "mode":string,
      "mime":string,
      "width":int,
      "height":int,
      "map":string,
      "algorithm":string
    },
    "burst":{"enable":bool, "count":int}
    "hdr":{"enable":bool, "count":int, "min_ev":int, "max_ev":int}
    "bracket":{"enable":bool, "count":int, "min_ev":int, "max_ev":int}
    "delay":int,
    "storagePath":string
  }
}
respone：
{
  "name": "camera._takePicture",
  "state": "done",
  "error":{"code":int, "description":string}
}
tips:
picture mime:jpeg/raw/raw+jpeg(同时存raw和jpeg)
mode:pano/3d_top_left/3d_top_right
delay:in second
map:cube/flat, default is flat
algorithm:normal/opticalFlow, default is normal
burst: count指拍摄张数
hdr: cout指拍摄张数（现在只支持3）

```

camera._pic_finish_

```
indication:
{
"name":"camera._pic_finish_",
"parameters":{
 "state":string,
 "results":{“_picUrl”:string},
 "error":{"code":int, "description":string}
}
}
```

camera._pic_origin_finish_

```
indication:
{
"name":"camera._pic_origin_finish_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}
```

camera._setImageParam

```
request:
{
"name": "camera._setImageParam",
"parameters":{“property”:string, "value":int}
}
respone:
{
"name":"camera._setImageParam",
“state”:"done",
"error":{"code":int, "description":string}
}
tips:
"aaa_mode": 1:auto, 0:manual
“wb”:0-14
"iso_value":0-9
"shutter_value":0-43
"brightness":-255:255
"contrast":0:255
"saturation":0:255
"hue":-15:15
"sharpness":0:6
"ev_bias":-255:255
"ae_meter":0:2
"dig_effect":0:8
"flicker":0:2
```

camera._getImageParam

```
request:
{
"name": "camera._getImageParam"
}
respone:
{
```

```
"name":"camera._getImageParam",
“state”:"done"
"results":{
    "aaa_mode":int,
    “wb”:int,
    "iso_value":int,
    "shutter_value":int,
    "brightness":int,
    "contrast":int,
    "saturation":int,
    "hue":int,
    "sharpness":int,
    "ev_bias":int,
    "ae_meter":int,
    "dig_effect":int,
    "flicker":int
    }
}
```

camera._connect

```
request:
{
  "name": "camera._connect"
  "parameters":{
    //old
    "date_time":"MMDDhhmm[[CC]YY][.ss]"

    //new 170728
    "hw_time":"MMDDhhmm[[CC]YY][.ss]"
    "time_zone":":str//( eg:  GMT+08:00/GMT-08:00)
}

respone:
{
  "name":"camera._connect",
  "state":"done"
  "machine":"pro2"
  "results":{
    "Fingerprint":string
    "_cam_state":int
    "url_list":{
      "_liveUrl": string,
      "_previewUrl": string,
      "_recordUrl": string
    }
    "last_info":{
      "state":string,
      "version":string,
      "moduleVersion":string,
```

```
      "origin":{
        "width":int,
        "height":int,
        "framerate":int,
        "bitrate":int,
        "mime":string,
        "saveOrigin":bool
    },
"preview":{"width":int, "height":int, "framerate":int, "bitrate":int,
"mime":string},
"live":{"width":int, "height":int, "framerate":int, "bitrate":int,
"mime":string,"url":string, “timePast”:int, "timeLeft":int,
"liveOnHdmi":bool},
"record":{"width":int, "height":int, "framerate":int, "bitrate":int,
"mime":string, “url”:string,“timePast”:int, "timeLeft":int}
“audio”{"mime":string,"sampleFormat":string,"channelLayout":string,"sample
rate":int,"bitrate":int}
"sys_info":{"k_v": "6.0", "r_v":"'000",
"p_v":"'1.00May 192017","'sn":"'xxxxxx", "h_v":"V0.9.0_2017.5.16"}
}
}
ps:
1 you can get "last_info" and "url_list" only when _cam_state is
not 0x00(STATE_IDLE)
(
_cam_state:
详见相机状态7(文档末尾）
)

2 last_info is according from _queryState results above provided by colin
3 date_time and hw_time is string(format is "MMDDhhmm[[CC]YY][.ss]"(月日时
分年.秒))
(091713272014.30 表示 Wed Sep 17 13:27:30 GMT 2014),but hw_time is utc
time(not adding time_zone)
time_zone is like  GMT+08:00/GMT-08:00
4
"k_v": kernel version(not used yet)
"r_v":"' rom version
"p_v":pro_service version
"h_v":http version
5
"machine": pro2/titan

```

camera._setWifiConfig

```
request:
{
"name": "camera._setWifiConfig"
```

```
"parameters":{
    "ssid":'insta360_pro', //长度:0-64
    "pwd":"7777777",//长度:8-64
    "open":1 // 1 代表设置密码,0代表不设置密码(暂时不支持)
       }
}
respone:
{
"name":"camera._setWifiConfig",
“state”:"done"
}

//_getStoragePath is added by colin, which not used in web service, that could got in poll info 2017.2.6
```

camera._changeStoragePath

```
request:
{
“name”:"camera._changeStoragePath",
"parameters":{"path":string}
}
respone:
{
"name":"camera._changeStoragePath",
“state”:"done",
"error":{"code":int, "description":string}
}
```

camera._queryStorage

```
request:
{
"name":"camera._queryStorage"
}
respone:
{
"name":"camera._queryStorage",
“state”:"done",
"error":{"code":int, "description":string}
"results":
    {
    "storagePath":string //nvidia上的存储路径
    "module":[{"index":int, "storage_total":int, "storage_left":int}] //6个模组上的存储容量
    }
}
```

camera._storage_state_

```
indication:
{
"name":"camera._storage_state_",
"parameters":
 {
 "module":{"index":int, "storage_total":int, "storage_left":int}
 }
}
```

camera._queryGpsStatus

```
request:
{
“name”:"camera._queryGpsStatus"
}
respone:
{
"name":"camera._queryGpsStatus",
“state”:"done",
"error":{"code":int, "description":string}
"results":
 {
   "fix_type":int //0/1:无定位 2：二维定位 3：三维定位
   "sv_status":{
     "sv_num":int,"status":[{"prn":int, "snr":float,"elevation":float,
"azimuth":float}]
     //snr:信噪比 elevation：仰角 azimuth：方位角
     //prn:Pseudo-random number
   }
 }
}
```

camera._change_module_usb_mode

```
request:
{
"name":"camera._change_module_usb_mode",
"parameters": {"mode":int}
}
respone:
{
"name":"camera._change_module_usb_mode",
“state”:"done",
```

```
"error":{"code":int, "description":string}
}

说明:
1.mode: 0:非storage模式 1:storage模式
2.storage模式的时候，camerad处于module_storage状态，此状态下不能做任何模组操作

```

camera._calibration

```
request:
{
"name":"camera._calibration"
"parameters":{"mode":string, "delay":int}
}
respone:
{
"name":"camera._calibration",
“state”:"done",
"error":{"code":int, "description":string}
}
Async result:
{
"name":"camera._calibration_finish_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}

tips:
mode:pano/3d
```

camera._getFWVersion

```
request:
{
"name":"camera._getFWVersion"
"parameters":{"index":int}
}
respone:
{
"name":"camera._getFWVersion",
“state”:"done",
"error":{"code":int, "description":string}
}
tips:
index:0~5, -1 means all, if index is not exist, default is master
```

camera._upgradeFw

```
request:
{
"name":"camera._upgradeFw"
"parameters":{"path":string}
}
respone:
{
"name":"camera._upgradeFw",
“state”:"done",
"error":{"code":int, "description":string}
}
```

camera._startQRCodeScan

```
request:
{
"name":"camera._startQRCodeScan"
}
respone:
{
"name":"camera._startQRCodeScan",
“state”:"done",
"error":{"code":int, "description":string}
}
tips:
state:QR code scan just can be done under idle state
```

camera._stopQRCodeScan

```
request:
{
"name":"camera._stopQRCodeScan"
}
respone:
{
"name":"camera._stopQRCodeScan",
“state”:"done",
"error":{"code":int, "description":string}
}
```

camera._qr_scan_finish_

```
{
```

```
"name":"camera._qr_scan_finish_",
"parameters":{
"state":string,
"results":{“content”:string},
"error":{"code":int, "description":string}
}
}
```

camera._gyroCalibration

```
request:
{
"name":"camera._gyroCalibration",
}
respone:
{
"name":"camera._gyroCalibration",
parameters{
“state”:"done",
"results":{"mode":string},
"error":{"code":int, "description":string}
}
}
Async result:
{
"name":"camera._gyro_calibration_finish_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}

```

camera._magmeterCalibration

```
request:
{
"name":"camera._magmeterCalibration",
}
respone:
{
"name":"camera._magmeterCalibration",
parameters{
“state”:"done",
"results":{"mode":string},
"error":{"code":int, "description":string}
}
}
```

```
Async result:
{
"name":"camera._magmeter_calibration_finish_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}
```

camera._storageSpeedTest

```
request:
{"name":"camera._storageSpeedTest"}
respone:
{
"name":"camera._storageSpeedTest",
parameters{
“state”:"done",
"results":{"mode":string},
"error":{"code":int, "description":string}
}
}
Async result:
{
"name":"camera._storage_speed_test_finish_",
"parameters":{
"state":"done",
"results":{
  "local":bool, //大卡
  "module":[{"index":int, "result":bool}] //小卡
}
}
}
```

camera._calibrationBlc

```
request:
{
"name":"camera._calibrationBlc",
"parameters":
    {
    "reset":bool
    },
}
respone:
{
"name":"camera._calibrationBlc",
parameters{
```

```
“state”:"done",
"results":{"mode":string},
"error":{"code":int, "description":string}
}
}
Async result:
{
"name":"camera._calibrationBlcResult_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}
说明：
1.可以发起的状态：idle/preview
2.由于矫正时间长，矫正结果异步返回
3.重置矫正：reset = true，如果不带reset默认为非重置（即reset=false）
```

camera._calibrationBpc

```
request:
{"name":"camera._calibrationBpc"}
respone:
{
"name":"camera._calibrationBc",
parameters{
“state”:"done",
"results":{"mode":string},
"error":{"code":int, "description":string}
}
}
Async result:
{
"name":"camera._calibrationBpcResult_",
"parameters":{
"state":string,
"error":{"code":int, "description":string}
}
}

说明：
1.可以发起的状态：idle/preview
2.由于矫正时间长，矫正结果异步返回

```

camera._deleteFile

```
request:
```

```
{"name":"camera._deleteFile","parameters":{"dir":["dir1", "dir2", "..."]}}
respone:
{
    "name":"camera._deleteFile",
    parameters{"state":"done"}
}
Async result:
{
    "name":"camera._delete_file_finish_",
    "parameters":{
        "state":string,
        "error":{"code":int, "description":string}
    }
}
```

camera._modulePowerOn

```
request:
{"name":"camera._modulePowerOn"}
respone:
{
"name":"camera._modulePowerOn",
parameters{“state”:"done"}
}
```

camera._modulePowerOff

```
request:
{"name":"camera._modulePowerOff"}
respone:
{
"name":"camera._modulePowerOff",
parameters{“state”:"done"}
}
```

camera._startMagmeterCalibration

```
request:
{"name":"camera._startMagmeterCalibration"}
respone:
{
"name":"camera._startMagmeterCalibration",
parameters{“state”:"done"}
}
```

camera._stopMagmeterCalibration

```
request:
{"name":"camera._stopMagmeterCalibration"}
respone:
{
"name":"camera._stopMagmeterCalibration",
parameters{“state”:"done"}
}
```

camera._resetIndication

```
indication:
{
"name":"camera._resetIndication"
}
tip:
send via fifo:/data/ins_fifo_to_client_father


```

camera._timelapse_pic_take_

```
indication:
{
"name":"camera._timelapse_pic_take_",
"parameters":{
"sequece":int
}
}
```

camera._live_stats_

```
indication:
{
"name":"camera._live_stats_",
"parameters":{
"framerate":float
}
}
```

camera._net_link_state_

```
indication:
{
"name":"camera._net_link_state_",
```

```
"parameters":{
"state":string
}
}
tips:
state: connected/connecting
```

camera._gps_state_

```
indication:
{
"name":"camera._gps_state_",
"parameters":{
"state":int //0:没有gps设备 1:未定位 2:2D定位 3:3D定位
}
}
```

camera._snd_state_

```
indication:
{
"name":"camera._snd_state_",
"parameters":{
"type":int,  //0:没有音频 1:内存mic 2:3.5mm 3:usb
"is_spatial":bool, //0:非全景声 1:全景声
"dev_name":string
}
}
```

camera._getResult

```
indication:
{
"name":"camera._getResult"
"parameters":
{
"list_ids":[1,2 ..] //id array
}
}
//used for get result asyncronous by controller only
```

camera._setCustom

```
request:
{
```

```
"name":"camera._setCustom",
"parameters":{
"name": "camera._takePicture",
"parameters":{
"origin":{"mime":string, "width":int, "height":int,"saveOrigin":bool},
"stiching":{"mode":string, "mime":string, "width":int, "height":int,
"map":string, "algorithm":string},
burst:{"enable":bool, "count":int}
"delay":int,
"storagePath":string
properties:{"audio_gain":int
,"len_param":
{"aaa_mode":2,"ev_bias":0,"wb":0,"shutter_value":21,"long_shutter":1,"iso_
value":7,"brightness":87,"saturation":156,"sharpness":4,"contrast":143},
"gamma_param":"AAAEAAkADQARABYAGgAe..."}
       }
}
respone:
{
"name":"camera._setCustom",
“state”:"done",
}
ps:
1 use takePicture as example, live and record should be same
2 "gamma_param" use data same as data in camera._update_gamma_curve
```

camera._setSysSetting

```
{"name":"camera._setSysSetting",
"parameters":
{"flicker":0,"speaker":0,"led_on":0,"fan_on":0,"aud_on":0,"aud_spatial":0,
"set_logo":0,"gyro_on":0,"video_fragment":0,"reset_all":0}}

respone:
{
"name":"camera._setSysSetting",
“state”:"done"
}
1) all the value set is 0 or 1
2) aud_spatial should only set when aud_on is 1, better set aud_on and
aud_spatial in one request
3) led_on when set
```

camera._getSysSetting

```
{"name":"camera._getSysSetting"}

respone:
```

```
{
"name":"camera._getSysSetting",
“state”:"done",
"results":
{"flicker":0,"speaker":0,"light_on":0,"fan_on":0,"aud_on":0,"aud_spatial":0,"set_logo":0,"gyro_on":0,"video_fragment",0,
"pic_gamma":string,
"vid_gamma":string,
'live_gamma":string}
}
1) all the results set is 0 or 1
2) aud_on is 0 or 1, when aud_on is 1, aud_spatial is 0 to indicate normal and 1 to indicate spatial
3) light_on when get ,different from get, a history mistake
```

camera._powerOff

```
request:
{
"name": "camera._powerOff"
}
respone:
{
"name":"camera._powerOff",
“state”:string,
"error": {
       "code": "xxx",
       "description": "."
   }
}
state: done, error...
```

camera._startShell

```
request:
{
"name": "camera._startShell"
"parameters":{
               "cmd":string
       }
}
respone:
{
"name":"camera._startShell",
“state”:string,
"error": {
       "code": "xxx",
       "description": "."
```

```
    }
}
```

camera._change_mount_mode

```
{
 "name":"camera._change_mount_mode",
 "parameters": {
  "mode": "ro"
 }
}
mode: "ro" - 切换只读方式挂载
"rw" - 切换为读写方式挂载
```

box接口

http request

path: /osc/commands/stitch
支持camera.listFiles和下面stich指令
获取轮询和以前一样

stitcher.start_stitching_box

```
request:
{
"name":"stitcher.start_stitching_box"
}
respone:
{
"name":"stitcher.start_stitching_box"
“state”:string
}
说明：
1.进入拼接机状态后，不接收任何非拼接机操作：比如录像/直播/拍照/扫描等
2.所有以下拼接机命令，都必须在进入拼接机状态后才有效
```

stitcher.stop_stitching_box

```
request:
{
"name":"stitcher.stop_stitching_box"
}
respone:
{
"name":"stitcher.stop_stitching_box"
```

```
“state”:string
}
```

stitcher.query_task_list

```
request:
{
"name":"stitcher.query_task_list"
}
respone:
{
"name":"stitcher.query_task_list"
“state”:string,
"results":{"list":
    [
    {"uuid":string, "state":string, "fileSizeKB":double,"progress":double,
"parameters":object,
    "error":{“code”:int, "description":string}}
    ]
    }
    "list_started":bool
}
说明:
1.5种state:idle/pending/running/error/finish
2.对于running state的才有"progress" key
3.对于error state的才有"error" object
```

stitcher.start_task_list

```
request:
{
"name":"stitcher.start_task_list"
}
respone:
{
"name":"stitcher.start_task_list"
“state”:string,
}
说明
1.启动后，开始拼接任务
```

stitcher.stop_task_list

```
request:
{
"name":"stitcher.stop_task_list"
}
```

```
respone:
{
"name":"stitcher.stop_task_list"
“state”:string,
}
说明：
1.停止拼接任务
```

stitcher.add_task

```
request:
{
"name":"stitcher.add_task"
"parameters":[
    {
"uuid":string
“state":string
"input":{"path":string}
"blend":{"mode":string, "calibration":
{"captureTime":double},"offset":string}
"gyro":{"enable":int}
"output":{"file":string,
"video":{"width":int, height:int,bitrate:int,"fps":double},
"audio"{"type":string}，//none/pano/normal
"image":{"width":int, height:int}}
    }

    ]
}
respone:
{
"name":"stitcher.add_task"
“state”:string,
“results”:{"detail":[{"uuid":string, "code":int, "description":string}]}
}
说明：
1.bitrate单位kbits
2.captureTime单位秒
```

stitcher.delete_task

```
request:
{
"name":"stitcher.delete_task"
"parameters":["uuid":string]
}
respone:
{
```

```
"name":"stitcher.delete_task"
“state”:string,
“results”:{"detail":[{"uuid":string, "code":int, "description":string}]}
}
说明：
1.不可以删除正在执行的任务

更新任务：
request:
{
"name":"stitcher.update_task"
"parameters":[
{
"uuid":string
"input":{"path":string}
"blend":{"mode":string, "calibration":{"captureTime":double}}
"gyro":{"enable":bool}
"output":{"file":string,
"video":{"width":int, height:int,bitrate:int,"fps":double},
"audio"{"type":string}}
}

]
}
respone:
{
"name":"stitcher.update_task"
“state”:string,
“results”:{"detail":[{"uuid":string, "code":int, "description":string}]}
}
说明：
1.不可以更新正在执行的任务
```

stitcher.query_task

```
request:
{
"name":"stitcher.query_task"
"parameters":["uuid":string]
}
respone:
{
"name":"stitcher.query_task"
“state”:string,
“results”:{"detail":
[
{"uuid":string, "state":string, "progress":double, "parameters":object,
"error":{“code”:int, "description":string}}
]
}
```

```
}
说明:
1.对于running state的才有"progress" key
2.对于error state的才有"error" object
```

stitcher.start_task

```
request:
{
"name":"stitcher.start_task"
"parameters":["uuid":string]
}
respone:
{
"name":"stitcher.start_task"
“state”:string,
“results”:{"detail":[{"uuid":string, "code":int, "description":string}]}
}
```

stitcher.stop_task

```
request:
{
"name":"stitcher.stop_task"
"parameters":["uuid":string]
}
respone:
{
"name":"stitcher.stop_task"
“state”:string,
“results”:{"detail":[{"uuid":string, "code":int, "description":string}]}
}
说明:
1.可以停止正在执行的任务
```

stitcher.task_stats_

```
request:
{
"name": "stitcher.task_stats_",
"parameters":
    {
    "total_cnt":int,
    "successful_cnt":int,
    "failing_cnt":int,
    "task_over":bool,
```

```
    “runing_task_progress”:int
    }
}
说明:
1.runing_task_progress范围[0.0,100.0]
2.如果task_over = true,没有“runing_task_progress”
```

模组接口

module tool

```
1.module tool 输入参数为json格式字符串
2.结果在stdout输出,也是json格式字符串
2.进行模组操作前要先上电，操作完要下电
3.json命令及响应:
req:{"name":"power_on"}
rsp:{"respone_code":int, "description":string}

req:{"name":"power_off"}
rsp:{"respone_code":int, "description":string}

req:{"name":"blc_calibration"}
rsp:{"respone_code":int, "description":string}

req:{"name":"bpc_calibration"}
rsp:{"respone_code":int, "description":string}

req:{"name":"get_uuid_sensorid"}
rsp:

req:
{"name":"set_calibration_data",
"parameters":{
  "index":int,
  "bpc":string,
  "vig":string,
  "vig_max_value":[int]}
}
rsp:{"respone_code":int, "description":string}

req:{"name":"test_module_communication"}{"name":"test_module_spi"}
rsp:{"respone_code":int, "description":string}

req:{"name":"query_storage"}:
rsp:
{"respone_code":int, "description":string,
"results":{
```

```
  [{ "storage_total": int, "storage_left": int, "index": int }]}
 }

//respone_code = 0代表成功，其他表示失败

eg:blc calibration
module_tool {"name":"power_on"}  //上电
{"respone_code":0, "description":"ok"} //成功响应
module_tool {"name":"blc_calibration"}" //操作
{"respone_code":0, "description":"ok"} //成功响应
module_tool {"name":"power_off"}" //下电
{"respone_code":0, "description":"ok"} //成功响应

```

配置文件

```
1.camera service: cam_config.xml
<option>
    <storage>/sdcard</storage>
    <fixstorage>0</fixstorage>
    <pid>6_5_4_3_2_1</pid>
    <hwVersion>3</hwVersion>
    <moduleHwVersion>0</moduleHwVersion>
    <gyro>1</gyro> //是否启用陀螺仪
    <gyro_gravity>1</gyro_gravity> //是否启用gravity矫正
    <audiograin>96</audiograin> //音频增益
    <denoise>1</denoise> //是否进行音频降噪
    <denoisedb>-25</denoisedb> //音频降噪幅度
</option>
```

工程文件

```
1.共有：
<project>
    <offset>
        <pano_4_3></pano_4_3>
        <pano_16_9></pano_16_9>
        <left_3d></left_3d>
        <right_3d></right_3d>
    </offset>
    <camera_info make="Insta360" model="Insta360 Pro2" sn=""/>
    <version firmware="" camerad="" module=""/>
</project>

```

2.视频：

```
<origin>
        <metadata type="video" mime="h264" width="3840" height="2880" framerate="29.9700" bitrate="60000K" storage_loc="1" audio_enable="1" samplerate="48000" audio_bitrate="128" sample_format="s16" channel_layout="mono"/>
        <filegroup>
          <file>origin_1.mp4</file>
          <file>origin_2.mp4</file>
          <file>origin_3.mp4</file>
          <file>origin_4.mp4</file>
          <file>origin_5.mp4</file>
          <file>origin_6.mp4</file>
        </filegroup>
</origin>
<auxorigin>
      <metadata type="video" mime="h264" width="3840" height="2880" framerate="29.9700" bitrate="60000K" storage_loc="1"/>
        <filegroup>
          <file>origin_1.mp4</file>
          <file>origin_2.mp4</file>
          <file>origin_3.mp4</file>
          <file>origin_4.mp4</file>
          <file>origin_5.mp4</file>
          <file>origin_6.mp4</file>
        <filegroup>
</auxorigin>
<stitching>
        <file_stitching type="video" mime="h264" src="3d.mp4" width="2560" height="2560" framerate="59.9401" bitrate="60000K"/>
        <file_preview type="video" mime="h264" src="preview.mp4" width="1920" height="960" framerate="29.97" bitrate="3000K"/>
</stitching>
```

3.照片：

```
<origin>
        <metadata type="photo" mime="jpeg" width="4000" height="3000" storage_loc="0"/>
        <file>origin_0.jpg</file>
        <file>origin_1.jpg</file>
        <file>origin_2.jpg</file>
        <file>origin_3.jpg</file>
        <file>origin_4.jpg</file>
        <file>origin_5.jpg</file>
</origin>
<stitching>
        <file_stitching type="photo" mime="jpeg" src="pano.jpg" width="3840" height="1920"/>
        <thumbnail src="thumbnail.jpg" width="1920" height="960"/>
</stitching>
```

性能

实时拼接性能

帧率	H264 pano max	H264 3d max	H265 pano max	H265 3d max
30fps	s:3840x1920o:2560x1440	s:3840x3840o:2560x1920	s:3840x1920o:2560x1440	s:3840x3840o:2560x1920
60fps	s:3840x1920o:1920x1080	s:2560x2560o:1920x1440	s:3840x1920o:1920x1080	s:2560x2560o:1920x1440

```
注：
30fps实时拼接，预览流：2160x1080@30fps
60fps实时拼接，预览流：2160x1080@15fps
```

usb传输性能

存储读写性能

错误码

code	description	
模组错误码		
301	接收json命令失败	
302	加密校验失败	
303	模组执行命令同步失败	
304	重传帧数据失败	
310	无卡	
311	没有存储空间	
312	卡保护、存储过程卡被移除、卡出现异常	
313	卡速不足	
314	文件大小达到最大限制	
camerad 错误码		
410	错误的消息格式	
411	错误的消息参数	
412	配置文件错误	
413	该状态下不允许的操作	
414	麦克风异常	
415	光流拼接异常	
416	读不到陀螺仪数据	
417	相机温度高	
418	camera service busy	
419	不支持调节音量	
420	未实现	
422	不存在	
423	磁力计数据静止	
430	文件打开错误	

431	文件io错误	
432	没有存储空间	
433	没有存储设备	
434	存储设备写速度不够	
435	mux写出错	
436	mux打开出错	
437	网络重连中	
438	网络重连失败	
439	网络断开	
450	编解码器卡死	
451	解码出错	
452	编码出错	
460	模组打开失败	
461	模组没有打开	
462	模组读命令失败	
463	模组写命令失败	
464	模组读数据失败	
465	模组写数据失败	
466	模组掉线	
467	模组帧不同步	
470	找不到工程文件	
471	工程文件损坏	
472	任务列表数据库损坏	
473	原始视频文件损坏	
474	陀螺仪数据文件损坏	
480	拼接任务不存在	
481	拼接任务已经存在	
482	拼接任务正在拼接中	
491	内存不足	
升级		

701	升级设备存储空间不足	
702	升级设备只读	
703	升级文件校验失败	
704	升级文件打开失败	
705	升级文件读取KEY失败	
706	KEY校验失败	
707	提取升级文件版本失败	
708	提取升级程序长度失败	
709	提取升级程序压缩包失败	
710	解压升级程序失败	
711	提取升级程序失败	
811	模组升级失败	