

各接口通用

相机IO口定义

相机IO口连接器采用：HIROSE # HR10A-10P-12S (73) 型连接器。



Pin	颜色	信号	用途
1	绿	12 VDC RTN	12V供电回线GND
2	灰	+ 12 VDC	+12V DC
3	白	保留	
4	粉	保留	
5	紫	OUT2 RTN	GPIO 输出2, 光耦触点1
6	黑	OUT1 RTN	GPIO 输出1, TTL回线
7	粉/红	OUT1	GPIO 输出1, TTL
8	红	IN1	GPIO 输入1, 光耦触点1
9	蓝	IN2	GPIO 输入2, TTL
10	蓝/红	IN1 RTN	GPIO 输入1, 光耦触点2
11	棕	IN2 RTN	GPIO 输入2, TTL回线
12	黄	OUT2	GPIO 输出2, 光耦触点2

线缆颜色以厂家标配线缆为例。

参数区

相机内部共有5组参数存储空间（参数区），一组为不可修改的出厂默认参数（factory），其余4组（UserSet0-3）可由用户自行读写。用户参数区具体参数定义根据不同产品型号，可能有所差异。

同时可通过命令选择相机上电时加载的参数区，以达到快速配置相机工作的目的。发送保存指令时，会将当前配置好的工作参数写入选中的参数组中（factory不可改）。当设置BootFrom时，可以确定下次相机启动的加载用户参数组。

USB接口专有

USB接口短波二次开发基于港宇SDK进行。

修改CyUsb3Xml的路径

在编译目标文件exe所在目录增加一个名为 CySdkConfig.json，内容写成

```
{"CyUsb3XmlPath": "CyUsb3Xml/"}
```

修改传输帧率

```
// 在CY_StartGrabbing之前修改该参数

// 打开帧率控制开关，如要恢复默认帧率则发false
CY_XML_SetBoolValue(pDevHandle, "AcquisitionFrameRateEnable", true);
CY_XML_SetEnumValueStr( pDevHandle, "TriggerMode", "off" );
// 修改帧率为100fps
CY_XML_SetFloatValue(pDevHandle, "AcquisitionFrameRate", (float)100);
```

开启ROI提高帧率

```
// 在CY_StartGrabbing之前修改该参数

// 设置模式为开窗，如果要恢复全画幅则发 All_Pixels

CY_XML_SetEnumValueStr( pDevHandle, "Mode", "Faster_Frame_Rate" );

// 设置窗口宽高为 160*160（16的整倍数）
CY_XML_SetIntValue( pDevHandle, "width", 160 );

CY_XML_SetIntValue( pDevHandle, "Height", 160 );

// ROI起始坐标 64,64
CY_XML_SetIntValue( pDevHandle, "OffsetX", 64 );
CY_XML_SetIntValue( pDevHandle, "OffsetY", 64 );
```

手动设置曝光时间

```
// 关闭自动曝光时间
CY_XML_SetEnumValueStr(pDevHandle, "ExposureAuto", "off" );
// 发送曝光时间1020μs
CY_XML_SetFloatValue(pDevHandle, "ExposureTime", (float)1020);
```

手动选择增益

```
// 关闭自动曝光时间
CY_XML_SetEnumValueStr(pDevHandle, "GainAuto", "Off" );
// 设置中增益
CY_XML_SetEnumValueStr(pDevHandle, "Gain_Mode", "中增益" );
```

CameraLink接口专有

串口读写介绍

相机使用串口对寄存器进行访问，串口协议格式包括 1位起始位、8位数据位及1位停止位（无握手，无校验）。默认波特率115200，可以配置波特率为9600, 19200, 38400, 57600, or 115200。

寄存器写操作

写操作向相机发送7字节序列写入寄存器，如果无误，相机会返回一个字节写应答。

写操作 (7 Bytes): <写命令> <寄存器地址> <数据>

Byte	Word
1	0x57 (写命令)
2	<Register Address_High> MSB
3	<Register Address_Low> LSB
4	<Register Data Byte 4> MSB
5	<Register Data Byte 3> ...
6	<Register Data Byte 2> ...
7	<Register Data Byte 1> LSB

写应答(1 Byte):

Byte	Word
1	0x06 (应答)

如果存在错误，相机返回两个字节错误码，1字节NCK+1字节Error Code。

Byte	Word
1	0x15 (NCK)
2	(Error Code. 错误码可查看错误码描述章节)

寄存器读操作

读操作向相机发送3字节序列读寄存器，如果无误，相机返回5字节，1字节应答+4字节数据。错误情况下，读操作也会返回1字节非应答+1字节错误码，但是如果指定错误的地址，会返回1字节应答+4字节0数据。

读操作 (3 Bytes): <读命令> <地址>

Byte	Word
1	0x52 (读命令)
2	
3	

相机返回(5 bytes): <应答> <数据>

Byte	Word
1	0x06 (应答)
2	<Register Data Byte 4> MSB
3	<Register Data Byte 3> ...
4	<Register Data Byte 2> ...
5	<Register Data Byte 1> LSB

如果存在错误，相机返回两个字节错误码，1字节NCK+1字节Error Code。

Byte	Word
1	0x15 (NCK)
2	(Error Code. 错误码可查看错误码描述章节)

错误码

串口通信出错时，系统返回错误代码，见下表

错误码	描述
0X15	非应答码，指示当前有错误发生。
0x01	非法命令，命令码不是0x52或0x57。
0x02	超时，读写命令超时时间为100ms。

数据格式

相机工作参数遵循GenICam规范，参数类别参照GenICam SFNC v2.7中的相关规定。开发及调试过程中，0x0044当前曝光时间、0x0058最小曝光时间、0x005C最大曝光时间等参数，需要使用float型（IEEE754规范）进行下发。若使用串口助手进行调试，可使用转换工具进行辅助转换；若在程序代码中，可直接使用float类型转换即可。

曝光时间最小步距为1 μ s。

IEEE754浮点数计算

相机的曝光时间、曝光时间上下限等参数均为IEEE754浮点类别，通过串口发送时，需要将常规浮点数转换为IEEE754类型，参考的纯C语言代码示例如下：

```
#include <stdio.h>
#include <stdint.h>

// Function to convert an int32_t to IEEE 754 single-precision hex
uint32_t intToIEEE754Hex(int32_t input) {
    // Define the sign bit, exponent, and mantissa parts
    uint32_t sign = (input < 0) ? 1 : 0;
    int32_t exponent = 0;
    uint32_t mantissa = 0;

    // inputs <= 0
    if (input == 0)
    {
        exponent = 0;
        mantissa = 0;
    }else
    {
        if (input < 0)
        {
            input = -input;
            sign = 1;
        }

        int shift = 0;
        while ((input >> shift) >= 2)
```

```

    {
        shift++;
    }

    // exponent and mantissa
    exponent = shift + 127; // adding bias of 127
    mantissa = (input << (23 - shift)) & 0x007FFFFFFF;
}

// combine sign, exponent, and mantissa into the IEEE 754 format
uint32_t ieee754 = ((uint32_t)sign << 31) |
    ((uint32_t)(exponent & 0xFF) << 23) |
    mantissa;

    return ieee754;
}

// test yo
int main() {
    int32_t input;
    printf("Exposure Time: ");
    scanf("%d", &input);

    printf("0x%08X\n", intToIEEE754Hex(input));

    return 0;
}

```

设置开窗模式

设置开窗可以提高相机的采集帧率，注意开窗的宽度和高度必须为16的整倍数。

功能	命令
打开开窗模式	57 00 10 00 00 00 01
设置图像水平宽度 (设为256)	57 00 18 00 00 01 00
设置图像垂直高度 (设为256)	57 00 20 00 00 01 00
恢复全尺寸模式	57 00 10 00 00 00 00

设置帧率

参照GenICam规范，设置帧率前需打开帧率控制使能开关。

功能	命令
开启帧率控制	57 00 48 00 00 00 01
设置帧率 (设为50)	57 00 4C 00 00 4E 20

功能	命令
关闭帧率控制	57 00 48 00 00 00 00

手动自动控光相关

手动控光功能介绍

短波红外相机手动控光（调节亮度）可分别对增益档位和曝光时间进行控制。

手动增益控制

功能	命令
关闭自动增益控制	57 01 44 00 00 00 00
设置为低增益档	57 00 00 00 00 00 02
设置为中增益档	57 00 00 00 00 00 01
设置为高增益档	57 00 00 00 00 00 00

手动曝光时间控制

功能	命令
关闭自动曝光时间控制	57 01 40 00 00 00 00
设置曝光时间（设为500 μ s）	57 00 44 43 FA 00 00

其中加粗的部分为IEEE754浮点数表示的曝光时间值，单位 μ s。

自动控光功能介绍

自动控光包括AGC（自动增益控制）和AEC（自动曝光时间），二者可分别使用也可组合使用。调整后的图像统计亮度由**自动控光目标值**决定，该目标值为指定区域（默认全画面）的灰度均值计算而来。由于提高增益也会同步增大噪声，因此当AGC和AEC同时开启时，有不同的调整策略。

亮度调整方向	调整策略
提高亮度	优先提高曝光时间，如果增大到上限后仍未达到目标值，则提高增益档
降低亮度	优先降低增益档，再修正曝光时间，使图像统计亮度达到目标值

调整速度由自动控光速度进行调节，x1最慢但精度更高，x4最快但精度略低。

功能	命令
自动控光目标值（设为50）	57 01 48 00 00 00 32
自动控光速度 x1	57 01 64 00 00 00 00

*功能**	命令
自动控光速度 x2	57 01 64 00 00 00 01
自动控光速度 x3	57 01 64 00 00 00 02
自动控光速度 x4	57 01 64 00 00 00 03

自动增益控制AGC

自动增益控制开关见下表。

功能	命令
自动增益开	57 01 44 00 00 00 01
自动增益关	57 01 44 00 00 00 00

自动曝光时间控制AEC

自动曝光时间控制，除了需要开关指令外，还需要配置调整范围的曝光时间上下限，单位均为 μs 。

功能	命令
自动曝光开	57 01 40 00 00 00 01
自动曝光关	57 01 40 00 00 00 00
自动曝光下限（设为100）	57 00 58 42 C8 00 00
自动曝光上限（设为1000）	57 00 5C 44 7A 00 00

附录

CameraLink控制协议

TBD