**SPI/I2C接口软件开发资料说明**

**SPI/I2C Interface Development Instructions**

目录

[SDK构成The SDK components 2](#_Toc109232278)

[DVP\_I2C\_sample目录：The Sample source code 2](#_Toc109232279)

[libirSDK.chm 3](#_Toc109232280)

[ASIC\_384\_640模组SDK接口V1.3.xlsx Interface Description 3](#_Toc109232281)

[用户开发标定User calibration instructions 3](#_Toc109232282)

[使用前提Premise 3](#_Toc109232283)

[SPI接口说明SPI Interface Description 3](#_Toc109232284)

[1. VOSPI协议介绍Introduction to vospi protocol 3](#_Toc109232285)

[2. SPI时钟相位和极性控制 SPI clock phase and polarity control 4](#_Toc109232286)

[3. 读取一帧新的图像数据时序Read a new frame of image data 5](#_Toc109232287)

[4. 读取一帧图像数据续传时序Read the continuous transmission sequence of one frame of image data 6](#_Toc109232288)

[5. 相关软件设置Related software settings 7](#_Toc109232289)

[I2C接口说明I2C Interface Description 7](#_Toc109232290)

[1.特殊命令接口Special command interface 8](#_Toc109232291)

[2. 其他命令接口Other command interface 8](#_Toc109232292)

[SDK其他功能介绍Other Functions 8](#_Toc109232293)

[固件升级功能Firmware Upgrade 8](#_Toc109232294)

[测温二次标定Secondary calibration 9](#_Toc109232295)

[盲元标定Dead pixel correction 9](#_Toc109232296)

[测温环境变量修正功能Ambient variable correction 9](#_Toc109232297)

[点线框测温说明Description of point/line/box temperature measurement 9](#_Toc109232298)

[其他库的功能介绍Other Functions 10](#_Toc109232299)

[数据格式说明Data Format Description 10](#_Toc109232300)

[1.图像数据Image data 10](#_Toc109232301)

[2.温度数据Temperature data 10](#_Toc109232302)

[图像显示方法Image display method 11](#_Toc109232303)

该版本的SDK主要针对ASIC\_384\_640系列产品在linux平台下通过SPI接口进行图像/温度数据传输和I2C接口进行命令控制。

This version of SDK mainly aims at image/temperature data transmission through SPI interface and command control through I2C interface for ASIC\_384\_640 series products on Linux platform.

## SDK构成The SDK components

### DVP\_I2C\_sample目录：The Sample source code

1. include目录：里面包含的是libiruvc、libirprocess、libirparse、libircmd、libiri2c库的对外开放头文件。

include directory: Libraries open header files of libiruvc, libirprocess, libirparse, libircmd、libiri2c.

1. sample.cpp: 包含sample的主函数，主要包含用户命令注册，打印等级设置，可设置数据采集的分辨率/帧率，以及伪彩/翻转/镜像/旋转/输出格式等功能。

sample.cpp: The main function of sample application, mainly including user command registration, print level setting, data collection resolution/frame rate setting, and pseudo-color/flip/mirror/rotate/output format.

1. cmd.cpp:主要包含用户发送命令的各种demo。

cmd.cpp: mainly contains various demos of commands sent by users.

1. display.cpp:获取图像帧信息之后，根据之前frame\_info里参数的设定，做图像数据格式转换、翻转/镜像、旋转等处理，最后调用opencv显示出来图像。

display.cpp: after obtaining image frame information, according to the Settings of parameters in the previous frame\_info, image data format conversion, flip/mirror, rotation and other processing, and finally call OpencV to display the image.

1. camera.cpp:用于获取红外模组数据流，当stream线程获取到原始红外帧信息的时候，会将红外帧信息raw frame切分为图像信息image frame和温度信息temp frame，并发送信号，传递给对应的模块做相应的处理，当image frame和temp frame处理完成后发送信号给camera线程，camera线程继续下一次循环。

camera. cpp: Used to get stream data from infrared module. When the stream thread obtains the original infrared frame information, it will slice the raw frame information into image frame information and temperature information temp frame, and send signals to the corresponding module for corresponding processing. When the image frame and Temp frame processing are completed, signals are sent to the camera thread, which continues the next loop.。

1. spi.cpp: 设置SPI接口的通道、模式和速度，获取SPI接口的数据。

spi.cpp: Set the channel, mode, and speed of the SPI interface, and get data of the SPI interface

1. readme.md: sample软件框架的说明。

readme. md: Description of the sample software framework in Chinese.

1. readme\_EN.md：sample软件框架的英文说明。

readme en. md: sample Software framework description in English

1. irsample\_structure.png：sample软件框架图解。客户可参考该框架进行开发。

irsample\_structure.png：Sample Software framework diagram. Customers can refer to this framework for development.

1. makefile：客户可根据该文件利用make工具对sample进行编译。

makefile：From this file, customers can compile sample using the make tool.

### ASIC\_384\_640模组SDK接口V1.5.xlsx Interface Description

概要说明SDK各个库中的函数接口，方便用户大致了解SDK所支持的功能。

This section describes the function interfaces in each SDK library to help users understand the functions supported by the SDK

### 用户开发标定User calibration instructions

包含盲元标定和测温标定的说明，以及温度的环境变量修正。

Contains the description of Secondary calibration, Dead pixel correction, Ambient variables correction

## 使用前提Premise

1. 客户的开发板或者目标平台芯片支持I2C接口和SPI接口。

Customer's development board or platform supports I2C&SPI interface

1. 客户的开发平台在设备树上添加I2C和SPI的设备节点，并且挂载I2C驱动和SPI驱动。可以通过查看dev目录下是否有i2c-x和spidevx.x来判断。

The customer's development platform adds I2C and SPI device nodes to the device tree, and mounts the I2C driver and SPI driver. You can check whether the dev directory contains i2C-x and spidevx.x.

1. 客户开发平台最好能支持串口调试。(非必要)

Customer development platform should support serial port debugging. (Not necessary)

1. 如果客户需要显示出图的话，需要安装opencv库。如果有其他显示方法可以不用安装这个库。

The OpencV library is installed if the client needs to display the graph. You don't need to install this library if you have other display methods.

## SPI接口说明SPI Interface Description

ASIC\_384\_640产品支持SPI图像/温度数据输出，即VOSPI。

The ASIC\_384\_640 product supports SPI image/temperature data output, i.e. VOSPI.

### 1. VOSPI协议介绍Introduction to vospi protocol

VOSPI全称video out SPI interface，作为ASIC\_384\_640搭载的红外ASIC芯片Video输出的一种接口定义。使用Motorola Serial Peripheral Interface (SPI) 4 lines 接口协议。支持SPI CPOL clock polarity=0/1和CPHA clock phase=0/1共4种传输模式，支持SPI 8bit MSB or LSB first。VOSPI定义了2种Command Format，包括“Read a New Frame” 和“Read a Continued Frame”。 “Read a New Frame”用于SPI master从ISP芯片获取一帧新图像。当一帧图像太长，一次continuous transfer不能传输完时，“Read a Continued Frame”用于前面“Read a New Frame” 一帧图像的续传。ASIC\_384\_640搭载的红外ASIC芯片VOSPI模块的最大频率为75Mhz。

The full name of vospi is video over SPI interface, which is an interface definition for the video output of the infrared ASIC chip carried by ASIC\_384\_640. Use Motorola serial peripheral interface (SPI) 4 lines interface protocol. Support SPI cpol clock polarity = 0 / 1 and CPHA clock phase = 0 / 1, and SPI 8bit MSB or LSB first. Vospi defines two command formats, including "read a new frame" and "read a continued frame". "Read a new frame" is used by SPI master to obtain a new image from ISP chip. When a frame of image is too long to be transmitted in a continuous transfer, "read a continued frame" is used for the subsequent transmission of the previous "read a new frame" image. The maximum frequency of vospi module of infrared ASIC chip carried by ASIC\_384\_640 is 75MHz.

### 2. SPI时钟相位和极性控制 SPI clock phase and polarity control

#### CPHA=0 Transfer Format

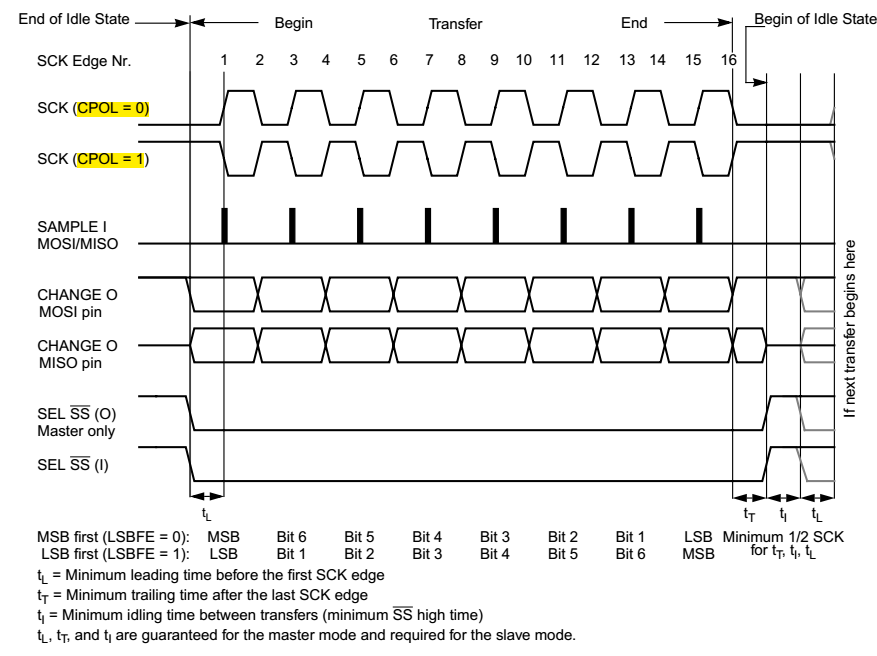


Figure 1. CPHA=0

#### CPHA=1 Transfer Format

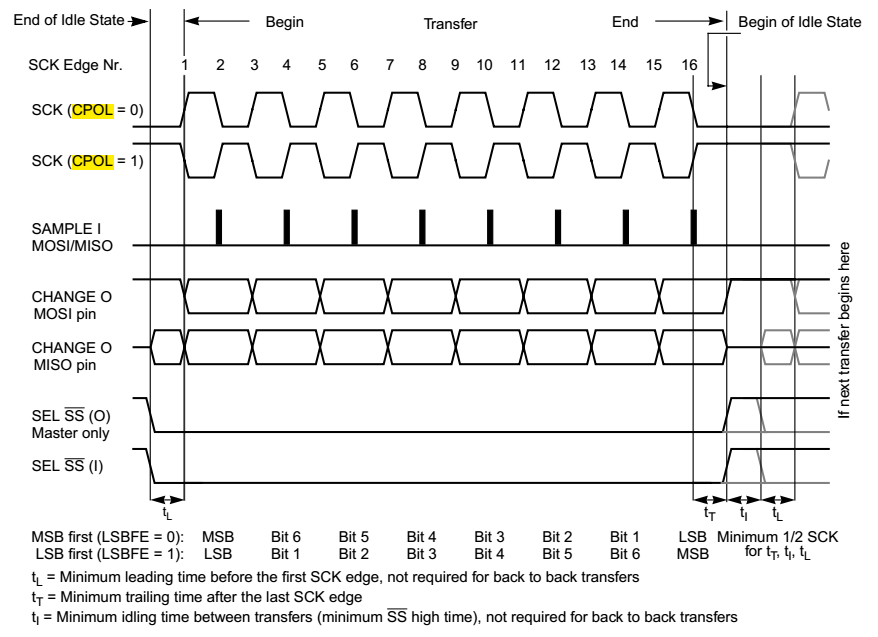


Figure 2. CPHA=1

目前ASIC\_384\_640产品采用CPOL=1，CPHA=1。即空闲状态为高电平，第二个跳变沿采样数据。

At present, ASIC\_384\_640 products adopt CPOL = 1 and CPHA = 1. That is, the idle state is high, and the second jump edge samples data.

### 3. 读取一帧新的图像数据时序Read a new frame of image data

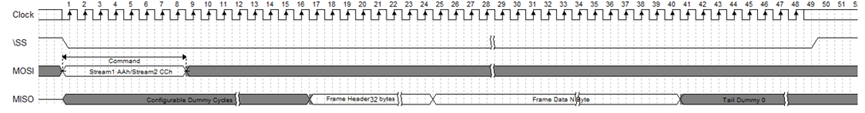


Figure 3. Read a New Frame

“Read a New Frame”用于SPI master从红外ASIC芯片获取一帧新图像。

Configurable Dummy Cycles表示SPI slave收到“Read a New Frame” command后，需要时间去DDR取数据，在取到数据前，先回dummy数据。Dummy Cycles长度由DDR取数据latency决定，长度为480。

Dummy传输完毕，紧接着传输frame header，frame header共32Bytes。Byte0[0]为1表示frame数据valid，为0表示frame数据invalid。Byte1 reserved。Byte2为frame\_index[7:0]，Byte3为frame\_index[15:8]。Byte 4~31 为reserved，可以寄存器配置，用于传输一些自定义状态信息。

SPI Slave可支持的continuous transfer长度可以无限长。当continuous transfer长度大于一帧图像时，在一帧后面为Tail Dummy 0数据。

"Read a new frame" is used by SPI master to obtain a new image from infrared ASIC chip.

Configurable dummy cycles means that after SPI slave receives the "read a new frame" command, it needs time to fetch data from DDR. Before fetching data, it returns dummy data first. The length of dummy cycles is determined by the latency of DDR data. The range is [512, XXX], and the unit is SCK cycle. It must be an integer multiple of 8.

The transmission of dummy is completed, followed by the transmission of frame header, which is 32bytes in total. If byte0 [0] is 1, it means frame data is valid, and if byte0 [0] is 0, it means frame data is invalid. Byte1 reserved。 Byte2 is frame\_ Index [7:0], byte3 is frame\_ index[15:8]。 Byte 4 ~ 31 is reserved and can be configured with registers to transmit some user-defined status information.

The length of continuous transfer supported by SPI slave can be unlimited. When the length of continuous transfer is greater than one frame of image, tail dummy 0 data is displayed after one frame.

Vospi帧头的大小是32字节，以下是具体每一个字节的定义

The size of vospi frame header is 32 bytes. The following is the definition of each byte

Table for definition of vospi header

|  |  |
| --- | --- |
| 0byte | FrameValid (bit0:1 vaild bit0:0 invaild) |
| 1byte | reserve |
| 2byte | FrameIndex\_L |
| 3byte | FrameIndex\_H |
| 4byte | Shutter state (0:closed 1:opened) |
| 5byte | Vtemp\_L |
| 6byte | Vtemp\_H |
| 7byte | Shutter temp\_L In ASIC\_384\_640 this bytes are always "0" |
| 8byte | Shutter temp\_H In ASIC\_384\_640 this bytes are always "0" |
| 9byte | current Gain state(1:high gain   0:low gain) |
| 10byte | Temp\_area In ASIC\_384\_640 this byte is always "1" |
| 11byte | Over exprosure state In ASIC\_384\_640 this byte is always "0" |
| 12byte | pix freeze state(1:in freeze state  0:not in freeze state) |
| 13byte-31byte | reserve |

### 4. 读取一帧图像数据续传时序Read the continuous transmission sequence of one frame of image data

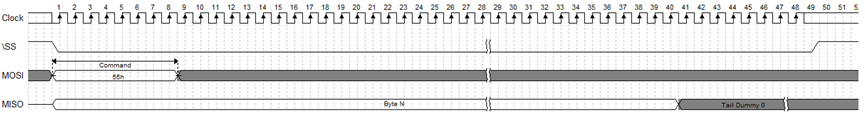


Figure 3. Read a Continued Frame

“Read a New Frame”用于SPI master从ISP芯片获取一帧新图像。当一帧图像太长，一次transfer不能传输完时，“Read a Continued Frame”用于前面“Read a New Frame” 一帧图像的续传。续传的数据从\SS拉低的第一个bit开始传输。

SPI Slave可支持的continuous transfer长度可以无限长。当continuous transfer长度大于一帧图像时，在一帧后面为Tail Dummy 0数据。

"Read a new frame" is used by SPI master to obtain a new image from ISP chip. When a frame of image is too long to be transmitted in one transfer, "read a continued frame" is used for the subsequent transmission of the previous "read a new frame" image. The continued data is transmitted from the first bit pulled down by \ SS.

The length of continuous transfer supported by SPI slave can be unlimited. When the length of continuous transfer is greater than one frame of image, tail dummy 0 data is displayed after one frame.

### 5. 相关软件设置Related software settings

ASIC\_384\_640产品支持SPI图像/温度数据输出，可通过SPI.cpp文件，修改设备节点名称，设置通道、SPI速率和SPI模式，目前的ASIC\_384\_640产品采用模式3。即

ASIC\_384\_640 products support SPI image/temperature data output, through the SPI.cpp file, modify the device node name, set channel, SPI rate and SPI mode, the current products use mode 3.

SPI推荐频率设置SPI recommended frequency Settings：

384\*288分辨率： 384\*288\*2\*8\*25=44.2368Mbps 所以应该大于等于44.2368 Mbps。

目前支持384\*288（640\*512）/384\*576（640\*1024）四种分辨率。具体说明见下表：

|  |  |  |
| --- | --- | --- |
| 分辨率  resolution | 数据  data | 备注  notes |
| 384\*576  （640\*1024） | 图像+温度  Image + Temperature | 同时获取图像数据和温度数据  Image data and temperature data are acquired simultaneously |
| 384\*288  （640\*512） | 图像/温度  Image/Temperature | 可通过basic\_preview\_mode\_select接口切换成温度输出  You can switch to the temperature output through the basic\_preview\_mode\_select interface |

目前sample中根据不同的需要可进行四种不同的图像/温度数据输出方式，分别通过宏进行设置。具体的说明见下表。

At present, four different image/temperature data output modes can be carried out in sample according to different needs, which are respectively set by macro. See the table below for specific instructions.

|  |  |  |
| --- | --- | --- |
| 分辨率  resolution | 宏macro | 说明notes |
| 384\*576  （640\*1024） | NORMAL | 该模式下每帧都获取图像数据和温度数据。  In this mode, image data and temperature data are acquired in each frame. |
| 384\*576  （640\*1024） | INTERLACE | 如果客户平台的SPI传输带宽不够，可采用该种模式，图像帧和温度帧交替获取，目前是获取5帧图像后再获取1帧温度数据。  If the SPI transmission bandwidth of the client platform is not enough, this mode can be adopted to obtain image frames and temperature frames alternately. At present, 5 frames of images are obtained and then 1 frame of temperature data is obtained. |
| 384\*288  （640\*512） | IMAGE\_OUTPUT | 输出图像数据。Output image data. |
| 384\*288  （640\*512） | TEMP\_OUTPUT | 输出温度数据。Output temperature data |

## I2C接口说明I2C Interface Description

LIbiri2c库中包含I2C控制命令接口。通过函数ircmd\_create\_handle函数注册相关的接口类型来使用不同的通信接口。该库的源码可开放给客户，也可以由客户发送交叉编译工具链由我们编译。目前已经编译好主流的嵌入式平台的库，客户可以根据需要进行选用。I2C接口最大时钟频率不超过400Khz。

The I2C control command interface is included in the Libiri2c library. The different communication interfaces are used by registering the related interface types with the function ircmd\_create\_handle. The source code of the library can be opened to the customer, or the customer can send a cross-compilation tool chain for us to compile. At present, we have compiled the mainstream embedded platform library, and customers can choose according to their needs. The maximum clock frequency of the I2C port does not exceed 400Khz.

### 1.特殊命令接口Special command interface

可参考《ASIC\_384\_640模组SDK接口V1.2.xlsx》表格，因SDK的产品兼容性考虑，部分接口并非本产品使用，有些接口存在冗余。用户只需使用《ASIC\_384\_640模组SDK接口V1.2.xlsx》表格中的指令即可，**表格中未列出的指令请不要使用，尤其是擦写flash的函数**。下表重点介绍几个常用的命令接口。

Refer to “ASIC\_384\_640模组SDK接口V1.2.xlsx”. Due to the compatibility of SDK products, some interfaces are not used by this product, and some interfaces are redundant. Users only need to use the instructions in the tabl e“ASIC\_384\_640模组SDK接口说明Interface Description1.2”, please do not use the instructions not listed in the table, especially the function to erase and write the flash. The following table focuses on several common command interfaces.

|  |  |  |
| --- | --- | --- |
| 函数接口Function | 功能Description | 说明Notes |
| basic\_y16\_preview | 用于切换出图数据源  (可切换成温度数据输出)  Use to switch out the graph data source(Can be switched to temperature data output) | 仅适用于384\*288/640\*512分辨率下使用  384\*288/640\*512 Only |
| adv\_dpc\_auto\_calc | 用于模组内置自动盲元标定  Auto Dead Pixel correction | 设定时间越长，效果越好  By increasing the setting time, the effect can be improved |
| adv\_tpd\_ktbt\_recal\_1point | 用于测温二次标定(单点标定)  One-point Secondary calibration | 需要等待较长时间，约10s  about 10s |
| adv\_tpd\_ktbt\_recal\_2point | 用于测温二次标定(两点标定) Two-point Secondary calibration | 需要等待较长时间，约15s  about 15s |

### 2. 其他命令接口Other command interface

可参考《ASIC\_384\_640模组SDK接口V1.3.xlsx》表格

Refer to libirSDK.chm and “ASIC\_384\_640模组SDK接口说明Interface Description1.3”.

## SDK其他功能介绍Other Functions

### 固件升级功能Firmware Upgrade

因为模组的固件还在不断的开发和完善中，如果出现固件bug或者需要使用固件支持的新功能，用户可以通过该功能进行固件升级。调用libiruvc库中的update\_fw函数，参考sample中的UPDATE\_FW，打开后即可进行固件升级，用户可将固件升级的相关代码移植到自己的应用软件中。

Because the firmware of the module is still under constant development and improvement, users can use this function to upgrade the firmware if there are firmware bugs or new features supported by the firmware. Call update\_fw in libiruvc library and refer to UPDATE\_FW in Sample. After opening update\_fw, you can upgrade firmware. You can transplant firmware upgrade codes to your own application software.

### 测温二次标定Secondary calibration

非制冷红外热像仪对温度敏感，无论模组出厂测温标定如何精确，当用户将模组集成到整机中后，由于热分布的变化、光学结构的变化（加窗口片等），必然引起一定的测温偏差。因此需要在整机端进行二次标定。该功能在模组固件中实现，详细的使用说明见《客户开发标定》目录下的《二次标定》。

When the user integrates the module into the whole machine, it will inevitably cause a certain temperature measurement deviation due to the changes of thermal distribution and optical structure (add window, etc.). Therefore, Secondary calibration is required at the end of the machine. This function is realized in the module firmware. For detailed instructions, see "Secondary calibration " in the directory of "User calibration instructions".

### 盲元标定Dead pixel correction

模组在使用过程中，受到比较严重的机械冲击或静电放电后，有极低的概率出现新增盲元，用户可将新增盲元添加到盲元表中。该功能有多种实现方法：1. 调用固件中的自动标定功能；2. 手动输入新增盲元坐标。详细的使用说明见《客户开发标定》目录下的《盲元标定》。

If a module is hit by a serious mechanical shock or electrostatic discharge (ESD), there is a very low probability that new dead pixels will appear. You can add new dead pixels to the dead pixel table. There are several ways to implement this function: 1. Call the automatic calibration function in firmware; 2. Manually enter the coordinates of the new Dead pixel. For detailed instructions, see Dead pixel correction in the catalogue of "User calibration instructions"

### 测温环境变量修正功能Ambient variable correction

红外测温的精度受到很多环境参数的影响，例如距离、环境温度、反射温度、湿度、目标发射率等等。Libirtemp SDK为用户提供了一种测温修正的方法，可以使测温精度更加接近真实值，接口为enhance\_distance\_temp\_correct。详细的修正方案见《客户开发标定》目录下的《环境变量修正》。

The temperature measurement of the module is affected by Emissivity, Atmospheric transmissivity, ambient reflection temperature, ambient atmospheric temperature, target distance, et Libirtemp SDK provides users with a correction method, which can make the temperature measurement accuracy closer to the real value，the interface is enhance\_distance\_temp\_correct. For detailed instructions, see "Ambient variable correction" in the directory of "User calibration instructions".

### 点线框测温说明Description of point/line/box temperature measurement

本产品模组内置点线框测温功能，可以分别调用libircmd库中的basic\_tpd\_get\_point\_temp\_info、basic\_tpd\_get\_line\_temp\_info，basic\_tpd\_get\_rect\_temp\_info来获取模组内的特定的点线框温度信息，也可以调用函数basic\_tpd\_get\_frame\_temp\_info函数获取整帧的温度最大/最小值信息。

同时SDK libirtemp中也提供了点线框的测温功能，使用前提是获取到了一整帧的温度数据，可以分别调用get\_point\_temp、get\_line\_temp、get\_rect\_temp函数获取特定的点线框温度信息。为了减小空域噪声的影响，其中点测温的功能是取点周围3\*3领域内的像素温度值去掉最大最小值后的平均值，线测温和框测温均以点测温为基础进行开发。若用户不满意该SDK的功能，可自行开发。

This product module has built-in temperature measurement function of point, line and box. You can call basic\_tpd\_get\_point\_temp\_info、basic\_tpd\_get\_line\_temp\_info、basic\_tpd\_get\_rect\_temp\_info to get the specific temperature information. You can also call the basic\_tpd\_get\_frame\_temp\_info function to obtain maximum/minimum temperature information for the entire frame.

At the same time, the SDK libirtemp also provides the temperature measurement function of point, line and box. The prerequisite is to obtain the temperature data of a whole frame. You can call get\_point\_temp, get\_line\_temp, and get\_rect\_temp respectively to obtain the specific temperature information. In order to reduce the influence of spatial noise, the function of point temperature measurement is to take the average value of pixel temperature values in 3\*3 fields around the point after removing the maximum and minimum values. Both line and box temperature measurement are developed on the basis of point temperature measurement. If users are not satisfied with the functions of the SDK, they can develop it by themselves.

### 其他库的功能介绍Other Functions

libirprocess、libirparse、libirtemp三个库文件均是软件层面的数据处理函数库，客户可根据具体的需求选用相关的函数接口。

libirprocess、libirparse、libirtemp are all data processing function libraries at the software layer. Customers can choose relevant function interfaces according to their specific needs.

libirprocess:实现了图像的旋转、镜像、翻转、图像增强、空域滤波等功能

The functions of image rotation, mirror, flip, image enhancement, Spatial Noise Reduction and so on are realized

libirparse:实现了图像数据格式转换功能

The functions of Data conversion are realized

libirtemp:实现了点、线、框测温和相关的测温修正函数

The functions of point/line/box temperature measurement are realized

## 数据格式说明Data Format Description

### 1.图像数据Image data

图像数据的输出格式为YUV422格式，即YUYV格式。

The output format of image data is YUV422 format, namely YUYV format.

### 2.温度数据Temperature data

#### 获得一整帧的温度数据Get a whole frame of temperature data

每个像素的温度需要2个byte存储，放在高14位，按照小端存放。如下图。

The temperature data of each pixel needs to be stored in 2 bytes, which are placed in the upper 14 bits and stored as the little end.

|  |  |
| --- | --- |
| 低地址 Low address | 高地址 High address |
| 低字节 Low byte | 高字节 high byte |

计算公式为The calculation formula：Celsius temp = Count/64-273.15

#### 发命令获取点线框温度数据Send the command to get the temperature data

每个像素的温度需要2个byte存储，放在低14位，按照小端存放。如下图。

The temperature data of each pixel needs to be stored in 2 bytes, which are placed in the upper 14 bits and stored as the little end.

|  |  |
| --- | --- |
| 低地址 Low address | 高地址 High address |
| 低字节 Low byte | 高字节 high byte |

计算公式为The calculation formula：Celsius temp = Count/16-273.15

## 图像显示方法Image display method

为了将图像显示出来，可以调用SDK中libirparse库中的相关转换函数，将Y14类型的原始数据转换成YUV格式或者RGB格式。

To display the image, you can call the relevant conversion function in the libirparse library in the SDK to convert the raw data of type Y14 to YUV or RGB format.

如果想输出伪彩图像，可以调用SDK中的libirprocess库中的伪彩转换函数。

If you want to output a pseudo-color image, you can call the pseudo-color conversion function in the libirprocess library in the SDK.

如果使用opencv出图，需要将RGB图像转换成BGR图像，相关函数在libirparse库中。

If you use OpencV to generate a graph, you need to convert RGB images into BGR images, related functions in libirparse library.

日历

描述已自动生成