# RTOS平台SPI/I2C接口软件开发资料说明

# SPI/I2C Interface Development Instructions

该版本的SDK主要针对Tiny系列和Mini256系列在STM32或者其他RTOS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。本SDK的开发平台为正点原子的STM32F767开发板，为方便用户开发，全部源码均开放，用户可基于该平台的例程自行设计转接板或者在其他RTOS或者裸机平台中开发本产品。

This version of SDK is mainly for Tiny series and Mini256 series on STM32 or other RTOS platform through SPI interface for image/temperature data transmission and I2C interface for command control. The development platform of this SDK is the STM32F767 development board of alientek. In order to facilitate the development of users, all the source code is open. Users can design the conversion board by themselves based on the routines of this platform or develop this product in other RTOS or bare embedded platforms.

## SDK工程目录介绍The SDK components

本SDK的例程是通过正点原子的NRF24L01的SPI接口官方例程进行修改而成，因此保留了主体的工程结构，相关的文件目录介绍如下：

The routine of this SDK is modified by the official routine of SPI interface of NRF24L01 of alientek, so the main engineering structure is retained. The related file directory is introduced as follows:

1. CORE:芯片相关的启动文件和相关的头文件。

CORE: chip related startup files and related header files.

1. HALLIB：芯片外设的固件驱动库，Src存放的是固件库的.c文件，Inc存放的是固件库的.h文件，每个外设对应一个.c文件和一个.h文件。

HALLIB: the firmware driver library of the chip peripherals. Src stores the. c file of the firmware library, Inc stores the. h file of the firmware library, and each peripheral corresponds to a. c file and a. h file.

1. HARDWARE：具体硬件操作的进一步封装。

HARDWARE: Further encapsulates specific HARDWARE operations.

1. FATFS: fatfs文件系统相关的文件，用于固件升级。

FATFS: File related to fatfs file system, used for firmware upgrade.

1. MALLOC:内存管理相关的文件，用于固件升级。

MALLOC: memory management-related file used for firmware upgrade.

1. SYSTEM:公用的系统、延时和串口函数。

SYSTEM: Common SYSTEM, delay, and serial port functions.

1. USER：主要是main.c用于用户自定义主函数的功能实现。

USER: Mainly main.c is used for the function realization of user-defined main functions.

1. USMART: 串口调试互交组件。

USMART: Serial port debugging interchangeable components.

1. VDCMD:模组相关的参数读写接口实现、命令解析以及底层通信接口适配等。

VDCMD: module related parameters read and write interface implementation, command parsing and underlying communication interface adaptation.

1. OTHER:其他SDK相关的功能实现，包括指令发送、环境变量修正和固件升级功能。

OTHER: other SDK-related functions, including instruction sending, ambient variable modification, and firmware upgrade.

## 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)

## SPI接口说明SPI Interface Description

Tiny1C产品支持SPI图像/温度数据输出，即VOSPI。

目前的Tiny1C产品SPI数据输出配置需要注意的地方如下Notes：

1. 串行同步时钟的空闲状态为高电平。即CPOL=1。

The idle state of the serial synchronous clock is high. The CPOL = 1.

1. 串行同步时钟的第二个跳变沿采样数据。即CPHA=1。

The second hop edge of a serial synchronous clock samples data. The CPHA = 1.

1. STM32F767平台的预分频系数选择SPI\_BAUDRATEPRESCALER\_4，频率太高图像会因为干扰出现很多彩色噪点。

SPI\_BAUDRATEPRESCALER\_4 is selected as the predivider coefficient of STM32F767 platform. If the frequency is too high, the image will have many color noises due to interference.

1. NSS信号由硬件控制。

The NSS signal is controlled by hardware.

1. SPI设置为双线模式。

SPI is set to dual-line mode.

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

VOSPI全称video out SPI interface，作为Tiny1C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 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RS001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。Tiny1C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 tiny1c. 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 tiny1c is 75MHz.

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

#### CPHA=0 Transfer Format

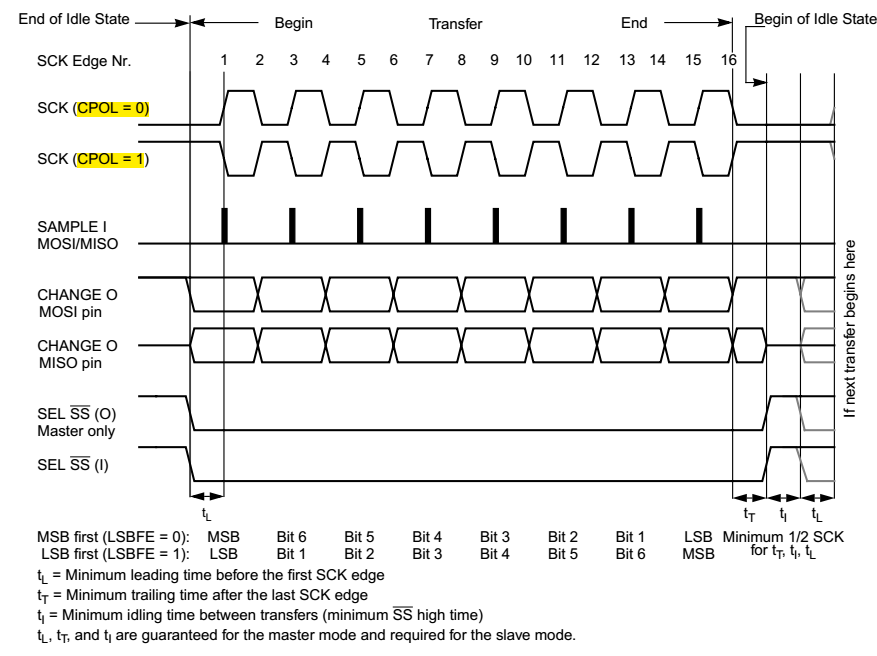


Figure 1. CPHA=0

#### CPHA=1 Transfer Format

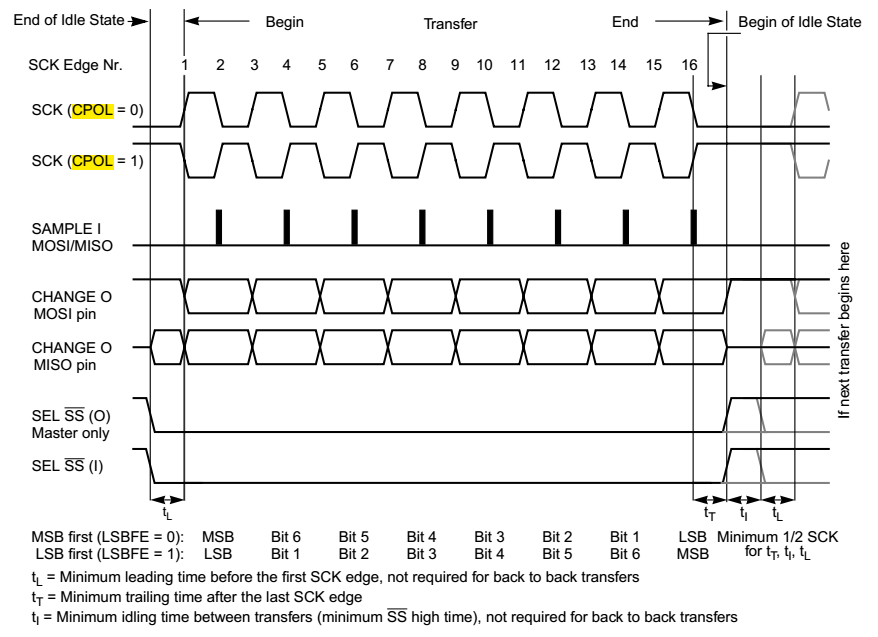


Figure 2. CPHA=1

目前Tiny1C产品采用CPOL=1，CPHA=1。

At present, tiny1c products adopt CPOL = 1 and CPHA = 1.

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

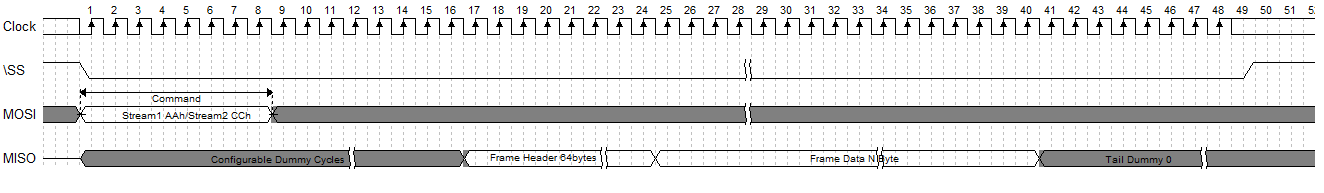


Figure 3. Read a New Frame

“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，范围[512, xxx]，单位SCK cycle，必须为8的整数倍。

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 Tiny1C this bytes are always "0" |
| 8byte | Shutter temp\_H In Tiny1C this bytes are always "0" |
| 9byte | current Gain state(1:high gain   0:low gain) |
| 10byte | Temp\_area In Tiny1C this byte is always "1" |
| 11byte | Over exprosure state In Tiny1C 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

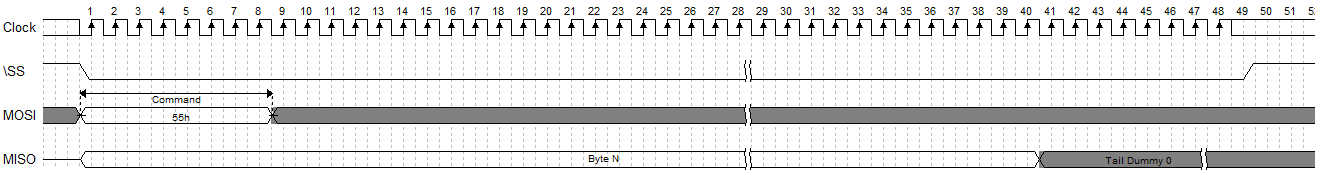


Figure 3. Read a Continued Frame

“Read a New Frame”用于SPI master从RS001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.

本SDK采用的是STM32的SPI4接口模块进行开发，客户可根据需要选择其他的SPI接口，相关的初始化参数见SPI4\_Init函数。获取数据采用的是阻塞模式的HAL库数据收发函数HAL\_SPI\_TransmitReceive，详细用法请看例程。

This SDK uses SPI4 interface module of STM32 for development. Customers can choose other SPI interfaces according to their needs. See SPI4\_Init function for related initialization parameters. The HAL library data HAL\_SPI\_TransmitReceive function is in blocking mode. For details, please see the routine.

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

SPI推荐频率设置SPI recommended frequency Settings：

256\*384分辨率： 256\*384\*2\*8\*25=39.3216Mbps 所以应该大于等于39.3216 Mbps。

256\*192分辨率： 256\*192\*2\*8\*25=19.6608Mbps 所以应该大于等于19.6608 Mbps。

目前支持256\*384/256\*192两种分辨率。具体说明见下表：

|  |  |  |
| --- | --- | --- |
| 分辨率  resolution | 数据  data | 备注  notes |
| 256\*384 | 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 |
| 256\*192 | 图像/温度  Image/Temperature | 可通过Y16\_preview\_start接口切换成温度输出  You can switch to the temperature output through the Y16\_preview\_start interface |

目前sample中根据不同的需要可进行三种不同的图像/温度数据输出方式，分别通过select\_output\_mode进行设置。对于SPI时钟频率不够的平台，可参考图像和温度隔帧输出的方法，具体的说明见下表。客户可参考SDK的相关代码去进行实现。

At present, there are three different image/temperature data output modes in sample according to different needs, which are set through select\_output\_mode respectively. For platforms with insufficient SPI clock frequency, we can refer to the method of image and temperature output at frame intervals. See the following table for detailed explanation. Customers can refer to the relevant code of SDK to implement.

|  |  |  |
| --- | --- | --- |
| 分辨率  resolution | 设置选项  settings | 说明  Description |
| 256\*384 | IMAGE\_AND\_TEMP\_OUTPUT | 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.  也可每帧获取图像数据和隔帧获取温度数据。  Another side，this mode can be adopted to obtain image frames and temperature frames alternately. |
| 256\*192 | IMAGE\_OUTPUT | 输出图像数据Output image data. |
| 259\*192 | TEMP\_OUTPUT | 输出温度数据Output temperature data |

## I2C接口说明I2C Interface Description

作为I2C从设备的7位地址为0x78（0111100b）。遵守MIPI CSI-2中的CCI协议。

The 7-bit address that serves as the I2C slave device is 0x78 (0111100b). Comply with CCI protocol in MIPI CSI-2.

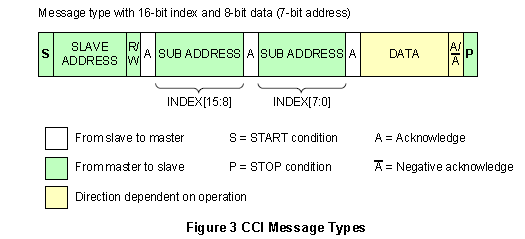
### 1. 芯片协议介绍Chip Protocol Introduction

Tiny1C搭载的ASIC芯片支持的I2C协议介绍如下。

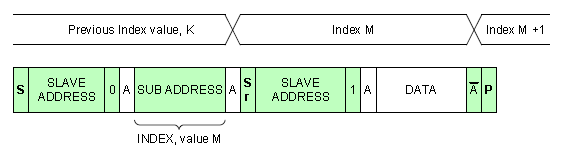
The following describes the I2C protocols supported by the module ASIC

* NDEX（Regsiter地址）是16bit，使用MSB first，高字节在前，低字节在后。

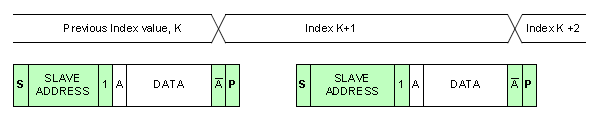
NDEX (Regsiter address) is 16 bits, using MSB first, with high bytes first and low bytes last.



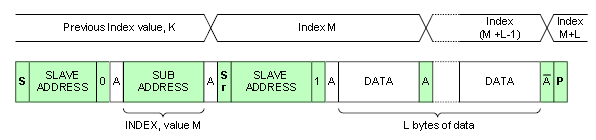
* **支持以下read/write操作The following read/write operations are supported：**
  + 1. 从任意位置读取单个数据（Single Read from Random location）

.

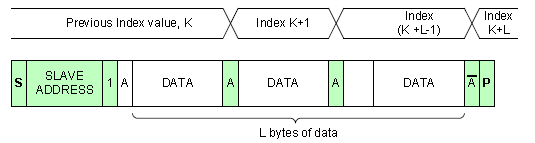
* + 1. 从当前位置读取单个数据（Single Read from the Current Location）



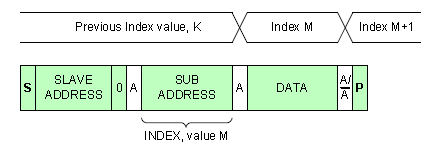
* + 1. 从任意位置读取连续数据（Sequential Read Starting from a Random Location）



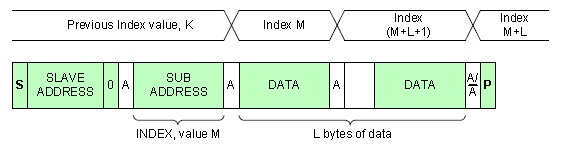
* + 1. 从当前位置读取连续数据（Sequential Read Starting from the current location）



* + 1. 从任意位置写入单个数据（Single write to a random location）



* + 1. 从任意位置写入连续数据（Sequential write to a random location）



* **支持多字节寄存器操作，遵从CCI协议Supports multi-byte register operation and complies with CCI protocol。**
* Multi-byte只支持2，4这两种

Multi-byte supports only 2 and 4

* 对齐：Multi-byte如果是2byte，地址必须以2byte对齐，如果是4byte为4byte对齐，8byte为8byte对齐。

Alignment: multi-byte If the value is 2 bytes, the address must be aligned with 2 bytes. If the value is 4 bytes, the address must be aligned with 4 bytes. If the value is 8 bytes, the address must be aligned with 8 bytes.

* 每种multi-byte分配在不同的地址段。

Each multi-byte is assigned to a different address segment.

* multi-byte register不支持部分访问。

multi-byte Registers do not support partial access.

* data为MSB first，高字节在前，低字节在后。

data is MSB first, with high bytes first and low bytes last.

* multi-byte register access必须做好同步。

multi-byte Register Access must synchronize data.

### 2. 开始出图:preview\_start

参数说明如下表：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| path | src | width | height | fps | mode |
| 数据通路  Data path  Stream0/Stream1 | 探测器数据Sensor data /  固定图案数据Fixed pattern data | 图像宽度Image width | 图像高度Image height | 帧率Frame rate | 输出模式  其中SPI为8  The output mode  Where SPI is 8 |

### 3. 停止出图:preview\_stop

参数path，与preview\_start的参数相对应

The path corresponds to the preview\_start

### 4. 特殊命令接口Special command interface

可参考libirSDK.chm和《Tiny1C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《Tiny1C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 libirSDK.chm and “Tiny1C模组SDK接口说明Interface Description1.2”. 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 table“Tiny1C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 |
| y16\_preview\_start | 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256\*192分辨率下使用  256\*192 Only |
| y16\_preview\_stop | 用于结束数据源切换  Used to end a data source switch | 仅适用于256\*192分辨率下使用  256\*192 Only |
| dpc\_auto\_calibration | 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 |
| rmcover\_auto\_calc | 用于图像锅盖标定  Lid pattern noise correction | 需要等待较长时间，约5s  about 5s |
| 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 |
| 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 |

说明：对于需要等待较长时间的标定命令，可通过宏I2C\_TRANSFER\_WAIT\_TIME\_MS来设置等待时间。

Note: For calibration commands that need to wait a long time, the macro I2C\_TRANSFER\_WAIT\_TIME\_MS can be used to set the wait time.

### 5. 其他命令接口Other command interface

可参考libirSDK.chm和《Tiny1C模组SDK接口V1.2.xlsx》表格。

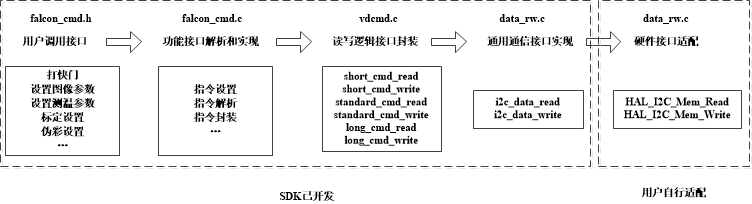
Refer to libirSDK.chm and “Tiny1C模组SDK接口说明Interface Description1.2”.

SDK开发包中的VDCMD目录即为相关用户参数读写函数接口的抽象和封装，采用注册函数的方式隔离底层的通信接口。方便在不同的平台进行移植，也可扩展和适配其他不同的通信接口，如USB和UART等等。

其中，falcon\_cmd.h中包含了所有的用户函数接口，falcon\_cmd.c是所有用户接口的实现，SDK根据芯片的特征将所有读写接口分别划分为short command，standard command和long command三种类型的读写函数，共六种命令抽象接口，在vdcmd.c文件中进行实现，并且再利用这些抽象接口以注册函数的形式调用不同的通用通信接口函数，然后再实现底层函数的适配。用户进行不同平台移植时，只需要将VDCMD文件夹内的文件进行复制后，替换 I2C的底层接口HAL\_I2C\_Mem\_Read/HAL\_I2C\_Mem\_Write即可。命令接口的整体实现框架如下：

The VDCMD directory in the SDK is the abstraction and encapsulation of the user parameter read and write function interface, and uses the registration function to isolate the underlying communication interface. It is convenient for porting on different platforms, and can also be extended and adapted to other different communication interfaces, such as USB and UART.

Falcon\_cmd. h contains all user function interfaces, falcon\_cmd.c is the realization of all user interfaces, and SDK divides all read and write interfaces into short commands according to the characteristics of chips. Standard command and long Command three types of reading and writing functions, a total of six command abstract interface, in vdcmd.c file to implement, and use these abstract interface in the form of registration function to call different general communication interface functions, and then realize the adaptation of the underlying function. When migrating to different platforms, you only need to copy the files in the VDCMD folder and replace the HAL\_I2C\_Mem\_Read/HAL\_I2C\_Mem\_Write interface of I2C. The overall implementation framework of the command interface is as follows:



## SDK其他功能介绍Other Functions

### 固件升级功能Firmware Upgrade:

例程中打开UPDATE\_FW宏再进行编译即可进行模组的固件升级功能。该例程采用的是用SD卡进行固件升级，移植了一个FATFS文件系统，通过文件系统挂载SD卡，初始化后读取SD卡内的固件文件，然后进行通过I2C接口进行固件数据下载。

主要的升级步骤如下：

To upgrade the firmware of the module, open the UPDATE\_FW macro and compile it. This routine uses SD card to upgrade firmware, transplants a FATFS file system, mounts the SD card through the file system, reads the firmware file in the SD card after initialization, and then downloads the firmware data through I2C interface.

The main upgrade steps are as follows

1. 挂载SD卡

Mount the SD card

1. 动态申请读写缓冲区

Dynamically applying for read/write buffers

1. 获取设备状态，如果不是ROM状态则进行切换，直到切换成ROM状态为止

Gets the device state, if not the ROM state, then switches until the ROM state is switched

1. 检查flash状态

Checking the Flash Status

1. 擦除flash代码区的扇区

Erases sectors in the Flash code area

1. 校验flash代码区的扇区

Verifies the sector of the Flash code area

1. 循环读取SD卡中的固件数据，写入flash并读出校验

Read the firmware data in the SD card, write to the flash, read and verify

1. 完成后写入cache tag

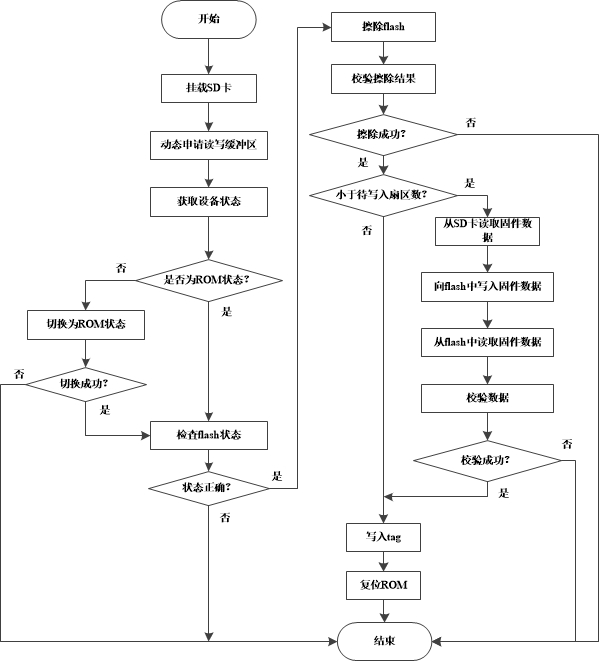
Then write the cache tag

1. 复位模组

Reset the module

1. 释放读写缓冲区

Release the read/write buffer



SD卡更新固件流程图

### 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《用户开发标定》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"

### 测温二次标定与锅盖标定Secondary calibration& Lid pattern noise correction

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和锅盖标定。该功能在模组固件中实现，详细的使用说明见《用户开发标定》目录下的《二次标定与锅盖标定》。客户应该先进行锅盖标定再进行测温二次标定。

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& Lid pattern noise correction is required at the end of the machine. This function is realized in the module firmware. For detailed instructions, see "Secondary calibration& Lid pattern noise correction" in the directory of "User calibration instructions". The customer should first calibrate the Lid pattern noise and then Secondary calibration.

### 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。该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。详细的修正方案见《用户开发标定》目录下的《环境变量修正》。对于STM32平台而言，SDK的测温修正需要从SD卡读取距离修正表。同时，SDK的工程依赖了一个libirtemp库，该库的源码并不开放。如果客户使用该方法，可以提供厂商相关的编译工具和编译方法，由厂商编译完成后再由客户调用开发。

The temperature measurement of the module is affected by Emissivity, Atmospheric transmissivity, ambient reflection temperature, ambient atmospheric temperature, target distance, etc. The SDK provides users with a correction method, which can make the temperature measurement accuracy closer to the real value. For detailed instructions, see "Ambient variable correction" in the directory of "User calibration instructions". For the STM32 platform, the SDK's temperature correction needs to read the correction table from the SD card. At the same time, the SDK project relies on a libirtemp library, which is not open source. If the customer uses this method, you can provide the compilation tools and methods related to the manufacturer. After the compilation is completed, the customer invokes the development.

### 防跌落保护Fall protection

由于红外模组双稳态快门的物理特性，当模组发生某个方向上的跌落或者撞击时，快门会有概率闭合上，导致红外图像异常。因此，模组固件内置防跌落功能，防跌落功能的实现策略大致为：检测图像画面，对画面的均匀性进行统计，如果发现画面输出值的离散程度小于设定的阈值，则固件会通过GPIO发送脉冲强制打开快门。通过set\_prop\_auto\_shutter\_params函数的SHUTTER\_PROP\_PROTECT\_SWITCH参数，可以设置该功能的开关。通过SHUTTER\_PROTECT\_THR\_HIGH\_GAIN可以设置高增益状态下的阈值，通过SHUTTER\_PROTECT\_THR\_LOW\_GAIN可以设置低增益下的阈值。

The shutter blade may open/close abnormally during strong mechanical shock (such as fall), and a monitoring process is designed in the firmware to correct the abnormal shutter switch in time. This feature is called fall protection. The strategy is as follows: detect the image and make statistics on the uniformity of the picture. If the dispersion degree of the picture is less than the set threshold, the firmware will send pulses through GPIO to force the shutter to open. You can switch this function on or off by using the SHUTTER\_PROP\_PROTECT\_SWITCH parameter of the set\_prop\_auto\_shutter\_params function. SHUTTER\_PROTECT\_THR\_HIGH\_GAIN allows you to set a threshold in the high gain state, and SHUTTER\_PROTECT\_THR\_LOW\_GAIN allows you to set a threshold in the low gain state.

快门防跌落

图3.快门防跌落保护策略

C:\Users\dell\Desktop\SDK资料说明\快门防跌落英文.emf

Figure 3. Shutter fall protection strategy

### 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falcon\_cmd.h库中的tpd\_get\_point\_temp\_info、tpd\_get\_line\_temp\_info，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tpd\_get\_max\_min\_temp\_info函数获取整帧的温度最大/最小值信息。

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

## 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

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 |

可调用temp\_value\_converter函数接口将原始温度数据转换成摄氏温度。

The temp\_value\_converter function interface can be called to convert raw temperature data to Celsius.

## 图像显示方法Image display method

为了将图像显示出来，可以调用YUV422ToRGB565函数将yuv422数据传输到RGB LCD显示屏进行显示。

To display the image, the YUV422ToRGB565 function can be called to transfer the YUV422 data to the RGB LCD screen for display