

## 产品说明

P8864-SMD-B15系列产品是3D TOF微型传感器模块，集成多达8x8 64像素3D ToF（飞行时间）传感器、微处理器和可选PMU（电源管理单元）。

该模块提供每个像素上的深度和强度信息，可测量5米（请参阅详细的零件号说明），适于室内或室外环境。

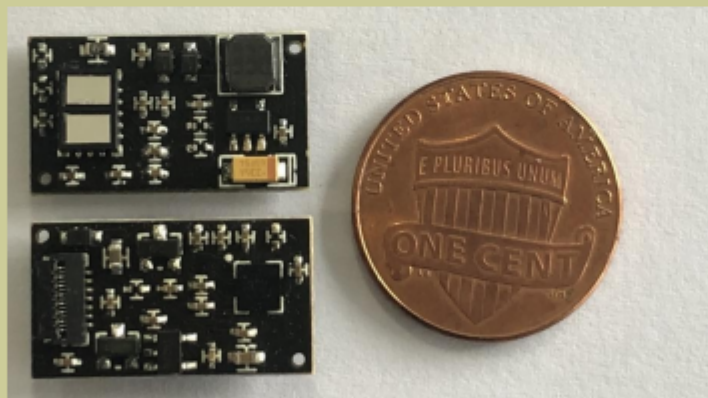
产品符合眼睛安全标准和各种国际安全标准。

## 产品亮点

- 基于TOF原理的3D面阵
- 8x8 像素
- 视野角达32°
- 微型模块尺寸
- 人眼安全，符合IEC60825-1 I类安全

## 产品应用

- 3D传感器
- 手势识别
- 人机交互
- 机器人导航和避障



**参数**

(25℃以下均为典型值。所有参数均基于PUYU评估系统，包括硬件、具有适当校准的固件和GUI。)

参数	注释	值	单位
工作距离	90%反射率	最小0.1-1.5米，典型值0.1-2米	m
光照强度		待定	KLUX
视野角FOV		16°	°
距离分辨率*1	0.1-0.5m:	1	cm
	0.5-1.5m:	2	
	1.5-2m	4	
距离精度*1	0.1-1米:	4	cm
	>1m:	4%	
距离可重复性*2	0.1-0.5米:	1	cm
	>0.5米:	1%	
发射波长		940	nm
供电电压		+5	V
功耗*3		典型值360 最大值420	mW
操作温度		-10-55	℃
存储温度		-20-80	℃

注意:

1. 基于平均 50 次测量。
2. 基于50个测量结果，1 $\sigma$ 。
3. 基于10 fps的类型，由-SMD测量

## 像素区

### TOF 传感器像素区域

像素标识为（行、列），8x8示例如下所示。像素的大小由零件号标识。

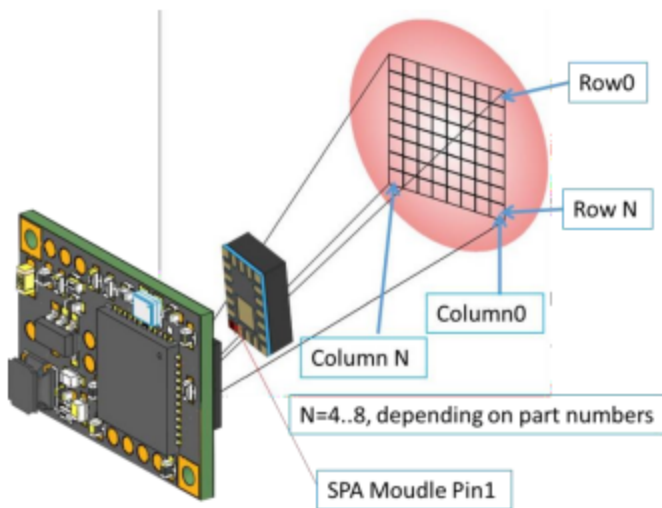
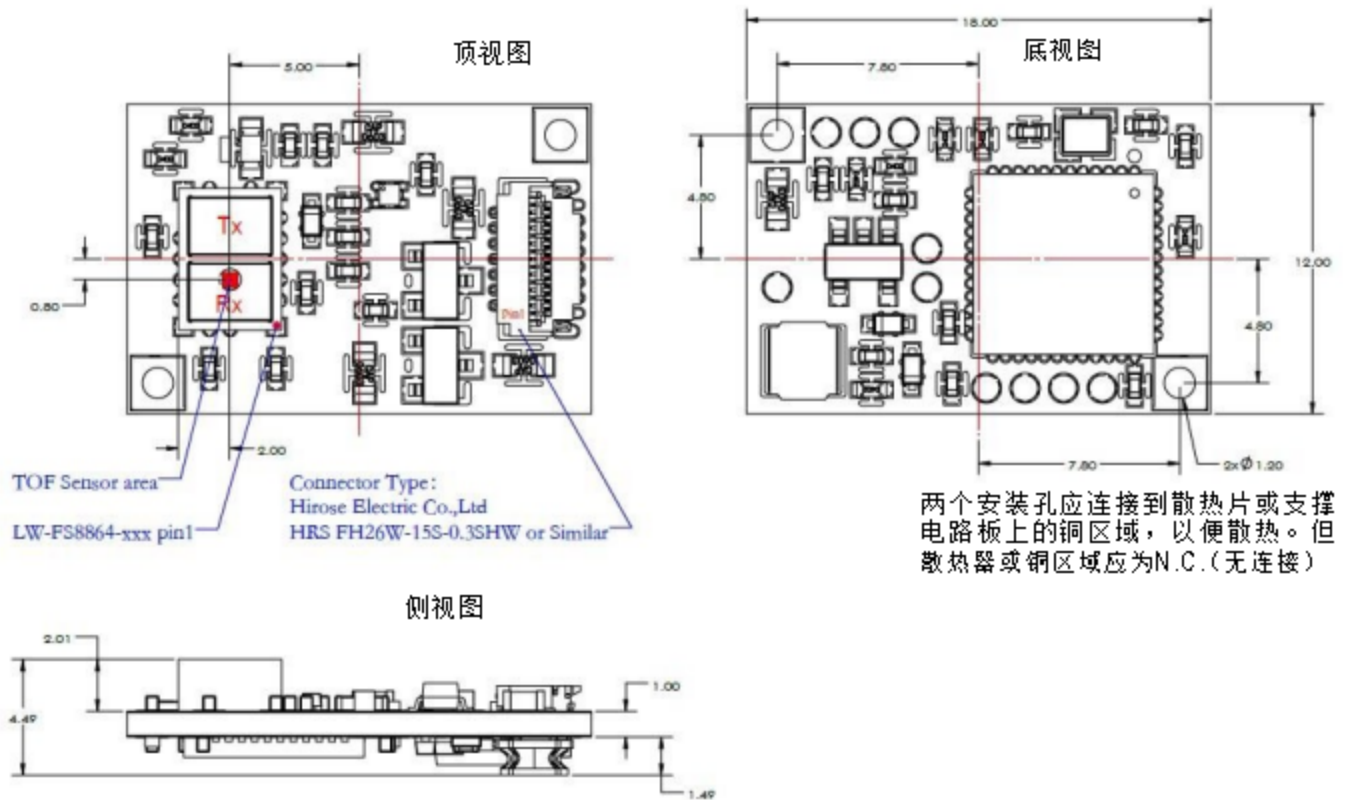
0,0	0,1	0,2	0,3	0,4	0,5	0,6	0,7
1,0	1,1	1,2	1,3	1,4	1,5	1,6	1,7
2,0	2,1	2,2	2,3	2,4	2,5	2,6	2,7
3,0	3,1	3,2	3,3	3,4	3,5	3,6	3,7
4,0	4,1	4,2	4,3	4,4	4,5	4,6	4,7
5,0	5,1	5,2	5,3	5,4	5,5	5,6	5,7
6,0	6,1	6,2	6,3	6,4	6,5	6,6	6,7
7,0	7,1	7,2	7,3	7,4	7,5	7,6	7,7

8x8 pixels

## 尺寸

L18 x W12 x H4.49 mm

单位：mm



## 引脚定义

引脚Pin	引脚定义
1	空
2	GND
3	GND
4	空
5	空
6	GND
7	空
8	空
9	GND
10	UART RXD
11	GND
12	UART TXD
13	+5V
14	+5V
15	空

## UART 通信

通信接口在LVTTTL电平（3.3V）上使用8位UART标准，其工作参数如下：

参数	参数
波特率	921600bps
起始位	1
数据	8
停止位	1
校验位	无



UART以主从模式工作，应用主控为主，传感器模块为从。

命令格式：

命令数据包的固定长度为14个字节：一个起始字节（值0xF5），后跟1个字节的命令ID（CMD）、8个字节的与命令相对应的参数以及4个带有32位CRC的结束字节。

1 byte	1 byte	8 bytes								4 bytes			
0xF5	CMD									32位CRC校验			
		LSB							MSB	LSB			MSB

响应格式：

应答数据包具有可变长度：一个起始字节（值0xFA），后跟1字节类型定义、2字节长度定义n、n字节数据和4个带32位CRC的结束字节。。

1 byte	1 byte	2 bytes		N bytes								4 bytes				
0xFA	TYPE	Length N		DATA								32位CRC校验				
		LSB	MSB	LSB								MSB	LSB			MSB

### CRC32:

CRC即循环冗余校验码（Cyclic Redundancy Check）：是数据通信领域中最常用的一种查错校验码，其特征是信息字段和校验字段的长度可以任意选定。循环冗余检查（CRC）是一种数据传输检错功能，对数据进行多项式计算，并将得到的结果附在帧的后面，接收设备也执行类似的算法，以保证数据传输的正确性和完整性。

- Byte-wise CRC 32
- Init value: 0xFFFFFFFF
- Xor value: 0x00000000
- Polynom: 0x04C11DB7

CRC 计算功能:

```
uint32_t CrcCalc::calcCrc32(const uint8_t *data, const uint32_t size)
```

```
{
    uint32_t crc = initialValue;
    for(uint32_t i = 0; i <
        size, i++)
    {
        crc = calcCrc32Uint8(crc, data[i]);
    }
    return crc ^ xorValue;
}
```

```
uint32_t CrcCalc::calcCrc32Uint8(uint32_t crc, uint8_t data)
```

```
{
    int32_t i;
    //This shift is done to make it compatible to the STM32
    hardware CRC crc = crc ^ (data << 24);
    for(i = 0; i < 8; i++)
    {
        if(crc & 0x80000000)
        {
            crc = (crc << 1) ^ polynom;
        }
        else
        {
            crc = (crc << 1);
        }
    }
    return(crc);
}
```

**命令列表:****SET\_POWER [0x40]**

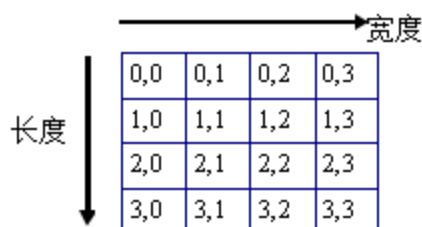
将传感器模块置于上电或关机模式，间隔时间 $t_{PROC}$ 返回数据。先上电，才能响应命令工作。

<b>参数</b>	byte 0: 0x00 断电; 0x01 上电; 其他:0x00
<b>命令示例</b>	0xF5  0x40  0x01 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x9C 0xD7 0xD6 0x91  (电源启用)
<b>响应类型</b>	0x00, 确认
<b>响应数据</b>	0 byte
<b>上电响应时间</b>	$t_{PROC} < 200 \text{ ms}$
<b>断电响应时间</b>	$t_{PROC} < 30 \mu\text{s}$
<b>响应示例</b>	0xFA  0x00  0x00 0x00  (0 bytes)  0xB2 0xAB 0xFC 0xE8

**GET\_REGION\_LENGTH\_WIDTH[0x57]**

获取ToF传感器像素区域的长度和宽度。

<b>参数</b>	所有字节 0x00
<b>命令示例</b>	0xF5  0x57  0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00  0xD7 0x7C 0xB3 0x05
<b>响应类型</b>	0x0C, 数据
<b>响应时间</b>	~ 40 $\mu\text{s}$
<b>响应示例</b>	0xFA  0x0C  0x02 0x00  0x08 0x08  0x7B 0xA5 0x60 0x0F   (8x8 像素, 长度 = 8 和宽度 = 8)
	0xFA  0x0C  0x02 0x00  0x04 0x04  0xE4 0x5C 0x00 0xF5   (4x4像素, 长度=4和宽度=4)





### GET\_REGION\_DISTANCE\_AMPLITUDE [0×29]

启动新的距离和ToF振幅采集，并在 t<sub>PROC</sub>后返回ToF 传感器像素区域的结果或状态。ToF振幅是距离结果的质量指标。

参数	所有字节 0x00
命令示例	0xF5  0x29  0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00   0xCB 0xFA 0xAB 0xE7
响应类型	0x0B, 数据
响应数据	Length×width×4 bytes 距离值，然后是length x width x 4 bytes 振幅值，共计: 2x length×width×4 bytes。例如，ToF 传感器像素区域的大小为 4x4 (16 像素)，数据为 16 x 4 字节距离，然后是16 x 4 字节振幅，总计 128 字节。 每个读出顺序: 从 (0, 0)、(0, 1)、(0, n)、(1, 0) 开始，以 (n, n) 结尾。
距离	byte 0~3: 距离, 0.1mm/LSB 或状态, 32 bit 无符号整数
振幅	byte 0~3: TOF振幅或状态, 32 bit 无符号整数。 以下距离数据，并不是距离数据，而是表示像素上的"错误"。 状态: 16'001'000d: TOF弱振幅 16'002'000d: 溢出 16'003'000d: 饱和 16'004'000d: 预留 16'005'000d: 流入不足 16'006'000d: TOF高振幅 以下ToF 振幅数据表示像素太"暗"或太"亮", 此像素的距离数据可能不准确。 Values < 100 LSB: 距离噪声显著: Values > 1900 LSB: 距离可能包含相当大的错误。 100 到 1900 之间的 TOF 振幅表示良好的距离数据。
响应时间	t <sub>PROC</sub> <25ms
响应示例	0xFA 0x0B 0x80 0x00 0x24 0x0F 0x00 0x00...0xE8 0x04 0x00 0x00... (共128bytes CRC (128bytes) (距离 387.6mm..振幅 1256 LSB) 0xFA  0x0B 0x00 0x02  0x24 0x0F 0x00 0x00...0xE8 0x04 0x00 0x00... (共512bytes) CRC (512 bytes) (距离 387.6mm..振幅 1256 LSB)

**SET\_ADAPTIVE\_INTEGRATION [0x55]**

打开自适应积分时间功能。积分时间将在测量过程中自动确定。根据测量对象的不同，每次测量的时间可能不同。打开模块后，**默认启用自适应积分时间功能**。启用自适应积分时间功能后，模块不会响应SET\_INTEGRATION\_TIME\_DIS [0x00] 命令。通过打开自适应积分时间功能，典型采集速度约为 12fps，这意味着模块可以在一秒钟内响应大约 12 条命令，如 GET\_REGION\_DISTANCE\_AMPLITUDE [0x29]。

<b>参数</b>	Byte 0: 0x00 开启自适应积分时间功能； 0x01 关闭自适应积分时间功能。
<b>命令示例</b>	0xF5 0x55 0x01 0x00 0x00 0x00 0x00 0x00 0x00 0x00   0x66 0XE9 0XD2 0xBE 关闭自适应时间功能 0xF5 0x55 0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00   0xAC 0X35 0X73 0xE5 开启自适应时间功能
<b>响应类型</b>	0x00, 确认
<b>响应数据</b>	0 bytes
<b>响应时间</b>	~ 40 $\mu$ s
<b>响应示例</b>	0xFA   0x00   0x00 0x00   (0 byte)   0xB2 0xAB 0xFC 0xE8

**SET\_PERIOD\_FREQUENCY [0x58]**

确定两次自适应积分时间之间的采集。此命令的目的是在某些应用中增加帧率fps。默认情况下，命令 SET\_ADAPTIVE\_INTEGRATION[0x55]在每次测量时启用自适应积分时间功能。如果需要更高的帧率fps，则可以使用命令SET\_PERIOD\_FREQUENCY[0x58]增加一些采集，而无需更新积分时间。例如，如果参数为5，积分时间在第一次采集时自适应更新，并保留在接下来的四次采集中，然后积分时间在第六次采集时再次自适应更新，并保存在接下来的四次采集中，依此类推。通过选择一个近似参数，fps最高可达30fps。具体取决于应用环境。参数可以是1到9之间的无符号整数。

<b>参数</b>	Byte 0: 自适应积分时间段内的采集；其他：0x00
<b>命令示例</b>	0xF5   0x58   0x05 0x00 0x00 0x00 0x00 0x00 0x00 0x00   0xEB 0XA1 0X97 0x6D 周期=5，一次采集具有自适应积分时间，然后在不改变积分时间的情况下进行四次采集
<b>响应类型</b>	0x00, 确认
<b>响应数据</b>	0 byte
<b>响应时间</b>	~ 40 $\mu$ s
<b>响应示例</b>	0xFA   0x00   0x00 0x00   (0 bytes)   0xB2 0xAB 0xFC 0xE8

**SET\_INTEGRATION\_TIME\_DIS [0x00]**

人为设置距离测量的积分时间，并在tTROC确认后返回。只要打开电源或设置新时间，积分时间就会被存储和使用。范围：1...1600μs。0不是有效值。要手动设置积分时间，必须先禁用自适应积分功能。

<b>参数</b>	Byte 0, 1: 积分时间（微秒），16位无符号整数
<b>命令示例</b>	0xF5  0x00  0x00 0x1E 0x00 0x00 0x00 0x00 0x00 0x00   0xD9 0x85 0x1A 0x99 30μs积分时间
<b>响应类型</b>	0x0 0, 确认
<b>响应数据</b>	0 byte
<b>响应时间</b>	~ 40 μs
<b>响应示例</b>	0xFA  0x00  0x00 0x00  (0 byte)  0xB2 0xAB 0xFC 0xE8

**GET\_TEMPERATURE [0x4A]**

获取上次距离采集期间的芯片温度。

<b>参数</b>	所有字节 0x00
<b>命令示例</b>	0xF5  0x4A   0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00   0x18 0x41 0xF5 0xA4
<b>响应类型</b>	0xFC, 数据
<b>响应数据</b>	2字节: 温度, 0.01°C/LSB , 16位2的补码有符号整数。
<b>响应时间</b>	~ 40 μs
<b>响应示例</b>	0xFA  0xFC  0x02 0x00  0x47 0x13   0x4F 0xEE 0x12 0x1F   (温度 49.35°C)

**GET\_FIRMWARE\_VERSION[0x49]**

获取固件版本（V.S.S）。

<b>参数</b>	所有字节 0x00
<b>命令示例</b>	0xF5  0x49  0x00 0x00 0x00 0x00 0x00 0x00 0x00 0x00   0x05 0xA2 0x35 0xB6
<b>响应类型</b>	0xFE, 数据
<b>响应数据</b>	4 bytes Byte 0,1: Subversion S, 16位无符号整数 Byte 2: 传感器型号 S, 8位无符号整数 Byte 3: 版本 V, 8位无符号整数
<b>响应时间</b>	~ 40 μs
<b>响应示例</b>	0xFA  0xFE  0x04 0x00  0x01 0x00 0x08 0x03  0x35 0x33 0x03 0x46 (版本 3.8.1)

**DATA\_NACK**

仅系统响应：命令未被接受或未知。

<b>响应类型</b>	0x01, 未确认
<b>响应数据</b>	0 byte
<b>响应示例</b>	0xFA   0x01   0x00 0x00   (0 byte)   0x35 0x07 0x24 0xE9

**DATA\_ERROR**

仅系统响应：在执行命令期间发生错误。响应，而不是所需的数据

<b>响应类型</b>	0xFF, 错误
<b>响应数据</b>	2 bytes bit 0~14: 错误序号。再试一次。如果错误仍然存在，请联系您的销售负责人。 Bit 15:0
<b>响应示例</b>	0xFA   0xFF   0x02 0x00   0x03 0x00   0x94 0xF6 0x35 0x81

**修订历史记录**

修订	更改
Rev1初步	初始版本, 有限使用
Rev2初步	更新格式
Rev3初步	添加UART通信部分
Rev4初步	更新规范, 更新机械图纸, 更新UART通信
Rev5初步	更新参数和测试条件
Rev6初步	更新CRC32示例
Rev7初步	更新UART通信
Rev8初步	更新参数, 更新UART通信

**文件识别****数据表**

与本产品和本文件所述应用或设计相关的信息被认为是可靠的，但此类信息仅作为指南提供，不对本文件中的任何错误或本文描述的应用或设计。

**谨慎****静电敏感器件**

除非在无静电工作站上，否则不要打开软件包或进行处理

