

YGW-R1 产品手册



YGW-R1

目录

一、 产品说明	1
二、 技术参数	1
1, 红外摄像头规格	1
2, 可见光摄像头规格	2
3, 毫米波雷达规格	2
4, 输入输出接口说明	3
5, 整机规格	4
三、 设备连接及使用	5
3.1 设备上电步骤	5
3.2 设备下电步骤	6
3.3 数据采集说明	7
四、 多维像素数据定义	7
4.1 多维像素数据包说明	7
4.2 Meta data 结构	8
4.3 Camera data 结构	11
4.4 IR data 结构	12
4.5 Radar data 结构	12
4.6 IR 和可见光 Camera 融合参数说明	15
4.7 同步触发说明	16

一、产品说明

昱感微融合产品 YGW-R1 集成了毫米波雷达 Radar、可见光摄像头 Camera、红外摄像头 IR，多传感器融合后生成时空对齐的多维像素数据，通过 GMSL2 接口输出。本产品为客户提供更加直接、高效和可扩展的环境与事件感知能力。

二、技术参数

产品正等测图如图 2-1 所示，红色数字标注为传感器序号：

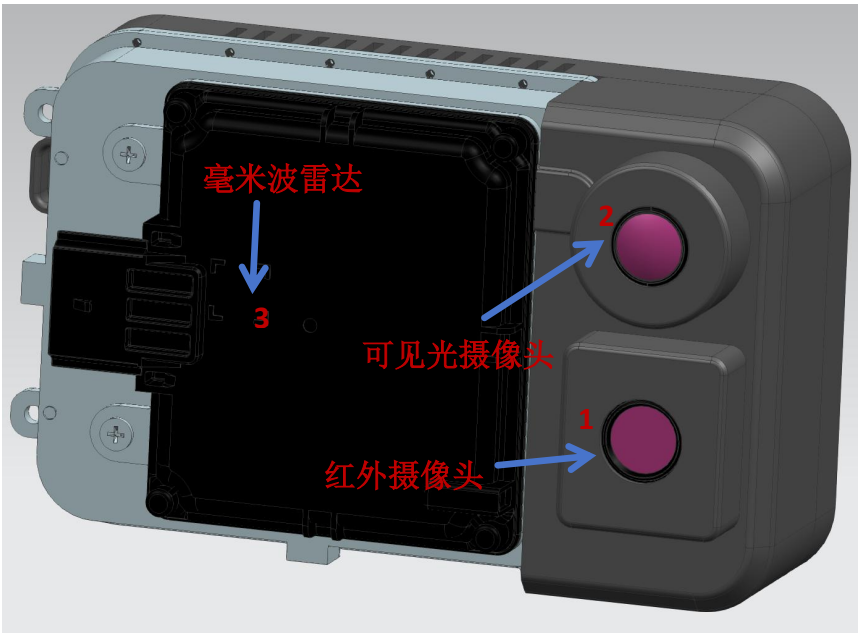


图 2-1 产品正等测图

1，红外摄像头规格

表 2-1 红外摄像头参数

分辨率	640*512
帧率	30fps
HFOV/VFOV	48° / 37°
焦距/光圈	9.1mm/F1.0

2，可见光摄像头规格

表 2-2 可见光摄像头参数

分辨率	8M (3840*2160)
帧率	30fps
HFOV/VFOV	120° / 65°
动态范围	≥120dB
焦距/光圈	4.0mm/F1.6

3，毫米波雷达规格

表 2-3 毫米波雷达参数

简介	77GHz 成像雷达传感器, 6 发 8 收, 48 通道先进级联雷达天线设计
测距	0.25m ~285m
距离精度	±0.4m (>150m) ±0.2m (>75m and <150m) ±0.1m (<75m)
速度范围	-115~55 (m/s)
速度分辨率	0.13 m/s
速度精度	±0.06m/s
水平角度范围	± 12deg (230m) ±60deg (40m)
俯仰角度范围	± 10deg (200m)
角度精度	Az: ±0.2deg@±12deg ±0.8deg@±60deg ±0.4deg@±45deg El: ±0.5deg@±10deg
刷新周期	55ms

4，输入输出接口说明

产品后视图如图 2-2 所示，图中红色字母为产品对外接口标号，具体描述如下：



图 2-2 产品后视图

- (1) **接口 A:** 多传感器融合即融合帧数据输出口，采用防水型 Fakra 公口，支持 GMSL2 协议，产品标配一根 3m 长的双头防水型 Fakra 母头线如图 2-3 所示。



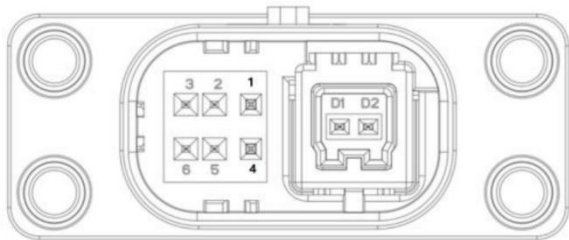
图 2-3 GMSL 线束



图 2-4 TE 线束

(2) **接口 B:** 供电及调试接口，采用 TE 接口，产品标配一根 3m 长的 TE 线束

如图 2-4 所示，具体管脚定义如图 2-5 所示：



编号	信号	说明
1	VCC	12V 电源正
2	TTL-Tx	3.3V 电平
3	预留	/
4	GND	12V 电源负
5	TTL-Rx	3.3V 电平
6	预留	/
D1	MDI-N	车载以太网
D2	MDI-P	车载以太网

图 2-5 TE 管脚定义

其中千兆车载以太网接口 (D1、D2), 主要有以下 3 个功能，不支持融合帧数据输出。

- 授时；
- 固件升级；
- 调试（暂不对客户开放）

5，整机规格

- (1) 尺寸（长高深）：175mm * 100mm * 60mm
- (2) 单机重量:1kg
- (3) 功耗：22W
- (4) 供电要求：12V 3A
- (5) 对时方式：本品支持 PTP/gPTP 时间同步

三、设备连接及使用

3.1 设备上电步骤

- (1) 分别使用配套的 GSML 线束和 TE 线束连接产品背面的“接口 A”与“接口 B”；
- (2) 电源转换盒正反面如图 3-1 所示，主要使用到 DC12V、串口和 MX3.0mm-4P 三个接口；
将 TE 线束接到电源转换盒上（只连接黑色的 MX3.0mm-4P 的端子即可，绿色的 KF2ED G3.81mm-2P 端子可不连接）；

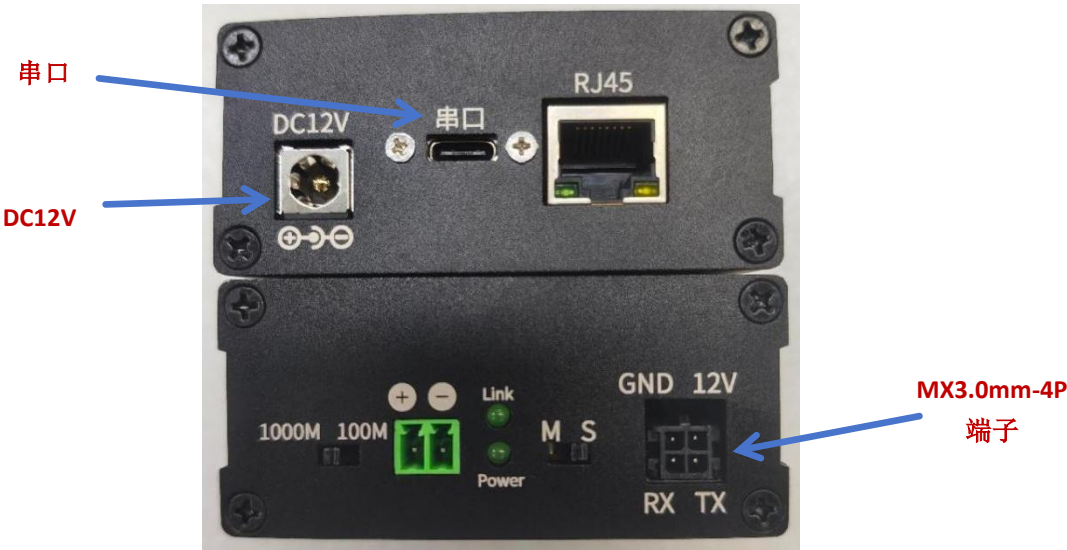


图 3-1 电源转换盒（正反面）

- (3) 将电源转换盒的串口通过 Type-C 线连接到电脑，波特率设置 115200, 给电源转换盒通电，电源要求电压为 12V、电流 $\geq 3A$ 的电源（本产品标配 12V 5A 的电源适配器），
如下图 3-2 和图 3-3 分别为 UART 串口连接设置以及设备完整连接图。

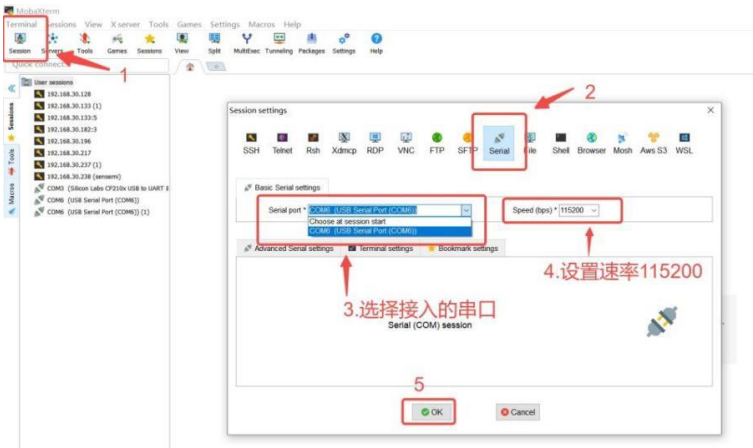


图 3-2 UART 串口连接设置



图 3-3 设备完整连接效果

- (4) 上电后串口会持续打印系统初始化信息，约一分钟左右此时串口打印“TX START”字样如图 3-4 所示，表明产品初始化完成，开始从 GSML2 口即“接口 A”输出融合后的多维像素数据。

```

root@xlnx-3eg:/run/media/mmcblk0p1# ==gdcmapfile address:0x3ED1EAE4===
intc-pl2ps-warp-done set before get priority = 0xA0, trigger = 0x1
intc-pl2ps-warp-done set after get priority = 0xA0, trigger = 0x3
trigger: internal
p_ddr_res_warpOut->nums_buf:9
set-before:
set-after:111111111
warp out buffer now is all released
=====TX START=====
[ 72.466554] systemd-journald[206]: Time jumped backwards, rotating.

```

图 3-4 串口打印的初始化信息

3.2 设备下电步骤

- (1) 先将 GSML 线束从“接口 A”移除；
- (2) 再将电源转换盒的 12V 电源断开，后拔除电源转换盒上的串口线；
- (3) 最后将 TE 线束从“接口 B”拔出。

3.3 数据采集说明

- (1) 若是连接车载域控制器，直连“接口 A”即可；
- (2) 若要连接电脑，则需要一张 GSML 图像采集卡。

四、多维像素数据定义

4.1 多维像素数据包说明

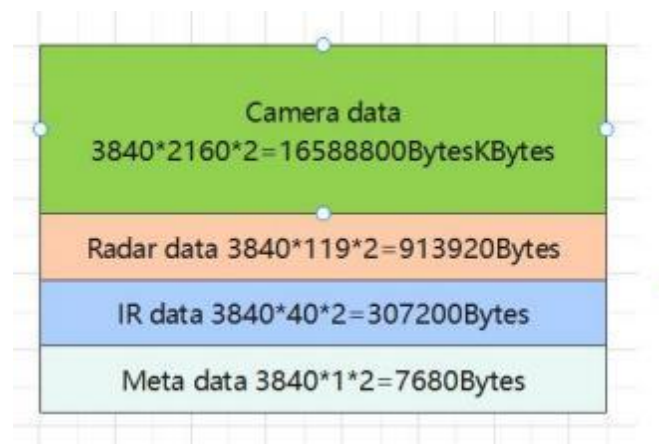


图 4-1 多维像素数据包

- (1) 昱感微多维像素大小会根据可见光 camera 分辨率改变而调整，目前支持 8M 可见光 camera。
- (2) 昱感融合数据包通过 MIPI TX CSI2 发送，每帧数据为 $3840 \times 2320 \times 2$ Bytes (8M camera, VGA IR)，不同 IR 和 CAMERA 分辨率帧大小不一样。
- (3) 每帧数据包格式如上图所示。内容为：可见光 camera 数据+雷达点云数据（激光/毫米波）+IR camera 数据+Meta data 数据。数据传输帧率为 30fps，对于 Radar 数据，如果是激光雷达，10fps 有效，毫米波雷达 15-20fps 有效。
- (4) 每帧数据包采用小端模式，采用默认对齐方式。
- (5) 可见光 camera data 大小为 $3840 \times 2160 \times 2 = 16588800$ Bytes (8M，分辨率不同可变)，雷达点云数据格式为 $3840 \times 119 \times 2 = 913920$ Bytes (实际长度看 metadata info)，IR data 大小为 $3840 \times 40 \times 2 = 307200$ Bytes (VGA，分辨率不同可变)，Meta data 大小为 $3840 \times 1 \times 2 = 7680$ Bytes (固定)。

(6) 下图是标准的摄像头图像输出时序，MIPI 接口硬件处理、隐藏了行场同步等信息，软件可以直接拿到标准的数据。本品可以等效看成一个标准的 camera，只是分辨率略有差别。

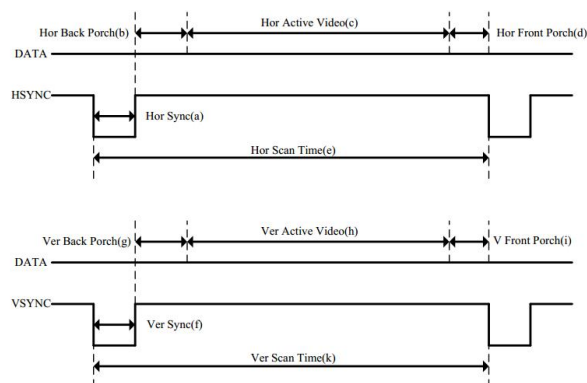


图 4-2 摄像头数据包输出时序

4.2 Meta data 结构

表 4-1 Meta data 数据包结构

起始字节	字节长度	类型	定义	备注
0	4	uint32	iMagicNumber	Magic number (有效帧: 0X59475700, 无效帧: 0xFFFFFFFF)
4	8	uint64	iCurrentFrameID	帧号(Frame)
12	2	uint16	iFwVersion	固件版本号
14	2	uint16	iMdVersion	数据格式定义版本号
16	4	uint32	iRadarType	radar 类型(激光: 0、毫米波: 1)
20	4	uint32	iCustomer ID	客户ID
24	4	uint32	iMetaDataValidSize	Metadata 有效数据字节数
28	4	uint32	iCameraDataOffset	Camera data 在融合数据包内偏移地址
32	4	uint32	iCameraDataValidSize	Camera data 有效数据字节数
36	4	uint32	iCameraDataType	Camera data 数据类型 (0: YUV422; 1: 420)
40	4	uint32	iCameraResolution	Camera 分辨率 (0: 2M; 1: 3M; 2: 5M; 3: 8M)
44	4	uint32	iRadarDataOffset	Radar data 在融合数据包内偏移地址
48	4	uint32	iRadarDataValidSize	Radar data 有效数据字节数

52	4	UInt32	iRadarDataDropNum	Radar data 被丢弃的点云个数
56	4	uint32	iRadarValid	本帧雷达数据数据是否有效(无效: 0, 有效: 1)
60	4	uint32	iIrDataOffset	IR data 在融合数据包内偏移地址
64	4	uint32	iIrDataValidSize	IR data 有效数据字节数
68	4	uint32	iIrResolution	IR Data 分辨率 (0: VGA; 1: QVGA)
72	4	uint32	iMultiDimensionType	雷达像素点云格式 (0: 稀疏; 1: 密集)
76	8	uint64	iCameraTimestamp	Camera 采样时间戳, 纳秒:bit[63:32], 秒:bit[31:0]

84	8	uint64	iRadarTimestamp	Radar 采样时间戳, 纳秒:bit[63:32], 秒:bit[31:0]
92	8	uint64	iIRTimestamp	IR 采样时间戳, 纳秒:bit[63:32], 秒:bit[31:0]
100	8	uint64	iTxTimestamp	数据发送时间戳计数值, 纳秒:bit[63:32], 秒:bit[31:0]
108	4	uint32	iRadarSyncMode	0: Radar 和 camera 时间同步后发出; 1: Camera 和雷达按照各自速度尽快发出, 不同步等待; 2: Radar 收满一帧融合发出
112	4	uint32	iRadarFrameNum	1: 本帧多维像素有 1 帧雷达数据; 2: 本帧多维像素有 2 帧雷达数据; 0: 本帧多维像素是上一帧的雷达数据的延续
116	4	uint32	iRadarDataend1	Radar frame1 发送结束标志; 0: 未结束; 1: 结束
120	4	uint32	iRadarDataend2	Radar frame2 发送结束标志; 0: 未结束; 1: 结束
124	4	uint32	iRadarDataOffset2	Radar frame2 在融合数据包内偏移地址
128	4	uint32	iRadarDataValidSize2	Radar frame2 有效数据字节数
132	8	uint64	iRadarTimestamp2	Radar frame2 采样时间戳, 纳秒:bit[63:32], 秒:bit[31:0]
140	8	uint64	iRadarFrameID	雷达帧序列号 (这是第几帧雷达数据, 如果本帧多维像素有两帧雷达数据, 此位置显示第一帧的序列号)
148	4	uint32	iWarningInfo	错误和警告信息
152	8	uint64	iRadarAlignNum	推荐的雷达对齐 Camera 的帧号, 如果一帧多维像素有两帧雷达的数据则此域显示后一帧雷达数据的对齐情况

160	4	uint32	iRadarRawdata	0: Radar 点云原始数据 X, Y, Z 坐标标示 1: Radar 点云原始数据径向距离, 水平, 垂直夹角标示
-----	---	--------	---------------	---

错误和警告信息:

错误现象	软件错误码	错误描述
Reserved	1<<15	TBD
Reserved	1<<14	TBD
Reserved	1<<13	TBD
Reserved	1<<12	TBD
camera_framestart_error	1<<11	1: no camera trigger signal for long time (1s)
ir_framestart_error	1<<10	1: no ir trigger signal for long time
lidar_framestart_error	1<<9	1: no lidar trigger signal for long time
start_IR_transmit_error	1<<8	1: ir sent less than 20 frames
start_8m_transmit_error	1<<7	1: camera sent less than 20 frames
vid_timing_locked_error	1<<6	1: VTC module was not locked for long time ,the value is fixed to high when there is no VTC module
camera_data_complete_error 5	1<<5	1 : camera has lost data before
ir_data_complete_error	1<<4	1: ir has lost data before
camera_no_data_error	1<<3	1: camera has no data for long time
ir_no_data_error	1<<2	1: ir has no data for long time
vdma_rd_no_data_error	1<<1	1: rd_vdma has no data for long time
mipi_tx_no_data_error	1<<0	1: mipi_tx has no data for long time ; However, this error detection signal represents a VTC locked signal
Reserved	1<<18 ~ 1<<32	TBD
meta_data_match_error	1<<17	The frame ID reading out before filling the meta date does not match the frame ID read out after filling the meta date
ir_frame_match_error	1<<16	1: IR read out different frame index while in 70% and 100% interrupt

表 4-2 错误和警告信息

注：时间戳定义参考 <https://tool.lu/timestamp/>

参考解析代码如下：

```
time_t tick = (time_t)stampTime; //stampTime 是 32bit 秒寄存器

struct tm tm;

char s[100];

tm = *localtime(&tick);

strftime(s, sizeof(s), "%Y-%m-%d %H:%M:%S", &tm);

纳秒寄存器的值*1000*1000 得到的是在上述时间的毫秒部分
```

雷达和可见光有多种同步模式：

- 1，雷达和 camera 时间对齐后发出，此种模式下 camera 会等相同时刻采样的雷达数据达到后，统一通过多维像素发出；
- 2，极低延时模式，camera 尽快发出，雷达不和 camera 做时间对齐，雷达数据融入最近一帧多维像素里，此模式下雷达数据不做缓存，收到就发出；
- 3，雷达收满一帧后融合入多维像素发出，此模式可以在单帧多维像素里得到完整一帧雷达数据（如果雷达数据量超过 913920bytes，会拆到两帧多维像素发出）。通过 iRadarSyncMode 字段来查询当前模式。

模式 2 为了尽快将雷达数据发出，在一个多维像素数据包里有可能会出现 2 帧雷达的数据，因此 metadata 里分别定义了两帧数据起始位置、长度、时间戳等信息，需要通过 iRadarFrameNum 来判断雷达帧数。

4.3 Camera data 结构

解析 metadata 结构体即可获得 camera 的数据，默认是 yuv422 排布，具体定义如下：紧凑格式（packed format）将 Y、U、V 值储存成 Macro Pixels 阵列（YUV 是混合在一起的），4:2:2 表示 2:1 的水平取样，垂直完全采样，每 2 个 Y 共用一组 UV 分量，每个像素数据量是 2bytes，具体如下：

YUV422 8bits输出格式

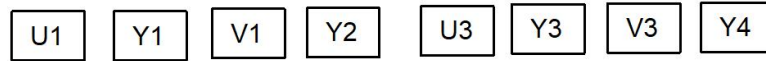


图 4-3 YUV422 8bits 数据格式

需要注意的是不同的图像采集卡在做图像数据采集存储的时候，存在 16bits 数据存储大小端（Big-Endian/Little-Endian）的情况，需要根据具体的情况做相应的数据处理。以下面例子说明：16bits 数据 0x1234 在存储器起始地址 0x4000 存储，需要占用两个字节，按照大小端的数据存储方式，有以下两种地址/数据对应关系：

表 4-3 大小端存储示例

内存地址	小端模式存放内容	大端模式存放内容
0x4000	0x34	0x12
0x4001	0x12	0x34

需要用户在图像数据解析时做相应的调整。

4.4 IR data 结构

通过解析 metadata 结构体可以获得红外传感器数据，红外默认是一个像素 1 个字节，是 Y 分量。

4.5 Radar data 结构

解析 metadata 结构体即可获得雷达数据，雷达数据有两种类型输出：稀疏输出和密集输出。

1. 毫米波雷达稀疏输出数据定义：

表 4-4 毫米波雷达稀疏输出数据定义

起始字节	字节长度	类型	定义	备注
0	2	int16	P (对应的 value[0])	雷达点在图像里的横坐标，这个数字除以 10，四舍五入取整就是对应像素坐标（0~3840） 针对 1.0 的后续版本，系统做了内存优化的一个版本，不再除以 10 进行计算；具体计算公式请参考如下 1) 的解释： iRadarScaleFactor = 9

				iRadarHorizontalOffset = 15687 iRadarVerticalOffset = 9810 。
2	2	int16	Q (对应的 value[1])	雷达点在图像里是纵坐标, 这个数字除以 10 , 四舍五入取整就是对应像素坐标 (0~2160) 针对 1.0 的后续版本, 系统做了内存优化的一个版本(同上备注相同) 请参考如下 1) 的解释
4	4	float	distance	m, relative to camera
8	4	float	speed	m/s, relative to camera
12	4	float	snr	dB, no target exists if 0
16	4	float	X or Azangle	iRadarRawdata=0 表示雷达点云数据的 X 坐标位置 iRadarRawdata=1 表示雷达点云数据的水平夹角
20	4	float	Y or Elangle	iRadarRawdata=0 表示雷达点云数据的 Y 坐标位置 iRadarRawdata=1 表示雷达点云数据的垂直夹角
24	4	float	Z	iRadarRawdata=0 表示雷达点云数据的 Z 坐标位置 iRadarRawdata=1 无效数据

1) 针对 1.0 的后续版本毫米波雷达组合的超级摄像头:

iRadarScaleFactor = 9

iRadarHorizontalOffset = 15687

iRadarVerticalOffset = 9810

$p = \text{round}(((\text{value}[0] + \text{iRadarHorizontalOffset}) / \text{iRadarScaleFactor}), 1)$

$q = \text{round}(((\text{value}[1] + \text{iRadarVerticalOffset}) / \text{iRadarScaleFactor}), 1)$

注: value[0] 和 value[1]是带符号的数值

2. 激光雷达稀疏输出数据定义:

表 4-5 激光雷达稀疏输出数据定义

起始字节	字节长度	类型	定义	备注
0	2	int16	P (对应的 value[0])	a) 雷达点在图像里的横坐标, 这个数字除以 10 , 四舍五入取整就是对应像素坐标 (0~3840) (参考如下 1) 的解释)

				b) 针对华域激光雷达组合的超级摄像头，系统做了内存优化的一个版本，不再除以 10 进行计算；具体计算公式请参考如下 2) 的解释： iRadarScaleFactor = 8.5 iRadarHorizontalOffset = 16420 iRadarVerticalOffset = 9006 。
2	2	int16	q (对应的 value[1])	a) 雷达点在图像里的纵坐标，这个数字除以 10，四舍五入取整就是对应像素坐标（0~2160）（参考如下 1）的解释） b) 针对华域激光雷达组合的超级摄像头，系统做了内存优化的一个版本(同上备注相同) 请参考如下 2) 的解释
4	4	float	distance	m, relative to camera
8	1	uint8	Reflectivity	目标反射率
9	4	float	X or Azangle	iRadarRawdata=0 表示雷达点云数据的 X 坐标位置 iRadarRawdata=1 表示雷达点云数据的水平夹角
13	4	float	Y or Elangle	iRadarRawdata=0 表示雷达点云数据的 Y 坐标位置 iRadarRawdata=1 表示雷达点云数据的垂直夹角
17	4	float	Z	iRadarRawdata=0 表示雷达点云数据的 Z 坐标位置 iRadarRawdata=1 无效数据

1) 像素坐标域为了更高的精度保留了小数点后一位，比如收到数字是 12346，则需要如下转换：12346/10=1234.6 再四舍五入得到 1235 为实际的像素坐标。

2) 针对华域激光雷达组合的超级摄像头，

iRadarScaleFactor = 8.5

iRadarHorizontalOffset = 16420

iRadarVerticalOffset = 9006

p = round(((value[0] + iRadarHorizontalOffset) / iRadarScaleFactor), 1)

q = round(((value[1] + iRadarVerticalOffset) / iRadarScaleFactor), 1)

注： value[0] 和 value[1]是带符号的数值

4.6 IR 和可见光 Camera 融合参数说明

参数是 2 个，放大倍率和四个角的坐标值。

1. 举例如下：经过测试和计算得出的放大倍率是 490%，也就是红外图像需要首先放大 4.9 倍。从 640*480 分辨率变为 3136*2352 分辨率。
2. 放大之后，红外 Camera 和可见光 Camera 中心对齐的情况下，实际情况是下面这样的。
黑色是可见光 Camera，红色是红外 Camera。

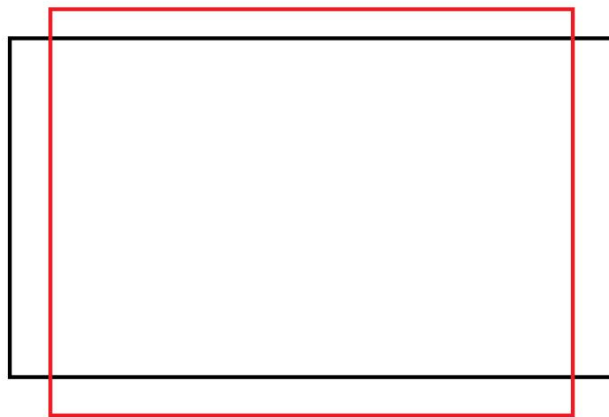


图 4-4 红外 Camera 和可见光 Camera 中心对齐的情况

3. 因为红外 Camera 和可见光 Camera 物理间距的原理。我们需要在可见光 camera 的上移动红外 camera 的图像。
4. 红外图像移动之后的状况如下图所示。也就是红外图像放大之后，只有上部的部分融合到了可见光 camera 上，下部有部分和可见光 camera 是没有重合的，需要舍弃掉。
5. 我们需要给出如图 A, B, C, D 四个点的坐标值。

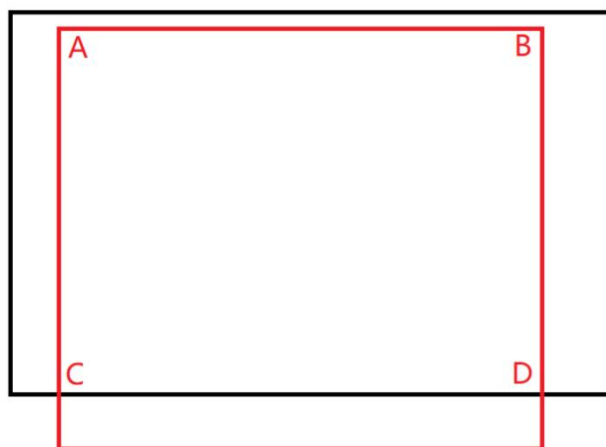


图 4-5 红外 Camera 和可见光 Camera 对齐坐标示意

6. 四个点的坐标值如下，A (266, 0)，B(3402, 0)，C(266, 2160)，D(3402, 2160)；

4.7 同步触发说明

昱感微融合设备支持外同步触发和内同步触发两种传感器触发模式。默认是内同步触发，在内同步触发模式下，可见光 30fps，红外 30fps，毫米波雷达 15fps 的触发信号由昱感微融合芯片内部产生。

外部同步触发，同步信号由域控通过 MX9296 产生。本设备使用 MAX96717 的 MFP0 作为外触发引脚，和艾利光 8M camera 控制方式一致。域控可以参考艾利光 8M camera 的外触发的驱动代码和寄存器配置内容。外同步触发支持 30/20/15/10fps 并且支持 1PPS 触发同步。

表 4-6 雷达同步触发频率

外触发频率	可见光频率	红外频率	毫米波雷达频率
30	30	30	15
20	20	20	10
15	15	15	15
10	10	10	10
1PPS	30	30	15

五、MAX96717 配置参考

以域控为例的硬件连接框图

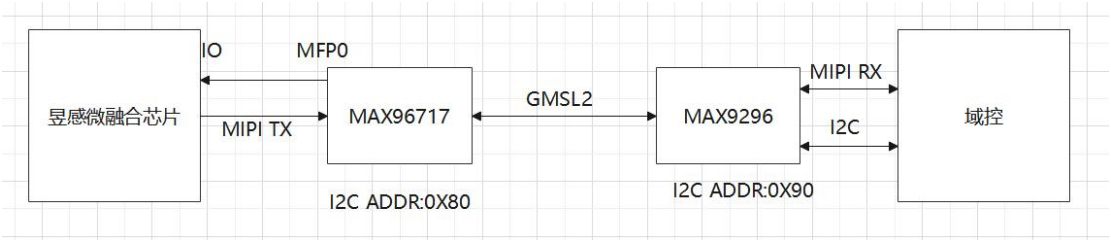


图 5-1 外部域控硬件连接图

需要域控端通过 GMSL2 配置 MAX96717，配置参考如下：

```
reg_tbl reg_serialize_tbl[] =
{
    {0x002, 0x03},
    {0x01a, 0xAD},
    {0x330, 0x00},

```

```
{0x383, 0x00},  
{0x331, 0x30},  
{0x332, 0xe0},  
{0x333, 0x04},  
{0x334, 0x00},  
{0x335, 0x00},  
{0x308, 0x64},  
{0x311, 0x40},  
{0x318, 0x5e},  
{0x319, 0x52},  
{0x315, 0x80},  
{0x30d, 0xff},  
{0x110, 0x60},  
{0x111, 0x50},  
{0x112, 0x08},  
{0x312, 0x04},  
{0x31e, 0x2c},  
{0x002, 0xf3},  
{0x2be, 0x84},  
{0x2bf, 0x20},  
{0x2c0, 0x00}  
};
```