

TITAN4 系列产品
相机同步触发说明



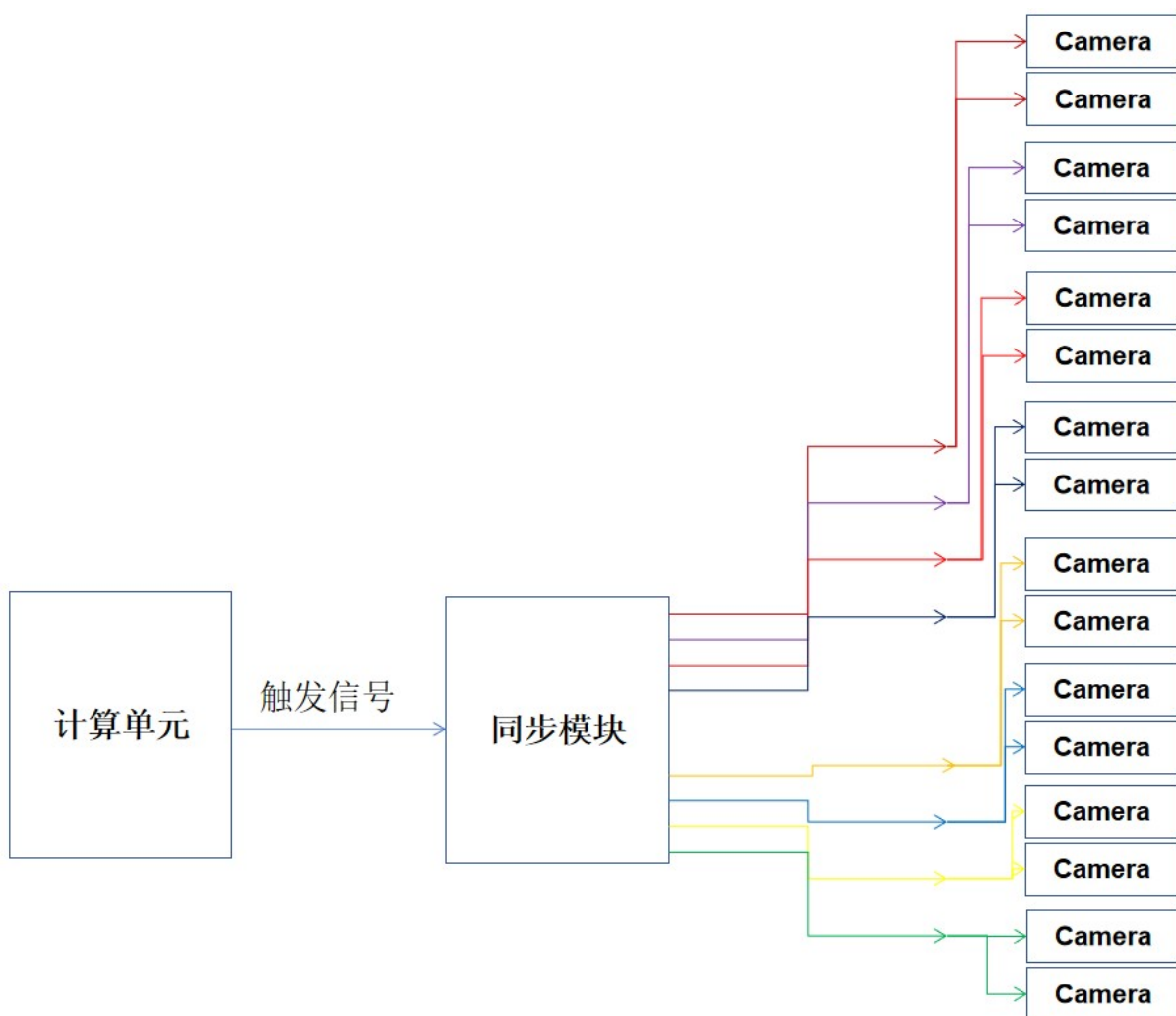
版本历史

版本号	适用于	时间	说明
Ver1.0.0	TITAN4	2021.4	初始版本
Ver1.0.1	TITAN4	2021.5	增加了相机同步触发的原理框图。 补充了 SPI 寄存器的说明附录。

相机同步触发

一、 原理说明:

根据相机安装角度和雷达的转速, 计算出每个角度相机基于相机 0° 的时延, 然后当雷达扫描到 0° 的时候, 下发一个触发信号给同步模块, 同步模块根据寄存器里存储的时延, 依次去下发触发脉冲给相机



二、 检查系统版本:

内核和 FPGA 需要同时支持同步的才能支持同步触发相机。



cat_version.sh

运行 `cat_version.sh` 读取设备版本号, 如下图:

```
titan@titan-ubuntu1: ~  
titan@titan-ubuntu1:~$ ./cat_version.sh  
kernel:Jan 8 16:50:22 CST 2021  
DTB :Jan 8 2021 16:38:33  
rootfs:201231  
FW :70 08 02 01 1F  
titan@titan-ubuntu1:~$
```

Kernel //内核版本, 相机同步版本日期为 2021/1/8

DTB //设备树版本号

rootfs //镜像版本号

FW //图中 02_01_1F 为 FPGA 版本号, 相机同步版本需要在 V2.1.25 或更新
新的版本才能支持

注: FPGA 程序版本 V2.1.25 和之后的版本, 默认同步脉冲是关闭的, 需要设置寄

存器 SPI 参数打开脉冲, 设置 `SPIx0a[4]=1:`, 命令如下:

- `spidev_test -v -D /dev/spidev1.0 -s 25000000 -p SPIW\x01\x0a\x11`

```
titan@titan-ubuntu1: ~
titan@titan-ubuntu1:~$ spidev_test -v -D /dev/spidev1.0 -s 25000000 -p SPIW\\x01\\x0a\\x11
spi mode: 0x0
bits per word: 8
max speed: 25000000 Hz (25000 KHz)
TX | 53 50 49 57 01 0A 11 -----
   | SPIW...
RX | 00 00 00 00 00 00 -----
   | .....
titan@titan-ubuntu1:~$ spidev_test -v -D /dev/spidev1.0 -s 25000000 -p SPIR\\x01\\x0aX
spi mode: 0x0
bits per word: 8
max speed: 25000000 Hz (25000 KHz)
TX | 53 50 49 52 01 0A 58 -----
   | SPIR..X
RX | 00 00 00 00 00 11 -----
   | .....
```

注：SPI 寄存器的详细说明见后文附录 1

脉冲打开后，先执行\$ `./killcamera.sh`，然后再开相机。

(执行 `killcamera.sh` 可关闭摄像头进程，当摄像头出问题后可执行该脚本，再重启摄像头)

三、 非同步版本升级为同步触发版本

3.1 更新内核文件



内核更新文件.rar

版本说明：

`install_FSYNC.sh` //相机同步内核（只支持宝隆 GSML 相机）

`install_imx390.sh` //宝隆 GSML 相机内核

`install_ov2775.sh` //FPDLINK 相机内核

步骤：

1.将 210108 文件夹拷贝到 home

2.执行相应版本的安装脚本

3.重启

3.2 更新 FPGA 程序

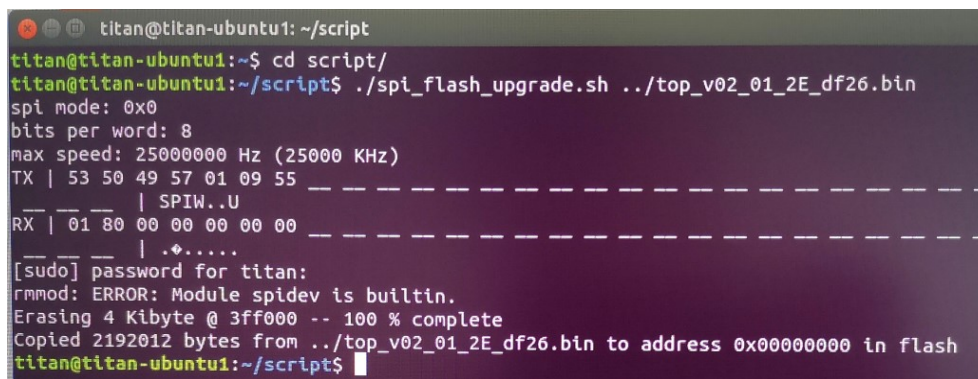
(注意: V2.1.25 或更新的版本无需更新)

将程序文件拷贝到 home 下, 打开终端, 输入:

```
$ ./script/spi_flash_upgrade.sh ./[升级程序文件]
```

输入密码: titan4

升级完成后如下图, 此时手动重启, 系统正常启动, 升级完成



```
titan@titan-ubuntu1: ~/script
titan@titan-ubuntu1:~$ cd script/
titan@titan-ubuntu1:~/script$ ./spi_flash_upgrade.sh ../top_v02_01_2E_df26.bin
spi mode: 0x0
bits per word: 8
max speed: 25000000 Hz (25000 KHz)
TX | 53 50 49 57 01 09 55  -----
   |          | SPIW..U
RX | 01 80 00 00 00 00 00  -----
   |          | .♦.....
[sudo] password for titan:
rmmod: ERROR: Module spidev is builtin.
Erasing 4 Kibyte @ 3ff000 -- 100 % complete
Copied 2192012 bytes from ../top_v02_01_2E_df26.bin to address 0x00000000 in flash
titan@titan-ubuntu1:~/script$
```

注: 升级过程禁止断电, 升级完成之后必须手动掉电重启, 程序文件必须拷贝到系统硬盘目录下再升级

3.3 相机同步触发验证

(1) 内核和 FW 程序更新完成后, 运行命令:

```
$ ./open1camera.sh [ID] # 打开摄像头画面
```

例如打开端口号为 1 的摄像头 (再脚本所在文件夹下运行终端), 如下图所示:

```

titan@titan-ubuntu1: ~/script
titan@titan-ubuntu1:~$ cd script/
titan@titan-ubuntu1:~/script$ ./openicamera.sh 1
Setting pipeline to PAUSED ...
Pipeline is live and does not need PREROLL ...
Setting pipeline to PLAYING ...
New clock: GstSystemClock
GST_ARGUS: Creating output stream
CONSUMER: Waiting until producer is connected...
GST_ARGUS: Available Sensor modes :
GST_ARGUS: 1920 x 1080 FR = 29.999999 fps Duration = 33333334 ; Analog Gain range min 1.000000, max 31.622776; Exposure Range min 118000, max 660000000;

GST_ARGUS: Running with following settings:
  Camera index = 1
  Camera mode = 0
  Output Stream W = 1920 H = 1080
  seconds to Run = 0
  Frame Rate = 29.999999
GST_ARGUS: Setup Complete, Starting captures for 0 seconds
GST_ARGUS: Starting repeat capture requests.
CONSUMER: Producer has connected; continuing.

```

(2) 验证同步，运行命令:

新开一个终端，输入命令

```
$ sudo camera_FSYNC_control
```

输入密码 titan4，每执行命令一次，摄像头画面闪动一次，以此验证相机同步触发 OK（可多个摄像头一起验证，在触发前打开多路摄像头）。

注意：上电之前必须连接好摄像头，否则系统无法识别将导致摄像头无法打开。

附录 1 :

SPI 寄存器定义

addr	Function	def	dir	Description
x00	ID[15:08]			硬件编号：由 FPGA 的 DNA 和 eFuse 与软件关联绑定，通过 ID[15:12]可追溯硬件版本号 A~G、ID[11:0]可追溯硬件编号 0~4095。
x01	ID[07:00]	--	R	
x02	Ver[23:16]			FPGA 软件版本号，Ver[23:16]为主版本号(重新设计)，Ver[15:08]表示次版本号(重大修复),Ver[07:00]表示修订版本号(优化升级)。
x03	Ver[15:08]	--	R	
x04	Ver[07:00]			

x05	5	PRSNT	--	R	硬件输入: [0]=Xavier1 在位检测, 1=在, 0=不在; [1]=Xavier2 在位检测, 1=在, 0=不在; [5:2]=GPI[3:0]; [7:6]保留=0
x06	6	STATE	0	RW	系统状态检测: [0]=4G, 状态由 Xavier 写入; [1]=lock_pps 锁定, [2]=lock_GPRMC 锁定(每秒刷新), [3]=lock_GPS 持续 5s 锁定, 1=锁定, 0=失锁; [4]=风扇 1, [5]=风扇 2, 0=转, 1= 停; [6]GPS 的 UTC 时间冗余校验, 0=正常, 1=异常; [7]保留
x08	8	SW[07:00]	0	RW	外设状态工作时置 1、指令完成后清零, 忙时不可切换, 默 认 0=空闲、1=忙。
					外设切换由在 XV1 位检测控制, 默认 0=XV1 可控, 1=XV2 可 控。
					[0/1]=FPGA_I2C_SW 状态/切换; [2/3]=FPGA_LTE_USB_S 状 态/切换;
					[4/5]=FPGA_SMI_SW 状态/切换; [6/7]=保留。
x09	9	SW[15:08]	0	RW	[15:08]功能保留, 环宇内部可见
x0a	10	SW[23:16]	1	RW	SW[16]:pps 选择, 默认 1=外部 GPS 直通(PPS_IN 和 PPS_RS232_RX 输入), 0=PPS 和 GPRMC 本地解析转发到 PPS_OUT1~5 和 PPS_RS232_TX1~5; [17]:FAN 控制, 0=XV 控, 1= 常开; [18]=GPO1, [19]=GPO2; [20]CAM_clk 输出使能, 1= 按照 spi11 输出 CAM_fps 频率控制、按照 spi64~87 输出相 位差 sync0~7, 0=无 CAM_clk 输出; [23:21]=保留
x0b	11	CAM_fps	1e	RW	摄像头帧同步脉冲频率对应帧率 5/10/15/20/25/30/60fps 可选, 默认 8x1e=30fps
x10	16	LED[07:00]	ff	RW	LED 控制, 功能保留。LED[11:10]=LED5, LED[9:8]=LED4, [7:6]=LED3, [5:4]=LED2, [3:2]=LED1, [1:0]=LED0。0=熄
x11	17	LED[15:08]	ff		灭, 1=常亮, 2=爆闪, 3=呼吸
x12	18	bps_Rx	7	RW	pps_Rx/Tx 的波特率(ZED-F9P 可配置范围 9600~921600bps), 对应 bps 关系如下:

x13	19	bps_Tx	7		1:110, 2:300, 3:600, 4:1200, 5:2400, 6:4800, 默认 7:9600, 8:14400, 9:19200, 10:38400, 11:57600, 12 或其他:115200, 13:230400, 14:460800, 15:921600。
x17	23			-- RW	功能保留, 环宇内部可见
x18	24				
x19	25				
x1a	26				
x20	32	Uart[07:00]	f0		Uart[75:0]每 4bit 对应 1 个端口的连接关系, 具体定义如下:
x21	33	Uart[15:08]	1f		[15:00]={XV1_Uart5, XV1_U3_DBG, XV1_Uart2, XV1_Uart1},
x22	34	Uart[23:16]	f2		[31:16]={XV2_Uart5, XV2_U3_DBG, XV2_Uart2, XV2_Uart1},
x23	35	Uart[31:24]	ff		[47:32]={GNSS2, GNSS1, LTE2, LTE1}, [63:48]={rs232_4, rs232_3, rs232_2, rs232_1},
x24	36	Uart[39:32]	ff		[75:64]={rs422, rs485_2, rs485_1}, [79:76]未定义。 ttyTHS0 对应 Uart1, ttyTHS4 对应 Uart5。
x25	37	Uart[47:40]	ff	RW	寄存器值表示对应的连接关系: 0=rs232_1, 1=rs232_2, 2=rs232_3, 3=rs232_4, 4=rs485_1, 5=rs485_2, 6=rs422(rs485_1 收+rs485_2 发), 7=LTE_TX1, 8=LTE_TX2, 9=GNSS_TX1, a=GNSS_TX2, b=GPS, else=未连接。
x26	38	Uart[55:48]	ff		例如, 寄存器 x20[3:0]表示 XV1_Uart1 的连接关系, 默认 0 表示对应 rs232_1; <code>spidev_test -v -D /dev/spidev1.0 -s 25000000 -p SPIW\x01\x20\xf4</code> 改为 4 则对应 rs485_1。 为避免端口设置冲突, Uart[79:0]低位优先级最高。
x27	39	Uart[63:56]	bf		Uart[63:60]=b 表示 rs232_4 默认连接到 GPS。左/右两侧的隐藏式按钮可把 XV1_DBG 或 XV2_DBG 连接到 rs232_4 进行调试, rs232_TX4/RX4 接 (USB↔串口) 转接器的 2/3 脚, 对应电脑的 RX/TX。

x28	40	Uart[71:64]	ff		
x29	41	Uart[79:72]	ff		
x30	48	UTC1	14	RW	YY 年, 00~99。无 GPS 输入时, 设备启动时 UTC 从 20 年 1 月 1 日 00:00:00 开始计时。有 GPS 时, pps 秒脉冲有效沿 UTC 自加 1s、ms/us/ns 清零, 收 GPRMC 后立即使用。可由 date_updata_from_FPGA 命令同步 UTC 时间到 Xavier。
x31	49	UTC2	1	RW	MM 月, 1~12
x32	50	UTC3	1	RW	DD 日, 1~31
x33	51	UTC4	0	RW	hh 时, 00~23
x34	52	UTC5	0	RW	mm 分, 00~59
x35	53	UTC6	0	RW	ss 秒, 00~59
x36	54	UTC7[15:08]	0	RW	ms, 000~999
x37	55	UTC7[07:00]	0		
x38	56	UTC8[15:08]	0	RW	us, 000~999
x39	57	UTC8[07:00]	0		
x3a	58	UTC9[15:08]	0	RW	ns, 000~999
x3b	59	UTC9[07:00]	0		
x40	64	sync0[07:00]	0	RW	摄像头同步脉冲 CAM1_GPIO0 相对 FSYNC 命令的延时, 根据帧率 fps 和角度 ang 设置, dly=(frq/fps)*(angle/360), 计数时钟 frq=100MHz, dly 计数单位 10ns, dly 最大范围 sync*=xFFFFFF=16777215=167.8ms, 对应 5.96fps-360° 或 5fps-302°
x41	65	sync0[15:08]	0		
x42	66	sync0[23:16]	0		
x43	67	sync1[07:00]	9B	RW	CAM1_GPIO1 的相对延时, 默认 sync1[23:0]=x065B9B=416667=4.17ms, 对应 30fps-45°
x44	68	sync1[15:08]	5B		

x45	69	sync1[23:16]	6		
x46	70	sync2[07:00]	35	RW	CAM1_GPI02 的相对延时,
x47	71	sync2[15:08]	B7		默认 sync2[23:0]=x0CB735=833333=8.33ms, 对应 30fps-90°
x48	72	sync2[23:16]	0C		
x49	73	sync3[07:00]	D0	RW	CAM1_GPI03 的相对延时,
x4a	74	sync3[15:08]	12		默认 sync3[23:0]=x1312D0=1250000=12.5ms, 对应 30fps-135°
x4b	75	sync3[23:16]	13		
x4c	76	sync4[07:00]	6B	RW	CAM2_GPI00 的相对延时,
x4d	77	sync4[15:08]	6E		默认 sync4[23:0]=x196E6B=1666667=16.7ms, 对应 30fps-180°
x4e	78	sync4[23:16]	19		
x4f	79	sync5[07:00]	5	RW	CAM2_GPI01 的相对延时,
x50	80	sync5[15:08]	CA		默认 sync5[23:0]=x1FCA05=2083333=20.8ms, 对应 30fps-225°
x51	81	sync5[23:16]	1F		
x52	82	sync6[07:00]	A0	RW	CAM2_GPI02 的相对延时,
x53	83	sync6[15:08]	25		默认 sync6[23:0]=x2625A0=2500000=25.0ms, 对应 30fps-270°
x54	84	sync6[23:16]	26		
x55	85	sync7[07:00]	3B	RW	CAM2_GPI03 的相对延时,
x56	86	sync7[15:08]	81		默认 sync7[23:0]=x2C813B=2916667=29.2ms, 对应 30fps-315°

x57	87	sync7[23:16]	2C	
else		未定义		