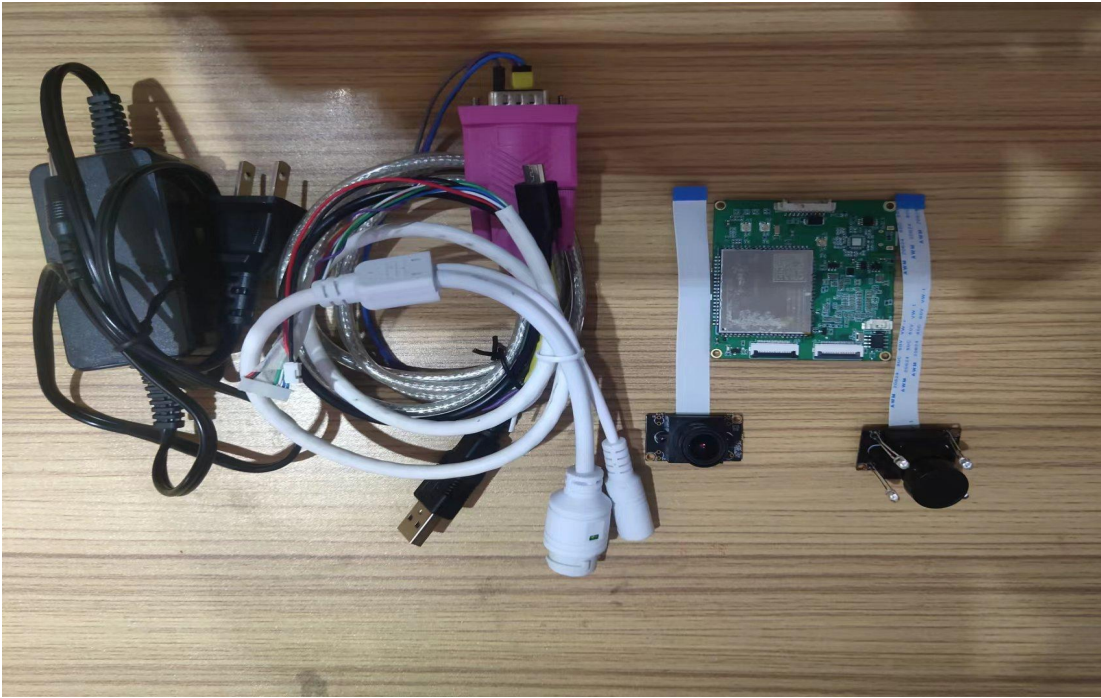


1. 发货清单

电源一个， 串口线一根， sensor 两个， 排线两根， 电源网线一根



连接方式：



2. 默认出厂时烧写 uboot, kernel, rootfs,此系统在自主开发学习并不一定适用，可能需要重新编译自己的程序，

Uboot:

- (1) 修改 uboot:include/configs/isvp_t40.h 开启宏 #define CONFIG_NORFLASH_32M，默认是 16M 的 nor.
- (2) 编译方式: 参见 t40 开发指南 uboot 编译

Kernel:

- (1) 当前的硬件上使用的是 i2c1 和 i2c3 作为 sensor 的连接，确认是 dts 设备树配置是否为 i2c1 : SDA_PA17, SCK_PA18 , i2c3 : SDA_PA21, SCK_PA22 ,
路径: /T40_SDK/opensource/kernel-4.4.94/arch/mips/boot/dts/ingenic/shark.dts

- (2) 编译方式: 参见 t40 开发指南 kernel 编译

Rootfs: 参见 t40 文件系统制作指南

环境变量配置: 使用 ivs 需要分配在 uboot 下分配算法内存

3. 驱动的加载

```
insmod avpu.ko
```

```
insmod tx-isp-t40.ko isp_clk=200000000
```

```
insmod sensor_gc2053_t40.ko
```

```
insmod sensor_gc2053s1_t40.ko
```

4. 确认效果文件是否存在：

开发板/etc/sensor/ 下存在两个效果文件 gc2053-t40.bin 和 gc2053s1-t40.bin

5. 验证当前的驱动环境是否正确，见 t40 开发指南：carrier 使用：

\t40-sdk\Ingenic-SDK-T40-1.0.4-20210618\resource\tools_t40\pc\Carrier-V4.0.0

Linux 下需要拷贝到开发板上：

E:\Work\t40\t40-sdk\Ingenic-SDK-T40-1.0.4-20210618\resource\tools_t40\bin\uclibc\carrier-server

双 sensor 执行：

./carrier-server --st=gc2053 --st1=gc2053s1 --sensornum=2 --dualmode=4

单 sensor 执行：

./carrier-server --st=gc2053

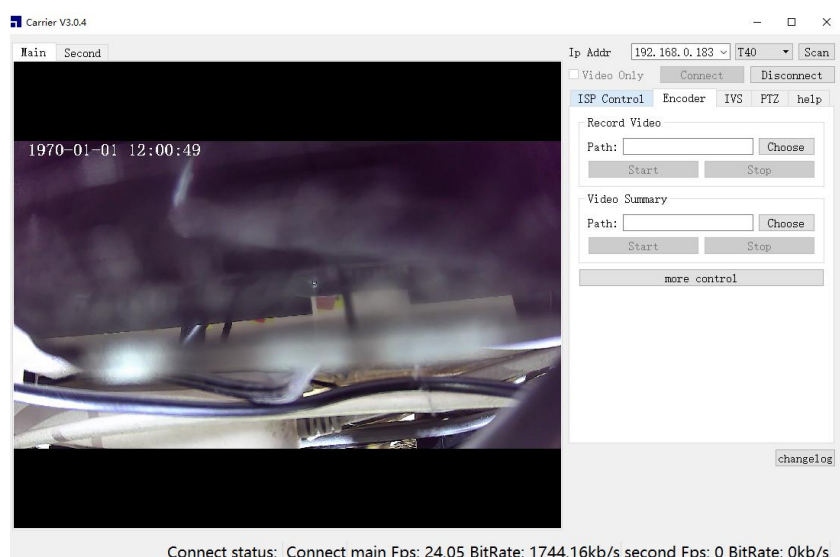
```
[root@Ingenic-ucl_1:samples]# ./carrier-server --st=gc2053 --st1=gc2053s1 --sensornum=2 --dualmode=4
sample-RTSPServer: An RTSP server for Ingenic libimp
Built using "LIVE555 Streaming Media": <http://www.live555.com/liveMedia/>
Built time: Jan 20 2022, 22:17:41
Sensor Type : gc2053
Sensor1 Type : gc2053s1

Argument print:
  Main EncType = h265 = 16777217
  Main EncType = h265 = 16777217
  Main VideoWidth = 1920
  Main VideoHeight = 1080
  Second VideoWidth = 640
  Second VideoHeight = 360
  FPSNum = 25
  FPSDen = 1
  Audio = 0
  RC = CapedQuality
  cli = 1

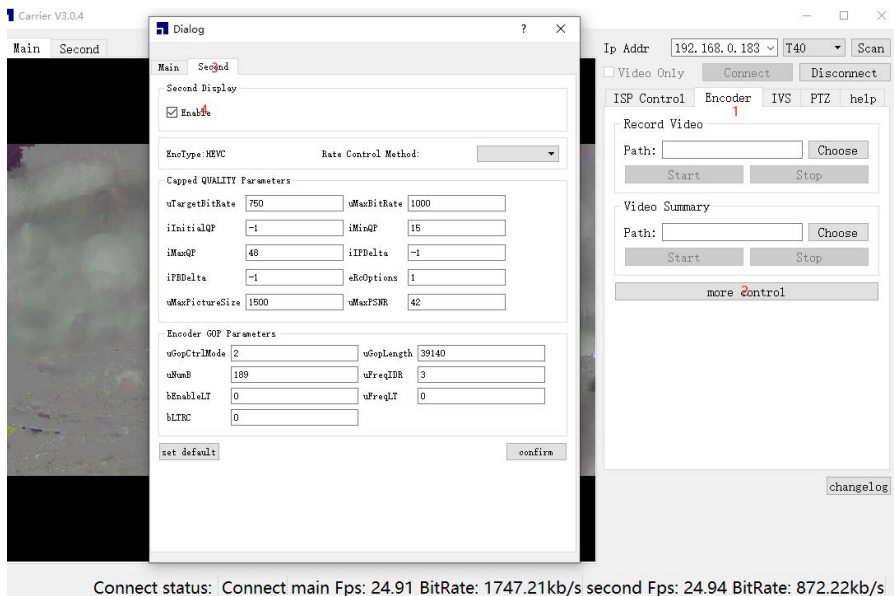
Function usage:
  showfps: Show FPS and bitrate infomation
  osd: Switch on/off timestamp OSD
  snap: Snapshot a JPEG picture
  display: Display on LCD

Initializing...
---- FPGA board is ready ----
  Board UID : 30AB6E51
  Board HW ID : 72000760
  Board rev. : ABA57397
  Board date : 20200311
-----
> Timestamp OSD start
OSD start
...done initializing
Play this video stream using the URL:
  rtsp://0.0.0.0:8554/main
Timestamp OSD start
OSD start
Play this video stream using the URL:
  rtsp://0.0.0.0:8554/second
```

pc 端运行截图如下：



如果第二个 sensor 不出图像，做如下修改：



6. 当第 5 步能正常出图就可以验证 samples 程序
sample-common.h 修改如下：

```
/* first sensor */
#define FIRST_SNESOR_NAME "gc2053" //sensor name (match with snesor driver name)
#define FIRST_I2C_ADDR 0x37 //sensor i2c address
#define FIRST_I2C_ADAPTER_ID 1 //sensor controller number used (0/1/2/3)
#define FIRST_SENSOR_WIDTH 1920 //sensor width
#define FIRST_SENSOR_HEIGHT 1080 //sensor height
#define FIRST_RST_GPIO GPIO_PC(27) //sensor reset gpio
#define FIRST_PWDN_GPIO -1 //sensor pwnn gpio
#define FIRST_POWER_GPIO -1 //sensor power gpio
#define FIRST_SENSOR_ID 0 //sensor index
#define FIRST_VIDEO_INTERFACE IMPISP_SENSOR_VI_MIPI_CSI0 //sensor interface type (dvp/csi0/csi1)
#define FIRST_MCLK IMPISP_SENSOR_MCLK1 //sensor clk source (mclk0/mclk1/mclk2)
#define FIRST_DEFAULT_BOOT 2 //sensor default mode(0/1/2/3/4)

/* second sensor */
#define SECOND_SNESOR_NAME "gc2053s1" //sensor name (match with snesor driver name)
#define SECOND_I2C_ADDR 0x37 //sensor i2c address
#define SECOND_I2C_ADAPTER_ID 3 //sensor controller number used (0/1/2/3)
#define SECOND_SENSOR_WIDTH 1920 //sensor width
#define SECOND_SENSOR_HEIGHT 1080 //sensor height
#define SECOND_RST_GPIO GPIO_PC(28) //sensor reset gpio
#define SECOND_PWDN_GPIO -1 //sensor pwnn gpio
#define SECOND_POWER_GPIO -1 //sensor power gpio
#define SECOND_SENSOR_ID 1 //sensor index
#define SECOND_VIDEO_INTERFACE IMPISP_SENSOR_VI_MIPI_CSI1 //sensor interface type (dvp/csi0/csi1)
#define SECOND_MCLK IMPISP_SENSOR_MCLK2 //sensor clk source (mclk0/mclk1/mclk2)
#define SECOND_DEFAULT_BOOT 2 //sensor default mode(0/1/2/3/4)

/* Multimedia Camera Mode */
#define SENSOR_NUM IMPISP_TOTAL_TWO //sensor total number (one/two/thr)
#define DUALSENSOR_MODE IMPISP_DUALSENSOR_DUAL_ALLCACHED_MODE //dualsensor mode ()
#define JOINT_MODE IMPISP_NOT_JOINT //joint mode ()

#define PERSECOND_MEANS_BY_MS 1000
#define AUDIO_PERFRAME_MINIMUM_COST_BY_MS 10
#define DEFAULT_AUDIO_COST_MULTIPLIER 4

/* Channel_en Choose */
/* 1:enable 0:disable */
/* first sensor output channel (0/1/2) */
/* second sensor output channel (3/4/5) */
/* third sensor output channel (6) */
/* dual sensor joint mode only channel (0/1/2) */
#define CHN0_EN 1
#define CHN1_EN 1
#define CHN2_EN 0

#define CHN3_EN 1
#define CHN4_EN 1
#define CHN5_EN 0
```

做完如下修改就可编译 samples 例程运行