

# 总钻风全局快门摄像头 使用说明

# 目录

目录.....	1
1. 总钻风优点简介 .....	3
1.1. 全局快门.....	3
1.2. 高动态性能 .....	4
1.3. 用户无需自己编写驱动 .....	4
1.4. 可进行自动曝光 .....	4
1.5. 曝光时间可随时设置（自动曝光关闭时） .....	5
1.6. FPS 可调.....	5
2. 总钻风使用详解 .....	6
2.1. 使用注意事项 .....	6
2.2. 使用 STM32 MCU 采集 .....	7
2.2.1. 硬件版本为 V1.1.....	7
2.2.2. 硬件版本为 MT9V032 V1.5 .....	8
2.2.3. 硬件版本为 MT9V034 V1.3 .....	8
2.3. FFC 排线连接 .....	9
2.4. 摄像头与单片机连接.....	11
2.5. 打开上位机并设置上位机 .....	12

---

2.6. 摄像头配置参数说明 .....	12
2.6.1. 摄像头初始化，设置 FPS 和曝光时间，配置自动曝光 .....	12
2.6.2. 单独设置曝光时间 .....	14
2.6.3. 读取当前状态 .....	15
2.7. 自行配置摄像头的参数 .....	16
2.7.1. 使用串口自行配置摄像头参数 .....	16
2.7.2. 使用 IIC 自行配置摄像头参数 .....	17
2.8. 结语 .....	17
3. 文档版本 .....	18

# 1.总钻风优点简介

## 1.1.全局快门

总钻风采用的是 MT9V034/MT9V032 芯片，CMOS 全局快门（Global Shutter）：在曝光前整个图像重置，像素可在曝光时间积累电荷；曝光结束后，每个像素积累的电荷同时传送到屏蔽光（对光不敏感）的存储区域；然后信号从此区域读出。因所有像素同时重置，曝光积分同样的间隔，同时传输到光屏蔽存储区域，对移动物体来说没有形变。



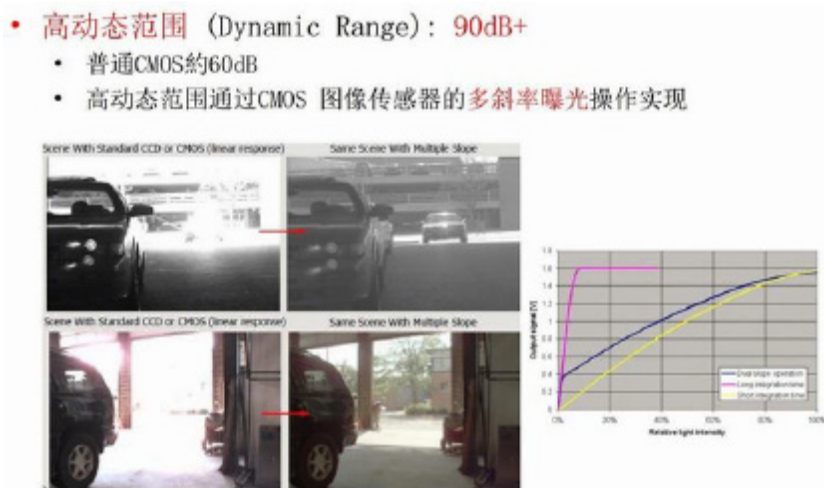
用一幅示意图来展示动态模糊、卷帘快门、全局快门拍摄小车高速运行的效果：



对于许多担心和苦恼小车在高速运动下，摄像头采集到的图像会失真的智能车大神，我们给您带来了福音，相信我们的这款全局快门（Global Shutter）的高性能摄像头会消除您的疑虑和担忧，会给您带来不一样的感受！

## 1.2.高动态性能

高动态范围简称 HDR。



高动态范围 (HDR) 图像为我们呈现了一个充满无限可能的世界，因为它们能够表示现实世界的全部可视动态范围。由于可以在 HDR 图像中按比例表示和存储真实场景中的所有明亮度值，因此，调整 HDR 图像的曝光度的方式与在真实环境中拍摄场景时调整曝光度的方式类似。HDR 是目前追求画面逼真度最新最先进的手段

简单通俗来说的话，就是在能看清暗处的同时，亮处也不会轻易过曝。

## 1.3.用户无需自己编写驱动

总钻风摄像头内部集成了驱动配置芯片，所以用户不需自己写摄像头驱动程序。通过串口即可非常简单的对摄像头参数进行配置。详细介绍参考后面的如何配置摄像头参数配置章节。

## 1.4.可进行自动曝光

我们的摄像头具有自动曝光功能，当环境变亮摄像头会自动减小曝光时间，当环境变暗摄像头会自动增加曝光时间。所以此款摄像头可以适应不同环境。

---

## 1.5.曝光时间可随时设置（自动曝光关闭时）

当摄像头采集到图像较亮（曝光过度）时，可以通过程序去调节曝光时间，使得图像不那么亮，并且改变曝光时间不需要复位摄像头。因此我们推荐自己编写程序识别当前像素点亮度是否满足自己的要求，如果不满足要求则通过程序手动来调节曝光时间，这样会比自动曝光更加靠谱。详细介绍参考后面的如何配置摄像头参数配置章节。

## 1.6.FPS 可调

软件可调 FPS，我们的总钻风摄像头可在 1-500 帧之间自由调节。这里需要特别说明，一般用户如果没有特殊需求建议设置为 50 帧或者 100 帧，以此避免工频干扰，有关工频干扰可以自行百度。一般来说 50 帧完全足够控制智能车运行，再快意义不大，因为舵机是一个响应速度较慢的执行机构，如果是直立车想要提高响应速度也可以将帧率设置为 100 帧，但是需要注意自己的处理算法应该在下一次的图像来之前处理完成。否则将毫无意义。

## 2.总钻风使用详解

### 2.1.使用注意事项

1.总钻风电源是 3.3V 的，切记不要接错。如果电源接错，有可能摄像头板子会发热，这时请立即断电，并检查电源。

2.摄像头供电不能直接从系统板或 J-LINK 取电，需使用外部电源供电。外部电源需要纹波比较小，这样图像效果才更好。

3.总钻风必须与单片机共地（使用同一个接地）。 如果没有共地就会出现如下现象：



4.使用串口发送图像到上位机时，建议不要使用 K60 的串口 4（仅限 K60 等 K 系列单片机），可能出现图像错位；

友情提示

控制器分为以下七个类别。

- （1）32 位 Kinetis (ARM® Cortex™-M0+)，主要包括 Kinetis E, EA, L, M 等系列；
- （2）32 位 Kinetis (ARM® Cortex™-M4)，主要包括 Kinetis K, W 等系列；
- （3）32 位 MPC56xx 系列；
- （4）16 位 9S12 系列；
- （5）32 位 ColdFire 系列；
- （6）DSC 系列；
- （7）8 位单片机系列；

同一所学校在报名的同一组别的两支队伍，必须使用不同系列的处理器芯片。如果在提高类别的两个组别（A1,A2）中，所选择的车型种类不同，则同一组别的两支队伍可以使用相同系列的微处理器。

- 使用微控制器的数量没有限制。
- 如果所选用的传感器或其它电子部件中也包含有微处理器，对此微处理器的种类和数量不做限制，但其不得参与对于赛道信息识别和处理、不参与车模运动决策与控制。

根据 2016 年恩智浦智能车竞赛新规定，我们的总钻风摄像头使用的 51 单片机只是用

于摄像头寄存器配置，不参与赛道信息的识别和处理，更不参与车模运动决策与控制，所以总钻风摄像头是符合主委会规定的一款可用于比赛的高性能摄像头，亲们可以放心大胆的使用！

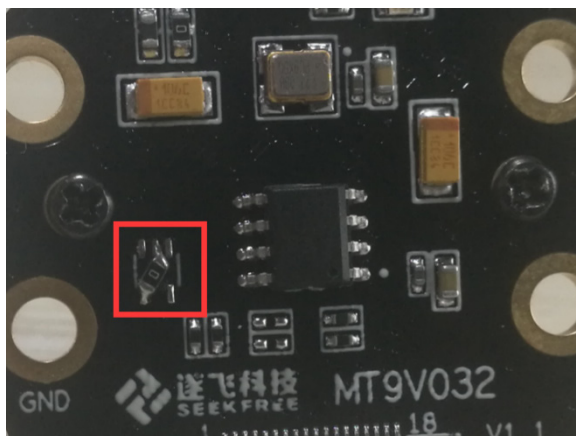
## 2.2.使用 STM32 MCU 的 DCMI 接口采集

由于 STM32 DCMI 采集方式与其他单片机采集方式存在区别，需要将总钻风的硬件做少许修改，才可以正常采集图像。

不同硬件版本的总钻风摄像头，修改方式存在一定区别，请参考以下说明。

### 2.2.1.硬件版本为 V1.1

首先拆掉红色框里的 5 脚芯片。



并使用 0 欧电阻或导线，将下图所示的两个焊盘直接连接在一起。

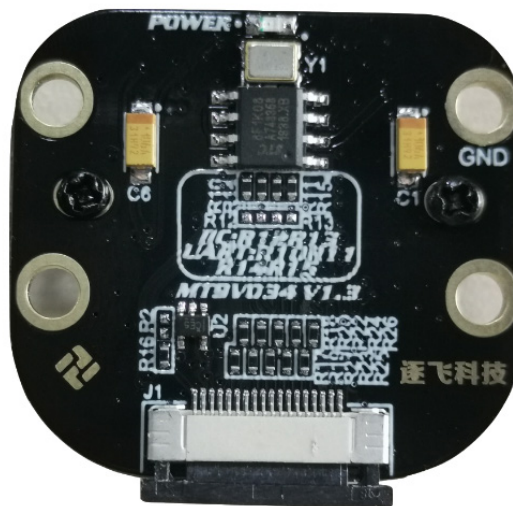




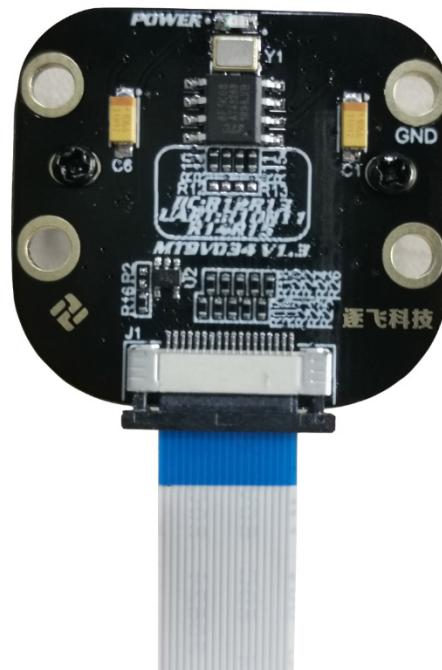


## 2.3.FFC 排线连接

第一步：将 FPC 座子的卡扣拉出来如下图所示，将卡扣拉出来的过程中请小心用力，蛮力容易导致卡扣损坏。



第二步：将 FFC 排线插入 FPC 座子中，如下图所示



第三步：将 FPC 座子卡扣推入



转接板的连接与摄像头类型如下图所示：



## 2.4.摄像头与单片机连接

具体连接应该参考例程，找到摄像头相关的文件，然后查阅最上方的接线备注可以找到，下图是 K60 例程的接线备注。

MT9V032接线定义：

模块管脚	单片机管脚
SDA	PTC17
SCL	PTC16
场中断	PTC6
像素中断	PTC18
数据口	PTC8-PTC15
串口	
波特率 115200	
数据位 8	
校验位 无	
停止位 1位	
流控 无	
RX	PTD3
TX	PTD2
OLED液晶	
D0	PTA15
D1	PTA16
RES	PTA14
DC	PTB16

VSYNC：场中断

HREF：行中断

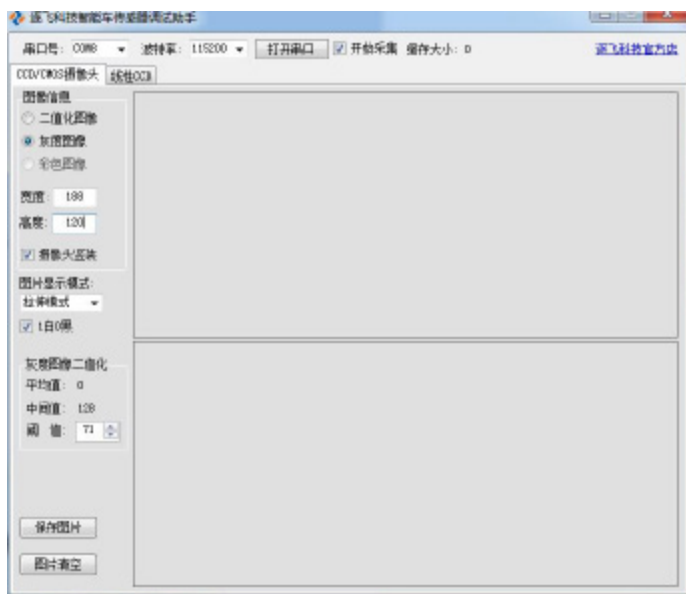
PLCK：像素中断

提示：如果使用上位机查看图像，需要准备 usb 转 ttl，以

左图为例，usb 转 ttl 的 RX 应该接单片机的 D3，TX 接 D2。

## 2.5.打开上位机并设置上位机

设置选项如下



图像的宽为 188

图像的高为 120

图片模式为拉伸模式

勾选 1 白 0 黑

勾选摄像头竖装

设置波特率为 115200

选择图像信息为灰度图像

然后打开串口勾选 开始采集 ，上位机即可显示摄像头采集到的图像。

注：WINDOS-XP 系统安装 NET4 才能使用本上位机！

由于上位机还在测试中，因此上位机存在一个 BUG。当开始采集图像时，如果要关闭串口，不要取消勾选 开始采集 。关闭时直接点击关闭串口即可。后期我们会屏蔽开始采集按钮，并做相应改进。

## 2.6.摄像头配置参数说明

### 2.6.1.摄像头初始化，设置 FPS 和曝光时间，配置自动曝光

//摄像头配置数组

{AUTO\_EXP, 0}, //自动曝光设置 范围 1-63 0 为关闭 如果自动曝

光开启 EXP\_TIME 命令设置的数据将会变为最大曝光时间，也就是自动曝光时间的上限，一般情况是不需要开启这个功能，因为比赛场地光线一般都比较均匀，如果遇到光线非常不均匀的情况可以尝试设置该值，增加图像稳定性

{EXP\_TIME, 500}, //曝光时间 摄像头收到后会自动计算出最大曝光时间，如果设置过大则设置为计算出来的最大曝光值

{FPS, 10}, //图像帧率 摄像头收到后会自动计算出最大FPS，如果过大则设置为计算出来的最大FPS

{SET\_COL, COL}, //图像列数量 范围 1-752 K60 采集不允许超过 188

{SET\_ROW, ROW}, //图像行数量 范围 1-480  
{LR\_OFFSET, 0}, //图像左右偏移量 正值 右偏移 负值 左偏移  
列为 188 376 752 时无法设置偏移 摄像头收偏移数据后会自动计算最大偏移，如果超出则设置计算出来的最大偏移

{UD\_OFFSET, 0}, //图像上下偏移量 正值 上偏移 负值 下偏移  
行为 120 240 480 时无法设置偏移 摄像头收偏移数据后会自动计算最大偏移，如果超出则设置计算出来的最大偏移

{GAIN, 32}, //图像增益 范围 16-64 增益可以在曝光时间固定的情况下改变图像亮暗程度

关于详细的参数意义介绍，可以查阅《【问题汇总】总钻风调试方法与问题锦集》文档。

这是一个 8 行 2 列的二维数组，每一行的第 0 位是存储的命令，第 1 位存储的是该命令需要写入的数据。因此在修改的时候仅仅只需要修改第 1 位即可。对摄像头进行配置的时候需

要按照一定的协议发送数据，数据是一个一个数据包发送的，每一个数据包有 4 个字节，第 0 字节是帧头 0xA5，第 1 字节是命令位，第三字节是数据高八位，第四字节是数据第八位。一个数据包发送后最好延时 2MS。对摄像头配置是可以采用所有配置数据发送完成然后发送初始化命令，如果想单独修改某个参数可以只发送相关命令及参数，然后发送初始化命令。

```
void set_config(UARTN_enum uartn, int16 buff[CONFIG_FINISH-1][2])
{
    uint16 temp, i;
    uint8 send_buffer[4];

    uart_receive_flag = 0;

    //设置参数 具体请参看问题锦集手册
    //开始配置摄像头并重新初始化
    for(i=0; i<CONFIG_FINISH; i++)
    {
        send_buffer[0] = 0xA5;
        send_buffer[1] = buff[i][0];
        temp = buff[i][1];
        send_buffer[2] = temp>>8;
        send_buffer[3] = (uint8)temp;

        uart_putbuff(uartn,send_buffer,4);
        systick_delay_ms(2);
    }
    //等待摄像头初始化成功
    while(!uart_receive_flag);
    uart_receive_flag = 0;
    while((0xff != receive[1]) || (0xff != receive[2]));
    //以上部分对摄像头配置的数据全部都会保存在摄像头51单片机的eeprom中
    //利用set_exposure_time函数单独配置的曝光数据不存储在eeprom中
}
```

上图所示为先发送全部分配置数据然后发送初始化命令，当摄像头接收到初始化命令就会开始用刚才接收到的配置数据进行初始化，初始化完成后会发送一个数据包回来，数据包有三个字节,第 0 字节是帧头 0xA5，第 1 字节是 0xFF，第三字节是 0xFF。

## 2.6.2.单独设置曝光时间

使用 set\_exposure\_time 函数可以非常简单的实现单独设置曝光时间。此参数不会被保存，掉电就会丢失。



```
uint16 set_exposure_time(UARTN_enum uartn, uint16 light)
{
    uint16 temp;
    uint8 send_buffer[4];

    send_buffer[0] = 0xA5;
    send_buffer[1] = SET_EXP_TIME;
    temp = light;
    send_buffer[2] = temp>>8;
    send_buffer[3] = (uint8)temp;

    uart_putbuff(uartn, send_buffer, 4);

    //等待接受回传数据
    while(!uart_receive_flag);
    uart_receive_flag = 0;

    temp = receive[1]<<8 | receive[2];
    return temp;
}
```

### 2.6.3.读取当前状态

使用 get\_config 函数可以轻易的读取摄像头配置信息，读取出来的数据保存在 GET\_CFG 数组里面

```
void get_config(UARTN_enum uartn, int16 buff[CONFIG_FINISH-1][2])
{
    uint16 temp, i;
    uint8 send_buffer[4];

    for(i=0; i<CONFIG_FINISH-1; i++)
    {
        send_buffer[0] = 0xA5;
        send_buffer[1] = GET_STATUS;
        temp = buff[i][0];
        send_buffer[2] = temp>>8;
        send_buffer[3] = (uint8)temp;

        uart_putbuff(uartn, send_buffer, 4);

        //等待接受回传数据
        while(!uart_receive_flag);
        uart_receive_flag = 0;

        buff[i][1] = receive[1]<<8 | receive[2];
    }
}
```



## 2.7.自行配置摄像头的参数

### 2.7.1.使用串口自行配置摄像头参数

在默认状态下，使用串口（UART）与总钻风摄像头上的 51 单片机通讯，再由 51 单片机里已有 IIC 的驱动程序对摄像头进行配置。

用户只需要通过串口给出需要操作（操作只能是对寄存器进行写入操作）的寄存器地址与需要写入的数据。使用 set\_mt9v032\_reg 函数可以实现对摄像头内部寄存器进行写入操作，写入完成后摄像头会再次读取该寄存器然后发送回来便于确认是否写入成功。一共发回来一个数据包，一个数据包有三个字节，第 0 字节是帧头 0xA5，第 1 字节是 0xFF，第三字节是 0xFF。

```
uint16 set_mt9v032_reg(uint8 addr, uint16 data)
{
    uint16 temp;
    uint8 send_buffer[4];

    send_buffer[0] = 0xA5;
    send_buffer[1] = SET_ADDR;
    temp = addr;
    send_buffer[2] = temp>>8;
    send_buffer[3] = (uint8)temp;

    uart_putbuff(MT9V032_COF_UART, send_buffer, 4);
    systick_delay_ms(10);

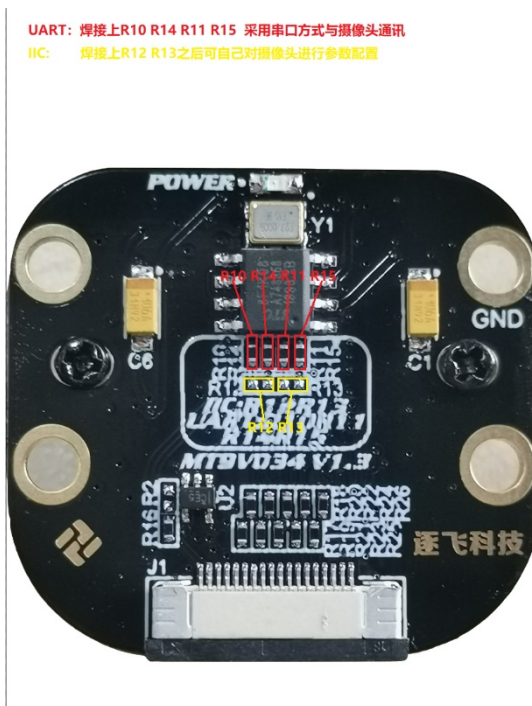
    send_buffer[0] = 0xA5;
    send_buffer[1] = SET_DATA;
    temp = data;
    send_buffer[2] = temp>>8;
    send_buffer[3] = (uint8)temp;

    uart_putbuff(MT9V032_COF_UART, send_buffer, 4);

    //等待接受回传数据
    while(!uart_receive_flag);
    uart_receive_flag = 0;

    temp = receive[1]<<8 | receive[2];
    return temp;
}
```

## 2.7.2.使用 IIC 自行配置摄像头参数



按照图中示意图，将电阻更换，更换之后摄像头上的 51 单片机不再发挥作用，CMOS 的 IIC 引脚将直接与 FFC 软排座连通，用户就可以参考 CMOS 的芯片手册，自行使用 IIC 协议配置摄像头。

## 2.8.结语

购买总钻风的各位亲！如有不懂，困惑的地方请及时与我们联系！

逐飞科技智能车技术交流群：一群 179029047（已满） ，二群 244861897（已满）

三群 824575535

逐飞科技官方店铺：<https://seekfree.taobao.com>

逐飞科技很高兴能为您服务！

### 3.文档版本

版本号	日期	内容变更
V2.1	2019-12-2	CMOS 型号变更为 MT9V034/ MT9V032
V2.2	2020-10-16	订正部分描述错误，增加使用 STM32 采集时硬件修改方式。