**模拟键盘鼠标三头USB模块使用说明书**

本模块采用CH9329芯片，通过USB端口方便连接电脑，实现模拟键盘、鼠标输入。可广泛应用于游戏挂机、游戏打币等二次元开发，减少游戏的检测和封号几率；也可用于升级下载、刷机板线等。（注：1、CH9329芯片支持即插即用，适用于Windows/Android/苹果等操作系统，完全免驱动，连接后即可使用。兼容安卓、苹果系统；完全免驱动）

**一、单USB口模拟键盘鼠标模块功能**

可实现模拟USB键盘和鼠标，将从主控机的USB口接收来的数据转换为键盘或鼠标格式的数据发给被控机。

模式0串口协议：

1. 模拟键盘把“A” 键按下，再释放：

主控机发送：57 AB 00 02 08 00 00 04 00 00 00 00 00 10

模块发送：57 AB 00 82 01 00 85

被控机接收到：“A”字母或者带A的快捷键界面功能。

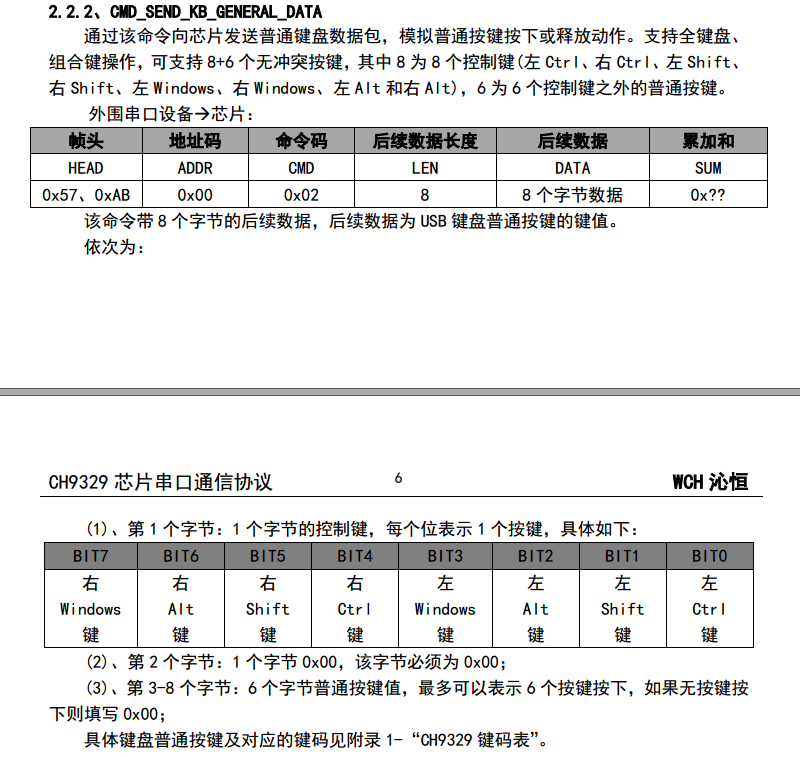
1. 模拟鼠标先按下“左”键，再释放：

主控机发送：57 AB 00 04 07 02 01 00 00 00 00 00 10 0x57 0xAB 0x00 0x04 0x07 0x02 0x01 0x00 0x00 0x00 0x00 0x00 0x10

模块发送：57 AB 00 84 01 00 87

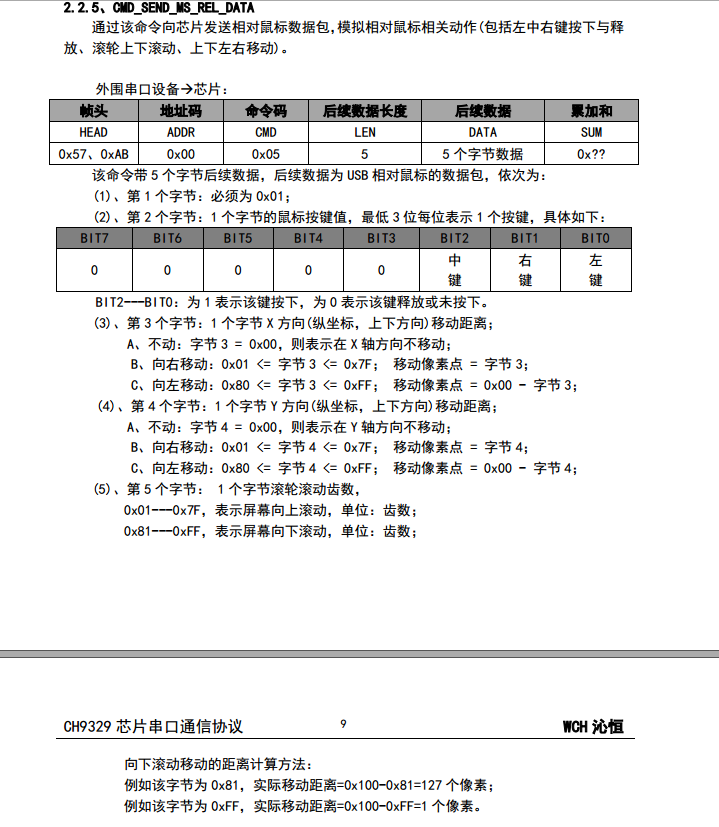
被控机接收到：单击“左”键带效果。

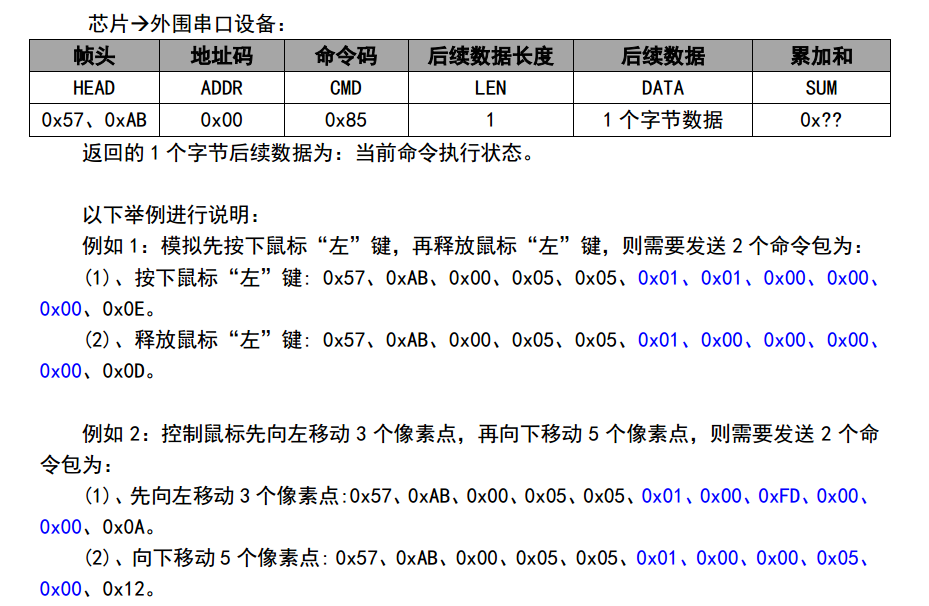
串口命令向模块发送普通键盘数据包，模拟普通按键按下或释放动作。 支持全键盘、组合键操作，可支持 8+6 个无冲突按键，其中 8 为 8 个控制键(左 Ctrl、 右 Ctrl、左 Shift、右 Shift、左 Windows、右 Windows、左 Alt 和右 Alt)， 6 为 6 个控制键之外的普通按键 。





命令向芯片发送绝对鼠标数据包，模拟绝对鼠标相关动作(包括左中右键按下与释放、滚轮上下滚动、上下左右移动)。



二、本模块可以通过跳帽短接模块上的端子二、本模块可以通过跳帽短接模块上的端子，从而可以选择芯片工作在您需要的模式。默认是工作在模式0、串口模式0（没跳帽短接任何一个排针）使用简单，可以直接通过串口往模块发特定命令，就可以模拟键盘、鼠标的输入！！二、本模块可以通过跳帽短接模块上的端子，从而可以选择芯片工作在您需要的模式。默认是工作在模式0、串口模式0（没跳帽短接任何一个排针）使用简单，可以直接通过串口往模块发特定命令，就可以模拟键盘、鼠标的输入！！

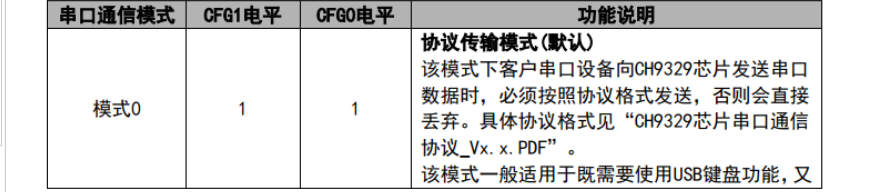
**二、模式选择**

要选模式时需要将相应模式的排针用跳帽短接起来，即把电平拉低。拨码开关从左到右分别为：MODE0、MODE1、CFG0、CFG1。其中MODE0、MODE1为工作模式选择，CFG0、CFG1为串口通信模式选择。

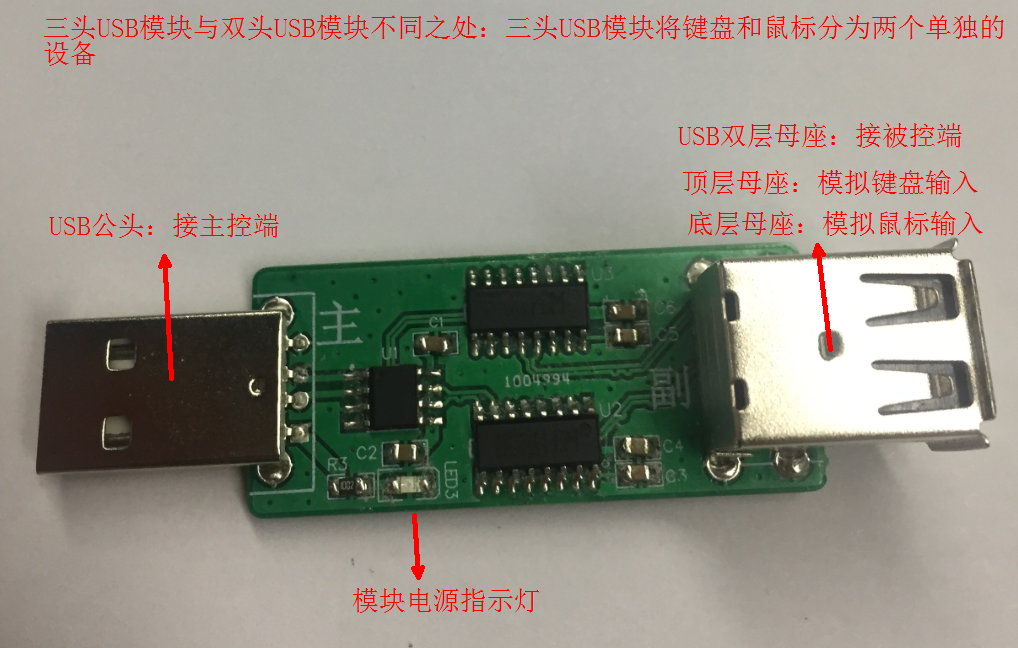
如果MODE0引脚和MODE1引脚悬空，则芯片默认处于工作模式0。

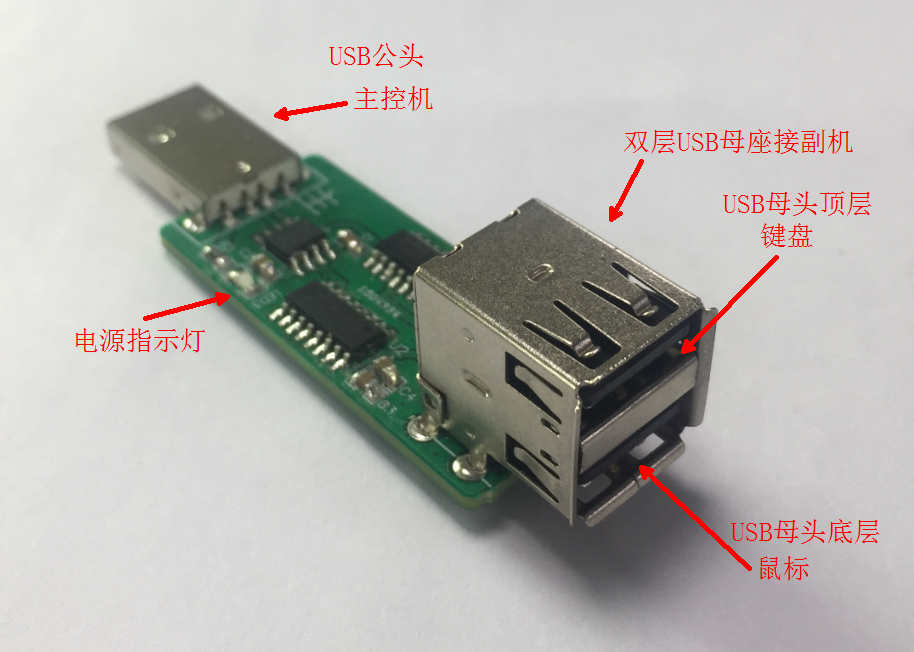


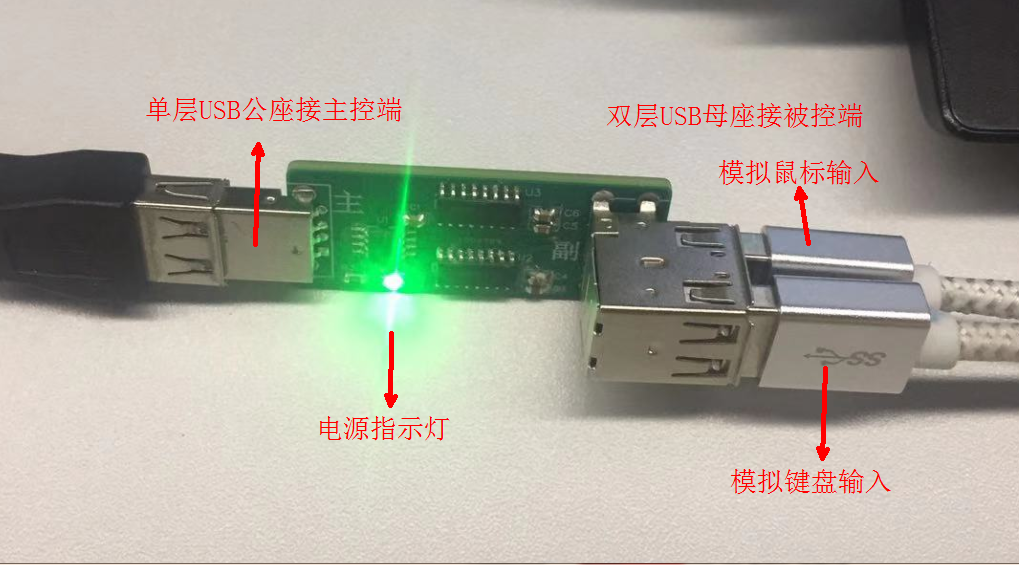
**如果CFG0引脚和CFG1引脚悬空，则芯片默认处于串口通信模式0。**













**附件：模拟鼠标键盘通信命令测试举例**

1、 CMD\_GET\_INFO------------------0x01-----------------获取芯片版本等信息

串口下传: 57 AB 00 01 00 03

芯片上传: 57 AB 00 81 08 30 01 03 00 00 00 00 00 BF

2、 CMD\_SEND\_KB\_GENERAL\_DATA------0x02-----------------发送USB键盘普通数据

(1)、模拟"A"键先按下,再释放

串口下传: 57 AB 00 02 08 00 00 04 00 00 00 00 00 10

芯片上传: 57 AB 00 82 01 00 85

串口下传: 57 AB 00 02 08 00 00 00 00 00 00 00 00 0C

芯片上传: 57 AB 00 82 01 00 85

(2)、模拟先同时按下“左Shift”+“A”键，再释放：

串口下传: 57 AB 00 02 08 02 00 04 00 00 00 00 00 12

芯片上传: 57 AB 00 82 01 00 85

串口下传: 57 AB 00 02 08 00 00 00 00 00 00 00 00 0C

芯片上传: 57 AB 00 82 01 00 85

3、 CMD\_SEND\_KB\_MEDIA\_DATA--------0x03-----------------发送USB键盘多媒体数据

(1)、多媒体按键(禁音)

串口下传: 57 AB 00 03 04 02 04 00 00 0F

芯片上传: 57 AB 00 83 01 00 86

串口下传: 57 AB 00 03 04 02 00 00 00 0B

芯片上传: 57 AB 00 83 01 00 86

4、 CMD\_SEND\_MS\_ABS\_DATA----------0x04-----------------发送USB绝对鼠标数据

(1)、模拟先按下鼠标“左”键，再释放鼠标“左”键，则需要发送2个命令包为：

串口下传: 57 AB 00 04 07 02 01 00 00 00 00 00 10

芯片上传: 57 AB 00 84 01 00 87

串口下传: 57 AB 00 04 07 02 00 00 00 00 00 00 0F

芯片上传: 57 AB 00 84 01 00 87

(2)、控制鼠标先移动到(100，100)位置，再移动到(968,500)位置，则需要发送2个命令包为：

根据计算机实际分辨率(1280 \* 768)计算位置:

计算位置X1 = ( 100 \* 4096 ) / 1280 = 320 = 0x140

计算位置Y1 = ( 100 \* 4096 ) / 768 = 533 = 0x215

串口下传: 57 AB 00 04 07 02 00 40 01 15 02 00 67

芯片上传: 57 AB 00 84 01 00 87

根据计算机实际分辨率(1280 \* 768)计算位置:

计算位置X1 = ( 968 \* 4096 ) / 1280 = 3097 = 0xC19

计算位置Y1 = ( 500 \* 4096 ) / 768 = 2667 = 0xA6B

串口下传: 57 AB 00 04 07 02 00 19 0C 6B 0A 00 A9

芯片上传: 57 AB 00 84 01 00 87

5、 CMD\_SEND\_MS\_REL\_DATA----------0x05-----------------发送USB相对鼠标数据

(1)、模拟先按下鼠标“左”键，再释放鼠标“左”键，则需要发送2个命令包为：

串口下传: 57 AB 00 05 05 01 01 00 00 00 0E

芯片上传: 57 AB 00 85 01 00 88

串口下传: 57 AB 00 05 05 01 00 00 00 00 0D

芯片上传: 57 AB 00 85 01 00 88

(2)、控制鼠标先向左移动3个像素点，再向下移动5个像素点，则需要发送2个命令包为：

串口下传: 57 AB 00 05 05 01 00 FD 00 00 0A

芯片上传: 57 AB 00 85 01 00 88

串口下传: 57 AB 00 05 05 01 00 00 05 00 12

芯片上传: 57 AB 00 85 01 00 88

6、 CMD\_SEND\_MY\_HID\_DATA----------0x06-----------------发送USB自定义HID设备数据

串口下传: 57 AB 00 06 10 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 90

芯片上传: 57 AB 00 85 01 00 88

7、 CMD\_READ\_MY\_HID\_DATA----------0x87-----------------读取USB自定义HID设备数据

如果计算机下传一包5个字节数据(0x12、0x34、0x56、0x78、0x90)，则：

芯片上传: 57 AB 00 87 05 12 34 56 78 90 32

8、 CMD\_GET\_PARA\_CFG--------------0x08-----------------获取参数配置

串口下传: 57 AB 00 08 00 0A

芯片上传: 57 AB 00 88 32 80 80 00 00 00 25 80 08 00 00 03 86 1A 29 E1 00 00 00 01 00 0D 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 2E

返回的值:

80--------------芯片工作模式(0x80表示由硬件引脚配置的模式0)

80--------------芯片串口通信模式(0x80表示由硬件引脚配置的模式0)

00--------------芯片串口通信地址

00 00 25 80-----芯片串口通信波特率

08 00-----------保留

00 03-----------芯片串口通信包间隔

86 1A 29 E1-----芯片USB的VID和PID

00 00-----------芯片USB键盘上传时间间隔

00 01-----------USB键盘释放延时时间

00--------------芯片USB键盘自动回车标志

0D 0A 00 00 00 00 00 00-----芯片USB键盘回车符

00 00 00 00 00 00 00 00-----芯片USB键盘过滤开始、结束字符串

00--------------芯片USB字符串使能标志

00 00 00 00 00 00 00 00 00 00 00 00 00----保留

9、 CMD\_SET\_PARA\_CFG--------------0x09-----------------设置参数配置

(1)、设置芯片工作模式3,其它不变

串口下传: 57 AB 00 09 32 03 00 00 00 00 25 80 08 00 00 03 86 1A 29 E1 00 00 00 01 00 0D 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 B2

芯片上传: 57 AB 00 89 01 00 8C

(2)、设置芯片工作模式3,串口通信波特率为115200

串口下传: 57 AB 00 09 32 03 00 00 00 01 C2 00 08 00 00 03 86 1A 29 E1 00 00 00 01 00 0D 0A 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 D0

芯片上传: 57 AB 00 89 01 00 8C

10、CMD\_GET\_USB\_STRING------------0x0A------------------获取字符串描述符配置

(1)、未设置过厂商字符串描述符

串口下传: 57 AB 00 0A 01 00 0D

芯片上传: 57 AB 00 8A 18 00 17 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF A5

11、CMD\_SET\_USB\_STRING------------0x0B-----------------设置字符串描述符配置

(1)、设置厂商字符串描述符: "0123456789"

串口下传: 57 AB 00 0B 0C 00 0A 30 31 32 33 34 35 36 37 38 39 30

芯片上传: 57 AB 00 8B 01 00 8E

12、CMD\_SET\_DEFAULT\_CFG-----------0x0C-----------------恢复出厂默认配置

串口下传: 57 AB 00 0C 00 0E

芯片上传: 57 AB 00 8C 01 00 8F

13、CMD\_JUMP\_TO\_ISP---------------0x0D-----------------跳转进入BOOT

串口下传: 57 AB 00 0D 00 0F

芯片上传: 57 AB 00 8D 01 00 90

14、CMD\_RESET---------------------0x0F-----------------复位芯片

串口下传: 57 AB 00 0F 00 11

芯片上传: 57 AB 00 8F 01 00 92

关于SUM累加和的理解：SUM = HEAD+ADDR+CMD+LEN+DATA

1. 鼠标释放: 57 AB 00 02 08 00 00 00 00 00 00 00 00 **0C**

SUM=57+AB+2+8=10C，然后只取低位十六进制数，即0C

2、模拟"A"键先按下:57 AB 00 02 08 00 00 04 00 00 00 00 00 **10**

SUM=57+AB+2+8+4=110，然后只取低位十六进制数，即10

1. 模拟“左Shift”+“A”键 : 57 AB 00 02 08 02 00 04 00 00 00 00 00 **12**

SUM=57+AB+2+8+2+4=112，然后只取低位十六进制数，即12