



矩阵控制协议

V2.0



矩阵RS232控制协议

1 协议简介

矩阵系统提供 RS-232通讯接口，用户可参考以下的通讯协议和控制代码，自行编写相应的控制软件，或在使用第三方控制系统来控制矩阵系统时，按以下的通讯协议和控制代码来设置所用的第三方控制系统通讯参数。

* 在用串口命令控制设备前，请仔细确认以下参数是否正确：

- 1) 波特率是否与控制设备一致；(开机LCD显示baud)
- 2) 设备地址是否与命令中的一致；(开机LCD显示DevID)
- 3) 确认校验和字节没有落掉，无论用户用或是不用校验，这个字节都不能少，若不用校验，该字节必须填0；
- 4) 确认串口线是交叉的，即第二针对第三针，第三针对第二针；

* 命令格式：

BAH(1) + 地址(2) + 命令(3) + 长度(4) + 切换模式(5) + 数据1…数据n(6) + 校验(7) 说明：

- 1) BAH字节表示帧起始，H表示“BA”为16进制数，BAH相当于10进制数的188；
- 2) 地址字节表示用户为切换器设定的地址；设备的地址在接口配置选项中由用户根据需要在(1-254)之间设定，主要用于设备的级连，文档中将用DevAddr来表示；FFH表示广播地址，设备会忽略地址接收广播地址的命令；
- 3) 命令字节表示本命令的功能和在命令集中的序列号；
- 4) 长度字节等于从本字节往后(不包括本字节)，到校验字节(包括校验字节)，所包括的字节个数；
- 5) 切换模式表示要切换的信号类型，也可视为不同的设备，见下表，其中AFH表示通配符，可切换任何设备，后面的指令范例均以VGA设备(A0H)为例；

命令	A0H	A1H	A2H	A3H	A4H	A5H	A6H	A7H	A8	AFH
类型	VGA	Video	Audio	VGA/Audio	AV	DVI	HDMI	SDI	混合	ALL

- 6) 数据段(数据1, …数据n)表示这条命令的数据部分，每条命令各不相同，之后详细解释；
- 7) 校验 = BAH + 地址 + 命令 + 长度 + 同异步模式字节 + 数1 + … + 数n(高位超过FFH溢出自动丢失)；

例如： BAH 01H 01H 04H A0H 00H 02H 63H

- 1) BAH为帧起始;
- 2) 01H表示设备地址;
- 3) 01H表示命令类型为切换;
- 4) 06H表示后面的字节长度(括号中的内容);
- 5) A0H为VGA矩阵
- 6) 00H 02H 02H 01H 为数据部分;
- 7) 68H为校验和 $68H = BAH + 01H + 01H + 06H + A0H + 00H + 02H + 02H + 01H$ 溢出位自动丢失。如果不用校验和直接写为00即可。

2 串口命令功能概述

- 1) 串行数据格式为:
10bit, 其中 8位数据, 1位起始位, 1位停止位, 无校验位。
- 2) 切换器在出厂时, 波特率被初始化为9600bps; 设备地址初始化为1。 本系统共有10条命令, 主要包括如下几方面的功能:
 - (1) 通道切换;
 - (2) 通道状态读取;
 - (3) 保存场景;
 - (4) 恢复场景;
 - (5) 场景循环号设置;
 - (6) 场景循环切换启动;
 - (7) 场景循环切换关闭;
 - (8) 场景循环切换时间设定;
 - (9) 系统参数查询;
 - (10) 设备地址修改。

3 通道切换指令

作用: 通道切换

格式: BAH, DevAddr, 01H, 长度, 切换模式, Out?, wIn?, Out?, wIn?, Out?, wIn? ..., 校验。

说明: Out?表示输出通道号, 用数(0 - n-1)表示实际通道号(1-n);
In?表示实际要选择的输入通道号(1 - n)。

例1: BAH 01H 01H 06H A0H 00H 02H 02H 01H 68H

功能: 把第3输入通道的信号切换到第1输出通道; 把第2输入通道切换到第3输出通道。

说明:

BAH: 帧起始。

01H: 设备的地址为1。

01H: 切换命令。

04H: 数据长度(Length)。

A0H: 表示要切换VGA设备(切换模式字节)。

00H: 设置第一输出通道。

02H: 把第三通道切换到前一个字节设定的输出通道(第一通道)。

63H: $63H = BAH + 01H + 01H + 04H + A0H + 00H + 02H$

4 通道状态读取指令

作用: 读取输出通道Out?, Out?, Out?...当前的状态,
In?, In?, In?... , 每次读取的通道数可以任意。

格式: BAH, DevAddr, 02H, 长度, A0H, Out?, Out?, Out?... , 校验。

说明: Out?表示用户要读取的输出通道号, 用数(0 - n-1)分别表示实际通道号(1 - n); 与前面的Out? 意义相同, In?表示前一指定输出通道的状态, 与第一条命令中wIn?的数据相同。

例: BAH 01H 02H 05H A0H 00H 01H 03H 42H

功能: 读取输出第一、第二、第四通道的切换状态。

说明: 前4个字节的意义和第一条命令相似。

A0H: 表示读取VGA设备通道的状态。

00H: 读取第一输出通道。

01H: 读取第二输出通道。

03H: 读取第四输出通道。

67H: $67H = BAH + 01H + 02H + 05H + A0H + 00H + 01H + 03H$

数据返回:

BAH, DevAddr, 02H, DevType, (Out?: In?, Out?: In?, ...) 校验。

例如返回: BAH 01H 02H 08H A0H 00H 00H 01H 01H 03H 05H 70H,
前面四个字节和前面的类似:

A0H 表示是VGA矩阵。

00H 00H 表示现在第1输出通道切换到第1输入通道。

01H 01H 表示现在第2输出通道切换到第2输入通道。

03H 05H 表示现在第4输出通道切换到第6输入通道。

5 场景存储指令

作用: 把当前的切换状态保存为场景, 切换器可以为用户保存16个场景。

格式: BAH, DevAddr, 11H, 长度, 00H, 场景号, 校验。

场景号表示把当前的状态存在哪个档案里, 有效数据为(1—16), 分别表示16个存档。

例1: BAH 01H 11H 03H 00H 01H D0H

说明: 前4个字节的意义和第一条命令相似。

00H: 本字节在这条命令中无任何用处, 规定为0。

01H: 表示存档号。

D0H: $D0H = BAH + 01H + 11H + 03H + 00H + 01H$

功能: 把当前的状态存到场景1中。

例2: BAH 01H 11H 03H 00H 02H D1H

功能: 把当前状态保存到场景2中。

例3: BAH 01H 11H 03H 00H 03H D2H

功能: 把当前状态保存到场景3中。

6 场景恢复指令

作用: 恢复已保存的场景。

格式: BAH, DevAddr, 15H, 长度, 00H, 场景号, 校验。

例: BAH 01H 15H 03H 00H 03H D6H

说明: 前4个字节的意义和第一条命令相似。

00H: 本字节在这条命令中无任何用处, 规定为0。

03H: 表示存档号。

D6H: $D6H = BAH + 01H + 15H + 03H + 00H + 03H$

功能: 把设备恢复到场景3的切换状态。

7 场景循环号设定

作用: 设定要循环切换的场景号。

格式: BAH, DevAddr, 19H, 长度, 00, BYTE2, BYTE1, 校验。

要循环切换的场景号一共2个字节, 每位代表一个场景, 2个字节共16位代表16个场景, 如果某位置“1”表示该场景参与循环切换, “0”该场景不参与切换。

BYTE1:

场景8	场景7	场景6	场景5	场景4	场景3	场景2	场景1
-----	-----	-----	-----	-----	-----	-----	-----

BYTE2:

场景16	场景15	场景14	场景13	场景12	场景11	场景10	场景9
------	------	------	------	------	------	------	-----

例: BAH 01H 19H 04H 00H 00H 0FH E7H

说明: 前4个字节的意义和第一条命令相似。

00H: 本字节规定为0。

00H: 表示9-16场景不参与切换。

0FH: 表示1-4场景参与循环切换, 5-8场景不参与切换。

E7H: $= BAH + 01H + 19H + 04H + 00H + 00H + 0FH$

功能: 场景循环切换设定。

8 场景循环切换启动指令

作用：启动已保存的场景按指定的时间循环切换。

格式：BAH, DevAddr, 1AH, 长度, 00H, 00, 校验。

例：BAH 01H 1AH 03H 00H 00H D8H

说明：前4个字节的意义和第一条命令相似。

00H：本字节规定为0。

00H：表示启动循环切换。

D8H：= BAH + 01H + 1AH + 03H + 00H + 00H

功能：启动场景循环切换。

9 场景循环切换停止指令

作用：停止场景循环切换。

格式：BAH, DevAddr, 1AH, 长度, 00H, FF, 校验。

例：BAH 01H 1AH 03H 00H FFH D8H

说明：前4个字节的意义和第一条命令相似。

00H：本字节在这条命令中无任何用处，规定为0。

FFH：表示停止循环切换。

D8H：D7H = BAH + 01H + 1AH + 03H + 00H + FFH

功能：停止场景循环切换。

10 场景循环切换间隔时间设定指令

作用：设定已保存的场景循环切换间隔时间。

格式：BAH, DevAddr, 1BH, 长度(Length), 00H, 时间, 校验。

例：BAH 01H 1BH 03H 00H 03H DCH

说明：前4个字节的意义和第一条命令相似。

00H：本字节在这条命令中无任何用处，规定为0。

03H：表示循环切换间隔时间为3秒，最大FF为255秒。

DCH：DCH = BAH + 01H + 1BH + 03H + 00H + 03H

功能：设定场景循环切换时间为3秒。

11 系统参数查询指令

作用：查询设备参数，如设备的类型，规模及地址号。

格式：BAH, DevAddr, 14H, 长度, 00, 校验。

例：BAH 01H 14H 02H 00H D1H

说明：前4个字节的意义和第一条命令相似。

00H：本字节在这条命令中无任何用处，规定为0。

D1H：D1H = D1H + 01H + 14H + 02H + 00H

功能：查询当前设备的参数。返回设备类型，输入输出通道数量。

返回：BAH, DevAddr, 14H, 长度, DevType, IN_NUM, OUT_NUM, 校验。

例如返回：BAH 01H A0H 03H 0FH 0FH 7D

A0H: 表示是VGA矩阵。

03H: 表示长度。

0FH: 表示输入通道数是16路。

0FH: 表示输出通道数是16路。

12 设备地址修改指令

作用：修改设备的地址。

格式：BAH, DevAddr, 16H, 长度, 00, AddrNew, 校验。

例：BAH 01H 16H 03H 00H 03H D7H

说明：前4个字节的意义和第一条命令相似。

00H: 本字节在这条命令中无任何用处，规定为0。

03H: 表示AddrNew指定的地址号。

D7H: $D7H = BAH + 01H + 16H + 03H + 00H + 03H$

功能：把当前设备的地址修改为3。

如果设备有按键，可通过按键修改设备地址，但地址号不能多大于8，先按住按键，再打开设备电源即可，前面板液晶会显示设备地址。

13 设备波特率修改指令

作用：修改设备的波特率。

格式：BAH, DevAddr, 18H, 长度, 00, BaudNew, 校验。

Baud: 00:9600

01:4800

02:2400

03:19200

例：BAH 01H 18H 03H 00H 02H D8

说明：前4个字节的意义和第一条命令相似。

00H: 本字节在这条命令中无任何用处，规定为0。

02H: 表示BaudNew指定的波特率。

D8H: $D8H = BAH + 01H + 18H + 03H + 00H + 02H$ 。

功能：把当前设备的波特率修改为2400。

注：HDMI16x16, DVI16x16以上矩阵不能修改波特率。

14 串口命令返回说明

1) 如果命令执行正确，将返回相应信息（BAH, DevAddr, 命令序列号, 长度, 返回关键字信息, 校验和）。

2) 关键字信息只有是55h表示命令执行成功，其他均表示失败，失败

原因见下面说明：

55H: 成功执行

FFH: 表示保留命令，可能是以前的设备具有的命令，而现在已逐步更新掉了。目的在于提醒用户，是不是还是用了老的命令在控制新的机器，如果是这样，请用户参考本公司为设备新配的说明书，或咨询我们公司的技术人员；

01H: 切换命令太长，原因是切换命令中指定要切换的通道数多于矩阵的规模，从而提醒用户命令已发生错误；

02H: 输出通道数过大，原因是指令中的某指定输出通道大于本矩阵的最大输出通道数，从而提醒用户命令已发生错误；

03H: 输入通道数过大，原因是指令中的某指定输入通道大于本矩阵的最大输入通道数，从而提醒用户命令已发生错误；

04H: 是扩展命令系列的指令，这些指令只是针对大矩阵才起用的（输入输出的规模大于64）；

05H: 存档数和取档数不在范围中，本系统提供最大的存档数和取档数为16，超出这个范围的数被认为是非法值。